SIEMENS

SINUMERIK

SINUMERIK 840D sl Программирование ЧПУ

Руководство по программированию

Введение 1

Основные указания по безопасности 2

Основная информация 3

Расширенное программирование 4

Таблицы 5

Приложение A

Действительно для

системы ЧПУ
SINUMERIK 840D sl / 840DE sl

ПО для системы ЧПУ версия 4.95

07/2021
6FC5398-2EP40-0PA2

Промэлектроника, sales@prom-elec.com

Правовая справочная информация

Система предупреждений

Данная инструкция содержит указания, которые Вы должны соблюдать для Вашей личной безопасности и для предотвращения материального ущерба. Указания по Вашей личной безопасности выделены предупреждающим треугольником, общие указания по предотвращению материального ущерба не имеют этого треугольника. В зависимости от степени опасности, предупреждающие указания представляются в убывающей последовательности следующим образом:

⚠ ОПАСНО

означает, что непринятие соответствующих мер предосторожности **приводит** к смерти или получению тяжелых телесных повреждений.

⚠ ПРЕДУПРЕЖДЕНИЕ

означает, что непринятие соответствующих мер предосторожности **может** привести к смерти или получению тяжелых телесных повреждений.

⚠ ОСТОРОЖНО

означает, что непринятие соответствующих мер предосторожности может привести к получению незначительных телесных повреждений.

ВНИМАНИЕ

означает, что непринятие соответствующих мер предосторожности может привести к материальному ущербу.

При возникновении нескольких степеней опасности всегда используется предупреждающее указание, относящееся к наивысшей степени. Если в предупреждении с предупреждающим треугольником речь идет о предупреждении ущерба, причиняемому людям, то в этом же предупреждении дополнительно могут иметься указания о предупреждении материального ущерба.

Квалифицированный персонал

Работать с изделием или системой, описываемой в данной документации, должен только **квалифицированный персонал**, допущенный для выполнения поставленных задач и соблюдающий соответствующие указания документации, в частности, указания и предупреждения по технике безопасности. Квалифицированный персонал в силу своих знаний и опыта в состоянии распознать риски при обращении с данными изделиями или системами и избежать возникающих угроз.

Использование изделий Siemens по назначению

Соблюдайте следующее:

⚠ ПРЕДУПРЕЖДЕНИЕ

Изделия Siemens разрешается использовать только для целей, указанных в каталоге и в соответствующей технической документации. Если предполагается использовать изделия и компоненты других производителей, то обязательным является получение рекомендации и/или разрешения на это от фирмы Siemens. Исходными условиями для безупречной и надежной работы изделий являются надлежащая транспортировка, хранение, размещение, монтаж, оснащение, ввод в эксплуатацию, обслуживание и поддержание в исправном состоянии. Необходимо соблюдать допустимые условия окружающей среды. Обязательно учитывайте указания в соответствующей документации.

Товарные знаки

Все наименования, обозначенные символом защищенных авторских прав ®, являются зарегистрированными товарными знаками компании Siemens AG. Другие наименования в данной документации могут быть товарные знаки, использование которых третьими лицами для их целей могут нарушать права владельцев.

Исключение ответственности

Мы проверили содержимое документации на соответствие с описанным аппаратным и программным обеспечением. Тем не менее, отклонения не могут быть исключены, в связи с чем мы не гарантируем полное соответствие. Данные в этой документации регулярно проверяются и соответствующие корректуры вносятся в последующие издания.

Siemens AG
Digital Industries
Postfach 48 48
90026 NÜRNBERG
ГЕРМАНИЯ

Номер для заказа документа: 6FC5398-2EP40-0PA2
Ⓟ 11/2021 Возможны изменения

Copyright © Siemens AG 2019 - 2021.
Все права защищены

Оглавление

1 Введение ... 19

1.1 Информация о SINUMERIK ... 19

1.2 Информация о данной документации ... 19

1.3 Документация в Интернете ... 22
1.3.1 Обзор документации SINUMERIK 840D sl ... 22
1.3.2 Обзор документации для компонентов управления системы SINUMERIK ... 22

1.4 Комментарии, касающиеся технической документации ... 23

1.5 mySupport-Dokumentation ... 23

1.6 Сервис и поддержка ... 24

1.7 Важная информация об изделии ... 26

2 Основные указания по безопасности ... 27

2.1 Общие указания по безопасности ... 27

2.2 Гарантийные обязательства и ответственность за прикладные примеры ... 27

2.3 Информация о безопасности ... 28

3 Основная информация ... 31

3.1 Геометрические основы ... 31
3.1.1 Позиции детали ... 31
3.1.1.1 Исходная система указаний позиций ... 31
3.1.1.2 Декартовы координаты ... 31
3.1.1.3 Полярные координаты ... 33
3.1.1.4 Абсолютный размер ... 34
3.1.1.5 Составной размер ... 36
3.1.2 Рабочие плоскости ... 37
3.1.3 Нулевые точки и исходные точки ... 38
3.1.4 Системы координат ... 40
3.1.4.1 Система координат станка (MCS) ... 40
3.1.4.2 Базовая кинематическая система (BKS) ... 42
3.1.4.3 Базовая система нулевой точки (BNS) ... 43
3.1.4.4 Настраиваемая система нулевой точки (ENS) ... 43
3.1.4.5 Система координат детали (WCS) ... 44
3.1.4.6 Как связаны различные системы координат? ... 45

3.2 Основы программирования ЧПУ ... 45
3.2.1 Наименование программы ЧПУ ... 46
3.2.2 Структура и содержание программы ЧПУ ... 47
3.2.2.1 Кадры и компоненты кадров ... 47
3.2.2.2 Правила для кадра ... 50
3.2.2.3 Присвоения значений ... 51
3.2.2.4 Комментарии ... 51
3.2.2.5 Пропуск кадров ... 52

3.3 Создание программы ЧПУ ... 54
3.3.1 Базовый принцип ... 54
3.3.2 Доступные символы ... 56
3.3.3 "Шапка" программы ... 57
3.3.4 Примеры программы ... 58
3.3.4.1 Пример 1: Первые шаги программирования ... 58
3.3.4.2 Пример 2: Программа ЧПУ для токарной обработки ... 59
3.3.4.3 Пример 3: Программа ЧПУ для фрезерной обработки ... 60

3.4 Смена инструмента ... 63
3.4.1 Смена инструмента при активном управлении инструментом ... 64
3.4.1.1 Смена инструмента при активном управлении инструментом с функцией T=место ... 64
3.4.1.2 Смена инструмента при активном управлении инструментом с M6 ... 67
3.4.2 Смена инструмента при неактивном управлении инструментом ... 68
3.4.2.1 Смена инструмента при неактивном управлении инструментом с номером T ... 68
3.4.2.2 Смена инструмента при неактивном управлении инструментом с M6 ... 69
3.4.3 Поведение при неправильном программировании T ... 70

3.5 Коррекции инструмента ... 70
3.5.1 Запрограммированный контур и траектория инструмента ... 70
3.5.2 Коррекция длин инструмента ... 71
3.5.3 Коррекция радиуса инструмента ... 72
3.5.4 Память коррекций инструмента ... 73
3.5.5 Типы инструментов ... 74
3.5.5.1 Номер типа инструмента и группы инструментов ... 74
3.5.5.2 Фрезерный инструмент ... 75
3.5.5.3 Сверло ... 77
3.5.5.4 Шлифовальный инструмент ... 78
3.5.5.5 Токарный инструмент ... 80
3.5.5.6 Специальный инструмент ... 82
3.5.6 Активация/деактивация коррекции на инструмент (D, D0) ... 84
3.5.7 Подавление коррекции инструмента (SUPD) ... 86
3.5.8 Программируемое смещение коррекции инструмента (TOFFL, TOFF, TOFFR, TOFFLR): ... 88

3.6 Движение шпинделя ... 93
3.6.1 Число оборотов шпинделя (S), направление вращения шпинделя (M3, M4, M5) ... 93
3.6.2 Скорость резания инструмента (SVC) ... 97
3.6.3 Постоянная скорость резания (G96/G961/G962, G97/G971/G972, G973, LIMS, SCC) ... 103
3.6.4 Включение / выключение постоянной окружной скорости круга (GWPSON, GWPSOF) ... 108
3.6.5 Программируемое ограничение числа оборотов шпинделя (G25, G26) ... 109

3.7 Регулирование подачи ... 110
3.7.1 Подача (G93, G94, G95, F, FGROUP, FL, FGREF) ... 110
3.7.2 Перемещение позиционирующих осей (POS, POSA, POSP, FA, WAITP, WAITMC) ... 118
3.7.3 Режим ориентации шпинделя (SPCON, SPCOF) ... 122
3.7.4 Позиционирование шпинделей (SPOS, SPOSA, M19, M70, WAITS) ... 123
3.7.5 Подача для позиционирующих осей/шпинделей (FA, FPR, FPRAON, FPRAOF) ... 129
3.7.6 Программируемая коррекция подачи (OVR, OVRRAP, OVRA) ... 132
3.7.7 Программируемая коррекция ускорения (ACC) ... 133
3.7.8 Подача с наложением маховичка (FD, FDA) ... 135
3.7.9 Оптимизация подачи для изогнутых участков траектории (CFTCP, CFC, CFIN) ... 138
3.7.10 Несколько значений подачи в одном кадре (F, ST, SR, FMA, STA, SRA) ... 141

3.7.11 Покадровая подача (FB) ... 144
3.7.12 Подача на зуб (G95 FZ) ... 145

3.8 Геометрические установки ... 150
3.8.1 Устанавливаемое смещение нулевой точки (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153) ... 150
3.8.2 Устанавливаемое смещение нулевой точки (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153): Дополнительная информация ... 152
3.8.3 Выбор рабочей плоскости (G17/G18/G19) ... 153
3.8.4 Указание размеров ... 156
3.8.4.1 Указание абсолютного размера (G90, AC) ... 157
3.8.4.2 Указание составного размера (G91, IC) ... 159
3.8.4.3 Указание абсолютного и составного размера при токарной обработке и фрезеровании (G90/G91) ... 163
3.8.4.4 Указание абсолютного размера для круговых осей (DC, ACP, ACN) ... 164
3.8.4.5 Дюймовая/метрическая система единиц (G70/G71, G700/G710) ... 166
3.8.4.6 Специфическое для канала программирование диаметра/радиуса (DIAMON, DIAM90, DIAMOF, DIAMCYCOF) ... 170
3.8.4.7 Специфическое для оси программирование диаметра/радиуса (DIAMONA, DIAM90A, DIAMOFA, DIACYCOFA, DIAMCHANA, DIAMCHAN, DAC, DIC, RAC, RIC) ... 172
3.8.5 Положение детали при токарной обработке ... 176

3.9 Команды перемещения ... 177
3.9.1 Общая информация по командам перемещения ... 177
3.9.2 Команды движения с декартовыми координатами (G0, G1, G2, G3, X..., Y..., Z...) ... 179
3.9.3 Команды движения с полярными координатами ... 181
3.9.3.1 Исходная точка полярных координат (G110, G111, G112) ... 181
3.9.3.2 Команды движения с полярными координатами (G0, G1, G2, G3, AP, RP) ... 182
3.9.4 Движения ускоренного хода ... 186
3.9.4.1 Активация ускоренного хода (G0) ... 186
3.9.4.2 Включение/выключение линейной интерполяции для движений ускоренным ходом (RTLION, RTLIOF) ... 188
3.9.4.3 Коррекция допусков для движений ускоренным ходом (STOLF, CTOLG0, OTOLG0) ... 190
3.9.5 Линейная интерполяция (G1) ... 194
3.9.6 Круговая интерполяция ... 197
3.9.6.1 Обзор ... 197
3.9.6.2 Круговая интерполяция с центром и конечной точкой (G2/G3, X... Y... Z..., I... J... K...) ... 198
3.9.6.3 Круговая интерполяция с радиусом и конечной точкой (G2/G3, X... Y... Z..., CR) ... 200
3.9.6.4 Круговая интерполяция с апертурным углом и конечной точкой / центром (G2/G3, X... Y... Z... / I... J... K..., AR) ... 202
3.9.6.5 Круговая интерполяция с полярными координатами (G2/G3, AP, RP) ... 204
3.9.6.6 Круговая интерполяция с промежуточной и конечной точкой (CIP, X... Y... Z..., I1... J1... K1...) ... 206
3.9.6.7 Круговая интерполяция с тангенциальным переходом (CT, X... Y... Z...) ... 208
3.9.7 Винтовая интерполяция (G2/G3, TURN) ... 212
3.9.8 Эвольвентная интерполяция (INVCW, INVCCW) ... 214
3.9.9 Линии контура ... 219
3.9.9.1 Программирование линии контура ... 219
3.9.9.2 Линии контура: Прямая ... 220
3.9.9.3 Линии контура: Две прямые ... 222
3.9.9.4 Линии контура: Три прямые ... 225
3.9.9.5 Линии контура: Программирование конечной точки с углом ... 229
3.9.10 Резьбонарезание ... 229

3.9.10.1 Резьбонарезание с постоянным шагом (G33, SF) ... 229
3.9.10.2 Резьбонарезание с увеличивающимся или уменьшающимся шагом (G34, G35) ... 236
3.9.10.3 Запрограммированный входной и выходной участок при G33, G34, G35 (DITS, DITE) ... 237
3.9.10.4 Быстрый обратный ход при резьбонарезании (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN) ... 239
3.9.10.5 Сферическая резьба (G335, G336) ... 243
3.9.11 Нарезание внутренней резьбы без компенсирующего патрона ... 248
3.9.11.1 Нарезание внутренней резьбы без компенсирующего патрона и движение отвода (G331, G332) ... 248
3.9.11.2 Пример: Нарезание внутренней резьбы с G331 / G332 ... 250
3.9.11.3 Пример: Вывод запрограммированной скорости сверления на текущей ступени редуктора ... 250
3.9.11.4 Пример: Использование второго блока данных ступеней редуктора ... 251
3.9.11.5 Пример: Скорость не программируется, контроль ступени редуктора ... 251
3.9.11.6 Пример: Смена ступеней редуктора невозможна, контроль ступени редуктора ... 251
3.9.11.7 Пример: Программирование без SPOS ... 252
3.9.12 Нарезание внутренней резьбы с компенсирующим патроном ... 253
3.9.12.1 ; Нарезание внутренней резьбы с компенсирующим патроном и движением отвода (G63) ... 253
3.9.13 Фаска, закругление (CHF, CHR, RND, RNDM, FRC, FRCM) ... 254

3.10 Коррекции радиуса инструмента ... 260
3.10.1 Коррекция радиуса инструмента (G40, G41, G42, OFFN) ... 260
3.10.2 Подвод к контуру и отвод (NORM, KONT, KONTC, KONTT) ... 269
3.10.3 Коррекция на наружных углах (G450, G451, DISC) ... 276
3.10.4 Мягкий подвод и отвод ... 279
3.10.4.1 Подвод и отвод (G140 до G143, G147, G148, G247, G248, G347, G348, G340, G341, DISR, DISCL, DISRP, FAD, PM, PR) ... 279
3.10.4.2 Подвод и отвод с расширенными стратегиями отвода (G460, G461, G462) ... 290
3.10.5 Включение/выключение контроля столкновений ("Распознавание бутылочного горлышка") (CDON, CDOF, CDOF2) ... 294
3.10.6 Коррекция инструмента 2 1/2 D (CUT2D, CUT2DD, CUT2DF, CUT2DFD) ... 295
3.10.7 Постоянная коррекция радиуса инструмента (CUTCONON, CUTCONOF) ... 298
3.10.8 Инструменты с релевантным положением резцов ... 300

3.11 Параметры движения по траектории ... 302
3.11.1 Точный останов (G60, G9, G601, G602, G603) ... 302
3.11.2 Режим управления траекторией (G64, G641, G642, G643, G644, G645, ADIS, ADISPOS) ... 305

3.12 Трансформации координат (фреймы) ... 314
3.12.1 Фреймы ... 314
3.12.2 Фрейм-операторы ... 316
3.12.3 программируемое смещение нулевой точки (TRANS, ATRANS) ... 319
3.12.4 Программируемое смещение нулевой точки (G58, G59) ... 323
3.12.5 Программируемое вращение (ROT, AROT, RPL) ... 325
3.12.6 Программируемые вращения фреймов с пространственными углами (ROTS, AROTS, CROTS) ... 331
3.12.7 Программируемый коэффициент масштабирования (SCALE, ASCALE) ... 334
3.12.8 Программируемое отражение (MIRROR, AMIRROR) ... 337
3.12.9 Создание фрейма по точной установке инструмента (TOFRAME, TOROT, PAROT) ... 342
3.12.10 Отключение фрейма (G53, G153, SUPA, G500) ... 345
3.12.11 Программирование: Отключение наложений (спец. для оси) (CORROF) ... 346
3.12.12 Отключение аддитивных смещений нулевой точки (DRFOF) ... 348

3.12.13 Специфическое для шлифования смещение нулевой точки (GFRAME0, GFRAME1 ... GFRAME100) ... 350

3.13 Вывод вспомогательных функций ... 351
3.13.1 Функции M ... 354

3.14 Дополнительные команды ... 357
3.14.1 Сообщения (MSG) ... 357
3.14.2 Запись строки символов в переменную BTSS (WRTPR) ... 358
3.14.3 Ограничение рабочего поля ... 360
3.14.3.1 Ограничение рабочего поля в BKS (G25/G26, WALIMON, WALIMOF) ... 360
3.14.3.2 Ограничение рабочего поля в WCS/ENS (WALCS0 ... WALCS10) ... 363
3.14.4 Реферирование (G74) ... 366
3.14.5 Движение к фиксированной точке (G75) ... 367
3.14.6 Наезд на жесткий упор (FXS, FXST, FXSW) ... 371
3.14.7 Время ожидания (G4) ... 376
3.14.8 Внутренняя остановка предварительной обработки ... 378

3.15 Прочая информация ... 378
3.15.1 Оси ... 378
3.15.1.1 Главные оси/геометрические оси ... 379
3.15.1.2 Дополнительные оси ... 380
3.15.1.3 Главный шпиндель, мастер-шпиндель ... 381
3.15.1.4 Оси станка ... 381
3.15.1.5 Оси канала ... 382
3.15.1.6 Траекторные оси ... 382
3.15.1.7 Позиционирующие оси ... 382
3.15.1.8 Синхронные оси ... 383
3.15.1.9 Командные оси ... 383
3.15.1.10 Оси PLC ... 384
3.15.1.11 Оси Link ... 384
3.15.1.12 Оси Lead-Link ... 386
3.15.2 От команды движения до движения станка ... 388
3.15.3 Вычисление хода ... 388
3.15.4 Адреса ... 389
3.15.5 Имя ... 391
3.15.6 Постоянные ... 393
3.15.7 Операторы и функции вычисления ... 394

4 Расширенное программирование ... 399

4.1 Гибкое программирование ЧПУ ... 399
4.1.1 Переменные ... 399
4.1.1.1 Системные данные ... 400
4.1.1.2 Предопределенные переменные пользователя: Специфические для канала R-параметры (R) ... 403
4.1.1.3 Предопределенные переменные пользователя: Глобальные R-параметры (RG) ... 404
4.1.1.4 Предопределенные переменные пользователя: Link-переменные ... 406
4.1.1.5 Определение переменных пользователя (DEF) ... 408
4.1.1.6 Переопределение системных данных, пользовательских данных и языковых команд ЧПУ (REDEF) ... 414
4.1.1.7 Атрибут: Инициализирующее значение ... 418
4.1.1.8 Атрибут: предельные значения (LLI, ULI) ... 421
4.1.1.9 Атрибут: Физическая единица (PHU) ... 422
4.1.1.10 Атрибут: Права доступа (APR, APW, APRP, APWP, APRB, APWB) ... 425

4.1.1.11 Обзор определяемых и переопределяемых атрибутов ... 429
4.1.1.12 Определение и инициализация переменных поля (DEF, SET, REP) ... 431
4.1.1.13 Типы данных ... 437
4.1.1.14 Минимум, максимум и диапазон переменных (MINVAL, MAXVAL и BOUND) ... 437
4.1.1.15 Проверка наличия переменной (ISVAR) ... 439
4.1.1.16 Чтение значений атрибута / типа данных (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP) ... 441
4.1.1.17 Возможные преобразования типов ... 446
4.1.2 Косвенное программирование ... 447
4.1.2.1 Косвенное программирование адресов ... 447
4.1.2.2 Косвенное программирование G-команд ... 449
4.1.2.3 Косвенное программирование атрибутов позиций (GP) ... 450
4.1.2.4 Косвенное программирование строк программы обработки детали (EXECSTRING) ... 453
4.1.3 Операции ... 454
4.1.3.1 Функции вычисления ... 454
4.1.3.2 Операторы сравнения и логические операторы ... 456
4.1.3.3 Приоритет операций ... 458
4.1.3.4 Коррекция точности при ошибках сравнения (TRUNC) ... 459
4.1.3.5 Округления с избытком (ROUNDUP) ... 460
4.1.4 Операции со строкой ... 461
4.1.4.1 Преобразование типов в STRING (AXSTRING) ... 462
4.1.4.2 Преобразование типов из STRING (NUMBER, ISNUMBER, AXNAME) ... 463
4.1.4.3 Соединение строк (<<) ... 464
4.1.4.4 Преобразование в строчные/прописные буквы (TOLOWER, TOUPPER) ... 465
4.1.4.5 Определение длины строки (STRLEN) ... 466
4.1.4.6 Поиск символа/строки в строке (INDEX, RINDEX, MINDEX, MATCH) ... 466
4.1.4.7 Выбор части строки (SUBSTR) ... 467
4.1.4.8 Чтение и запись отдельных символов ... 468
4.1.4.9 Форматирование строки (SPRINT) ... 469
4.1.5 Переходы и ветвления в программе ... 478
4.1.5.1 Возврат на начало программы (GOTOS) ... 478
4.1.5.2 Переходы на метки перехода (GOTOB, GOTOF, GOTO, GOTOC) ... 479
4.1.5.3 Ветвление программы (CASE ... OF ... DEFAULT ...) ... 483
4.1.6 Повторение блока программы (REPEAT, REPEATB, ENDLABEL, P) ... 485
4.1.7 Управляющие структуры ... 491
4.1.7.1 Условный оператора и ветвление (IF, ELSE, ENDIF) ... 493
4.1.7.2 Бесконечный программный цикл (LOOP, ENDLOOP) ... 495
4.1.7.3 Счетный цикл (FOR ... TO ..., ENDFOR) ... 495
4.1.7.4 Программный цикл с условием в начале цикла (WHILE, ENDWHILE) ... 497
4.1.7.5 Программный цикл с условием на конце цикла (REPEAT, UNTIL) ... 497
4.1.7.6 Пример программы со вложенными управляющими структурами ... 498
4.1.8 Выходящая за рамки канала координация программ (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) ... 499
4.1.9 Техника макросов (DEFINE ... AS) ... 505

4.2 Техника подпрограмм ... 507
4.2.1 Общая информация ... 507
4.2.1.1 Подпрограмма ... 507
4.2.1.2 Имена подпрограмм ... 509
4.2.1.3 Вложенность подпрограмм ... 510
4.2.1.4 Маршрут поиска ... 511
4.2.1.5 Формальные и фактические параметры ... 512
4.2.1.6 Передача параметров ... 512

4.2.2 Определение подпрограммы ... 514
4.2.2.1 Подпрограмма без передачи параметров ... 514
4.2.2.2 Подпрограмма с передачей параметров по значению (PROC) ... 515
4.2.2.3 Подпрограмма с передачей параметров по ссылке (PROC, VAR) ... 517
4.2.2.4 Сохранить модальные функции G (SAVE) ... 519
4.2.2.5 Подавление покадровой обработки (SBLOF, SBLON) ... 520
4.2.2.6 Подавление актуальной индикации кадра (DISPLOF, DISPLON, ACTBLOCNO) ... 525
4.2.2.7 Обозначить подпрограммы с подготовкой (PREPRO) ... 528
4.2.2.8 Возврат из подпрограммы M17 ... 529
4.2.2.9 Возврат из подпрограммы RET ... 530
4.2.2.10 Параметрируемый возврат из подпрограммы (RET ...) ... 531
4.2.2.11 Параметрируемый возврат из подпрограммы (RETB ...) ... 538
4.2.3 Вызов подпрограммы ... 542
4.2.3.1 Вызовы подпрограмм без передачи параметров ... 542
4.2.3.2 Вызов подпрограммы с передачей параметров (EXTERN) ... 544
4.2.3.3 Число повторений программы (P) ... 546
4.2.3.4 Модальный вызов подпрограммы (MCALL) ... 548
4.2.3.5 Косвенный вызов подпрограммы (CALL) ... 550
4.2.3.6 Косвенный вызов подпрограммы с указанием выполняемого программного блока (CALL BLOCK ... TO ...) ... 551
4.2.3.7 Косвенный вызов запрограммированной на языке ISO программы (ISOCALL) ... 552
4.2.3.8 Вызов подпрограммы с указанием пути и параметрами (PCALL) ... 553
4.2.3.9 Расширить маршрут поиска для вызовов подпрограмм (CALLPATH) ... 553
4.2.3.10 Выполнение внешней подпрограммы (EXTCALL) ... 555

4.3 Обработчик прерываний (ASUP) ... 559
4.3.1 Функция обработчика прерываний ... 559
4.3.2 Создание обработчика прерываний ... 560
4.3.3 Согласование и запуск обработчика прерываний (SETINT, PRIO, BLSYNC) ... 561
4.3.4 Деактивация/повторная активация согласования обработчика прерываний (DISABLE, ENABLE) ... 563
4.3.5 Удаление согласования обработчика прерываний (CLRINT) ... 563
4.3.6 Быстрый отвод от контура (SETINT LIFTFAST, ALF) ... 564
4.3.7 Направление перемещения при быстром отводе от контура ... 567
4.3.8 Процесс движения для обработчиков прерываний ... 570

4.4 Управление файлами и программами ... 570
4.4.1 Программная память ... 570
4.4.1.1 Программная память в NCK ... 570
4.4.1.2 Внешняя программная память ... 573
4.4.1.3 Адресация файлов программной памяти ... 574
4.4.1.4 Маршрут поиска при вызове подпрограммы ... 579
4.4.1.5 Запрос пути и имени файла ... 581
4.4.2 Оперативная память (CHANDATA, COMPLETE, INITIAL) ... 582

4.5 Обработка файла ... 585
4.5.1 Записать файл (WRITE) ... 585
4.5.2 Удалить файл (DELETE) ... 589
4.5.3 Чтение строк в файле (READ) ... 590
4.5.4 Проверка наличия файла (ISFILE) ... 592
4.5.5 Выгрузка файловой информации (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) ... 593

4.6 Защищенные области ... 595
4.6.1 Определение защищенных областей (CPROTDEF, NPROTDEF) ... 595

4.6.2 Активация/деактивация защищенных областей (CPROT, NPROT) ... 599
4.6.3 Проверка нарушения защищенной области, ограничения рабочего поля и программных конечных выключателей (CALCPOSI) ... 602

4.7 Специальные команды перемещения ... 613
4.7.1 Подвод к кодированным позициям (CAC, CIC, CDC, CACP, CACN) ... 613
4.7.2 Сплайн-интерполяция (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) ... 614
4.7.3 Соединение сплайнов (SPLINEPATH) ... 625
4.7.4 Включение / выключение сжатия кадра ЧПУ (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF) ... 626
4.7.5 Полиномиальная интерполяция (POLY, POLYPATH, PO, PL) ... 627
4.7.6 Устанавливаемое соотношение траекторий (SPATH, UPATH) ... 632
4.7.7 Специфическое для канала измерение (MEAS, MEAW) ... 634
4.7.8 Спец. для оси измерение (MEASA, MEAWA, MEAC) (опция): ... 637
4.7.9 Специальные функции для OEM-пользователя (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) ... 646
4.7.10 Уменьшение подачи с замедлением на углах (FENDNORM, G62, G621) ... 647
4.7.11 Программируемый критерий окончания движения (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA) ... 648

4.8 Трансформации координат (фреймы) ... 650
4.8.1 Трансформация координат через фрейм-переменные ... 650
4.8.1.1 Предопределенные фрейм-переменные ($P_CHBFRAME, $P_IFRAME, $P_PFRAME, $P_ACTFRAME) ... 653
4.8.2 Присвоение значений фреймам ... 656
4.8.2.1 Присвоение прямых значений (значение оси, угол, масштаб) ... 656
4.8.2.2 Чтение и изменение компонентов фрейма (TR, FI, RT, SC, MI) ... 658
4.8.2.3 Вычисление с фреймами ... 660
4.8.2.4 Определение фрейм-переменных (DEF FRAME) ... 661
4.8.3 Грубое и точное смещение (CTRANS, CFINE) ... 662
4.8.4 Внешнее смещение нулевой точки ($AA_ETRANS) ... 664
4.8.5 Установка фактического значения с потерей состояния реферирования (PRESETON) ... 665
4.8.6 Установка фактического значения без потери состояния реферирования (PRESETONS) ... 666
4.8.7 Вычисление фрейма из 3 точек измерения в пространстве (MEAFRAME) ... 668
4.8.8 Глобальные фреймы ... 672
4.8.8.1 Специфические для канала фреймы ($P_CHBFR, $P_UBFR) ... 673
4.8.8.2 Действующие в канале фреймы ... 673

4.9 Трансформации ... 678
4.9.1 Общее программирование типов трансформаций ... 678
4.9.1.1 Движения ориентации при трансформациях ... 681
4.9.1.2 Обзор трансформации ориентации TRAORI ... 685
4.9.2 3-, 4- и 5-осевая трансформация (TRAORI) ... 687
4.9.2.1 Общие связи карданной инструментальной головки ... 687
4.9.2.2 3-, 4- и 5-осевая трансформация (TRAORI) ... 690
4.9.2.3 Варианты программирования ориентации и первичная установка (ORIRESET) ... 691
4.9.2.4 Программирование ориентации инструмента (A..., B..., C..., LEAD, TILT) ... 693
4.9.2.5 Торцовое фрезерование (A4, B4, C4, A5, B5, C5) ... 699
4.9.2.6 Нулевая точка осей ориентации (ORIWKS, ORIMKS) ... 700
4.9.2.7 Программирование осей ориентации (ORIAXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) ... 702

4.9.2.8 Программирование ориентации вдоль боковой поверхности конуса (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) ... 704
4.9.2.9 Задача ориентации двух контактных точек (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=) ... 708
4.9.3 Полиномы ориентации (PO[угол], PO[координата]) ... 709
4.9.4 Вращения ориентации инструмента (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) ... 711
4.9.5 Ориентации относительно траектории ... 714
4.9.5.1 Типы ориентаций относительно траектории ... 714
4.9.5.2 Вращение ориентации инструмента относительно траектории (ORIPATH, ORIPATHS, угол поворота) ... 715
4.9.5.3 Интерполяция вращения инструмента относительно траектории (ORIROTC, THETA) ... 717
4.9.5.4 Сглаживание характеристики ориентации (ORIPATHS A8=, B8=, C8=) ... 719
4.9.6 Сжатие ориентации (COMPON, COMPCURV, COMPCAD, COMPSURF) ... 720
4.9.7 Включение/выключение сглаживания характеристики ориентации (ORISON, ORISOF) ... 723
4.9.8 Кинематическая трансформация ... 724
4.9.8.1 Активация трансформации с торцевой стороны (TRANSMIT) ... 724
4.9.8.2 Включение трансформации боковой поверхности цилиндра (TRACYL) ... 724
4.9.8.3 Включение косоугольной трансформации с программируемым углом (TRAANG) ... 727
4.9.8.4 Наклонное врезание на шлифовальных станках (G5, G7) ... 729
4.9.9 Включить связанную трансформацию (TRACON) ... 731
4.9.10 Движение “от точки к точке” в декартовой системе координат ... 732
4.9.10.1 Включение / выключение движения “от точки к точке” в декартовой системе координат (PTP, PTPG0, PTPWOC, CP) ... 732
4.9.10.2 Задание положения шарниров (STAT) ... 734
4.9.10.3 Знак угла между осями (TU) ... 738
4.9.10.4 Пример 1: Движение “от точки к точке" 6-осевого робота с трансформацией ROBX ... 740
4.9.10.5 Пример 2: Движение "от точки к точке" при базовой 5-осевой трансформации ... 742
4.9.10.6 Пример 3: PTPG0 и ... 742
4.9.11 Граничные условия при выборе трансформации ... 744
4.9.12 Отключение трансформации (TRAFOOF) ... 745

4.10 Кинематические цепочки ... 745
4.10.1 Удаление компонентов (DELOBJ) ... 745
4.10.2 Определение индекса по имени (NAMETOINT) ... 749

4.11 Предотвращение столкновений с кинематическими цепочками ... 750
4.11.1 Проверка на соударяемую пару (COLLPAIR) ... 750
4.11.2 Запрос повторного вычисления модели станка для предотвращения столкновений (PROTA) ... 750
4.11.3 Установка состояния защищенной области (PROTS) ... 751
4.11.4 Определение интервала между двумя защищенными областями (PROTD) ... 752

4.12 Трансформации с кинематическими цепями ... 754
4.12.1 Активация трансформации (TRAFOON) ... 754
4.12.2 Модифицирование трансформации ориентации после замеров станка (CORRTRAFO) ... 755

4.13 Коррекции инструмента ... 762
4.13.1 Память коррекций ... 762
4.13.2 Аддитивные коррекции ... 765
4.13.2.1 Выбор аддитивных коррекций (DL) ... 765
4.13.2.2 Определение значений износа и установочных значений ($TC_SCPxy[t,d], $TC_ECPxy[t,d]) ... 767
4.13.2.3 Удаление аддитивных коррекций (DELDL) ... 767

4.13.3 Коррекция инструмента - специальная обработка ... 767
4.13.3.1 Отражение длин инструмента ... 769
4.13.3.2 Нормирование знака износа ... 770
4.13.3.3 Система координат активной обработки (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS) ... 771
4.13.3.4 Длина инструмента и смена плоскостей ... 773
4.13.4 Коррекция инструмента Online ... 774
4.13.4.1 Определение функции полинома (FCTDEF) ... 774
4.13.4.2 Онлайн-запись коррекции инструмента, непрерывная (PUTFTOCF) ... 776
4.13.4.3 Дискретная запись онлайн-коррекции инструмента (PUTFTOC) ... 777
4.13.4.4 Включение/выключение онлайн-коррекции инструмента (FTOCON/FTOCOF) ... 778
4.13.5 Коррекция радиуса инструмента 3D ... 778
4.13.5.1 Коррекция на радиус инструмента 3D для периферийного фрезерования 3D (CUT3DC, CUT3DCD, ISD) ... 778
4.13.5.2 Коррекция на радиус инструмента 3D для торцового фрезерования 3D (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD) ... 783
4.13.5.3 Периферийное фрезерование 3D с учетом ограничивающей поверхности (CUT3DCC, CUT3DCCD) ... 789
4.13.6 Ориентация инструмента (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) ... 794
4.13.7 Присвоение свободных номеров D, номеров резцов ... 800
4.13.7.1 Присвоение свободных номеров D, номеров резцов (адрес CE) ... 800
4.13.7.2 Присвоение свободных номеров D: Проверка номеров D (CHKDNO) ... 800
4.13.7.3 Присвоение свободных номеров D: Переименование номеров D (GETDNO, SETDNO) ... 801
4.13.7.4 Присвоение свободных номеров D: Определение номера T для заданного номера D (GETACTTD) ... 802
4.13.7.5 Присвоение свободных номеров D: Установка номеров D как недействительных (DZERO) ... 802
4.13.8 Кинематика инструментального суппорта ... 802
4.13.9 Коррекция длин инструмента для ориентируемого инструментального суппорта (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ) ... 808
4.13.10 Модифицирование ориентируемых инструментальных суппортов после замеров станка (CORRTC) ... 813
4.13.11 Коррекция длин инструмента Online (TOFFON, TOFFOF) ... 817
4.13.12 Изменение данных коррекции для вращающихся инструментов ... 819
4.13.12.1 Вычисление ориентаций (ORISOLH) ... 819
4.13.12.2 Активация модификации данных коррекции для вращающихся инструментов (CUTMOD, CUTMODK) ... 830
4.13.13 Работа с инструментальным окружением ... 838
4.13.13.1 Сохранение инструментального окружения (TOOLENV) ... 838
4.13.13.2 Удаление инструментального окружения (DELTOOLENV) ... 841
4.13.13.3 Чтение номеров T, D и DL (GETTENV) ... 842
4.13.13.4 Считывание данных сохраняемого инструментального окружения ($P_TOOLENVN, ($P_TOOLENV) ... 843
4.13.13.5 Считывание компонентов длины инструмента или компонентов длины (GETTCOR) ... 844
4.13.13.6 Изменение компонентов инструмента (SETTCOR) ... 850
4.13.14 Считывание соотнесения длин инструмента L1, L2, L3 с осями координат (LENTOAX) ... 862

4.14 Параметры траектории ... 865
4.14.1 Характеристика подачи (FNORM, FLIN, FCUB, FPO) ... 865
4.14.2 Динамические характеристики ... 870
4.14.2.1 Режим ускорения (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA) ... 870
4.14.2.2 Управление ускорением для ведомых осей (VELOLIMA, ACCLIMA, JERKLIMA) ... 872

4.14.2.3 Активация спец. для технологии динамических значений (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH, DYNPREC) ... 874
4.14.3 Движение с предуправлением (FFWON, FFWOF) ... 875
4.14.4 Программирование допуска контура/ориентации (CTOL, OTOL, ATOL) ... 876
4.14.5 Включение/выключение программируемой точности контура (CPRECON, CPRECOF) ... 881
4.14.6 Включение/выключение автоматического переключения фильтров (AFISON, AFISOF) ... 883
4.14.7 Выполнение программы с памятью предварительной обработки (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE) ... 886
4.14.8 Определение областей задержки останова (DELAYFSTON, DELAYFSTOF) ... 888
4.14.9 Пропуск места в программе для SERUPRO (IPTRLOCK, IPTRUNLOCK) ... 891
4.14.10 Повторный подвод к контуру (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) ... 893
4.14.11 Воздействие на управление движением ... 901
4.14.11.1 Коррекция максимальной скорости оси или частоты вращения шпинделя (VELOLIM) ... 901
4.14.11.2 Коррекция максимального рывка оси (JERKLIM) ... 903
4.14.11.3 Коррекция максимальной скорости движения по траектории (FLIM) ... 904
4.14.11.4 Коррекция максимального ускорения по траектории (PACCLIM) ... 906
4.14.12 Параметры смены кадров при активном соединении (CPBC) ... 907

4.15 Функции осей ... 908
4.15.1 Переход оси, переход шпинделя (RELEASE, GET, GETD) ... 908
4.15.2 Передача оси в другой канал (AXTOCHAN) ... 912
4.15.3 Осевые функции (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) ... 914
4.15.4 Переключаемые геометрические оси (GEOAX) ... 916
4.15.5 Осевой контейнер (AXCTSWE, AXCTSWED, AXCTSWEC) ... 920
4.15.6 Ожидание действительной позиции оси (WAITENC) ... 922
4.15.7 Программируемое переключение блоков параметров (SCPARA) ... 923
4.15.8 Включение/выключение адаптации (CADAPTON, CADAPTOF) ... 924
4.15.9 Коррекция FIR-фильтра для динамического режима (CALCFIR) ... 927
4.15.10 Чтение/запись параметров привода в программе обработки детали (DRVPRD, DRVPWR) ... 929

4.16 Соединения осей ... 932
4.16.1 Буксировка (TRAILON, TRAILOF) ... 932
4.16.2 Таблицы кривых (CTAB) ... 936
4.16.2.1 Определение таблиц кривых (CTABDEF, CATBEND) ... 936
4.16.2.2 Проверка наличия таблицы кривых (CTABEXISTS) ... 942
4.16.2.3 Удаление таблиц кривых (CTABDEL) ... 942
4.16.2.4 Блокировка таблиц кривых от удаления и перезаписи (CTABLOCK, CTABUNLOCK) ... 943
4.16.2.5 Таблицы кривых: Определение свойств таблиц (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD) ... 944
4.16.2.6 Чтение значений таблиц кривых (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) ... 946
4.16.2.7 Таблицы кривых: Проверка использования ресурсов (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) ... 950
4.16.3 Осевое соединение по главному значению (LEADON, LEADOF) ... 952
4.16.4 Электронный редуктор (EG) ... 957
4.16.4.1 Определение электронного редуктора (EGDEF) ... 958
4.16.4.2 Включение электронного редуктора (EGON, EGONSYN, EGONSYNE) ... 959
4.16.4.3 Выключение электронного редуктора (EGOFS, EGOFC) ... 962

4.16.4.4 Удаление определения электронного редуктора (EGDEL) ... 963
4.16.4.5 Окружная подача (G95) / электронный редуктор (FPR) ... 963
4.16.5 Синхронный шпиндель ... 963
4.16.5.1 Синхронный шпиндель: Программирование (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) ... 964
4.16.6 Базовое соединение осей (CP...) ... 974
4.16.7 Тангенциальное управление ... 982
4.16.7.1 Определение соединения (TANG) ... 982
4.16.7.2 Включение создания промежуточного кадра (TLIFT) ... 984
4.16.7.3 Включение соединения (TANGON) ... 985
4.16.7.4 Выключение соединения (TANGOF) ... 986
4.16.7.5 Удаление соединения (TANGDEL) ... 987
4.16.8 Соединение Master/Slave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS) ... 988

4.17 Синхронные действия ... 992
4.17.1 Определение синхронного действия ... 992

4.18 Качание ... 993
4.18.1 Асинхронное качание (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) ... 993
4.18.2 Управляемое через синхронные действия качание (OSCILL) ... 998

4.19 Штамповка и вырубка ... 1005
4.19.1 Активация/деактивация ... 1005
4.19.1.1 Включение или выключение штамповки и вырубки (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) ... 1005
4.19.2 Автоматическое разделение пути ... 1009
4.19.2.1 Разделение пути для траекторных осей ... 1012
4.19.2.2 Разделение пути для отдельных осей ... 1013

4.20 Шлифование ... 1015
4.20.1 Включение / выключение специфического для шлифования контроля инструмента (TMON, TMOF) ... 1015

4.21 Расширенный останов и отвод (ESR) ... 1016
4.21.1 Управляемый ЧПУ ESR ... 1017
4.21.1.1 Управляемый ЧПУ отвод (POLF, POLFA, POLFMASK, POLFMLIN) ... 1017
4.21.1.2 Управляемый ЧПУ останов ... 1021
4.21.2 Автономный для привода ESR ... 1021
4.21.2.1 Конфигурирование автономного останова привода (ESRS) ... 1021
4.21.2.2 Конфигурирование автономного отвода привода (ESRS) ... 1022

4.22 Время выполнения программы / счетчик деталей ... 1023
4.22.1 Время выполнения программы ... 1024
4.22.2 Счетчики деталей ... 1027

4.23 Симуляция программы ... 1029
4.23.1 Функция ... 1029
4.23.2 Симуляция программы в сочетании с компилируемыми циклами ... 1029

4.24 Другие функции ... 1030
4.24.1 Активация машинных данных (NEWCONF) ... 1030
4.24.2 Проверка имеющейся языковой среды ЧПУ (STRINGIS) ... 1031
4.24.3 Интерактивный вызов окон из программы обработки детали (MMC) ... 1034
4.24.4 Process DataShare - Вывод на внешнее устройство/файл (EXTOPEN, WRITE, EXTCLOSE). 1039
4.24.5 Ошибки (SETAL) ... 1043
4.24.6 Определение заготовки (WORKPIECE) ... 1045

4.24.7 Переключение языкового режима (G290, G291) ... 1049

4.25 Собственные программы обработки резаньем ... 1051
4.25.1 Функции поддержки для обработки резаньем ... 1051
4.25.2 Создание таблицы контуров (CONTPRON) ... 1051
4.25.3 Создание кодированной таблицы контуров (CONTDCON) ... 1057
4.25.4 Определить точку пересечения между двумя элементами контура (INTERSEC) ... 1061
4.25.5 Покадровая обработка элементов контура таблицы (EXECTAB) ... 1062
4.25.6 Вычисление данных окружности (CALCDAT) ... 1063
4.25.7 Отключить подготовку контура (EXECUTE) ... 1065

4.26 Внешнее программирование циклов ... 1066
4.26.1 Технологические циклы ... 1066
4.26.1.1 Введение ... 1066
4.26.1.2 Обзор по технологиям ... 1068
4.26.1.3 HOLES1 – Образец позиции "Линия" ... 1070
4.26.1.4 HOLES2 – Образец позиции "Окружность или неполная окружность" ... 1071
4.26.1.5 POCKET3 – Прямоугольный карман ... 1073
4.26.1.6 POCKET4 – Круговой карман ... 1075
4.26.1.7 SLOT1 - Продольный паз ... 1078
4.26.1.8 SLOT2 - кольцевая канавка ... 1081
4.26.1.9 LONGHOLE - Продольный паз ... 1083
4.26.1.10 CYCLE60 – Гравирование ... 1085
4.26.1.11 CYCLE61 - плоское фрезерование ... 1088
4.26.1.12 CYCLE62 - Вызов контура ... 1090
4.26.1.13 CYCLE63 – Фрезерование контурного кармана / Контурный карман – остаточный материал / Фрезерование контурной цапфы / Контурная цапфа – остаточный материал ... 1091
4.26.1.14 CYCLE64 - Предварительное сверление кармана ... 1093
4.26.1.15 CYCLE70 - резьбофрезерование ... 1095
4.26.1.16 CYCLE72 - фрезерование траектории ... 1097
4.26.1.17 CYCLE76 – Прямоугольная цапфа ... 1101
4.26.1.18 CYCLE77 – Круговая цапфа ... 1103
4.26.1.19 CYCLE78 - Сверлильное резьбофрезерование ... 1105
4.26.1.20 CYCLE79 - многогранник ... 1108
4.26.1.21 CYCLE81 - сверление, центрование ... 1111
4.26.1.22 CYCLE82 - сверление, цекование ... 1112
4.26.1.23 CYCLE83 – Глубокое сверление 1 ... 1114
4.26.1.24 CYCLE84 - нарезание внутренней резьбы без компенсирующего патрона ... 1117
4.26.1.25 CYCLE85 - развертывание ... 1120
4.26.1.26 CYCLE86 - растачивание ... 1121
4.26.1.27 CYCLE92 - отрез ... 1123
4.26.1.28 CYCLE95 - съём припуска вдоль контура ... 1124
4.26.1.29 CYCLE98 - цепочка резьб ... 1126
4.26.1.30 CYCLE99 - нарезание резьбы резцом ... 1130
4.26.1.31 CYCLE435 - установка системы координат правящего инструмента ... 1135
4.26.1.32 CYCLE495 - Профилирование ... 1135
4.26.1.33 CYCLE782 – Коррекция по загрузке ... 1137
4.26.1.34 CYCLE800 – Поворот плоскости, поворот инструмента, точная установка инструмента ... 1140
4.26.1.35 CYCLE801 – Образец позиции "Решетка или рамка" ... 1144
4.26.1.36 CYCLE802 - любые позиции ... 1145
4.26.1.37 CYCLE806 — интерполированное вращение ... 1148

4.26.1.38 CYCLE830 - Глубокое сверление 2 ... 1149
4.26.1.39 CYCLE832 - High Speed Settings ... 1156
4.26.1.40 CYCLE840 - Нарезание внутренней резьбы с компенсирующим патроном ... 1160
4.26.1.41 CYCLE899 – открытый паз ... 1162
4.26.1.42 CYCLE930 - выточка ... 1165
4.26.1.43 CYCLE940 – Канавка формы E и F, резьбовая канавка ... 1168
4.26.1.44 CYCLE951 - обработка резанием ... 1171
4.26.1.45 CYCLE952 – Обработка резаньем / Обработка резаньем – остаток / Прорез / Прорез – остаток / Токарная выточка / Токарная выточка – остаток ... 1174
4.26.1.46 CYCLE4071 - Продольное шлифование с подачей в точке возврата ... 1180
4.26.1.47 CYCLE4072 - Продольное шлифование с подачей в точке возврата и сигналом отмены ... 1182
4.26.1.48 CYCLE4073 - Продольное шлифование с непрерывной подачей на глубину ... 1186
4.26.1.49 CYCLE4074 - Продольное шлифование с непрерывной подачей на глубину и сигналом отмены ... 1187
4.26.1.50 CYCLE4075 - Плоское шлифование с подачей в точке возврата ... 1190
4.26.1.51 CYCLE4077 - Плоское шлифование с подачей в точке возврата и сигналом отмены. 1193
4.26.1.52 CYCLE4078 - Плоское шлифование с непрерывной подачей на глубину ... 1197
4.26.1.53 CYCLE4079 - Плоское шлифование с прерывистой подачей на глубину ... 1199
4.26.1.54 GROUP_BEGIN — Начало блока программы ... 1201
4.26.1.55 GROUP_END — Конец блока программы ... 1202
4.26.1.56 GROUP_ADDEND — конец дополнения ввода ... 1202
4.26.1.57 Граничные условия ... 1203
4.26.2 Измерительные циклы ... 1204
4.26.2.1 Обзор параметров измерительных циклов ... 1204

5 Таблицы ... 1247

5.1 Операторы ... 1247

5.2 Адреса ... 1289
5.2.1 Буквы адреса ... 1289
5.2.2 Постоянные адреса ... 1290
5.2.3 Устанавливаемые адреса ... 1296

5.3 G-команды ... 1302
5.3.1 G-команды ... 1302
5.3.2 G-группа 1: Действующие модально команды движения ... 1302
5.3.3 G-группа 2: Действующие не модально движения, время ожидания ... 1303
5.3.4 G-группа 3: Программируемый фрейм, ограничение рабочего поля и программирование полюса ... 1304
5.3.5 G-группа 4: FIFO ... 1305
5.3.6 G-группа 6: Выбор плоскости ... 1305
5.3.7 G-группа 7: коррекция радиуса инструмента ... 1305
5.3.8 G-группа 8: Настраиваемое смещение нулевой точки ... 1305
5.3.9 G-группа 9: Подавление фрейма и коррекции инструмента ... 1306
5.3.10 G-группа 10: Точный останов – режим управления траекторией ... 1306
5.3.11 G-группа 11: Не модальный точный останов ... 1307
5.3.12 G-группа 12: Критерии смены кадра для точного останова (G60/G9) ... 1307
5.3.13 G-группа 13: Дюймовая/метрическая постановка размеров детали ... 1307
5.3.14 G-группа 14: Абсолютная/инкрементальная постановка размеров детали ... 1308
5.3.15 G-группа 15: Тип подачи ... 1308
5.3.16 G-группа 16: Процентовка подачи на внутреннем и наружном изгибе ... 1309
5.3.17 G-группа 17: Режим подвода/отвода, коррекция на инструмент ... 1309

5.3.18 G-группа 18: Поведение на углах, коррекция на инструмент ... 1309
5.3.19 G-группа 19: Криволинейный переход для начала сплайна ... 1309
5.3.20 G-группа 20: Криволинейный переход для конца сплайна ... 1310
5.3.21 G-группа 21: Профиль ускорения ... 1310
5.3.22 G-группа 22: Тип коррекции на инструмент ... 1310
5.3.23 G-группа 23: Контроль столкновений на внутренних контурах ... 1311
5.3.24 G-группа 24: Предуправление ... 1311
5.3.25 G-группа 25: Референция ориентации инструмента ... 1312
5.3.26 G-группа 26: Режим повторного подвода для REPOS (действует модально) ... 1312
5.3.27 G-группа 27: Коррекция на инструмент при изменении ориентации на наружных углах ... 1312
5.3.28 G-группа 28: Ограничение рабочего поля ... 1312
5.3.29 G-группа 29: Программирование радиуса/диаметра ... 1313
5.3.30 G-группа 30: Сжатие кадров УП ... 1313
5.3.31 G-группа 31: OEM-G-команды ... 1313
5.3.32 G-группа 32: OEM-G-команды ... 1314
5.3.33 G-группа 33: Настраиваемая точная коррекция на инструмент ... 1314
5.3.34 G-группа 34: Сглаживание ориентации инструмента ... 1314
5.3.35 G-группа 35: Штамповка и вырубка ... 1315
5.3.36 G-группа 36: Штамповка с задержкой ... 1315
5.3.37 G-группа 37: Профиль подачи ... 1315
5.3.38 G-группа 38: Назначение быстрых входов/выходов для штамповки/вырубки ... 1316
5.3.39 G-группа 39: Программируемая точность контура ... 1316
5.3.40 G-группа 40: Постоянная коррекция на радиус инструмента ... 1316
5.3.41 G-группа 41: Резьбонарезание с возможностью прерывания ... 1316
5.3.42 G-группа 42: Инструментальный суппорт ... 1317
5.3.43 G-группа 43: Направление подвода SAR ... 1317
5.3.44 G-группа 44: Разделение пути SAR ... 1317
5.3.45 G-группа 45: Соотношение траекторий осей FGROUP ... 1318
5.3.46 G-группа 46: Выбор плоскости для быстрого отвода ... 1318
5.3.47 G-группа 47: Переключение режима для внешнего кода УП ... 1318
5.3.48 G-группа 48: Режим подвода/отвода при коррекции на радиус инструмента ... 1318
5.3.49 G-группа 49: Позиционное движение ... 1319
5.3.50 G-группа 50: Программирование ориентации ... 1319
5.3.51 G-группа 51: Тип интерполяции программирования ориентации ... 1319
5.3.52 G-группа 52: Вращения фрейма по отношению к детали ... 1320
5.3.53 G-группа 53: Вращения фрейма по отношению к инструменту ... 1320
5.3.54 G-группа 54: Вращение вектора при программировании полинома ... 1321
5.3.55 G-группа 55: Движение ускоренного хода с/без линейной интерполяции ... 1321
5.3.56 G-группа 56: Учет износа инструмента ... 1321
5.3.57 G-группа 57: Замедление на углах ... 1322
5.3.58 G-группа 59: Динамический режим для интерполяции траектории ... 1322
5.3.59 G-группа 60: Ограничение рабочего поля ... 1322
5.3.60 G-группа 61: Сглаживание ориентации инструмента ... 1323
5.3.61 G-группа 62: Режим повторного подвода для REPOS (действует не модально) ... 1323
5.3.62 G-группа 64: Цикловые фреймы ... 1323
5.4 Предопределенные процедуры ... 1324
5.5 Предопределенные процедуры в синхронных действиях ... 1348
5.6 Предопределенные функции ... 1350
5.7 Текущий язык в HMI ... 1364

A Приложение 1367
A.1 Список сокращений 1367
Указатель 1373

Введение

1

1.1 Информация о SINUMERIK

ЧПУ SINUMERIK предлагают походящее решение для любой концепции станков: начиная от простейших стандартных станков с ЧПУ, до стандартизированных станков и модульных концепций премиум-класса. Идет ли речь о штучном или массовом производстве, простых или сложных изделиях – SINUMERIK представляет собой высокопроизводительное решение по автоматизации всех без исключения производственных участков, начиная от изготовления прототипов и форм и заканчивая крупносерийным производством.

Дополнительная информация представлена на веб-странице SINUMERIK (https://www.siemens.com/sinumerik).

1.2 Информация о данной документации

Настоящая документация относится к группе руководств по программированию SINUMERIK.

Руководство по программированию "Программирование ЧПУ"

Руководство по программированию "Программирование ЧПУ" содержит всю информацию, необходимую для программирования функций системы ЧПУ SINUMERIK.

Целевой группой являются программисты и проектировщики.

Руководство по программированию помогает целевой группе в разработке, написании, тестировании и устранении ошибок программ и программных интерфейсов.

Обзор содержания

Из обзора содержания на титульной странице видно, что руководство по программированию "Программирование ЧПУ" содержит два основных раздела:

1. **Основы**
   Раздел "Основы" предназначена для профессиональных операторов станков и предполагает наличие соответствующих знаний в областях сверлильной, фрезерной и токарной обработки. На простых примерах программирования объясняются и известные по DIN 66025 команды и операторы.
2. **Подготовка к работе**
   Раздел "Подготовка к работе" предназначена для технологов со знанием всех возможностей программирования. Система ЧПУ SINUMERIK предлагает специальный язык для программирования комплексной программы обработки детали (к примеру, поверхности свободной формы, координация каналов,...) и облегчает трудоемкое программирование для технологов.

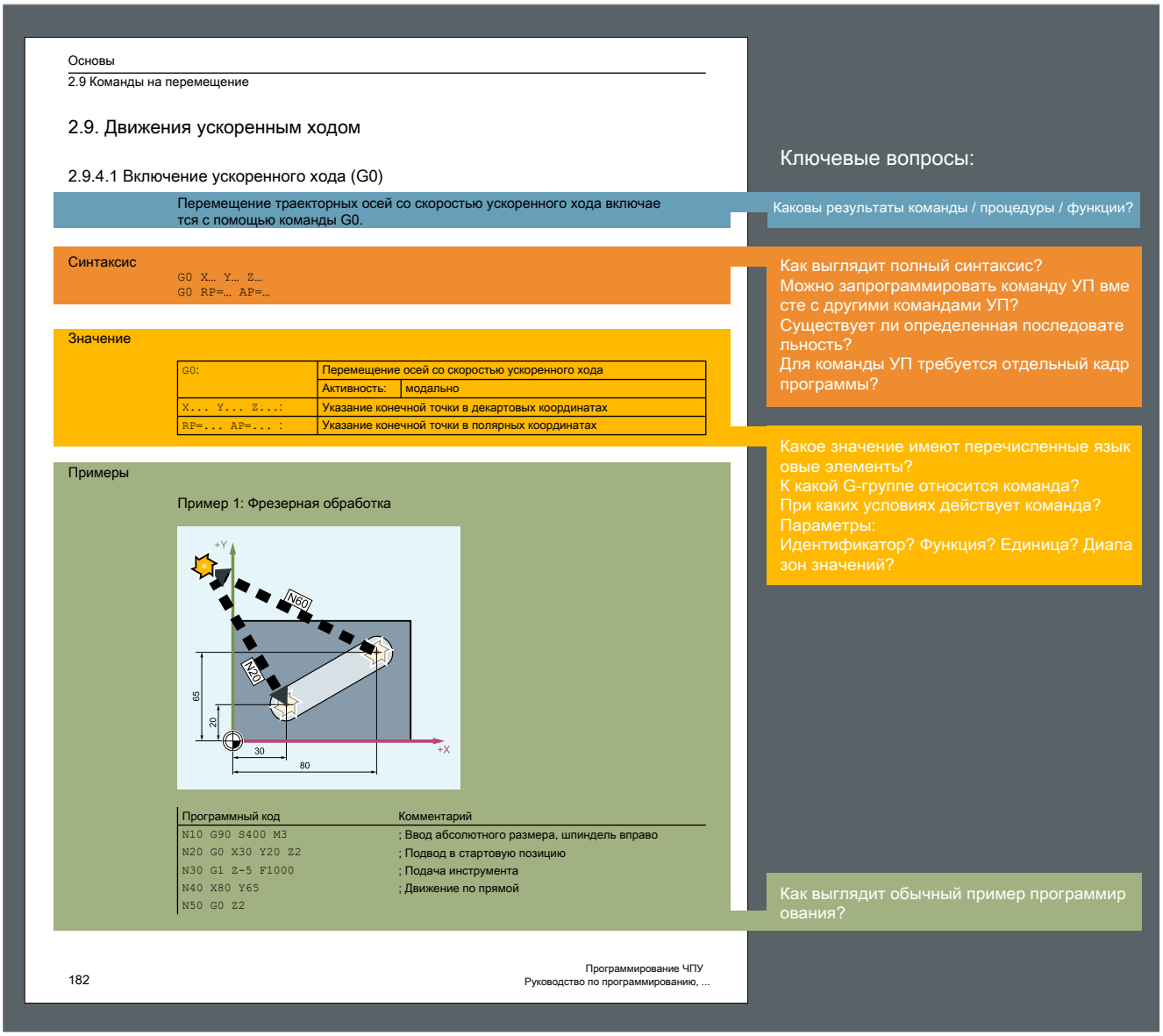
Сфера действия

Титульная страница также содержит информацию о сфере действия документации, т.е. к каким системам SINUMERIK и версиям ПО относится данное руководство по программированию.

Структура глав

Описания элементов языка подготовки УП (G-команда, процедура, функция, ...) имеют типовую структуру, при разработке содержимого учитываются предварительно определенные ключевые вопросы.

Рисунок ниже поясняет это на основе примера:



Наряду с показанными в примере и присутствующими всегда точками структурирования (синтаксис, значение и примеры), при необходимости, могут добавляться и другие точки структурирования:

Дополнительные точки структурирования	**Использование**
Условия	Если для использования программируемых функций должны быть выполнены определенные условия (напр., лицензирование опции).
Активность	Если программируемая функция может быть активна только при определенных условиях (напр., только в определенном рабочем состоянии).
Граничные условия	Если при программировании необходимо учитывать взаимодействие с другими функциями.
Дополнительная информация	Если для понимания программирования функции требуются дополнительные пояснения.

Стандартный объем

В настоящей документации описаны функции стандартного объема. Он может отличаться от объема функций поставленной системы. Функции поставленной системы указаны исключительно в документации по заказу.

Ваша система может иметь дополнительные функции, не описанные в данной документации. Однако претензии на наличие указанных функций не принимаются ни при поставке, ни при техническом обслуживании.

По соображениям наглядности данная документация может содержать не всю детальную информацию о различных типах изделия. Кроме того, в данной документации не могут быть учтены все возможные варианты расстановки, эксплуатации и ремонта.

Дополнения и изменения изделия, осуществляемые изготовителем станка, документирует изготовитель станка.

Веб-страницы третьих лиц

Настоящий документ может содержать гиперссылки на веб-страницы третьих лиц. Компания компания Siemens не несет ответственности за содержание этих веб-страниц и не идентифицирует себя с их содержанием. Компания Siemens не контролирует информацию на указанных веб-страницах и не несет ответственности за их содержание и представленные на них сведения. Риски, связанные с их использованием, несет пользователь.

1.3 Документация в Интернете

1.3.1 Обзор документации SINUMERIK 840D sl

Комплексная документация, касающаяся функций SINUMERIK 840D sl, начиная с версии 4.8 SP4, представлена в Обзоре документации 840D sl (https://support.industry.siemens.com/cs/ww/en/view/109766213).

Существует возможность просмотра документов или их скачивания в форматах PDF или HTML5.

Документация подразделяется на следующие категории:

- Пользователь: Управление
- Пользователь: Программирование
- Изготовитель/Сервис: Функции
- Изготовитель/Сервис: Аппаратное обеспечение
- Изготовитель/Сервис: Проектирование/ввод в эксплуатацию
- Изготовитель/Сервис: Safety Integrated
- Изготовитель/Сервис: SINUMERIK Integrate / MindApp
- Информация и обучение
- Изготовитель/Сервис: SINAMICS

1.3.2 Обзор документации для компонентов управления системы SINUMERIK

Большой объем документации для компонентов управления системы SINUMERIK представлен здесь: Обзор документации для компонентов управления системы SINUMERIK (https://support.industry.siemens.com/cs/document/109783841/technische-dokumentation-zu-sinumerik-bedienkomponenten?dti=0&lc=en-WW).

Существует возможность просмотра документов или их скачивания в форматах PDF или HTML5.

Документация подразделяется на следующие категории:

- Пульты оператора
- Станочные пульты
- Кнопочные станочные панели
- Миниатюрные ручные устройства
- Прочие компоненты систем управления

Обзор важнейших документов, статей и ссылок по теме "SINUMERIK" представлен здесь: Тематический обзор по системе SINUMERIK (https://support.industry.siemens.com/cs/document/109766201/sinumerik-an-overview-of-the-most-important-documents-and-links?dti=0&lc=en-WW).

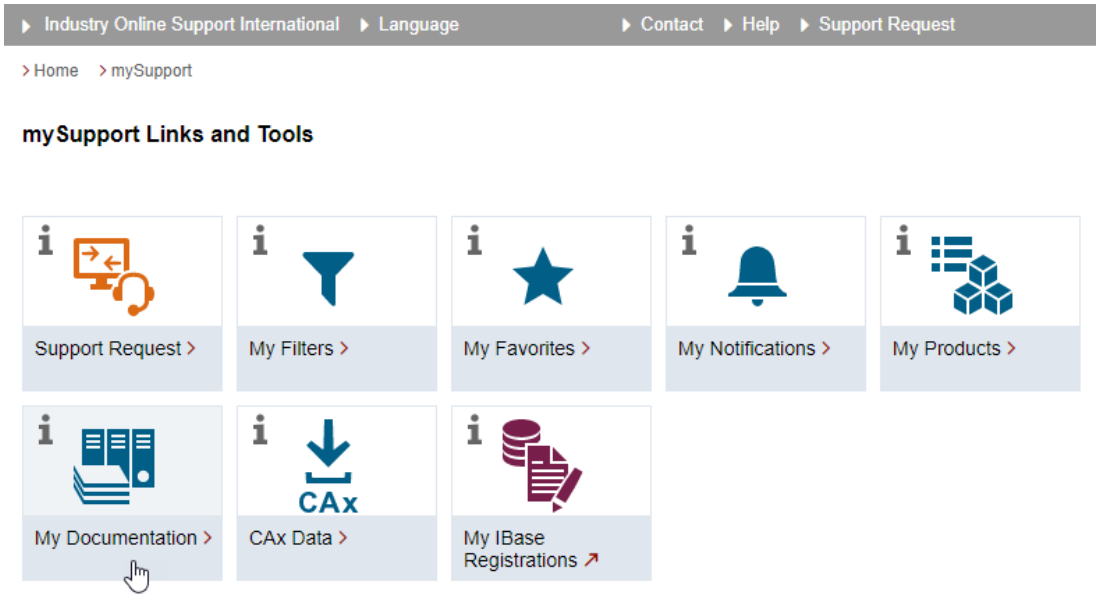
1.4 Комментарии, касающиеся технической документации

При наличии вопросов, пожеланий или исправлений к технической документации, опубликованной на странице Siemens Industry Online Support, просим использовать ссылку "Feedback senden" (обратная связь) в конце соответствующей статьи.

1.5 mySupport-Dokumentation

Веб-базированная система "mySupport-Dokumentation" позволяет составлять индивидуальные пакеты документации на основе контента Siemens и адаптировать ее к собственной документации на оборудование.

Для пуска приложения щелкните на кнопку "Meine Dokumentation" (моя документация) на исходной странице mySupport (https://support.industry.siemens.com/cs/ww/en/my):



Экспорт конфигурированного руководства возможен в форматах RTF, PDF или XML.

Примечание

Контент компании Siemens, совместимый с приложением "mySupport-Dokumentation", обозначен ссылкой "Konfigurieren" (конфигурировать).

1.6 Сервис и поддержка

Сопровождение изделий

Дополнительная информация об изделии представлена в Интернете:

Техническая поддержка (https://support.industry.siemens.com/cs/ww/en/)

На данной странице представлены:

- Текущая информация об изделии (сообщения об изделии)
- FAQ (часто задаваемые вопросы)
- Руководства
- Скачивание информации
- Уведомление о новейших сведениях, касающихся ваших изделий
- Форум для дополнительного обмена информацией и опытом между пользователями и специалистами
- Местные контактные лица (через нашу базу данных → "Kontakt" (контакт))
- Информация о местных услугах по обслуживанию, ремонту, обеспечению запасными частями и многом другом (→ "Services" (услуги))

Техническая поддержка

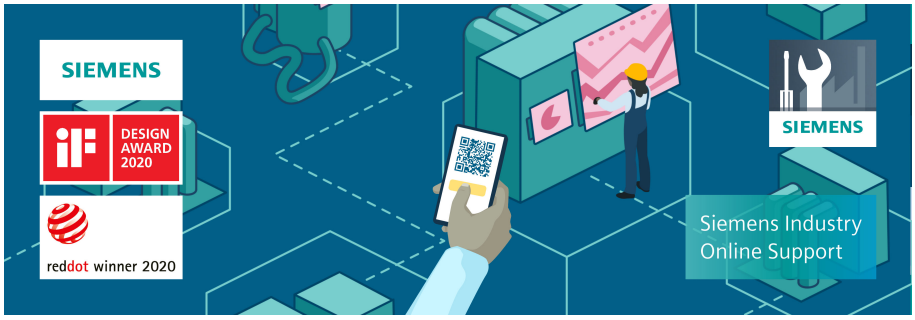
Телефонные номера технической поддержки в конкретных странах указаны в Интернете по данному адресу (https://support.industry.siemens.com/cs/ww/en/sc/4868) в разделе "Kontakt" (контакт).

При наличии технических вопросов используйте онлайн-формуляр в разделе "Support Request" (запрос на поддержку).

Обучение

Информация о SITRAIN представлена по следующему адресу (https://www.siemens.com/sitrain).
SITRAIN предлагает обучение работе с изделиями, системами и решениями для приводов и технологий автоматизации Siemens.

Поддержка от Siemens в любом месте в любое время



С помощью премированного приложения "Siemens Industry Online Support" (служба онлайн-поддержки сектора Siemens Industry) вы можете в любое время и отовсюду получить доступ более чем к 300.000 документов, касающихся продукции сектора Siemens Industry. Помимо прочего, приложение предназначено для поддержки в следующих сферах:

- Решение проблем при реализации проектов
- Порядок устранения отказов
- Расширение или перепланировка установок

Кроме того, вы получаете доступ к техническому форуму и прочей информации, подготовленной нашими специалистами:

- FAQ
- Примеры использования
- Руководства
- Сертификаты
- Информация об изделиях и многое другое

Приложение "Siemens Industry Online Support" (служба онлайн-поддержки сектора Siemens Industry) доступно для Apple iOS и Android.

Матричный штрих-код на паспортной табличке

Матричный штрих-код на паспортной табличке содержит индивидуальные характеристики устройства. Этот код может быть считан с помощью любого смартфона, мобильное приложение "Siemens Industry Online Support" (служба онлайн-поддержки сектора Siemens Industry) позволяет просмотреть техническую информацию о соответствующем устройстве.

1.7 Важная информация об изделии

Использование OpenSSL

Данный продукт может содержать следующее программное обеспечение:

- программное обеспечение, разработанное в рамках проекта OpenSSL для использования в комплекте инструментов OpenSSL
- криптографическое программное обеспечение, разработанное Эриком Янгом (Eric Young)
- программное обеспечение, разработанное Эриком Янгом (Eric Young)

Дополнительная информация представлена в Интернете:

- OpenSSL (https://www.openssl.org)
- Cryptsoft (https://www.cryptsoft.com)

Соблюдение общего регламента по защите данных

Компания Siemens соблюдает принципы защиты данных, в частности, принцип минимизации объема данных (проектируемая конфиденциальность).

В отношении данного продукта это означает следующее:

Продукт не обрабатывает и не сохраняет никаких персональных данных, а лишь технические функциональные параметры (например, метки времени). Если пользователь связывает эти данные с другими данными (например, с графиками смен) или сохраняет персональные данные на том же носителе (например, на жестком диске), делая, тем самым, возможным их соотнесение с определенными лицами, обязанность по соблюдению требований законодательства о защите данных переходит к нему.

Основные указания по безопасности 2

2.1 Общие указания по безопасности

⚠ ПРЕДУПРЕЖДЕНИЕ

Опасность для жизни вследствие несоблюдения указаний по безопасности и остаточным рискам

Несоблюдение указаний по безопасности и остаточным рискам, приведенных в соответствующей документации аппаратного обеспечения, может стать причиной тяжелых травм или смерти.

- Строго соблюдайте правила техники безопасности, указанные в документации на аппаратное обеспечение.
- При оценке риска необходимо учитывать остаточные риски.

⚠ ПРЕДУПРЕЖДЕНИЕ

Неполадки машины вследствие ошибок или изменения параметризации

Ошибки или изменения параметризации могут стать причиной неполадок машин, которые, в свою очередь, могут привести к тяжким или смертельным травмам.

- Обеспечьте защиту введенных параметров от несанкционированного доступа.
- В случае неполадок следует принимать надлежащие меры (например, АВАРИЙНЫЙ ОСТАНОВ или АВАРИЙНОЕ ОТКЛЮЧЕНИЕ).

2.2 Гарантийные обязательства и ответственность за прикладные примеры

Прикладные примеры не носят обязательного характера и не претендуют на полноту в отношении конфигурации и оснащения, а также на универсальность для всех возможных случаев. Прикладные примеры не представляют собой специализированные решения, а лишь предлагают помощь в решении типичных задач.

Вы как пользователь несете ответственность за надлежащую эксплуатацию описанных продуктов. Прикладные примеры не снимают с эксплуатирующего лица ответственности за безопасное обращение при использовании, установке, эксплуатации и техническом обслуживании.

2.3 Информация о безопасности

Siemens предоставляет продукты и решения для обеспечения промышленной безопасности при эксплуатации производственных комплексов, систем, рабочих станций и сетей.

Для защиты производственных комплексов, систем, машинного оборудования и сетей от киберугроз необходимо внедрение и поддержка комплексной высокотехнологичной модели промышленной безопасности. Продукты и решения Siemens являются одним из компонентов такой модели.

Клиенты отвечают за предотвращение несанкционированного доступа к их производственным комплексам, системам, рабочим станциям и сетям. Доступ таких систем, рабочих станций и их компонентов к корпоративной сети или сети Интернет должен быть организован только если такой доступ необходим и с применением соответствующих локальных мер безопасности (например, использование брандмауэров и/или деление сети на подсети).

Для получения дополнительных сведений о возможных мерах промышленной безопасности см. https://www.siemens.com/industrialsecurity (https://www.siemens.com/industrialsecurity).

Продукты и решения Siemens постоянно совершенствуются для обеспечения максимальной степени безопасности. Siemens настоятельно рекомендует выполнять обновления сразу после их выпуска и всегда использовать самые последние версии продуктов. Использование неподдерживаемых версий продуктов и неприменение последних обновлений повышает риск киберугроз для клиента.

Для получения сведений об обновлениях продуктов, подпишитесь на RSS-канал Siemens по промышленной безопасности:
https://www.siemens.com/industrialsecurity (https://new.siemens.com/global/en/products/services/cert.html#Subscriptions).

Дополнительная информация представлена в Интернете:

Справочник по проектированию, промышленная безопасность (https://support.industry.siemens.com/cs/ww/en/view/108862708)

⚠ ПРЕДУПРЕЖДЕНИЕ

Опасные режимы эксплуатации вследствие внесения несанкционированных изменений в программное обеспечение

Несанкционированные изменения программного обеспечения, например, под воздействием вирусов, троянов, вредоносного ПО или червей, может стать причиной опасных режимов эксплуатации установки, и как следствие, привести к тяжким или смертельным травмам и к повреждению оборудования.

- Постоянно обновляйте ПО.
- Интегрируйте компоненты автоматизации и приводов в единую концепцию промышленной безопасности установки или машины, соответствующую современному уровню развития техники.
- Единая концепция промышленной безопасности должна учитывать все используемые изделия.
- Принимайте необходимые меры по защите файлов на сменных носителях от вредоносного ПО, например, антивирусные программы.
- По завершении пусконаладки проконтролируйте все настройки, касающиеся безопасности.

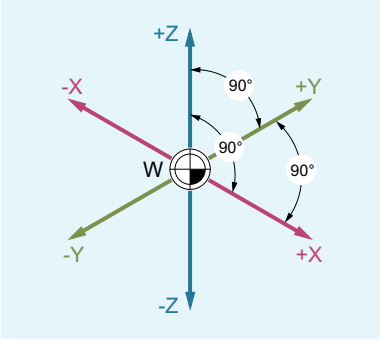
Основная информация 3

3.1 Геометрические основы

3.1.1 Позиции детали

3.1.1.1 Исходная система указаний позиций

Чтобы станок или СЧПУ могли работать с указанными в программе ЧПУ позициями, эти указанные позиции должны относиться к базовой системе, которую можно перенести на направления движения осей станка. Для этого на станках в качестве системы координат детали используются декартовы, т. е. правовращающиеся прямоугольные системы координат по стандарту DIN 66217.



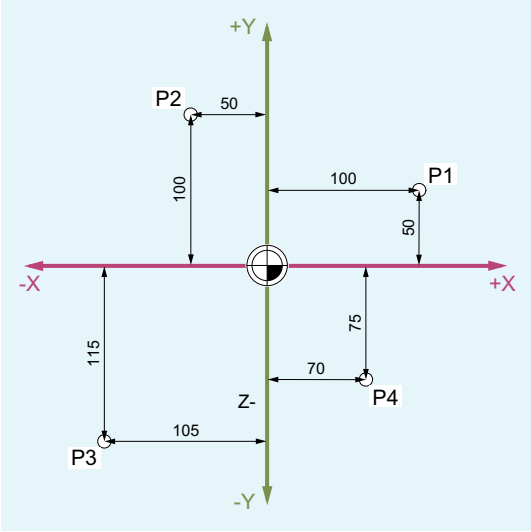
Нулевая точка детали (W) является исходной точкой системы координат детали.

3.1.1.2 Декартовы координаты

Оси в системе координат измерены. За счет этого существует возможность однозначного описания любой точки в системе координат и тем самым любой позиции детали через направление (X, Y и Z) и три числовых значения. Нулевая точка детали всегда имеет координаты X0, Y0 и Z0.

Указания позиций в форме декартовых координат

Для упрощения в этом примере рассматривается только одна плоскость системы координат — плоскость X/Y:

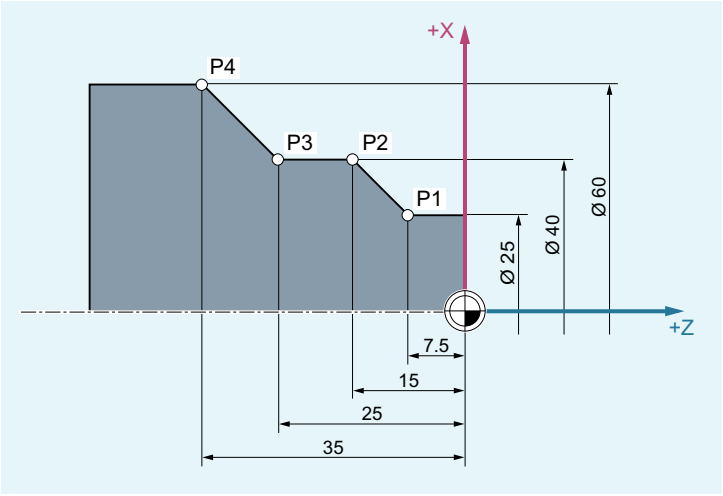


В этом случае точки от P1 до P4 имеют следующие координаты:

Позиция	Координаты
P1	X100 Y50
P2	X-50 Y100
P3	X-105 Y-115
P4	X70 Y-75

Пример: Позиции детали при токарной обработке

В случае токарных станков для описания контура достаточно одной плоскости:

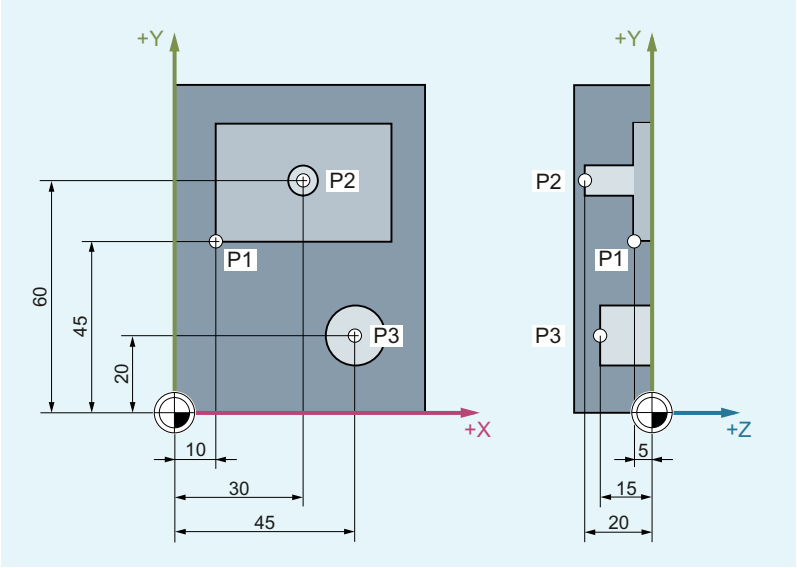


В этом случае точки от P1 до P4 имеют следующие координаты:

Позиция	Координаты
P1	X25 Z-7.5
P2	X40 Z-15
P3	X40 Z-25
P4	X60 Z-35

Пример: Позиции детали при фрезерной обработке

При фрезерных обработках необходимо описать и глубину подачи, т. е. третьей координате (в этом случае Z) также необходимо присвоить числовое значение.



Точки от P1 до P3 имеют следующие координаты:

Позиция	Координаты
P1	X10 Y45 Z-5
P2	X30 Y60 Z-20
P3	X45 Y20 Z-15

3.1.1.3 Полярные координаты

Вместо декартовых координат для описания позиций детали могут использоваться и полярные координаты. Это имееет смысл в тех случаях, когда деталь или часть детали измерены с радиусом и углом. Исходная точка измерения называется "Полюс".

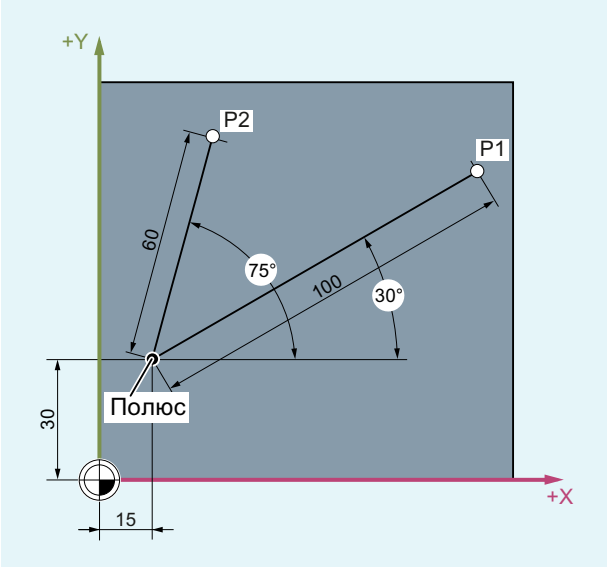
Указание позиций в форме полярных координат

Полярные координаты состоят из **полярного радиуса** и **полярного угла**.

Полярный радиус это расстояние между полюсом и позицией.

Полярный угол это угол между полярным радиусом и горизонтальной осью рабочей плоскости. Отрицательные полярные углы отсчитываются по часовой стрелке, положительные – против часовой стрелки.

Пример



Точки P1 и P2 могут быть описаны относительно полюса следующим образом:

Позиция	Полярные координаты
P1	RP=100 AP=30
P2	RP=60 AP=75
RP: Полярный радиус AP: Полярный угол	

3.1.1.4 Абсолютный размер

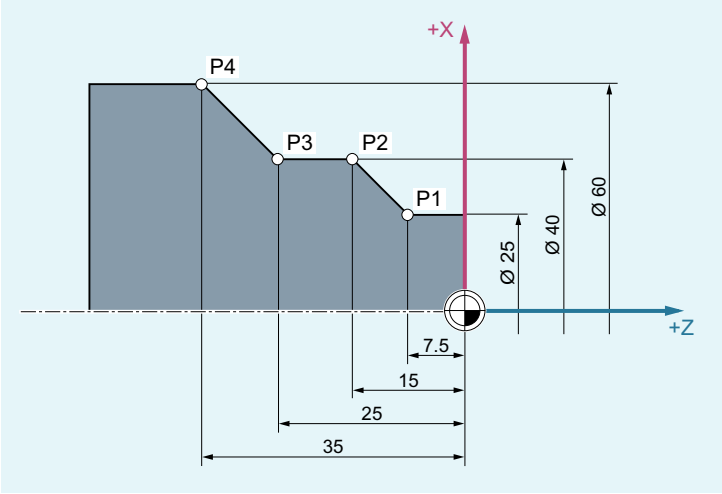
Указание позиций в абсолютном размере

У абсолютного размера все указания позиций всегда относятся к действующей в данный момент нулевой точке.

В отношении движения инструмента это означает:

Указание абсолютного размера описывает позицию, к которой должен двигаться инструмент.

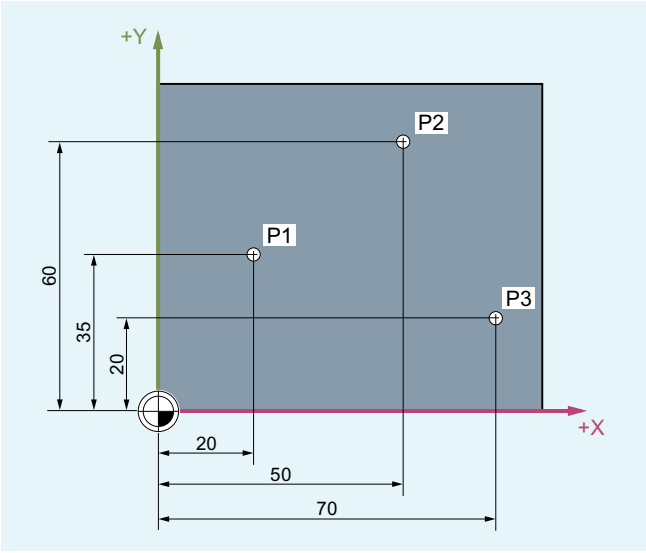
Пример: Токарная обработка



В абсолютном размере для точек P1 до P4 получаются следующие данные позиций:

Позиция	Указание позиций в абсолютном размере
P1	X25 Z-7,5
P2	X40 Z-15
P3	X40 Z-25
P4	X60 Z-35

Пример: Фрезерная обработка



В абсолютном размере для точек P1 до P3 получаются следующие данные позиций:

Позиция	Указание позиций в абсолютном размере
P1	X20 Y35
P2	X50 Y60
P3	X70 Y20

3.1.1.5 Составной размер

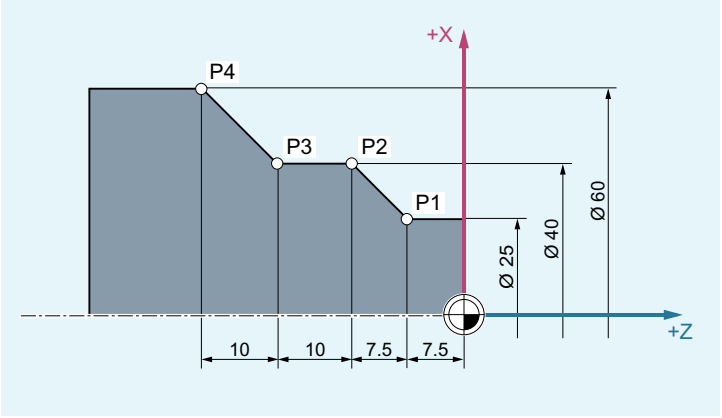
Указание позиций в составном размере (инкрементальном размере)

На рабочих чертежах размеры часто относятся не к нулевой точке, а к иной точке детали. Во избежание пересчета данных размеров существует возможность указания составного или инкрементального размера. При таком способе указания составного размера данные позиции относятся к соответствующей предыдущей точке.

В отношении движения инструмента это означает:

Данные составного размера описывают, на сколько должен переместиться инструмент.

Пример: Токарная обработка



В составном размере для точек от P2 до P4 получаются следующие данные позиций:

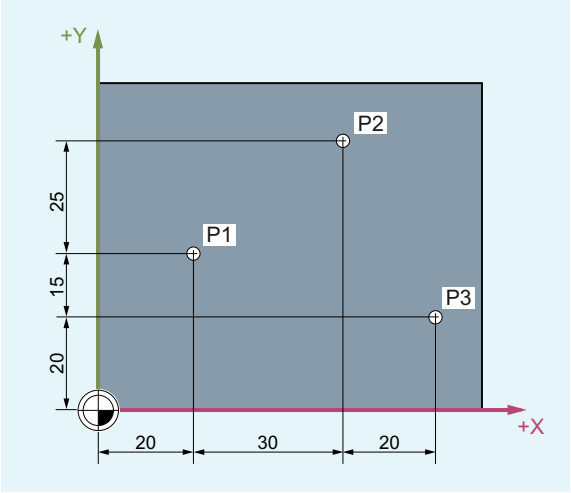
Позиция	Указание позиций в составном размере	Данные относятся к:
P2	X15 Z-7,5	P1
P3	Z-10	P2
P4	X20 Z-10	P3

Примечание

При имеющейся DIAMOF или DIAM90 (Страница 170) заданный путь при указании составного размера (G91) программируется как размер радиуса.

Пример: Фрезерование

Указание позиции для точек от P1 до P3 в составном размере:



В составном размере для точек от P1 до P3 получаются следующие данные позиций:

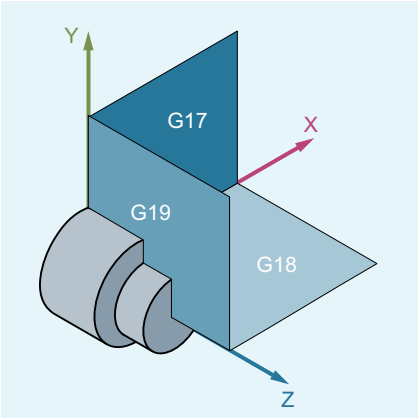
Позиция	Указание позиций в составном размере	Данные относятся к:
P1	X20 Y35	Нулевая точка
P2	X30 Y20	P1
P3	X20 Y-35	P2

3.1.2 Рабочие плоскости

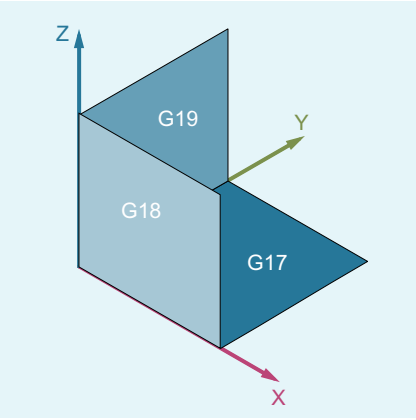
Любой программе ЧПУ необходима информация о том, в какой плоскости выполняется обработка. Только в этом случае система управления может, например, правильно рассчитывать значения коррекции инструмента. Кроме того, указание рабочей плоскости необходимо для определенных видов программирования окружностей и для полярных координат.

Рабочая плоскость во взятой за основу декартовой системе координат детали определяется двумя осями координат. Третья ось координат стоит вертикально на этой рабочей плоскости и определяет направление подачи инструмента (например, для обработки 2D).

Рабочие плоскости при токарной / фрезерной обработке



Рабочие плоскости при токарной обработке



Рабочие плоскости при фрезерной обработке

Активация рабочей плоскости

Рабочие плоскости активируются в программе ЧПУ с помощью G-команд G17, G18 и G19 . Эта взаимосвязь определяется следующим образом:

G-команда	Рабочая плоскость	Абсцисса	Ордината	Аппликата ≙ направление подачи
G17	X / Y	X	Y	Z
G18	Z / X	Z	X	Y
G19	Y / Z	Y	Z	X

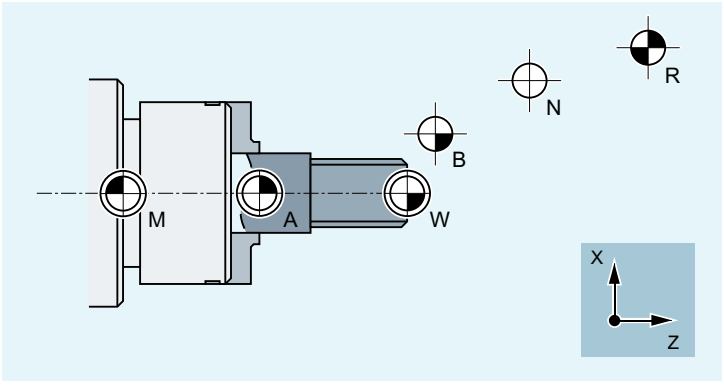
3.1.3 Нулевые точки и исходные точки

На станке с ЧПУ определены различные нулевые и исходные точки:

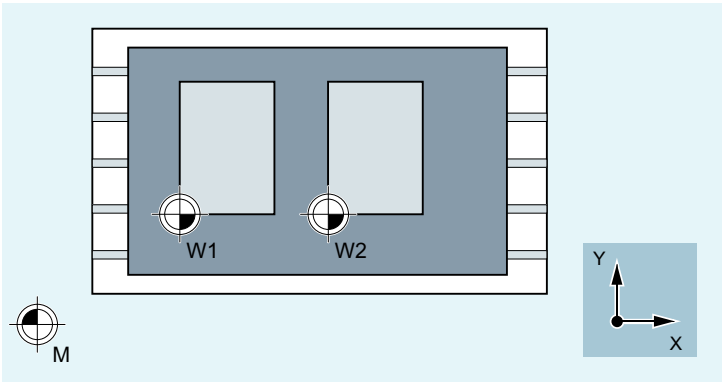
Нулевые точки		
	M	Нулевая точка станка С помощью нулевой точки станка определяется система координат станка (MCS). К нулевой точке станка относятся все другие исходные точки.
	W	Нулевая точка детали = нулевая точка программы Нулевая точка детали определяет систему координат детали относительно нулевой точки станка.
	A	Точка упора Может совпадать с нулевой точкой детали (только у токарных станков).

Исходные точки		
	r	Референтная точка Определенная кулачками и измерительной системой позиция. Расстояние до нулевой точки станка **M** должно быть известным, чтобы позиция оси в этом месте могла быть установлена точно на это значение.
	B	Стартовая точка Может быть определена через программу. Здесь 1-й инструмент начинает обработку.
	T	Исходная точка инструментального суппорта Находится на зажиме инструмента. Посредством ввода длин инструмента СЧПУ вычисляет расстояние от острия инструмента до исходной точки инструментального суппорта.
	N	Точка смены инструмента

Нулевые и исходные точки при токарной обработке



Нулевые точки при фрезеровании



3.1.4 Системы координат

Различаются следующие системы координат:

- Система координат станка (MCS) (Страница 40) с нулевой точкой станка **M**
- Базовая кинематическая система (BKS) (Страница 42)
- Базовая система нулевой точки (BNS) (Страница 43)
- Настраиваемая система нулевой точки (ENS) (Страница 43)
- Система координат детали (WCS) (Страница 44) с нулевой точкой детали **W**

3.1.4.1 Система координат станка (MCS)

Система координат станка образуется из всех физически имеющихся осей станка.

В системе координат станка определяются референтные точки, точки смены инструмента и паллет (фиксированные точки станка).

Если программирование осуществляется непосредственно в системе координат станка (возможно для некоторых G-команд), то обращение осуществляется напрямую к физическим осям станка. Зажим детали (если имеется) при этом не учитывается.

Примечание

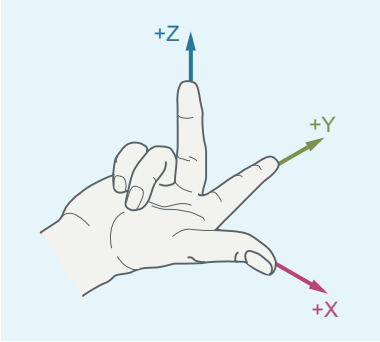
Если существуют различные системы координат станка (например, 5-осевая трансформация), то через внутреннюю трансформацию кинематика станка отображается на систему координат, в которой осуществляется программирование.

Правило правой руки

Положение системы координат относительно станка зависит от типа станка. Осевые направления определяются по так называемому "Правилу **правой** руки" (по DIN 66217).

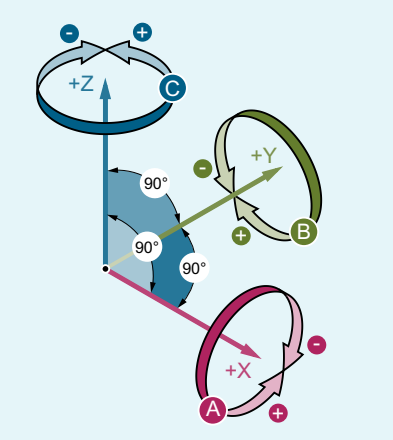
Если встать перед станком, то средний палец правой руки направлен против направления подачи главного шпинделя. Тогда:

- Большой палец обозначает направление +X
- Указательный палец — направление +Y
- Средний палец — направление +Z



Вращения вокруг осей координат X, Y и Z обозначаются посредством A, B и C. Направление вращения получается из направления движения вращения, если смотреть в положительном направлении оси координат:

Направление движения вращения	Направление вращения
по часовой стрелке	положительное
против часовой стрелки	отрицательное

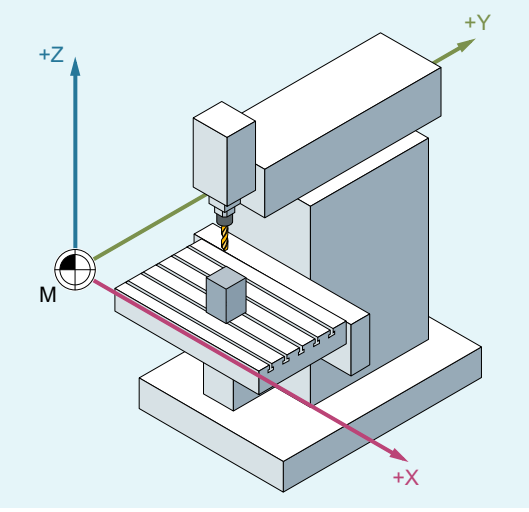


X, Y, Z Оси координат, расположенные вертикально друг на друге

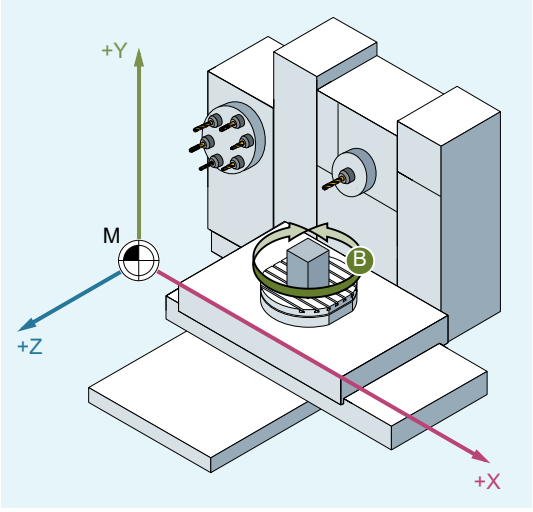
A, B, C Круговые оси, вращающиеся вокруг X, Y, Z

Положение системы координат у различных типов станков

Положение системы координат, получаемое по "правилу правой руки", может иметь различную ориентацию у различных типов станков (см. следующие примеры):



Вертикальный 3-осевой фрезерный станок



Горизонтальный 4-осевой фрезерный станок

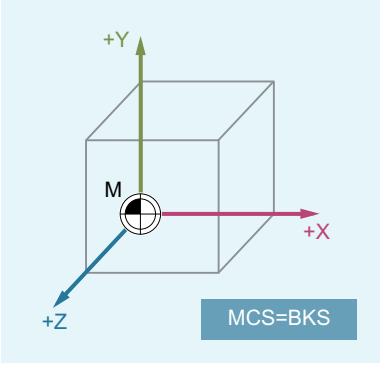
3.1.4.2 Базовая кинематическая система (BKS)

Базовая кинематическая система (BKS) состоит из трех расположенных под прямым углом осей (гео-оси), а также других осей (дополнительные оси) без геометрической связи.

Станки без кинематической трансформации

BKS и MCS всегда совпадают, если BKS может быть отображена на MCS без кинематической трансформации (например, 5-осевая трансформация, TRANSMIT / TRACYL / TRAANG).

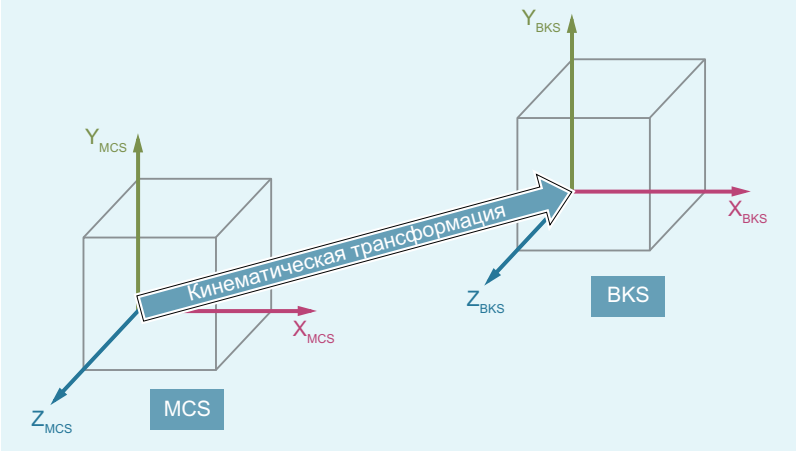
У таких станков имена осей станка и гео-осей могут быть идентичными.



Станки с кинематической трансформацией

BKS и MCS не совпадают, если BKS отображается на MCS с кинематической трансформацией (например, 5-осевая трансформация, TRANSMIT / TRACYL / TRAANG).

У таких станков имена осей станка и гео-осей должны быть различными.

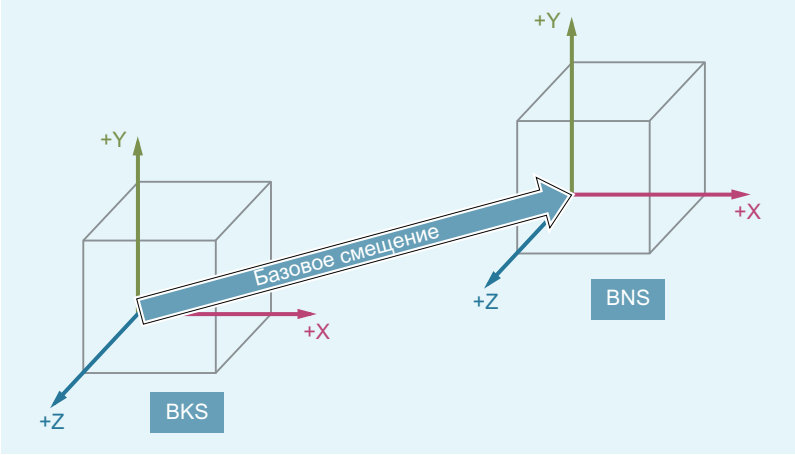


Кинематика станка

Деталь всегда программируется в двух- или трехмерной прямоугольной системе координат (WCS). Но для изготовления этих деталей все чаще используются станки с круговыми осями или расположенными не под прямым углом линейными осями. Кинематическая трансформация преобразует запрограммированные в WCS координаты (прямоугольные) в реальные движения осей станка.

3.1.4.3 Базовая система нулевой точки (BNS)

Базовая система нулевой точки (BNS) получается из базовой кинематической системы посредством базового смещения.



Базовое смещение

Базовое смещение описывает преобразование координат между BKS и BNS. С его помощью можно, например, установить нулевую точку паллеты.

Базовое смещение состоит из:

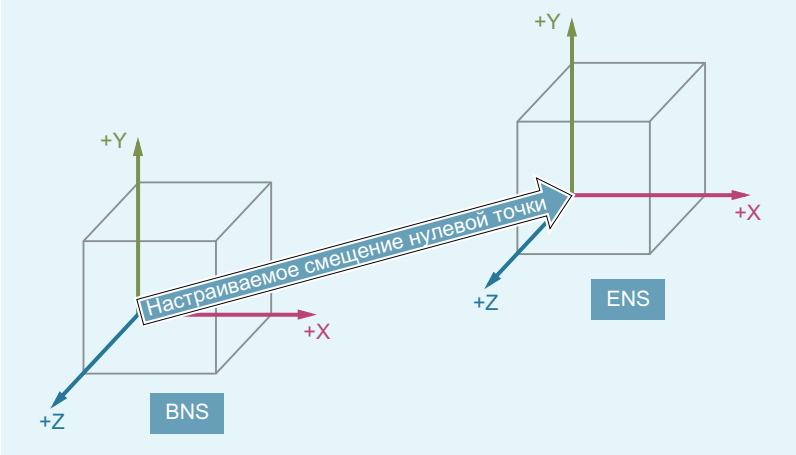
- Внешнее смещение нулевой точки
- Смещение DRF
- Наложенное движение
- Связанные системные фреймы
- Связанные базовые фреймы

3.1.4.4 Настраиваемая система нулевой точки (ENS)

Настраиваемое смещение нулевой точки

Через настраиваемое смещение нулевой точки из базовой системы нулевой точки (BNS) получается "Настраиваемая система нулевой точки" (ENS).

Настраиваемые смещения нулевой точки активируются в программе ЧПУ с помощью команд G G54 ... G57 и G505 ... G599.



Если нет активных программируемых трансформации координат (фреймы), то "Настраиваемой системой нулевой точки" является система координат детали (WCS).

Программируемые трансформации координат (фреймы)

Иногда имеет смысл или требуется переместить в программе ЧПУ первоначально выбранную систему координат детали (или "Настраиваемую систему нулевой точки") на другую позицию и при необходимости повернуть, отразить и/или масштабировать. Это осуществляется через программируемые трансформации координат (фреймы) (Страница 314).

Примечание

Программируемые трансформации координат (фреймы) всегда относятся к "Настраиваемой системе нулевой точки".

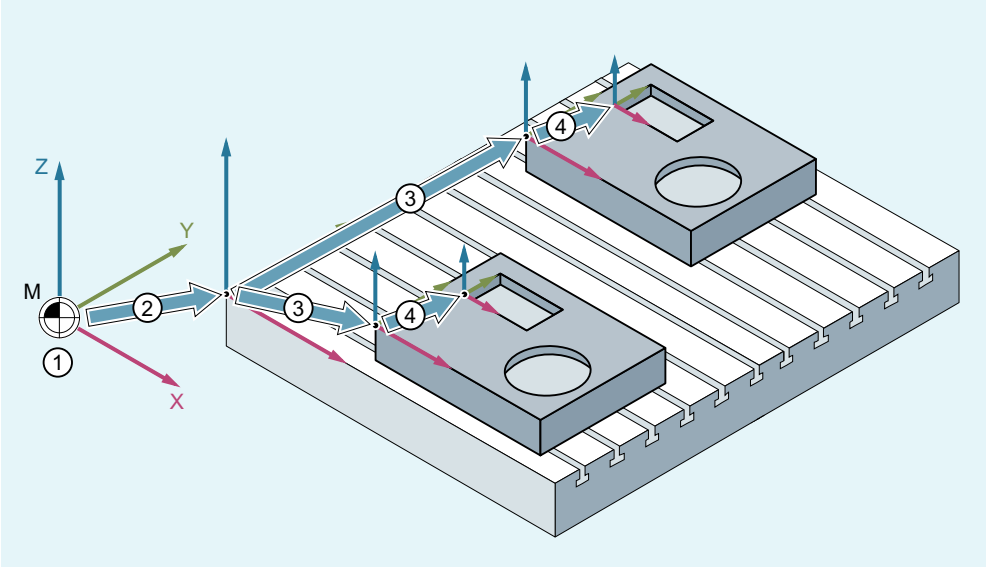
3.1.4.5 Система координат детали (WCS)

В системе координат детали (WCS) описывается геометрия детали. Т.е: данные в программе ЧПУ относятся к системе координат детали.

Система координат детали всегда является декартовой системой координат и соответствует определенной детали.

3.1.4.6 Как связаны различные системы координат?

В следующем примере показаны связи различных систем координат:



① Кинематическая трансформация не активна. Это означает, что система координат станка и базовая кинематическая система совпадают.

② Через базовое смещение получается базовая система нулевой точки (BNS) с нулевой точкой палеты.

③ Через настраиваемое смещение нулевой точки G54 или G55 определяется "Настраиваемая система нулевой точки" (ENS) для детали 1 или детали 2.

④ Через программируемую трансформацию координат получается система координат детали (WCS).

3.2 Основы программирования ЧПУ

Примечание

Директивным руководством по программированию ЧПУ является DIN 66025.

3.2.1 Наименование программы ЧПУ

Правила

Любой программе ЧПУ при создании необходимо присвоить имя (идентификатор). Имя программы можно выбирать произвольно, соблюдая следующие правила:

- Разрешенные символы:
  - Буквы: A ... Z, a ... z
  - Цифры: 0 ... 9
  - Знак подчеркивания: _
- Первыми двумя символами должны быть две буквы или знак подчеркивания с буквой после него.

  Примечание

  Если это условие выполнено, программа ЧПУ может быть вызвана из другой программы как подпрограмма путем указания имени программы. Если имя программы начинается с цифр, вызов подпрограммы возможен только через оператор CALL.

- Максимальная длина: 24 символа

Примечание

Прописные / строчные буквы

В языке ЧПУ SINUMERIK **нет** различия между прописными и строчными буквами.

Примечание

Недопустимые имена программ

Во избежание проблем с приложениями Windows **запрещено** использовать следующие имена программ:

- CON, PRN, AUX, NUL
- COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9
- LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8, LPT9

Другие ограничения: см. "Имя (Страница 391)".

Расширения внутри СЧПУ

К присвоенному при создании программы имени программы СЧПУ добавляет префикс и постфикс:

- Префикс: _N_
- Постфикс:
  - Главные программы: _MPF
  - Подпрограммы: _SPF

Файлы в формате RS232

Созданные на внешнем устройстве программные файлы, которые нужно загружать через интерфейс V.24, должны иметь формат RS232.

Для программного имени файла в формате RS232 действуют следующие дополнительные правила:

- Первый символ: %
- В конце — расширение файла из четырех символов: _xxx

Примеры:

- %_N_WELLE123_MPF
- %Flansch3_MPF

3.2.2 Структура и содержание программы ЧПУ

3.2.2.1 Кадры и компоненты кадров

Кадры

Программа ЧПУ состоит из последовательности кадров ЧПУ. Каждый кадр содержит данные для выполнения рабочей операции при обработке детали.

Компоненты кадра

Кадры ЧПУ состоят из следующих компонентов:

- Команды (операторы) по DIN 66025
- Элементы высокоуровневого языка программирования ЧПУ

Команды по DIN 66025

Команды по DIN 66025 состоят из символа адреса и цифры или последовательности цифр, представляющей арифметическое значение.

Символ адреса (адрес)

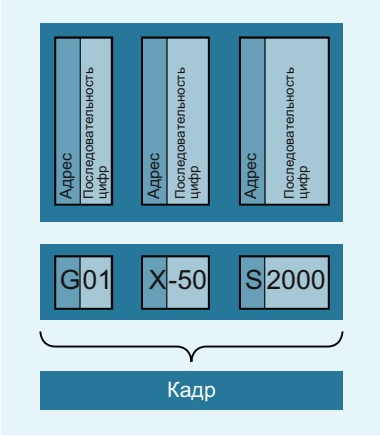
Символ адреса (чаще всего буква) определяет значение команды.

Примеры:

Символ адреса	Объяснение
G	G-команда (функция перемещения)
X	Информация о длине перемещения для оси X
S	Число оборотов шпинделя

Последовательность цифр

Последовательность цифр — это присвоенное символу адреса значение. Последовательность цифр может включать знак и десятичную точку, при этом знак всегда стоит между буквой адреса и последовательностью цифр. Положительный знак (+) и вводные нули (0) не записываются.



Элементы высокоуровневого языка программирования ЧПУ

Так как набора команд по DIN 66025 более недостаточно для программирования сложных процессов обработки на современных станках, он был дополнен элементами высокоуровневого языка программирования ЧПУ.

К ним, среди прочего, относятся:

- Команды высокоуровневого языка программирования ЧПУ
  В отличие от команд по DIN 66025 команды высокоуровневого языка программирования ЧПУ состоят из нескольких букв адреса, например:
  - `OVR` для коррекции скорости (процентовка)
  - `SPOS` для позиционирования шпинделя
- Идентификаторы (определенные имена) для:
  - системных переменных
  - определенных пользователем переменных
  - подпрограмм
  - кодовых слов
  - меток перехода
  - макросов

  Примечание

  Идентификатор должен быть однозначным и не может использоваться для различных объектов.
- Операторы сравнения
- Логические операторы
- Функции вычисления
- Управляющие структуры

Принцип действия команд

Команды действуют либо модально, либо покадрово:

- Модально
  Модально действующие команды сохраняют свою значимость с запрограммированным значением до тех пор (во всех последующих кадрах), пока:
  - под той же командой не будет запрограммировано новое значение.
  - не будет запрограммирована команда, отменяющая действовавшую до этого команду.
- Покадрово
  Действующие покадрово команды действуют только в том кадре, в котором они программируются.

Конец программы

Последний кадр в последовательности операций содержит специальное слово для конца программы: `M2`, `M17` bzw. `M30`.

3.2.2.2 Правила для кадра

Начало кадра

Кадры ЧПУ могут быть обозначены в начале кадра номерами кадров. Они состоят из символа "N" и положительного целого числа, например:
`N40 ...`

Последовательность номеров кадров может быть любой, но рекомендуется растущая последовательность номеров кадров.

Примечание

Для получения однозначного результата поиска номера кадров внутри одной программы должны быть однозначными.

Конец кадра

Кадр завершается символом LF (LINE FEED = новая строка).

Примечание

Запись символа LF не требуется. Он автоматически создается через переключение строки.

Длина кадра

Один кадр может содержать макс. **512 символов** (включая комментарий и символ конца кадра LF).

Примечание

Обычно в текущей индикации кадра на дисплее показываются три кадра с макс. 66 символами каждый. Комментарии также показываются. Сообщения показываются в отдельном окне сообщений.

Последовательность операторов

Для большей наглядности структуры кадра, операторы в кадре должны быть расположены в следующей последовательности:
`N... G... X... Y... Z... F... S... T... D... M... H...`

Адрес	Значение
`N`	Адрес номера кадра
`G`	Функция перемещения
`X,Y,Z`	Информация о длине перемещения
`F`	Подача

S	Скорость
T	Инструмент
D	Номер коррекции инструмента
M	Дополнительная функция
H	Вспомогательная функция

Примечание

Некоторые адреса могут использоваться несколько раз в одном кадре, например:

G..., M..., H...

3.2.2.3 Присвоения значений

Адресам могут присваиваться значения. При этом действуют следующие правила:

- Символ "=" должен быть записан между адресом и значением, если:
  - адрес состоит более чем из одной буквы.
  - значение состоит более чем из одной постоянной.

  Символ "=" не нужен, если адресом является одна единственная буква и значение состоит только из одной постоянной.
- Знаки разрешены.
- Разделительные символы после буквы адреса разрешены.

Примеры:

X10	Присваивание значения (10) адресу X, "=" не требуется
X1=10	Присваивание значения (10) адресу (X) с цифровым расширением (1), "=" требуется
X=10*(5+SIN(37.5))	Присваивание значения через цифровое выражение, "=" требуется

Примечание

За цифровым расширением всегда должен следовать один из специальных символов "=", "(", "[", ")", "]", "," или оператор, чтобы отличать адрес с цифровым расширением от буквы адреса со значением.

3.2.2.4 Комментарии

Для того, чтобы сделать программу ЧПУ более понятной, кадры ЧПУ могут быть снабжены комментариями.

Комментарий стоит в конце кадра и отделяется от программного блока кадра ЧПУ точкой с запятой (";").

Пример 1:

Программный код	Комментарий
`N10 G1 F100 X10 Y20`	; Комментарий для пояснения кадра ЧПУ

Пример 2:

Программный код	Комментарий
`N10`	; Фирма `G&S, заказ № 12A71`
`N20`	; Программа создана г-ном Мюллером, отдел `TV 4, 21.11.94`
`N50`	; Деталь № 12, корпус для погружного насоса типа `TP23A`

Примечание

Комментарии сохраняются и появляются при выполнении программы в актуальной индикации кадра.

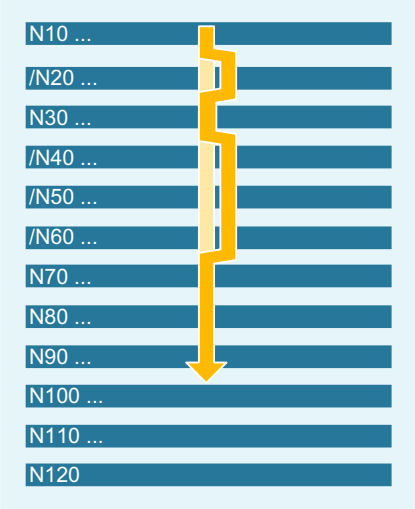
3.2.2.5 Пропуск кадров

Кадры ЧПУ, которые не должны выполняться при каждом проходе программы, могут быть при выполнении пропущены. Эта функция используется, например, при тестировании или отладке новых программ.

Пропускаемые кадры

Кадры, которые должны быть пропущены, обозначаются символом "/" (косая черта) перед номером кадра. Могут пропускаться и несколько последовательных кадров. Операторы в пропущенных кадрах не исполняются, программа продолжается на соответствующем следующем не пропущенном кадре.

Пример:



Программный код	Комментарий
N10 ...	; выполняется
/N20 ...	; пропущен
N30 ...	; выполняется
/N40 ...	; пропущен
/N50 ...	; пропущен
/N60 ...	; пропущен
N70 ...	; выполняется
...	

Уровни пропуска

Кадрам могут быть присвоены уровни пропуска (макс. 10), которые могут быть активированы через интерфейс или через программу электроавтоматики.

Присвоение выполняется в программе ЧПУ через предустановку косой черты с последующим номером уровня пропуска. На один кадр может быть указан только один уровень пропуска.

Пример:

Программный код	Комментарий
/ ...	; Кадр пропускается (1-й уровень пропуска)
/0 ...	; Кадр пропускается (1-й уровень пропуска)
/1 N010...	; Кадр пропускается (2-й уровень пропуска)
/2 N020...	; Кадр пропускается (3-й уровень пропуска)
...	
/7 N100...	; Кадр пропускается (8-й уровень пропуска)
/8 N080...	; Кадр пропускается (9-й уровень пропуска)
/9 N090...	; Кадр пропускается (10-й уровень пропуска)

Примечание

Изменения пропускаемых уровней возможны только в состояниях Stop/Reset СЧПУ.

Примечание

Количество уровней пропуска, которые могут использоваться, зависит от машинных данных индикации.

Примечание

Пропуск кадров программы активен и при поиске кадров.

Примечание

Изменяемые программные процессы могут создаваться и посредством использования системных и пользовательских переменных для условных переходов.

3.3 Создание программы ЧПУ

3.3.1 Базовый принцип

При создании программы ЧПУ само программирование, т.е. преобразование отдельных рабочих операций в язык ЧПУ, часто является лишь небольшой частью работы по программированию.

Перед программированием необходимо осуществить планирование и подготовку рабочих операций. Чем точнее будет спланировано начало и структура программы ЧПУ, тем быстрее и проще будет осуществляться само программирование и тем более наглядной и менее подверженной ошибкам будет готовая программа ЧПУ. Преимущество наглядных программ особо проявляется тогда, когда позднее необходимо вносить изменения.

Так как не каждая часть выглядит точно также, как другая, то не имеет смысла, создавать каждую программу точно по одному и тому же методу. Но для большинства случаев описанный ниже принцип является целесообразным.

Принцип действий

1. **Подготовка чертежа детали**
   - Определение нулевой точки детали
   - Нанесение системы координат
   - Вычисление возможно отсутствующих координат
2. **Определение процесса обработки**
   - Когда будут использоваться какие инструменты и для обработки каких контуров?
   - В какой последовательности будут изготовляться отдельные элементы детали?
   - Какие отдельные элементы повторяются (возможно в повернутом виде) и должны быть сохранены в подпрограмме?
   - Имеются ли в других программах обработки деталей или подпрограммах контуры деталей, которые могут быть использованы повторно для актуальной детали?
   - Где целесообразны или необходимы смещения нулевой точки, вращение, отражение, масштабирование (концепция фрейма)?
3. **Создание технологической карты**
   Определить поочередно все процессы обработки станка, к примеру:
   - Движения ускоренным ходом для позиционирования
   - Смена инструмента
   - Определение плоскости обработки
   - Свободный ход для дополнительного измерения
   - Включение/выключение шпинделя, СОЖ
   - Вызов данных инструмента
   - Подача
   - Коррекция траектории
   - Подвод к контуру
   - Отвод от контура
   - и т.д.
4. **Перевод рабочих операций на язык программирования**
   - Запись каждой отдельной операции как кадра ЧПУ (или кадров ЧПУ).
5. **Соединение всех отдельных операций в одной программе**

3.3.2 Доступные символы

Для создания программ ЧПУ имеются следующие символы:

- Прописные буквы:
  A, B, C, D, E, F, G, H, I, J, K, L, M, N,(O),P, Q, R, S, T, U, V, W, X, Y, Z
- Строчные буквы:
  a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z
- Цифры:
  0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Специальные символы:
  См. таблицу ниже!

Специальные символы	**Значение**
%	Символ начала программы (только для создания программы на внешнем PC)
(	Заключение в скобки параметров или в выражениях
)	Заключение в скобки параметров или в выражениях
[	Заключение в скобки адресов или индексов поля
]	Заключение в скобки адресов или индексов поля
<	меньше
>	больше
:	Главный кадр, конец метки, связывающий оператор
=	Присваивание, часть равенства
/	Деление, пропуск кадра
*	Умножение
+	Сложение
-	Вычитание, отрицательный знак
"	Кавычки, идентификация для цепочки символов
'	Апостроф, идентификация для специальных числовых данных: шестнадцатеричные, двоичные
$	Системная идентификация переменных
s_	Символ подчеркивания, относится к буквам
?	Зарезервировано
!	зарезервировано
.	Десятичная точка
,	Запятая, знак разделения параметров
;	Начало комментария
&	Символ форматирования, то же действие, что и пробел
LF	Конец кадра
Табулятор	Разделительный символ
Пробел	Разделительный символ (пробел)

Примечание

Не путать букву "O" с числом "0".

Примечание

Прописные и строчные буквы не различаются (исключение: вызов инструмента).

Примечание

Скрытые специальные символы обрабатываются как символы пробела.

3.3.3 "Шапка" программы

Кадры ЧПУ, стоящие перед кадрами движения для изготовления контура детали, обозначаются как "шапка" программы.

"Шапка" программы содержит информацию / операторы для:

- Смены инструмента
- Коррекции на инструмент
- Движения шпинделя
- Регулирования подачи
- Геометрических установок (смещение нулевой точки, выбор рабочей плоскости)

"Шапка" программы при токарной обработке

Пример ниже показывает типичную структуру "шапки" программы ЧПУ для токарной обработки:

Программный код	Комментарий
`N10 G0 G153 X200 Z500 T0 D0`	; Отвести инструментальный суппорт до поворота инструментального револьвера.
`N20 T5`	; Повернуть инструмент 5.
`N30 D1`	; Активировать блок данных резцов инструмента.
`N40 G96 S300 LIMS=3000 M4 M8`	; Постоянная скорость резания (`Vc) = 300` м/мин, ограничение числа оборотов = 3000 об/мин, левое направление вращения, охлаждение вкл.
`N50 DIAMON`	; Ось `X` программируется в диаметре.
`N60 G54 G18 G0 X82 Z0.2`	; Вызвать смещение нулевой точки и рабочую плоскость, подвести в стартовую позицию.
...	

"Шапка" программы при фрезеровании

Пример ниже показывает типичную структуру "шапки" программы ЧПУ для фрезерной обработки:

Программный код	Комментарий
`N10 T="SF12"`	; альтернатива: `T123`
`N20 M6`	; Запустить смену инструмента
`N30 D1`	; Активировать блок данных резцов инструмента
`N40 G54 G17`	; Смещение нулевой точки и рабочая плоскость
`N50 G0 X0 Y0 Z2 S2000 M3 M8`	; Движение подвода к детали, шпиндель и СОЖ вкл
...	

Если работа осуществляется с ориентацией инструмента / трансформацией координат, то в начале программы необходимо отключить возможно еще активные трансформации:

Программный код	Комментарий
`N10 CYCLE800()`	; Сброс повернутой плоскости
`N20 TRAFOOF`	; Сброс `TRAORI, TRANSMIT, TRACYL, ...`
...	

3.3.4 Примеры программы

3.3.4.1 Пример 1: Первые шаги программирования

Пример программы 1 служит для выполнения и проверки первых шагов программирования.

Порядок действий

1. Создание новой программы обработки детали (имя)
2. Редактирование программы обработки детали
3. Выбор программы обработки детали
4. Активация отдельного кадра
5. Запуск программы обработки детали

Примечание

Чтобы программа работала на станке, должны быть правильно установлены машинные данные (→ см. указания изготовителя станка!).

Примечание

При проверке программы могут возникать аварийные сообщения, которые необходимо сбросить.

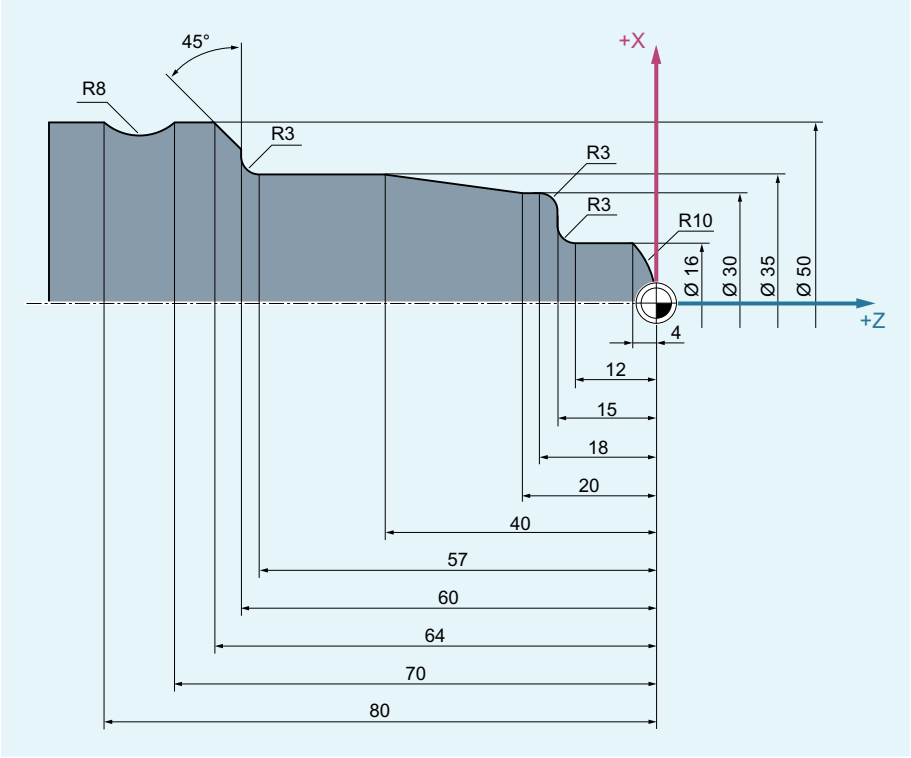
Программа ЧПУ

Программный код	Комментарий
`N10 MSG("ЭТО МОЯ ПРОГРАММА ЧПУ")`	; Вывести сообщение "ЭТО МОЯ ПРОГРАММА ЧПУ" в строке сообщений
`N20 F200 S900 T1 D2 M3`	; Подача, шпиндель, инструмент, коррекция инструмента, шпиндель вправо
`N30 G0 X100 Y100`	; Подвод к позиции ускоренным ходом
`N40 G1 X150`	; Прямоугольник с подачей, прямая в `X`
`N50 Y120`	; Прямая в `Y`
`N60 X100`	; Прямая в `X`
`N70 Y100`	; Прямая в `Y`
`N80 G0 X0 Y0`	; Отвод ускоренным ходом
`N100 M30`	; Конец кадра

3.3.4.2 Пример 2: Программа ЧПУ для токарной обработки

Пример программы 2 предназначен для обработки детали на токарном станке. Он содержит программирование радиуса и коррекцию радиуса инструмента.

Габаритный чертеж детали



Программа ЧПУ

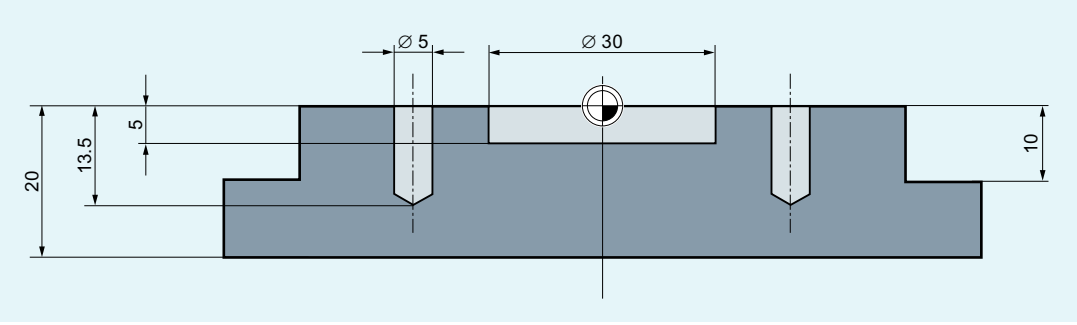
Программный код	Комментарий
N5 G0 G53 X280 Z380 D0	; Стартовая точка
N10 TRANS X0 Z250	; Смещение нулевой точки
N15 LIMS=4000	; Ограничение скорости (G96)
N20 G96 S250 M3	; Выбор постоянной скорости резания
N25 G90 T1 D1 M8	; Выбор инструмента и коррекции
N30 G0 G42 X-1.5 Z1	; Подвод инструмента в рабочее положение с коррекцией радиуса инструмента
N35 G1 X0 Z0 F0.25	
N40 G3 X16 Z-4 I0 K-10	; Обтачивание радиуса 10
N45 G1 Z-12	
N50 G2 X22 Z-15 CR=3	; Обтачивание радиуса 3
N55 G1 X24	
N60 G3 X30 Z-18 I0 K-3	; Обтачивание радиуса 3
N65 G1 Z-20	
N70 X35 Z-40	
N75 Z-57	
N80 G2 X41 Z-60 CR=3	; Обтачивание радиуса 3
N85 G1 X46	
N90 X52 Z-63	
N95 G0 G40 G97 X100 Z50 M9	; Отключение коррекции радиуса инструмента и подвод к точке смены инструмента
N100 T2 D2	; Вызов инструмента и выбор коррекции
N105 G96 S210 M3	; Выбор постоянной скорости резания
N110 G0 G42 X50 Z-60 M8	; Подвод инструмента в рабочее положение с коррекцией радиуса инструмента
N115 G1 Z-70 F0.12	; Обтачивание диаметра 50
N120 G2 X50 Z-80 I6.245 K-5	; Обтачивание радиуса 8
N125 G0 G40 X100 Z50 M9	; Отвод инструмента и выключение коррекции радиуса инструмента
N130 G0 G53 X280 Z380 D0 M5	; Переход к точке смены инструмента
N135 M30	; Конец программы

3.3.4.3 Пример 3: Программа ЧПУ для фрезерной обработки

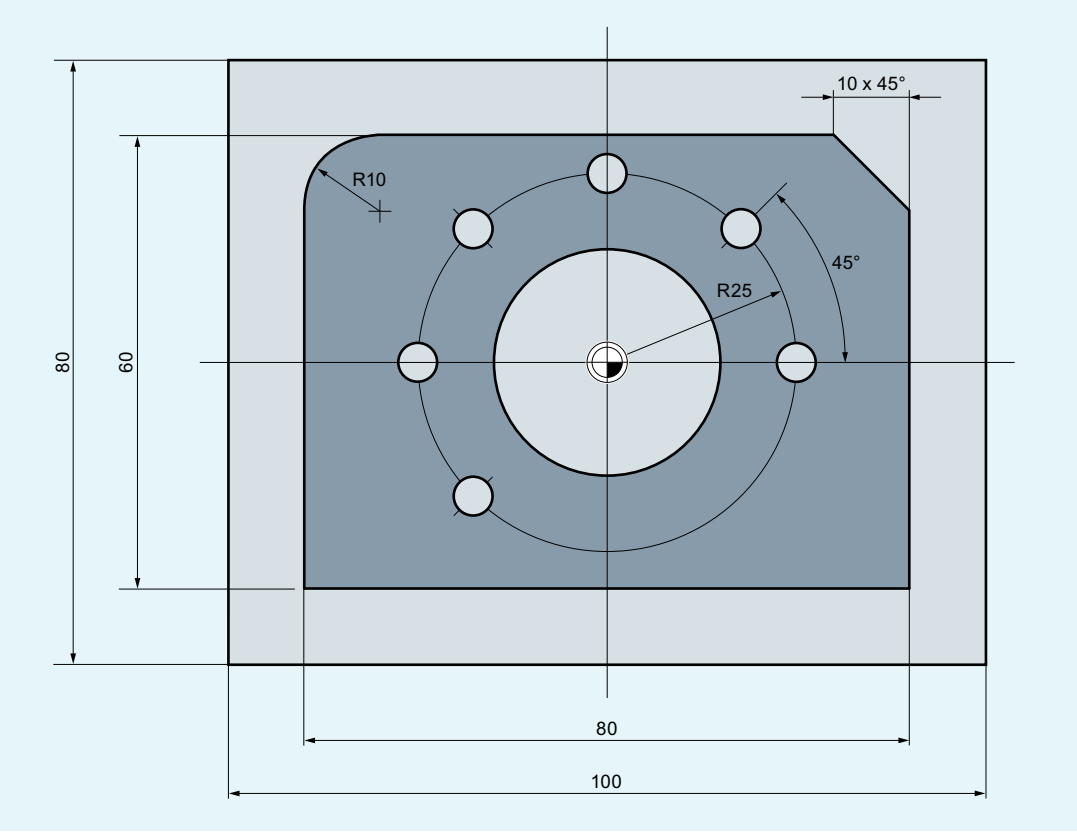
Пример программы 3 предназначен для обработки детали на вертикальном фрезерном станке. Он содержит фрезерование поверхности и боковых сторон, а также сверление.

Габаритный чертеж детали

Вид сбоку



Вид сверху



Программа ЧПУ

Программный код	Комментарий
`N10 T="PF60"`	; Предварительный выбор инструмента с именем `PF60`.
`N20 M6`	; Установить инструмент в шпиндель.

Программный код	Комментарий
N30 S2000 M3 M8	; Число оборотов, направление вращения, охлаждение вкл.
N40 G90 G64 G54 G17 G0 X-72 Y-72	; Первичные установки геометрии и подвод в стартовую точку.
N50 G0 Z2	; Ось Z на безопасное расстояние.
N60 G450 CFTCP	; Поведение при активной G41/G42.
N70 G1 Z-10 F3000	; Фреза на глубине контакта с подачей = 3000 мм/мин.
N80 G1 G41 X-40	; Включение коррекции радиуса фрезы.
N90 G1 X-40 Y30 RND=10 F1200	; Движение по контуру с подачей = 1200 мм/мин.
N100 G1 X40 Y30 CHR=10	
N110 G1 X40 Y-30	
N120 G1 X-41 Y-30	
N130 G1 G40 Y-72 F3000	; Выключение коррекции радиуса фрезы.
N140 G0 Z200 M5 M9	; Подъем фрезы, шпиндель + охлаждение выкл.
N150 T="SF10"	; Предварительный выбор инструмента с именем SF10.
N160 M6	; Установить инструмент в шпиндель.
N170 S2800 M3 M8	; Число оборотов, направление вращения, охлаждение вкл.
N180 G90 G64 G54 G17 G0 X0 Y0	; Первичные установки геометрии и подвод в стартовую точку.
N190 G0 Z2	
N200 POCKET4(2,0,1,-5,15,0,0,0,0,0,800,1300,0,21,5,,,2,0.5)	; Вызов цикла фрезерования кармана.
N210 G0 Z200 M5 M9	; Подъем фрезы, шпиндель + охлаждение выкл.
N220 T="ZB6"	; Вызвать центровое сверло 6 мм.
N230 M6	
N240 S5000 M3 M8	
N250 G90 G60 G54 G17 X25 Y0	; Точный останов G60 из-за точного позиционирования.
N260 G0 Z2	
N270 MCALL CYCLE82(2,0,1,-2.6,,0)	; Модальный вызов цикла сверления.
N280 POSITION:	; Метка перехода для повторения.
N290 HOLES2(0,0,25,0,45,6)	; Образец позиции для схемы сверления.
N300 ENDLABEL:	; Конечный идентификатор для повторения.
N310 MCALL	; Сброс модального вызова.
N320 G0 Z200 M5 M9	
N330 T="SPB5"	; Вызвать спиральное сверло D5мм.
N340 M6	
N350 S2600 M3 M8	
N360 G90 G60 G54 G17 X25 Y0	
N370 MCALL CYCLE82(2,0,1,-13.5,,0)	; Модальный вызов цикла сверления.
N380 REPEAT POSITION	; Повторение описания позиции из центрования.
N390 MCALL	; Сброс цикла сверления.
N400 G0 Z200 M5 M9	
N410 M30	; Конец программы.

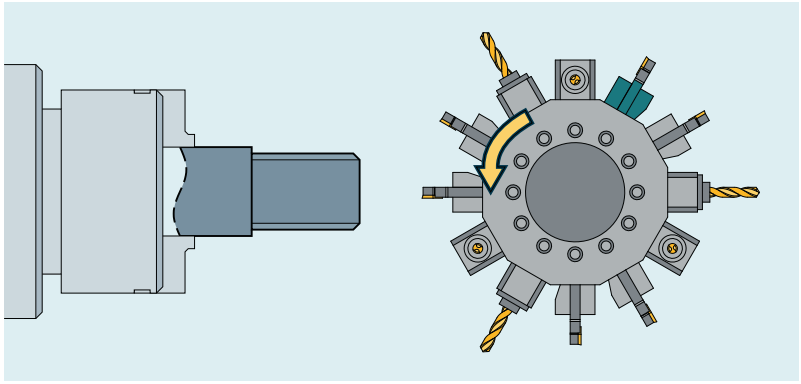
3.4 Смена инструмента

Тип смены инструмента

Примечание

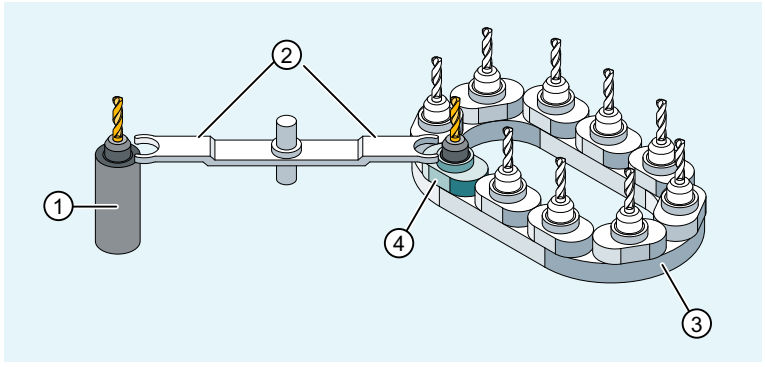
Тип смены инструмента устанавливает изготовитель станка при вводе в эксплуатацию.

Смена инструмента на токарных станках с револьверным магазином



В случае револьверных магазинов на токарных станках смена инструмента, т. е. поиск и установка, выполняются только с помощью команды T.

Смена инструмента на станках с цепными, дисковыми или плоскими магазинами



① Шпиндель
② Захват
③ Магазин (здесь: цепной магазин)
④ Место замены шпинделя

У цепных, дисковых и плоских магазинов процесс смены инструмента обычно подразделяется на два этапа:

1. При помощи команды T выполняется поиск инструмента в магазине.
2. Затем с помощью команды M выполняется установка инструмента в шпиндель.

Программирование смены инструмента при активном/неактивном управлении инструментом

Программирование смены инструмента при активном управлении инструментом отличается от программирования смены инструмента при неактивном управлении инструментом. Поэтому описаны оба варианта:

→ Смена инструмента при активном управлении инструментом (Страница 64)

→ Смена инструмента при неактивном управлении инструментом (Страница 68)

Программирование рабочей плоскости

Со сменой инструмента должна быть запрограммирована соответствующая рабочая плоскость (Страница 37) (первичная установка: G18). Так обеспечивается согласование коррекции длин инструмента с надлежащей осью.

Активация коррекции инструмента

Смена инструмента активирует значения коррекции инструмента, сохраненные под одним номером D (Страница 84).

3.4.1 Смена инструмента при активном управлении инструментом

Управление инструментом

Функция "Управление инструментом" обеспечивает наличие на станке в любое время надлежащего инструмента на правильном месте и соответствие относящихся к инструменту данных актуальной версии. Она также обеспечивает быструю установку инструмента, за счет контроля времени использования инструмента не допускает брака, а учет запасных инструментов не допускает простоя станка.

Имена инструментов

На станке с активным управлением инструментом для однозначной идентификации инструментам должны быть присвоены имена и номера (например, "Сверло", "3").

В этом случае инструмент можно вызывать по имени, например:
`T="Сверло"`

Примечание

Имя инструмента не должно содержать специальных символов.

3.4.1.1 Смена инструмента при активном управлении инструментом с функцией T=место

Программирование функции T=место обеспечивает прямую смену инструмента.

Использование

На токарных станках с револьверным магазином.

Синтаксис

Выбор инструмента
T=<Nr>
T=<имя>
T<n>=<Nr>
T<n>=<имя>

Отмена инструмента
T0

Объяснение

T=:	Адрес для выбора инструмента, включая смену инструмента и активацию коррекции инструмента Как присвоение значения возможно:	
	<Nr>:	Номер места в магазине
	<имя>:	Имя инструмента **Примечание:** при программировании имени инструмента обратить внимание на правильное написание (прописные/строчные буквы).
<n>:	Номер шпинделя как расширение адреса **Примечание:** Возможность программирования номера шпинделя как расширения адреса зависит от конфигурации станка (→ см. Указания изготовителя станка).	
T0:	Отмена инструмента (место в магазине свободно)	

Примечание

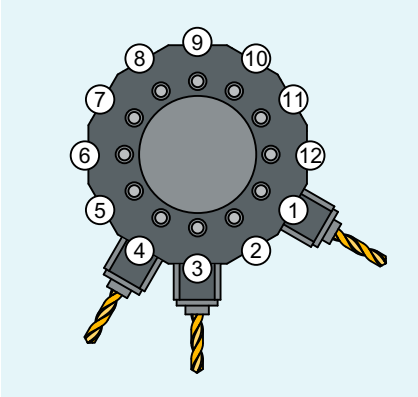
Если в магазине инструмента выбранное место не занято, то команда инструмента действует как T0. Выбор незанятого места в магазине может использоваться для позиционирования свободного места.

Пример

В револьверном магазине места с 1 по 12 имеют следующее распределение инструментов:

Место	Инструмент	Внутренний номер T	Состояние
1	Сверло, номер гнезда = 1	T15	заблокировано
2	свободно		
3	Сверло, номер гнезда = 2	T10	разрешено

Место	Инструмент	Внутренний номер T	Состояние
4	Сверло, номер гнезда = 3	T1	активно
5 ... 12	свободно		



① ... ⑫ Номера мест в магазине

В программе ЧПУ запрограммирован следующий вызов инструмента:

```
N10 T=1
```

Вызов обрабатывается следующим образом:

1. Рассматривается место в магазине 1 и при этом определяется идентификатор инструмента.
2. Управление инструментом определяет, что этот инструмент заблокирован и не может использоваться.
3. Поиск инструмента T="сверло" запускается согласно установленной стратегии поиска: "Искать активный инструмент, иначе использовать инструмент со следующим по величине номером гнезда".
4. В качестве готового к использованию инструмента найден:
   "Сверло" номер гнезда 3 (на месте 4 в магазине)
   Тем самым выбор инструмента завершен, и запускается смена инструмента.

Примечание

При стратегии поиска "Использовать первый доступный инструмент из группы" последовательность инструментов не определена. В этом случае выбирается T10 **или** T1.

Согласно стратегии поиска "Использовать первый инструмент со статусом 'активный' из группы" устанавливается T1.

3.4.1.2 Смена инструмента при активном управлении инструментом с M6

При помощи программирования команды T выбирается инструмент. Инструмент активен только при M6 (включая коррекцию инструмента).

Использование

На станках с цепными, дисковыми или плоскими магазинами.

Синтаксис

Выбор инструмента
```
T=<Nr>
T=<имя>
T<n>=<Nr>
T<n>=<имя>
```

Смена инструмента
```
M6
```

Отмена инструмента
```
T0
```

Объяснение

T=:	Адрес выбора инструмента Как присвоение значения возможно:	
	<Nr>:	Номер места в магазине
	<имя>:	Имя инструмента **Примечание:** при программировании имени инструмента обратить внимание на правильное написание (прописные/строчные буквы).
<n>:	Номер шпинделя как расширение адреса **Примечание:** Возможность программирования номера шпинделя как расширения адреса зависит от конфигурации станка (→ см. Указания изготовителя станка).	
M6:	Функция M для смены инструмента (согласно DIN 66025) С помощью M6 активируются выбранный инструмент (T...) и коррекция инструмента (D...).	
T0:	Отмена инструмента (место в магазине свободно)	

Пример

Программный код	Комментарий
N10 T=1 M6	; Установка инструмента на месте 1 магазина.
N20 D1	; Выбор коррекции длин инструмента.
N30 G1 X10 ...	; Работа с инструментом T=1.
...	

Программный код	Комментарий
`N70 T="Сверло"`	; Предварительный выбор инструмента с именем "Сверло".
`N80 ...`	; Работа с инструментом `T=1`.
...	
`N100 M6`	; Установка инструмента `T="Сверло"`.
`N140 D1 G1 X10 ...`	; Работа с инструментом `T="Сверло"`.
...	

3.4.2 Смена инструмента при неактивном управлении инструментом

3.4.2.1 Смена инструмента при неактивном управлении инструментом с номером T

Программирование номера T обеспечивает прямую смену инструмента.

Использование

На токарных станках с револьверным магазином.

Синтаксис

Выбор инструмента

```
T<Nr>
T=<Nr>
T<n>=<Nr>
```

Отмена инструмента

```
T0
T0=<Nr>
```

Объяснение

`T`:	Адрес для выбора инструмента, включая смену инструмента и активацию коррекции инструмента	
`<Nr>`:	Номер инструмента	
	Диапазон значений:	0 ... 32000
`<n>`:	Номер шпинделя как расширение адреса **Примечание:** Возможность программирования номера шпинделя как расширения адреса зависит от конфигурации станка (→ см. Указания изготовителя станка).	
`T0`:	Выключение активного инструмента	

Пример

Программный код	Комментарий
`N10 T1 D1`	; Установка инструмента `T1` и активация коррекции инструмента `D1`.
...	
`N70 T0`	; Отключить инструмент `T1`.
...	

3.4.2.2 Смена инструмента при неактивном управлении инструментом с M6

При помощи программирования команды T выбирается инструмент. Инструмент активен только при M6 (включая коррекцию инструмента).

Использование

На станках с цепными, дисковыми или плоскими магазинами.

Синтаксис

Выбор инструмента
```
T<Nr>
T=<Nr>
T<n>=<Nr>
```

Смена инструмента
```
M6
```

Отмена инструмента
```
T0
T0=<Nr>
```

Объяснение

`T`:	Адрес выбора инструмента	
`<Nr>`:	Номер инструмента	
	Диапазон значений:	0 ... 32000
`<n>`:	Номер шпинделя как расширение адреса **Примечание:** Возможность программирования номера шпинделя как расширения адреса зависит от конфигурации станка (→ см. Указания изготовителя станка).	
`M6`:	Функция M для смены инструмента (согласно DIN 66025) С помощью M6 активируются выбранный инструмент (T...) и коррекция инструмента (D...).	
`T0`:	Выключение активного инструмента	

Пример

Программный код	Комментарий
`N10 T1 M6`	; Установка инструмента `T1`.
`N20 D1`	; Выбор коррекции длин инструмента.
`N30 G1 X10 ...`	; Работа с `T1`.
...	
`N70 T5`	; Предварительный выбор инструмента `T5`.
`N80 ...`	; Работа с `T1`.
...	
`N100 M6`	; Установка инструмента `T5`.
`N110 D1 G1 X10 ...`	; Работа с инструментом `T5`.
...	

3.4.3 Поведение при неправильном программировании T

Поведение при неправильном программировании T зависит от конфигурирования станка:

MD22562 TOOL_CHANGE_ERROR_MODE		
Бит	**Значение**	**Объяснение**
7	0	Первичная установка! При программировании T сразу же проверяется, знает ли ЧПУ номер T. Если это не так, то появляется ошибка.
	1	Запрограммированный номер T проверяется только после выбора D. Если ЧПУ не знает номера T, то при выборе D сигнализируется ошибка. Такое поведение желательно тогда, когда программирование T, например, должно вызвать и позиционирование и данные инструмента для этого отсутствуют (револьверный магазин).

3.5 Коррекции инструмента

3.5.1 Запрограммированный контур и траектория инструмента

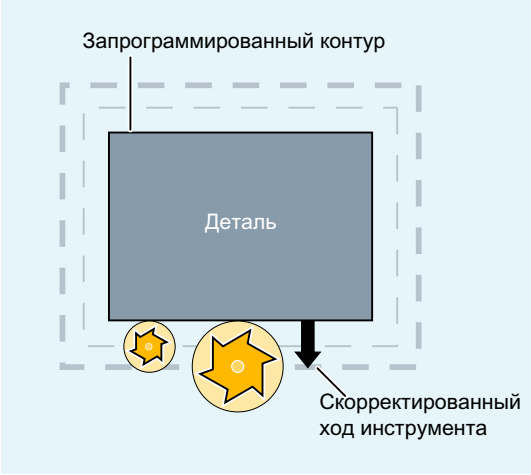
Программирование размеров детали осуществляется напрямую (например, по рабочему чертежу). Поэтому при создании программы нет необходимости учитывать такие данные инструмента, как, например, диаметр фрезы, положение кромок токарного резца (левый/правый токарный резец) и длины инструмента.

СЧПУ корректирует путь перемещения

При изготовлении детали управление путями перемещения инструмента в зависимости от соответствующей геометрии инструмента осуществляется так, что запрограммированный контур можно изготовить с помощью любого используемого инструмента.

Чтобы СЧПУ могла рассчитывать пути перемещения инструмента, данные инструмента должны быть внесены в память коррекций инструмента СЧПУ. Через программу ЧПУ вызывается только необходимый инструмент (T...) и необходимый блок данных коррекции (D...).

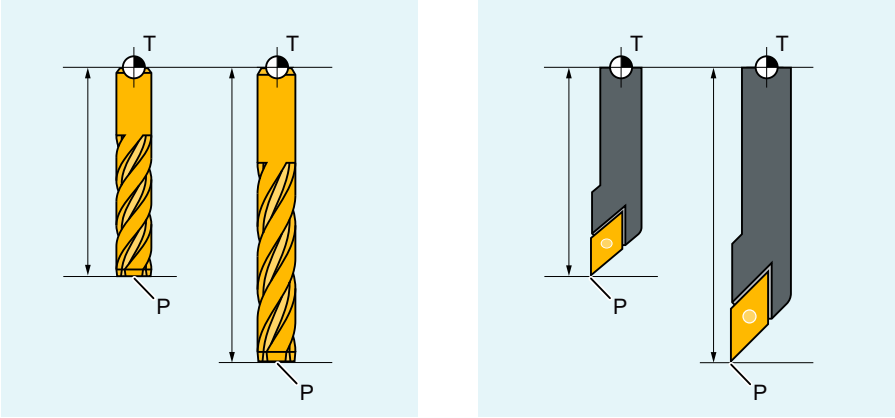
Во время обработки программы СЧПУ получает необходимые данные коррекции из памяти коррекции инструмента и индивидуально исправляет траекторию различных инструментов:



3.5.2 Коррекция длин инструмента

С помощью коррекции длин инструмента компенсируются разницы длин используемых инструментов.

Длина инструмента — это расстояние от исходной точки инструментального суппорта до острия инструмента:



T Исходная точка инструментального суппорта

P Острие инструмента

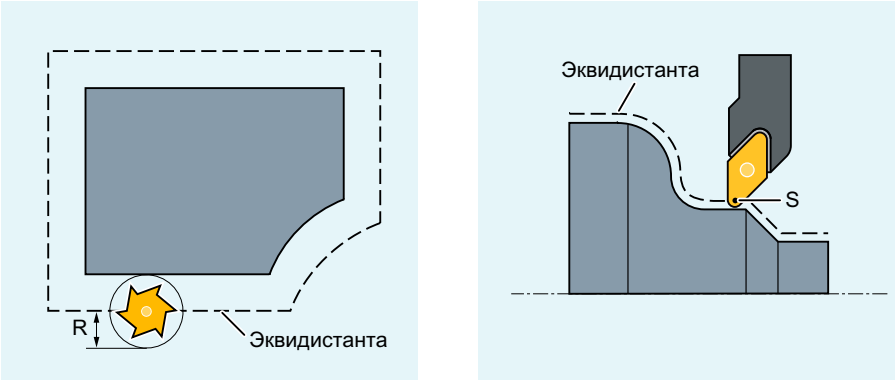
Эта длина измеряется и вместе с задаваемыми значениями износа вводится в память коррекции инструмента СЧПУ. По ним СЧПУ вычисляет движения перемещения в направлении подачи.

Примечание

Значение коррекции длины инструмента зависит от ориентации инструмента в пространстве.

3.5.3 Коррекция радиуса инструмента

Контур и траектория инструмента не идентичны. Центр фрезы или кромки резца должен перемещаться по эквидистанте к контуру в соответствии с радиусом инструмента (траектория центра инструмента). Для этого система управления во время выполнения программы смещает запрограммированную траекторию центра инструмента с учетом радиуса активного инструмента (память коррекций инструмента) таким образом, чтобы резец инструмента перемещался точно по запрограммированному контуру.



R Радиус инструмента

S Центр резца

Коррекция радиуса инструмента детально описана в главе "Коррекции радиуса инструмента (Страница 260)".

См. также

Коррекция инструмента 2 1/2 D (CUT2D, CUT2DD, CUT2DF, CUT2DFD) (Страница 295)

3.5.4 Память коррекций инструмента

В памяти коррекций инструмента СЧПУ для каждого резца инструмента должны находиться следующие данные:

- Тип инструмента
- Положение резца
- Геометрические размеры инструмента (длина, радиус)

Эти данные вносятся как параметры инструмента (макс. 25). Какие параметры необходимы для инструмента, зависит от типа инструмента. Ненужным параметрам инструмента присваивается значение "ноль" (соответствует системной предустановке).

Примечание

Однажды внесенные в память коррекции значения учитываются при каждом вызове инструмента.

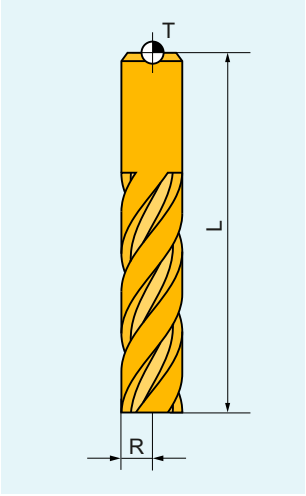
Тип инструмента

Тип инструмента (сверло или фреза или токарные инструменты) определяет, какие геометрические данные необходимы и как они вычисляются.

Положение резца

Положение резца описывает положение острия инструмента относительно центра резца. Положение резца вместе с радиусом резца необходимо для вычисления коррекции радиуса токарных инструментов (тип инструмента 5xx) (Страница 80).

Геометрические размеры инструмента (длина, радиус)



T	Исходная точка инструментального суппорта
R	Радиус инструмента
L	Длина инструмента

Геометрические размеры инструмента состоят из нескольких компонентов (геометрия, износ). По этим компонентам СЧПУ вычисляет результирующую величину (например, общая длина 1, общий радиус). Соответствующий общий размер начинает действовать при активации памяти коррекций.

Как эти значения перерассчитываются в осях, зависит от типа инструмента и текущей плоскости (G17/G18/G19).

3.5.5 Типы инструментов

3.5.5.1 Номер типа инструмента и группы инструментов

Каждому типу инструмента присвоен уникальный 3-значный номер. По первой цифре (разряд сотен в числе) выполняется согласование инструмента с одной из следующих технологий или групп инструментов:

Тип инструмента	Группа инструментов
1xy	Фреза (Страница 75)
2xy	Сверло (Страница 77)
3xy	зарезервировано
4xy	Шлифовальные инструменты (Страница 78)
5xy	Токарные инструменты (Страница 80)
6xy	зарезервировано
7xy	Специальные инструменты (Страница 82)

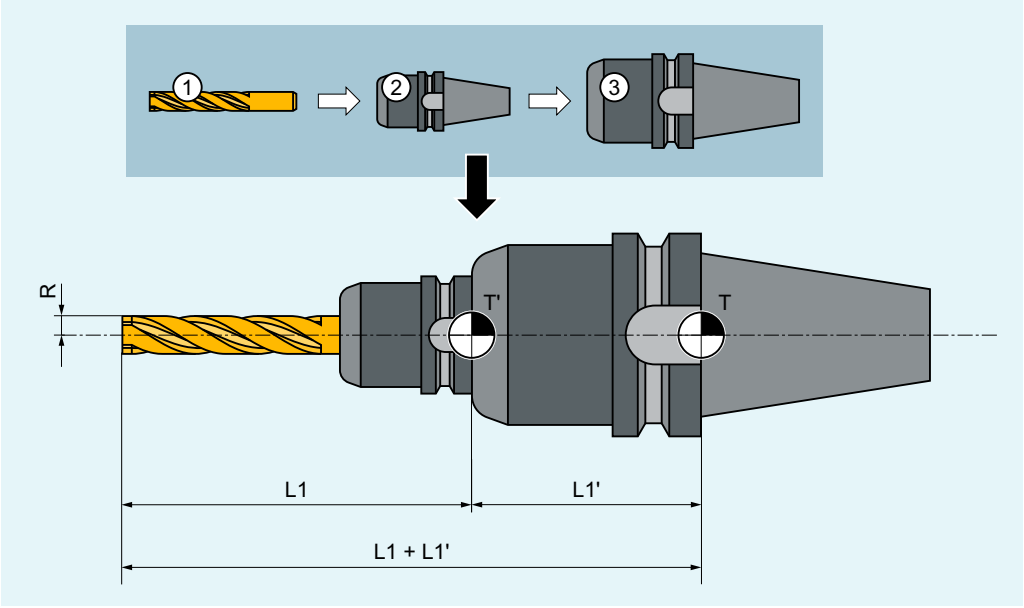
3.5.5.2 Фрезерный инструмент

В группе инструментов "Фрезерные инструменты" существуют следующие типы инструментов:

100	Фрезерный инструмент согласно CLDATA (Cutter Location Data)
110	Сферическая фреза
111	Цилиндрическая фреза для изготовления штампов
120	Концевая фреза без закругления углов
121	Концевая фреза с закруглением углов
130	Угловая фреза без закругления углов
131	Угловая фреза с закруглением углов
140	Торцовая фреза
145	Резьбовая фреза
150	Дисковая фреза
151	Пила
155	Усеченная коническая фреза без закругления углов
156	Усеченная коническая фреза с закруглением углов
157	Коническая фреза для изготовления штампов
160	Сверлильная резьбовая фреза

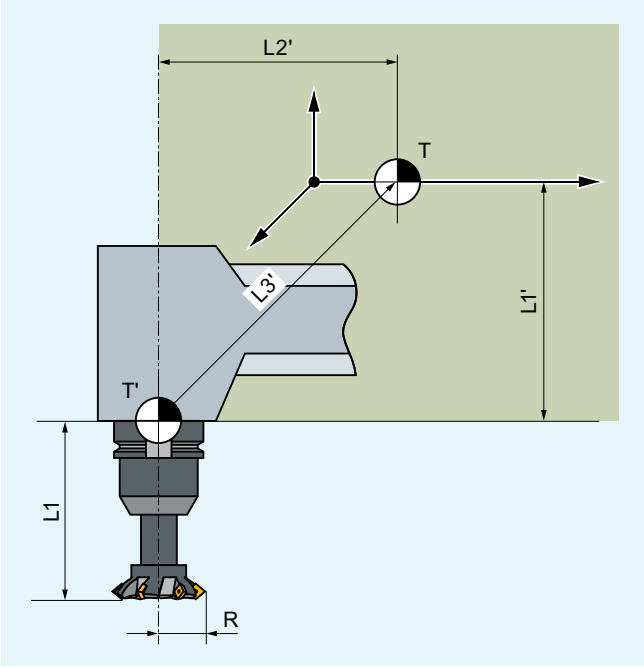
Параметры инструмента

На изображениях ниже показано, какие параметры инструмента вносятся в память коррекций в случае фрезерных инструментов:



①	Инструмент
②	Зажим инструмента
③	Адаптер инструмента
T	Исходная точка адаптера (для вставленного инструмента = исходная точка инструментального суппорта)
T	Исходная точка зажима инструмента
L1	Геометрия – длина 1
L1'	Размер адаптера – длина 1
L1 + L1'	Общая длина L1
R	Радиус

Параметры инструмента	Объяснение
$TC_DP1	Тип инструмента 1xy
$TC_DP3	Геометрия – длина 1
$TC_DP6	Геометрия – радиус
$TC_DP21	Размер адаптера – длина 1
• Значения износа согласно требованиям. • Установить прочие значения на 0.	



T Исходная точка инструментального суппорта
T Исходная точка зажима инструмента
L1 Геометрия – длина 1
R Радиус инструмента
L1' Базовый размер – длина 1
L2' Базовый размер – длина 2
L3' Базовый размер – длина 3

Параметры инструмента	Объяснение
$TC_DP1	Тип инструмента
$TC_DP3	Геометрия – длина 1
$TC_DP6	Геометрия – радиус
$TC_DP21	Базовый размер – длина 1
$TC_DP22	Базовый размер – длина 2
$TC_DP23	Базовый размер – длина 3
• Значения износа согласно требованиям. • Установить прочие значения на 0.	

3.5.5.3 Сверло

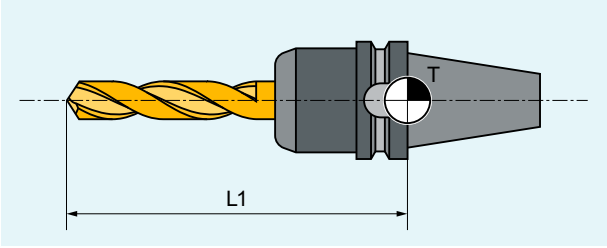
В группе инструментов "Сверло" представлены следующие типы инструментов:

№	Тип инструмента
200	Спиральное сверло
205	Сплошное сверло
210	Расточная оправка

№	Тип инструмента
220	Центровое сверло
230	Коническая зенковка
231	Цековка
240	Метчик основной резьбы
241	Метчик точной резьбы
242	Метчик резьбы Withworth
250	Развертка

Параметры инструмента

На следующем рисунке показано, какие параметры инструмента вносятся в память коррекций для сверл:



T Исходная точка инструментального суппорта

L1 Длина 1

Параметры инструмента	Объяснение
$TC_DP1	Тип инструмента
$TC_DP3	Геометрия – длина 1
• Значения износа согласно требованиям. • Установить прочие значения на 0.	

3.5.5.4 Шлифовальный инструмент

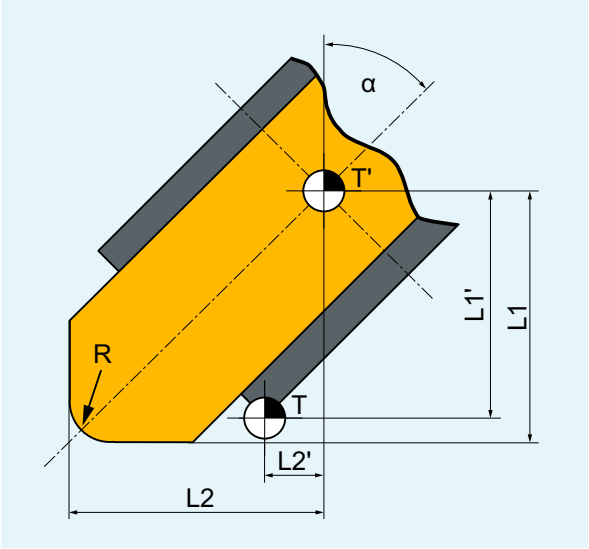
В группе инструментов "Шлифовальные инструменты" представлены следующие типы инструментов:

400	Периферийный шлифовальный круг
401	Периферийный шлифовальный круг с контролем
402	Периферийный шлифовальный круг без контроля без базового размера (управление инструментом)
403	Периферийный шлифовальный круг с контролем без базового размера для окружной скорости круга SUG
410	Планшайба
411	Планшайба (управление инструментом) с контролем
412	Планшайба (управление инструментом) без контроля

413	Планшайба с контролем без базового размера для окружной скорости круга SUG
490	Инструмент для правки

Параметры инструмента

На следующем рисунке показано, какие параметры инструмента вносятся в память коррекций для шлифовальных инструментов:



T Исходная точка инструментального суппорта
T Исходная точка зажима инструмента
L1 Геометрия – длина 1
L1' Базовый размер – длина 1
L2 Геометрия – длина 2
L2' Базовый размер – длина 2
R Радиус
α Угол наклонного круга

Спец. параметры резцов	**Объяснение**
$TC_DP1	Тип инструмента 4xy
$TC_DP2	Положение резца
$TC_DP3	Геометрия – длина 1
$TC_DP4	Геометрия – длина 2
$TC_DP6	Радиус
$TC_DP21	Базовый размер – длина 1
$TC_DP22	Базовый размер – длина 2
• Значения износа согласно требованиям. • Установить прочие значения на 0.	

Специфические для инструмента параметры	**Объяснение**
$TC_TPG1	Номер шпинделя
$TC_TPG2	Правило соединения [1)]
$TC_TPG3	Минимальный радиус круга
$TC_TPG4	Минимальная ширина круга
$TC_TPG5	Текущая ширина круга
$TC_TPG6	Максимальная частота вращения
$TC_TPG7	Максимальная окружная скорость
$TC_TPG8	Угол наклонного круга
$TC_TPG9	Номер параметра для вычисления радиуса
$TC_TPG_DRSPATH	Путь директории до программы правки
$TC_TPG_DRSPROG	Имя программы правки

1) Коррекции длин "геометрия", "износ" и "базовый размер" могут быть соответственно связаны для левой и правой коррекции круга. Это означает, что при изменении коррекции длин для левого резца, значения автоматически вносятся и для правого резца и наоборот.

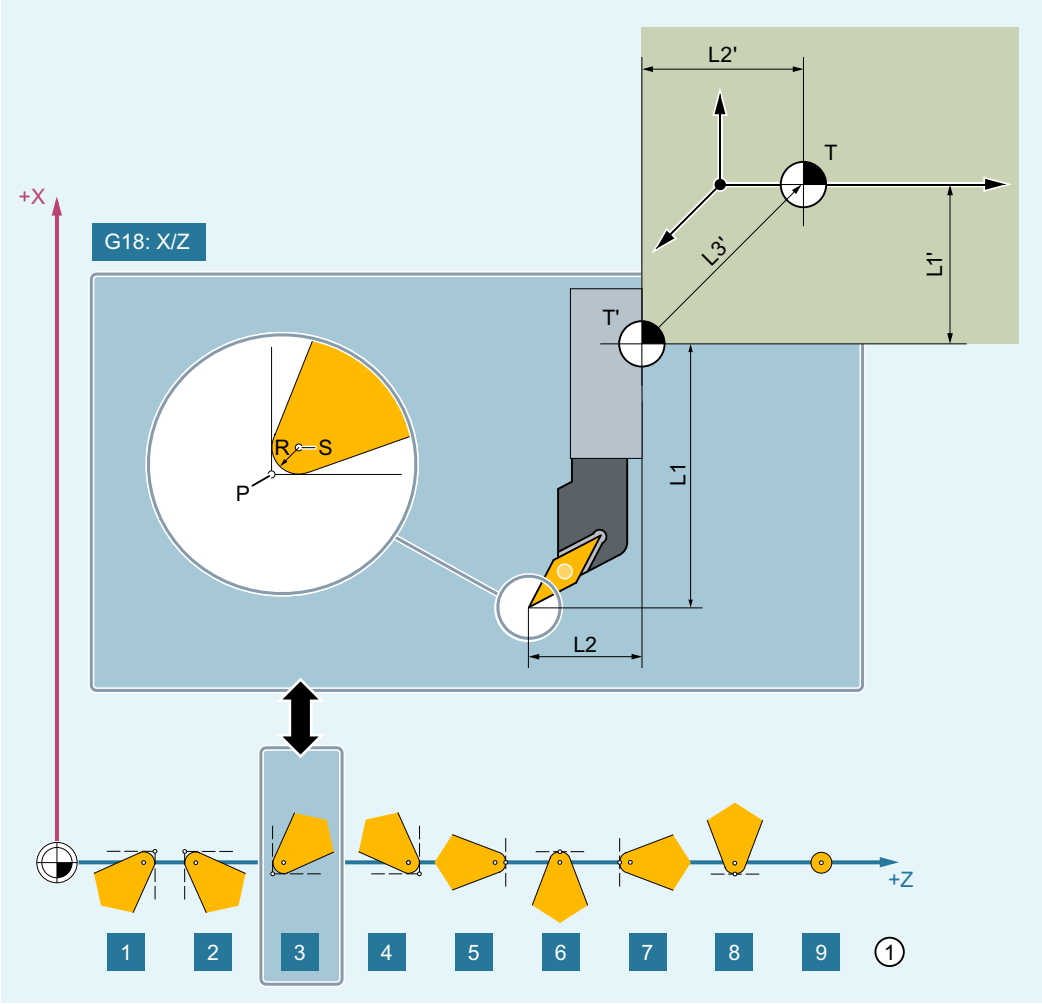
3.5.5.5 Токарный инструмент

В группе инструментов "Токарные инструменты" представлены следующие типы инструментов:

500	Обдирочный резец
510	Чистовой резец
520	Прорезной резец
530	Отрезной резец
540	Резьбовой резец
550	Фигурный резец / профильный резец (управление инструментом)
560	Коловорот (ECOCUT)
580	Измерительный щуп с параметром положения резцов

Параметры инструмента

На следующем рисунке показано, какие параметры инструмента вносятся в память коррекций для токарных инструментов:



①	Положение резца (1–9) при обработке позади центра вращения
P	Острие инструмента
S	Центр резца
R	Радиус резца
T	Исходная точка инструментального суппорта
T	Исходная точка зажима инструмента
L1	Геометрия – длина 1
L2	Геометрия – длина 2
L1'	Базовый размер – длина 1
L2'	Базовый размер – длина 2
L3'	Базовый размер – длина 3

Параметры инструмента	Объяснение
$TC_DP1	Тип инструмента
$TC_DP2	Положение резца
$TC_DP3	Геометрия – длина 1
$TC_DP4	Геометрия – длина 2
$TC_DP6	Геометрия – радиус (радиус резца)
$TC_DP21	Базовый размер – длина 1
$TC_DP22	Базовый размер – длина 2
$TC_DP23	Базовый размер – длина 3
• Значения износа согласно требованиям. • Установить прочие значения на 0.	

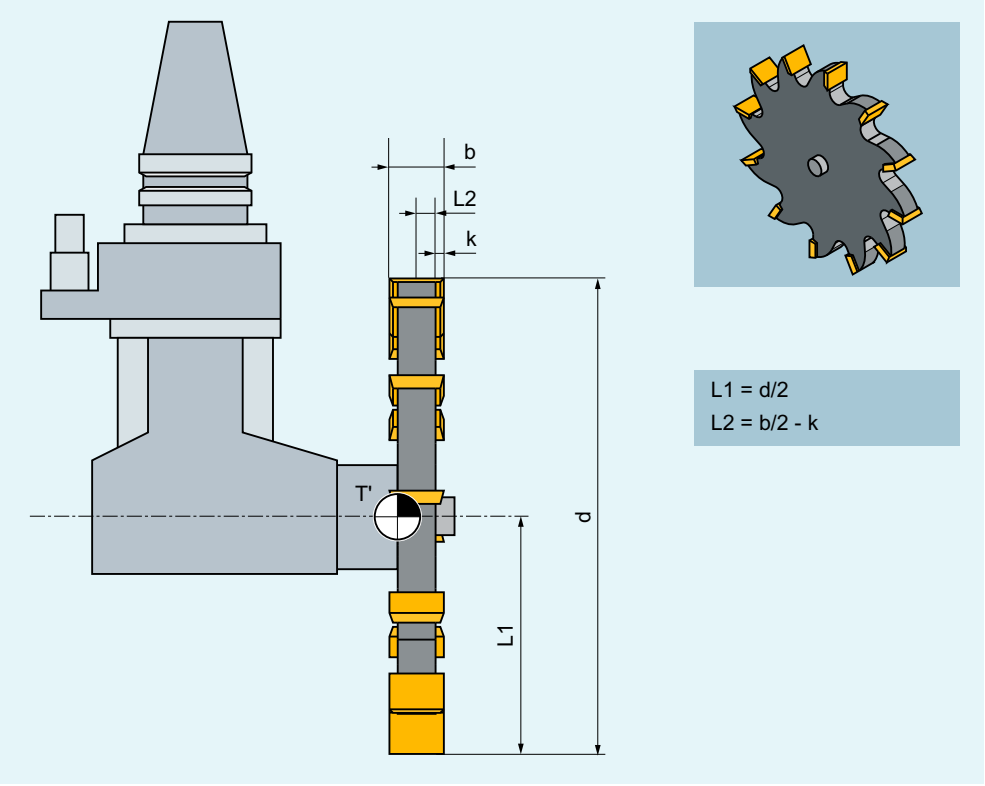
3.5.5.6 Специальный инструмент

В группе инструментов "Специальные инструменты" представлены следующие типы инструментов:

700	Наградка
710	Измерительный щуп 3D
711	Контурный щуп
712	Монощуп
713	L-образный щуп
714	Звездообразный щуп
725	Калибровочный инструмент
730	Упор
731	Пиноль
732	Люнет

Параметры инструмента

На следующем рисунке показано, какие параметры инструмента вносятся в память коррекций для инструмента "Наградка":



T	Исходная точка зажима инструмента
L1	Геометрия – длина 1
L2	Геометрия – длина 2
d	Диаметр
b	Ширина паза
k	Выступ

Параметры инструмента	Объяснение
$TC_DP1	Тип инструмента
$TC_DP3	Геометрия – длина 1
$TC_DP4	Геометрия – длина 2
$TC_DP6	Диаметр
$TC_DP7	Ширина паза
$TC_DP8	Выступ
$TC_DP21	Базовый размер – длина 1
$TC_DP22	Базовый размер – длина 2
$TC_DP23	Базовый размер – длина 3
• Значения износа согласно требованиям. • Установить прочие значения на 0.	

3.5.6 Активация/деактивация коррекции на инструмент (D, D0)

Для резцов с 1-го по 8-й (при активном управлении инструментом 12) одного инструмента могут быть согласованы различные блоки данных коррекции инструмента (например, различные значения коррекции для левой и правой кромки прорезного резца).

Активация параметров коррекции (среди них данные для коррекции длин инструмента) специального резца выполняется через вызов номера D. При программировании D0 коррекции для инструмента не действуют.

Коррекция радиуса инструмента должна быть включена дополнительно через G41/G42.

Примечание

Коррекции длин инструмента действуют, если запрограммирован номер D. Если номер D не запрограммирован, то при смене инструмента активна определенная через машинные данные стандартная установка (→ см. Указания изготовителя станка).

Синтаксис

```
D<Nr>
X... Y... Z...
G41/G42 X... Y... Z...
G40
D0
```

Объяснение

D:	Адрес активации блока данных коррекции для активного инструмента Коррекция длин инструмента выводится при первом запрограммированном перемещении соответствующей оси коррекции длин. **Внимание:** Коррекция длин инструмента действует и без программирования D, если для смены инструмента сконфигурирована автоматическая активация резца инструмента (→ См. Указания изготовителя станка).
<Nr>:	Через номер D указывается активируемый блок данных коррекции инструмента. Тип программирования D зависит от конфигурирования станка (→ См. Указания изготовителя станка). Существуют следующие возможности: • Номер D = номер резца Для каждого инструмента T<номер> (без управления инструментом) или T="имя" (с управлением инструментом) существуют номера D от 1 до макс. 12. Эти номера D напрямую присвоены резцам инструментов. Для каждого номера D (= номер резца) существует блок данных коррекции ($TC_DPx[<t>,<d>]). • Свободный выбор номеров D Номера D могут свободно согласовываться с номерами резцов инструмента. Верхняя граница используемых номеров D установлена через машинные данные.
	Диапазон значений: 0 - 32000
D0:	Деактивация блока данных коррекции для активного инструмента
G41:	Команда включения коррекции радиуса инструмента с направлением обработки **слева** от контура
G42:	Команда включения коррекции радиуса инструмента с направлением обработки **справа** от контура
G40:	Команда отключения коррекции радиуса инструмента

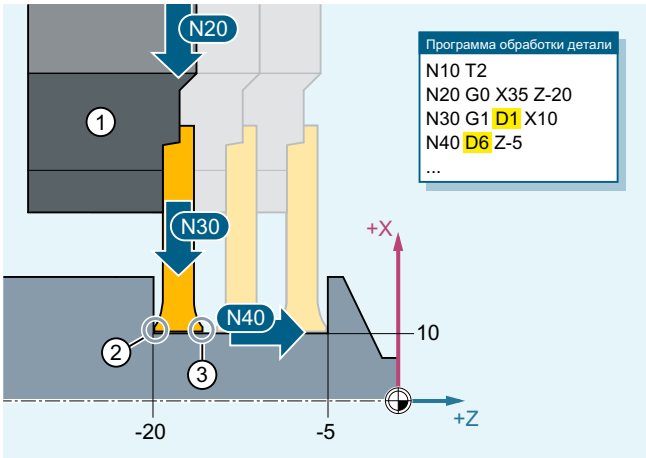
Дополнительная информация по G40/G41/G42 представлена в главе "Коррекции радиуса инструмента (Страница 260)".

Примеры

Пример 1: Смена инструмента с командой T (токарная обработка)

Программный код	Комментарий
N10 T1 D1	; Установить инструмент T1 и активировать блок данных коррекции инструмента D1 из T1.
N11 G0 X... Z...	; Выводятся коррекции длин.
N50 T4 D2	; Установить инструмент T4 и активировать блок данных коррекции инструмента D2 из T4.
...	
N70 G0 Z... D1	; Активировать другой резец D1 для инструмента T4.

Пример 2: Разные значения коррекции для левой и правой кромки прорезного резца



① Прорезной резец (T2)
② Режущая кромка D1
③ Режущая кромка D6

См. также

Коррекция радиуса инструмента (Страница 72)

Дополнительная информация

Изменение данных коррекции инструмента

Согласно стандартной установке изменения данных коррекции инструмента начинают действовать только после повторного программирования T или D.

Через следующие машинные данные можно установить, что соответствующие данные коррекции инструмента будут активированы сразу же:

MD9440 $MM_ACTIVATE_SEL_USER

⚠ ПРЕДУПРЕЖДЕНИЕ
Опасность столкновений Если MD9440 установлены, то коррекции инструмента, возникшие из-за изменений данных коррекции инструмента при останове программы обработки детали, выводятся при продолжении программы обработки детали.

3.5.7 Подавление коррекции инструмента (SUPD)

Команда SUPD позволяет покадрово подавлять коррекции инструмента.

Преимущество в сравнении с отменой коррекции инструмента посредством D0 заключается в том, что при SUPD информация о коррекции инструмента сохраняется, и

тем самым не требуется повторная активация данных коррекции инструмента посредством программирования номера D.

SUPD используется исключительно для подавления коррекции длин инструмента, например, в кадрах программы обработки детали, в которых необходимо вычислить нулевую точку, но не длину инструмента. Использование при активной G41/G42 для дополнительного подавления коррекций радиуса инструмента не рекомендуется.

Синтаксис

```
D<Nr.>
X... Y... Z...
X... Y... Z... SUPD
...
D0
```

Объяснение

SUPD:	G-команда деактивации блока данных коррекции для активного инструмента в активном кадре	
	Активность:	покадрово

Граничные условия

- SUPD следует использовать только в линейных кадрах.
- SUPD не может использоваться в синхронных действиях.
- SUPD не может использоваться вместе со следующими функциями:
  - Коррекция на радиус инструмента 3D для торцового фрезерования 3D (CUT3DFxx)
  - Таблицы кривых (CTAB)
- Использование SUPD при активной коррекции радиуса инструмента (G41/G42) возможно, но не рекомендуется.
  При необходимости использования этой функции следующие установочные данные должны быть "0":
  SD42480 $SC_STOP_CUTCOM_STOPRE = 0
  Так предотвращается прерывание программы при активной G41/G42.

Пример

В подпрограмме SUB_SUP необходимо при перемещении подавить длину инструмента.

Программа обработки детали

Программный код	Комментарий
...	
N300 $P_UIFR[1]=CTRANS(X,1000,Y,400,Z,-120)	
N310 T="BALL_D3"	

Программный код	Комментарий
N320 M6	
N330 TRAFOOF	
N340 G54 G0 Z49 D1	
N350 G0 X1100 Y500 C0 A0	
N360 **SUB_SUP**	; Вызов подпрограммы.
N370 G0 X1200	
N380 M30	

Подпрограмма с D0

Программный код	Комментарий
N10 PROC **SUB_SUP**	
N20 DEF INT NUMBER	
N30 NUMBER=$P_TOOL	
N40 G0 Z49 **D0**	; Отмена коррекций инструмента.
N50 D=NUMBER	; Повторный выбор коррекций инструмент.
N60 RET	

Подпрограмма с SUPD

Программный код	Комментарий
N10 PROC **SUB_SUP**	
N40 G0 Z49 **SUPD**	; Подавление коррекций инструмента в активном кадре.
N60 RET	

3.5.8 Программируемое смещение коррекции инструмента (TOFFL, TOFF, TOFFR, TOFFLR):

Через адреса TOFFx пользователь может в программе ЧПУ изменить эффективную длину инструмента или эффективный радиус инструмента, не изменяя находящиеся в памяти коррекций данные коррекции инструмента.

При завершении программы эти запрограммированные смещения снова удаляются.

Синтаксис

Смещение длины инструмента

```
TOFFL=<Value>
TOFFL[1]=<Value> TOFFL[2]=<Value> TOFFL[3]=<Value>
TOFF[<GeoAx>]=<Value>
```

Длина инструмента может быть изменена одновременно во всех трех компонентах. Но в одном кадре не могут одновременно использоваться команды группы TOFFL/TOFFL[1..3] с одной стороны и группы TOFF[<GeoAx>] с другой стороны. В одном кадре также не могут быть одновременно записаны TOFFL и TOFFL[1].

Если в кадре программируются не все три компонента длин инструмента, то не запрограммированные компоненты остаются без изменений. За счет этого возможно покадровое составление коррекций для нескольких компонентов. Но это действует, только пока компоненты инструмента изменяются только с TOFFL или только с TOFF. Смена варианта программирования с TOFFL на TOFF или наоборот сначала удаляет все запрограммированные ранее смещения длин инструмента (см. пример 3).

Смещение радиуса инструмента

```
TOFFR=<Value>
```

Одновременное смещение длины инструмента и радиуса инструмента

```
TOFFLR=<Value>
```

Объяснение

TOFFL:	Коррекция эффективной длины инструмента TOFFL может быть запрограммирована с индексом или без индекса:	
	TOFFL=...	Запрограммированное значение смещения действует в направлении, в котором действует и находящийся в памяти коррекций компонент длин инструмента **L1**. Операторы TOFFL и TOFFL[1] идентичны по своему действию.
	TOFFL[1]=... TOFFL[2]=... TOFFL[3]=...	Запрограммированное значение смещения действует в направлении, в котором действует и находящийся в памяти коррекций компонент длин инструмента **L1**, **L2** или **L3**.
	Примечание: Как значения коррекции длин инструмента будут пересчитаны в осях, определяет тип инструмента и текущая рабочая плоскость (G17/G18/G19).	
TOFF:	Коррекция длины инструмента в компоненте, параллельном указанной гео-оси TOFF действует в направлении компонента длин инструмента, который у не повернутого инструмента (ориентируемый инструментальный суппорт или трансформация ориентации) действует параллельно указанной в индексе гео-оси. **Примечание:** Фрейм не влияет на согласование запрограммированных значений с компонентами длин инструмента. Это означает, что для согласования компонентов длин инструмента с гео-осями за основу берется не система координат детали (WCS), а система координат инструмента в исходном положении инструмента.	
<GeoAx>:	Идентификатор гео-оси	
TOFFR:	Коррекция эффективного радиуса инструмента TOFFR изменяет эффективный радиус инструмента **при активной коррекции радиуса инструмента** на запрограммированное значение смещения.	
TOFFLR:	Коррекция эффективной длины инструмента в компоненте L1 **и** эффективного радиуса инструмента **Примечание:** В случае инструментов с закруглением углов (типов 111, 121, 131, 156) TOFFLR корректирует и угловой радиус.	
<Value>:	Значение смещения	
	Тип:	REAL

Примеры

Пример 1: Положительное смещение длины инструмента

Активным инструментом является сверло длиной L1 = 100 мм.

Активной плоскостью является G17. Это означает, что сверло указывает в направлении Z.

Необходимо увеличить эффективную длину сверла на 1 мм. Для программирования этого смещения длины инструмента существуют следующие варианты:

- `TOFFL=1`
- `TOFFL[1]=1`
- `TOFF[Z]=1`

Пример 2: Отрицательное смещение длины инструмента

Активным инструментом является сверло длиной L1 = 100 мм.

Активной плоскостью является G18. Это означает, что сверло указывает в направлении Y.

Необходимо уменьшить эффективную длину сверла на 1 мм. Для программирования этого смещения длины инструмента существуют следующие варианты:

- `TOFFL=-1`
- `TOFFL[1]=-1`
- `TOFF[Y]=1`

Пример 3: Смена варианта программирования с TOFFL на TOFF

Активным инструментом является фрезерный инструмент. Активной плоскостью является G17.

Программный код	Комментарий
`N10 TOFFL[1]=3 TOFFL[3]=5`	`; Эффективные смещения: L1=3, L2=0, L3=5`
`N20 TOFFL[2]=4`	`; Эффективные смещения: L1=3, L2=4, L3=5`
`N30 TOFF[Z]=1.3`	`; Эффективные смещения: L1=0, L2=0, L3=1.3`

Пример 4: Согласование значений смещения после смены плоскости

Программный код	Комментарий
`N10 $TC_DP1[1,1]=120`	
`N20 $TC_DP3[1,1]=100`	`; Длина инструмента L1=100 мм`
`N30 T1 D1 G17`	
`N40 TOFF[Z]=1.0`	`; Смещение в направлении Z (соответствует L1 при G17).`
`N50 G0 X0 Y0 Z0`	`; Позиция осей станка X0 Y0 Z101`
`N60 G18 G0 X0 Y0 Z0`	`; Позиция осей станка X0 Y100 Z1`
`N70 G17`	
`N80 TOFFL=1.0`	`; Смещение в направлении L1 (соответствует Z при G17).`
`N90 G0 X0 Y0 Z0`	`; Позиция осей станка X0 Y0 Z101.`
`N100 G18 G0 X0 Y0 Z0`	`; Позиция осей станка X0 Y101 Z0.`

В этом примере при смене на G18 в кадре N60 сохраняется смещение в 1 мм в оси Z, эффективная длина инструмента в оси Y — это длина инструмента в 100 мм без изменений.

В кадре N100 смещение при смене на G18 действует, напротив, в оси Y, т. к. при программировании оно было согласовано с длиной инструмента L1, и этот компонент длины при G18 действует в оси Y.

Пример 5: Одновременное смещение длины инструмента и радиуса инструмента

a, концевая фреза **без** закругления углов (тип инструмента 120):

Программный код	Комментарий
...	
`TOFFLR=0.1`	; Эффективные смещения:
	; Смещение, длина инструмента (`L1) = 5`
	; Смещение, радиус инструмента = 5
...	

b, концевая фреза **с** закруглением углов (тип инструмента 121):

Программный код	Комментарий
...	
`TOFFLR=0.1`	; Эффективные смещения:
	; Смещение, длина инструмента (`L1) = 0,1`
	; Смещение, радиус инструмента = 0,1
	; Смещение, угловой радиус = 0,1
...	

Дополнительная информация

Смещения длин инструмента

Запрограммированные смещения длин инструмента, в зависимости от типа программирования, согласуются либо с находящимися в памяти коррекций компонентами длин инструмента L1, L2 и L3 (TOFFL), либо с геометрическими осями (TOFF). В соответствии с этим запрограммированные смещения обрабатываются при смене плоскости (G17/G18/G19 ↔ G17/G18/G19):

- Если значения смещения согласованы с компонентами длин инструмента, то направления, в которых действуют запрограммированные смещения, соответственно изменяются.
- Если значения смещения согласованы с гео-осями, то смена плоскостей не влияет на согласование касательно осей координат.

При согласовании запрограммированных значений смешения с компонентами длин инструмента обрабатываются следующие установочные данные:

SD42940 $SC_TOOL_LENGTH_CONST (смена компонентов длин инструмента при смене плоскости)

SD42950 $SC_TOOL_LENGTH_TYPE (согласование компенсации длин инструмента независимо от типа инструмента)

Если у этих установочных данных есть действительные значения, отличные от 0, то они имеют приоритет перед содержанием группы G-команд 6 (выбор плоскостей G17/G18/G19) или перед указанным в данных инструмента типом инструмента ($TC_DP1[<T–Nr.>,<D–Nr.>]). Это означает, что эти установочные данные влияют на нормирование смещений так же, как и компоненты длин инструмента от L1 до L3.

Смещение радиуса инструмента

Адрес TOFFR имеет почти такое же действие, как адрес OFFN (Страница 260). Разница возникает только при активной криволинейной трансформации боковой поверхности (TRACYL) и активной коррекции стенки паза. В этом случае OFFN действует с отрицательным знаком на радиус инструмента, TOFFR, напротив, с положительным знаком.

OFFN и TOFFR могут действовать одновременно. В этом случае они, как правило, действуют аддитивно (за исключением случая коррекции стенки паза).

Смена инструмента

Все значения смещений сохраняются при смене инструмента (смене резцов). Это означает, что они продолжают действовать и при новом инструменте (новом резце).

Системные переменные для чтения текущих значений смещения

Актуальные действующие смещения могут быть считаны с помощью следующих системных переменных:

Системная переменная	Объяснение
$P_TOFFL [<n>] где 0 ≤ n ≤ 3	Считывает текущее значение смещения TOFFL (при n = 0) или TOFFL[1...3] (при n = 1, 2, 3) в контексте предварительной обработки.
$P_TOFF [<GeoAx>]	Считывает текущее значение смещения TOFF[<GeoAx>] в контексте предварительной обработки.
$P_TOFFR	Считывает текущее значение смещения TOFFR в контексте предварительной обработки.
$P_TOFFCR	Считывает текущее значение смещения углового радиуса в контексте предварительной обработки.
$AC_TOFFL[<n>] где 0 ≤ n ≤ 3	Считывает текущее значение смещения TOFFL (при n = 0) или TOFFL[1...3] (при n = 1, 2, 3) в контексте главного хода (синхронные действия).
$AC_TOFF[<GeoAx>]	Считывает текущее значение смещения TOFF[<GeoAx>] в контексте главного хода (синхронные действия).
$AC_TOFFR	Считывает текущее значение смещения TOFFR в контексте главного хода (синхронные действия).
$AC_TOFFCR	Считывает текущее значение смещения углового радиуса в контексте главного хода (синхронные действия).

Примечание

Системные переменные $AC_TOFFL, $AC_TOFF, AC_TOFFR и AC_TOFFCR при чтении из контекста предварительной обработки (программа ЧПУ) вызывают автоматическую остановку предварительной обработки.

Задачи

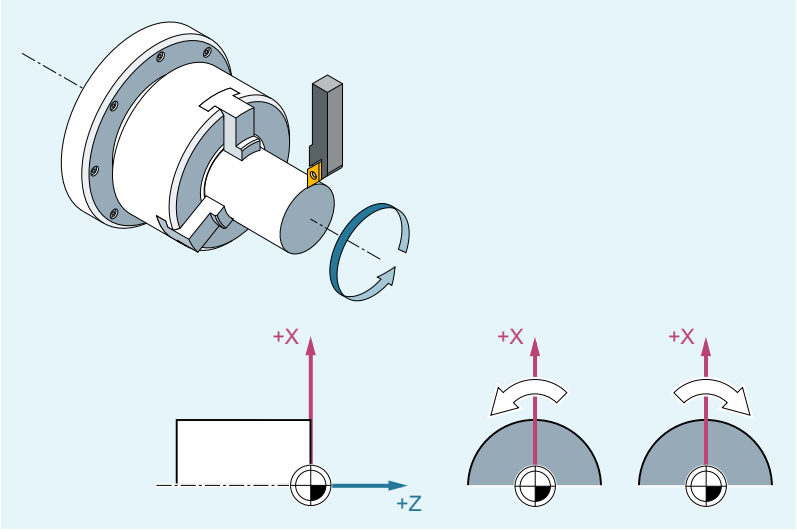
Функция "Программируемое смещение коррекции инструмента" особенно интересна для шаровой фрезы и фрезы с угловыми радиусами, т. к. они в системе CAM часто вычисляются на центре сферы, а не на вершине сферы. Но при измерении инструмента, как правило, измеряется острие инструмента и помещается в качестве длины инструмента в память коррекций.

Для 3D коррекции на радиус инструмента при использовании шаровой фрезы рекомендуется одновременно корректировать на одинаковое значение длину и радиус инструмента. Для этого в распоряжении пользователя есть адрес TOFFLR.

3.6 Движение шпинделя

3.6.1 Число оборотов шпинделя (S), направление вращения шпинделя (M3, M4, M5)

Указание скорости и направления вращения запускают вращательное движение шпинделя и создают условие для обработки резаньем.



Наряду с главным шпинделем возможно наличие и других шпинделей (например, у токарных станков встречный шпиндель или вращающийся инструмент). Как правило, главный шпиндель назначается через машинные данные мастер-шпинделем. Назначение может быть изменено через команду ЧПУ.

Синтаксис

`S... / S<n>=...`

`M3 / M<n>=3`

`M4 / M<n>=4`

M5 / M<n>=5

SETMS(<n>)	
...	
SETMS	

Объяснение

S...:	Скорость шпинделя в оборотов/мин для мастер-шпинделя
S<n>=...:	Скорость шпинделя в оборотах/мин для шпинделя <n>
	Примечание: Указанная с S0... скорость относится к мастер-шпинделю.
M3:	Направление вращения шпинделя вправо для мастер-шпинделя
M<n>=3:	Направление вращения шпинделя вправо для шпинделя <n>
M4:	Направление вращения шпинделя влево для мастер-шпинделя
M<n>=4:	Направление вращения шпинделя влево для шпинделя <n>
M5:	Остановка шпинделя для мастер-шпинделя
M<n>=5:	Остановка шпинделя для шпинделя <n>
SETMS(<n>):	Шпиндель <n> должен считаться мастер-шпинделем
SETMS:	SETMS без указания шпинделя выполняет возврат на сконфигурированный мастер-шпиндель

Примечание

На кадр ЧПУ разрешено программировать макс. 3 значения S, например:

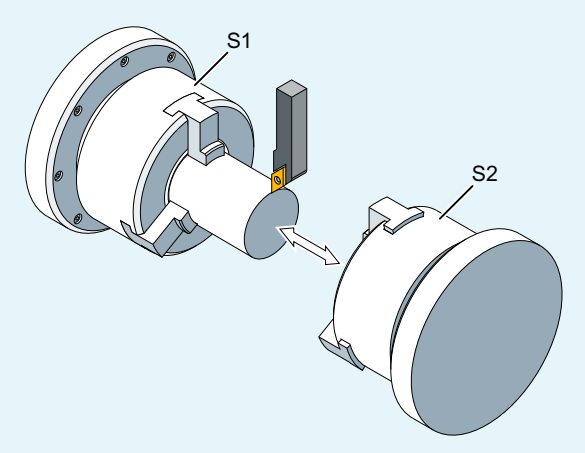
```
S... S2=... S3=...
```

Примечание

SETMS должна стоять в собственном кадре.

Пример

S1 — мастер-шпиндель, S2 — второй шпиндель изделия. Токарная деталь должна быть обработана с 2-х сторон. Для этого необходимо разделение рабочих операций. После отреза синхронное устройство (S2) принимает деталь для обработки со стороны отреза. Для этого этот шпиндель S2 определяется как мастер-шпиндель, теперь для него действует G95.



Программный код	Комментарий
`N10 S300 M3`	; Скорость и направление вращения для приводного шпинделя = предустановленного мастер-шпинделя.
...	; Обработка правой стороны детали.
`N100 SETMS(2)`	`; S2` теперь мастер-шпиндель.
`N110 S400 G95 F…`	; Скорость для нового мастер-шпинделя.
...	; Обработка левой стороны детали.
`N160 SETMS`	; Возврат к мастер-шпинделю `S1`.

Дополнительная информация

Интерпретация значения S для мастер-шпинделя

Если в группе G 1 (действующие модально команды движения) активна функция G331 или G332, то запрограммированное значение S всегда интерпретируется как скорость в оборотах/мин. В иных случаях интерпретация значения S зависит от группы G 15 (тип подачи): При активной G96, G961 или G962 значение S интерпретируется как постоянная скорость резания в м/мин, во всех других случаях как скорость в оборотах/мин.

При переключении с G96/G961/G962 на G331/G332 значение постоянной скорости резания устанавливается на ноль, при переключении с G331/G332 на функцию из группы G 1, отличную от G331/G332, значение скорости устанавливается на ноль. Соответствующие значения S при необходимости должны быть запрограммированы заново.

Предустановленные команды M3, M4, M5

В кадре с осевыми командами функции M3, M4, M5 включаются **до** запуска движений осей (первичная настройка СЧПУ).

Пример:

Программный код	Комментарий
`N10 G1 F500 X70 Y20 S270 M3`	; Шпиндель разгоняется до 270 об/мин, после осуществляются движения в `X` и `Y`.
`N100 G0 Z150 M5`	; Остановка шпинделя перед движением отвода в `Z`.

Примечание

Через машинные данные может быть установлено, будут ли осевые движения выполнены лишь после разгона шпинделя до заданной скорости или остановки шпинделя, или перемещение будет осуществляться сразу же после запрограммированных процессов переключения.

Работа с несколькими шпинделями

В одном канале одновременно может быть 5 шпинделей (мастер-шпиндель плюс 4 дополнительных шпинделя).

Один шпиндель определяется через машинные данные как **мастер-шпиндель**. Для этого шпинделя действуют специальные функции, как то, например, резьбонарезание, нарезание внутренней резьбы, окружная подача, время ожидания. Для прочих шпинделей (например, второго шпинделя изделия и вращающегося инструмента) для скорости и направления вращения/остановки шпинделя должны быть указаны соответствующие номера.

Пример:

Программный код	Комментарий
`N10 S300 M3 S2=780 M2=4`	; Мастер-шпиндель: 300 об/мин, правое вращение 2-й шпиндель: 780 об/мин, левое вращение

Программируемое переключение мастер-шпинделя

Через команду `SETMS(<n>)` в программе ЧПУ можно определить любой шпиндель мастер-шпинделем. `SETMS` должна стоять в собственном кадре.

Пример:

Программный код	Комментарий
`N10 SETMS(2)`	; Шпиндель 2 теперь мастер-шпиндель.

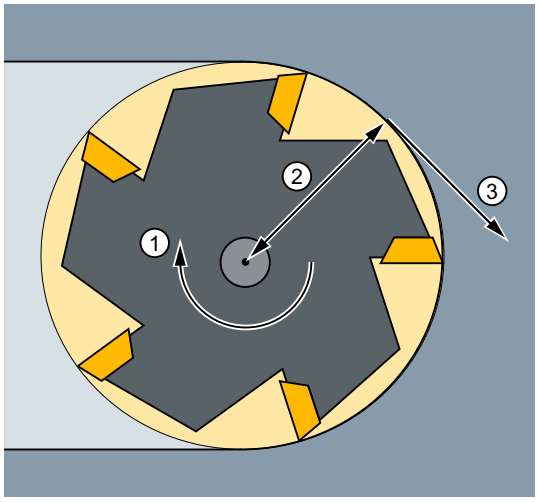
Примечание

Теперь для заново назначенного мастер-шпинделя действует указанная с `S...` скорость и запрограммированные с `M3`, `M4`, `M5` функции.

С помощью `SETMS` без указания шпинделя осуществляется возврат к установленному в машинных данных мастер-шпинделю.

3.6.2 Скорость резания инструмента (SVC)

В качестве альтернативы к скорости шпинделя для фрезерных обработок может быть запрограммирована и более часто используемая на практике скорость резания инструмента:



① Скорость шпинделя
② Радиус инструмента
③ Скорость резания инструмента

Через радиус активного инструмента СЧПУ по запрограммированной скорости резания инструмента вычисляет действующую скорость шпинделя:

$S = (SVC * 1000) / (R_{WKZ} * 2\pi)$

где:
- S: скорость шпинделя в об/мин
- SVC: скорость резания инструмента в м/мин или футах/мин
- R_{WKZ}: радиус активного инструмента в мм

Тип ($TC_DP1) активного инструмента не учитывается.

Запрограммированная скорость резания инструмента не зависит от подачи по траектории F, а также от группы G-функций 15 (тип подачи). Направление вращения и старт шпинделя осуществляются через M3 или M4, стоп шпинделя – через M5.

Изменение данных радиуса инструмента в памяти коррекций вступает в силу при следующем выборе коррекции инструмента или следующей актуализации активных данных коррекции.

Смена инструмента и включение/выключение блока данных коррекции инструмента приводят к новому вычислению действующей скорости шпинделя.

Условия

Для программирования скорости резания инструмента необходимы:

- геометрические отношения вращающегося инструмента (фрезерный или сверлильный инструмент)
- активный блок данных коррекции инструмента

Синтаксис

```
T... D... SVC[<n>]=<значение>
...
S... M3/M4
```

Объяснение

SVC:	Кодовое слово для программирования скорости резания инструмента	
[<n>]:	Номер шпинделя С помощью этого расширения адреса указывается, для какого шпинделя должна действовать запрограммированная скорость резания. Без расширения адреса данные всегда относятся к актуальному мастер-шпинделю. **Примечание:** Для каждого шпинделя может быть задана собственная скорость резания. **Примечание:** Программирование SVC без расширения адреса предполагает, что мастер-шпиндель имеет активный инструмент. При смене мастер-шпинделя пользователь должен выбрать соответствующий инструмент.	
<Value>:	Значение скорости резания инструмента	
	Единица:	м/мин (для G71/G710) или футов/мин (для G70/G700)
T... D...:	В кадре с SVC должен быть известен радиус инструмента. Это означает, что соответствующий инструмент и, в том числе, блок данных коррекции инструмента, должен быть активен или выбран в кадре. Последовательность SVC и выбора T/D при программировании в одном и том же кадре является произвольной.	
S... M3/M4:	Программирование скорости шпинделя ведет к выключению скорости резания инструмента. **Указание:** Возможно произвольное переключение между программированием SVC и S, в том числе при вращающемся шпинделе. Соответствующее не активное значение удаляется.	

Примечание

Программирование SVC невозможно при следующих активных движениях подачи шпинделя:

- Постоянная скорость резания: G96/G961/G962 S... (Страница 103)
- Постоянная окружная скорость круга: SUG (Страница 108)
- Позиционирование шпинделя: SPOS/SPOSA/M19 (Страница 123)
- Переключение мастер-шпинделя в осевой режим: M70 (Страница 123)

И наоборот, программирование одной из этих функций приводит к отмене SVC (скорость резания инструмента).

Примечание

Максимальная скорость инструмента

Через системную переменную $TC_TP_MAX_VELO[<номер T>] можно задать макс. скорость инструмента (скорость шпинделя).
Если граница скорости не определена, контроль не выполняется.

Примечание

Например, созданные через системы CAD траектории "стандартных инструментов", которые уже учитывают радиус инструмента и содержат только разницу со стандартным инструментом в радиусе резцов, не поддерживаются в комбинации с программированием SVC.

Примеры

Для всех примеров должно действовать: зажим инструмента = шпиндель (для стандартного фрезерования)

Пример 1: фреза с радиусом 6 мм

Программный код	Комментарий
`N10 G0 X10 T1 D1`	; Выбор фрезерного инструмента с, например, `$TC_DP6[1,1] = 6` (радиус инструмента = 6 мм)
`N20 SVC=100 M3`	; Скорость резания = 100 м/мин
	⇒ полученная скорость шпинделя:
	`S = (100` м/мин * 1000) / (6,0 мм * 2 * 3,14) = 2653,93 об/мин
`N30 G1 X50 G95 FZ=0,03`	`; SVC` и подача на зуб
...	

Пример 2: выбор инструмента и SVC в одном и том же кадре

Программный код	Комментарий
`N10 G0 X20`	
`N20 T1 D1 SVC=100`	; Выбор блока данных инструмента и коррекции вместе с `SVC` в кадре (произвольная последовательность).

Программный код	Комментарий
`N30 X30 M3`	; Старт шпинделя с правым направлением вращения, скорость резания 100 м/мин
`N40 G1 X20 F0.3 G95`	`;` `SVC` и окружная подача

Пример 3: задание скоростей резания для двух шпинделей

Программный код	Комментарий
`N10 SVC[3]=100 M6 T1 D1`	
`N20 SVC[5]=200`	; Радиус инструмента активной коррекции инструмента идентичен для обеих шпинделей, различная действующая скорость для шпинделя 3 и шпинделя 5.

Пример 4:

Допущения:

Мастер касательно смены инструмента определяется через зажим инструмента:

MD20124 $MC_TOOL_MANAGEMENT_TOOLHOLDER > 1

При смене инструмента старая коррекция инструмента сохраняется, и только при программировании `D` активируется коррекция нового инструмента:

MD20270 $MC_CUTTING_EDGE_DEFAULT = - 2

Программный код	Комментарий
`N10 $TC_MPP1[9998,1]=2`	; Место в магазине это зажим инструмента
`N11 $TC_MPP5[9998,1]=1`	; Место в магазине это зажим инструмента 1
`N12 $TC_MPP_SP[9998,1]=3`	; Зажим инструмента 1 согласован со шпинделем 3
`N20 $TC_MPP1[9998,2]=2`	; Место в магазине это зажим инструмента
`N21 $TC_MPP5[9998,2]=4`	; Место в магазине это зажим инструмента 4
`N22 $TC_MPP_SP[9998,2]=6`	; Зажим инструмента 4 согласован со шпинделем 6
`N30 $TC_TP2[2]="WZ2"`	
`N31 $TC_DP6[2,1]=5.0`	; Радиус = 5,0 мм из `T2,` коррекция `D1`
`N40 $TC_TP2[8]="WZ8"`	
`N41 $TC_DP6[8,1]=9.0`	; Радиус = 9,0 мм из `T8,` коррекция `D1`
`N42 $TC_DP6[8,4]=7.0`	; Радиус = 7,0 мм из `T8,` коррекция `D4`
...	
`N100 SETMTH(1)`	; Установить номер мастер-зажима инструмента
`N110 T="WZ2" M6 D1`	; Устанавливается инструмент `T2,` и активируется коррекция `D1.`
`N120 G1 G94 F1000 M3=3 SVC=100`	`;` `S3 = (100` м/мин * 1000) / (5,0 мм * 2 * 3,14) = 3184,71 об/мин
`N130 SETMTH(4)`	; Установить номер мастер-зажима инструмента
`N140 T="WZ8"`	; Соответствует `T8="WZ8"`
`N150 M6`	; Соответствует `M4=6`
	Инструмент `"WZ8"` поступает в мастер-зажим инструмента, но из-за `MD20270 = -2` остается активной старая коррекция инструмента.
`N160 SVC=50`	`;` `S3 = (50` м/мин * 1000) / (5,0 мм * 2 * 3,14) = 1592,36 об/мин
	Коррекция зажима инструмента 1 еще активна, и он согласован со шпинделем 3.

Программный код	Комментарий
N170 D4	; Активируется коррекция D4 нового инструмента "WZ8" (на зажиме инструмента 4).
N180 SVC=300	; S6 = (300 м/мин * 1000) / (7,0 мм * 2 * 3,14) = 6824,39 об/мин
	Шпиндель 6 согласован с зажимом инструмента 4.

Пример 5:

Допущения:

Шпиндели являются одновременно зажимами инструмента:

MD20124 $MC_TOOL_MANAGEMENT_TOOLHOLDER = 0

При смене инструмента автоматически выбирается блок данных коррекции инструмента D4:

MD20270 $MC_CUTTING_EDGE_DEFAULT = 4

Программный код	Комментарий
N10 $TC_MPP1[9998,1]=2	; Место в магазине это зажим инструмента
N11 $TC_MPP5[9998,1]=1	; Место в магазине это зажим инструмента 1 = шпиндель 1
N20 $TC_MPP1[9998,2]=2	; Место в магазине это зажим инструмента
N21 $TC_MPP5[9998,2]=3	; Место в магазине это зажим инструмента 3 = шпиндель 3
N30 $TC_TP2[2]="WZ2"	
N31 $TC_DP6[2,1]=5.0	; Радиус = 5,0 мм из T2, коррекция D1
N40 $TC_TP2[8]="WZ8"	
N41 $TC_DP6[8,1]=9.0	; Радиус = 9,0 мм из T8, коррекция D1
N42 $TC_DP6[8,4]=7.0	; Радиус = 7,0 мм из T8, коррекция D4
...	
N100 SETMS(1)	; Шпиндель 1 = мастер-шпиндель
N110 T="WZ2" M6 D1	; Устанавливается инструмент T2, и активируется коррекция D1.
N120 G1 G94 F1000 M3 SVC=100	; S1 = (100 м/мин * 1000) / (5,0 мм * 2 * 3,14) = 3184,71 об/мин
N200 SETMS(3)	; Шпиндель 3 = мастер-шпиндель
N210 M4 SVC=150	; S3 = (150 м/мин * 1000) / (5,0 мм * 2 * 3,14) = 4777,07 об/мин
	Относится к коррекции инструмента D1 из T="WZ2", S1 продолжает вращаться со старой скоростью.
N220 T="WZ8"	; Соответствует T8="WZ8"
N230 M4 SVC=200	; S3 = (200 м/мин * 1000) / (5,0 мм * 2 * 3,14) = 6369,43 об/мин
	Относится к коррекции инструмента D1 из T="WZ2".
N240 M6	; Соответствует M3=6
	Инструмент "WZ8" поступает в мастер-шпиндель, коррекция инструмента D4 нового инструмента активируется.
N250 SVC=50	; S3 = (50 м/мин * 1000) / (7,0 мм * 2 * 3,14) = 1137,40 об/мин
	Коррекция D4 на мастер-шпинделе активна.
N260 D1	; Коррекция D1 нового инструмента "WZ8" активна.
N270 SVC[1]=300	; S1 = (300 м/мин * 1000) / (9,0 мм * 2 * 3,14) = 5307,86 об/мин
	S3 = (50 м/мин * 1000) / (9,0 мм * 2 * 3,14) = 884,64 об/мин
...	

Дополнительная информация

Радиус инструмента

Следующие данные коррекции инструмента (активного инструмента) относятся к радиусу инструмента:

- $TC_DP6 (радиус — геометрия)
- $TC_DP15 (радиус — износ)
- $TC_SCPx6 (коррекция для $TC_DP6)
- $TC_ECPx6 (коррекция для $TC_DP6)

Не учитываются:

- Коррекции радиуса Online
- Припуск к запрограммированному контуру (OFFN)

Коррекция радиуса инструмента (G41/G42)

Коррекция радиуса инструмента (G41/G42) и SVC относятся к радиусу инструмента, но функционально не связаны и не зависят друг от друга.

Нарезание внутренней резьбы без компенсирующего патрона (G331, G332)

Программирование SVC возможно и в комбинации с G331 или G332.

Синхронные действия

Ввод SVC из синхронных действий невозможен.

Чтение скорости резания и вариантов программирования скорости шпинделя

Скорость резания шпинделя и варианты программирования скорости (скорость шпинделя S или скорость резания инструмента SVC) могут быть считаны через системные переменные:

- С остановкой предварительной обработки в программе обработки детали через системные переменные:

$AC_SVC[<n>]	Скорость резания, которая действовала при подготовке актуального кадра главного хода для шпинделя с номером <n>.
$AC_S_TYPE[<n>]	Вариант программирования скорости шпинделя, который действовал при подготовке актуального кадра главного хода для шпинделя с номером <n>.

Величина:	**Значение:**
1	Скорость шпинделя в об/мин
2	Скорость резания инструмента SVC в м/мин или футах/мин

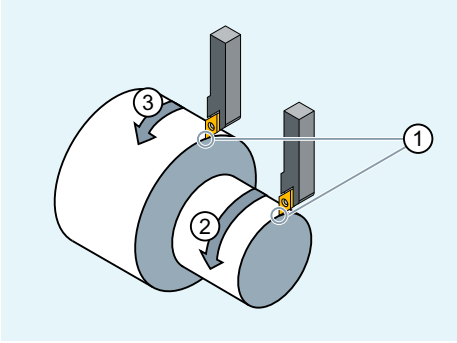
- Без остановки предварительной обработки в программе обработки детали через системные переменные:

$P_SVC[<n>]	Запрограммированная скорость резания для шпинделя <n>
$P_S_TYPE[<n>]	Запрограммированный вариант программирования скорости шпинделя для шпинделя <n>

Величина:	**Значение:**
1	Скорость шпинделя в об/мин
2	Скорость резания инструмента SVC в м/мин или футах/мин

3.6.3 Постоянная скорость резания (G96/G961/G962, G97/G971/G972, G973, LIMS, SCC)

При активной функции "Постоянная скорость резания" в зависимости от соответствующего диаметра детали скорость шпинделя изменяется так, что скорость резания S в м/мин или футов/мин остается на резце инструмента постоянной.



① Скорость резания постоянная
② Число оборотов шпинделя повышенное
③ Число оборотов шпинделя уменьшенное

Это дает следующие преимущества:

- Равномерные поверхности после обточки и тем самым высокое качество поверхностей
- Щадящая для инструмента обработка

Синтаксис

Включение/выключение постоянной скорости резания для мастер-шпинделя:

```
G96/G961/G962 S...
...
G97/G971/G972/G973
```

Ограничение скорости для мастер-шпинделя:
LIMS=<значение>
LIMS[<шпиндель>]=<значение>

Другая ось отсчета для G96/G961/G962:
SCC[<ось>]

Примечание

SCC[<ось>] может быть запрограммирована отдельно или вместе с G96/G961/G962.

Объяснение

G96:	Окружная подача (как у G95 (Страница 110)) и постоянная скорость резания С G96 автоматически включается G95. Если до этого G95 не была включена, при вызове G96 необходимо указать новое значение подачи F....	
G961:	Линейная подача (как у G94 (Страница 110)) и постоянная скорость резания	
G962:	Линейная подача или окружная подача и постоянная скорость резания	
S...:	В комбинации с G96, G961 или G962 S... интерпретируется не как скорость шпинделя, а как скорость резания. Скорость резания всегда действует на мастер-шпиндель.	
	Единица:	м/мин (для G71/G710) или футов/мин (для G70/G700)
	Диапазон значений:	0,1 м/мин ... 9999 9999,9 м/мин
G97:	Окружная подача и постоянная скорость шпинделя (постоянная скорость резания ВЫКЛ.)	
G971:	Линейная подача и постоянная скорость шпинделя (постоянная скорость резания ВЫКЛ)	
G972:	Линейная подача или окружная подача и постоянная скорость шпинделя (постоянная скорость резания ВЫКЛ)	
G973:	Окружная подача без ограничения скорости шпинделя и постоянная скорость шпинделя (G97 без LIMS для режима ISO)	
	Указание: После G97 (или G971 ... G973) S... снова интерпретируется как скорость шпинделя в оборотов/мин. Если новая скорость шпинделя не указана, сохраняется последняя установленная через G96 (или G961 или G962) скорость.	
LIMS:	Ограничение скорости для мастер-шпинделя (действует только при активной G96/G961/G97) Для станков с переключаемыми мастер-шпинделями в одном кадре макс. для 4 шпинделей могут быть запрограммированы ограничения с различными значениями.	
	<шпиндель>:	Номер шпинделя
	<значение>:	Верхняя граница скорости шпинделя в оборотов/мин
SCC:	При активной функции G96/G961/G962 с помощью SCC[<ось>] любая гео-ось может быть назначена осью отсчета.	

Примечание

При первом выборе G96/G961/G962 необходимо ввести постоянную скорость резания S..., при повторном выборе G96/G961/G962 указание является опциональным.

Примечание

Запрограммированное посредством LIMS ограничение скорости вращения не должно превышать предельной скорости, запрограммированной посредством G26 или определенной в установочных данных.

Примечание

Осью отсчета для G96/G961/G962 на момент программирования SCC[<ось>] должна быть известная в канале гео-ось. Программирование SCC[<ось>] возможно и при активной G96/G961/G962.

Примеры

Пример 1: Включение постоянной скорости резания с ограничением скорости

Программный код	Комментарий
`N10 SETMS(3)`	
`N20 G96 S100 LIMS=2500`	; Постоянная скорость резания = 100 м/мин, макс. скорость = 2500 об/мин
...	
`N60 G96 G90 X0 Z10 F8 S100 LIMS=444`	; Макс. скорость = 444 об/мин

Пример 2: Задача ограничения скорости для 4 шпинделей

Устанавливаются ограничения скорости для шпинделя 1 (мастер-шпиндель) и шпинделей 2, 3 и 4:

Программный код
`N10 LIMS=300 LIMS[2]=450 LIMS[3]=800 LIMS[4]=1500`
...

Пример 3: Согласование оси Y при поперечной обработке с осью X

Программный код	Комментарий
`N10 G18 LIMS=3000 T1 D1`	; Ограничение скорости до 3000 об/мин
`N20 G0 X100 Z200`	
`N30 Z100`	
`N40 G96 S20 M3`	; Постоянная скорость резания 20 м/мин, зависит от оси `X`.
`N50 G0 X80`	
`N60 G1 F1.2 X34`	; Поперечная обработка в `X` с 1,2 мм/об.
`N70 G0 G94 X100`	
`N80 Z80`	

Программный код	Комментарий
`N100 T2 D1`	
`N110 G96 S40 SCC[Y]`	; Ось `Y` согласуется с `G96`, и `G96` активируется (возможно в одном кадре). Постоянная скорость резания составляет 40 м/мин и зависит от оси `Y`.
...	
`N140 Y30`	
`N150 G01 F1.2 Y=27`	; Прорезка в `Y`, подача `F = 1,2` мм/оборот.
`N160 G97`	; Постоянная скорость резания выкл.
`N170 G0 Y100`	

Дополнительная информация

Вычисление скорости шпинделя

Основой для вычисления скорости шпинделя из запрограммированной скорости резания является позиция ENS поперечной оси (радиус).

Примечание

Фреймы между WCS и ENS (например, программируемые фреймы как SCALE, TRANS или ROT) учитываются при вычислении скорости шпинделя и могут вызвать изменение скорости (например, если при SCALE изменяется эффективный диаметр).

Ограничение скорости LIMS

Если необходимо обработать деталь с большими разностями в диаметре, рекомендуется указать ограничение скорости шпинделя с помощью LIMS (макс. скорость шпинделя). Таким образом, можно исключить недопустимо высокую скорость при маленьких диаметрах. LIMS действует только при активной G96, G961 и G97. При G971 LIMS не действует. При установке кадра в главный ход все запрограммированные значения передаются в установочные данные.

Примечание

Измененные в программе обработки детали с помощью LIMS границы скорости передаются в установочные данные, сохраняясь тем самым и после завершения программы.

И напротив, если измененные с помощью LIMS границы скорости после завершения программы, действовать не должны, необходимо вставить следующее определение в блок GUD изготовителя станка:

REDEF $SA_SPIND_MAX_VELO_LIMS PRLOC

Выключение постоянной скорости резания (G97/G971/G972/G973)

После G97 (или G971 ... G973) S... снова интерпретируется как скорость шпинделя в оборотах/мин. Если новая скорость шпинделя не указана, то сохраняется последняя установленная через G96 (или G961 или G962) скорость.

Функцию G96/G961 также можно выключать с помощью G94 или G95. В этом случае действует последняя запрограммированная скорость S... для дальнейшего процесса обработки.

G97 можно программировать без предшествующей G96. В этом случае функция действует как G95, дополнительно может быть запрограммирована LIMS.

С помощью G973 постоянную скорость резания можно отключить без активации ограничения скорости шпинделя.

Примечание

Поперечная ось должна быть определена через машинные данные.

Движение ускоренным ходом G0

При движении ускоренным ходом G0 изменения числа оборотов не предпринимаются.

Исключение:

Если подвод к контуру осуществляется ускоренным ходом, и следующий кадр ЧПУ содержит траекторную команду G1/G2/G3/..., то скорость для следующей траекторной команды устанавливается уже в кадре подвода G0.

Другая ось отсчета для G96/G961/G962

При активной функции G96/G961/G962 с помощью SCC[<ось>] любая гео-ось может быть назначена осью отсчета. При изменении оси отсчета и тем самым исходной позиции острия инструмента (TCP-Tool Center Point) для постоянной скорости резания, результирующая скорость шпинделя достигается по установленной рампе торможения или разгона.

Изменение согласованной оси канала

Свойство оси отсчета для G96/G961/G962 всегда присвоено гео-оси. При изменении согласованной оси канала свойство оси отсчета для G96/G961/G962 остается в старом канале.

Изменение гео-оси не влияет на присваивание гео-оси постоянной скорости резания. Если в результате изменения гео-оси изменяется исходная позиция TCP для G96/G961/G962, то шпиндель разгоняется в соответствии с рампой до новой скорости.

Если в результате изменения гео-оси не происходит согласования новой оси канала (например, GEOAX(0,X)), то скорость шпинделя замораживается согласно G97.

Примеры изменения гео-оси с согласованиями оси отсчета:

Программный код	Комментарий
`N05 G95 F0.1`	
`N10 GEOAX(1,X1)`	;Ось канала `X1` становится первой гео-осью.
`N20 SCC[X]`	;Первая гео-ось (`X`) становится осью отсчета
	; для `G96/G961/G962`.
`N30 GEOAX(1,X2)`	;Ось канала `X2` становится первой гео-осью.
`N40 G96 M3 S20`	;Ось отсчета для это ось канала X2.

Программный код	Комментарий
N05 G95 F0.1	
N10 GEOAX(1,X1)	;Ось канала X1 становится первой гео-осью.
N20 SCC[X1]	;X1 и не явно первая гео-ось (X) становится
	;осью отсчета для G96/G961/G962.
N30 GEOAX(1,X2)	;Ось канала X2 становится первой гео-осью.
N40 G96 M3 S20	;Ось отсчета для G96 это X2 или X, без аварийного сообщения.

Программный код	Комментарий
N05 G95 F0.1	
N10 GEOAX(1, X2)	;Ось канала X2 становится первой гео-осью.
N20 SCC[X1]	;X1 это не гео-ось, аварийное сообщение.

Программный код	Комментарий
N05 G0 Z50	
N10 X35 Y30	
N15 SCC[X]	:Ось отсчета для G96/G961/G962 это X.
N20 G96 M3 S20	;Постоянная скорость резания с 10 мм/мин вкл.
N25 G1 F1.5 X20	;Поперечная обработка в X с 1,5 мм/об.
N30 G0 Z51	
N35 SCC[Y]	;Ось отсчета для G96 это Y,
	;уменьшение скорости шпинделя (Y30).
N40 G1 F1.2 Y25	;Поперечная обработка в Y с 1,2 мм/об.

3.6.4 Включение / выключение постоянной окружной скорости круга (GWPSON, GWPSOF)

Предопределенные процедуры GWPSON(...) и GWPSOF(...) позволяют включать / выключать постоянную окружную скорость круга (SUG) для шлифовальных инструментов (тип инструмента: 400 - 499).

Синтаксис

```
GWPSON(<TNr>)
S<n>=...
...
GWPSOF(<TNr>)
```

Значение

GWPSON(...):	Включение постоянной окружной скорости круга
GWPSOF(...):	Выключение постоянной окружной скорости круга
<TNr>:	номер T **Примечание:** требуется только в том случае, если постоянная окружная скорость круга вместо активного, находящегося в эксплуатации инструмента должна быть включена / выключена для неактивного шлифовального круга.
S<n>=...:	Окружная скорость круга в м/с или фут/с для шпинделя <n>
S0=... или S... :	Окружная скорость круга для ведущего шпинделя

Запрос состояния

С помощью следующей системной переменной из программы обработки деталей можно запросить, активна ли постоянная окружная скорость круга для определенного шпинделя.

$P_GWPS[<n>] ; где <n> = номер шпинделя

Величина	Значение
0 (= FALSE)	SUG **выкл**ючена.
1 (= TRUE)	SUG **вкл**ючена.

3.6.5 Программируемое ограничение числа оборотов шпинделя (G25, G26)

Определенные в машинных данных мин. и макс. скорости шпинделя могут быть изменены через команду программы обработки детали.

Запрограммированные ограничения скорости шпинделя возможны для всех шпинделей канала.

Синтаксис

```
G25 S... S1=... S2=...
G26 S... S1=... S2=...
```

Объяснение

G25:	**Нижнее** ограничение скорости шпинделя
G26:	**Верхнее** ограничение скорости шпинделя
S... S1=... S2=... :	Минимальная или максимальная скорости(и) шпинделя **Примечание:** В кадре может быть запрограммировано макс. три ограничения скорости шпинделя.
Диапазон значений:	0.1 ... 9999 9999.9 об/мин

Примечание

Запрограммированное с G25 или G26 ограничение скорости шпинделя заменяет предельные скорости и установочных данных и тем самым сохраняется и после завершения программы.

Если же измененные с G25/G26 границы скорости после завершения программы, напротив, более не должны действовать, то вставить следующие определения в блок GUD изготовителя станка:

REDEF $SA_SPIND_MIN_VELO_G25 PRLOC

REDEF $SA_SPIND_MAX_VELO_G26 PRLOC

Пример

Программный код	Комментарий
N10 G26 S1400 S2=350 S3=600	; Верхнее предельное число оборотов для мастер-шпинделя, шпинделя 2 и шпинделя 3

3.7 Регулирование подачи

3.7.1 Подача (G93, G94, G95, F, FGROUP, FL, FGREF)

С помощью этих команд в программе ЧПУ устанавливаются скорости подачи для всех участвующих в последовательности обработки осей.

Синтаксис

```
G93
G94
G95
F<значение>
FGROUP(<ось_1>,<ось_2>,...)
FGREF[<круговая ось>]=<исходный радиус>
FL[<ось>]=<значение>
```

Значение

G93:	Тип подачи по траектории: Обратная по времени подача [1/мин]
G94:	Тип подачи по траектории: Линейная подача [мм/мин], [дюймов/мин] или [градусов/мин]
G95:	Тип подачи по траектории: Окружная подача [мм/оборот] или [дюймов/оборот] Окружная подача может на выбор отводиться от мастер-шпинделя, любого другого шпинделя или круговой оси.

F< значение>	Подача по траектории для всех или выбранных через FGROUP траекторных осей.
FGROUP:	определение траекторных осей, к которым относится запрограммированная через F подача по траектории.
FGREF:	С FGREF для каждой из указанных в FGROUP круговых осей программируется эффективный радиус (<исходный радиус>)
FL:	Предельная скорость для синхронных/траекторных осей Действует установленная с G94 единица. На ось (ось канала, гео-ось или ось ориентации) может быть запрограммировано одно значение FL.
<ось>:	Имя оси канала, тип: AXIS

Примеры

Пример 1: Принцип действия FGROUP

Следующий пример должен пояснить воздействие FGROUP на ход траектории и подачу по траектории. Переменная $AC_TIME содержит время с начала кадра в секундах. Она может использоваться только в синхронных действиях.

Программный код	Комментарий
N100 G0 X0 A0	
N110 FGROUP(X,A)	
N120 G91 G1 G710 F100	; Подача= 100мм/мин или 100градусов/мин
N130 DO $R1=$AC_TIME	
N140 X10	; Подача= 100 мм/мин, ход траектории= 10 мм, R1= около 6 с
N150 DO $R2=$AC_TIME	
N160 X10 A10	; Подача= 100 мм/мин, ход траектории= 14.14 мм, R2= около 8 с
N170 DO $R3=$AC_TIME	
N180 A10	; Подача= 100 градусов/мин, ход траектории= 10 градусов, R3= около 6 с
N190 DO $R4=$AC_TIME	
N200 X0.001 A10	; Подача= 100 мм/мин, ход траектории= 10 мм, R4= около 6 с
N210 G700 F100	; Подача= 2540 мм/мин или 100 градусов/мин
N220 DO $R5=$AC_TIME	
N230 X10	; Подача= 2540 мм/мин, ход траектории= 254 мм, R5= около 6 с
N240 DO $R6=$AC_TIME	
N250 X10 A10	; Подача= 2540 мм/мин, ход траектории= 254,2 мм, R6= около 6 с
N260 DO $R7=$AC_TIME	
N270 A10	; Подача= 100 градусов/мин, ход траектории= 10 градусов, R7= около 6 с
N280 DO $R8=$AC_TIME	

Программный код	Комментарий
N290 X0.001 A10	; Подача= 2540 мм/мин, ход траектории= 10 мм, R8= около 0.288 с
N300 FGREF[A]=360/(2*$PI)	; 1 градус =1 дюйм, установить через эффективный радиус.
N310 DO $R9=$AC_TIME	
N320 X0.001 A10	; Подача= 2540 мм/мин, ход траектории= 254 мм, R9= около 6 с
N330 M30	

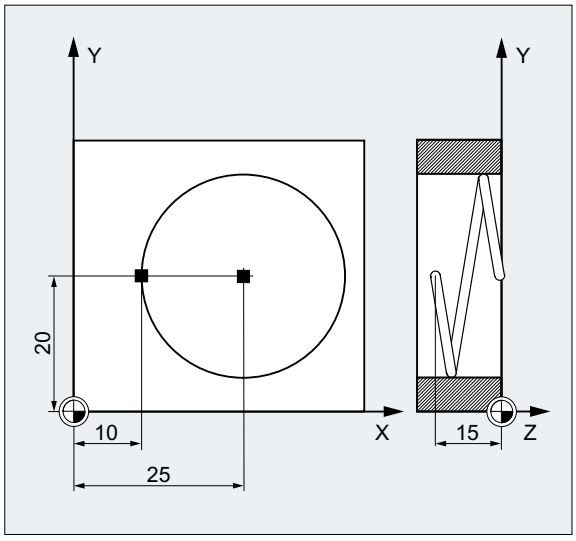
Пример 2: Перемещение синхронных осей с предельной скоростью FL

Скорость движения по траектории траекторных осей уменьшается, если синхронная ось Z достигает предельной скорости.

Программный код
N10 G0 X0 Y0
N20 FGROUP(X)
N30 G1 X1000 Y1000 G94 F1000 FL[Y]=500
N40 Z-50

Пример 3: Винтовая интерполяция

Траекторные оси X и Y двигаются с запрограммированной подачей, ось подачи Z является синхронной осью.



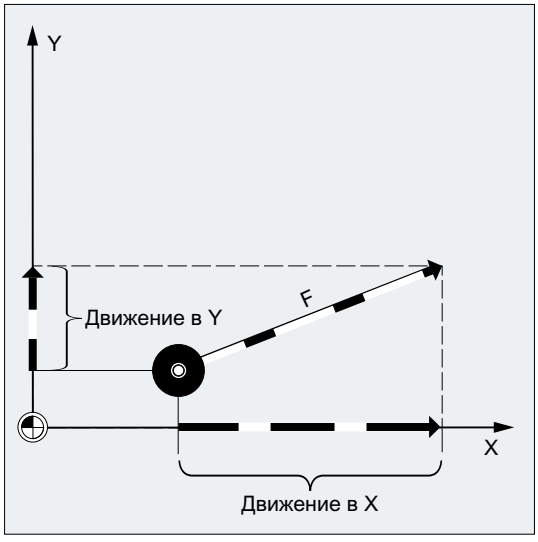
Программный код	Комментарий
N10 G17 G94 G1 Z0 F500	; Подача инструмента.
N20 X10 Y20	; Подвод в стартовую позицию.
N25 FGROUP(X,Y)	; Оси X/Y это траекторные оси, Z это синхронная ось.

Программный код	Комментарий
`N30 G2 X10 Y20 Z-15 I15 J0 F1000 FL[Z]=200`	; На круговой траектории действует подача 1000 мм/мин, перемещение в направлении `Z` осуществляется синхронно.
...	
`N100 FL[Z]=$MA_AX_VELO_LIMIT[0,Z]`	; Через считывание скорости из `MD` предельная скорость отключается, считывание значения из `MD`.
`N110 M30`	; Конец программы.

Дополнительная информация

Скорость подачи для траекторных осей (F)

Обычно подача по траектории складывается из отдельных компонентов скорости всех участвующих в движении геометрических осей и относится к центру фрезы или к острию токарного резца.



Скорость подачи указывается по адресу `F`. В зависимости от предустановки в машинных данных действуют определенные через команды G единицы измерения в мм или дюймах.

На кадр ЧПУ может быть запрограммировано одно значение `F`. Единица скорости подачи определяется через одну из G-команд `G93`/`G94`/`G95`. Подача `F` воздействует только на траекторные оси и действует до тех пор, пока не будет запрограммировано новое значение подачи. После адреса `F` допускаются разделительные символы.

Примеры:

`F100` или `F 100`

`F.5`

`F=2*FEED`

Тип подачи (G93/G94/G95)

G-команды G93, G94 и G95 действуют модально. Если выполняется переключение между G93, G94 и G95, то необходимо заново запрограммировать значение подачи по траектории. Для обработки с круговыми осями подача может указываться и в градусах/мин.

Обратная по времени подача (G93)

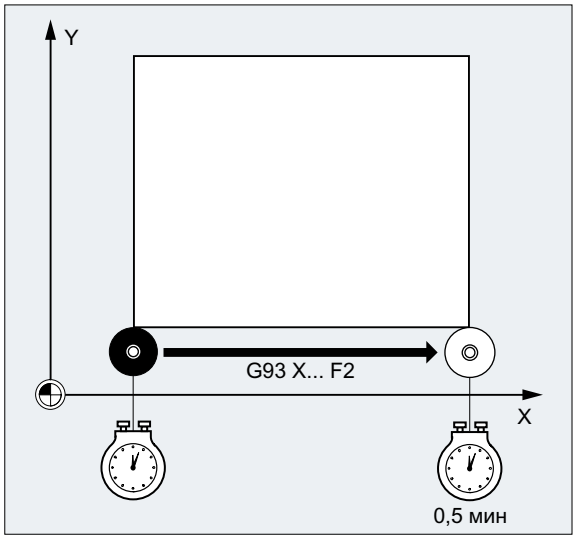
Обратная по времени подача указывает продолжительность прохода одного кадра.

Единица: 1/мин

Пример:

```
N10 G93 G01 X100 F2
```

Означает: запрограммированный ход траектории проходится за 0,5 мин.



Примечание

Если длины траектории от кадра к кадру сильно различаются, то для G93 в каждом кадре должно быть определено новое значение F. Для обработки с круговыми осями подача может указываться и в градусах/мин.

Подача для синхронных осей

Запрограммированная по адресу F подача действует для всех запрограммированных в кадре траекторных осей, но не для синхронных осей. Управление синхронными осями осуществляется таким образом, что им для их перемещения необходимо то же время, что и для траекторных осей и все оси достигают их конечной точки в одно и то же время.

Предельная скорость для синхронных осей (FL)

С помощью команды FL для синхронных осей может быть запрограммирована предельная скорость. Если FL не программируется, то действует скорость ускоренного хода. Отключение FL осуществляется присвоением MD (MD36200 $MA_AX_VELO_LIMIT).

Перемещение траекторной оси как синхронной оси (FGROUP)

С помощью FGROUP определяется, должна ли траекторная ось перемещаться с подачей по траектории или как синхронная ось. При винтовой интерполяции можно, к примеру,

установить, что только две геометрические оси X и Y должны перемещаться с запрограммированной подачей. В этом случае ось подачи Z была бы синхронной осью.

Пример: `FGROUP(X,Y)`

Изменить FGROUP

Изменить сделанную с `FGROUP` установку можно:

1. через повторное программирование `FGROUP`: к примеру, `FGROUP(X,Y,Z)`
2. через программирование `FGROUP` без указания оси: `FGROUP()`
   После `FGROUP()` действует установленное в машинных данных исходной состояние. Гео-оси теперь снова движутся в структуре траекторных осей.

Примечание

Идентификаторами осей для `FGROUP` должны быть имена осей канала.

Единицы измерения для подачи F

С помощью G-команд `G700` и `G710` дополнительно в геометрическим данным определяется и система единиц для подач `F`, т.е.:

- для `G700`: [дюймов/мин]
- для `G710`: [мм/мин]

Примечание

`G70`/`G71` **не** влияют на данные подачи.

Единица измерения для синхронных осей с предельной скоростью FL

Установленная для `F` через G-команду `G700`/`G710` единица измерения действует и для `FL`.

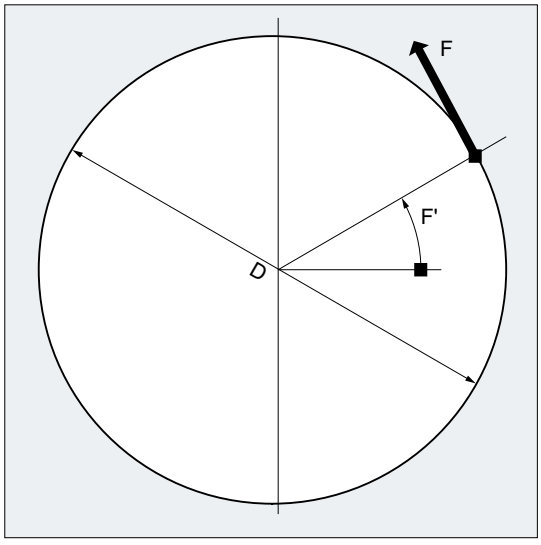
Единица измерения для круговых и линейных осей

Для линейных и круговых осей, которые связаны друг с другом через `FGROUP` и вместе движутся по траектории, действует подача в единице измерения линейных осей. В зависимости от предустановки с `G94`/`G95` в мм/мин или дюймов/мин или мм/оборот или дюймов/оборот.

Тангенциальная скорость круговой оси в мм/мин или дюймов/мин вычисляется по формуле:

F[мм/мин] = F'[градусов/мин] * π * D[мм] / 360[градусов]

где: F: тангенциальная скорость
F': угловая скорость
π: постоянная окружности
D: диаметр



Перемещение круговых осей со скоростью движения по траектории F (FGREF)

Для процессов обработки, при которых инструмент или деталь или обе приводятся в движение круговой осью, должна существовать возможность программирования эффективной подачи обработки как подачи по траектории через значение F. Для этого для каждой из участвующих круговых осей должен быть указан эффективный радиус (исходный радиус).

Единица исходного радиуса зависит от установки G70/G71/G700/G710.

Чтобы способствовать вычислению подачи по траектории, все участвующие оси должны быть включены в команду FGROUP.

Для поддержания совместимости с процессами без программирования FGREF, после пуска системы и при RESET действует нормирование 1 градус = 1 мм. Это соответствует исходному радиусу FGREF=360 мм / (2π) = 57.296 мм.

Примечание

Эта предустановка не зависит от активной основной системы (MD10240 $MN_SCALING_SYSTEM_IS_METRIC) и от актуальной действующей установки G70/G71/G700/G710.

Особенности:

Программный код

```
N100 FGROUP(X,Y,Z,A)
N110 G1 G91 A10 F100
N120 G1 G91 A10 X0.0001 F100
```

При этом программировании запрограммированное значение F в N110 нормируется как подача круговой оси в градусах/мин, в то время как нормированием подачи в N120 в

зависимости от актуальной действующей установки G70/G71/G700/G710 является либо 100 дюймов/мин или 100 мм/мин.

ВНИМАНИЕ
Разница подач Нормирование FGREF действует и тогда, когда в кадре запрограммированы только круговые оси. Обычная интерпретация значения F в градусах/мин действует в этом случае только тогда, когда отношение радиуса соответствует предварительной установке FGREF: • для G71/G710: FGREF[A]=57.296 • для G70/G700: FGREF[A]=57.296/25.4

Чтение исходного радиуса

Значение исходного радиуса круговой оси может быть считано через системные переменные:

- В синхронных действиях или с остановкой предварительной обработки в программе обработки детали через системные переменные:

  $AA_FGREF[<ось>] Актуальное значение главного хода

- Без остановки предварительной обработки в программе обработки детали через системные переменные:

  $PA_FGREF[<ось>] Запрограммированное значение

Если значения не запрограммированы, то в обеих переменных для круговых осей считывается предустановка 360 мм / (2π) = 57.296 мм (соответствует 1 мм на градус).

Для линейных осей в обеих переменных всегда считывается значение 1 мм.

Чтение определяющих скорость траекторных осей

Участвующие в траекторной интерполяции оси могут считываться через системные переменные:

- В синхронных действиях или с остановкой предварительной обработки в программе обработки детали через системные переменные:

  $AA_FGROUP[<ось>] Выводит значение "1", если указанная ось через первичную установку или через программирование FGROUP влияет на скорость движения по траектории в актуальном кадре главного хода. Если нет, то переменная выводит значение "0".

  $AC_FGROUP_MASK Выводит битовый ключ запрограммированных с FGROUP осей канала, которые должны содействовать скорости движения по траектории.

- Без остановки предварительной обработки в программе обработки детали через системные переменные:

$PA_FGROUP[<ось>]	Выводит значение "1", если указанная ось через первичную установку или через программирование FGROUP влияет на скорость движения по траектории. Если нет, то переменная выводит значение "0".
$P_FGROUP_MASK	Выводит битовый ключ запрограммированных с FGROUP осей канала, которые должны содействовать скорости движения по траектории.

Коэффициенты соотношения траекторий для осей ориентации с FGREF

Для осей ориентации принцип действия коэффициентов FGREF[] зависит от того, осуществляется ли изменение ориентации инструмента через интерполяцию круговой оси или векторную интерполяцию.

При **интерполяции круговой оси** соответствующие коэффициенты FGREF осей ориентации учитываются как для круговых осей по отдельности как исходный радиус для хода осей.

При **векторной интерполяции** активируется эффективный коэффициент FGREF, который определяется как геометрическое среднее значение из отдельных коэффициентов FGREF:

FGREF[эфф.] = n-ный корень из:[(FGREF[A] * FGREF[B]...)]

где: A: идентификатор 1-ой оси ориентации
B: идентификатор 2-ой оси ориентации
C: идентификатор 3-ей оси ориентации
n: Число осей ориентации

Пример:

Для стандартной 5-осевой трансформации существует две оси ориентации и тем самым эффективный коэффициент как корень из результата двух осевых коэффициентов:

FGREF[эфф.] = квадратный корень из:[(FGREF[A] * FGREF[B])]

Примечание

С помощью эффективного коэффициента для осей ориентации FGREF может быть определена исходная точка на инструменте, к которой относится запрограммированная подача по траектории.

3.7.2 Перемещение позиционирующих осей (POS, POSA, POSP, FA, WAITP, WAITMC)

Позиционирующие оси перемещаются независимо от траекторных осей со своей специфической для оси подачей. Команды интерполяции не действуют. С помощью команд POS/POSA/POSP осуществляется перемещение позиционирующих осей и одновременная координация процессов движения.

Типичными примерами для позиционирующих осей являются:

- устройства загрузки палет
- измерительные установки

С помощью WAITP можно обозначить в программе ЧПУ место, на котором осуществляется ожидание до тех пор, пока запрограммированная в одном из предыдущих кадров ЧПУ в POSA ось не достигнет своей конечной позиции.

При WAITMC при поступлении указанной метки ожидания сразу же устанавливается следующий кадр ЧПУ.

Синтаксис

`POS[`<ось>`]=`<позиция>

`POSA[`<ось>`]=`<позиция>

`POSP[`<ось>`]=(`<конечная позиция>`,`<длина фрагмента>`,`<режим>`)`

`FA[`<ось>`]=`<значение>

`WAITP(`<ось>`)` ; Программирование в отдельном кадре ЧПУ!

`WAITMC(`<метка ожидания>`)`

Объяснение

`POS` / `POSA`:	Переместить позиционирующую ось на указанную позицию `POS` и `POSA` имеют идентичную функциональность, но отличаются в параметрах смены кадра: • С `POS` последовательное включение кадра ЧПУ происходит только после достижения позиции подвода. • С `POSA` последовательное включение кадра ЧПУ происходит и если позиции подвода не достигнута.		
	<ось>:	Имя перемещаемой оси (идентификатор оси канала или гео-оси)	
	<позиция>:	Позиция оси для подвода	
		Тип:	REAL

POSP:	Постепенное перемещение позиционирующей оси на указанную конечную позицию		
	<конечная позиция>:	Конечная позиция оси для подвода	
	<длина фрагмента>:	Длина фрагмента	
	<режим>:	Режим подвода	
		= 0:	Для обоих последних фрагментов выполняется разделение оставшегося пути до конечной позиции на два заключительных фрагмента равного размера (предустановка).
		= 1:	Длина фрагмента согласуется таким образом, что сумма всех вычисленных длин фрагментов точна равна пути до конечной позиции.
	Примечание: POSP используется специально для программирования маятниковых движений (Страница 993).		
FA:	Подача для указанной позиционирующей оси		
	<ось>:	Имя перемещаемой оси (идентификатор оси канала или гео-оси)	
	<значение>:	Скорость подачи	
		Единица:	мм/мин или дюйм/мин или градус/мин
	Примечание: На кадр ЧПУ может быть запрограммировано макс. 5 значения FA.		
WAITP:	Ожидать завершения перемещения позиционирующей оси Обработка последующих кадров не начинается до тех пор, пока указанная и запрограммированная в предшествующем кадре с помощью POSA позиционирующая ось не достигнет своей конечной позиции (с точным остановом точным).		
	<ось>:	Имя оси (идентификатор оси канала или гео-оси), для которой должна действовать команда WAITP	
	Примечание: С помощью WAITP можно разрешить ось в качестве качающейся оси или для перемещения в качестве конкурирующей позиционирующей оси (через PLC).		
WAITMC:	Ожидать поступления указанной метки ожидания При поступлении метки ожидания сразу же устанавливается следующий кадр ЧПУ.		
	<метка ожидания>:	Номер метки ожидания	

⚠ ОСТОРОЖНО

Движение с POSA

Если в последующем кадре считывается команда, которая не явно создает остановку предварительной обработки, то последующий кадр выполняется только после того, как полностью выполнены все подготовленные и сохраненные до этого кадры.
Предыдущий кадр останавливается с точным остановом (как для G9).

Примеры

Пример 1: Движение с POSA и доступ к данным состояния станка

При обращении к данным состояния станка ($A...) СЧПУ осуществляет внутреннюю остановку предварительной обработки. Обработка останавливается до тех пор, пока не будут полностью выполнены все прежде подготовленные и сохраненные кадры.

Программный код	Комментарий
`N40 POSA[X]=100`	
`N50 IF $AA_IM[X]==R100 GOTOF MARKE1`	; Обращение к данным состояния станка.
`N60 G0 Y100`	
`N70 WAITP(X)`	
`N80 MARKE1:`	
`N...`	

Пример 2: Ожидание конца перемещения с WAITP

Устройство подачи паллет

Ось U: накопитель палет

Перемещение палеты с обрабатываемыми деталями в рабочую зону

Ось V: система передачи к измерительной станции, на которой осуществляется параллельный процессу выборочный контроль качества

Программный код	Комментарий
`N10 FA[U]=100 FA[V]=100`	; Специфические для осей параметры подачи для отдельных позиционирующих осей `U` и `V`.
`N20 POSA[V]=90 POSA[U]=100 G0 X50 Y70`	; Перемещение позиционирующих и траекторных осей.
`N50 WAITP(U)`	; Выполнение программы продолжается только после достижения осью `U` запрограммированной в `N20` позиции.
…	

Дополнительная информация

Движение с POSA

`POSA` не влияет на последовательное включение кадра или выполнение программы. Движение к конечной точке может осуществляться параллельно с обработкой последующих кадров ЧПУ.

Движение с POS

Последовательное включение кадра осуществляется только после того, как все запрограммированные в `POS` оси достигнут своих конечных позиций.

Ожидание конца перемещения с WAITP

После WAITP ось считается более не занятой из программы ЧПУ до тех пор, пока она не будут запрограммирована заново. Эта ось после может использоваться через PLC как позиционирующая ось, или из программы ЧПУ/PLC или HMI как качающаяся ось.

Смена кадра на рампе торможения с IPOBRKA и WAITMC

Ось затормаживается только тогда, когда метка ожидания еще не достигнута или иной критерий окончания кадра препятствует смене кадров. После WAITMC ось сразу же запускается, если иной критерий окончания кадра не препятствует смене кадров.

3.7.3 Режим ориентации шпинделя (SPCON, SPCOF)

Команды SPCON или SPCOF включают или выключают режим управления положением шпинделя явным образом.

Примечание

Для включения режима управления положением посредством SPCON необходимо не более трех тактов регулятора положения.

Синтаксис

```
SPCON
SPCON(<n>)
SPCON(<n>,<m>,...)
SPCOF
SPCOF(<n>)
SPCOF(<n>,<m>,...)
```

Объяснение

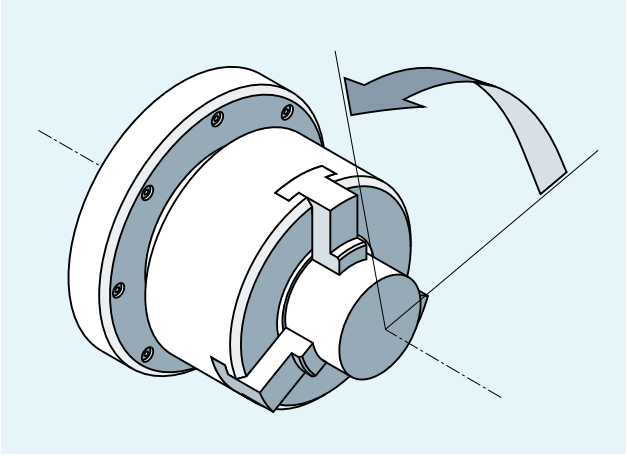
SPCON:	Включить режим управления положением Указанный шпиндель переключается из управления скоростью в управление положением. SPCON действует модально и сохраняется до SPCOF.
SPCOF:	Выключить режим управления положением Указанный шпиндель переключается из управления положением в управление скоростью.
<n>,<m>,...:	Номера шпинделя Без указания номера шпинделя: мастер-шпиндель канала

Примечание

При наличии синхронного шпинделя с соединением по заданному значению запрещено переключать ведущий шпиндель посредством SPCOF в режим управления числом оборотов.

3.7.4 Позиционирование шпинделей (SPOS, SPOSA, M19, M70, WAITS)

С помощью SPOS, SPOSA или M19 шпиндели могут позиционироваться в определенные наклонные положения, к например, при смене инструмента.



SPOS, SPOSA и M19 вызывают временное переключение в режим ориентации до следующего M3/M4/M5/M41 ... M45.

Позиционирование в осевом режиме

Шпиндель может перемещаться и по своему определенному в машинных данных адресу как траекторная, синхронная или позиционирующая ось. После указания идентификатора оси шпиндель находится в осевом режиме. С помощью M70 шпиндель включается непосредственно в осевой режим.

Конец позиционирования

Критерий окончания движения при позиционировании шпинделя может быть запрограммирован через FINEA, CORSEA, IPOENDA или IPOBRKA.

Если критерии окончания движения для всех обрабатываемых в кадре шпинделей или осей, а также критерий смены кадров для траекторной интерполяции выполнены, то осуществляется смена кадра.

Синхронизация

Для синхронизации движений шпинделя с помощью WAITS можно ожидать достижения позиции шпинделя.

Условия

Позиционируемый шпиндель должен быть способен работать в режиме ориентации.

Синтаксис

Позиционировать шпиндель:

`SPOS`=<значение> / `SPOS[<n>]`=<значение>

`SPOSA`=<значение> / `SPOSA[<n>]`=<значение>

`M19` / `M<n>=19`

Переключить шпиндель в осевой режим:

`M70` / `M<n>=70`

Определить критерий окончания движения:

`FINEA` / `FINEA[S<n>]`

`COARSEA` / `COARSEA[S<n>]`

`IPOENDA` / `IPOENDA[S<n>]`

`IPOBRKA` / `IPOBRKA(`<ось>[,<момент времени>]) ; Программирование в отдельном кадре ЧПУ!

Синхронизировать движения шпинделя:

`WAITS` / `WAITS(<n>,<m>)` ; Программирование в отдельном кадре ЧПУ!

Объяснение

SPOS / SPOSA:	Позиционировать шпиндель в указанное наклонное положение SPOS и SPOSA имеют идентичную функциональность, но отличаются в параметрах смены кадра: • Со SPOS последовательное включение кадра ЧПУ происходит только после достижения позиции. • Со SPOSA последовательное включение кадра ЧПУ происходит, и если позиция не достигнута.			
	<n>:	Номер шпинделя, который должен быть позиционирован. Без указания номера шпинделя или с номером шпинделя "0" SPOS или SPOSA относится к мастер-шпинделю.		
	<значение>:	Наклонное положение, в которое должен быть позиционирован шпиндель		
		Единица:	Градус	
		Тип:	REAL	
		Для программирования режима подвода к позиции существуют следующие возможности:		
		=AC(<значение>):	Абсолютное указание размера	
			Диапазон значений:	0 ... 359,9999
		=IC(<значение>):	Инкрементальное указание размера	
			Диапазон значений:	0 ... ±99 999,999
		=DC(<значение>):	Подвод по прямому пути к абсолютному значению	
		=ACN(<значение>):	Абсолютное указание размера, подвод в отрицательном направлении	
		=ACP(<значение>):	Абсолютное указание размера, подвод в положительном направлении	
		=<значение>:	как DC(<значение>)	
M<n>=19:	Позиционировать мастер-шпиндель (M19 или M0=19) или шпиндель с номером <n> (M<n>=19) в заданное с SD43240 $SA_M19_SPOS наклонное положение в заданном в SD43250 $SA_M19_SPOSMODE режиме подвода к позиции Последовательное включение кадра ЧПУ только после достижения позиции.			
M<n>=70:	Переключить мастер-шпиндель (M70 или M0=70) или шпиндель с номером <n> (M<n>=70) в осевой режим Переход к определенной позиции не выполняется. Последовательное включение кадра ЧПУ после выполнения переключения.			
FINEA:	Завершение движения, если достигнут "Точный останов точный"			
COARSEA:	Завершение движения, если достигнут "Точный останов грубый"			
IPOENDA:	Завершение движения, если достигнут "Останов интерполятора"			

S<n>:	Шпиндель, для которого должен действовать запрограммированный критерий окончания движения		
	<n>:	Номер шпинделя	
	Без указания шпинделя [S<n>] или с номером шпинделя "0" запрограммированный критерий окончания движения относится к мастер-шпинделю.		
IPOBRKA:	Смена кадра на рампе торможения возможна		
	<ось>:	Идентификатор оси канала	
	<момент времени>:	Момент времени смены кадра, относительно рампы торможения	
		Единица:	Процент
		Диапазон значений:	100 (момент включения рампы торможения) ... 0 (конец рампы торможения)
		Без указания параметра <момент времени> вступает в силу актуальное значение установочных данных: SD43600 $SA_IPOBRAKE_BLOCK_EXCHANGE **Примечание:** IBOBRKA с моментом времени "0" идентична IPOENDA.	
WAITS:	Команда синхронизации для указанного(ых) шпинделя(ей) Обработка последующих кадров не начинается до тех пор, пока указанный(ые) и запрограммированный(ые) в предшествующем кадре с помощью SPOSA шпиндель(и) не достигнет(ут) своей позиции (с точным остановом точным).		
	WAITS после M5:	Ожидать остановки указанного(ых) шпинделя(ей).	
	WAITS после M3/M4:	Ожидать достижения указанным(и) шпинделем(ями) своей заданной скорости.	
	<n>,<m>:	Номера шпинделей, для которых должна действовать команда синхронизации Без указания номера шпинделя или с номером шпинделя "0" WAITS относится к мастер-шпинделю.	

Примечание

На кадр ЧПУ возможно 3 указания позиции шпинделя.

Примечание

При инкрементальном указании размера IC(<значение>) позиционирование шпинделя возможно на несколько оборотов.

Примечание

Если перед SPOS было включено управление положением со SPCON, то оно сохраняется до SPCOF.

Примечание

СЧПУ на основе последовательности программирования самостоятельно определяет переход в осевой режим. Поэтому явного программирования M70 в программе обработки детали более не требуется. Но M70 может продолжать программироваться, чтобы, например, улучшить читабельность программы обработки детали.

Дополнительная информация

Позиционирование со SPOSA

SPOSA не влияет на последовательное включение кадра или выполнение программы. Позиционирование шпинделя может осуществляться параллельно с выполнением последующих кадров ЧПУ. Смена кадра происходит после достижения всеми запрограммированными в кадре функциями (кроме шпинделя) их критерия конца кадра. Позиционирование шпинделя при этом может растягиваться на несколько кадров (см. WAITS).

Примечание

Если в одном из последующих кадров считывается команда, не явно создающая остановку предварительной обработки, то обработка останавливается в этом кадре до остановки всех позиционирующих шпинделей.

Позиционирование со SPOS / M19

Последовательное включении кадра осуществляется только после того, как все запрограммированные в кадре функции достигли своего критерия конца кадра (к примеру, все вспомогательные функции были квитированы с PLC, все оси достигли конечной точки) и шпиндель достиг запрограммированной позиции.

Скорость движений:

Скорость и характеристика торможения для позиционирования зафиксированы в машинных данных. Сконфигурированные значения могут быть изменены через программирование или через синхронные действия, см.:

- Подача для позиционирующих осей/шпинделей (FA, FPR, FPRAON, FPRAOF) (Страница 129)
- Программируемая коррекция ускорения (ACC) (Страница 133)

Указание позиций шпинделя:

Т.к. команды `G90`/`G91` здесь не действуют, явно действуют соответствующие указания размеров, к примеру, `AC`, `IC`, `DC`, `ACN`, `ACP`. Перемещение без параметров автоматически осуществляется как при параметре `DC`.

Синхронизация движений шпинделя с WAITS

С помощью `WAITS` в программе ЧПУ может быть обозначено место, на котором осуществляется ожидание достижения одним или несколькими запрограммированными в более раннем кадре ЧПУ в `SPOSA` шпинделями их позиции.

Пример:

Программный код	Комментарий
`N10 SPOSA[2]=180 SPOSA[3]=0`	
...	
`N40 WAITS(2,3)`	; Ожидание в кадре осуществляется до тех пор, пока шпинделя 2 и 3 не достигнут позиций, указанных в кадре `N10`.

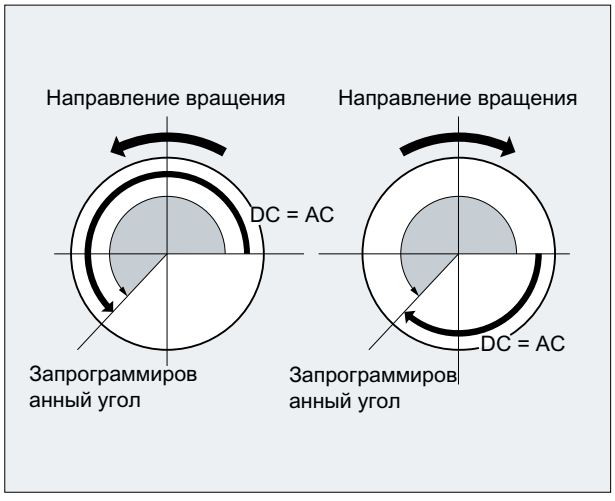
После `M5` с помощью `WAITS` можно ожидать перехода шпинделя(ей) в состояние покоя. После `M3`/`M4` с помощью `WAITS` можно ожидать достижения шпинделем(ями) заданной скорости/направления вращения.

Примечание

Если шпиндель еще не синхронизирован с синхронными метками, то положительное направление вращения берется из машинных данных (состояние при поставке).

Позиционирование шпинделя из вращения (M3/M4)

При включенных `M3` или `M4` шпиндель останавливается на запрограммированном значении.



Параметры `DC` и `AC` идентичны. В обоих случаях вращение продолжается в выбранном через `M3`/`M4` направлении до абсолютной конечной позиции. При `ACN` и `ACP` при необходимости осуществляется торможение и удержание соответствующего направления подвода. При указании `IC` вращение, исходя из актуальной позиции шпинделя, продолжается на указанное значение.

Позиционирование шпинделя из состояния покоя (M5)

Запрограммированный путь проходится точно из состояния покоя (`M5`) в соответствии с введенными данными.

3.7.5 Подача для позиционирующих осей/шпинделей (FA, FPR, FPRAON, FPRAOF)

Наряду с этим существует возможность получения для траекторных или синхронных осей или для отдельных позиционирующих осей/шпинделей от другой круговой оси или шпинделя.

Позиционирующие оси, к примеру, системы транспортировки детали, револьвер или люнеты, перемещаются независимо от траекторных и синхронных осей. Поэтому для каждой позиционирующей оси определяется собственная подача.

И для шпинделей можно запрограммировать собственную осевую подачу.

Синтаксис

Подача для позиционирующей оси:
`FA[`<ось>`]=...`

Осевая подача для шпинделя:
`FA[SPI(<n>)]=...`
`FA[S<n>]=...`

Получение окружной подачи для траекторных/синхронных осей:

`FPR(`<круговая ось>`)`

`FPR(SPI(<n>))`

`FPR(S<n>)`

Получение окружной подачи для позиционирующих осей/шпинделей:

`FPRAON(`<ось>`,`<круговая ось>`)`

`FPRAON(`<ось>`,SPI(<n>))`

`FPRAON(`<ось>`,S<n>)`

`FPRAON(SPI(<n>),`<круговая ось>`)`

`FPRAON(S<n>,`<круговая ось>`)`

`FPRAON(SPI(<n>),SPI(<n>))`

`FPRAON(S<n>,S<n>)`

`FPRAOF(`<ось>`,SPI(<n>),...)`

`FPRAOF(`<ось>`,S<n>,...)`

Значение

`FA[...]=...:`	Подача для указанной позиционирующей оси или скорость позиционирования (осевая подача) для указанного шпинделя	
	Единица:	мм/мин или дюйм/мин или градус/мин
	Диапазон значений:	... 999 999,999 мм/мин, градусов/мин ... 39 999,9999 дюймов/мин

FPR(...):	С FPR обозначается круговая ось (<круговая ось>) или шпиндель (SPI(<n>) / S<n>), от которой должна быть получена запрограммированная в G95 окружная подача для траекторных и синхронных осей.
FPRAON(...):	Получение окружной подачи для позиционирующих осей и шпинделей Первый параметр (<ось> / SPI(<n>) / S<n>) обозначает позиционирующую ось/шпиндель, которая должна перемещаться с окружной подачей. Второй параметр (<круговая ось> / SPI(<n>) / S<n>) обозначает круговую ось/шпиндель, от которой должна быть получена окружная подача. **Примечание:** Второй параметр может быть не нужен, тогда подача отводится от мастер-шпинделя.
FPRAOF(...):	С FPRAOF полученная окружная подача для указанных осей или шпинделей отменяется.
<ось>:	Идентификатор оси (позиционирующая или гео-ось)
SPI(<n>) / S<n>:	Идентификатор шпинделя SPI(<n>) и S<n> функционально идентичны.
	<n>: Номер шпинделя
	Примечание: SPI преобразует номер шпинделя в идентификатор оси. Передаваемый параметр (<n>) должен содержать действительный номер шпинделя.

Примечание

Запрограммированная подача FA[...] действует модально.

На кадр ЧПУ может быть запрограммировано макс. 5 подач для позиционирующих осей/шпинделей.

Примечание

Отводимая подача вычисляется по следующей формуле:

Отводимая подача =
запрограммированная подача * значение главной подачи

Примеры

Пример 1: Соединение синхронных шпинделей

Для соединения синхронных шпинделей скорость позиционирования ведомого шпинделя может программироваться независимо от ведущего шпинделя – к примеру, для позиционирования.

Программный код	Комментарий
...	
FA[S2]=100	; Скорость позиционирования ведомого шпинделя (шпиндель 2) = 100 градусов/мин
...	

Пример 2: Отводимая окружная подача для траекторных осей

Траекторные оси X, Y должны перемещаться с окружной подачей, отводимой от круговой оси A:

Программный код
...
N40 FPR(A)
N50 G95 X50 Y50 F500
...

Пример 3: Получение окружной подачи для мастер-шпинделя

Программный код	Комментарий
N30 FPRAON(S1,S2)	; Окружная подача для мастер-шпинделя (S1) должна отводиться от шпинделя 2.
N40 SPOS=150	; Позиционировать мастер-шпиндель.
N50 FPRAOF(S1)	; Отключить отводимую окружную подачу для мастер-шпинделя.

Пример 4: Получение окружной подачи для позиционирующей оси

Программный код	Комментарий
N30 FPRAON(X)	; Окружная подача для позиционирующей оси X должна отводиться от мастер-шпинделя.
N40 POS[X]=50 FA[X]=500	; Позиционирующая ось движется с 500 мм/оборот мастер-шпинделя.
N50 FPRAOF(X)	

Дополнительная информация

FA[...]

Всегда действует тип подачи G94. Если G70/G71 активна, то единица измерения метрическая/дюймовая зависит от предварительной установки в машинных данных. С помощью G700/G710 единица измерения может изменяться в программе.

Примечание

Если FA не запрограммирована, то действует установленное в машинных данных значение.

FPR(...)

С помощью FPR в качестве расширения команды G95 (окружная подача относительно мастер-шпинделя) окружная подача может отводиться от любого шпинделя или круговой оси. G95 FPR(...) действует для траекторных и синхронных осей.

Если обозначенная с FPR круговая ось/шпиндель работает с ориентацией, то действует соединение по заданному значению, в иных случаях – соединение по фактическому значению.

FPRAON(...)

С помощью FPRAON возможен осевой отвод окружной подачи для позиционирующих осей и шпинделей от мгновенной подачи другой круговой оси или шпинделя.

FPRAOF(...)

С помощью FPRAOF можно выключить окружную подачу для одной или нескольких осей/шпинделей одновременно.

3.7.6 Программируемая коррекция подачи (OVR, OVRRAP, OVRA)

Скорость траекторных/позиционирующих осей и шпинделей может изменяться в программе ЧПУ.

Синтаксис

```
OVR=<значение>
OVRRAP=<значение>
OVRA[<ось>]=<значение>
OVRA[SPI(<n>)]=<значение>
OVRA[S<n>]=<значение>
```

Объяснение

OVR:	Изменение подачи для подачи по траектории F	
OVRRAP:	Изменение подачи для скорости ускоренного хода	
OVRA:	Изменение подачи для подачи позиционирования FA или для скорости шпинделя S	
<ось>:	Идентификатор оси (позиционирующая или гео-ось)	
SPI(<n>) / S<n>:	Идентификатор шпинделя SPI(<n>) и S<n> функционально идентичны.	
	<n>:	Номер шпинделя
	Примечание: SPI преобразует номер шпинделя в идентификатор оси. Передаваемый параметр (<n>) должен содержать действительный номер шпинделя.	

<значение>:	Изменение подачи в процентах Значение относится или накладывается на установленную на станочном пульте процентовку подачи.	
	Диапазон значений:	0 % ... 200 %, целочисленные
	Примечание: При коррекции траектории и ускоренного хода установленные в машинных данных максимальные скорости не превышаются.	

3.7.7 Программируемая коррекция ускорения (ACC)

В критических сегментах программы может возникнуть необходимость, ограничить ускорение до максимального возможного значения, чтобы, к примеру, избежать механических вибраций.

С помощью программируемой коррекции ускорения для каждой траекторной оси или шпинделя через команду в программе ЧПУ может быть изменено ускорение. Ограничение действует во всех типах интерполяции. В качестве 100 % ускорения действуют установленные в машинных данных значения.

Синтаксис

```
ACC[<ось>]=<значение>
ACC[SPI(<n>)]=<значение>
ACC(S<n>)=<значение>
```

Выключение:

```
ACC[...]=100
```

Синтаксис

ACC:	Изменение ускорения для указанной траекторной оси или изменение скорости для указанного шпинделя	
<ось>:	Имя оси канала траекторной оси	
SPI(<n>) / S<n>:	Идентификатор шпинделя SPI(<n>) и S<n> функционально идентичны.	
	<n>:	Номер шпинделя
	Примечание: SPI преобразует номер шпинделя в идентификатор оси. Передаваемый параметр (<n>) должен содержать действительный номер шпинделя.	
<значение>:	Изменение ускорения в процентах Значение относится или накладывается на установленную на станочном пульте процентовку подачи.	
	Диапазон значений:	1 ... 200%, целочисленные

Примечание

При большем ускорении могут быть превышены разрешенные изготовителем станка значения.

Пример

Программный код	Комментарий
`N50 ACC[X]=80`	; Осевые салазки в направлении X должны перемещаться только с ускорением 80%.
`N60 ACC[SPI(1)]=50`	; Шпиндель 1 должен ускоряться или тормозиться только с 50% способности разгона.

Дополнительная информация

Запрограммированная с ACC коррекция ускорения

Запрограммированная с ACC[...] коррекция ускорения всегда учитывается как в системной переменной $AA_ACC при выводе. Выгрузка в программе обработки детали и в синхронных действиях происходит в различные моменты времени в обработке ЧПУ.

В программе обработки детали

Записанное в программе обработки детали значение учитывается в системной переменной $AA_ACC как записано в программе обработки детали, только если ACC за прошедшее время не изменялась из синхронного действия.

В синхронных действиях

Соответственно действует: Записанное в синхронном действии значение учитывается в системной переменной $AA_ACC как записано в синхронном действии, только если ACC за прошедшее время не изменялась из программы обработки детали.

Заданное ускорение может изменяться и через синхронные действия.

Пример:

Программный код
...
`N100 EVERY $A_IN[1] DO POS[X]=50 FA[X]=2000 ACC[X]=140`

Актуальное значение ускорения может быть запрошено с помощью системной переменной $AA_ACC[<ось>]. Через машинные данные можно установить, какое значение должно действовать при RESET / завершении программы обработки детали — последнее установленное значение ACC или 100 %.

3.7.8 Подача с наложением маховичка (FD, FDA)

С помощью команд FD и FDA можно маховичками перемещать оси в процессе выполнения программы обработки детали. При этом на запрограммированные движения перемещения осей накладываются нормированные как данные хода или скорости сигналы маховичка.

Траекторные оси
Для траекторных осей возможно наложение запрограммированной подачи по траектории. При этом обрабатывается маховичок 1-ой гео-оси канала. Нормированные в зависимости от направления вращения импульсы маховичка на такт IPO соответствуют накладываемой скорости движения по траектории. Предельные значения скорости движения по траектории, которые могут быть достигнуты через наложение маховичка:

- Минимальное значение: 0
- Максимальное значение: Предельные значения машинных данных участвующих в движении перемещения траекторных осей

Примечание

Подача по траектории

Подача по траектории F и подача маховичка FD не могут быть запрограммированы в одном кадре ЧПУ.

Позиционирующие оси
Для позиционирующих осей возможно осевое наложение пути перемещения или скорости. При этом обрабатывается согласованный с осью маховичок.

- Наложение перемещения
  Нормированные в зависимости от направления вращения импульсы маховичка соответствуют пути, который должен быть пройден осью. При этом учитываются только импульсы маховичка в направлении к запрограммированной позиции.
- Наложение скорости
  Нормированные в зависимости от направления вращения импульсы маховичка на такт IPO соответствуют накладываемой осевой скорости. Предельные значения скорости движения по траектории, которые могут быть достигнуты через наложение маховичка:
  - Минимальное значение: 0
  - Максимальное значение: Предельные значения машинных данных позиционирующей оси

Синтаксис

```
FD=<скорость>
FDA[<ось>]=<скорость>
```

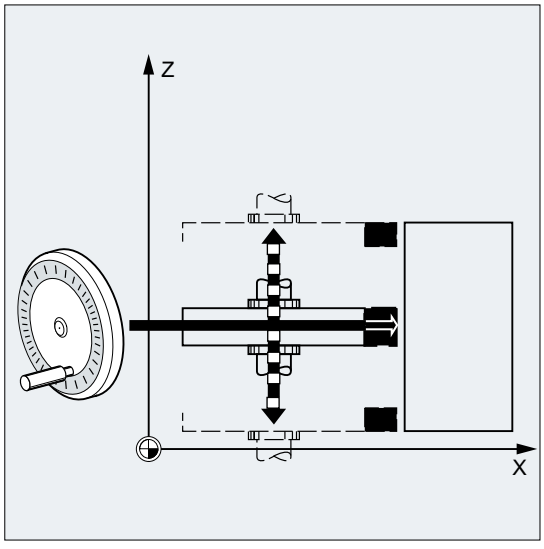
Объяснение

FD=< скорость > :	Подача по траектории и разрешение наложения скорости через маховичок. <скорость>: • Значение = 0: запрещено! • Значение ≠ 0: Скорость движения по траектории
FDA[<ось>]=<скорость> :	Осевая подача <скорость>: • Значение = 0: Задача перемещения через маховичок • Значение ≠ 0: Осевая скорость
<ось>:	Идентификатор позиционирующей оси

Примечание

FD и FDA действуют покадрово.

Пример



Заданное перемещение: Качающийся в направлении Z шлифовальный круг через маховичок движется в направлении X к детали.

При этом оператор может осуществлять ручную подачу до равномерного искрения. Посредством активации "Стирания остатка пути" осуществляется переход в следующий кадр ЧПУ и работа продолжается в режиме АВТО.

Дополнительная информация

Перемещение траекторных осей с наложением скорости (FD=<скорость>)
Для кадра программы обработки детали, в котором запрограммировано наложение скорости движения по траектории, должны быть выполнены следующие условия:

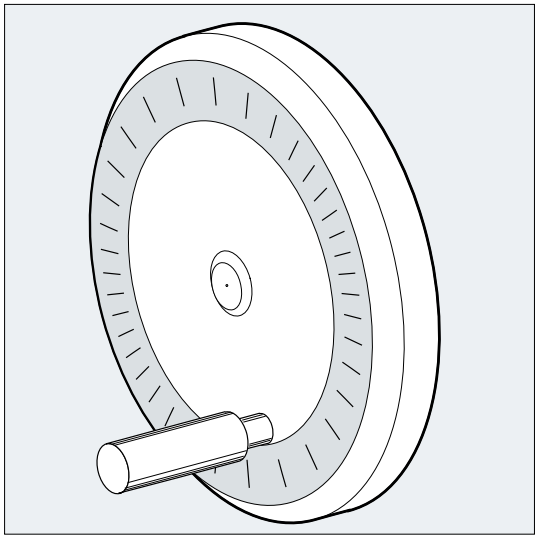
- Команда перемещения `G1`, `G2` или `G3` активна
- Точный останов `G60` активен
- Линейная подача `G94` активна

Процентовка подачи
Процентовка подачи действует только на запрограммированную скорость движения по траектории, но не на созданный маховичком компонент скорости (исключение: процентовка подачи = 0).

Пример:

Программный код	Описание
`N10 X… Y… F500`	; Подача по траектории = 500 мм/мин
`N20 X… Y… FD=700`	; Подача по траектории = 700 мм/мин и наложение скорости через маховичок.
	; В `N20` происходит разгон с 500 до 700 мм/мин. Через маховичок в зависимости от направления вращения можно изменять скорость движения по траектории между 0 и макс. значением (машинные данные).

Перемещение позиционирующих осей с заданным перемещением (FDA[<ось>]=0)
В кадре ЧПУ с запрограммированной `FDA[`<ось>`]=0` подача устанавливается на ноль, поэтому со стороны программы движение перемещения не выполняется. Запрограммированное движение перемещения к заданному конечному положению теперь управляется исключительно оператором посредством вращения маховичка.



Пример:

Программный код	Описание
...	
`N20 POS[V]=90 FDA[V]=0`	; Заданное конечное положение = 90 мм, осевая подача = 0 мм/мин и
	наложение перемещения через маховичок.
	; Скорость оси `V` в начале кадра = 0 мм/мин.
	; Перемещение и скорость задаются через импульсы маховичка

Направление движения, скорость движения:
Оси перемещаются в соответствии со знаком точно по заданному импульсами маховичка пути. В зависимости от направления вращения, возможно движение вперед и назад. Чем быстрее вращается маховичок, тем выше скорость перемещения.

Диапазон перемещения:
Диапазон перемещения ограничивается стартовой позицией и запрограммированной конечной точкой.

Перемещение позиционирующих осей с наложением скорости (FDA[<ось>]=<скорость>)

В кадре ЧПУ с запрограммированной `FDA[...]=...` подача ускоряется или замедляется от последнего запрограммированного значения `FA` до запрограммированного в `FDA`значения. Исходя из актуальной подачи `FDA` запрограммированное движение к позиции назначения с помощью вращения маховичка может ускоряться или замедляться до нуля. В качестве максимальной скорости действуют спараметрированные в машинных данных значения.

Пример:

Программный код	Описание
`N10 POS[V]=... FA[V]=100`	; Осевая подача = 100 мм/мин
`N20 POS[V]=100 FAD[V]=200`	; Осевое заданное конечное положение = 100, осевая подача = 200 мм/мин
	; и наложение скорости через маховичок.
	; В `N20` происходит разгон с 100 до 200 мм/мин. Через
	; маховичок возможно зависимое от направления вращения изменение скорости
	; между 0 и макс. значением (машинные данные)
	.

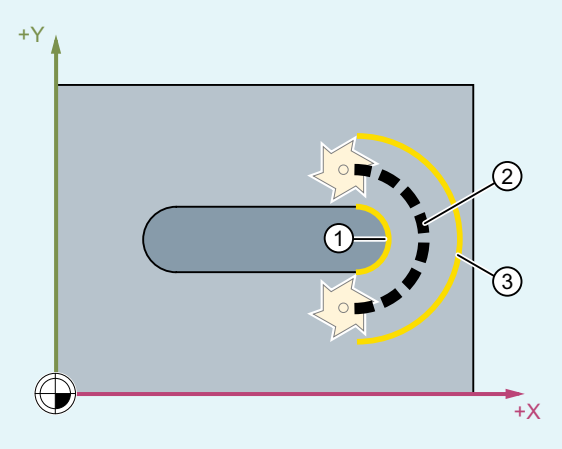
Диапазон перемещения:
Диапазон перемещения ограничивается стартовой позицией и запрограммированной конечной точкой.

3.7.9 Оптимизация подачи для изогнутых участков траектории (CFTCP, CFC, CFIN)

Запрограммированная подача относится при включенном режиме коррекции G41/G42 для радиуса фрезы, прежде всего, к траектории центра фрезы (ср. главу "Трансформации координат (фреймы) (Страница 314)").

При фрезеровании окружности (это же относится и к полиномиальной и сплайн-интерполяции) подача на кромке фрезы при определенных обстоятельствах изменятся до такой степени, что это отрицательно влияет на результат обработки.

Пример: Фрезерование небольшого наружного радиуса большим инструментом. Путь, который должна пройти наружная сторона фрезы, значительно больше пути вдоль контура.



Из-за этого работа на контуре осуществляется с очень маленькой подачей. Чтобы избежать таких эффектов, необходимо соответствующее управление подачей для изогнутых контуров.

Синтаксис

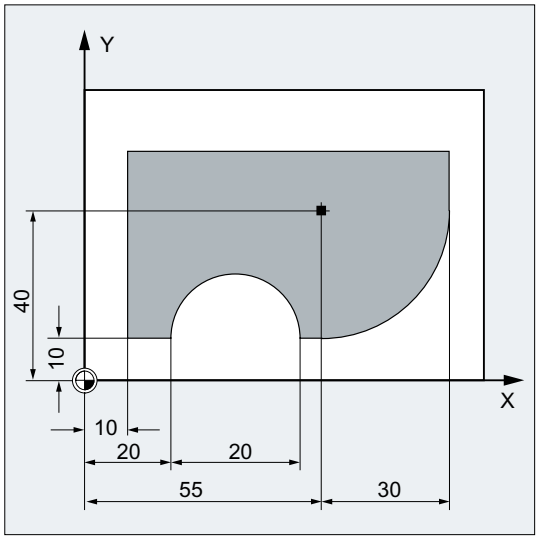
```
CFTCP
CFC
CFIN
```

Объяснение

`CFTCP`:	Постоянная подача на траектории центра фрезы СЧПУ поддерживает скорость подачи постоянной, процентовки подачи отключаются.
`CFC`:	Постоянная подача на контуре (резец инструмента) Эта функция является стандартной предварительной установкой.
`CFIN`:	Постоянная подача на резце инструмента только на изогнутых внутрь контурах, в иных случаях на траектории центра фрезы Скорость подачи уменьшается на внутренних радиусах.

Пример

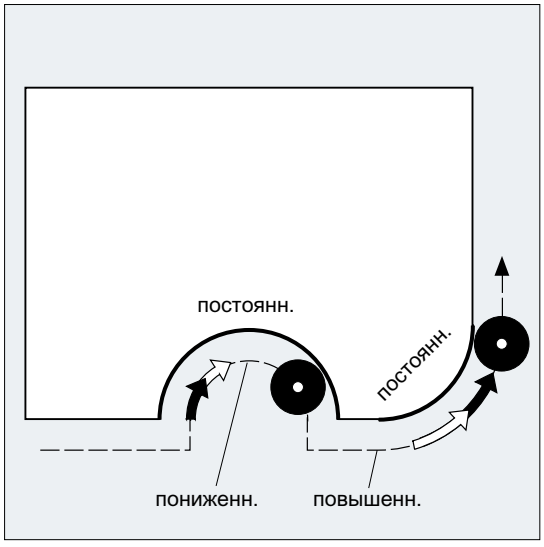
В этом примере сначала изготавливается контур с откорректированной CFC подачей. При чистовой обработке фрезеруемая поверхность дополнительно обрабатывается с CFIN. За счет этого удается предотвратить повреждение фрезеруемой поверхности на наружных радиусах из-за слишком высокой скорости подачи.



Программный код	Комментарий
N10 G17 G54 G64 T1 M6	
N20 S3000 M3 CFC F500 G41	
N30 G0 X-10	
N40 Y0 Z-10	; Подача на первую глубину резания
N50 KONTUR1	; Вызов подпрограммы
N40 CFIN Z-25	; Подача на вторую глубину резания
N50 KONTUR1	; Вызов подпрограммы
N60 Y120	
N70 X200 M30	

Дополнительная информация

Постоянная подача на контуре с CFC



Скорость подачи уменьшается на внутренних радиусах, увеличивается на наружных радиусах. Таким образом, скорость на резце инструмента и тем самым на контуре остается постоянной.

3.7.10 Несколько значений подачи в одном кадре (F, ST, SR, FMA, STA, SRA)

С помощью функции "Несколько значений подачи в одном кадре", в зависимости от внешних цифровых и/или аналоговых входов, возможна синхронная с движением активация различных значений подачи одного кадра ЧПУ, времени ожидания, а также отвода

Синтаксис

Движение по траектории:
```
F=... F7=... F6=... F5=... F4=... F3=... F2=... ST=... SR=...
```

Осевое движение:
```
FA[<Ax>]=... FMA[7,<Ax>]=... FMA[6,<Ax>]=... FMA[5,<Ax>]=...
FMA[4,<Ax>]=... FMA[3,<Ax>]=... FMA[2,<Ax>]=... STA[<Ax>]=...
SRA[<Ax>]=...
```

Значение

`F=...`:	По адресу `F` программируется подача по траектории, действующая до тех пор, пока нет входного сигнала.	
	Активность:	модально

F2=... до F7=... :	Дополнительно к подаче по траектории могут быть запрограммированы до 6-ти других подач в кадре. Числовое расширение указывает битовый номер входа, при изменении которого начинает действовать подача.	
	Активность:	покадрово
ST=... :	Время ожидания в сек (для шлифовальной технологии: время выхаживания)	
	Входной бит:	1
	Активность:	покадрово
SR=... :	Путь отвода Единица для пути отвода относится к актуальной действующей единице измерения (мм или дюймы).	
	Входной бит:	0
	Активность:	покадрово
FA[<Ax>]=... :	По адресу FA программируется осевая подача, действующая до тех пор, пока нет входного сигнала.	
	Активность:	модально
FMA[2,<Ax>]=... до FMA[7,<Ax>]=... :	В дополнение к осевой подаче FA, с помощью FMA в кадре может быть запрограммировано до 6 других подач на ось. Первый параметр указывает битовый номер входа, второй ось, для которой должна действовать подача.	
	Активность:	покадрово
STA[<Ax>]=...:	осевое время ожидания в сек (для шлифовальной технологии: время выхаживания)	
	Входной бит:	1
	Активность:	покадрово
SRA[<Ax>]=...:	осевой путь отвода	
	Входной бит:	0
	Активность:	покадрово
<Ax>:	ось, для которой должна действовать подача	

Примечание

Приоритет сигналов

Последовательность опроса сигналов действует от входного бита 0 (E0) в растущей последовательности. Тем самым движение отвода имеет высший, а подача F7 низший приоритет. Время ожидания и движение отвода завершают движения подачи, активированные с F2 до F7.

Сигнал с высшим приоритетом определяет актуальную подачу.

Примечание

Стирание остатка пути

Если активируется вход бит 1 для времени ожидания или путь отвода бит 0, то остаточный путь для траекторных осей или соответствующих отдельных осей стирается и запускается время ожидания или отвод.

Примечание

Путь отвода

Единица для пути отвода относится к актуальной действующей единице измерения (мм или дюймы).

Отвод всегда выполняется в направлении, противоположном текущему движению. С `SR`/`SRA` всегда программируется величина отвода. Знак не программируется.

Примечание

POS вместо POSA

Если для оси запрограммированы подачи, время ожидания или путь отвода на основе внешнего входа, то эта ось в этом кадре не может быть запрограммирована как ось POSA (позиционирующая ось за границы кадра).

Примечание

Опрос состояния

Состояние входа может быть запрошено и для синхронных команд различных осей.

Примечание

LookAhead

LookAhead действует и при нескольких подачах в одном кадре. Таким образом, актуальная подача может быть ограничена через LookAhead.

Примеры

Пример 1: Движение по траектории

Программный код	Комментарий
`G1 X48 F1000 F7=200 F6=50 F5=25 F4=5 ST=1.5 SR=0.5`	; Подача по траектории = 1000 ; Дополнительные значения подачи по траектории: ; 200 (входной бит 7) ; 50 (входной бит 6) ; 25 (входной бит 5) ; 5 (входной бит 4) ; Время ожидания 1,5 с ; Отвод 0,5 мм

Пример 2: Осевое движение

Программный код	Комментарий
`POS[A]=300 FA[A]=800 FMA[7,A]=720 FMA[6,A]=640 FMA[5,A]=560 STA[A]=1.5 SRA[A]=0.5`	; Подача для оси `A = 800` ; Дополнительные значения подачи для оси `A: 720` (входной бит 7) ; 640 (входной бит 6) ; 560 (входной бит 5) ; Осевое время ожидания: 1,5 с ; Осевой отвод: 0,5 мм

Пример 3: Несколько рабочих ходов в одном кадре

Программный код	Комментарий
`N20 T1 D1 F500 G0 X100`	Исходное положение
`N25 G1 X105 F=20 F7=5 F3=2.5 F2=0.5 ST=1.5 SR=0.5`	; Обычная подача с `F`, ; черновая обработка с `F7`, ; чистовая обработка с `F3`, ; отделочная обработка с `F2`, ; время ожидания 1,5 с, ; путь отвода 0,5 мм
...	

3.7.11 Покадровая подача (FB)

С помощью функции "Покадровая подача" для отдельного кадра может быть задана отдельная подача. После этого кадра снова активна действовавшая до этого модальная подача.

Синтаксис

`FB=`<значение>

Объяснение

`FB`:	Подача только для текущего кадра
<ЗНАЧЕНИЕ>:	Запрограммированное значение должно быть больше нуля. Интерпретация осуществляется в соответствии с активным типом подачи: • G94: подача в мм/мин или градусах/мин • G95: подача в мм/оборот или дюймов/оборот • G96: постоянная скорость резания

Примечание

Если в кадре не запрограммировано движение перемещения (например, кадр вычисления), то FB не действует.

Если не запрограммировано явной подачи для фаски/закругления, то значение FB действует и для имеющегося в этом кадре элемента контура "фаска/закругление".

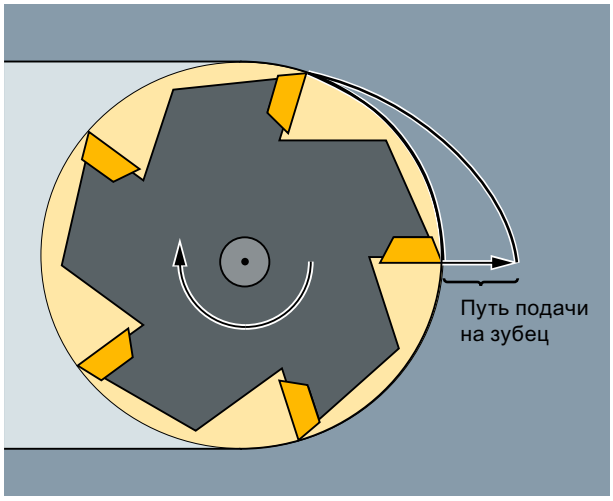
Интерполяции подачи FLIN, FCUB, ... возможны без ограничений.

Одновременное программирование FB и FD (перемещение вручную маховичком с наложением подачи) или F (модальная подача по траектории) **не** возможно.

Пример

Программный код	Комментарий
N10 G0 X0 Y0 G17 F100 G94	; Исходное положение
N20 G1 X10	; Подача 100 мм/мин
N30 X20 FB=80	; Подача 80 мм/мин
N40 X30	; Подача снова 100 мм/мин.
...	

3.7.12 Подача на зуб (G95 FZ)



Через параметр инструмента $TC_DPNT (число зубьев) активного блока данных коррекции инструмента СЧПУ из запрограммированной подачи на зуб вычисляет для каждого кадра перемещения эффективную окружную подачу:

F = FZ * $TC_DPNT

где:	F:	Окружная подача в мм/об или дюймов/об
	FZ:	Подача на зуб в мм/зуб или дюймов/зуб
	$TC_DPNT:	Системная переменная параметров инструмента: Число зубьев/об

Тип инструмента ($TC_DP1) активного инструмента не учитывается.

Запрограммированная подача на зуб не зависит от смены инструмента и выбора/отмены блока данных коррекции инструмента и сохраняется модально.

Изменение параметра инструмента $TC_DPNT активного резца вступает в силу при следующем выборе коррекции инструмента или следующей актуализации активных данных коррекции.

Смена инструмента и выбор/отмена блока данных коррекции инструмента приводят к новому вычислению действующей окружной подачи.

Примечание

Подача на зуб относится только в траектории, спец. для оси программирование невозможно.

Синтаксис

```
G95 FZ...
```

Объяснение

G95:	Тип подачи: Окружная подача в мм/об или дюймов/об (в зависимости от G700/G710) По G95 см. "Подача (G93, G94, G95, F, FGROUP, FL, FGREF) (Страница 110)"	
FZ:	Скорость подачи на зуб	
	Активация:	с G95
	Активность:	модально
	Единица измерения:	мм/зуб или дюймов/зуб (в зависимости от G700/G710)

ВНИМАНИЕ

Смена инструмента / смена мастер-шпинделя

Последующая смена инструмента или смена мастер-шпинделя должны быть учтены пользователем через соответствующее программирование, к примеру, повторное программирование FZ.

ВНИМАНИЕ
Не определенное использование инструмента Технологические требования, к примеру, попутное или встречное фрезерование, торцовое или окружное плоское фрезерование и т. п., а также геометрия траектории (прямая, окружность, ...) не учитываются автоматически. Поэтому необходимо учитывать данные факторы при программировании подачи на зуб.

Примечание

Переключение между G95 F... и G95 FZ...

При переключении между `G95 F...` (окружная подача) и `G95 FZ...` (подача на зуб) соответствующее не активное значение подачи удаляется.

Примечание

Производная подача с FPR

С помощью `FPR`, аналогично окружной подачи, и подача на зуб может быть производной от любой круговой оси или шпинделя (см. "Подача для позиционирующих осей/ шпинделей (FA, FPR, FPRAON, FPRAOF) (Страница 129)").

Примеры

Пример 1: Фреза с 5 зубьями ($TC_DPNT = 5)

Программный код	Комментарий
`N10 G0 X100 Y50`	
`N20 G1 G95 FZ=0.02`	; Подача на зуб 0,02 мм/зуб
`N30 T3 D1`	; Установить инструмент и активировать блок данных коррекции на инструмент.
`M40 M3 S200`	; Скорость шпинделя 200 об/мин
`N50 X20`	; Фрезерование с: `FZ = 0,02` мм/зуб эффективная окружная подача: `F = 0,02` мм/зуб * 5 зубьев/об = 0,1 мм/об или `F = 0,1` мм/об * 200 об/мин = 20 мм/мин
...	

Пример 2: Переключение между G95 F... и G95 FZ...

Программный код	Комментарий
`N10 G0 X100 Y50`	
`N20 G1 G95 F0.1`	; Окружная подача 0,1 мм/об
`N30 T1 M6`	
`N35 M3 S100 D1`	
`N40 X20`	

Программный код	Комментарий
N50 G0 X100 M5	
N60 M6 T3 D1	; Установить инструмент с 5 зубьями ($TC_DPNE = 5).
N70 X22 M3 S300	
N80 G1 X3 G95 FZ=0.02	; Переключение с G95 F… на G95 FZ…, подача на зуб с 0,02 мм/зуб активна.
…	

Пример 3: Отвод подачи на зуб от шпинделя (FBR)

Программный код	Комментарий
…	
N41 FPR(S4)	; Инструмент в шпинделе 4 (не мастер-шпиндель).
N51 G95 X51 FZ=0.5	; Подача на зуб 0,5 мм/зуб в зависимости от шпинделя S4.
…	

Пример 4: Последующая смена инструмента

Программный код	Комментарий
N10 G0 X50 Y5	
N20 G1 G95 FZ=0.03	; Подача на зуб 0,03 мм/зуб
N30 M6 T11 D1	; Установить инструмент с 7 зубьями ($TC_DPNE = 7).
N30 M3 S100	
N40 X30	; Эффективная окружная подача 0,21 мм/об
N50 G0 X100 M5	
N60 M6 T33 D1	; Установить инструмент с 5 зубьями ($TC_DPNE = 5).
N70 X22 M3 S300	
N80 G1 X3	; Подача на зуб модальная 0,03 мм/зуб, эффективная окружная подача 0,15 мм/об
…	

Пример 5: Смена мастер-шпинделя

Программный код	Комментарий
N10 SETMS(1)	; Шпиндель 1 это мастер-шпиндель.
N20 T3 D3 M6	; Инструмент 3 устанавливается в шпиндель 1.
N30 S400 M3	; Скорость S400 шпинделя 1 (и тем самым T3).
N40 G95 G1 FZ0.03	; Подача на зуб 0,03 мм/зуб
N50 X50	; Движение по траектории, эффективная подача зависит от: - подача на зуб FZ - скорость шпинделя 1 - число зубьев активного инструмента T3
N60 G0 X60	
…	
N100 SETMS(2)	; Шпиндель 2 становится мастер-шпинделем.
N110 T1 D1 M6	; Инструмент 1 устанавливается в шпиндель 2.
N120 S500 M3	; Скорость S500 шпинделя 2 (и тем самым T1).

Программный код	Комментарий
`N130 G95 G1 FZ0.03 X20`	; Движение по траектории, эффективная подача зависит от: - подача на зуб `FZ` - скорость шпинделя 2 - число зубьев активного инструмента `T1`

Примечание

После смены мастер-шпинделя (`N100`) необходимо установить инструмент (`N110`), который приводится шпинделем 2.

Дополнительная информация

Переключение между G93, G94 и G95

`FZ` может быть запрограммирована и при не активной `G95`, но не действует и удаляется при выборе `G95`, т. е. при переключении между `G93`, `G94` и `G95` аналогично `F` удаляется и значение `FZ`.

Повторный выбор G95

Повторный выбор `G95` при уже активной `G95` не действует (если при этом не запрограммировано переключения между `F` и `FZ`).

Действующая покадрово подача (FB)

Действующая покадрово подача `FB...` при активной `G95 FZ...` (модально) интерпретируется как подача на зуб.

Механизм SAVE

В подпрограммах с атрибутом `SAVE FZ`, аналогично `F`, записывается на значение перед стартом подпрограммы.

Несколько значений подачи в одном кадре

Функция "Несколько значений подачи в одном кадре" при подаче на зуб невозможна.

Синхронные действия

Подача `FZ` из синхронных действий невозможна.

Чтение скорости подачи на зуб и типа подачи по траектории

Возможно чтение скорости подачи на зуб и типа подачи по траектории через системные переменные:

- С остановкой предварительной обработки в программе обработки детали через системные переменные:

	$AC_FZ	Скорость подачи на зуб, которая действовала при подготовке актуального кадра главного хода.	
	$AC_F_TYPE	Тип подачи по траектории, который действовал при подготовке актуального кадра главного хода.	
		Величина:	**Значение:**
		0	мм/мин
		1	мм/об
		2	дюймов/мин
		3	дюймов/об
		11	мм/зуб
		33	дюймов/зуб

- Без остановки предварительной обработки в программе обработки детали через системные переменные:

	$P_FZ	Запрограммированная скорость подачи на зуб	
	$P_F_TYPE	Запрограммированный тип подачи по траектории	
		Величина:	**Значение:**
		0	мм/мин
		1	мм/об
		2	дюймов/мин
		3	дюймов/об
		11	мм/зуб
		33	дюймов/зуб

Примечание

Если G95 не активна, то переменные $P_FZ и $AC_FZ всегда дают значение ноль.

3.8 Геометрические установки

3.8.1 Устанавливаемое смещение нулевой точки (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153)

С помощью команд G54–G57 и G505–G599 установленные через интерфейс пользователя смещения нулевой точки активируются для смещения системы координат детали относительно базовой системы координат.

синтаксис

Включение:
G54
...
G57
G505
...
G599

Выключение или подавление:

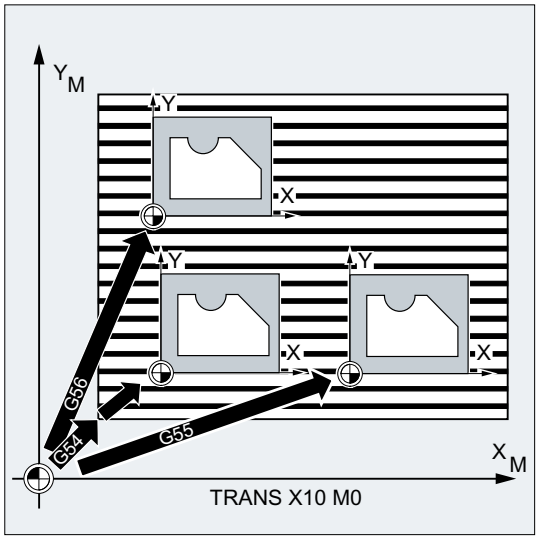
G500 / G53 / G153 / SUPA

Объяснение

G54 ... G57:	Вызов 1-го до 4-го настраиваемого смещения нулевой точки (WO)	
G505 ... G599:	Вызов 5-го до 99-го настраиваемого WO	
G500:	Отключение текущего настраиваемого WO	
	G500=Нулевой фрейм: (стандартная настройка; не содержит смещения, вращения, отражения или масштабирования)	Отключение настраиваемого WO до следующего вызова, активация общего базового фрейма ($P_ACTBFRAME).
	G500 не равна 0:	Активация первого настраиваемого смещения нулевой точки ($P_UIFR[0]) и активация общего базового фрейма ($P_ACTBFRAME) или активируется возможно измененный базовый фрейм.
G53:	G53 подавляет покадрово настраиваемое WO и программируемое WO.	
G153:	G153 действует как G53 и дополнительно подавляет общий базовый фрейм.	
SUPA:	SUPA действует как G153 и дополнительно подавляет: • смещения маховичком (DRF) • наложенные движения • внешнее WO • смещение PRESET	

Пример

Три детали, расположенные на одной палете согласно значениям смещения нулевой точки G54–G56, должны быть обработаны последовательно. Последовательность обработки запрограммирована в подпрограмме L47.



Программный код	Комментарий
N10 G0 G90 X10 Y10 F500 T1	; подвод
N20 G54 S1000 M3	; Вызов первого WO, шпиндель вправо
N30 L47	; Выполнение программы как подпрограммы
N40 G55 G0 Z200	; Вызов второго WO, Z через препятствие
N50 L47	; Выполнение программы как подпрограммы
N60 G56	; Вызов третьего WO
N70 L47	; Выполнение программы как подпрограммы
N80 G53 X200 Y300 M30	; Подавление смещения нулевой точки, конец программы

3.8.2 Устанавливаемое смещение нулевой точки (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153): Дополнительная информация

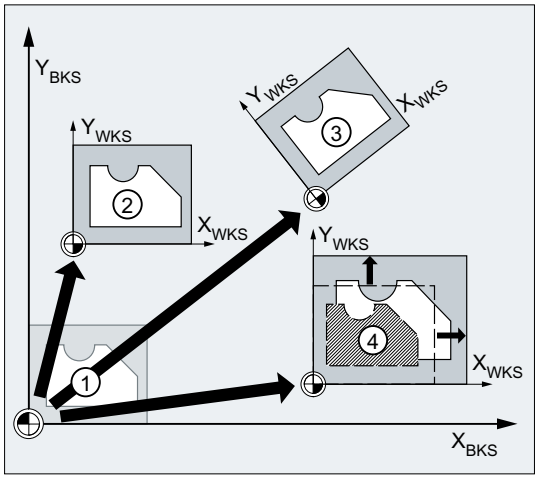
Дополнительная информация

Настраиваемое смещение нулевой точки

В принципе, настраиваемое смещение нулевой точки – это настраиваемый фрейм (Страница 314). Поэтому при настраиваемом смещении нулевой точки доступны следующие компоненты или значения фрейма:

- Смещение
- Вращение

- Масштабирование
- Масштаб



① Исходное положение в BKS
② Смещение
③ Смещение + вращение
④ Смещение + масштабирование

Ввод значений фрейма для настраиваемых смещений нулевой точки осуществляется через интерфейс пользователя:

SINUMERIK Operate: Область управления "Параметры" > "Смещения нулевой точки" > "Подробно"

Параметрируемое число настраиваемых фреймов (G505–G599)

Число настраиваемых пользователем смещений нулевой точки (G505–G599) для отдельного канала можно устанавливать через:

MD28080 $MC_MM_NUM_USER_FRAMES = <число>

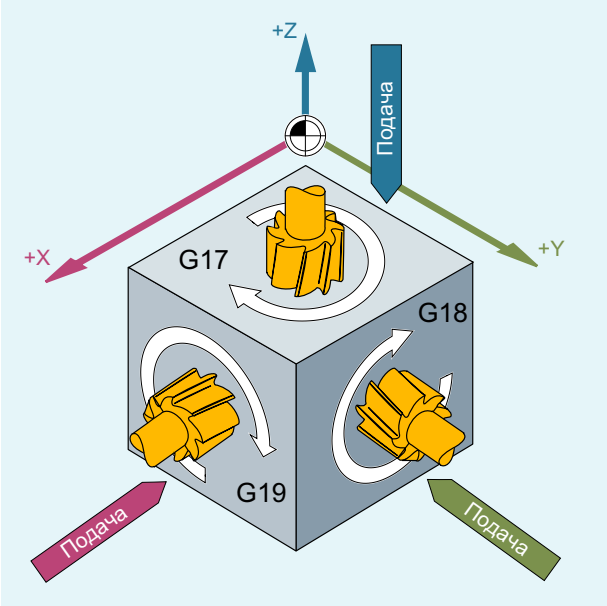
См. также

Программируемое смещение нулевой точки (G58, G59) (Страница 323)

3.8.3 Выбор рабочей плоскости (G17/G18/G19)

Посредством указания рабочей плоскости, в которой должен быть изготовлен желаемый контур, одновременно устанавливаются следующие функции:

- Плоскость для коррекции радиуса инструмента.
- Направление подачи для коррекции длин инструмента в зависимости от типа инструмента.
- Плоскость для круговой интерполяции.



Синтаксис

```
G17/G18/G19 ...
```

Значение

G17:	Рабочая плоскость X/Y Направление подачи Z выбор плоскости 1-й – 2-й гео-оси
G18:	Рабочая плоскость Z/X Направление подачи Y выбор плоскости 3-й – 1-й гео-оси
G19:	Рабочая плоскость Y/Z Направление подачи X выбор плоскости 2-й – 3-й гео-оси

Примечание

В первичной установке для фрезерования предустановленна G17 (плоскость X/Y), а для токарной обработки - G18 (плоскость Z/X).

При вызове коррекции траектории инструмента G41/G42 (см. главу "Коррекции радиуса инструмента (Страница 260)") рабочая плоскость должна быть указана, чтобы СЧПУ могла исправить длину и радиус инструмента.

Пример

"Классический" метод при фрезеровании:

1. Определить рабочую плоскость (G17 основная установка для фрезерования).
2. Вызвать тип инструмента (T) и значения коррекции инструмента (D).

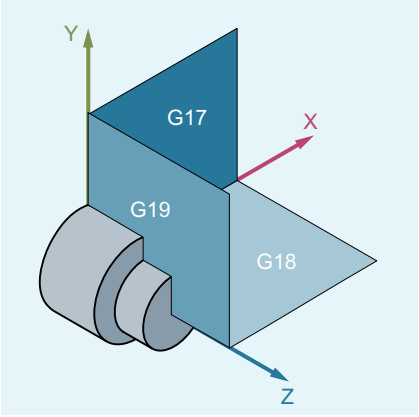
3. Включить коррекцию траектории (G41).
4. Запрограммировать движения перемещения.

Программный код	Комментарий
N10 G17 T5 D8	; Вызов рабочей плоскости X/Y, вызов инструмента. Коррекция длин осуществляется в направлении Z.
N20 G1 G41 X10 Y30 Z-5 F500	; Коррекция радиуса осуществляется в плоскости X/Y.
N30 G2 X22.5 Y40 I50 J40	; Круговая интерполяция/коррекция радиуса инструмента в плоскости X/Y.

Дополнительная информация

Общая информация

Рекомендуется установить рабочую плоскость G17 до G19 уже в начале программы. В первичной настройке для токарной обработки G18 предустановлена плоскость Z/X.

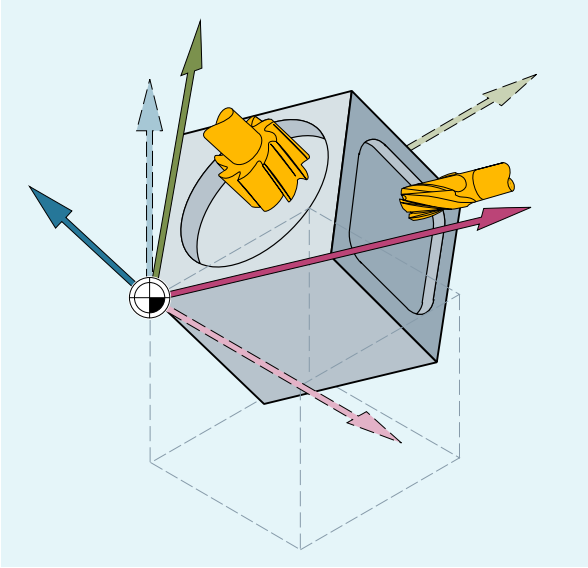


СЧПУ для вычисления направление вращения необходимо указание рабочей плоскости. Дополнительная информация представлена в главе "Круговая интерполяция (Страница 197)".

Обработка в наклонных плоскостях

Посредством вращения системы координат с помощью ROT (Страница 325) оси координат накладываются на наклонную поверхность. Рабочие плоскости также соответственно поворачиваются.

Коррекция длин инструмента в наклонных плоскостях



Коррекция длин инструмента всегда вычисляется относительно зафиксированной в пространстве, не повернутой рабочей плоскости.

Примечание

С помощью функций для "Коррекции длин инструмента для ориентируемых инструментов" можно вычислить компоненты длин инструмента в соответствии с повернутыми рабочими плоскостями.

Выбор плоскости коррекции осуществляется с помощью CUT2D, CUT2DF. Дополнительная информация по этой теме, а также по описанию этой возможности вычисления представлена в главе "Коррекции радиуса инструмента (Страница 260)".

Для пространственного определения рабочей плоскости СЧПУ предлагает очень удобные возможности для трансформаций координат. Дополнительная информация представлена в главе "Трансформации координат (фреймы) (Страница 314)".

3.8.4 Указание размеров

Основой большинства программ ЧПУ является чертеж детали с правильным указанием размеров.

Размеры могут быть указаны:

- как абсолютный или составной размер
- в миллиметрах или дюймах
- в радиусе или диаметре (для токарной обработки)

Для прямой передачи данных из габаритного чертежа (без пересчета) в программу ЧПУ, пользователю для различных возможностей по указанию размеров доступны специфические команды программирования.

3.8.4.1 Указание абсолютного размера (G90, AC)

При указании абсолютных размеров данные позиций всегда относятся к нулевой точке актуальной действующей системы координат, т. е. программируется абсолютная позиция, к которой должен двигаться инструмент.

Действующее модально указание абсолютного размера

Действующее модально указание абсолютного размера активируется с помощью команды `G90`. Оно всегда действует для всех осей, которые программируются в последующих кадрах ЧПУ.

Действующее покадрово указание абсолютного размера

При предустановленном составном размере (`G91`) с помощью команды `AC` для отдельных осей возможно покадровое указание абсолютного размера.

Примечание

Действующее покадрово указание абсолютного размера(`AC`) возможно и для позиционирований шпинделя (`SPOS`, `SPOSA`) и параметров интерполяции (`I`, `J`, `K`).

Синтаксис

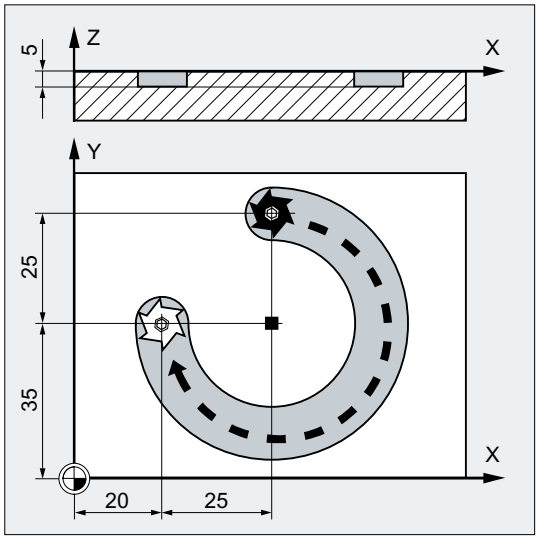
```
G90
<ось>=AC(<значение>)
```

Значение

`G90`:	Команда для активации действующего модально указания абсолютного размера
`AC`:	Команда для активации действующего покадрово указания абсолютного размера
<ось>:	Идентификатор перемещаемой оси
<значение>:	Заданная позиция перемещаемой оси в абсолютном размере

Примеры

Пример 1: Фрезерование

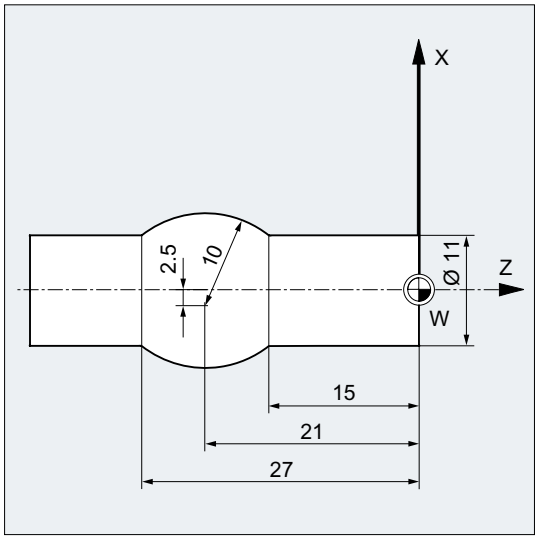


Программный код	Комментарий
`N10 G90 G0 X45 Y60 Z2 T1 S2000 M3`	; Ввод абсолютного размера, ускоренным ходом на позицию `XYZ`, выбор инструмента, шпиндель включен с правым направлением вращения.
`N20 G1 Z-5 F500`	; Линейная интерполяция, подача инструмента.
`N30 G2 X20 Y35 I=AC(45) J=AC(35)`	; Круговая интерполяция по часовой стрелке, конечная точка и центр окружности в абсолютном размере.
`N40 G0 Z2`	; Вывод.
`N50 M30`	; Конец кадра.

Примечание

По вводу координат центра окружности I и J см. главу "Круговая интерполяция".

Пример 2: Токарная обработка



Программный код	Комментарий
`N5 T1 D1 S2000 M3`	; Установка инструмента `T1`, шпиндель включен с правым направлением вращения.
`N10 G0 G90 X11 Z1`	; Ввод абсолютного размера, ускоренным ходом на позицию `XZ`.
`N20 G1 Z-15 F0.2`	; Линейная интерполяция, подача инструмента.
`N30 G3 X11 Z-27 I=AC(-5) K=AC(-21)`	; Круговая интерполяция против часовой стрелки, конечная точка и центр окружности в абсолютном размере.
`N40 G1 Z-40`	; Вывод.
`N50 M30`	; Конец кадра.

Примечание

По вводу координат центра окружности I и J см. главу "Круговая интерполяция".

См. также

Указание абсолютного и составного размера при токарной обработке и фрезеровании (G90/G91) (Страница 163)

3.8.4.2 Указание составного размера (G91, IC)

При указании составного размера данные позиции относятся к последней точке, к которой был осуществлен подвод, т. е. программирование в составном размере описывает, на сколько должен быть перемещен инструмент.

Действующее модально указание составного размера

Действующее модально указание составного размера активируется с помощью команды `G91`. Оно всегда действует для всех осей, которые программируются в последующих кадрах ЧПУ.

Действующее покадрово указание составного размера

При предустановленном абсолютном размере (G90) с помощью команды IC для отдельных осей возможно покадровое указание составного размера.

Примечание

Действующее покадрово указание составного размера(IC) возможно и для позиционирований шпинделя (SPOS, SPOSA) и параметров интерполяции (I, J, K).

Синтаксис

```
G91
<ось>=IC(<значение>)
```

Значение

G91:	Команда для активации действующего модально указания составного размера
IC:	Команда для активации действующего покадрово указания составного размера
<ось>:	Идентификатор перемещаемой оси
<значение>:	Заданная позиция перемещаемой оси в составном размере

Расширение G91

Для определенных приложений, как то, к примеру, "режим касания", требуется пройти в составном размере только запрограммированный путь. Активное смещение нулевой точки или коррекция длин инструмента не проходится.

Такое поведение может быть установлено отдельно для активного смещения нулевой точки и коррекции длин инструмента через следующие установочные данные:

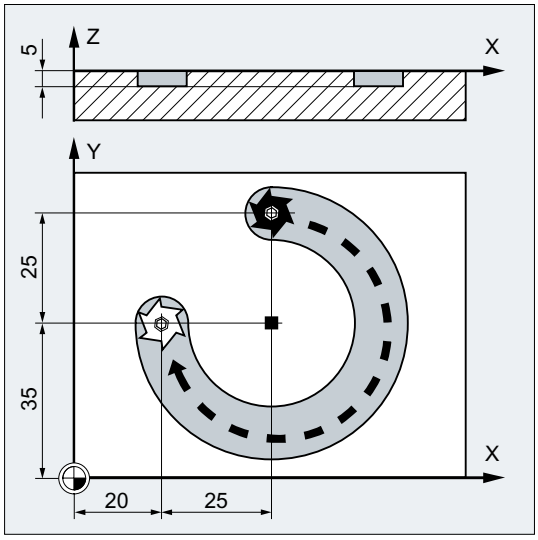
SD42440 $SC_FRAME_OFFSET_INCR_PROG (смещения нулевой точки во фреймах)

SD42442 $SC_TOOL_OFFSET_INCR_PROG (коррекции длин инструмента)

Значение	Значение
0	При инкрементальном программировании (указание составного размера) оси активное смещение нулевой точки или коррекция длин инструмента **не** выводится.
1	При инкрементальном программировании (указание составного размера) оси активное смещение нулевой точки или коррекция длин инструмента выводится.

Примеры

Пример 1: Фрезерная обработка

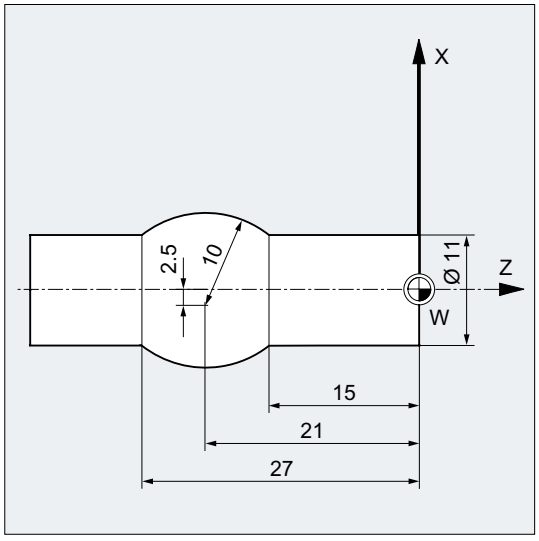


Программный код	Комментарий
`N10 G90 G0 X45 Y60 Z2 T1 S2000 M3`	; Ввод абсолютного размера, ускоренным ходом на позицию `XYZ`, выбор инструмента, шпиндель включен с правым направлением вращения.
`N20 G1 Z-5 F500`	; Линейная интерполяция, подача инструмента.
`N30 G2 X20 Y35 I0 J-25`	; Круговая интерполяция по часовой стрелке, конечная точка окружности в абсолютном размере, центр окружности в составном размере.
`N40 G0 Z2`	; Вывод.
`N50 M30`	; Конец кадра.

Примечание

По вводу координат центра окружности I и J см. главу "Круговая интерполяция".

Пример 2: Токарная обработка



Программный код	Комментарий
`N5 T1 D1 S2000 M3`	; Установка инструмента `T1`, шпиндель включен с правым направлением вращения.
`N10 G0 G90 X11 Z1`	; Ввод абсолютного размера, ускоренным ходом на позицию `XZ`.
`N20 G1 Z-15 F0.2`	; Линейная интерполяция, подача инструмента.
`N30 G3 X11 Z-27 I-8 K-6`	; Круговая интерполяция против часовой стрелки, конечная точка окружности в абсолютном размере, центр окружности в составном размере.
`N40 G1 Z-40`	; Вывод.
`N50 M30`	; Конец кадра.

Примечание

По вводу координат центра окружности I и J см. главу "Круговая интерполяция".

Пример 3: Указание составного размера без вывода активного смещения нулевой точки

Установки:

- G54 содержит смещение в X на 25
- SD42440 $SC_FRAME_OFFSET_INCR_PROG = 0

Программный код	Комментарий
`N10 G90 G0 G54 X100`	
`N20 G1 G91 X10`	; Указание составного размера активно, движение в `X` на 10мм (смещение нулевой точки не проходится).

Программный код	Комментарий
`N30 G90 X50`	; Указание абсолютного размера активно, движение на позицию `X75` (смещение нулевой точки проходится).

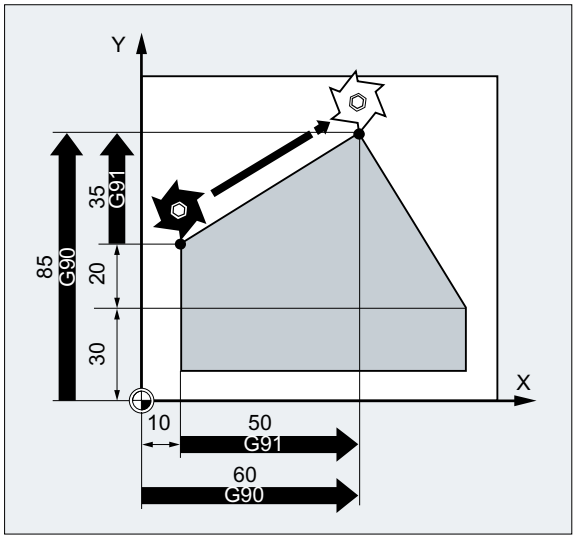
См. также

Указание абсолютного и составного размера при токарной обработке и фрезеровании (G90/G91) (Страница 163)

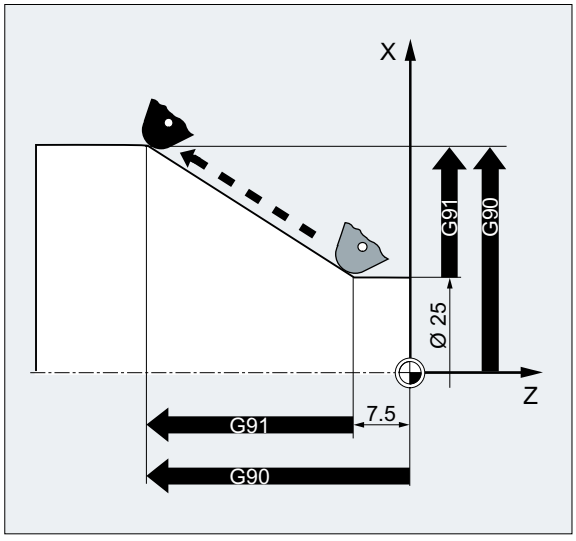
3.8.4.3 Указание абсолютного и составного размера при токарной обработке и фрезеровании (G90/G91)

Оба рисунка ниже поясняют программирование с указанием абсолютного размера (`G90`) или указанием составного размера (`G91`) на примере токарной и фрезерной технологии.

Фрезерование:



Токарная обработка:



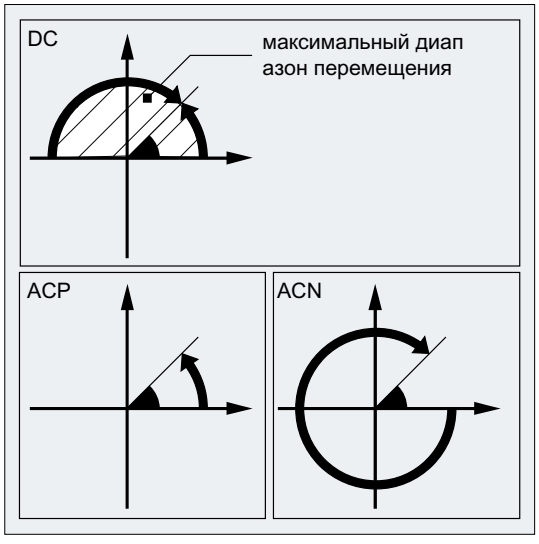
Примечание

На обычных токарных станках принято рассматривать инкрементальные кадры перемещения в поперечной оси как значения радиуса, в то время, как данные диаметра действуют для исходных размеров. Это переключение для `G90` осуществляется с помощью команд `DIAMON`, `DIAMOF` или `DIAM90`.

3.8.4.4 Указание абсолютного размера для круговых осей (DC, ACP, ACN)

Для позиционирования круговых осей в абсолютном размере имеются действующие покадрово и не зависимые от G90/G91 команды DC, ACP и ACN.

DC, ACP и ACN отличаются друг от друга базовой стратегией подвода:



Синтаксис

<круговая ось>=DC(<значение>)
<круговая ось>=ACP(<значение>)
<круговая ось>=ACN(<значение>)

Объяснение

<круговая ось>:	Идентификатор круговой оси, которая должна быть перемещена (к примеру, A, B или C)	
DC:	Команда для **прямого** подвода к позиции Круговая ось движется к запрограммированной позиции по прямому, кратчайшему пути. Круговая ось перемещается в макс. диапазоне в 180°.	
ACP:	Команда для подвода к позиции в **положительном** направлении Круговая ось движется к запрограммированной позиции в положительном направлении вращения оси (против часовой стрелки).	
ACN:	Команда для подвода к позиции в **отрицательном** направлении Круговая ось движется к запрограммированной позиции в отрицательном направлении вращения оси (по часовой стрелке).	
<значение>:	Позиция подвода круговой оси в абсолютном размере	
	Диапазон значений:	0 - 360 градусов

Примечание

Положительное направление вращения (по часовой или против часовой стрелки) устанавливается в машинных данных.

Примечание

Для позиционирования с указанием направления (ACP, ACN) в машинных данных должен быть установлен диапазон перемещения между 0 ° и 360 ° (характеристика модуло). Для перемещения круговых осей модуло в одном кадре более чем на 360 ° необходимо запрограммировать G91 или IC.

Примечание

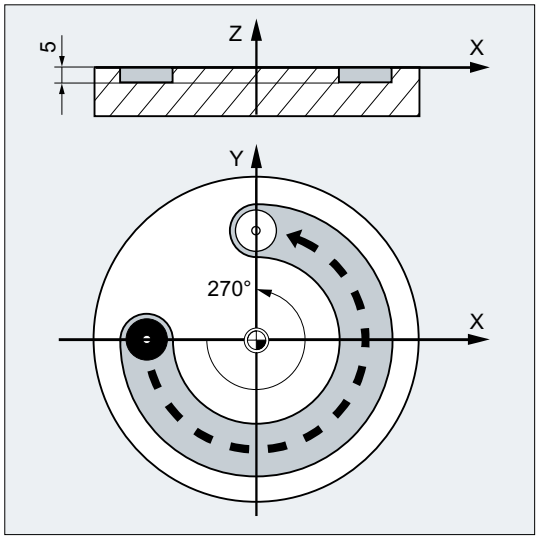
Команды DC, ACP и ACN могут использоваться и для позиционирования шпинделя (SPOS, SPOSA) из состояния покоя.

Пример: SPOS=DC(45)

Пример

Фрезерная обработка на круглом столе

Инструмент неподвижен, стол поворачивается на 270° **по часовой стрелке**. При этом получается кольцевая канавка.



Программный код	Комментарий
`N10 SPOS=0`	; Шпиндель в режиме управления по положению.
`N20 G90 G0 X-20 Y0 Z2 T1`	; Указание абсолютного размера, подача инструмента `T1` ускоренным ходом.
`N30 G1 Z-5 F500`	; Движение инструмента вниз с подачей.
`N40 C=ACP(270)`	; Стол поворачивается на 270 градусов по часовой стрелке (положительно), инструмент фрезерует кольцевую канавку.
`N50 G0 Z2 M30`	; Отвод, конец программы.

3.8.4.5 Дюймовая/метрическая система единиц (G70/G71, G700/G710)

В программе обработки детали команды группы G 13 (дюймовая/метрическая система единиц) обеспечивают переключение между дюймовой и метрической системами единиц.

Активация

Команды G700 и G710 доступны, только если включена расширенная функциональная возможность системы единиц измерения (MD10260 $MN_CONVERT_SCALING_SYSTEM = 1).

Синтаксис

```
G70
G71
G700
G710
```

Объяснение

G70:	Включение дюймовой системы единиц **Содержащие длины геометрические** данные считываются и записываются в **дюймовой системе единиц**. **Содержащие длины технологические** данные (например, подачи, коррекции инструмента, настраиваемые смещения нулевой точки, машинные данные и системные переменные) считываются и записываются в **спараметрированной основной системе**.	
	G-группа:	13
	Исходное положение:	Настройка через MD20150 $MC_GCODE_RESET_VALUES
	Активность:	Модально
G71:	Включение метрической системы единиц **Содержащие длины геометрические** данные считываются и записываются в **метрической системе единиц**. **Содержащие длины технологические** данные (например, подачи, коррекции инструмента, настраиваемые смещения нулевой точки, машинные данные и системные переменные) считываются и записываются в **спараметрированной основной системе**.	
	G-группа:	13
	Исходное положение:	Настройка через MD20150 $MC_GCODE_RESET_VALUES
	Активность:	Модально
G700:	Включение дюймовой системы единиц Все содержащие длины геометрические и технологические данные считываются и записываются в дюймовой системе единиц.	
	G-группа:	13
	Исходное положение:	Настройка через MD20150 $MC_GCODE_RESET_VALUES
	Активность:	Модально
G710:	Включение метрической системы единиц Все содержащие длины геометрические и технологические данные считываются и записываются в метрической системе единиц.	
	G-группа:	13
	Исходное положение:	Настройка через MD20150 $MC_GCODE_RESET_VALUES
	Активность:	Модально

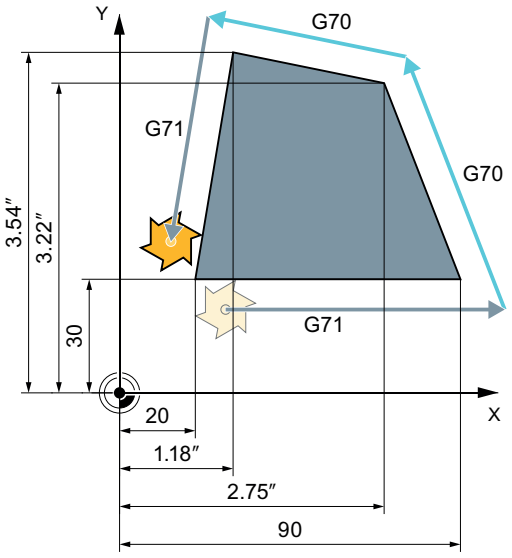
ВНИМАНИЕ

Спец. для оси данные круговых осей

Спец. для оси данные круговых осей всегда считываются и записываются в спараметрированной основной системе.

Пример

Основная система метрическая (MD10240 $MN_SCALING_SYSTEM_IS_METRIC = 1). Тем не менее, на чертеже детали размеры указаны в дюймах. Поэтому в программе обработки детали выполняется переключение на дюймовую систему единиц. После указания размеров в дюймах выполняется обратное переключение на метрическую систему единицу.



Программный код	Комментарий
`N10 G0 G90 X20 Y30 Z2 S2000 M3 T1`	`; X=20` мм, `Y=30` мм, `Z=2` мм, `F=`ускоренный ход мм/мин
`N20 G1 Z-5 F500`	`; Z=-5` мм, `F=500` мм/мин
`N30 X90`	`; X=90` мм
`N40` **`G70`** `X2.75 Y3.22`	; Запрограммированная система единиц: дюймовая
	`; X=2,75` дюйма, `Y=3,22` дюйма, `F=500` мм/мин
`N50 X1.18 Y3.54`	`; X=1,18` дюйма, `Y=3,54` дюйма, `F=500` мм/мин
`N60` **`G71`** `X20 Y30`	; Запрограммированная система единиц: метрическая
	`; X=20` мм, `Y=30` мм, `F=500` мм/мин
`N70 G0 Z2`	`; Z=2` мм, `F=`ускоренный ход мм/мин
`N80 M30`	; конец программы

Дополнительная информация

Чтение и запись данных при G70/G71 и G700/G710

Область данных	G70 / G71		G700 / G710	
	Чтение	Запись	Чтение	Запись
Индикации, места после запятой (WCS)	P	P	P	P
Индикации, места после запятой (MCS)	g	g	g	g

Область данных	G70 / G71		G700 / G710	
	Чтение	Запись	Чтение	Запись
Подачи	g	g	P	P
Данные позиций X, Y, Z	P	P	P	P
Параметры интерполяции I, J, K	P	P	P	P
Радиус окружности (CR)	P	P	P	P
Полярный радиус (RP)	P	P	P	P
Шаг резьбы	P	P	P	P
Программируемый ФРЕЙМ	P	P	P	P
Настраиваемые ФРЕЙМЫ	g	g	P	P
Базовые фреймы	g	g	P	P
Внешние смещения нулевой точки	g	g	P	P
Осевое Preset-смещение	g	g	P	P
Ограничения рабочего поля (G25/ G26)	g	g	P	P
Защищенные области	P	P	P	P
Коррекции инструмента	g	g	P	P
Содержащие длины машинные данные	g	g	P	P
Содержащие длины установочные данные	g	g	P	P
Содержащие длины системные переменные	g	g	P	P
GUD	g	g	g	g
LUD	g	g	g	g
PUD	g	g	g	g
R-параметры	g	g	g	g
Циклы Siemens	P	P	P	P
Нормирование инкремента Jog/маховичок	g	g	g	g
P: запись/чтение осуществляются в запрограммированной системе единиц G: запись/чтение осуществляются в спараметрированной основной системе				

Синхронные действия

Примечание

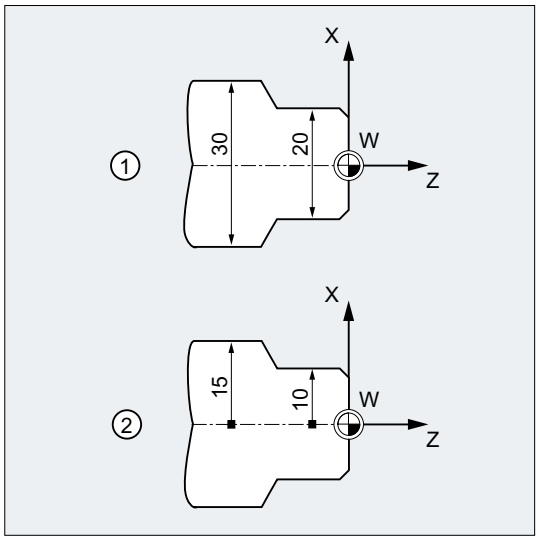
Чтение данных позиций в синхронных действиях

Без явного программирования системы единиц в синхронном действии (условная часть и/или операционная часть или технологическая функция) **содержащие длины данные позиций** всегда считываются в синхронном действии в спараметрированной **основной системе**.

Дополнительная информация: Описание функций "Синхронные действия"

3.8.4.6 Специфическое для канала программирование диаметра/радиуса (DIAMON, DIAM90, DIAMOF, DIAMCYCOF)

При токарной обработке размеры **для поперечной оси** могут быть указаны в диаметре (①) или в радиусе (①):



Для того, чтобы можно было передать эти указания размеров напрямую без пересчета из технического чертежа в программу ЧПУ, через действующие модально команды `DIAMON`, `DIAM90`, `DIAMOF` и `DIAMCYCOF` включается спец. для канала программирование диаметра или радиуса.

Примечание

Спец. для канала программирование диаметра/радиуса относится к определенной через MD20100 $MC_DIAMETER_AX_DEF а качестве поперечной оси гео-оси (→ см. Указания изготовителя станка!).

Через MD20100 на канал может быть определена только одна поперечная ось.

Синтаксис

```
DIAMON
DIAM90
DIAMOF
```

Объяснение

DIAMON:	Команда для включения **независимого** спец. для канала программирования диаметра Действие DIAMON не зависит от запрограммированного режима указания размеров (указание абсолютных размеров G90 или указание составных размеров G91):	
	• при G90:	Указание размеров в диаметре
	• при G91:	Указание размеров в диаметре
DIAM90:	Команда для включения **зависимого** спец. для канала программирования диаметра Действие DIAM90 зависит от запрограммированного режима указания размеров:	
	• при G90:	Указание размеров в диаметре
	• при G91:	Указание размеров в радиусе
DIAMOF:	Команда для выключения спец. для канала программирования диаметра При отключении программирования диаметра начинает действовать спец. для канала программирование радиуса. Действие DIAMOF не зависит от запрограммированного режима указания размеров:	
	• при G90:	Указание размеров в радиусе
	• при G91:	Указание размеров в радиусе
DIAMCYCOF:	Команда для выключения спец. для канала программирования диаметра при обработке цикла Тем самым расчеты в цикле всегда могут выполняться в радиусе. Для индикации позиции и базовой индикации кадра остается активной последняя активная G-команда этой группы.	

Примечание

При DIAMON или DIAM90 фактические значения поперечной оси всегда индицируются как диаметр. Это же относится и к считыванию фактических значений в системе координат детали при MEAS, MEAW, $P_EP[x] и $AA_IW[x].

Пример

Программный код	Комментарий
N10 G0 X0 Z0	; Подвод в стартовую точку.
N20 DIAMOF	; Программирование диаметра выкл.
N30 G1 X30 S2000 M03 F0.7	; Ось X = поперечная ось, программирование радиуса активно, движение на позицию радиуса X30.
N40 DIAMON	; Для поперечной оси активно программирование диаметра.
N50 G1 X70 Z-20	; Перемещение на позицию диаметра X70 и Z-20.
N60 Z-30	
N70 DIAM90	; Программирование диаметра для исходного размера и программирование радиуса для составного размера.
N80 G91 X10 Z-20	; Составной размер активен.
N90 G90 X10	; Исходный размер активен.

Программный код	Комментарий
N100 M30	; Конец программы.

Дополнительная информация

Значения диаметра (DIAMON/DIAM90)

Значения диаметра действуют для следующих данных:

- индикация фактического значения поперечной оси в системе координат детали
- Режим JOG: Инкременты для размера шага и перемещения вручную маховичком
- Программирование конечных позиций:
  параметры интерполяции I, J, K при G2/G3, если они запрограммированы с AC абсолютно.
  При инкрементальном программировании (IC) von I, J, K всегда учитывается радиус.
- Чтение фактических значений в системе координат детали при:
  MEAS, MEAW, $P_EP[X], $AA_IW[X]

3.8.4.7 Специфическое для оси программирование диаметра/радиуса (DIAMONA, DIAM90A, DIAMOFA, DIACYCOFA, DIAMCHANA, DIAMCHAN, DAC, DIC, RAC, RIC)

В дополнение к спец. для канала программированию диаметра спец. для оси программирование диаметра обеспечивает для одной или нескольких осей действующее модально или покадрово указание размеров с индикацию в диаметре.

Примечание

Спец. для оси программирование диаметра возможно только для осей, допущенных через MD30460 $MA_BASE_FUNCTION_MASK в качестве дополнительных поперечных осей для спец. для оси программирования диаметра (→ см. Указания изготовителя станка!).

Синтаксис

Действующее модально спец. для оси программирование диаметра для нескольких поперечных осей в канале:
DIAMONA[<ось>]
DIAM90A[<ось>]
DIAMOFA[<ось>]
DIACYCOFA[<ось>]

Применение спец. для канала программирования диаметра/радиуса:
DIAMCHANA[<ось>]
DIAMCHAN

Действующее покадрово спец. для оси программирования диаметра/радиуса:
<ось>=DAC(<значение>)
<ось>=DIC(<значение>)
<ось>=RAC(<значение>)
<ось>=RIC(<значение>)

Объяснение

Действующее модально спец. для оси программирование диаметра		
DIAMONA:	Команда для включения **независимого** спец. для оси программирования диаметра Действие DIAMONA не зависит от запрограммированного режима указания размера (G90/G91 или AC/IC):	
	• при G90, AC:	Указание размеров в диаметре
	• при G91, IC:	Указание размеров в диаметре
DIAM90A:	Команда для включения **зависимого** спец. для оси программирования диаметра Действие DIAM90A зависит от запрограммированного режима указания размеров:	
	• при G90, AC:	Указание размеров в диаметре
	• при G91, IC:	Указание размеров в радиусе
DIAMOFA:	Команда для выключения спец. для оси программирования диаметра При отключении программирования диаметра начинает действовать спец. для оси программирование радиуса. Действие DIAMOFA не зависит от запрограммированного режима указания размеров:	
	• при G90, AC:	Указание размеров в радиусе
	• при G91, IC:	Указание размеров в радиусе
DIACYCOFA:	Команда для выключения спец. для оси программирования диаметра при обработке цикла Тем самым расчеты в цикле всегда могут выполняться в радиусе. Для индикации позиции и базовой индикации кадра остается активной последняя активная G-команда этой группы.	
<ось>:	Идентификатор оси, для которой должно быть активировано спец. для оси программирование диаметра Разрешенными осевыми идентификаторами являются: • Имя гео-оси/оси канала или • Имя оси станка	
	Диапазон значений:	Указанной осью должна быть известная в канале ось. Прочие условия: • Ось через MD30460 $MA_BASE_FUNCTION_MASK должна быть допущена для спец. для оси программирование диаметра. • Круговые оси не разрешаются в качестве поперечных осей.
Применение спец. для канала программирования диаметра/радиуса		
DIAMCHANA:	С помощью команды DIAMCHANA[<ось>] **указанная ось** принимает состояние канала программирования диаметра/радиуса и после подвергается спец. для канала программированию диаметра/радиуса.	
DIAMCHAN:	С помощью команды DIAMCHANA **все** допущенные для спец. для оси программирование диаметра принимают состояние канала программирования диаметра/радиуса и после подвергаются спец. для канала программированию диаметра/радиуса.	

Действующее покадрово спец. для оси программирования диаметра/радиуса Действующее покадрово спец. для оси программирование диаметра/радиуса определяет тип указания размеров как значение диаметра или радиуса в программе обработки детали и синхронных действиях. Модальное состояние программирования диаметра/радиуса не изменяется.	
DAC:	С помощью команды DAC для указанной оси следующие данные размеров действуют покадрово: диаметр в абсолютном размере
DIC:	С помощью команды DIC для указанной оси следующие данные размеров действуют покадрово: диаметр в составном размере
RAC:	С помощью команды RAC для указанной оси следующие данные размеров действуют покадрово: радиус в абсолютном размере
RIC:	С помощью команды RIC для указанной оси следующие данные размеров действуют покадрово: радиус в составном размере

Примечание

При DIAMONA[<ось>] или DIAM90A[<ось>] фактические значения поперечной оси всегда индицируются как диаметр. Это же относится и к считыванию фактических значений в системе координат детали при MEAS, MEAW, $P_EP[x] и $AA_IW[x].

Примечание

При переходе дополнительной поперечной оси из-за требования GET с RELEASE[<ось>] состояние программирования диаметра/радиуса берется в другом канале.

Примеры

Пример 1: Действующее модально спец. для оси программирования диаметра/радиуса

X это поперечная ось в канале, для Y разрешено спец. для оси программирование диаметра.

Программный код	Комментарий
N10 G0 X0 Z0 DIAMON	; Спец. для канала программирование диаметра активно для X.
N15 DIAMOF	; Спец. для канала программирование диаметра выкл.
N20 DIAMONA[Y]	; Действующее модально спец. для оси программирование диаметра активно для Y.
N25 X200 Y100	; Программирование радиуса активно для X.
N30 DIAMCHANA[Y]	; Y получает состояние спец. для канала программирования диаметра/радиуса и зависит от него
N35 X50 Y100	; Программирование радиуса активно для X и Y.
N40 DIAMON	; Спец. для канала программирование диаметра вкл.
N45 X50 Y100	; Программирование диаметра активно для X и Y.

Пример 2: Действующее покадрово спец. для оси программирования диаметра/ радиуса

X это поперечная ось в канале, для Y разрешено спец. для оси программирование диаметра.

Программный код	Комментарий
`N10 DIAMON`	; Спец. для канала программирование диаметра вкл.
`N15 G0 G90 X20 Y40 DIAMONA[Y]`	; Действующее модально спец. для оси программирование диаметра активно для `Y`.
`N20 G01 X=RIC(5)`	; Действующее для этого кадра указание размера для `X`: радиус в составном размере.
`N25 X=RAC(80)`	; Действующее для этого кадра указание размера для `X`: радиус в абсолютном размере.
`N30 WHEN $SAA_IM[Y]>50 DO POS[X]=RIC(1)`	`;` `X` – это командная ось. Действующее для этого кадра указание размера для `X`: радиус в составном размере.
`N40 WHEN $SAA_IM[Y]>60 DO POS[X]=DAC(10)`	`;` `X` – это командная ось. Действующее для этого кадра указание размера для `X`: радиус в абсолютном размере.
`N50 G4 F3`	

Дополнительная информация

Значения диаметра (DIAMON/DIAM90A)

Значения диаметра действуют для следующих данных:

- индикация фактического значения поперечной оси в системе координат детали
- Режим JOG: Инкременты для размера шага и перемещения вручную маховичком
- Программирование конечных позиций:
  параметры интерполяции `I`, `J`, `K` при `G2`/`G3`, если они запрограммированы с `AC` абсолютно.
  При инкрементальном программировании (`IC`) из `I`, `J`, `K` всегда учитывается радиус.
- Чтение фактических значений в системе координат детали при:
  `MEAS`, `MEAW`, `$P_EP[X]`, `$AA_IW[X]`

Действующее покадрово спец. для оси программирование диаметра (DAC, DIC, RAC, RIC)

Операторы `DAC`, `DIC`, `RAC`, `RIC` разрешены для всех команд, для которых учитывается спец. для канала программирование диаметра:

- Позиция оси: `X...`, `POS`, `POSA`
- Качание: `OSP1`, `OSP2`, `OSS`, `OSE`, `POSP`

- Параметры интерполяции: `I`, `J`, `K`
- Линия контура: Прямая с указанием угла
- Быстрый отвод: `POLF[AX]`
- Перемещение в направлении инструмента: `MOVT`
- Мягкий подвод и отвод:
  `G140` до `G143`, `G147`, G148, `G247`, `G248`, `G347`, `G348`, `G340`, `G341`

3.8.5 Положение детали при токарной обработке

Обозначения осей

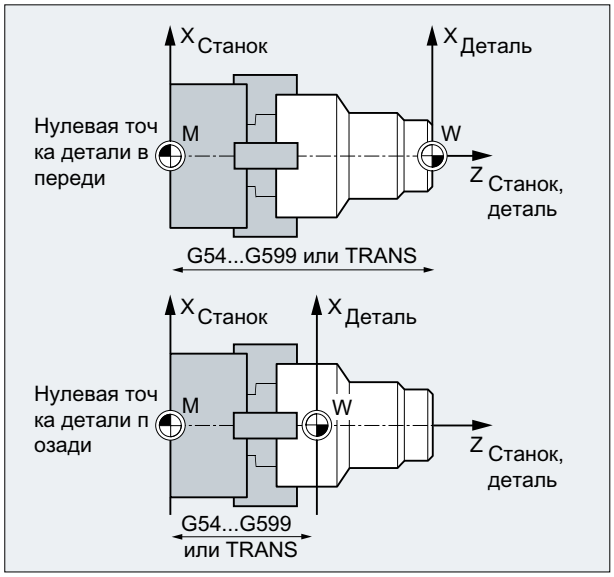
Обе расположенные вертикально друг на друге геометрические оси обычно обозначаются как:

продольная ось	= ось Z (абсцисса)
Поперечная ось	= ось X (ордината)

Нулевая точка детали

В то время как нулевая точка станка задана фиксировано, то положение нулевой точки детали может свободно выбираться на продольной оси. Обычно нулевая точка детали находится на передней или задней стороне детали.

Нулевая точка как станка, так и детали, находятся на центре вращения. Таким образом, настраиваемое смещение по оси X получается равным нулю.

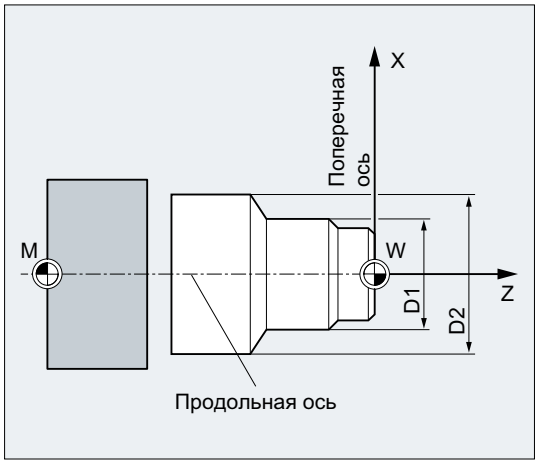


M	Нулевая точка станка
W	Нулевая точка детали

`Z`	продольная ось
`X`	Поперечная ось
`G54` до `G599` или `TRANS`	Вызов для положения нулевой точки детали

Поперечная ось

Для поперечной оси указание размеров обычно осуществляется как указание диаметра (двойной размер хода по сравнению с другими осями):



Какая из геометрических осей служит поперечной осью, должно быть установлено в машинных данных (→ изготовитель станка!).

3.9 Команды перемещения

3.9.1 Общая информация по командам перемещения

Элементы контура

Запрограммированный контур детали может состоять из следующих элементов контура:

- прямые
- дуги окружности
- винтовые линии (через наложение прямых и дуг окружностей)

Команды движения

Для изготовления этих элементов контура имеются различные команды движения:

- Движение ускоренным ходом (G0)
- Линейная интерполяция (G1)
- Круговая интерполяция по часовой стрелке (G2)
- Круговая интерполяция против часовой стрелки (G3)

Команды движения действуют модально.

Заданные конечные положения

Кадр движения содержит заданные конечные положения для перемещаемых осей (траекторные оси, синхронные оси, позиционирующие оси).

Программирование заданных конечных положений может осуществляться в декартовых или в полярных координатах.

Примечание

Один адрес оси может быть запрограммирован в кадре только один раз.

Стартовая точка-заданная точка

Движения перемещения всегда осуществляются от последней позиции, к которой был осуществлен переход, к запрограммированному заданному конечному положению. Это заданное конечное положение, в свою очередь, является стартовой позицией для следующей команды перемещения.

Контур детали

ВНИМАНИЕ
Не определенное использование инструмента Перед началом процесса обработки необходимо выполнить предварительное позиционирование инструмента, исключающее повреждение инструмента и детали.

Выполненные последовательно друг за другом элементы контура создают контур детали:

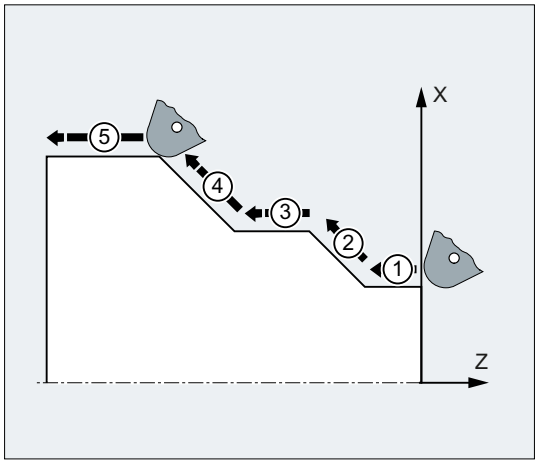


Рисунок 3-1 Кадры движения при токарной обработке

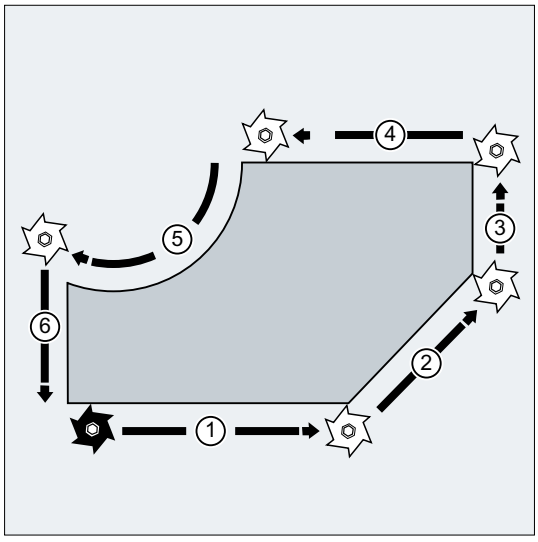


Рисунок 3-2 Кадры движения при фрезерной обработке

3.9.2 Команды движения с декартовыми координатами (G0, G1, G2, G3, X..., Y..., Z...)

Подвод в кадре ЧПУ к указанной в декартовых координатах позиций возможен с ускоренным ходом `G0`, линейной интерполяцией `G1` или круговой интерполяцией `G2` /`G3` .

Синтаксис

```
G0 X... Y... Z...
G1 X... Y... Z...
G2 X... Y... Z... ...
G3 X... Y... Z... ...
```

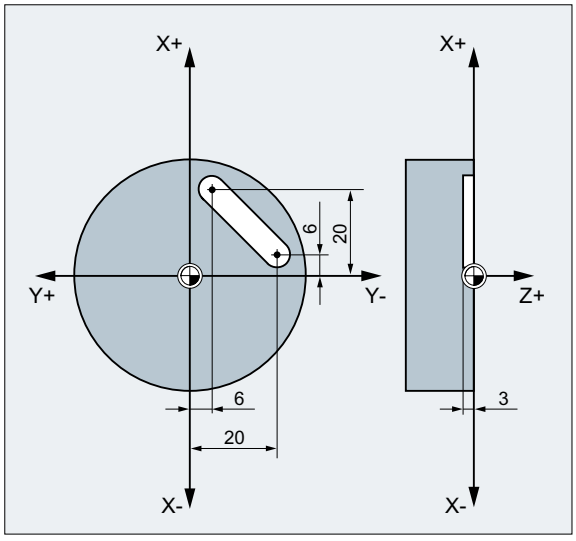
Значение

G0:	Команда для включения движения ускоренным ходом
G1:	Команда для включения линейной интерполяции
G2:	Команда для включения круговой интерполяции по часовой стрелке
G3:	Команда для включения круговой интерполяции против часовой стрелки
X...:	Декартова координата заданного конечного положения в направлении X
Y...:	Декартова координата заданного конечного положения в направлении Y
Z...:	Декартова координата заданного конечного положения в направлении Z

Примечание

Для круговой интерполяции G2 / G3 кроме координат заданного конечного положения X..., Y..., Z... необходимы и дополнительные данные (к примеру, координаты центра окружности; см. "Обзор (Страница 197)").

Пример



Программный код	Комментарий
N10 G17 S400 M3	; Выбор рабочей плоскости, шпиндель вправо
N20 G0 X40 Y-6 Z2	; Подвод в указанную в декартовых координатах стартовую позицию ускоренным ходом
N30 G1 Z-3 F40	; Включение линейной интерполяции, подача инструмента
N40 X12 Y-20	; Движение по наклонной прямой к указанной в декартовых координатах конечной позиции
N50 G0 Z100 M30	; Отвод для смены инструмента ускоренным ходом

3.9.3 Команды движения с полярными координатами

3.9.3.1 Исходная точка полярных координат (G110, G111, G112)

Исходная точка измерения называется "Полюс".

Указание полюса может осуществляться в декартовых или полярных координатах.

С помощью команд G110 до G112 исходная точка для полярных координат определяется однозначно. Поэтому на это не влияет ввод данных в абсолютном или составном размере.

Синтаксис

```
G110/G111/G112 X… Y… Z…
G110/G111/G112 AP=… RP=…
```

Значение

G110 ...:	С помощью команды G110 последующие полярные координаты относятся **к последней позиции, к которой был осуществлен подвод**.		
G111 ...:	С помощью команды G111 последующие полярные координаты относятся **к нулевой точке актуальной системы координат детали**.		
G112 ...:	С помощью команды G112 последующие полярные координаты относятся **к последнему действительному полюсу**.		
	Примечание: Команды G110...G112 должны программироваться в собственном кадре ЧПУ.		
X… Y… Z…:	Указание полюса в декартовых координатах		
AP=… RP=…:	Указание полюса в полярных координатах		
	AP=…:	Полярный угол Угол между полярным радиусом и горизонтальной осью рабочей плоскости (к примеру, ось X при G17). Положительное направление вращения осуществляется против часовой стрелки.	
		Диапазон значений:	± 0...360°
	RP=…:	Полярный радиус Данные указываются **всегда в абсолютных положительных значениях** в [мм] или [дюймах].	

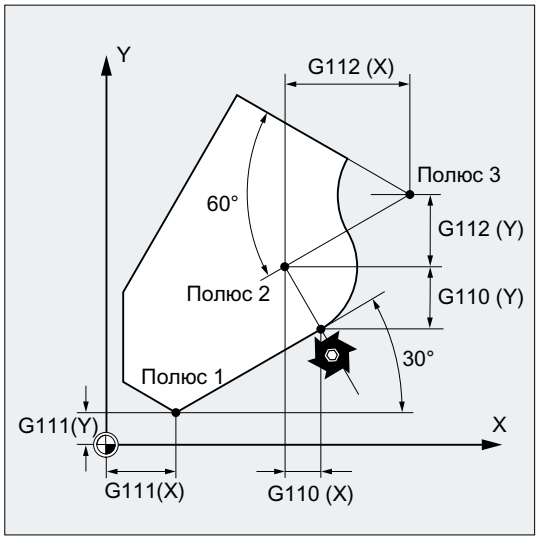
Примечание

В программе ЧПУ можно осуществлять покадровое переключение полярных и декартовых указаний размеров. Через использование декартовых идентификаторов координат (X..., Y..., Z...) происходит прямой возврат в декартову систему. Определенный полюс сохраняется до конца программы.

Примечание

Если полюс не указывается, то действует нулевая точка актуальной системы координат детали.

Пример

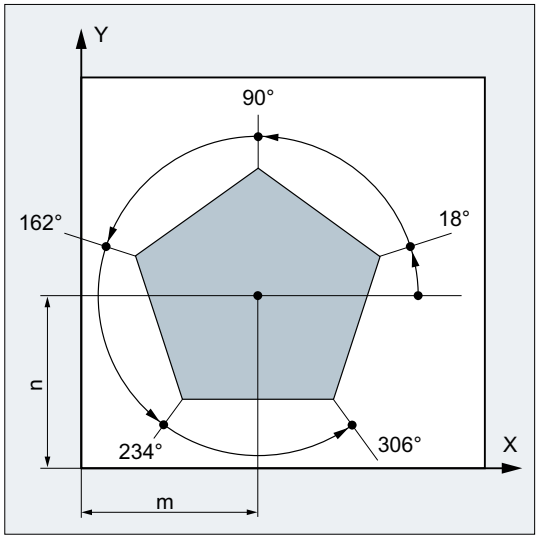


Полюса 1 до 3 определяются следующим образом:

- Полюс 1 с `G111 X... Y...`
- Полюс 2 с `G110 X... Y...`
- Полюс 3 с `G112 X... Y...`

3.9.3.2 Команды движения с полярными координатами (G0, G1, G2, G3, AP, RP)

Команды движения с полярными координатами имеют смысл тогда, когда деталь или часть детали измеряется из центральной точки и размеры указаны с углами и радиусами (к примеру, на схемах сверления).



Синтаксис

```
G0/G1/G2/G3 AP=… RP=…
```

Объяснение

G0:	Команда для включения движения ускоренным ходом	
G1:	Команда для включения линейной интерполяции	
G2:	Команда для включения круговой интерполяции по часовой стрелке	
G3:	Команда для включения круговой интерполяции против часовой стрелки	
AP:	Полярный угол Угол между полярным радиусом и горизонтальной осью рабочей плоскости (к примеру, ось X при G17). Положительное направление вращения осуществляется против часовой стрелки.	
	Диапазон значений:	± 0...360°
	Угол может быть указан как абсолютно, так и инкрементально:	
	AP=AC(...):	Ввод абсолютного размера
	AP=IC(...):	Ввод составного размера При вводе составного размера референтной точкой является последний запрограммированный угол.
	Полярный угол сохраняется до тех пор, пока не будет определен новый полюс или осуществлена смена рабочей плоскости.	
RP:	Полярный радиус Данные указываются **всегда в абсолютных положительных значениях** в [мм] или [дюймах]. Полярный радиус сохраняется до ввода нового значения.	

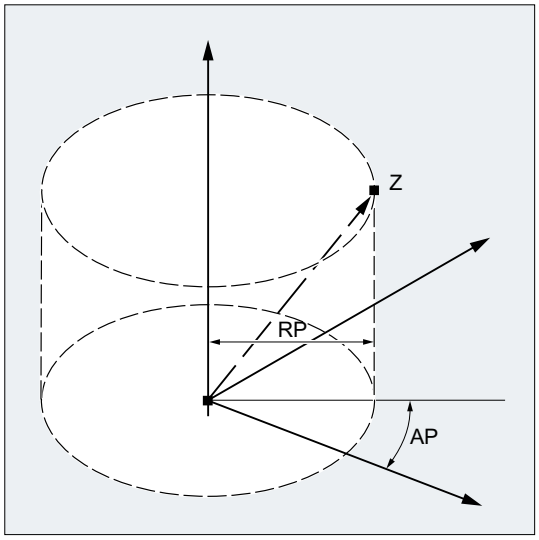
Примечание

Полярные координаты относятся к определенному с G110 ... G112 полюсу и действуют в выбранной с G17 до G19 рабочей плоскости.

Примечание

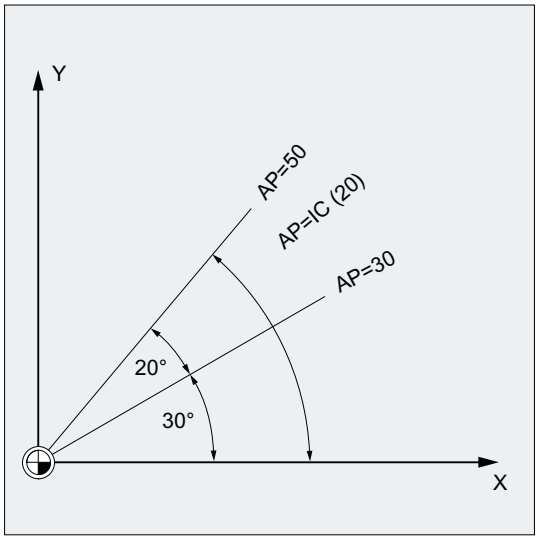
Расположенная вертикально к рабочей плоскости 3-ья геометрическая ось может быть дополнительно указана как декартова координата (см. рисунок ниже). Таким образом, можно программировать пространственные характеристики в цилиндрических координатах.

Пример: G17 G0 AP… RP… Z…



Граничные условия

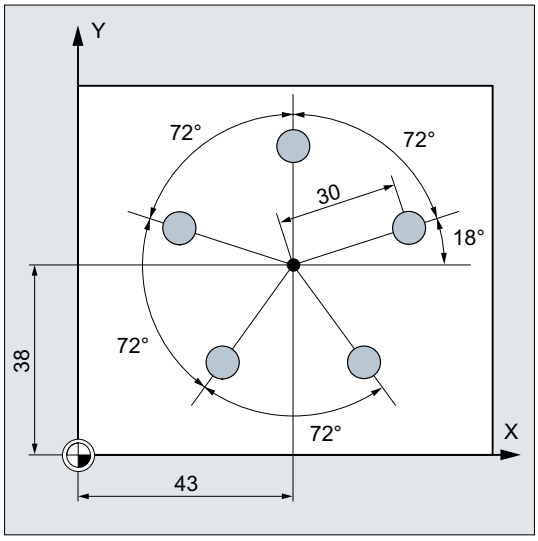
- В кадрах ЧПУ с полярным указанием конечной точки для выбранной рабочей плоскости не могут программироваться декартовы координаты, как то параметры интерполяции, адреса осей и т. п.
- Если с помощью `G110` ... `G112` полюс не определяется, то в качестве полюса автоматически рассматривается нулевая точка актуальной системы координат детали:



- Полярный радиус RP = 0
  Полярный радиус вычисляется из расстояния между вектором стартовой точки в плоскости полюса и активным вектором полюса. После этого вычисленный полярный радиус сохраняется модально.
  Это действует независимо от выбранного определения полюса (`G110` ... `G112`). Если обе точки запрограммированы идентично, то этот радиус = 0 и выводится ошибка 14095.
- Запрограммирован только полярный угол AP
  Если в актуальном кадре программируется не полярный радиус RP, а полярный угол AP, то при разнице между актуальной позицией и полюсом в координатах детали эта разница используется как полярный радиус и сохраняется модально. Если разница = 0, то заново задаются полярные координаты и модальный полярный радиус остается на нуле.

Пример

Создание схемы сверления



Позиции отверстий указаны в полярных координатах.

Каждое отверстие изготовляется одним и тем же способом:

предварительное сверление, сверление по размеру, развертывание ...

Последовательность обработки зафиксирована в подпрограмме.

Программный код	Комментарий
`N10 G17 G54`	; Рабочая плоскость `X/Y`, нулевая точка детали.
`N20 G111 X43 Y38`	; Определение полюса.
`N30 G0 RP=30 AP=18 Z5`	; Подвод в стартовую точку, данные в цилиндрических координатах.
`N40 L10`	; Вызов подпрограммы.
`N50 G91 AP=72`	; Переход к следующей позиции ускоренным ходом, полярный угол в составном размере, полярный радиус из кадра `N30` сохраняется и не должен указываться.
`N60 L10`	; Вызов подпрограммы.
`N70 AP=IC(72)`	.
`N80 L10`	...

Программный код	Комментарий
`N90 AP=IC(72)`	
`N100 L10`	…
`N110 AP=IC(72)`	
`N120 L10`	…
`N130 G0 X300 Y200 Z100 M30`	; Отвод инструмента, конец программы.

См. также

Обзор (Страница 197)

3.9.4 Движения ускоренного хода

3.9.4.1 Активация ускоренного хода (G0)

Перемещение траекторных осей со скоростью ускоренного хода включается при помощи G-команды G0.

Синтаксис

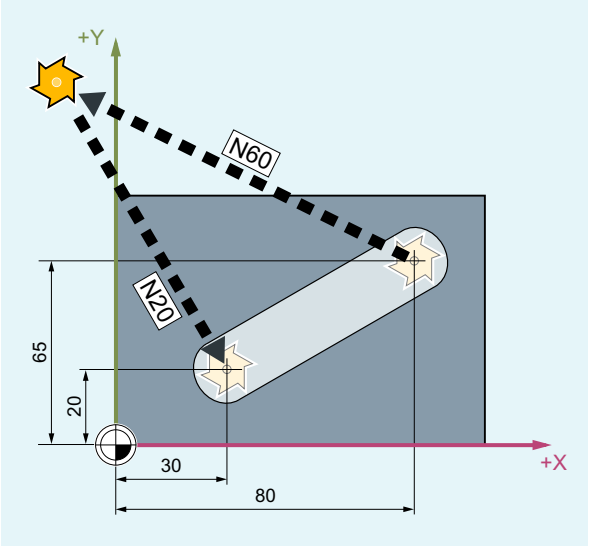
```
G0 X… Y… Z…
G0 RP=… AP=…
```

Объяснение

`G0`:	Перемещение осей со скоростью ускоренного хода	
	Активность:	модально
`X... Y... Z...`:	Указание конечной точки в декартовых координатах	
`RP=... AP=...` :	Указание конечной точки в полярных координатах	

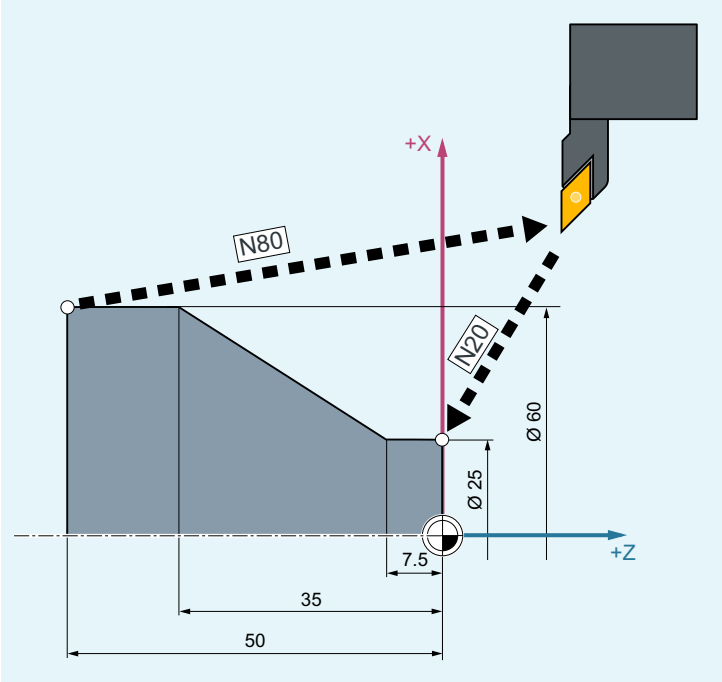
Примеры

Пример 1: Фрезерная обработка



Программный код	Комментарий
`N10 G90 S400 M3`	; Ввод абсолютного размера, шпиндель вправо
`N20 G0 X30 Y20 Z2`	; Подвод в стартовую позицию
`N30 G1 Z-5 F1000`	; Подача инструмента
`N40 X80 Y65`	; Движение по прямой
`N50 G0 Z2`	
`N60 G0 X-20 Y100 Z100 M30`	; Отвод инструмента, конец программы

Пример 2: Токарная обработка



Программный код	Комментарий
`N10 G90 S400 M3`	; Ввод абсолютного размера, шпиндель вправо
`N20 G0 X25 Z5`	; Подвод в стартовую позицию
`N30 G1 G94 Z0 F1000`	; Подача инструмента
`N40 G95 Z-7.5 F0.2`	
`N50 X60 Z-35`	; Движение по прямой
`N60 Z-50`	
`N70 G0 X62`	
`N80 G0 X80 Z20 M30`	; Отвод инструмента, конец программы

3.9.4.2 Включение/выключение линейной интерполяции для движений ускоренным ходом (RTLION, RTLIOF)

Независимо от предварительной настройки (MD20730 $MC_G0_LINEAR_MODE) поведение интерполяции при движениях ускоренным ходом можно настроить с помощью команд G-группы 55 и в программе обработки детали.

Синтаксис

```
RTLIOF
...
RTLION
```

Объяснение

RTLIOF:	G-команда для выключения линейной интерполяции ⇒ При ускоренном ходе (G0) **нелинейная** интерполяция активна. Все траекторные оси достигают своей конечной точки независимо друг от друга.	
	Активность:	модально
RTLION:	G-команда для включения линейной интерполяции ⇒ При ускоренном ходе (G0) **линейная** интерполяция активна. Все траекторные оси достигают своей конечной точки одновременно.	
	Активность:	модально

Примечание

Условия для RTLIOF

Для **нелинейной** интерполяции при RTLIOF должны быть выполнены следующие условия:

- Активных трансформаций (TRAORI, TRANSMIT, ...) нет.
- G60 активна (остановка в конце кадра).
- Активных компрессоров (COMPOF) нет.
- Активной коррекции на радиус инструмента (G40) нет.
- Контурные маховички не выбраны.
- Вырубка неактивна.

Если одно из этих условий не выполнено, при RTLION интерполяция линейная.

Пример

Программный код	Комментарий
	; Линейная интерполяция предустановлена:
	; MD20730 $MC_GO_LINEAR_MODE == TRUE
...	
N30 **RTLIOF**	; Линейная интерполяция выключена.
N40 G0 X0 Y10	; Кадры G0 перемещаются с нелинейной интерполяцией.
N50 G41 X20 Y20	; КРИ активна ⇒ Кадры G0 перемещаются с линейной интерполяцией.
N60 G40 X30 Y30	; КРИ неактивна ⇒ Кадры G0 перемещаются с нелинейной интерполяцией.
N70 **RTLION**	; Линейная интерполяция включена.
...	

Дополнительная информация

Чтение актуального поведения интерполяции

Системные переменные $AA_G0MODE позволяют считывать актуальное поведение интерполяции.

3.9.4.3 Коррекция допусков для движений ускоренным ходом (STOLF, CTOLG0, OTOLG0)

Настраиваемые через машинные данные допуски для движений ускоренным ходом (допуски G0) можно временно корректировать в программе обработки детали. При этом настройки в машинных данных не изменяются. После сброса канала или конца программы спроектированные допуски снова вступают в силу.

Условия

Допуски G0 вступают в силу, только если выполнены следующие условия:

- Активна одна из следующих функций:
  - Функция компрессора COMP...
  - Функция перешлифовки G642 или G645
  - Перешлифовка ориентации OST
  - Сглаживание ориентации ORISON
  - Сглаживание при ориентации относительно траектории ORIPATH
- В программе обработки детали несколько (≥ 2) кадров G0 следуют друг за другом. В случае одного единственного кадра G0 допуски G0 не вступают в силу, т. к. при переходе от движения не-G0 к движению G0 (и наоборот) всегда действует "**меньший допуск**" (допуск обработки детали)!

Синтаксис

Настройка относительного допуска G0

```
STOLF=<Value>
```

Настройка абсолютных допусков G0

```
CTOLG0=<Value>
OTOLG0=<Value>
```

Объяснение

STOLF:	Адрес программирования временно действующего коэффициента допуска для движений ускоренным ходом			
	<Value>:	Коэффициент допуска G0		
		Тип:	REAL	
		Значение:	≥ 0:	Коэффициент допуска G0 может быть как больше, так и меньше 1,0. Если коэффициент равен 1,0 (стандартное значение), то для движений ускоренным ходом действуют те же допуски, что и для движений неускоренным ходом. В обычной ситуации устанавливается коэффициент допуска > 1,0. Запрограммированный коэффициент допуска G0 действует, пока не будет перезаписан новым программированием STOLF, сменен программированием CTOLG0/OTOLG0 или удален сбросом канала или конца программы.
			< 0:	Удаление запрограммированного коэффициента допуска ⇒ Снова действует коэффициент допуска, установленный в машинных данных.
CTOLG0:	Адрес программирования временно действующего допуска контура для движений ускоренным ходом			
	<Value>:	Абсолютное значение допуска контура		
		Тип:	REAL	
		Значение:	≥ 0:	Запрограммированное абсолютное значение допуска контура действует, пока не будет перезаписано новым программированием CTOLG0, сменено программированием STOLF или удалено сбросом канала или конца программы.
			< 0:	Удаление запрограммированного значения допуска ⇒ Снова действует значение допуска, установленное в машинных данных.
OTOLG0:	Адрес программирования временно действующего допуска ориентации для движений ускоренным ходом			
	<Value>:	Абсолютное значение допуска ориентации		
		Тип:	REAL	
		Значение:	≥ 0:	Запрограммированное абсолютное значение допуска ориентации действует, пока не будет перезаписано новым программированием OTOLG0, сменено программированием STOLF или удалено сбросом канала или конца программы.
			< 0:	Удаление запрограммированного значения допуска ⇒ Снова действует значение допуска, установленное в машинных данных.

Примечание

Приоритет всегда имеет последний запрограммированный адрес, как это видно в следующих примерах:

- При программировании CTOLG0 при существующем STOLF для перешлифовки контура используется значение допуска, запрограммированное с CTOLG0.
- При программировании OTOLG0 при существующем STOLF для перешлифовки ориентации также используется значение допуска, запрограммированное с OTOLG0.
- После нового программирования STOLF снова используется коэффициент допуска для допуска контура и ориентации.

Примеры

Пример 1: Настройка относительного допуска G0

Программный код	Комментарий
COMPCAD G645 G1 F10000	; Функция компрессора COMPCAD
X... Y... Z...	; Здесь действуют машинные и установочные данные.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
G0 X... Y... Z...	; Здесь действуют машинные данные $MC_G0_TOLERANCE_FACTOR (например = 3), т. е. допуск перешлифовки из: $MC_G0_TOLERANCE_FACTOR * $MA_COMPRESS_POS_TOL
CTOL=0.02	
STOLF=4	
G1 X... Y... Z...	; С этого места действует допуск контура в 0,02 мм.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
X... Y... Z...	; С этого места действует коэффициент допуска G0 в 4, т. е. допуск контура в 0,08 мм.
...	

Пример 2: Настройка абсолютных допусков G0

В машинных данных должны быть предустановлены следующие абсолютные допуски G0:

- Допуск контура G0: 0,1
- Допуск ориентации G0: 1,0

В программе обработки детали эти допуски должны быть временно скорректированы:

Программный код	Комментарий
COMPCAD G645 G1 F10000	; Функция компрессора COMPCAD
X... Y... Z...	; С этого места действуют настроенные допуски обработки детали.
X... Y... Z...	

Программный код	Комментарий
X... Y... Z...	
G0 X... Y... Z...	
G0 X... Y... Z...	; Здесь действуют настроенные абсолютные допуски G0.
CTOLG0=0.2 OTOLG0=2.0	; Программирование абсолютных допусков G0.
G1 X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
X... Y... Z...	; С этого места действуют запрограммированные допуски G0.
...	

Дополнительная информация

Чтение коэффициента допуска G0

Текущий действующий коэффициент допуска для движений ускоренным ходом можно считывать через системные переменные:

- В синхронных действиях или с остановкой предварительной обработки в программе обработки детали через системные переменные:

$AC_STOLF	Активный коэффициент допуска G0 Коэффициент допуска G0, который действовал при подготовке текущего кадра главного хода.

- Без остановки предварительной обработки в программе обработки детали через системные переменные:

$P_STOLF	Запрограммированный коэффициент допуска G0

Если в активной программе обработки детали нет запрограммированного со STOLF значения, то обе эти системные переменные возвращают установленное в машинных данных значение.

Если в кадре нет активного ускоренного хода (G0), то эти системные переменные всегда возвращают значение 1.

Считывание абсолютных допусков G0

Текущие действительные абсолютные допуски для движений ускоренным ходом можно считывать через системные переменные:

- В синхронных действиях или с остановкой предварительной обработки в программе обработки детали через системные переменные:

$AC_CTOL_G0_ABS	Активный допуск контура в случае движений G0 Допуск контура G0, который действовал при подготовке текущего кадра главного хода.
$AC_OTOL_ G0_ABS	Активный допуск ориентации в случае движений G0 Допуск ориентации G0, который действовал при подготовке текущего кадра главного хода.

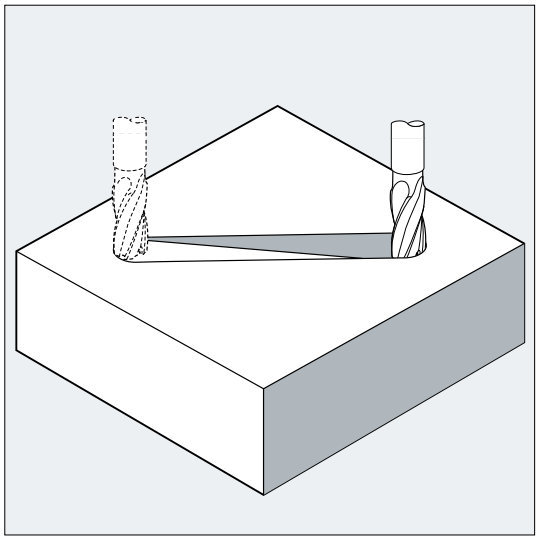
- Без остановки предварительной обработки в программе обработки детали через системные переменные:

$P_CTOL_ G0_ABS	Запрограммированный допуск контура в случае движений G0
$P_OTOL_ G0_ABS	Запрограммированный допуск ориентации в случае движений G0

Если в активной программе обработки детали не запрограммированы абсолютные допуски G0 с CTOLG0 и OTOLG0, то эти системные переменные возвращают значения, установленные в машинных данных.

3.9.5 Линейная интерполяция (G1)

С помощью `G1` инструмент движется по параллельным осям, наклонным или имеющим любое расположение в пространстве прямым. Линейная интерполяция позволяет создавать поверхности 3D, пазы и много другое.



Синтаксис

```
G1 X… Y… Z … F…
G1 AP=… RP=… F…
```

Объяснение

`G1`:	Линейная интерполяция (линейная интерполяция с подачей)
`X... Y... Z...`:	Конечная точка в декартовых координатах
`AP=...`:	Конечная точка в полярных координатах, здесь полярный угол

RP=...:	Конечная точка в полярных координатах, здесь полярный радиус
F...:	Скорость подачи в мм/мин. Инструмент движется с подачей F по прямой от текущей стартовой точки к запрограммированной заданной точке. Заданная точка вводится в декартовых или полярных координатах. На этой траектории осуществляется обработка детали. Пример: G1 G94 X100 Y20 Z30 A40 F100 Движение к конечной точке в X, Y, Z осуществляется с подачей 100 мм/мин, круговая ось A перемещается в качестве синхронной оси так, что все четыре движения завершаются в одно время.

Примечание

G1 действует модально.

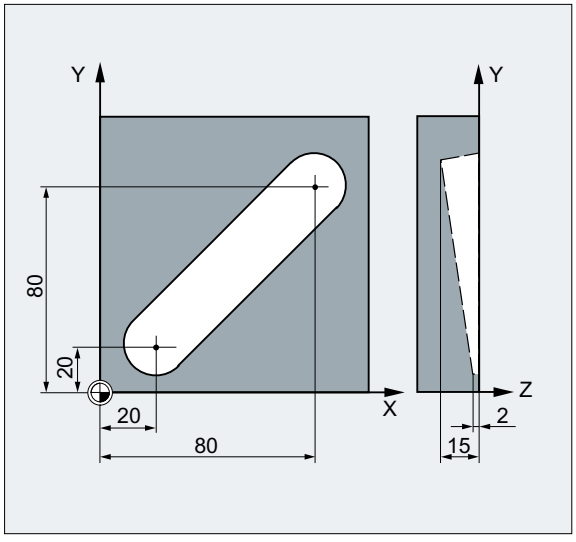
Для обработки необходимо указать число оборотов шпинделя S и направление вращения шпинделя M3/M4.

С помощью FGROUP можно определить группы осей, для которых действует подача по траектории F. Подробности см. в главе "Параметры траектории".

Примеры

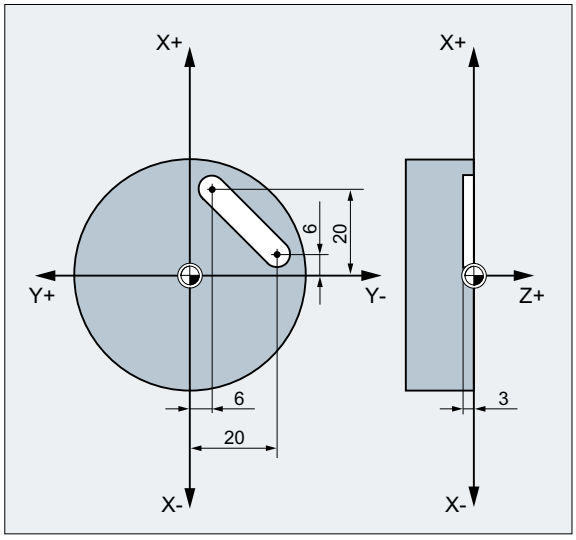
Пример 1: Изготовление паза (фрезерование)

Инструмент движется от стартовой к конечной точке в направлении X/Y. Одновременно происходит подача в направлении Z.



Программный код	Комментарий
N10 G17 S400 M3	; Выбор рабочей плоскости, шпиндель вправо
N20 G0 X20 Y20 Z2	; Подвод в стартовую позицию
N30 G1 Z-2 F40	; Подача инструмента
N40 X80 Y80 Z-15	; Движение по наклонной прямой
N50 G0 Z100 M30	; Отвод для смены инструмента

Пример 2: Изготовление паза (токарная обработка)



Программный код	Комментарий
`N10 G17 S400 M3`	; Выбор рабочей плоскости, шпиндель вправо
`N20 G0 X40 Y-6 Z2`	; Подвод в стартовую позицию
`N30 G1 Z-3 F40`	; Подача инструмента
`N40 X12 Y-20`	; Движение по наклонной прямой
`N50 G0 Z100 M30`	; Отвод для смены инструмента

3.9.6 Круговая интерполяция

3.9.6.1 Обзор

Круговая интерполяция позволяет изготавливать полные круги или дуги окружностей.

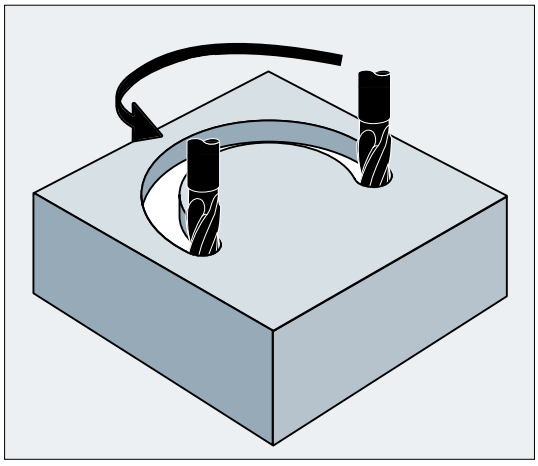


Рисунок 3-3 Пример использования: Фрезерование кругового паза

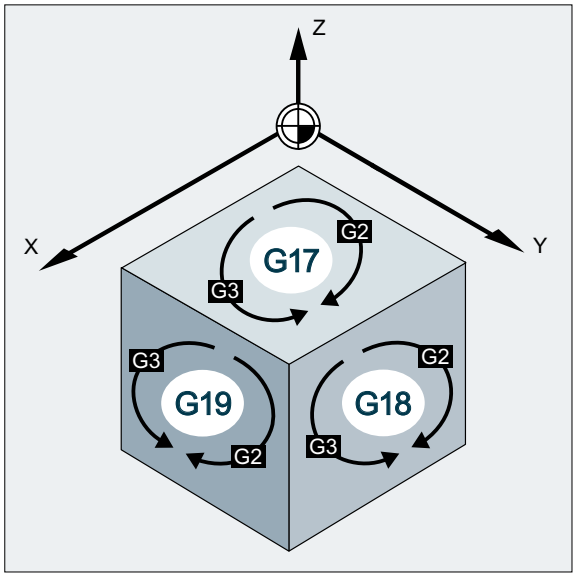
Варианты программирования

СЧПУ предлагает различные возможности программирования круговых движений. Благодаря этому возможен прямой перенос практически любого вида измерения чертежа:

- Круговая интерполяция с центром и конечной точкой (G2/G3, X... Y... Z..., I... J... K...) (Страница 198)
- Круговая интерполяция с радиусом и конечной точкой (G2/G3, X... Y... Z..., CR) (Страница 200)
- Круговая интерполяция с апертурным углом и конечной точкой / центром (G2/G3, X... Y... Z... / I... J... K..., AR) (Страница 202)
- Круговая интерполяция с полярными координатами (G2/G3, AP, RP) (Страница 204)
- Круговая интерполяция с промежуточной и конечной точкой (CIP, X... Y... Z..., I1... J1... K1...) (Страница 206)
- Круговая интерполяция с тангенциальным переходом (CT, X... Y... Z...) (Страница 208)

Плоскость для круговой интерполяции

Для вычисления направления вращения окружности – с G2 по часовой стрелке или G3 против часовой стрелки – СЧПУ требуется указание рабочей плоскости (Страница 153).



Исключение:

Можно также создавать окружности вне выбранной рабочей плоскости (не при указанном апертурном угле). В этом случае плоскость окружности определяют адреса осей, программируемые как конечные точки окружности.

3.9.6.2 Круговая интерполяция с центром и конечной точкой (G2/G3, X... Y... Z..., I... J... K...)

Вариант круговой интерполяции, использующий для интерполяции **центр** и **конечную точку** кругового элемента контура.

Если окружность программируется без конечной точки, то получается полный круг.

Синтаксис

```
G2/G3 X… Y… Z… I… J… K…
G2/G3 X… Y… Z… I=AC(…) J=AC(…) K=(AC…)
```

Объяснение

G2:	Круговая интерполяция по часовой стрелке	
	Активность:	модально
G3:	Круговая интерполяция против часовой стрелки	
	Активность:	модально

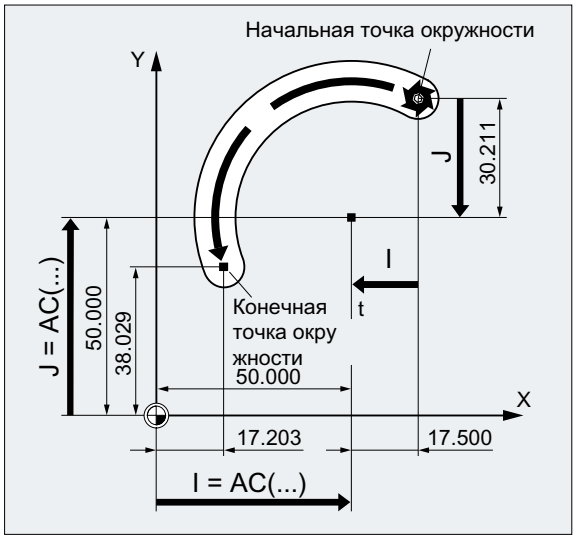
`X... Y... Z...:`	Конечная точка окружности в декартовых координатах В зависимости от текущей действительной настройки указания размеров `G90`/`G91` или `...=AC(...)` / `...=IC(...)` координаты конечной точки окружности интерпретируются либо в абсолютном размере, либо в составном размере.
`I... J... K...:`	Параметр интерполяции для указания координат центра окружности в направлении X, Y, Z Координаты центра окружности стандартно указываются в составном размере относительно начальной точки окружности. Если координаты центра окружности необходимо указать в абсолютном размере относительно нулевой точки детали, то параметры интерполяции I, J, K необходимо запрограммировать следующим образом: `I=AC(...) J=AC(...) K=AC(...)` **Указание** Параметр интерполяции со значением 0 может не указываться, соответствующий второй параметр должен быть указан в любом случае.

Примечание

Предустановка `G90`/`G91` абсолютного или составного размера действует только для конечной точки окружности.

Примеры

Пример 1: Фрезерная обработка



Указание центра в составном размере

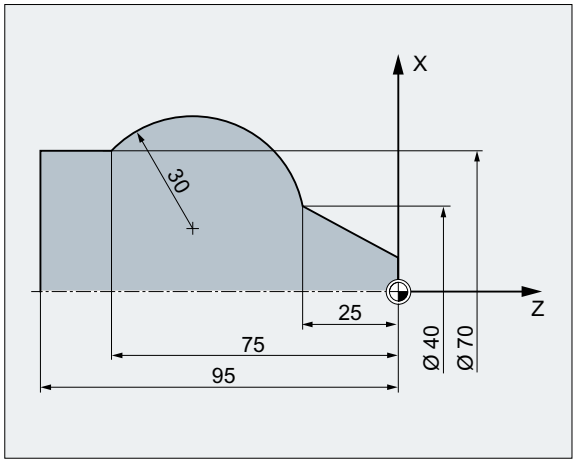
```
N10 G0 X67.5 Y80.211
N20 G3 X17.203 Y38.029 I-17.5 J-30.211 F500
```

Указание центра в абсолютном размере

```
N10 G0 X67.5 Y80.211
```

```
N20 G3 X17.203 Y38.029 I=AC(50) J=AC(50)
```

Пример 2: Токарная обработка



Указание центра в составном размере

```
N120 G0 X12 Z0
N125 G1 X40 Z-25 F0.2
N130 G3 X70 Z-75 I-3.335 K-29.25
N135 G1 Z-95
```

Указание центра в абсолютном размере

```
N120 G0 X12 Z0
N125 G1 X40 Z-25 F0.2
N130 G3 X70 Z-75 I=AC(33.33) K=AC(-54.25)
N135 G1 Z-95
```

3.9.6.3 Круговая интерполяция с радиусом и конечной точкой (G2/G3, X... Y... Z..., CR)

Вариант круговой интерполяции, использующий для интерполяции **радиус** и **конечную точку** кругового элемента контура.

Примечание

Этот вариант **не** позволяет программировать полные круги (угол перемещения 360°).

Синтаксис

```
G2/G3 X… Y… Z… CR=±...
```

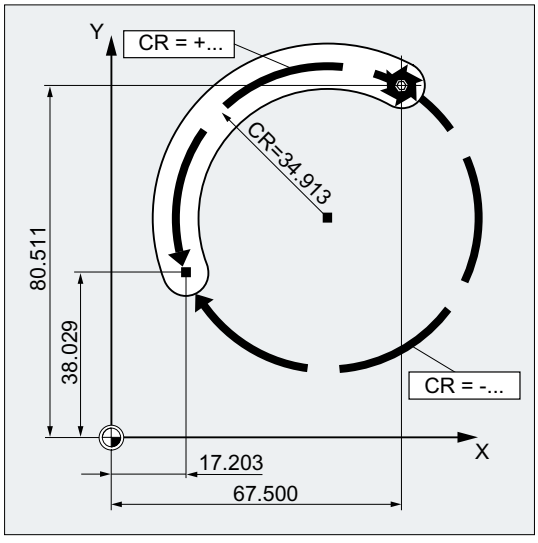
Объяснение

G2:	Круговая интерполяция по часовой стрелке	
	Активность:	модально

G3:	Круговая интерполяция против часовой стрелки	
	Активность:	модально
X... Y... Z...:	Конечная точка окружности в декартовых координатах В зависимости от текущей действительной настройки указания размеров G90/G91 или ...=AC(...) / ...=IC(...) координаты конечной точки интерпретируются либо в абсолютном размере, либо в составном размере.	
CR=±...:	Радиус окружности Посредством знака указывается, должен ли угол перемещения быть больше или меньше 180°. Положительный знак можно не указывать.	
	CR=+...:	Угол перемещения ≤ 180°
	CR=-...:	Угол перемещения > 180°
	Указание Исходя из существующей практики, не существует ограничения для размера максимального программируемого радиуса.	

Примеры

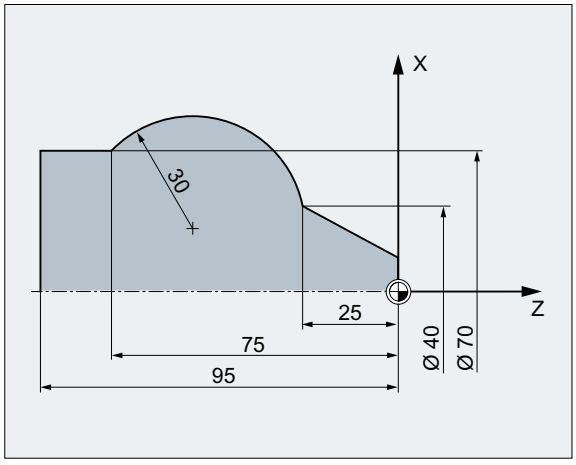
Пример 1: Фрезерная обработка



Программный код

```
N10 G0 X67.5 Y80.511
N20 G3 X17.203 Y38.029 CR=34.913 F500
...
```

Пример 2: Токарная обработка



Программный код

```
...
N125 G1 X40 Z-25 F0.2
N130 G3 X70 Z-75 CR=30
N135 G1 Z-95
...
```

3.9.6.4 Круговая интерполяция с апертурным углом и конечной точкой / центром (G2/G3, X... Y... Z... / I... J... K..., AR)

Вариант круговой интерполяции, использующий для интерполяции **апертурный угол** и **центра** или **конечную точку** кругового элемента контура.

Примечание

Этот вариант **не** позволяет программировать полные круги (угол перемещения 360°).

Синтаксис

```
G2/G3 X… Y… Z… AR=...
G2/G3 I… J… K… AR=...
```

Объяснение

G2:	Круговая интерполяция по часовой стрелке	
	Активность:	модально
G3:	Круговая интерполяция против часовой стрелки	
	Активность:	модально

`X... Y... Z...:`	Конечная точка окружности в декартовых координатах В зависимости от текущей действительной настройки указания размеров `G90`/`G91` или `...=AC(...)` / `...=IC(...)` координаты конечной точки окружности интерпретируются либо в абсолютном размере, либо в составном размере.	
`I... J... K...:`	Параметр интерполяции для указания координат центра окружности в направлении X, Y, Z Координаты центра окружности стандартно указываются в составном размере относительно начальной точки окружности. Если координаты центра окружности необходимо указать в абсолютном размере относительно нулевой точки детали, то параметры интерполяции I, J, K необходимо запрограммировать следующим образом: `I=AC(...) J=AC(...) K=AC(...)` **Указание** Параметр интерполяции со значением 0 может не указываться, соответствующий второй параметр должен быть указан в любом случае.	
`AR=...:`	Апертурный угол	
	Диапазон значений:	0° ... 360°

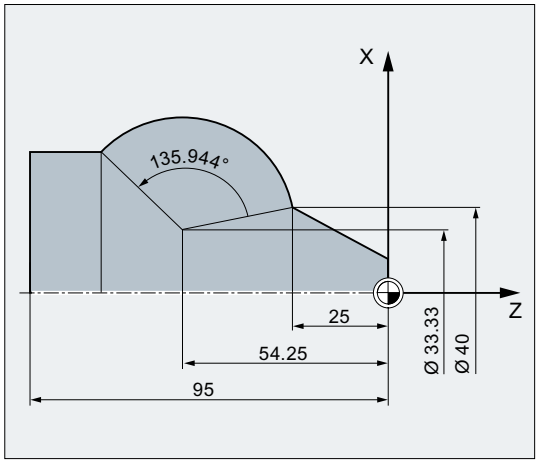
Примеры

Пример 1: Фрезерная обработка

Программный код

```
N10 G0 X67.5 Y80.211
N20 G3 X17.203 Y38.029 AR=140.134 F500
N20 G3 I-17.5 J-30.211 AR=140.134 F500
```

Пример 2: Токарная обработка



Программный код

```
N125 G1 X40 Z-25 F0.2
N130 G3 X70 Z-75 AR=135.944
N130 G3 I-3.335 K-29.25 AR=135.944
N130 G3 I=AC(33.33) K=AC(-54.25) AR=135.944
N135 G1 Z-95
```

3.9.6.5 Круговая интерполяция с полярными координатами (G2/G3, AP, RP)

Вариант круговой интерполяции, использующий для интерполяции **конечную точку окружности в полярных координатах**.

При этом действует следующее соглашение:

- Полюс лежит в центре окружности.
- Полярный радиус соответствует радиусу окружности.

Синтаксис

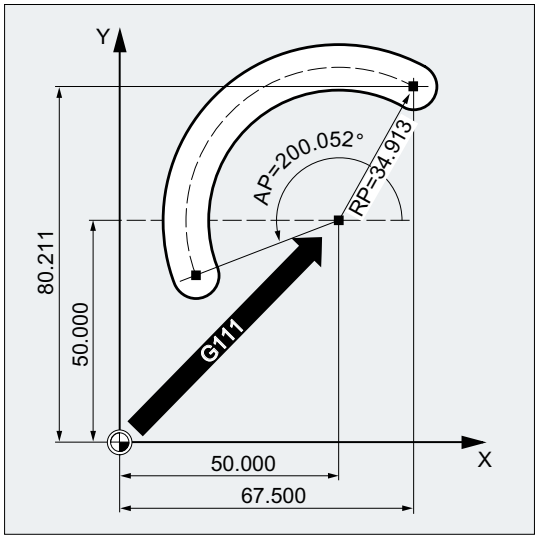
```
G2/G3 AP=... RP=...
```

Объяснение

G2:	Круговая интерполяция по часовой стрелке	
	Активность:	модально
G3:	Круговая интерполяция против часовой стрелки	
	Активность:	модально
AP=... RP=...:	Конечная точка окружности в полярных координатах	
	AP=...:	Полярный угол
	RP=...:	Полярный радиус (≙ радиус окружности)

Примеры

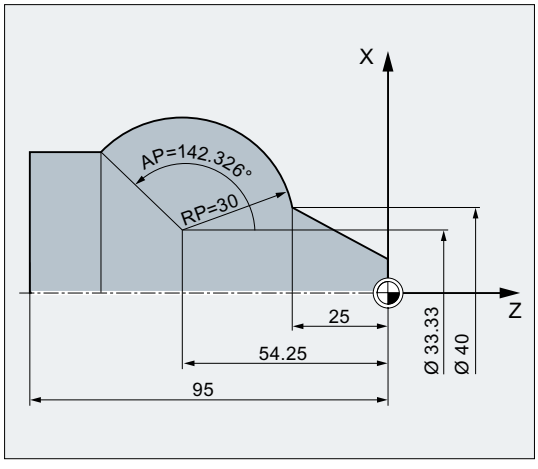
Пример 1: Фрезерная обработка



Программный код

```
N10 G0 X67.5 Y80.211
N20 G111 X50 Y50
N30 G3 RP=34.913 AP=200.052 F500
```

Пример 2: Токарная обработка



Программный код

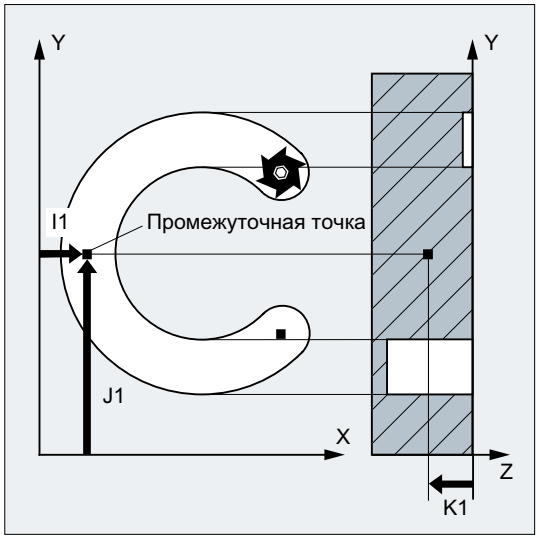
```
N125 G1 X40 Z-25 F0.2
N130 G111 X33.33 Z-54.25
N135 G3 RP=30 AP=142.326
N140 G1 Z-95
```

3.9.6.6 Круговая интерполяция с промежуточной и конечной точкой (CIP, X... Y... Z..., I1... J1... K1...)

Запрограммированный G-командой `CIP` вариант круговой интерполяции позволяет получать интерполяцию расположенных под углом в пространстве дуг окружностей.

Круговое движение описывается через **промежуточную точку** и **конечную точку** кругового контура.

Направление перемещения следует из последовательности: начальная точка → промежуточная точка → конечная точка.



Синтаксис

```
CIP X… Y… Z… I1=AC(…) J1=AC(…) K1=(AC…)
```

Объяснение

`CIP`:	Круговая интерполяция через промежуточную точку	
	Активность:	модально
`X... Y... Z...`:	Конечная точка окружности в декартовых координатах В зависимости от текущей действительной настройки указания размеров `G90`/`G91` или `...=AC(...)` / `...=IC(...)` координаты конечной точки окружности интерпретируются либо в абсолютном размере, либо в составном размере.	
`I1... J1... K1...`:	Параметр интерполяции для указания координат промежуточной точки окружности в направлении X, Y, Z В зависимости от текущей действительной настройки указания размеров `G90`/`G91` или `...=AC(...)` / `...=IC(...)` координаты промежуточной точки окружности интерпретируются либо в абсолютном размере, либо в составном размере. **Указание** Параметр интерполяции со значением 0 может не указываться, соответствующий второй параметр должен быть указан в любом случае.	

Примечание

Предустановки G90/G91 (указание абсолютного или составного размера) действуют для промежуточной точки окружности и конечной точки окружности.

При активном указании составного размера G91 или ...=IC(...) исходной точкой для промежуточной и конечной точки окружности служит начальная точка окружности.

Примечание

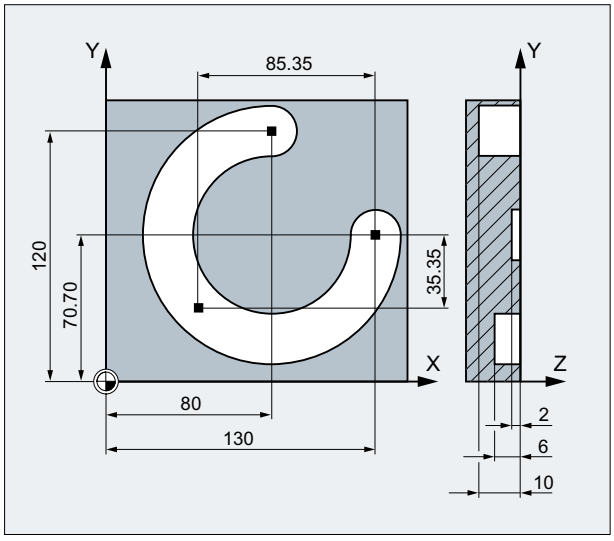
Токарная технология

Программирование диаметра в параметре интерполяции для поперечной оси при программировании окружностей с помощью CIP не поддерживается. Поэтому параметр интерполяции для поперечной оси следует программировать в **радиусе**.

Примеры

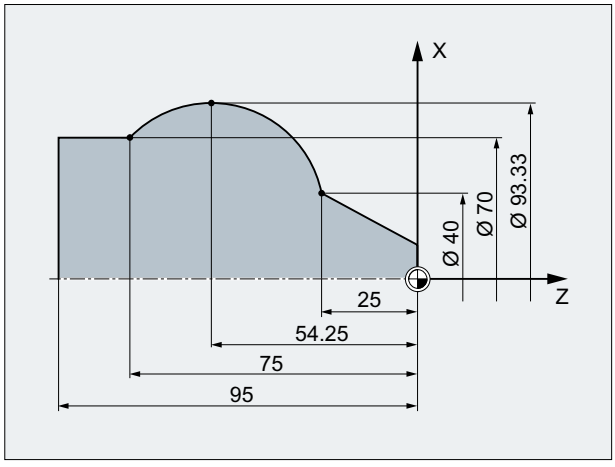
Пример 1: Фрезерная обработка

Для изготовления расположенной под углом в пространстве кольцевой канавки описывается окружность через указание промежуточной точки с 3 параметрами интерполяции и конечной точки также с 3 координатами.



Программный код	Комментарий
N10 G0 G90 X130 Y70.70 S800 M3	; Подвод в стартовую точку.
N20 G17 G1 Z-2 F100	; Подача инструмента.
N30 CIP X80 Y120 Z-10 I1=IC(-85.35) J1=IC(-35.35) K1=-6	; Конечная точка окружности и промежуточная точка:.
	; Координаты для всех 3 геометрических осей.
N40 M30	; Конец программы.

Пример 2: Токарная обработка



Программный код	Комментарий
...	
`N125 G1 G90 X40 Z-25 F0.2`	
`N130 CIP X70 Z-75` **`I1=IC(26.665)`** `K1=IC(-29.25)`	; Параметр интерполяции `I1` для поперечной оси должен быть запрограммирован в радиусе.
; или	
`; N130 CIP X70 Z-75` **`I1=46.665`** `K1=-54.25`	
`N135 G1 Z-95`	

3.9.6.7 Круговая интерполяция с тангенциальным переходом (CT, X... Y... Z...)

Запрограммированный G-командой `CT` вариант круговой интерполяции позволяет получать интерполяцию дуг окружностей, примыкающих по касательной к запрограммированному ранее элементу контура.

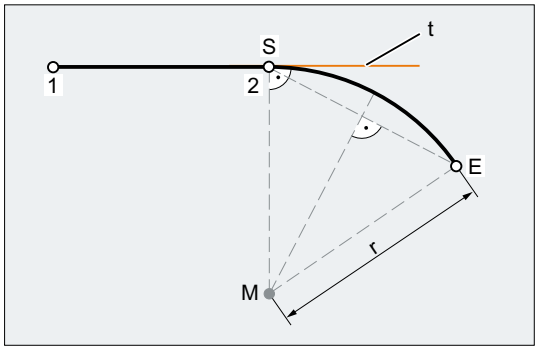
Окружность определяется при этом через **стартовую и конечную точку** и **направление касательной в стартовой точке**.

Примечание

Направление касательной в стартовой точке

Направление касательной в стартовой точке кадра CT определяется из конечной касательной запрограммированного контура последнего предшествующего кадра с движением перемещения.

Между этим кадром и текущим кадром может находиться любое количество кадров без информации перемещения.



S Стартовая точка
E Конечная точка
M Центр окружности
r Радиус окружности
t Конечная касательная запрограммированного контура последнего предшествующего кадра с движением перемещения

Рисунок 3-4 Круговая траектория S-E, примыкающая по касательной к участку прямой 1-2

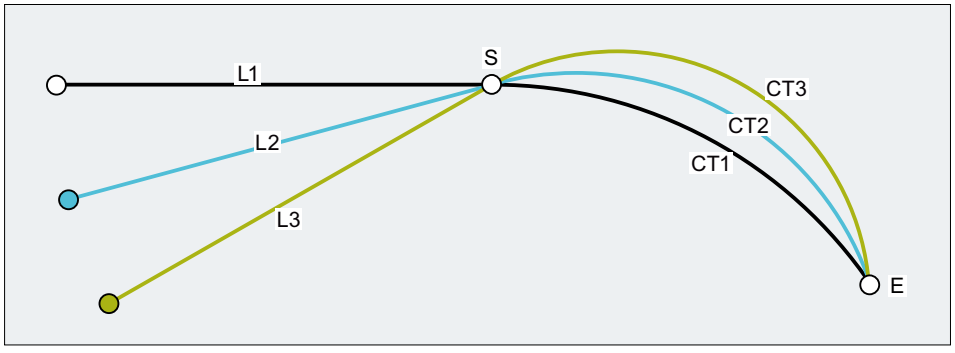


Рисунок 3-5 Примыкающие по касательной круговые траектории зависят от предшествующего элемента контура.

Синтаксис

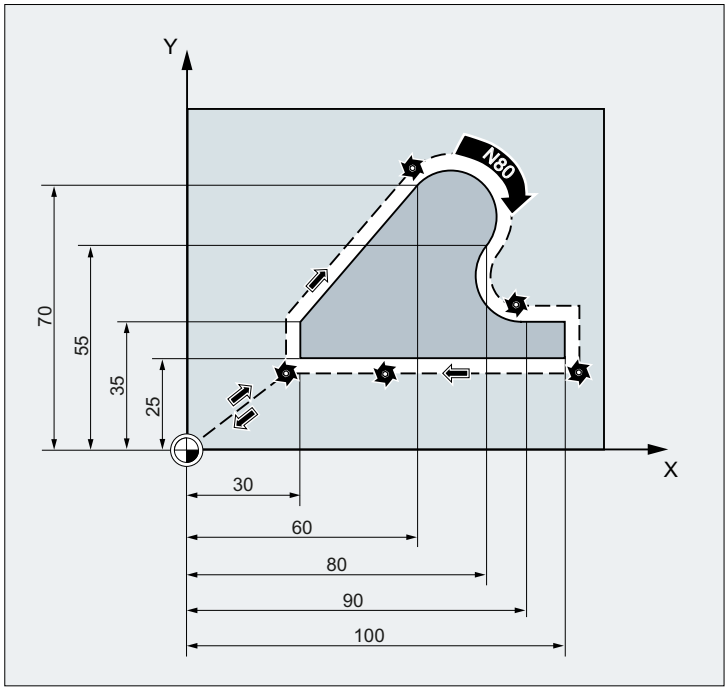
```
CT X... Y... Z...
```

Объяснение

`CT`:	Круговая интерполяция с тангенциальным переходом	
	Активность:	модально
`X... Y... Z...`:	Конечная точка окружности в декартовых координатах В зависимости от текущей действительной настройки указания размеров `G90`/`G91` или `...=AC(...)` / `...=IC(...)` координаты конечной точки окружности интерпретируются либо в абсолютном размере, либо в составном размере.	

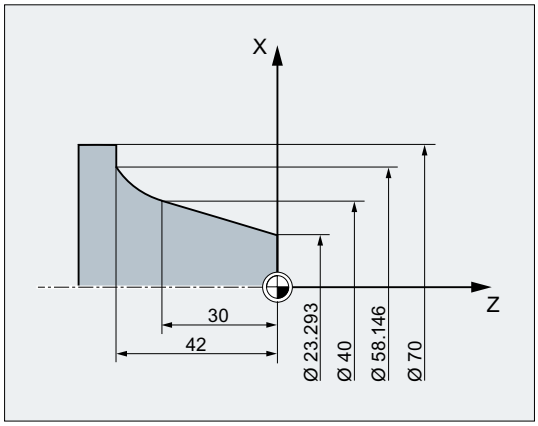
Примеры

Пример 1: Фрезерная обработка



Программный код	Комментарий
N10 G0 Z100	
N20 G17 T1 M6	
N30 G0 X0 Y0 Z2 M3 S300 D1	
N40 Z-5 F1000	; Подача инструмента.
N50 G41 X30 Y25 G1 F1000	; Включение коррекции радиуса инструмента.
N60 Y35	; Фрезерование контура.
N70 X60 Y70	
N80 CT X80 Y55	**; Программирование окружности с тангенциальным переходом.**
N90 X90 Y35	
N100 G1 X100	
N110 Y25	
N120 X30	
N130 G0 G40 X0 Y0	; Отключение коррекции радиуса инструмента.
N140 Z100	; Отвод инструмента.
N140 M30	

Пример 2: Токарная обработка



Программный код	Комментарий
...	
N110 G1 X23.293 Z0 F10	
N115 X40 Z-30 F0.2	
N120 CT X58.146 Z-42	**; Программирование окружности с тангенциальным переходом.**
N125 G1 X70	
...	

Дополнительная информация

Сплайны

У сплайнов тангенциальное направление определяется через прямую через последние две точки. Это направление у сплайнов A и C при активной ENAT или EAUTO в общем и целом не идентично направлению в конечной точке сплайна.

Переход сплайна B всегда является тангенциальным, при этом направление касательной определено как у сплайнов A и C и активной ETAN.

Смена фрейма

Если между определяющим касательную кадром и кадром CT происходит смена фрейма, то касательная подчиняется этой смене.

Граничная ситуация

Если продолжение стартовой касательной проходит через конечную точку, то вместо окружности создается прямая (граничная ситуация окружности с бесконечным радиусом). В этом специальном случае либо нельзя программировать TURN, либо должно действовать TURN=0.

Примечание

При приближении к этой граничной ситуации получаются окружности с радиусом любого размера, так что при TURN, не равном 0, как правило, обработка прерывается с ошибкой из-за нарушения предела программного обеспечения.

Положение плоскости окружности

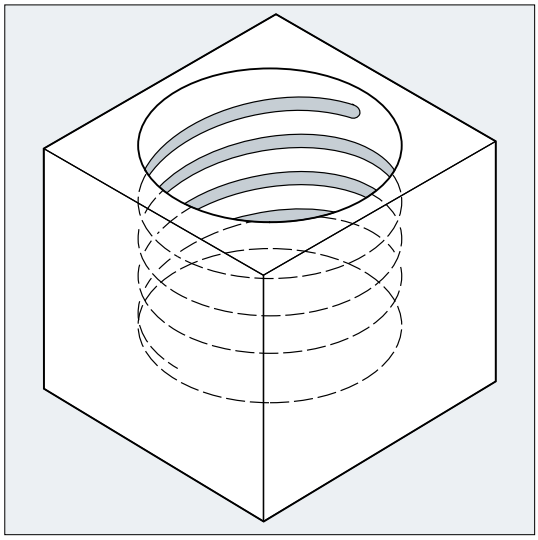
Положение плоскости окружности зависит от активной плоскости (G17-G19).

Если касательная предыдущего кадра находится не в активной плоскости, то используется ее проекция в активную плоскость.

Если стартовая и конечная точка имеют разные позиционные компоненты вертикально к активной плоскости, то вместо окружности создается спираль.

3.9.7 Винтовая интерполяция (G2/G3, TURN)

Винтовая интерполяция (спиральная интерполяция) позволяет, например, выполнять резьбу или смазочные канавки.



При винтовой интерполяции накладываются и параллельно выполняются два движения:

- ровное круговое движение, на которое
- накладывается вертикальное линейное движение.

Синтаксис

```
G2/G3 X… Y… Z… I… J… K… TURN=
G2/G3 X… Y… Z… I… J… K… TURN=
G2/G3 AR=… I… J… K… TURN=
G2/G3 AR=… X… Y… Z… TURN=
G2/G3 AP… RP=… TURN=
```

Объяснение

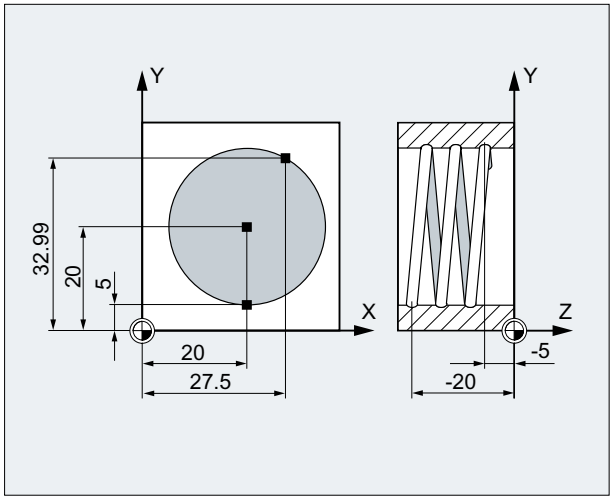
G2:	Движение по круговой траектории по часовой стрелке
G3:	Движение по круговой траектории против часовой стрелки
X Y Z:	Конечная точка в декартовых координатах

I J K:	Центр окружности в декартовых координатах
AR:	Апертурный угол
TURN=:	Количество дополнительных проходов круга в диапазоне от 0 до 999
AP=:	Полярный угол
RP=:	Полярный радиус

Примечание

G2 и G3 действуют модально.

Пример



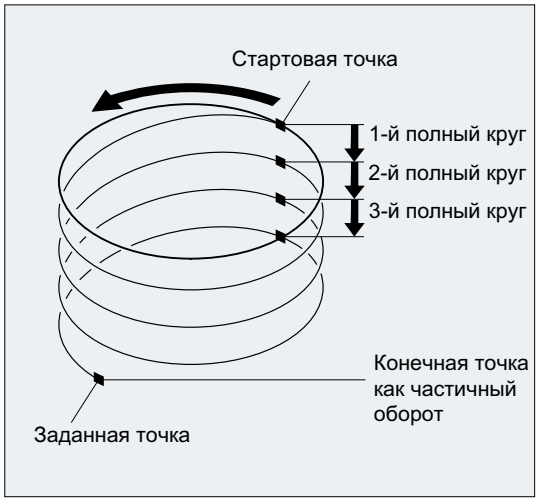
Программный код	Комментарий
N10 G17 G0 X27.5 Y32.99 Z3	; Подвод в стартовую позицию.
N20 G1 Z-5 F50	; Подача инструмента.
N30 G3 X20 Y5 Z-20 I=AC(20) J=AC(20) TURN=2	; Винтовая линия с данными: Выполнить от стартовой позиции 2 полных круга, затем переход в конечную точку.
N40 M30	; Конец программы.

Дополнительная информация

Последовательность движений

1. Переход к стартовой точке
2. С TURN= исполнение запрограммированных полных кругов.
3. Подвод к конечной точки окружности, например, как частичный оборот.
4. Исполнение пунктов 2 и 3 на глубину подачи.

Из количества полных кругов плюс запрограммированной конечной точки окружности (выполненными на глубину подачи) получается шаг, с которым должна быть изготовлена винтовая линия.



Программирование конечной точки винтовой интерполяции

Подробное описание параметров интерполяции см. круговую интерполяцию.

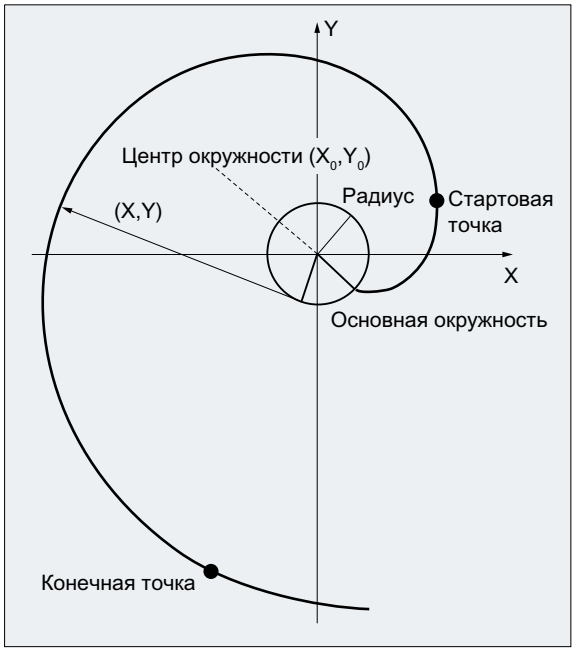
Запрограммированная подача

При винтовой интерполяции рекомендуется указывать запрограммированные процентовки подачи (`CFC`). С помощью `FGROUP` можно установить, какие оси должны перемещаться с запрограммированной подачей. Подробности см. главу "Параметры траектории".

3.9.8 Эвольвентная интерполяция (INVCW, INVCCW)

Эвольвента окружности это кривая, описываемая от конечной точки жестко натянутой, развернутой от окружности нити.

Эвольвентная интерполяция позволяет создавать траектории вдоль эвольвенты. Она выполняется в плоскости, в которой определена основная окружность, и проходит от запрограммированной стартовой точки до запрограммированной конечной точки.



Возможно два способа программирования конечной точки:

1. Напрямую через декартовы координаты
2. Косвенно через указание апертурного угла (сравни программирование апертурного угла при программировании окружности)

Если стартовая и конечная точка не лежат в плоскости основной окружности, то, аналогично винтовой интерполяции у окружностей, получается наложение на кривую в пространстве.

При дополнительном вводе ходов траектории вертикально к активной плоскости (можно сравнить с винтовой интерполяцией для окружностей) эвольвента может перемещаться в пространстве.

Синтаксис

```
INVCW X... Y... Z... I... J... K... CR=...
INVCCW X... Y... Z... I... J... K... CR=...
INVCW I... J... K... CR=... AR=...
INVCCW I... J... K... CR=... AR=...
```

Объяснение

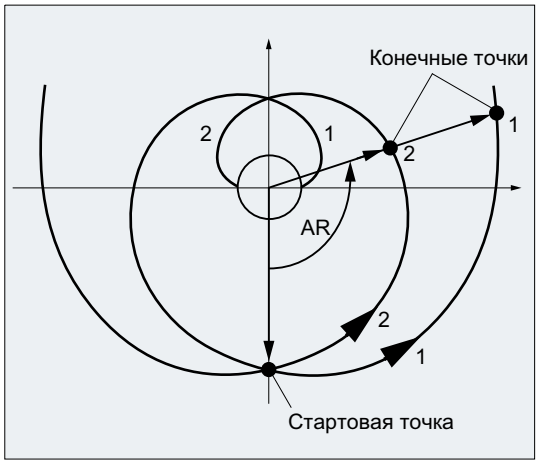
`INVCW`:	Команда движения по эвольвенте по часовой стрелке
`INVCCW`:	Команда движения по эвольвенте против часовой стрелки
`X... Y... Z...`:	Прямое программирование конечной точки в декартовых координатах

`I... J... K...:`	Параметры интерполяции для описания центра основной окружности в декартовых координатах **Примечание:** Координаты указываются относительно стартовой точки эвольвенты.	
`CR=...:`	Радиус основной окружности	
`AR=...:`	Косвенное программирование конечной точки через указание апертурного угла (угла поворота) Началом отсчета апертурного угла является прямая от центра окружности к стартовой точке.	
	`AR` > 0:	Траектория на эвольвенте движется **от основной окружности**.
	`AR` < 0:	Траектория на эвольвенте движется **к основной окружности**. Для AR < 0 максимальный угол поворота ограничен тем, что конечная точка всегда должна находиться вне основной окружности.

Косвенное программирование конечной точки через указание апертурного угла

ВНИМАНИЕ
Апертурный угол не определен При косвенном программировании конечной точки через указание апертурного угла AR необходимо учитывать знак угла, т. к. смена знака может привести к другой эвольвенте и тем самым к другой траектории.

Это должен пояснить следующий пример:

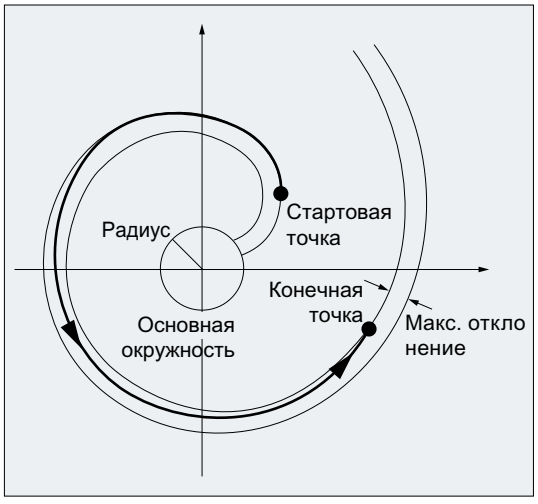


У эвольвент 1 и 2 совпадают параметры радиуса и центра основной окружности, а также стартовой точки и направления вращения (INVCW / INVCCW). Единственным различием является знак апертурного угла:

- При AR > 0 перемещается траектория на эвольвенте 1 и выполняется подвод к конечной точке 1.
- При AR < 0 перемещается траектория на эвольвенте 2 и выполняется подвод к конечной точке 2.

Граничные условия

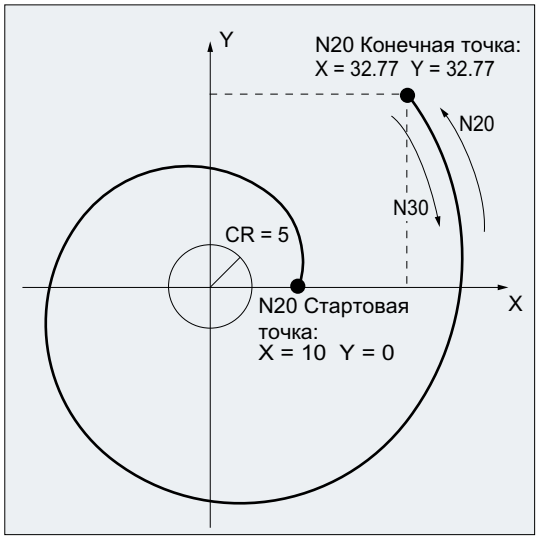
- Как стартовая точка, так и конечная точка, должны находится вне поверхности основной окружности эвольвенты (окружность с радиусом CR вокруг определенного через I, J, K центра). Если это условие не соблюдается, то выводится ошибка и обработка программы отменяется.
- Обе возможности программирования конечной точки (напрямую через декартовы координаты или косвенно через указание апертурного угла) являются взаимоисключающими. Поэтому в кадре может использоваться только одна из этих двух возможностей программирования.
- Если запрограммированная конечная точка находится не точно на определенной через стартовую точку и основную окружность эвольвенте, то осуществляется интерполяция между обеими эвольвентами, определенными через стартовую или конечную точку (см. рисунок ниже).



Максимальное отклонение конечной точки устанавливается через машинные данные (→ Изготовитель станка!). Если отклонение запрограммированной конечной точки в радиальном направлении больше, чем установленное через эти MD значение, то выводится ошибка и выполнение программы отменяется.

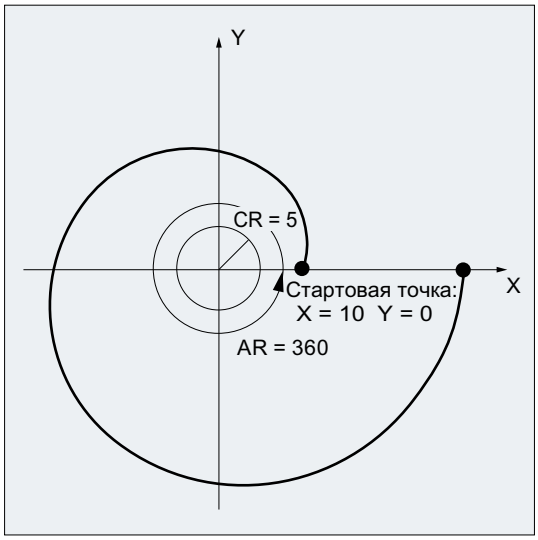
Примеры

Пример 1: Левовращающаяся эвольвента от стартовой точки к запрограммированной конечной точке и как правовращающаяся эвольвента обратно



Программный код	Комментарий
`N10 G1 X10 Y0 F5000`	; Подвод в стартовую позицию.
`N15 G17`	; Выбор плоскости `X/Y` как рабочей плоскости.
`N20 INVCCW X32.77 Y32.77 CR=5 I-10 J0`	; Эвольвента против часовой стрелки, конечная точка в декартовых координатах.
`N30 INVCW X10 Y0 CR=5 I-32.77 J-32.77`	; Эвольвента по часовой стрелке, стартовая точка это конечная точка из `N20`, новая конечная точка это стартовая точка из `N20`, новый центр окружности относится к новой стартовой точке и равен старому центру окружности.
...	

Пример 2: Левовращающаяся эвольвента с косвенным программированием конечной точки через указание апертурного угла



Программный код	Комментарий
`N10 G1 X10 Y0 F5000`	; Подвод в стартовую позицию.
`N15 G17`	; Выбор плоскости `X/Y` как рабочей плоскости.
`N20 INVCCW CR=5 I-10 J0 AR=360`	; Эвольвента против часовой стрелки и от основной окружности (т.к. указан положительный угол) с полным оборотом (360 градусов).
...	

3.9.9 Линии контура

3.9.9.1 Программирование линии контура

Функция

Программирование линии контура служит для быстрого ввода простых контуров.

Могут программироваться линии контура с 1, 2, 3 или более точками с переходными элементами "фаска" или "закругление" через указание декартовых координат и / или углов (ANG или ANG1 и ANG2).

В кадрах, описывающих линии контура, могут использоваться любые другие адреса ЧПУ, например, буквы адреса для других осей (отдельные оси или ось вертикально к плоскости обработки), данные вспомогательных функций, G-команды, скорости и т. д.

Примечание

Контурный вычислитель

Простое программирование линии контура возможно и с помощью контурного вычислителя. При этом речь идет об инструменте интерфейса, который обеспечивает программирование и графическое представление простых и сложных контуров деталей. Запрограммированные через контурный вычислитель контуры передаются в программу обработки детали.

Дополнительная информация: Справочник пользователя

Параметрирование

Идентификаторы для угла, радиуса и фаски определяются через машинные данные:

MD10652 $MN_CONTOUR_DEF_ANGLE_NAME (имя угла для линий контура)

MD10654 $MN_RADIUS_NAME (имя радиуса для линий контура)

MD10656 $MN_CHAMFER_NAME (имя фаски для линий контура)

Примечание

См. указания изготовителя станка.

3.9.9.2 Линии контура: Прямая

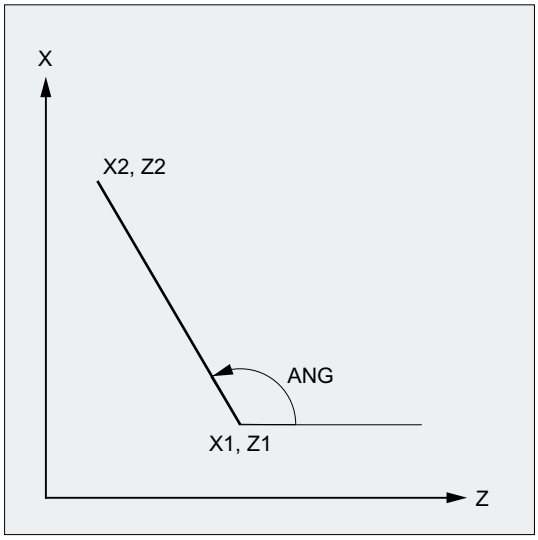
Примечание

В описании ниже предполагается, что:

- G18 активна (⇒ активной рабочей плоскостью является плоскость Z/X).
  (Но программирование линий контура без ограничений возможно и при G17 или G19 .)
- для угла, радиуса и фаски определены следующие идентификаторы:
  - ANG (угол)
  - RND (радиус)
  - CHR (фаска)

Конечная точка прямой определяется следующими данными:

- Угол ANG
- **Одна** декартова координата конечной точки (X2 или Z2)



ANG: Угол прямой
X1, Z1: Начальные координаты
X2, Z2: Координаты конечной точки прямой

Синтаксис

```
X... ANG=...
Z... ANG=...
```

Значение

`X...`:	Координата конечной точки в направлении X
`Z...`:	Координата конечной точки в направлении Z
`ANG`:	Идентификатор для программирования угла Указанное значение (угол) относится к абсциссе активной рабочей плоскости (ось Z при `G18`).

Пример

Программный код	Комментарий
`N10 X5 Z70 F1000 G18`	; Подвод в стартовую позицию
`N20 X88.8 ANG=110`	; Прямая с указанием угла
`N30 ...`	

или:

Программный код	Комментарий
`N10 X5 Z70 F1000 G18`	; Подвод в стартовую позицию
`N20 Z39.5 ANG=110`	; Прямая с указанием угла
`N30 ...`	

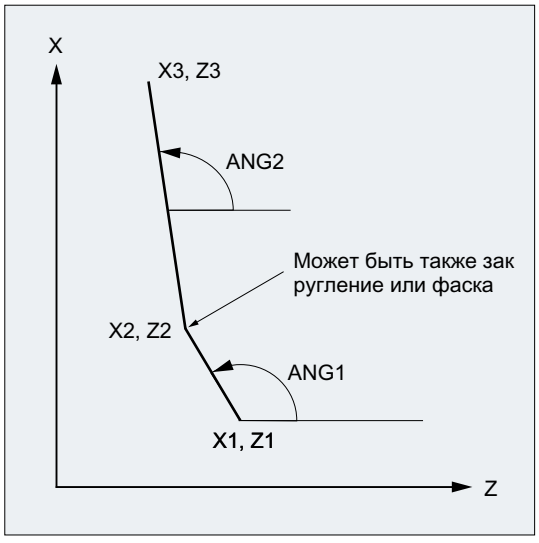
3.9.9.3 Линии контура: Две прямые

Примечание

В описании ниже предполагается, что:

- G18 активна (⇒ активной рабочей плоскостью является плоскость Z/X).
  (Но программирование линий контура без ограничений возможно и при G17 или G19 .)
- Для угла, радиуса и фаски определены следующие идентификаторы:
  – ANG (угол)
  – RND (радиус)
  – CHR (фаска)

Конечная точка первой прямой может быть запрограммирована через указание декартовых координат или через указание угла обоих прямых. Конечная точка второй прямой всегда должны быть запрограммирована декартово. Точка пересечения обеих прямых может быть выполнена как угол, закругление или как фаска.



ANG1:	Угол первой прямой
ANG2:	Угол второй прямой
X1, Z1:	Начальные координаты первой прямой
X2, Z2:	Координаты конечной точки первой прямой или начальные координаты второй прямой
X3, Z3:	Координаты конечной точки второй прямой

Синтаксис

Программирование конечной точки первой прямой через указание угла

- Угол как переход между прямыми:

```
ANG=…
X… Z… ANG=…
```

- Закругление как переход между прямыми:

```
ANG=… RND=...
X… Z… ANG=…
```

- Фаска как переход между прямыми:

```
ANG=… CHR=...
X… Z… ANG=…
```

Программирование конечной точки первой прямой через указание координат

- Угол как переход между прямыми:

```
X… Z…
X… Z…
```

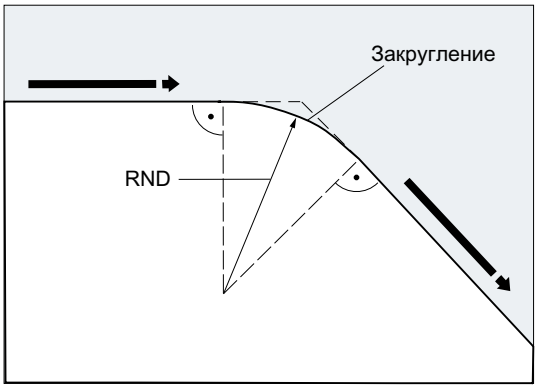
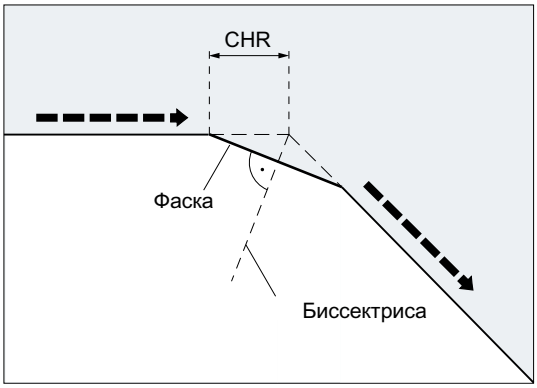
- Закругление как переход между прямыми:

```
X… Z… RND=...
X… Z…
```

- Фаска как переход между прямыми:

```
X… Z… CHR=...
X… Z…
```

Значение

ANG=... :	Идентификатор для программирования угла Указанное значение (угол) относится к абсциссе активной рабочей плоскости (ось Z при G18).
RND=... :	Идентификатор для программирования закругления Указанное значение соответствует радиусу закругления: 
CHR=... :	Идентификатор для программирования фаски Указанное значение соответствует ширине фаски в направлении движения: 
X... :	Координаты в направлении X
Z... :	Координаты в направлении Z

Примечание

Дополнительную информацию по программированию фаски или закругления см. " Фаска, закругление (CHF, CHR, RND, RNDM, FRC, FRCM) (Страница 254) ".

Пример

Программный код	Комментарий
`N10 X10 Z80 F1000 G18`	; Подвод в стартовую позицию.
`N20 ANG=148.65 CHR=5.5`	; Прямая с указанием угла и фаски.
`N30 X85 Z40 ANG=100`	; Прямая с указанием угла и конечной точки.
`N40 ...`	

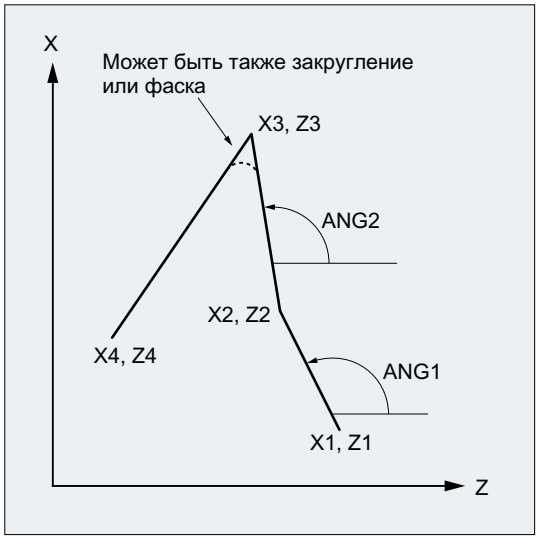
3.9.9.4 Линии контура: Три прямые

Примечание

В описании ниже предполагается, что:

- G18 активна (⇒ активной рабочей плоскостью является плоскость Z/X).
  (Но программирование линий контура без ограничений возможно и при G17 или G19 .)
- Для угла, радиуса и фаски определены следующие идентификаторы:
  – ANG (угол)
  – RND (радиус)
  – CHR (фаска)

Конечная точка первой прямой может быть запрограммирована через указание декартовых координат или через указание угла обоих прямых. Конечная точка второй и третьей прямой всегда должны быть запрограммирована декартово. Точка пересечения прямых может быть выполнена как угол, закругление или как фаска.



ANG1:	Угол первой прямой
ANG2:	Угол второй прямой
X1, Z1:	Начальные координаты первой прямой
X2, Z2:	Координаты конечной точки первой прямой или начальные координаты второй прямой
X3, Z3:	Координаты конечной точки второй прямой или начальная точка третьей прямой
X4, Z4:	Координаты конечной точки третьей прямой

Примечание

Объясняемое здесь программирование для 3-точечной линии контура может быть произвольно продолжено для линий контура более чем из трех точек.

Синтаксис

Программирование конечной точки первой прямой через указание угла

- Угол как переход между прямыми:

```
ANG=…
X… Z… ANG=…
X… Z…
```

- Закругление как переход между прямыми:

```
ANG=… RND=...
X… Z… ANG=… RND=...
X… Z…
```

- Фаска как переход между прямыми:

```
ANG=… CHR=...
X… Z… ANG=… CHR=...
X… Z…
```

Программирование конечной точки первой прямой через указание координат

- Угол как переход между прямыми:

```
X… Z…
X… Z…
X… Z…
```

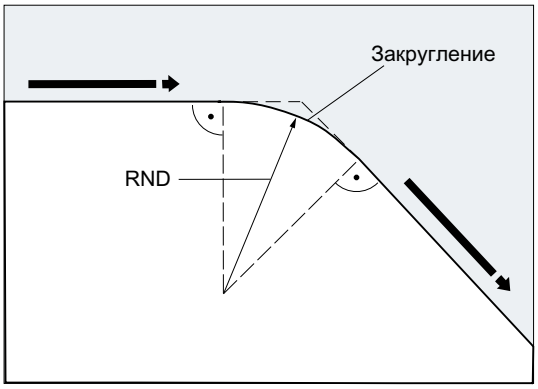
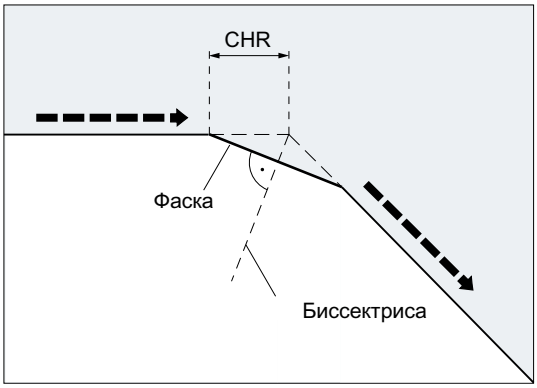
- Закругление как переход между прямыми:

```
X… Z… RND=...
X… Z… RND=...
X… Z…
```

- Фаска как переход между прямыми:

```
X… Z… CHR=...
X… Z… CHR=...
X… Z…
```

Значение

ANG=... :	Идентификатор для программирования угла Указанное значение (угол) относится к абсциссе активной рабочей плоскости (ось Z при G18).
RND=... :	Идентификатор для программирования закругления Указанное значение соответствует радиусу закругления: 
CHR=... :	Идентификатор для программирования фаски Указанное значение соответствует ширине фаски в направлении движения: 
X... :	Координаты в направлении X
Z... :	Координаты в направлении Z

Примечание

Дополнительную информацию по программированию фаски или закругления см. " Фаска, закругление (CHF, CHR, RND, RNDM, FRC, FRCM) (Страница 254) ".

Пример

Программный код	Комментарий
`N10 X10 Z100 F1000 G18`	; Подвод в стартовую позицию
`N20 ANG=140 CHR=7.5`	; Прямая с указанием угла и фаски
`N30 X80 Z70 ANG=95.824 RND=10`	; Прямая на промежуточную точку с указанием угла и закругления
`N40 X70 Z50`	; Прямая на конечную точку

3.9.9.5 Линии контура: Программирование конечной точки с углом

Функция

Если в кадре ЧПУ появляется буква адреса A, до дополнительно не может быть запрограммировано ни одной, может быть запрограммировано одна или обе оси активной плоскости.

Количество запрограммированных осей

- Если **ни одна из осей** активной плоскости не запрограммирована, то речь идет либо о первом, либо о втором кадре линии контура, которая состоит из двух кадров. Если это второй кадр такой линии контура, то это означает, что стартовая и конечная точка в активной плоскости идентичны. Тогда в любом случае линия контура состоит из одного движения вертикально к активной плоскости.
- Если запрограммирована **точно одна ось** активной плоскости, то речь идет либо об отдельной прямой, конечная точка которой однозначно определена углом и запрограммированной декартовой координатой, либо о втором кадре состоящей из двух кадров линии контура. Во втором случае отсутствующая координата устанавливается равной последней достигнутой (модальной) позиции.
- Если запрограммированы **две оси** активной плоскости, то речь идет о втором кадре линии контура, состоящей из двух кадров. Если перед актуальным кадром не стоял кадр с программированием угла без запрограммированных осей активной плоскости, то такой кадр не допускается.

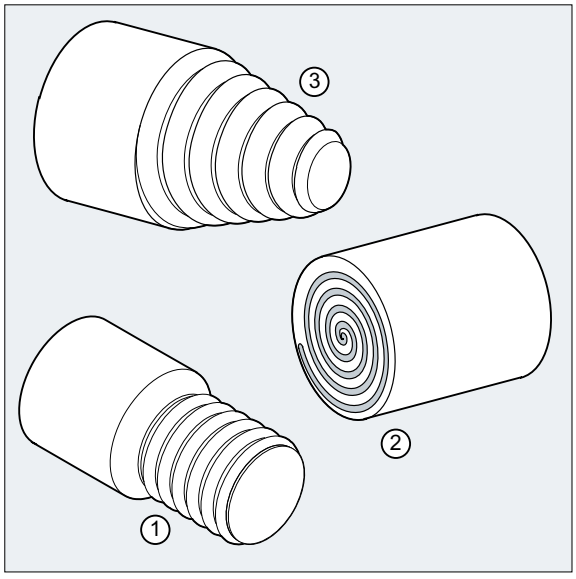
Угол A может программироваться только при линейной или сплайн-интерполяции.

3.9.10 Резьбонарезание

3.9.10.1 Резьбонарезание с постоянным шагом (G33, SF)

С помощью `G33` можно изготовить резьбу с постоянным шагом:

- Цилиндрическая резьба ①
- Спиральная резьба ②
- Коническая резьба ③

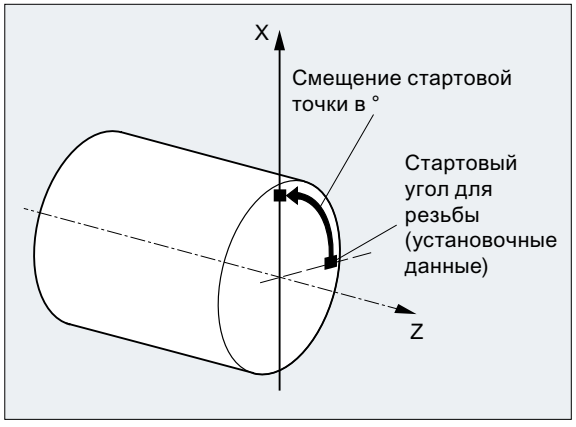


Примечание

Техническим условием для резьбонарезания с G33 является шпиндель с регулируемой скоростью с системой измерения перемещения.

Многозаходная резьба

Многозаходные резьбы (резьбы со смещенными проходами резца) могут изготовляться посредством указания смещения стартовой точки. Программирование осуществляется в кадре G33 по адресу SF.

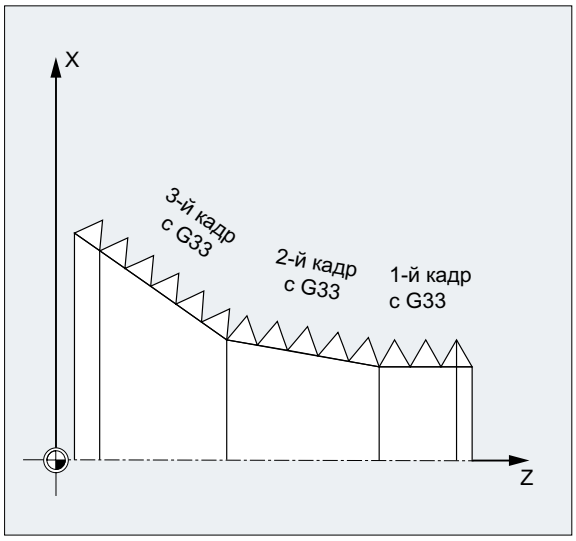


Примечание

Если смещение стартовой точки не указано, то используется определенный в установочных данных "Стартовый угол для резьбы".

Цепочка резьб

С помощью нескольких, последовательно запрограммированных кадров G33можно изготовить цепочку резьб:



Примечание

С помощью режима управления траекторией G64 посредством опережающего управления скоростью кадры связываются друг с другом таким образом, что не возникает скачков скорости.

Направление вращения резьбы

Направление вращения резьбы определяется через направление вращения шпинделя:

- правое вращение с M3 создает правую резьбу
- левое вращение с M4 создает левую резьбу

Синтаксис

Цилиндрическая резьба:

```
G33 Z… K…
G33 Z… K… SF=…
```

Спиральная резьба:

```
G33 X… I…
G33 X… I… SF=…
```

Коническая резьба:

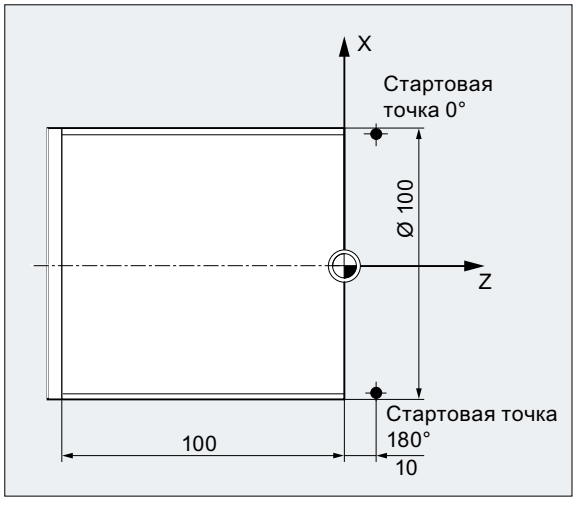
```
G33 X… Z… K…
G33 X… Z… K… SF=…
G33 X… Z… I…
G33 X… Z… I… SF=…
```

Значение

G33:	Команда для резьбонарезания с постоянным шагом	
X... Y... Z...:	Конечная(ые) точка(и) в декартовых координатах	
I...:	Шаг резьбы в направлении X	
J...:	Шаг резьбы в направлении Y	
K...:	Шаг резьбы в направлении Z	
Z:	Продольная ось	
X:	Поперечная ось	
Z... K...:	Длина и шаг для цилиндрической резьбы	
X... I...:	Диаметр и шаг для спиральной резьбы	
I... или K...	Шаг для конической резьбы Указание (I... или K...) зависит от угла конуса:	
	< 45°:	Шаг резьбы указывается с K... (шаг резьбы в продольном направлении).
	> 45°:	Шаг резьбы указывается с I... (шаг резьбы в поперечном направлении).
	= 45°:	Шаг резьбы может быть указан с I... или K....
SF=...:	Смещение стартовой точки (необходимо только для многозаходных резьб!) Смещение стартовой точки указывается как абсолютная угловая позиция.	
	Диапазон значений:	0.0000 до 359.999 градусов

Примеры

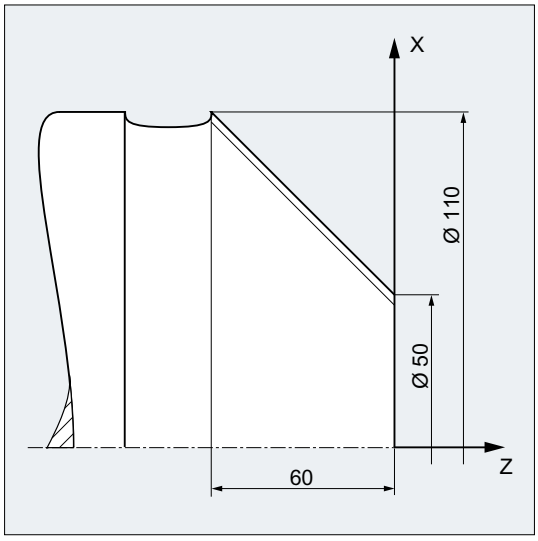
Пример 1: Двухзаходная цилиндрическая резьба со смещением стартовой точки 180°



Программный код	Комментарий
N10 G1 G54 X99 Z10 S500 F100 M3	; Смещение нулевой точки, подвод в стартовую точку, включение шпинделя.

Программный код	Комментарий
N20 G33 Z-100 K4	; Цилиндрическая резьба: конечная точка в Z
N30 G0 X102	; Отвод в стартовую позицию.
N40 G0 Z10	
N50 G1 X99	
N60 G33 Z-100 K4 SF=180	; 2. проход резца: смещение стартовой точки 180°
N70 G0 X110	; Отвод инструмента.
N80 G0 Z10	
N90 M30	; Конец программы.

Пример 2: Коническая резьба с углом меньше 45°

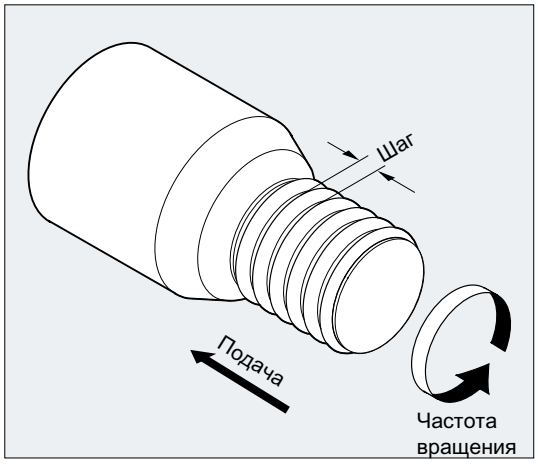


Программный код	Комментарий
N10 G1 X50 Z0 S500 F100 M3	; Подвод в стартовую точку, включение шпинделя.
N20 G33 X110 Z-60 K4	; Коническая резьба: Конечная точка в X и Z, указание шага резьбы с K... в направлении Z (т.к. угол конуса < 45°).
N30 G0 Z0 M30	; Отвод, конец программы.

Дополнительная информация

Подача для резьбонарезания с G33

СЧПУ вычисляет из запрограммированного числа оборотов шпинделя и шага резьбы необходимую подачу, с которой токарный резец будет перемещаться по длине резьбы в продольном и/или поперечном направлении. Подача F не учитывается для G33, ограничение до максимальной осевой скорости (ускоренный ход) контролируется СЧПУ.



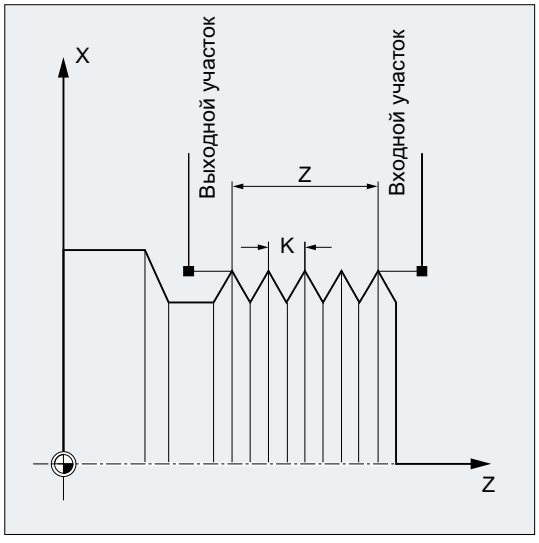
Цилиндрическая резьба

Цилиндрическая резьба описывается через:

- длину резьбы
- шаг резьбы

Длина резьбы вводится с одной из декартовых координат X, Y или Z в абсолютном или составном размере (на токарных станках преимущественно в направлении Z). Дополнительно учитываются входные и выходные участки, на которых подача увеличивается или уменьшается.

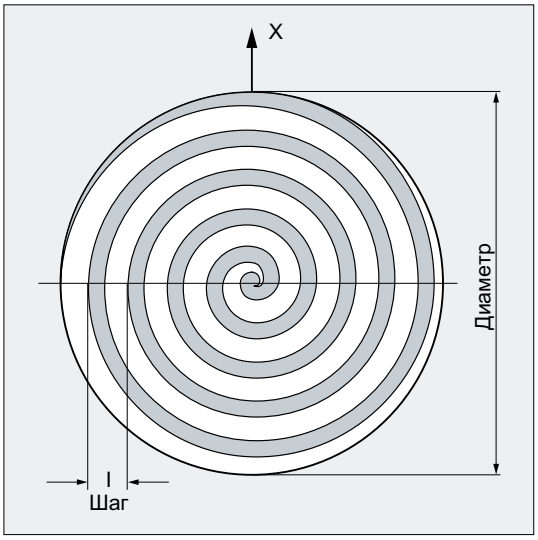
Шаг резьбы вводиться по адресам `I`, `J`, `K` (у токарных станков преимущественно с `K`).



Спиральная резьба

Спиральная резьба описывается через:

- диаметр резьбы (преимущественно в направлении X)
- шаг резьбы (преимущественно с `I`)



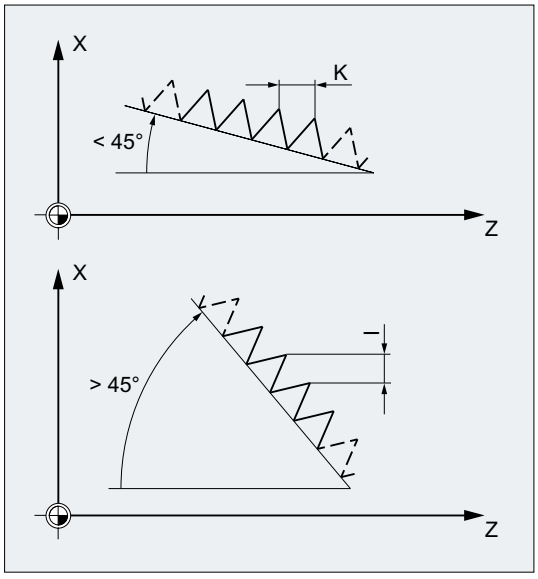
Коническая резьба

Коническая резьба описывается через:

- конечную точку в продольном и поперечном направлении (контур конуса)
- шаг резьбы

Контур конуса вводится в декартовых координатах X, Y, Z в основном или составном размере, при обработке на токарных станках преимущественно в направлении X и Z. Дополнительно учитываются входные и выходные участки, на которых подача увеличивается или уменьшается.

Указание шага зависит от угла конуса (угол между продольной осью и боковой поверхностью конуса):



3.9.10.2 Резьбонарезание с увеличивающимся или уменьшающимся шагом (G34, G35)

Команды G34 и G35 расширили функционал G33: появилась возможность дополнительного программирования изменения шага резьбы по адресу F. В случае G34 это приводит к линейному увеличению, в случае G35 — к линейному уменьшению шага резьбы. Таким образом, команды G34 и G35 могут использоваться для изготовления самонарезающихся резьб.

Синтаксис

Цилиндрическая резьбы с увеличивающимся шагом:
```
G34 Z… K… F...
```
Цилиндрическая резьбы с уменьшающимся шагом:
```
G35 Z… K… F...
```
Спиральная резьбы с увеличивающимся шагом:
```
G34 X… I… F...
```
Спиральная резьбы с уменьшающимся шагом:
```
G35 X… I… F...
```
Коническая резьбы с увеличивающимся шагом:
```
G34 X… Z… K… F...
G34 X… Z… I… F...
```
Коническая резьбы с уменьшающимся шагом:
```
G35 X… Z… K… F...
G35 X… Z… I… F...
```

Объяснение

`G34`:	Команда для резьбонарезания с линейно **увеличивающимся**шагом	
`G35`:	Команда для резьбонарезания с линейно **уменьшающимся**шагом	
`X... Y... Z...`:	Конечная(ые) точка(и) в декартовых координатах	
`I...`:	Шаг резьбы в направлении X	
`J...`:	Шаг резьбы в направлении Y	
`K...`:	Шаг резьбы в направлении Z	
`F...`:	Изменение шага резьбы Если начальный и конечный шаг резьбы известны, то программируемое изменение шага резьбы может быть вычислено по следующей формуле: $F = \frac{k_e^2 - k_a^2}{2 * l_G}$ [мм/об²] Где:	
	k_e:	шаг резьбы (шаг резьбы координаты заданной точки) [мм/об]
	k_a:	начальный шаг резьбы (запрограммирован в I, J или K) [мм/об]
	l_G:	длина резьбы [мм]

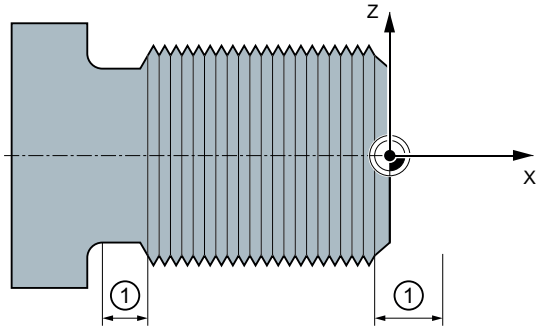
Пример

Программный код	Комментарий
N1608 M3 S10	; Шпиндель вкл.
N1609 G0 G64 Z40 X216	; Подвод в стартовую точку.
N1610 G33 Z0 K100 SF=R14	; Резьбонарезание с постоянным шагом (100 мм/об)
N1611 G35 Z-200 K100 F17.045455	; Уменьшение шага: 17.0454 мм/U2
	Шаг на конце кадра: 50мм/об
N1612 G33 Z-240 K50	; Проход кадра резьбы без рывка.
N1613 G0 X218	
N1614 G0 Z40	
N1615 M17	

3.9.10.3 Запрограммированный входной и выходной участок при G33, G34, G35 (DITS, DITE)

Адреса DITS и DITE позволяют задать в программе обработки детали входной и выходной участок резьбы.

Ось резьбы ускоряется или затормаживается в пределах заданного участка.



① Входной или выходной участок, в зависимости от направления обработки

Короткий входной участок

Из-за буртика на входе резьбы недостаточно места для рампы разгона инструмента. Поэтому ее необходимо сократить через DITS.

Короткий выходной участок

Из-за буртика на выходе резьбы недостаточно места для рампы торможения инструмента, из-за чего существует опасность столкновения детали с резцом. Более короткую рампу торможения инструмента можно задать через DITE. Но из-за инертности механики риск столкновения сохраняется.

Способ устранения: запрограммировать более короткую резьбу, уменьшить число оборотов шпинделя.

Примечание

DITE действует на конце резьбы как интервал перешлифовки. Тем самым обеспечивается плавное изменение движения оси.

Последствия

Запрограммированный входной и выходной участок действует на траекторию исключительно ростом ускорения. Если один из этих участков задается более длинным, чем требуется для оси резьбы с активным ускорением, то ось резьбы ускоряется или затормаживается с максимальным ускорением.

Синтаксис

DITS=<значение> DITE=<значение>

Значение

DITS:	Определить входной участок резьбы
DITE:	Определить выходной участок резьбы
<значение>:	В DITS и DITE программируются исключительно пути, но не позиции. Запрограммированный входной/выходной участок обрабатывается в соответствии с актуальной установкой размеров (дюймовая, метрическая).

Пример

Программный код	Комментарий
...	
N40 G90 G0 Z100 X10 SOFT M3 S500	
N50 G33 Z50 K5 SF=180 DITS=1 DITE=3	; Начало перешлифовки на Z=53.
N60 G0 X20	

Дополнительная информация

SD42010 $SC_THREAD_RAMP_DISP

При установке кадра в главный ход при помощи команды DITS и/или DITE запрограммированный входной или выходной участок передается в установочные данные SD42010 $SC_THREAD_RAMP_DISP:

- SD42010 $SC_THREAD_RAMP_DISP[**0**] = запрограммированное значение DITS
- SD42010 $SC_THREAD_RAMP_DISP[**1**] = запрограммированное значение DITE

Если перед кадром резьбы или в первом кадре резьбы входной или выходной участок не запрограммированы, используется соответствующее текущее значение установочных данных.

Поведение после сброса (Reset) канала / группы режимов работы (ГРР) / конца программы

Перезаписанные посредством DITS и/или DITE значения SD42010 действуют и после сброса (Reset) канала / группы режимов работы (ГРР) / конца программы.

Поведение после программной перезагрузки

В случае программной перезагрузки установочные данные сбрасываются до значений, действовавших до перезаписи посредством DITS и/или DITE (стандартное поведение).

Если после программной перезагрузки должны действовать запрограммированные посредством DITS и DITE значения, установочные данные SD42010 $SC_THREAD_RAMP_DISP должны быть указаны в машинных данных MD10710 $MN_PROG_SD_RESET_SAVE_TAB:

MD10710 $MN_PROG_SD_RESET_SAVE_TAB[<n>] = 42010

Поведение при слишком маленьком входном и/или выходном участке

При слишком маленьком входном и/или выходном участке, ось резьбы ускоряется сильнее, чем это предусмотрено в проектировании. Поэтому возникает перегрузка оси по ускорению.

Для входа резьбы в этом случае выводится ошибка 22280 "Слишком короткий запрограммированный входной участок" (при соответствующем проектировании в MD 11411 ENABLE_ALARM_MASK). Аварийное сообщение носит сугубо информативный характер и не влияет на выполнение программы обработки детали.

3.9.10.4 Быстрый обратный ход при резьбонарезании (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN)

Функция "Быстрый отвод при резьбонарезании (G33)" обеспечивает неразрушающее прерывание резьбонарезания в следующих ситуациях:

- NC-Stop, вызванный через DB21, ... DBX7.3
- NC-Stop, вызванный не явно через аварийное сообщение
- Включение быстрого входа (Страница 564)

Движение отвода может быть запрограммировано через:

- Путь отвода и направление отвода (относительные)
- Позиция отвода (абсолютная)

Примечание

Сигналы NC-Stop

Следующие сигналы NC-Stop не запускают быстрого отвода при резьбонарезании:

- DB21, ... DBX3.4 (NC-Stop оси плюс шпиндели)
- DB21, ... DBX7.2 (NC-Stop на границе кадра)

Нарезание внутренней резьбы

Функция "Быстрый отвод" **не** может использоваться для **нарезания** внутренней резьбы (G331/G332).

Синтаксис

Разрешить быстрый отвод, движение отвода через путь отвода и направление отвода:
`G33 ... LFON DILF=<значение> LFTXT/LFWP ALF=<значение>`

Разрешить быстрый отвод, движение отвода через позицию отвода:

```
POLF[<идентификатор оси>]=<значение> LFPOS
POLFMASK/POLFMLIN(<имя оси1>,<имя оси2>,...)
G33 ... LFON
```

Блокировать быстрый отвод при резьбонарезании:
`LFOF`

Объяснение

`LFON`:	Разрешить быстрый отвод при резьбонарезании (`G33`)	
`LFOF`:	Блокировать быстрый отвод при резьбонарезании (`G33`)	
`DILF=` :	Определить длину пути отвода	
	Предустановленное через конфигурирование MD (MD21200 $MC_LIFTFAST_DIST) значение может быть изменено в программе обработки детали через программирование `DILF`. **Примечание:** После NC-RESET всегда активно сконфигурированное значение MD.	
`LFTXT` `LFWP`:	Направление отвода в комбинации с `ALF` управляется G-командами `LFTXT` и `LFWP`.	
	`LFTXT`:	Плоскость, в которой осуществляется движение отвода, вычисляется из касательной к траектории и направления инструмента (стандартная настройка).
	`LFWP`:	Плоскость, в которой осуществляется движение отвода, является активной рабочей плоскостью.
`ALF=` :	В плоскости движения отвода с помощью `ALF` программируется направление с дискретным шагом в градусах.	
	При `LFTXT` для `ALF=1` установлен отвод в направлении инструмента. При `LFWP` направление в рабочей плоскости получается согласно следующему согласованию: • `G17` (плоскость X/Y) `ALF=1` ; отвод в направлении X `ALF=3` ; отвод в направлении Y • `G18` (плоскость Z/X) `ALF=1` ; отвод в направлении Z `ALF=3` ; отвод в направлении X • `G19` (плоскость Y/Z) `ALF=1` ; отвод в направлении Y `ALF=3` ; отвод в направлении Z	
	Дополнительная информация по возможностям программирования с `ALF` представлена в главе "Направление перемещения при быстром отводе от контура (Страница 567)".	
`LFPOS`:	Отвод названной с помощью `POLFMASK` или `POLFMLIN` оси на запрограммированную с `POLF` абсолютную позицию оси.	

POLFMASK:	Разрешение осей (<имя оси1>,<имя оси1>,...) для независимого отвода на абсолютную позицию	
POLFMLIN:	Разрешение осей для отвода на абсолютную позицию в линейной связи **Примечание:** Линейная связь, в зависимости от динамических характеристик всех участвующих осей, не всегда может быть установлена до достижения позиции отвода.	
POLF[]:	Определить абсолютную позицию отвода для указанной в индексе гео-оси или оси станка	
	Активность:	модально
	=<значение>:	Для гео-осей присвоенное значение интерпретируется как позиция в системе координат детали (WCS), для осей станка как позиция в системе координат станка (MCS). Присваивание значений может быть запрограммировано и как указание составного размера: =IC<значение>
<идентификатор оси>:	идентификатор гео-оси или оси станка	

Примечание

LFON или LFOF можно запрограммировать всегда, но обработка осуществляется исключительно при резьбонарезании (G33).

Примечание

POLF с POLFMASK/POLFMLIN не ограничиваются использованием при резьбонарезании.

Примеры

Пример 1: Разрешить быстрый отвод при резьбонарезании

Программный код	Комментарий
N55 M3 S500 G90 G18	; Активная плоскость обработки
...	; Подвод в стартовую позицию
N65 MSG ("Резьбонарезание")	; Подача инструмента
MM_THREAD:	
N67 $AC_LIFTFAST=0	; Сбросить перед началом резьбы.
N68 G0 Z5	
N68 X10	
N70 G33 Z30 K5 LFON DILF=10 LFWP ALF=7	; Разрешить быстрый отвод при резьбонарезании. Путь отвода =10мм Плоскость отвода: Z/X (из-за G18) Направление отвода: -X (с ALF=3: направление отвода +X)
N71 G33 Z55 X15	
N72 G1	; Отмена резьбонарезания.
N69 IF $AC_LIFTFAST GOTOB MM_THREAD	; Если резьбонарезание было прервано.

Программный код	Комментарий
N90 MSG("")	
...	
N70 M30	

Пример 2: Выключение быстрого отвода перед нарезанием внутренней резьбы

Программный код	Комментарий
N55 M3 S500 G90 G0 X0 Z0	
...	
N87 MSG ("Нарезание внутренней резьбы")	
N88 LFOF	; Выключение быстрого отвода перед нарезанием внутренней резьбы.
N89 CYCLE...	; Цикл нарезания внутренней резьбы с G33.
N90 MSG("")	
...	
N99 M30	

Пример 3: Быстрый отвод на абсолютную позицию отвода

При останове траекторная интерполяция X подавляется и вместо нее интерполируется движение с макс. скоростью на позицию POLF[X]. Движение других осей продолжает определяться запрограммированным контуром или шагом резьбы и скоростью шпинделя.

Программный код	Комментарий
N10 G0 G90 X200 Z0 S200 M3	
N20 G0 G90 X170	
N22 POLF[X]=210 LFPOS	
N23 POLFMASK(X)	; Активация (разрешение) быстрого отвода оси X.
N25 G33 X100 I10 LFON	
N30 X135 Z-45 K10	
N40 X155 Z-128 K10	
N50 X145 Z-168 K10	
N55 X210 I10	
N60 G0 Z0 LFOF	
N70 POLFMASK()	; Блокировать отвод для всех осей.
M30	

3.9.10.5 Сферическая резьба (G335, G336)

G-командами G335 и G336 можно нарезать резцом сферическую (= отклоняющуюся от цилиндрической формы) резьбу. Это применяется при обработке очень крупных деталей, провисающих на станке под собственным весом. Нарезание резьбы параллельно оси привело бы к тому, что ход резьбы в середине детали был бы слишком мал. С помощью сферических резьб это можно компенсировать.

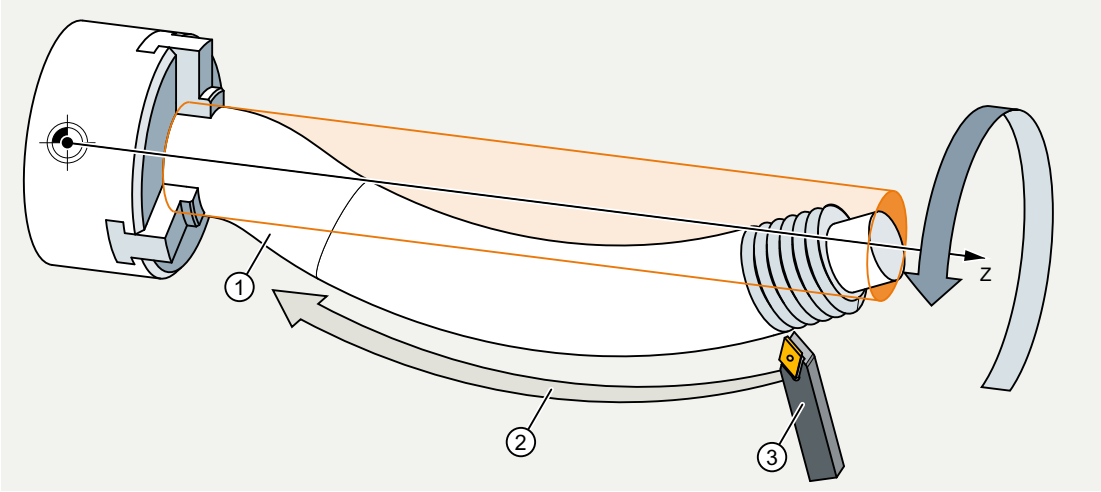


Рисунок 3-6 Нарезание сферической резьбы резцом

Программирование

Нарезание сферической резьбы резцом программируется с помощью `G335` или `G336`:

`G335`:	Нарезание сферической резьбы резцом на проходящей **по часовой стрелке** круговой траектории инструмента
`G336`:	Нарезание сферической резьбы резцом на проходящей **против часовой стрелки** круговой траектории инструмента

Программирование осуществляется сначала, как при линейной резьбе, путем указания осевых конечных точек кадра и шага через параметры `I`, `J`, и `K` (см. "Резьбонарезание с постоянным шагом (G33, SF) (Страница 229)").

Дополнительно указывается еще и дуга окружности. Ее можно запрограммировать, как в случае `G2`/`G3`, указав центр, радиус, апертурный угол или промежуточные точки (см. "Круговая интерполяция (Страница 197)"). При этом в случае программирования сферической резьбы с указанием центра нужно учитывать следующее: Поскольку при резьбонарезании для шага используются `I`, `J`, и `K`, параметры окружности при программировании с указанием центра нужно программировать через `IR=...`, `JR=...` и `KR=...`.

`IR=...`:	Декартова координата центра окружности в направлении X
`JR=...`:	Декартова координата центра окружности в направлении Y
`KR=...`:	Декартова координата центра окружности в направлении Z

Примечание

IR, JR и KR – стандартные значения настраиваемых через машинные данные (MD10651 $MN_IPO_PARAM_THREAD_NAME_TAB) имен параметров интерполяции для сферической резьбы.

Отклонения от этих стандартных значений см. в данных изготовителя станка!

Опционально можно указать еще и смещение стартовой точки `SF` (см. "Резьбонарезание с постоянным шагом (G33, SF) (Страница 229)").

Синтаксис

Таким образом, синтаксис для программирования сферической резьбы имеет следующую общую форму:
`G335/G336` <координата(-ы) конечной(-ых) точки(-ек) оси> <шаг> <дуга окружности> [<смещение стартовой точки>]

Примеры

Пример 1: Сферическая резьба по часовой стрелке с программированием конечных точек и центра

Программный код	Комментарий
`N5 G0 G18 X50 Z50`	; Подвод в начальную точку.
`N10 G335 Z100 K=3.5 KR=25 IR=-20 SF=90`	; Нарезание резцом сферической резьбы по часовой стрелке.

Рисунок 3-7 Сферическая резьба по часовой стрелке с программированием конечных точек и центра

Пример 2: Сферическая резьба против часовой стрелки с программированием конечных точек и центра

Программный код	Комментарий
`N5 G0 G18 X50 Z50`	; Подвод в начальную точку.

Программный код	Комментарий
`N10 G336 Z100 K=3.5 KR=25 IR=20 SF=90`	; Нарезание резцом сферической резьбы против часовой стрелки.

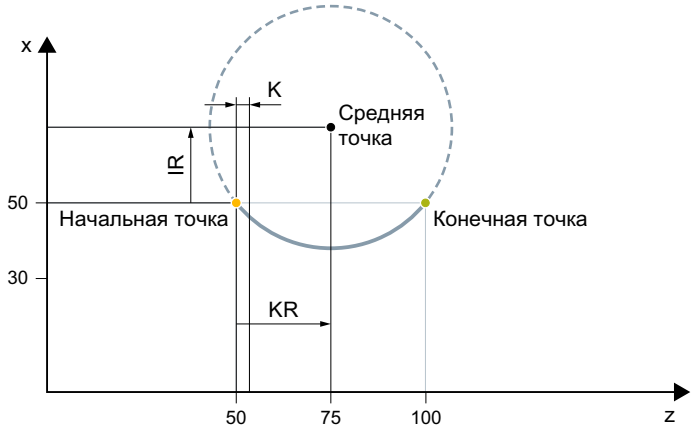


Рисунок 3-8 Сферическая резьба против часовой стрелки с программированием конечных точек и центра

Пример 3: Сферическая резьба по часовой стрелке с программированием конечных точек и радиуса

Программный код

```
N5 G0 G18 X50 Z50
N10 G335 Z100 K=3.5 CR=32 SF=90
```

Рисунок 3-9 Сферическая резьба по часовой стрелке с программированием конечных точек и радиуса

Пример 4: Сферическая резьба по часовой стрелке с программированием конечных точек и апертурного угла

Программный код

```
N5 G0 G18 X50 Z50
```

Программный код

```
N10 G335 Z100 K=3.5 AR=102.75 SF=90
```

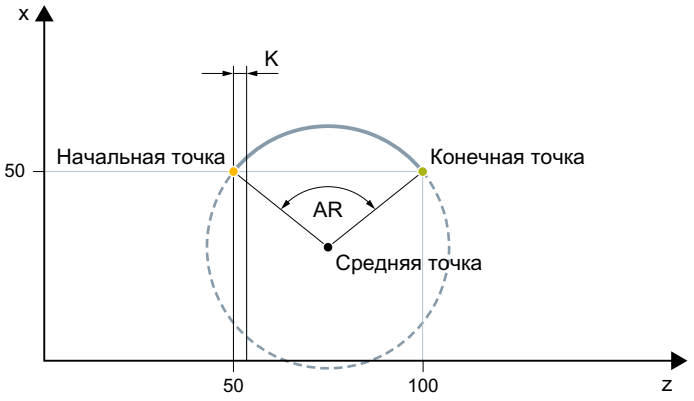


Рисунок 3-10 Сферическая резьба по часовой стрелке с программированием конечных точек и апертурного угла

Пример 5: Сферическая резьба по часовой стрелке с программированием центра и апертурного угла

Программный код

```
N5 G0 G18 X50 Z50
N10 G335 K=3.5 KR=25 IR=-20 AR=102.75 SF=90
```

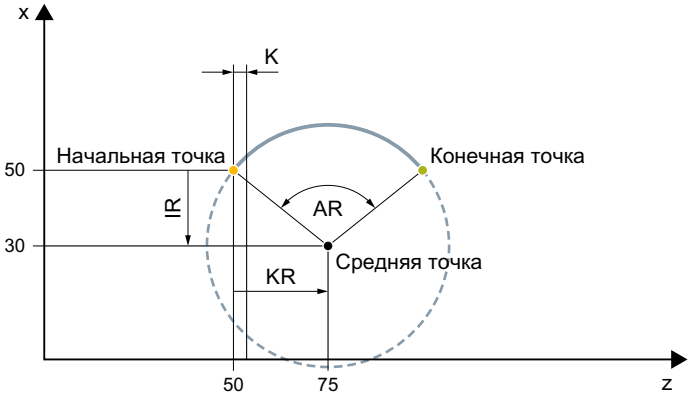


Рисунок 3-11 Сферическая резьба по часовой стрелке с программированием центра и апертурного угла

Пример 6: Сферическая резьба по часовой стрелке с программированием конечных и промежуточных точек

Программный код

```
N5 G0 G18 X50 Z50
N10 G335 Z100 K=3.5 I1=60 K1=64
```

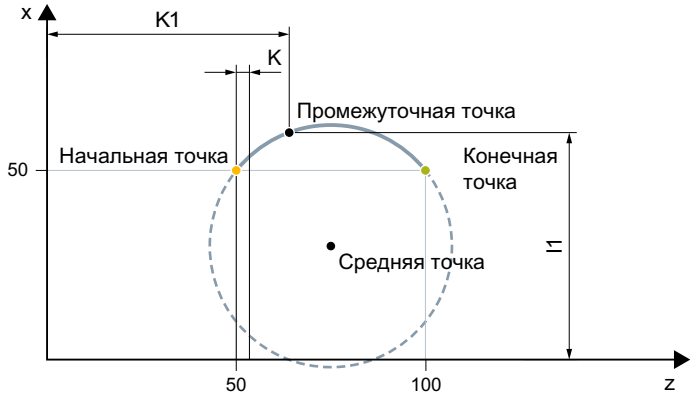
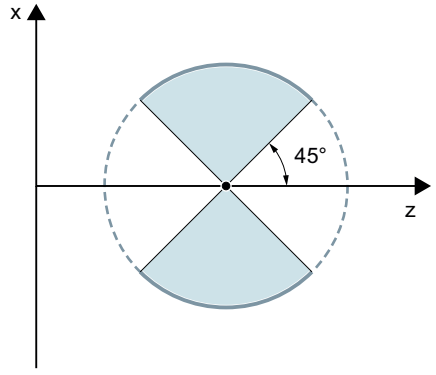


Рисунок 3-12 Сферическая резьба по часовой стрелке с программированием конечных и промежуточных точек

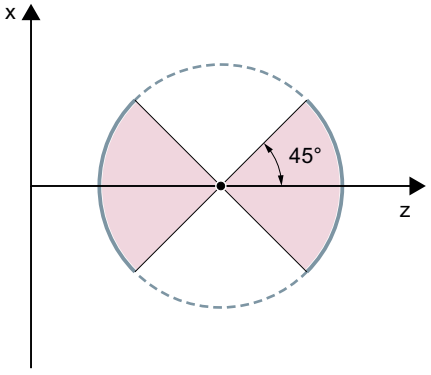
Дополнительная информация

Разрешенные области дуги окружности

Запрограммированная через G335/G336 дуга окружности должна лежать в области, в которой указанная главная ось резьбы (I, J или K) имеет часть главной оси на дуге окружности по всей длине дуги:



Разрешенные области для оси **Z** (шаг программируется через K)



Разрешенные области для оси **X** (шаг программируется через I)

Смена главной оси резьбы, как показано на следующем рисунке, **не** разрешена:

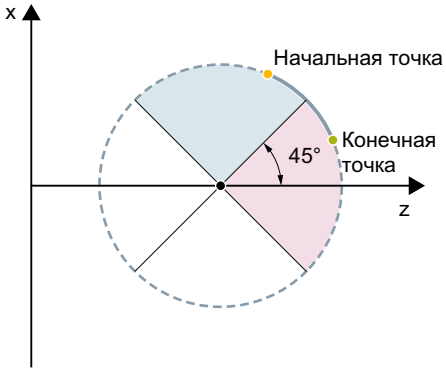


Рисунок 3-13 Сферическая резьба: Неразрешенная область

Фреймы

G335 и G336 возможны и при активных фреймах. Только нужно следить, чтобы в базовой кинематической системе (BKS) соблюдались разрешенные области дуги окружности.

Граничные условия программирования окружностей

Для программирования окружностей через G335/G336 действительны граничные условия, описанные для программирования окружностей через G2/G3 (см. "Круговая интерполяция (Страница 197)").

3.9.11 Нарезание внутренней резьбы без компенсирующего патрона

3.9.11.1 Нарезание внутренней резьбы без компенсирующего патрона и движение отвода (G331, G332)

При нарезании внутренней резьбы без компенсирующего патрона с помощью команд `G331` и `G332` выполняются следующие движения перемещения.

- `G331`: Нарезание внутренней резьбы в направлении сверления до конечной точки резьбы
- `G332`: Движение отвода до кадра нарезания внутренней резьбы `G331` с автоматическим реверсированием направления вращения шпинделя

Синтаксис

```
G331 <ось> <шаг резьбы>
G331 <ось> <шаг резьбы> S...
G332 <ось> <шаг резьбы>
```

Объяснение

G331:	Нарезание внутренней резьбы Нарезание внутренней резьбы описывается через движение перемещения оси (глубина сверления) и шаг резьбы.	
	Активность:	модально
G332:	Движение отвода при нарезании внутренней резьбы Движение отвода необходимо выполнять с тем же шагом, что и нарезание внутренней резьбы (G331). Реверсирование шпинделя осуществляется автоматически.	
	Активность:	модально
<ось>:	Путь/позиция перемещения гео-оси (X, Y или Z) до конечной точки резьбы, например, Z50	
<шаг резьбы>:	Шаг резьбы I (X), J (Y) или K (Z): • Положительный шаг: Правая резьба, например, K1,25 • Отрицательный шаг: Левая резьба, например, K-1,25	
	Диапазон значений:	от ±0,001 до ±2000,00 мм/виток
S...:	Число оборотов шпинделя Если число оборотов шпинделя не указано, перемещение выполняется с последним действительным числом оборотов шпинделя.	

Примечание

Второй блок данных ступеней редуктора

Для достижения эффективного согласования скорости шпинделя и момента двигателя и возможности более быстрого разгона при нарезании внутренней резьбы, в спец. для оси машинных данных может быть предустановлен второй блок данных ступеней редуктора, отличный от первого блока данных ступеней редуктора и также независимый от его порогов переключения скорости, для двух других конфигурируемых порогов переключения (макс. скорость и мин. скорость). Необходимо учитывать заданные изготовителем станка параметры.

Дополнительная информация: см.: Описание функций – Оси и шпиндели.

Примеры

- Пример: Нарезание внутренней резьбы с G331 / G332 (Страница 250)
- Пример: Вывод запрограммированной скорости сверления на текущей ступени редуктора (Страница 250)
- Пример: Использование второго блока данных ступеней редуктора (Страница 251)
- Пример: Скорость не программируется, контроль ступени редуктора (Страница 251)
- Пример: Смена ступеней редуктора невозможна, контроль ступени редуктора (Страница 251)
- Пример: Программирование без SPOS (Страница 252)

3.9.11.2 Пример: Нарезание внутренней резьбы с G331 / G332

Программный код	Комментарий
`N10 SPOS[n]=0`	; Шпиндель: Режим управления положением
	; Стартовая позиция 0 градусов
`N20 G0 X0 Y0 Z2`	; Оси: Подвод к стартовой позиции
`N30 G331 Z-50 K-4 S200`	; Нарезание внутренней резьбы в `Z`,
	; Шаг `K-4` отрицательный =>
	; Направление вращения шпинделя: Левое вращение,
	; Скорость шпинделя 200 об/мин
`N40 G332 Z3 K-4`	; Движение отвода в `Z`,
	; Шаг `K-4` отрицательный (левое вращение),
	; Автономное реверсирование направления вращения =>
	; Правое направление вращения шпинделя
`N50 G1 F1000 X100 Y100 Z100 S300 M3`	Шпиндель в шпиндельном режиме

3.9.11.3 Пример: Вывод запрограммированной скорости сверления на текущей ступени редуктора

Программный код	Комментарий
`N05 M40 S500`	; запрограммированное число оборотов шпинделя: 500 об/мин =>
	; Ступень редуктора 1 (20–1028 об/мин)
...	
`N55 SPOS=0`	; Позиционировать шпиндель
`N60 G331 Z-10 K5 S800`	; нарезание резьбы
	; Скорость шпинделя 800 об/мин => ступень редуктора 1

Подходящая для запрограммированной скорости шпинделя `S500` ступень редуктора для `M40` определяется из первого блока данных ступеней редуктора. Запрограммированная скорость сверления, например `S800`, выводится на текущей ступени редуктора и при необходимости ограничена до макс. скорости ступени редуктора. Автоматическая смена ступеней редуктора после выполнения `SPOS` невозможна. Условием автоматической смены ступеней редуктора является режим управления по скорости шпинделя.

Примечание

Если при скорости шпинделя в 800 об/мин необходимо выбрать ступень редуктора 2, то для этого пороги переключения для макс. и мин. скорости должны быть сконфигурированы в соответствующих машинных данных второго блока данных ступеней редуктора (см. примеры ниже).

3.9.11.4 Пример: Использование второго блока данных ступеней редуктора

Пороги переключения второго блока данных ступеней редуктора для макс. и мин. скорости обрабатываются модально при G331/G332 и программировании значения S для активного мастер-шпинделя. Автоматическая смена ступеней редуктора M40 должна быть активной. Определенная таким образом ступень редуктора сравнивается с активной ступенью редуктора. Если между ними существует различие, то выполняется смена ступеней редуктора.

Программный код	Комментарий
N05 M40 S500	; запрограммированное число оборотов шпинделя: 500 об/мин
...	
N50 G331 S800	; Мастер-шпиндель: выбирается ступень редуктора 2
N55 SPOS=0	; Позиционировать шпиндель
N60 G331 Z-10 K5	; нарезание резьбы
	; Разгон шпинделя из 2-го блока данных ступеней редуктора

3.9.11.5 Пример: Скорость не программируется, контроль ступени редуктора

Если при использовании второго блока данных ступеней редуктора с G331 скорость не программируется, то резьба изготовляется с последней запрограммированной скоростью. Смена ступеней редуктора не выполняется. Но в этом случае контролируется, лежит ли последняя запрограммированная скорость в заданном диапазоне скорости (пороги переключения для макс. и мин. скорости) активной ступени редуктора. В ином случае сигнализируется ошибка 16748.

Программный код	Комментарий
N05 M40 S800	; запрограммированное число оборотов шпинделя: 800 об/мин
...	
N55 SPOS=0	; Позиционировать шпиндель
N60 G331 Z-10 K5	; нарезание резьбы
	; Контроль скорости шпинделя 800 об/мин
	; Ступени редуктора 1 активны
	; Ступень редуктора 2 должна была бы быть активна => Аварийное сообщение 16748

3.9.11.6 Пример: Смена ступеней редуктора невозможна, контроль ступени редуктора

Если при использовании второго блока данных ступеней редуктора в кадре G331 дополнительно к геометрии программируется скорость шпинделя, то, если скорость выходит за пределы заданного диапазона скоростей (пороги переключения для макс. и мин. скорости) активной ступени редуктора, смена ступеней редуктора не может быть выполнена, т.к. в этом случае не было бы выдержано движение по траектории шпинделя и оси(ей) подачи.

Как и в примере выше, в кадре G331 контролируется скорость и ступень редуктора и при необходимости сигнализируется ошибка 16748.

Программный код	Комментарий
N05 M40 S500	; запрограммированное число оборотов шпинделя: 500 об/мин =>
	; Ступень редуктора 1
...	
N55 SPOS=0	; Позиционировать шпиндель
N60 G331 Z-10 K5 S800	; нарезание резьбы
	; Смена ступеней редуктора невозможна,
	; Контроль скорости шпинделя 800 об/мин
	; с 1-м блоком данных ступеней редуктора: Ступень редуктора 2
	; должна была бы быть активна => аварийное сообщение 16748

3.9.11.7 Пример: Программирование без SPOS

Программный код	Комментарий
N05 M40 S500	; запрограммированное число оборотов шпинделя: 500 об/мин =>
	; Ступень редуктора 1 (20–1028 об/мин)
...	
N50 G331 S800	; Мастер-шпиндель: выбирается ступень редуктора 2
N60 G331 Z-10 K5	; нарезание резьбы
	; Разгон шпинделя из 2-го блока данных ступеней редуктора

Резьбовая интерполяция для шпинделя начинается от текущей позиции, которая зависит от обработанной прежде области программы обработки детали, например, если была выполнена смена ступеней редуктора. Поэтому дополнительная обработка резьбы может стать невозможной.

Примечание

Учитывать, что при обработке с несколькими шпинделями сверлильный шпиндель должен быть и мастер-шпинделем. Через программирование SETMS (<номер шпинделя>) сверлильный шпиндель может быть назначен мастер-шпинделем.

3.9.12 Нарезание внутренней резьбы с компенсирующим патроном

3.9.12.1 ; Нарезание внутренней резьбы с компенсирующим патроном и движением отвода (G63)

При нарезании внутренней резьбы без компенсирующего патрона командой `G63` выполняются следующие движения перемещения:

- `G63`: Нарезание внутренней резьбы в направлении сверления до конечной точки резьбы
- `G63`: Движение отвода запрограммированным реверсированием направления вращения шпинделя

Примечание

После кадра `G63` последний действительный тип интерполяции `G0`, `G1`, `G2` снова активен.

Синтаксис

`G63` <ось> <направление вращения> <число оборотов> <подача>

Объяснение

`G63`:	Нарезание внутренней резьбы с компенсирующим патроном	
	Активность:	покадрово
<ось>:	Путь/позиция перемещения гео-оси (X, Y или Z) до конечной точки резьбы, например, Z50	
<направление вращения>:	Направление вращения шпинделя: • `M3`: Направление вращения по часовой стрелке, правая резьба • `M4`: Направление вращения против часовой стрелки, левая резьба	
<число оборотов>:	Число оборотов шпинделя, максимально допустимое при нарезании внутренней резьбы, например, S100	
<подача>:	Подача оси сверления F, где F = число оборотов шпинделя * шаг резьбы	

Пример

Нарезание сферической резьбы M5

- Шаг резьбы согласно стандарту 0,8 мм/об
- Число оборотов шпинделя S: 200 об/мин
- Подача F = 200 об/мин * 0,8 мм/об = 160 мм/мин.

Программный код	Комментарий
`N10 G1 X0 Y0 Z2 F1000 S200 M3`	; Движение к стартовой точке ; Шпиндель по часовой стрелке, 200 об/мин

Программный код	Комментарий
`N20 G63 Z-50 F160`	; Нарезание внутренней резьбы с компенсирующим патроном
	; Глубина сверления: абсолютная `Z = 50` мм
	; Подача: 160 мм/мин
`N30 G63 Z3 M4`	; Движение отвода: абсолютное `Z = 3` мм
	; Реверсирование направления вращения
	; Шпиндель против часовой стрелки, 200 об/мин

3.9.13 Фаска, закругление (CHF, CHR, RND, RNDM, FRC, FRCM)

Углы контура в пределах активной рабочей плоскости могут быть выполнены как закругление или фаска.

Для оптимизации качества поверхности для снятия фаски/закругление может быть запрограммирована собственная подача. Если подача не программируется, то действует обычная подача по траектории F.

С помощью функции "Модальное закругление" возможно последовательное однотипное закругление нескольких углов контура.

Синтаксис

Снятие фаски угла контура:
`G... X... Z... CHR/CHF=<значение> FRC/FRCM=<значение>`
`G... X... Z...`

Закругление угол контура:
`G... X... Z... RND=<значение> FRC=<значение>`
`G... X... Z...`

Модальное закругление:

```
G... X... Z... RNDM=<значение> FRCM=<значение>
...
RNDM=0
```

Примечание

Технология (подача, тип подачи, команды M ...) для снятия фаски/закругления берется в зависимости от установки бита 0 в машинных данных
MD20201 $MC_CHFRND_MODE_MASK (параметры фаски/закругления) либо из предшествующего, либо из следующего кадра. Рекомендуемой установкой является получение из предшествующего кадра (бит 0 = 1).

Значение

CHF=… :	Снятие фасок угла контура	
	<значение>:	Длина фаски (единица измерения согласно G70/G71)
CHR=… :	Снятие фасок угла контура	
	<значение>:	Ширина фаски в первоначальном направлении движения (единица измерения согласно G70/G71)
RND=… :	Закругление угла контура	
	<значение>:	Радиус закругления (единица измерения согласно G70/G71)
RNDM=… :	Модальное закругление (однотипное закругление нескольких последовательных углов контура)	
	<значение>:	Радиус закруглений (единица измерения согласно G70/G71)
		С RNDM=0 модальное закругление отключается.
FRC=… :	Действующая покадрово подача для снятия фаски/закругления	
	<значение>:	Скорость подачи в мм/мин (при активной G94) или мм/об (при активной G95)
FRCM=… :	Действующая модально подача для снятия фаски/закругления	
	<значение>:	Скорость подачи в мм/мин (при активной G94) или мм/об (при активной G95)
		С FRCM=0 действующая модально подача для снятия фаски/закругления отключается и активна запрограммированная в F подача.

Примечание

Фаска/закругление слишком большие

Если запрограммированные значения для фаски (CHF/CHR) или закругления (RND/RNDM) для участвующих элементов контура слишком велики, то фаска или закругление автоматически адаптируются:

1. Если установлено MD11411 $MN_ENABLE_ALARM_MASK бит 4, то выдается предупреждение 10833 "Фаску или закругление нужно уменьшить" (Cancel-предупреждение).
2. Фаска/закругление уменьшается так, чтобы соответствовать углу контура. При этом возникает как минимум один кадр без движения. На этом кадре движение обязательно останавливается.

Примечание

Фаска/закругление невозможны

Фаска/закругление не вставляется, если:

- линейный или круговой контур отсутствует в плоскости.
- движение происходит вне плоскости.
- осуществляется смена плоскости.
- превышено установленное в машинных данных количество кадров, не содержащих информации для перемещения (к примеру, только выводы команд).

Примечание

FRC/FRCM

FRC/FRCM не действует, если фаска перемещается с G0; программирование возможно в соответствии со значением F без сообщения об ошибке.

FRC действует только тогда, когда в кадре запрограммирована и фаска/закругление, или была активирована RNDM.

FRC переписывает в актуальном кадре значение F или FRCM.

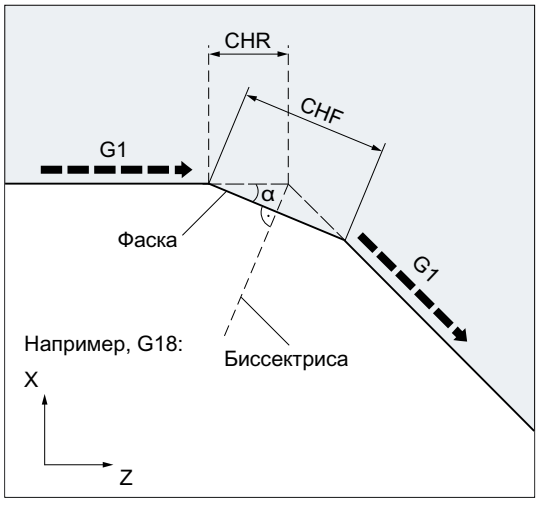
Запрограммированная в FRC подача должна быть больше нуля.

FRCM=0 активирует для снятия фаски/закругления запрограммированную в F подачу.

Если запрограммирована FRCM, то эквивалентно F значение FRCM при переходе G94 ↔ G95 и т. п. должно быть запрограммировано заново. Если заново программируется только F, и перед переключением типа подачи FRCM > 0, то следует сообщение об ошибке.

Примеры

Пример 1: Снятие фаски между двумя прямыми



- MD20201 Бит 0 = 1 (получение из предшествующего кадра)
- G71 активна.
- Ширина фаски в направлении движения (CHR) должна составить 2 мм, подача для снятия фасок 100 мм/мин.

Возможно два способа программирования:

- Программирование с CHR

Программный код

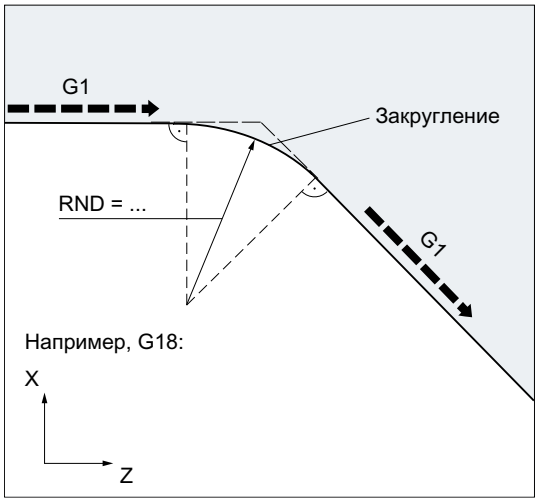
```
...
N30 G1 Z… CHR=2 FRC=100
N40 G1 X…
...
```

- Программирование с CHF

Программный код

```
...
N30 G1 Z… CHF=2(cosα*2) FRC=100
N40 G1 X…
...
```

Пример 2: Закругление между двумя прямыми



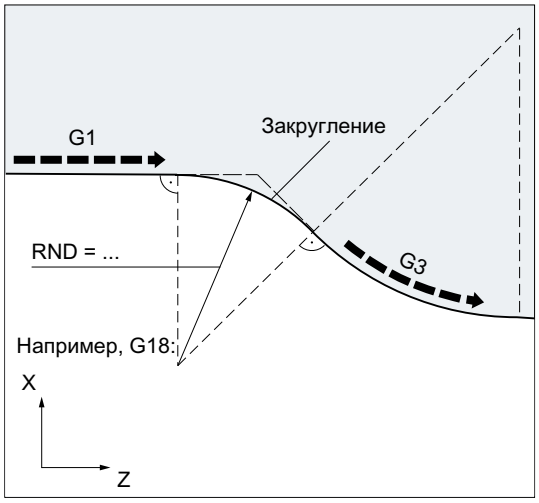
- MD20201 Бит 0 = 1 (получение из предшествующего кадра)
- G71 активна.
- Радиус закругления должен быть 2 мм, подача для закругления 50 мм/мин.

Программный код

```
...
N30 G1 Z… RND=2 FRC=50
N40 G1 X…
...
```

Пример 3: Закругление между прямой и окружностью

Между линейными и круговыми контурами в любой комбинации с помощью функции RND с тангенциальным примыкания может быть вставлен элемент кругового контура.



- MD20201 Бит 0 = 1 (получение из предшествующего кадра)
- G71 активна.
- Радиус закругления должен быть 2 мм, подача для закругления 50 мм/мин.

Программный код
...
N30 G1 Z… RND=2 FRC=50
N40 G3 X… Z… I… K…
...

Пример 4: Модальное закругление для снятия грата острых кромок детали

Программный код	**Комментарий**
...	
N30 G1 X… Z… RNDM=2 FRCM=50	; Включить модальное закругление. Радиус закругления: 2мм Подача для закругления: 50 мм/мин
N40...	
N120 RNDM=0	; Выключить модальное закругление.
...	

Пример 5: Получение технологии из последующего или предшествующего кадра

- MD20201 Бит 0 = 0: Получение из последующего кадра (стандартная установка!)

Программный код	**Комментарий**
N10 G0 X0 Y0 G17 F100 G94	
N20 G1 X10 CHF=2	; Фаска N20-N30 с F=100 мм/мин
N30 Y10 CHF=4	; Фаска N30-N40 с FRC=200 мм/мин
N40 X20 CHF=3 FRC=200	; Фаска N40-N60 с FRCM=50 мм/мин
N50 RNDM=2 FRCM=50	

Программный код	Комментарий
N60 Y20	; Модальное закругление N60-N70 с FRCM=50 мм/мин
N70 X30	; Модальное закругление N70-N80 с FRCM=50 мм/мин
N80 Y30 CHF=3 FRC=100	; Фаска N80-N90 с FRC=100 мм/мин
N90 X40	; Модальное закругление N90-N100 с F=100 мм/мин (выключение FRCM)
N100 Y40 FRCM=0	; Модальное закругление N100-N120 с G95 FRC=1 мм/оборот
N110 S1000 M3	
N120 X50 G95 F3 FRC=1	
...	
M02	

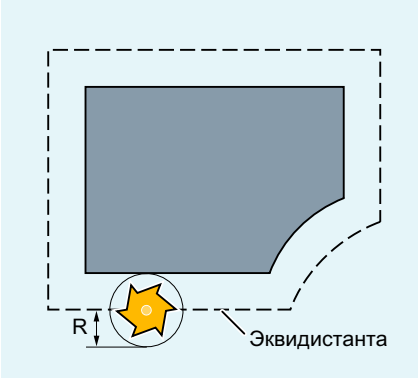
- MD20201 Бит 0 = 1: Получение из предшествующего кадра (рекомендуемая установка!)

Программный код	Комментарий
N10 G0 X0 Y0 G17 F100 G94	
N20 G1 X10 CHF=2	; Фаска N20-N30 с F=100 мм/мин
N30 Y10 CHF=4 FRC=120	; Фаска N30-N40 с FRC=120 мм/мин
N40 X20 CHF=3 FRC=200	; Фаска N40-N60 с FRC=200 мм/мин
N50 RNDM=2 FRCM=50	
N60 Y20	; Модальное закругление N60-N70 с FRCM=50 мм/мин
N70 X30	; Модальное закругление N70-N80 с FRCM=50 мм/мин
N80 Y30 CHF=3 FRC=100	; Фаска N80-N90 с FRC=100 мм/мин
N90 X40	; Модальное закругление N90-N100 с FRCM=50 мм/мин
N100 Y40 FRCM=0	; Модальное закругление N100-N120 с F=100 мм/мин
N110 S1000 M3	
N120 X50 CHF=4 G95 F3 FRC=1	; Фаска N120-N130 с G95 FRC=1 мм/оборот
N130 Y50	; Модальное закругление N130-N140 с F=3 мм/оборот
N140 X60	
...	
M02	

3.10 Коррекции радиуса инструмента

3.10.1 Коррекция радиуса инструмента (G40, G41, G42, OFFN)

При включенной коррекции радиуса инструмента (КРИ) СЧПУ автоматически вычисляет для различных инструментов соответствующие эквидистантные ходы инструмента.



Синтаксис

G0/G1 X... Y… Z... **G41**/**G42** [**OFFN=<значение>**]	
...	
G40 X... Y… Z...	

Объяснение

G41:	Включить КРИ с направлением обработки **слева** от контура
G42:	Включить КРИ с направлением обработки **справа** от контура
OFFN=<значение>:	Припуск к запрограммированному контуру (смещение нормального контура) (опция) Например, для создания эквидистантных траекторий для получистовой обработки.
G40:	Выключить КРИ

Примечание

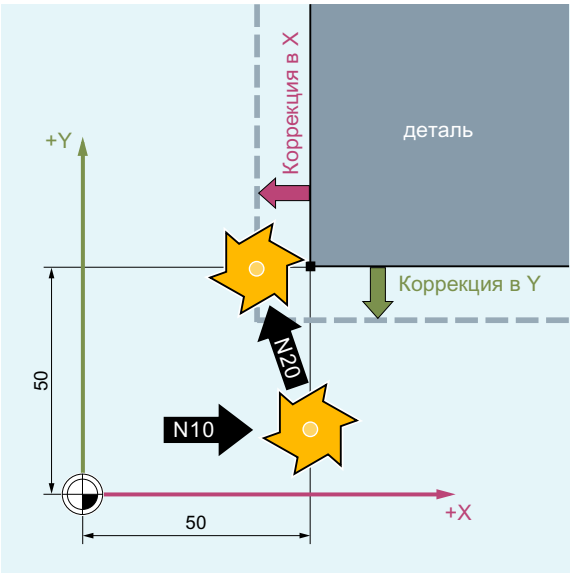
В кадре ЧПУ с G40/G41/G42 должна быть активна G0 или G1 и указана как минимум одна ось выбранной рабочей плоскости.

Если при включении указывается только одна ось, то последняя позиция второй оси автоматически дополняется и перемещение осуществляется в **обоих** осях.

Обе оси должны быть активны в канале в качестве гео-осей. Это может обеспечиваться программированием GEOAX.

Примеры

Пример 1: Фрезерная обработка

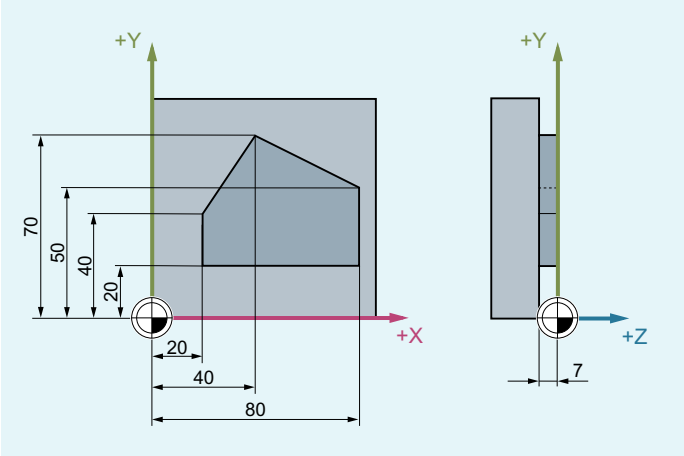


Программный код	Комментарий
`N10 G0 X50 T1 D1`	; Включается только коррекция длин инструмента. Подвод к `X50` выполняется без коррекции.
`N20 G1 G41 Y50 F200`	; Включается коррекция радиуса, подвод к точке `X50/Y50` осуществляется с коррекцией.
`N30 Y100`	
...	

Пример 2: "Классический" принцип действий на примере фрезерования

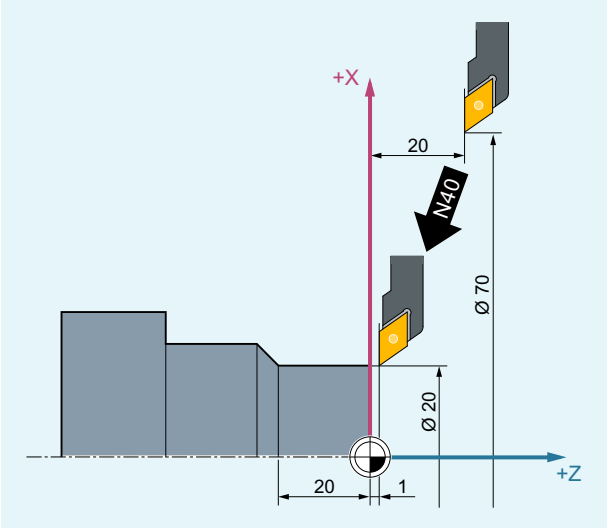
"Классический" принцип действий:

1. Вызов инструмента
2. Установить инструмент.
3. Включить рабочую плоскость и коррекцию радиуса инструмента.



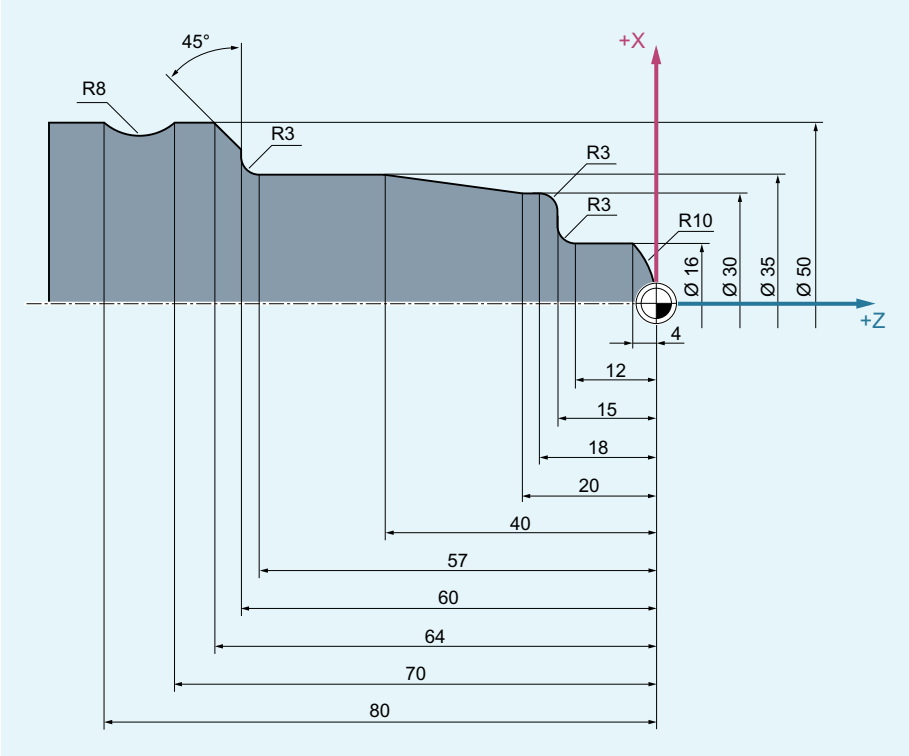
Программный код	Комментарий
`N10 G0 Z100`	; Отвод для смены инструмента.
`N20 G17 T1 M6`	; Смена инструмента
`N30 G0 X0 Y0 Z1 M3 S300 D1`	; Вызов значений коррекции инструмента, выбор коррекции длин.
`N40 Z-7 F500`	; Подача инструмента.
`N50 G41 X20 Y20`	; Включение коррекции радиуса инструмента, инструмент работает слева от контура.
`N60 Y40`	; Фрезерование контура.
`N70 X40 Y70`	
`N80 X80 Y50`	
`N90 Y20`	
`N100 X20`	
`N110 G40 G0 Z100 M30`	; Отвод инструмента, конец программы.

Пример 3: Токарная обработка



Программный код	Комментарий
...	
`N20 T1 D1`	; Включается только коррекция длин инструмента.
`N30 G0 X70 Z20`	; Подвод к `X70 Z20` выполняется без коррекции.
`N40 G42 X20 Z1`	; Включается коррекция радиуса, подвод к точке `X20/Z1` выполняется с коррекцией.
`N50 G1 Z-20 F0.2`	
...	

Пример 4: Токарная обработка



Программный код	Комментарий
`N5 G0 G53 X280 Z380 D0`	; Стартовая точка
`N10 TRANS X0 Z250`	; Смещение нулевой точки
`N15 LIMS=4000`	; Ограничение скорости (`G96`)
`N20 G96 S250 M3`	; Выбор постоянной подачи
`N25 G90 T1 D1 M8`	; Выбор инструмента и коррекции
`N30 G0 G42 X-1.5 Z1`	; Подвод инструмента в рабочее положение с коррекцией радиуса инструмента
`N35 G1 X0 Z0 F0.25`	
`N40 G3 X16 Z-4 I0 K-10`	; Обтачивание радиуса 10
`N45 G1 Z-12`	

Программный код	Комментарий
N50 G2 X22 Z-15 CR=3	; Обтачивание радиуса 3
N55 G1 X24	
N60 G3 X30 Z-18 I0 K-3	; Обтачивание радиуса 3
N65 G1 Z-20	
N70 X35 Z-40	
N75 Z-57	
N80 G2 X41 Z-60 CR=3	; Обтачивание радиуса 3
N85 G1 X46	
N90 X52 Z-63	
N95 G0 G40 G97 X100 Z50 M9	; Отключение коррекции радиуса инструмента и подвод к точке смены инструмента
N100 T2 D2	; Вызов инструмента и выбор коррекции
N105 G96 S210 M3	; Выбор постоянной скорости резания
N110 G0 G42 X50 Z-60 M8	; Подвод инструмента в рабочее положение с коррекцией радиуса инструмента
N115 G1 Z-70 F0.12	; Обтачивание диаметра 50
N120 G2 X50 Z-80 I6.245 K-5	; Обтачивание радиуса 8
N125 G0 G40 X100 Z50 M9	; Отвод инструмента и выключение коррекции радиуса инструмента
N130 G0 G53 X280 Z380 D0 M5	; Переход к точке смены инструмента
N135 M30	; Конец программы

Дополнительная информация

Расчет ходов инструмента

Для вычисления ходов инструмента СЧПУ необходима следующая информация:

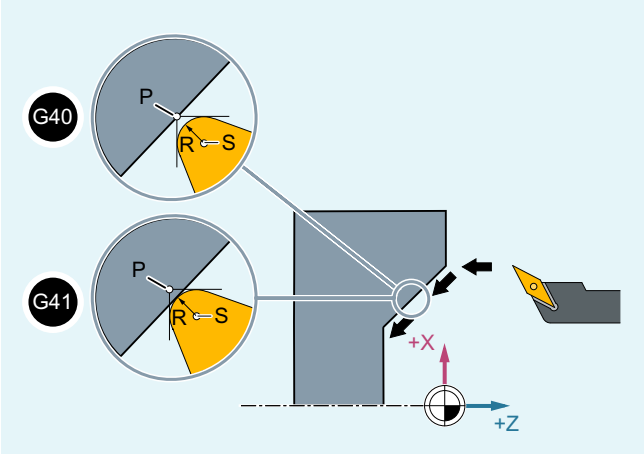
- Номер инструмента (T...), номер резца (D...)
- Направление обработки (G41/G42)
- Рабочая плоскость (G17/G18/G19)

Номер инструмента (T...), номер резца (D...)

Из радиусов фрез или радиусов резцов и данных по положению резцов вычисляется расстояние между траекторией инструмента и контуром детали.

Направление обработки (G41/G42)

Исходя из этого СЧПУ определяет направление, в котором должна смещаться траектория инструмента.

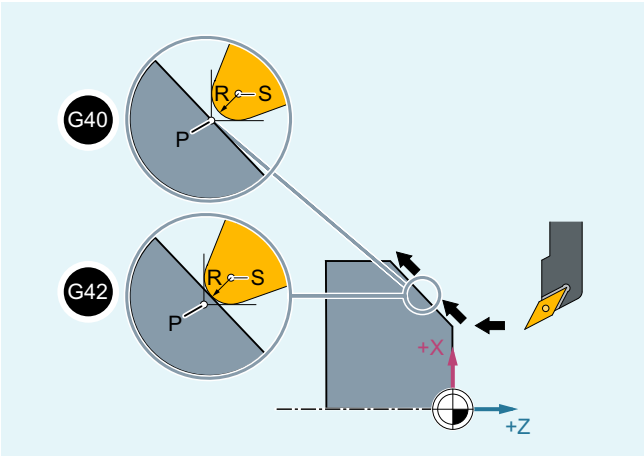


P Острие инструмента

S Центр резца

R Радиус резца

Рисунок 3-14 Коррекция радиуса инструмента при обтачивании с направлением обработки слева от контура (G41)



P Острие инструмента

S Центр резца

R Радиус резца

Рисунок 3-15 Коррекция радиуса инструмента при обтачивании с направлением обработки вправо от контура (G42)

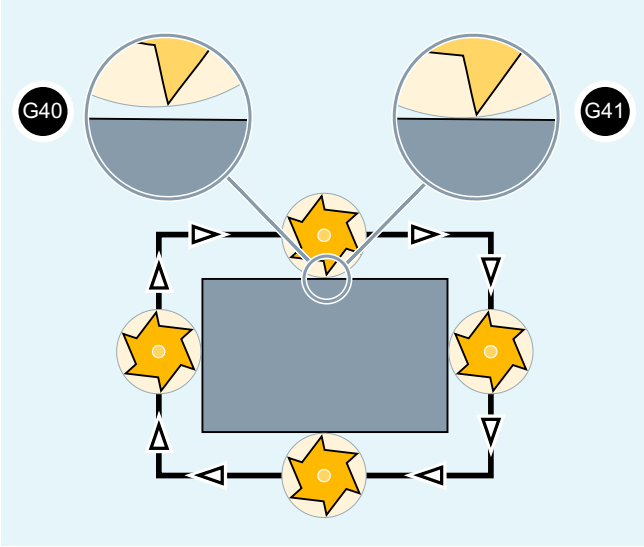


Рисунок 3-16 Коррекция радиуса инструмента при фрезеровании с направлением обработки слева от контура (G41)

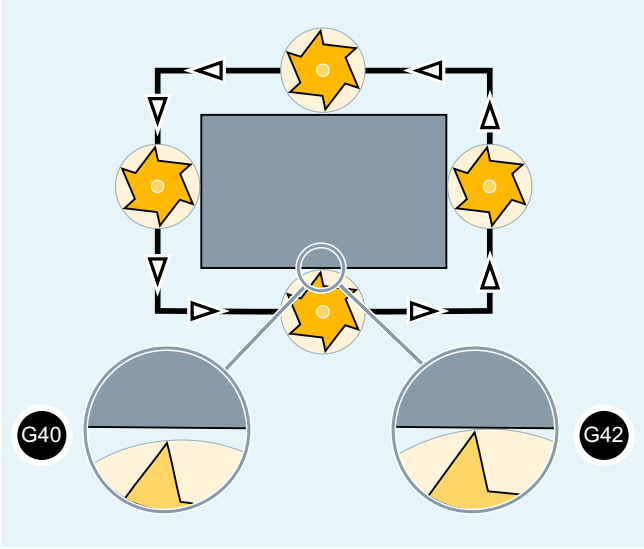


Рисунок 3-17 Коррекция радиуса инструмента при фрезеровании с направлением обработки вправо от контура (G42)

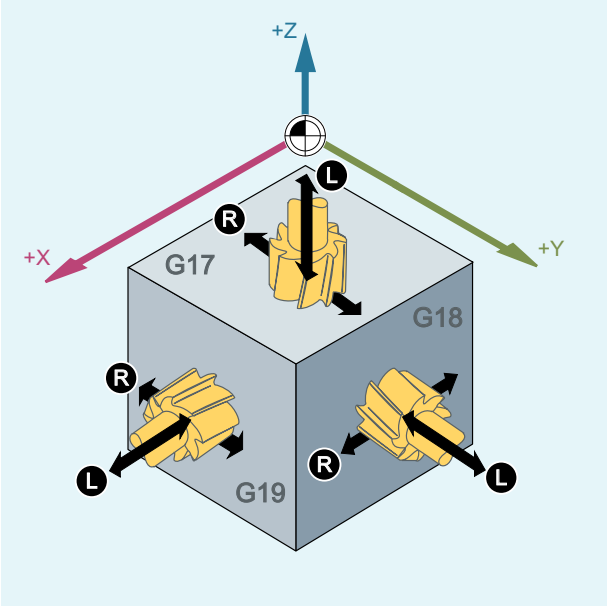
Примечание

Отрицательное значение коррекции идентично смене стороны коррекции (G41 ↔ G42).

Рабочая плоскость (G17/G18/G19)

Исходя из этого СЧПУ определяет плоскость и тем самым осевые направления, в которых осуществляется коррекция.

Пример: Фрезерный инструмент



l	Длина
R	Радиус

Программный код	Комментарий
...	
`N10 G17 G41 …`	; Коррекция на радиус инструмента осуществляется в плоскости `X/Y`, коррекция длины инструмента в направлении `Z`.
...	

Примечание

У двухосевых станков коррекция на радиус инструмента возможна только в "реальных" плоскостях, как правило, для G18.

Коррекция длины инструмента

Согласованный при выборе инструмента оси диаметра параметр износа может быть определен через машинные данные как значение диаметра. При последующей смене плоскостей это согласование не изменяется автоматически. Для этого после смены плоскостей инструмент необходимо выбрать заново.

Режим коррекции на радиус инструмента при замкнутом контуре

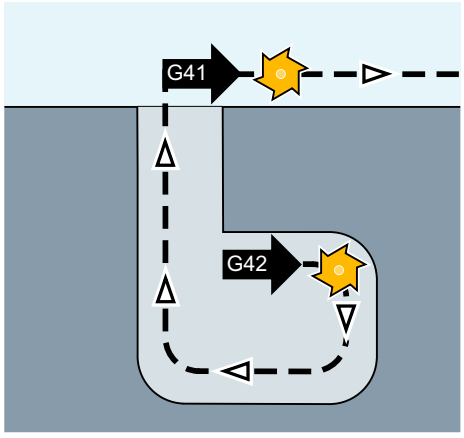
Режим коррекции на радиус инструмента при замкнутом контуре настраивается через установочные данные:

SD42496 $SC_CUTCOM_CLSD_CONT

Значение	Объяснение
false	Если в случае (практически) замкнутого контура, состоящего из двух последовательных круговых кадров или одного кругового и одного линейного кадра, получается две точки пересечения при коррекции на внутренней стороне, то стандартно выбирается точка пересечения, расположенная на первом подконтуре ближе к концу кадра. Контур рассматривается как (практически) замкнутый в том случае, когда расстояние между стартовой точкой первого кадра и конечной точкой второго кадра меньше, чем 10 % эффективного радиуса коррекции, но не больше, чем 1000 дискрет (соответствует 1 мм при 3 позициях после запятой).
TRUE	В ситуации, идентичной описанной выше, выбирается точка пересечения, расположенная на первом подконтуре ближе к началу кадра.

Смена направления коррекции (G41 ↔ G42)

Смена направления коррекции (G41 ↔ G42) может быть запрограммирована без промежуточной G40.



Смена рабочей плоскости

Смена рабочей плоскости (G17/G18/G19) при активной G41/G42 **невозможна**.

Смена блока данных коррекции инструмента (D...)

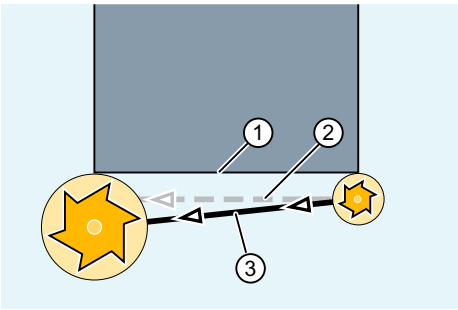
Блок данных коррекции инструмента может быть переключен в режиме коррекции.

Измененный радиус инструмента начинает действовать уже с того кадра, в котором стоит новый номер D.

Примечание

Изменение радиуса или компенсационное движение растягивается на весь кадр и достигает нового эквидистантного расстояния только в запрограммированной конечной точке.

При линейных движениях инструмент движется по наклонной траектории между начальной и конечной точкой:



① Запрограммированный контур
② Траектория инструмента без смены блока данных коррекции инструмента в активном кадре
③ Траектория инструмента при смене блока данных коррекции инструмента в активном кадре и связанном изменении радиуса инструмента

При круговых интерполяций возникают спиральные движения.

Изменение радиуса инструмента

Изменение возможно, например, через системные переменные. Для процесса действует то же, что и для смены блока данных коррекции инструмента (D...).

Примечание

Измененные значения начинают действовать после повторного программирования T или D. Изменение действует только в следующем кадре.

Режим коррекции

Режим коррекции может быть прерван только определенным количеством следующих друг за другом кадров или команд M, не содержащих команд движения или характеристик перемещения в плоскости коррекции.

Примечание

Максимальное количество следующих друг за другом кадров прерывания или команд M может устанавливаться через машинные данные (см. указания изготовителя станка!).

Примечание

Кадр с ходом траектории ноль также считается прерыванием!

См. также

Подвод к контуру и отвод (NORM, KONT, KONTC, KONTT) (Страница 269)

3.10.2 Подвод к контуру и отвод (NORM, KONT, KONTC, KONTT)

Условие

Команды KONTC и KONTT доступны только при разрешении в СЧПУ опции "Полиномиальная интерполяция".

Функция

С помощью команд NORM, KONT, KONTC или KONTT при включенной коррекции радиуса инструмента (G41/G42) путь подвода и отвода инструмента может быть согласован с желаемым ходом контура или с формой заготовки.

С KONTC или KONTT условия постоянства соблюдаются во всех трех осях. Тем самым допускается одновременное программирование компонента хода вертикально к плоскости коррекции.

Синтаксис

G41/G42 **NORM**/**KONT**/**KONTC**/**KONTT** X... Y... Z...	
...	
G40 X... Y... Z...	

Значение

NORM:	Включить точный подвод/отвод по прямой Инструмент устанавливается точно вертикально к точке контура.
KONT:	Включить подвод/отвод с обходом начальной/конечной точки согласно запрограммированному поведению на углах G450 или G451
KONTC:	Включить стабильный по кривизне подвод/отвод
KONTT:	Включить подвод/отвод с постоянным касанием

Примечание

В качестве оригинальных кадров подвода/отвода для KONTC и KONTT разрешены только кадры G1. Они заменяются СЧПУ полиномами для соответствующей траектории подвода/отвода.

Граничные условия

KONTT и KONTC недоступны для вариантов 3D коррекции радиуса инструмента (CUT3DC, CUT3DCC, CUT3DF). Если они все же программируются, то внутри СЧПУ без сигнализации ошибки происходит переключение на NORM.

Пример

KONTC

Подвод к полной окружности, начиная из центра окружности. При этом в конечной точке кадра подвода направление и радиус изгиба идентичны значениям следующей

окружности. Подача в обоих кадрах подвода/отвода осуществляется одновременно в направлении Z. Следующий рисунок показывает вертикальную проекцию траектории инструмента:

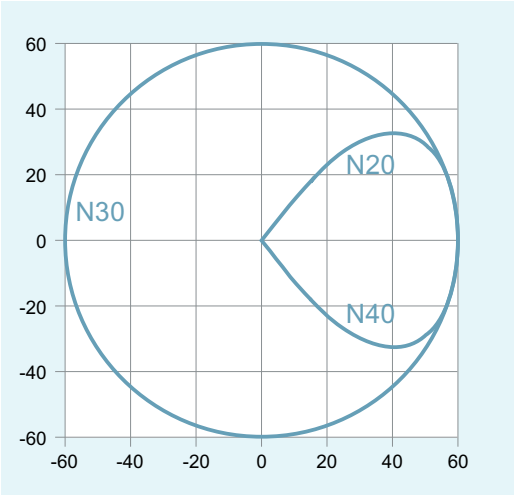


Рисунок 3-18 Вертикальная проекция

Соответствующая часть программы ЧПУ выглядит следующим образом:

Программный код	Комментарий
`$TC_DP1[1,1]=121`	; Фреза
`$TC_DP6[1,1]=10`	; Радиус 10 мм
`N10 G1 X0 Y0 Z60 G64 T1 D1 F10000`	
`N20 G41 KONTC X70 Y0 Z0`	; подвод
`N30 G2 I-70`	; Полный круг
`N40 G40 G1 X0 Y0 Z60`	; Отвод
`N50 M30`	

Одновременно для согласования изгиба с круговой траекторией полной окружности происходит перемещение с Z60 на плоскость окружности Z0:

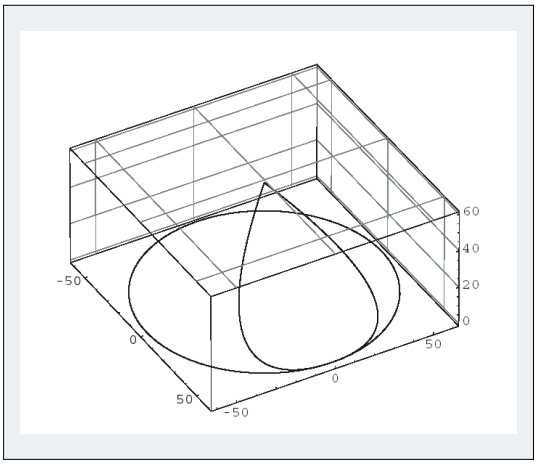
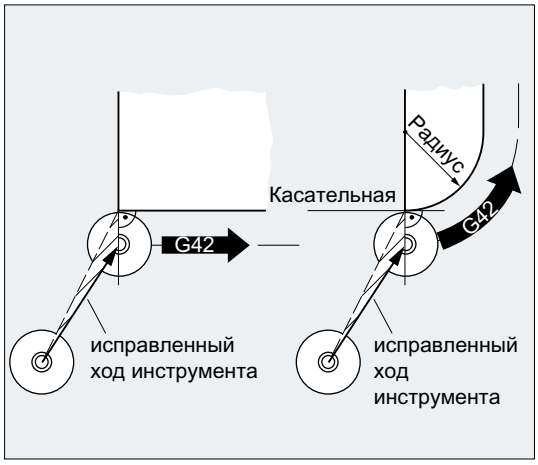


Рисунок 3-19 Представление в пространстве

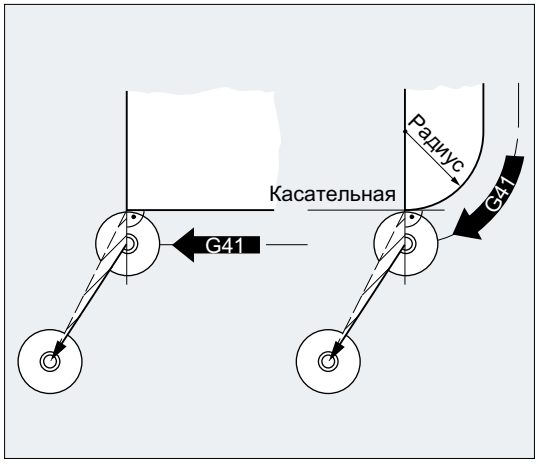
Дополнительная информация

Подвод/отвод с NORM

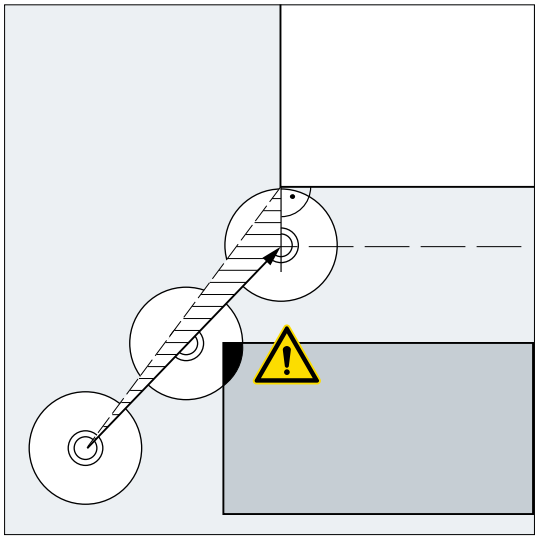
1. Подвод:
   При включенной NORM инструмент движется точно по прямой к исправленной стартовой позиции (независимо от заданного через запрограммированное движение перемещения угла подвода) и устанавливается в начальной точке точно вертикально к касательной к траектории:



2. Отвод:
   Инструмент стоит в вертикальной позиции к последней исправленной конечной точке траектории и движется после этого (независимо от заданного через запрограммированное движение перемещения угла подвода) точно по прямой к следующей, неисправленной позиции, к примеру, к точке смены инструмента:



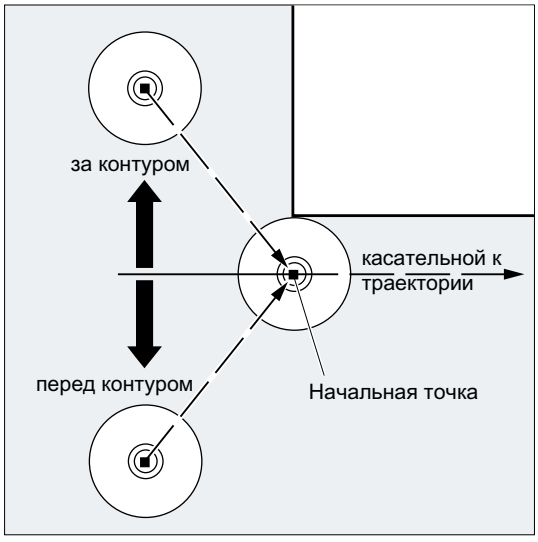
Измененные углы подвода/отвода представляют собой опасность столкновения:



ВНИМАНИЕ
Опасность столкновений Во избежание возможных столкновений учитывать измененные углы подвода/отвода при программировании.

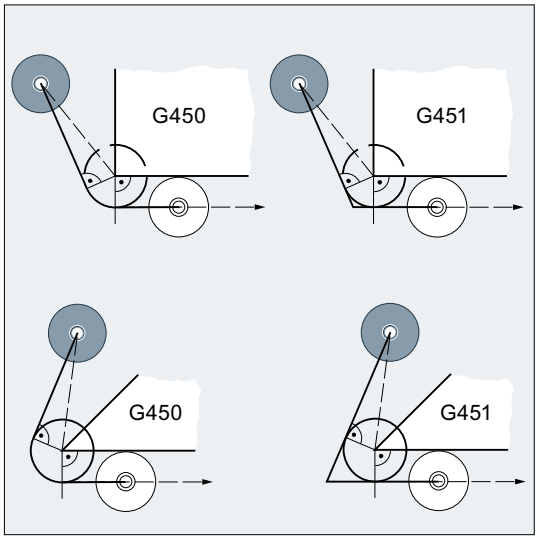
Подвод/отвод с KONT

Перед подводом инструмента может находиться **перед** или **за** контуром. Разделительной линией при этом является касательная к траектории в начальной точке:

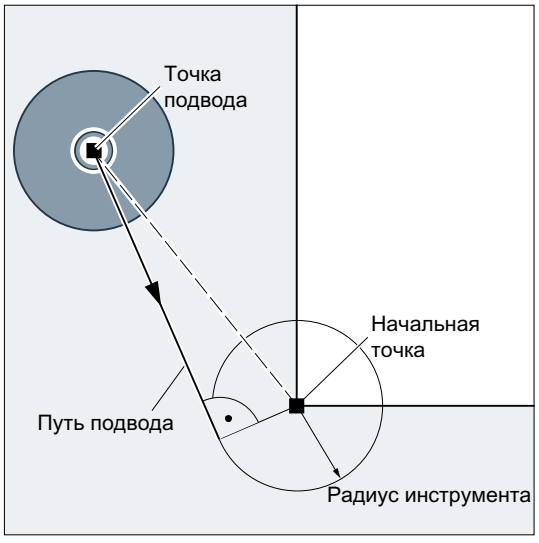


Согласно этому, при подводе/отводе с KONT различается два случая:

1. Инструмент находится перед контуром.
   → Стратегия подвода/отвода как при NORM.
2. Инструмент находится за контуром
   – Подвод:
   Инструмент обходит начальную точку в зависимости от запрограммированного поведения на углах (G450/G451) по круговой траектории или через точку пересечения эквидистант.
   Команды G450/G451 относятся к переходу от актуального кадра к следующему кадру:



В обоих случаях (G450/G451) создается следующий путь подвода:



От неисправленной точки подвода проводится прямая, являющаяся касательной к окружности с радиусом окружности = радиусу инструмента. Центр окружности находится в начальной точке.

– Отвод:
  Для отвода действуют те же правила, что и для подвода, только в обратной последовательности.

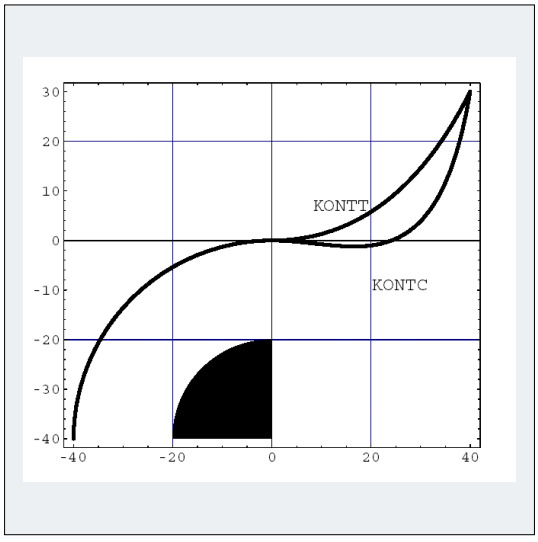
Подвод/отвод с KONTC

Подвод/отвод от точки контура осуществляется с постоянным изгибом. В точке контура не происходит скачка ускорения. Интерполяция траектории от исходной точки до точки контура осуществляется как полином.

Подвод/отвод с KONTT

Подвод/отвод от точки контура осуществляется по постоянной касательной. В точке контура может возникнуть скачок ускорения. Интерполяция траектории от исходной точки до точки контура осуществляется как полином.

Различия между KONTT и KONTC

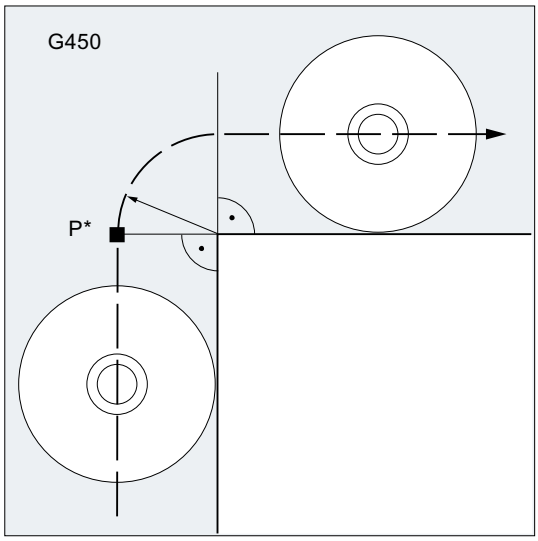


На этом рисунке представлены различия в поведении подвода/отвода для `KONTT` и `KONTC`. Окружность с радиусом 20 мм вокруг центра в X0 Y-40 исправляется инструментом с радиусом 20 мм на наружной стороне. Поэтому получается круговое движение центра инструмента с радиусом 40 мм. Конечная точка кадра отвода находится в X40 Y30. Переход между круговым кадром и кадром отвода лежит в нулевой точке. Из-за удлиненного постоянного изгиба при `KONTC` кадр отвода сначала выполняет движение с отрицательным компонентом Y. Часто это является нежелательным. Кадр отвода с `KONTT` не имеет такого поведения. Но в этом случае на переходе кадра возникает скачок ускорения.

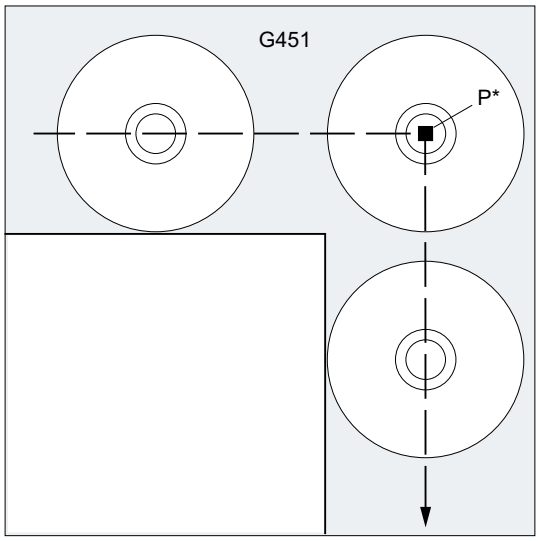
Если кадр `KONTT` или `KONTC` является не кадром отвода, а кадром подвода, то получается точно такой же контур, который лишь проходится в противоположном направлении.

3.10.3 Коррекция на наружных углах (G450, G451, DISC)

С помощью команды G450 или G451 при включенной коррекции радиуса инструмента (G41/G42) определяется ход исправленной траектории инструмента при обходе наружных углов:



С G450 центр инструмента обходит угол детали по дуге окружности с радиусом инструмента.



С G451 центр инструмента подводится к точке пересечения двух эквидистант, находящихся на расстоянии радиуса инструмента от запрограммированного контура. G451 действует только для прямых и окружностей.

Примечание

С G450/G451 также определяется путь подвода при активной KONT и точка подвода за контуром (см. "Подвод к контуру и отвод (NORM, KONT, KONTC, KONTT) (Страница 269)").

С помощью команды DISC переходные окружности при G450 могут быть искажены для создания острых углов контура.

Синтаксис

```
G450 [DISC=<значение>]
G451
```

Объяснение

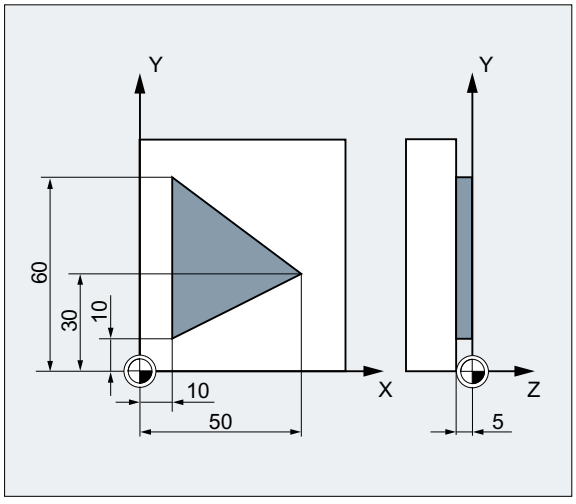
G450:	С G450 углы детали обходятся по круговой траектории.			
DISC:	Гибкое программирование круговой траектории при G450 (опция)			
	<значение>:	Тип:	INT	
		Диапазон значений:	0, 1, 2, ... 100	
		Объяснение:	0	Переходная окружность
			100	Точка пересечения эквидистант (теоретическое значение)
G451:	С G451 на углах детали выполняется подвод к точке пересечения обоих эквидистант. Инструмент выполняет свободное резание в углу детали.			

Примечание

DISC действует только при вызове G450, но может быть запрограммирована в предыдущем кадре без G450. Обе команды действуют модально.

Пример

В следующем примере на всех наружных углах вставляется переходный радиус (согласно программированию поведения на углах в кадре N30). Благодаря этому удается избежать остановки и свободного резания инструмента для смены направления.



Программный код	Комментарий
N10 G17 T1 G0 X35 Y0 Z0 F500	; Условия старта
N20 G1 Z-5	; Подача инструмента.
N30 G41 KONT **G450** X10 Y10	; Включить КРИ с режимом подвода/отвода KONT и **поведением на углах G450.**
N40 Y60	; Фрезерование контура.
N50 X50 Y30	
N60 X10 Y10	

Программный код	Комментарий
`N80 G40 X-20 Y50`	; Выключение режима коррекции, отвод на переходную окружность.
`N90 G0 Y100`	
`N100 X200 M30`	

Дополнительная информация

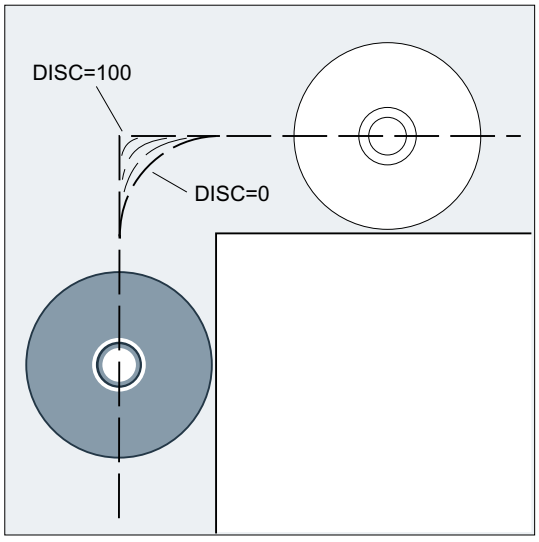
G450/G451

В промежуточной точке P* СЧПУ выполняет команды, как, к примеру, движения подачи или функции переключения. Эти операторы программируются в кадрах, находящихся между двумя кадрами, образующими угол.

Переходная окружность при `G450` с точки зрения передачи и обработки данных относится к последующей команде движения.

DISC

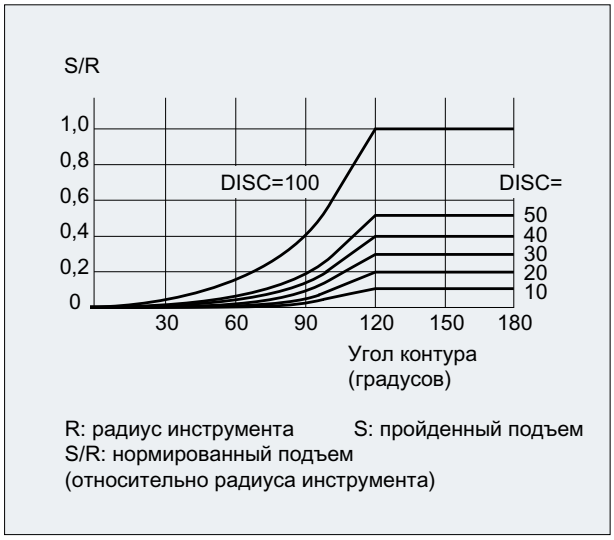
При указании значений `DISC` больше 0 промежуточные окружности представляются в вытянутом виде – при этом возникают переходные эллипсы или параболы или гиперболы:



Через машинные данные можно установить верхнее предельное значение – как правило DISC=50.

Характеристика движения

При включенной `G450` инструмент при острых углах контура и высоких значениях `DISC` на углах отводится от контура. При угле контура от 120°, контур обходится равномерно:

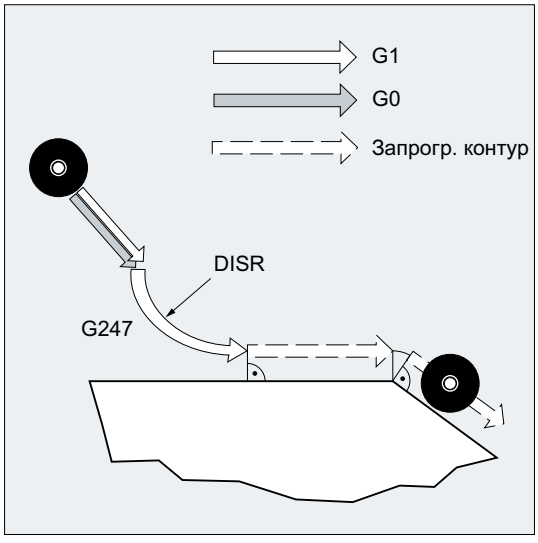


При включенной `G451` при острых углах контура из-за движений отвода может возникнуть избыточный холостой ход инструмента. Через машинные данные можно установить, чтобы в таких случаях автоматически осуществлялось переключение на переходную окружность.

3.10.4 Мягкий подвод и отвод

3.10.4.1 Подвод и отвод (G140 до G143, G147, G148, G247, G248, G347, G348, G340, G341, DISR, DISCL, DISRP, FAD, PM, PR)

Функция "Мягкий подвод и отвод (SAR)" служит для того, чтобы осуществить подвод по касательной в стартовой точке контура, независимо от положения исходной точки.



Функция используется преимущественно в комбинации с коррекцией радиуса инструмента.

При активации функции СЧПУ берет на себя задачу расчета промежуточных точек таким образом, чтобы переход в последующий кадр (или переход из предшествующего кадра при отводе) был бы выполнен согласно указанным параметрам.

Движение подвода состоит макс. из 4 вспомогательных движений. Стартовая точка движения в дальнейшем будет обозначаться как P_0, конечная точка как P_4. Между ними может находиться макс. три промежуточные точки P_1, P_2 и P_3. Точки P_0, P_3 и P_4 определены всегда. Промежуточные точки P_1 и P_2, в зависимости от параметрирования и геометрических свойств, могут отсутствовать. При отводе точки проходятся в обратной последовательности, т. е. начиная с P_4 и заканчивая в P_0.

Синтаксис

Мягкий подвод:

- по прямой:
  `G147 G340/G341 ... DISR=..., DISCL=..., DISRP=... FAD=...`
- по четверти круга/полукругу:
  `G247/G347 G340/G341 G140/G141/G142/G143 ... DISR=... DISCL=... DISRP=... FAD=...`

Мягкий отвод:

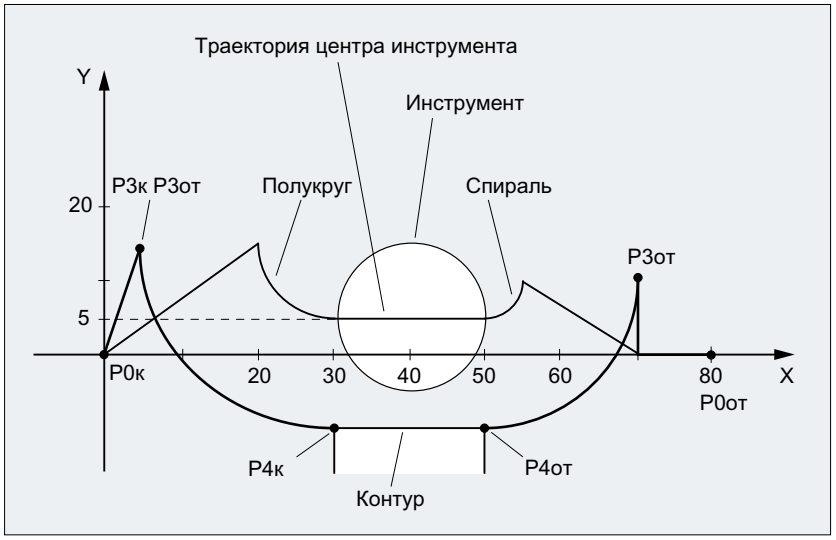
- по прямой:
  `G148 G340/G341 ... DISR=..., DISCL=..., DISRP=... FAD=...`
- по четверти круга/полукругу:
  `G248/G348 G340/G341 G140/G141/G142/G143 ... DISR=... DISCL=... DISRP=... FAD=...`

Объяснение

G147:	Подвод по прямой
G148:	Отвод по прямой
G247:	Подвод по четверти круга
G248:	Отвод по четверти круга
G347:	Подвод по полукругу
G348:	Отвод по полукругу
G340:	Пространственный подвод и отвод (первичная настройка)
G341:	Подвод и отвод в плоскости
G140:	Направление подвода и отвода в зависимости от текущей стороны коррекции (первичная настройка)
G141:	Подвод слева или отвод влево
G142:	Подвод справа или отвод вправо
G143:	Направление подвода и отвода в зависимости от относительного положения стартовой или конечной точки к направлению касательных

`DISR=...:`	1. При подводе и отводе по прямой (G147/G148): Расстояние от кромки фрезы от стартовой точки контура 2. При подводе и отводе по кругу (G247, G347/G248, G348): Радиус траектории центра инструмента **Внимание:** при REPOS по полукругу DISR обозначает диаметр окружности
`DISCL=...:`	Расстояние от конечной точки быстрого движения подачи до плоскости обработки DISCL=AC(...) указание абсолютной позиции конечной точки быстрого движения подачи
`DISCL=AC(...):`	указание абсолютной позиции конечной точки быстрого движения подачи
`DISRP:`	Расстояние от точки P1 (плоскость отвода) до плоскости обработки
`DISRP=AC(...):`	Указание абсолютного положения точки P1
`FAD=...:`	Скорость медленного движения подачи Запрограммированное значение действует согласно активному типу подачи (группа G 15).
`FAD=PM(...):`	Запрограммированное значение интерпретируется независимо от активного типа подачи как линейная подача (как G94).
`FAD=PR(...):`	Запрограммированное значение интерпретируется независимо от активного типа подачи как окружная подача (как G95).

Пример



- Мягкий подвод (кадр N20 активирован)
- Движение подвода по четверти круга (G247)
- Направление подвода не запрограммировано, действует G140, т. е. КРИ активна (G41)
- Смещение контура OFFN=5 (N10)

- Текущий радиус инструмента=10, таким образом, эффективный радиус коррекции для КРИ=15, радиус контура SAR=25, таким образом, радиус траектории центра инструмента будет равен DISR=10
- Конечная точка окружности получается из N30, т.к. в N20 запрограммирована только позиция Z
- Движение подачи
  - Из Z20 в Z7 (DISCL=AC(7)) ускоренным ходом
  - После в Z0 с FAD=200.
  - Круг подвода в плоскости X-Y и последующие кадры с F1500 (чтобы эта скорость стала активной в последующих кадрах, необходимо переписать активную G0 в N30 с G1, в ином случае обработка контура была бы продолжена с G0)
- Мягкий отвод (кадр N60 активирован)
- Движение отвода по четверти круга (G248) и спирали (G340)
- FAD не запрограммирована, так как не имеет значения при G340
- Z=2 в стартовой точке; Z=8 в конечной точке, т.к. DISCL=6
- При DISR=5 радиус контура SAR=20, радиус траектории центра инструмента=5

Движения перемещения из Z8 в Z20 и движение параллельно плоскости X-Y к X70 Y0.

Программный код	Комментарий
`$TC_DP1[1,1]=120`	; определение инструмента `T1/D1`
`$TC_DP6[1,1]=10`	; радиус
`N10 G0 X0 Y0 Z20 G64 D1 T1 OFFN=5`	; (к `P0`)
`N20 G41 G247 G341 Z0 DISCL=AC(7) DISR=10 F1500 FAD=200`	; Подвод (к `P3`)
`N30 G1 X30 Y-10`	; (к `P4`)
`N40 X40 Z2`	
`N50 X50`	; (от `P4`)
`N60 G248 G340 X70 Y0 Z20 DISCL=6 DISR=5 G40 F10000`	; Отвод (от `P3`)
`N70 X80 Y0`	; (от `P0`)
`N80 M30`	

Дополнительная информация

Выбор контура подвода или отвода

Выбор контура подвода или отвода осуществляется с помощью соответствующей G-команды 2-й группы G:

G147:	Подвод по прямой
G247:	Подвод по четверти круга
G347:	Подвод по полукругу
G148:	Отвод по прямой
G248:	Отвод по четверти круга
G348:	Отвод по полукругу

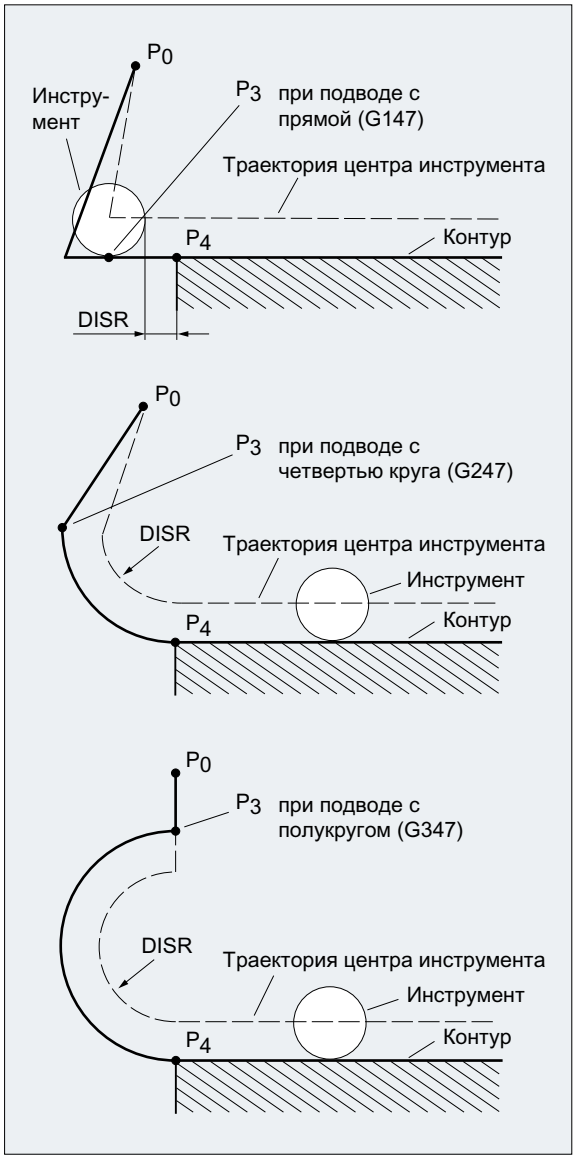


Рисунок 3-20 Движения подвода при одновременной активации коррекции радиуса инструмента

Выбор направления подвода или отвода

Определение направления подвода и отвода с помощью коррекции радиуса инструмента (G140, первичная настройка) при положительном радиусе инструмента:

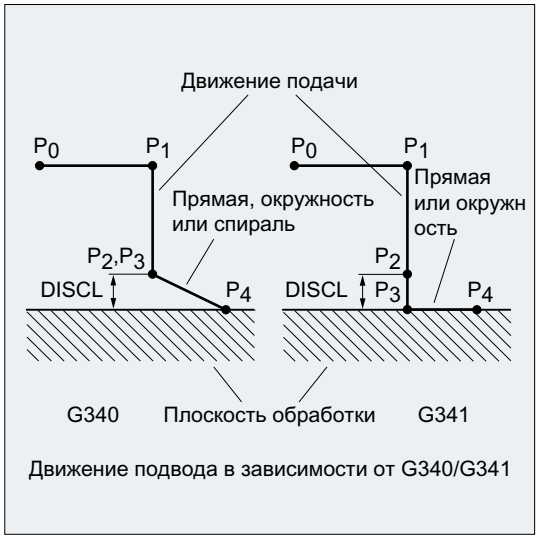
- G41 активна → подвод слева
- G42 активна → подвод справа

Прочие возможности подвода доступны с помощью G141, G142 и G143.

Эти G-команды имеют значение, только если контур подвода является четвертью круга или полукругом.

Подразделение движения от стартовой к конечной точке (G340 и G341)

В любом случае движения состоят из одной или нескольких прямых, а также, в зависимости от G-команды для определения контура подвода, из дополнительной прямой или четверти круга / полукруга. 2 варианта разделения пути представлены на рисунке ниже:



Движение подвода в зависимости от G340/G341

G340:	Подвод по прямой от точки P_0 к точке P_1. Эта прямая параллельна плоскости обработки, если параметр DISRP не был запрограммирован. Подача вертикально к плоскости обработки от точки P_1 к точке P_3 на определенное через параметр DISCL безопасное расстояние к плоскости обработки. Подвод к конечной точке P4 по определенной через G-команду второй группы кривой (прямая, окружность, спираль). Если активна G247 или G347 (четверть круга или полукруг) и стартовая точка P_3 не лежит в определенной через конечную точку P_4 плоскости обработки, то вместо окружности вставляется спираль. Точка P_2 не определена или совпадает с P_3. Плоскость окружности или ось спирали при этом определяется через активную в SAR-кадре плоскость (G17/G18/G19), т. е. последующим кадром для определения окружности используется не сама стартовая касательная, а ее проекция в активную плоскость. Движение от точки P_0 к точке P_3 выполняется в двух прямых со скоростью, которая действовала до SAR-кадра.
G341:	Подвод по прямой от точки P_0 к точке P_1. Эта прямая параллельна плоскости обработки, если параметр DISRP не был запрограммирован. Подача вертикально к плоскости обработки от точки P_1 до определенного через параметр DISCL безопасного расстояния к плоскости обработки в точке P_2. Подача вертикально к плоскости обработки от точки P_2 к точке P_3. Подвод к конечной точке по определенной через G-команду второй группы кривой. P_3 и P_4 лежат в плоскости обработки, поэтому при G247 или G347 никогда не вставляется спираль, а всегда окружность.

Во всех случаях, когда положение активной плоскости допускает G17/G18/G19 (плоскость окружности, ось спирали, движение подачи вертикально к активной плоскости), учитывается возможно активный вращающий фрейм.

Длина прямой подвода или радиуса в случае окружностей подвода (DISR)

- Подвод/отвод по прямой
  DISR указывает расстояние от кромки фрезы до стартовой точки контура, т. е. длина прямой при активной КРИ получается из суммы радиуса инструмента и запрограммированного значения DISR. Учитывается только положительный радиус инструмента.
  Полученная длина прямой должна быть положительной, т. е. отрицательные значения для DISR допускаются, пока величина DISR меньше, чем радиус инструмента.
- Подвод/отвод по кругу
  DISR указывает радиус траектории центра инструмента. Если КРИ активирована, то создается круг с таким радиусом, чтобы и в этом случае получалась траектория центра инструмента с запрограммированным радиусом.

Расстояние от точки P2 до плоскости обработки (DISCL)

Если позиция точки P_2 на оси вертикально к плоскости окружности должна быть указана абсолютно, то значение программируется в форме `DISCL=AC(...)`.

Для DISCL=0:

- Для G340: Все движение подвода состоит только из двух кадров (P_1, P_2 и P_3 совпадают). Контур подвода образуется от P_1 к P_4.
- Для G341: Все движение подвода состоит из трех кадров (P_2 и P_3 совпадают). Если P_0 и P_4 лежат в одной плоскости, то получается только два кадра (движение подачи от P_1 к P_3 отсутствует).
- Контролируется, чтобы определенная через DISCL точка лежала бы между P_1 и P_3, т. е. при всех движениях, имеющих один компонент вертикально к плоскости обработки, этот компонент должен иметь тот же знак.
- При определении реверсирования разрешается определенный через машинные данные MD20204 $MC_WAB_CLEARANCE_TOLERANCE допуск.

Расстояние от точки P1 (плоскость отвода) до плоскости обработки (DISRP)

Если позиция точки P_1 на оси вертикально к плоскости обработки должна быть указана абсолютно, то значение программируется в форме `DISRP=AC(...)`.

Если этот параметр не запрограммирован, то расстояние от точки P_1 до плоскости обработки идентично таковому от точки P_0, т. е. прямая подвода $P_0 \rightarrow P_1$ параллельна плоскости обработки.

Контролируется, чтобы определенная через DISCL точка лежала бы между P_0 и P_2, т. е. при всех движениях, имеющих один компонент вертикально к плоскости обработки (движения подачи, движение подвода от P_3 к P_4), этот компонент должен иметь тот же знак. Реверсирование не допускается. При необходимости выводится аварийное сообщение.

При определении реверсирования разрешается определенный через машинные данные MD20204 $MC_WAB_CLEARANCE_TOLERANCE допуск. Если P_1 лежит вне определенной через P_0 и P_2 области, но отклонение меньше или равно этому допуску, то предполагается, что P_1 лежит в определенной через P_0 или P_2 плоскости.

Программирование конечной точки

Конечная точка программируется, как правило, с помощью X... Y... Z...

Программирование конечной точки контура при подводе значительно отличается от такового при отводе. Поэтому оба случая рассматриваются здесь отдельно.

Программирование конечной точки P4 при подводе

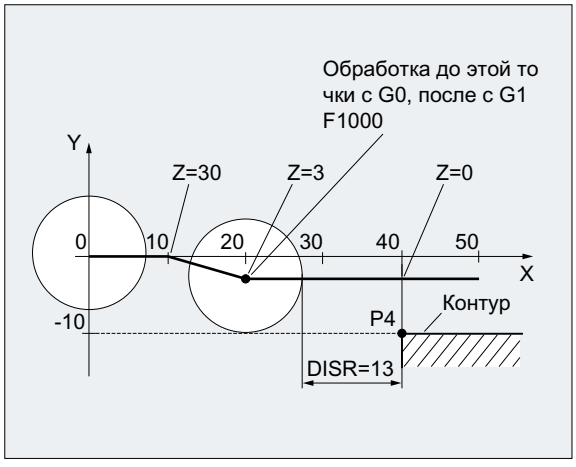
Конечная точка P_4 может быть запрограммирована в самом SAR-кадре. В качестве альтернативы можно определить P_4 через конечную точку следующего кадра перемещения. Между кадром WAB и следующим кадром перемещения могут быть вставлены другие кадры без движения гео-осей.

Пример:

Программный код	Комментарий
`$TC_DP1[1,1]=120`	; фрезерный инструмент `T1/D1`
`$TC_DP6[1,1]=7`	; Инструмент с радиусом 7 мм
`N10 G90 G0 X0 Y0 Z30 D1 T1`	
`N20 X10`	
`N30 G41 G147 DISCL=3 DISR=13 Z=0 F1000`	
`N40 G1 X40 Y-10`	
`N50 G1 X50`	
...	

`N30/N40` может быть заменена на:
`N30 G41 G147 DISCL=3 DISR=13 X40 Y-10 Z0 F1000`

или
`N30 G41 G147 DISCL=3 DISR=13 F1000`
`N40 G1 X40 Y-10 Z0`



Программирование конечной точки P0 при отводе

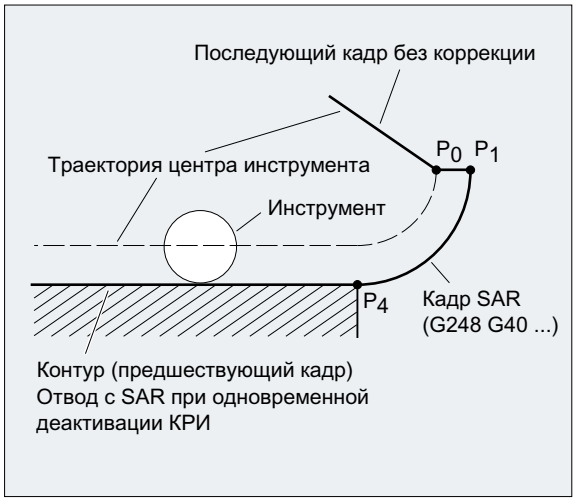
При отводе программирование конечной точки SAR-контура в последующем кадре не предусмотрено, т. е. конечная позиция всегда извлекается из самого SAR-кадра,

независимо от количества запрограммированных осей. При определении конечной точки здесь различаются три следующих случая:

1. Гео-ось в SAR-кадре не запрограммирована. В этом случае контур заканчивается в точке P_1 (если запрограммирована DISRP), в точке P_2 (если запрограммирована DISCL, а не DISRP), или в точке P_3 (если не запрограммированы ни DICLS, ни DISRP). Позиция в осях, образующих плоскость обработки, получается из контура отвода (конечная точка прямой или окружности). Соответствующий вертикальный осевой компонент определяется через DISCL или DISPR. Если в этом случае как DISCL=0, так и DISRP=0, то движение полностью происходит в плоскости, т. е. точки P_0 до P_3 совпадают.
2. В SAR-кадре запрограммирована только ось вертикально к плоскости обработки. В этом случае контур оканчивается в точке P_0. Если запрограммирована DISRP (т. е. обе точки P_0 и P_1 не совпадают), то прямая $P_1 \rightarrow P_0$ располагается вертикально к плоскости обработки. Позиции обеих оставшихся осей получаются как в 1.
3. Запрограммирована как минимум одна ось плоскости обработки. Возможно отсутствующая вторая ось плоскости обработки модально добавляется из ее последней позиции в предшествующем кадре.

Позиция оси вертикально к плоскости обработки формируется - в зависимости от того, запрограммирована ли эта ось или нет - как в 1 или 2. Оформленная таким образом позиция определяет конечную точку P_0. Если SAR-кадр отвода одновременно является кадром деактивации коррекции радиуса инструмента, то в первых двух случаях дополнительный компонент хода вставляется в плоскости обработки от P_1 к P_0 так, что при деактивации коррекции радиуса инструмента в конце контура отвода не получается движения, т. е. эта точка здесь определяет не позицию на исправляемом контуре, а центр инструмента. В случае три отдельной обработки отмены коррекции радиуса инструмента не требуется, т.к. запрограммированная точка P_0 уже непосредственно определяет позицию центра инструмента на конце всего контура.

Поведение в случаях 1 и 2, т. е. при не явно запрограммированной конечной точке в плоскости обработки при одновременной отмене коррекции радиуса инструмента, представлено на рис. ниже:

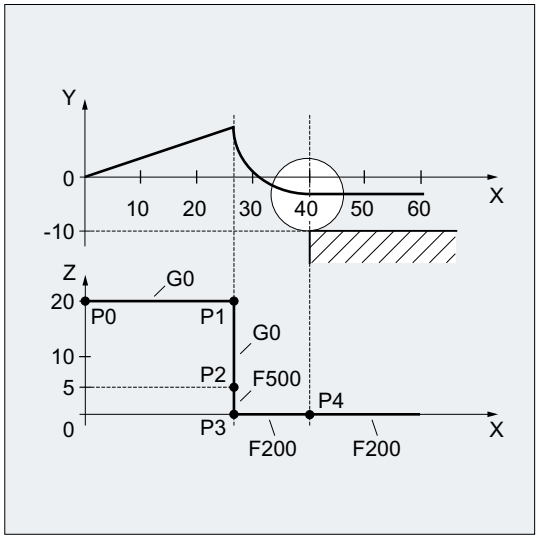


Скорости подвода или отвода

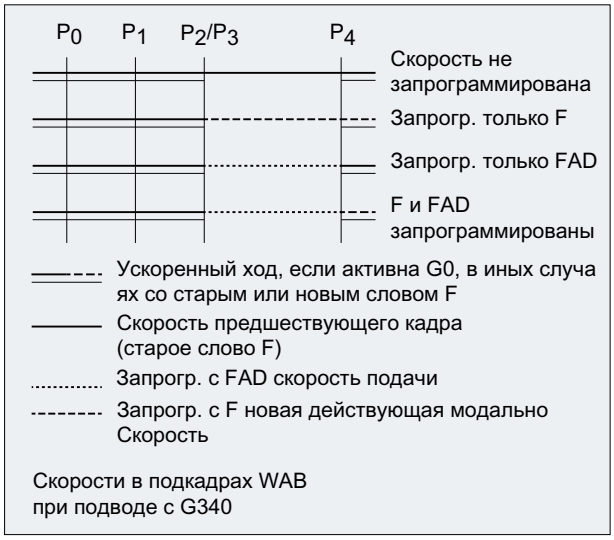
- Скорость предыдущего кадра (G0)
  С этой скоростью выполняются все движения от P_0 до P_2, т. е. движения параллельно плоскости обработки и часть движения подачи до безопасного расстояния.
- Программирование с FAD
  Указание скорости подачи для
  - G341: Движение подачи вертикально к плоскости обработки от P_2 к P_3
  - G340: от точки P_2 или P_3 к P_4
    Если FAD не программируется, то эта часть контура также проходится с действующей модельно скоростью предыдущего кадра, если в SAR-кадре не запрограммировано F-слово.
- Запрограммированная подача F
  Это значение подачи действует от P_3 или P_2, если FAD не запрограммирована. Если в кадре SAR слово F не программируется, то действует скорость предыдущего кадра.

Пример:

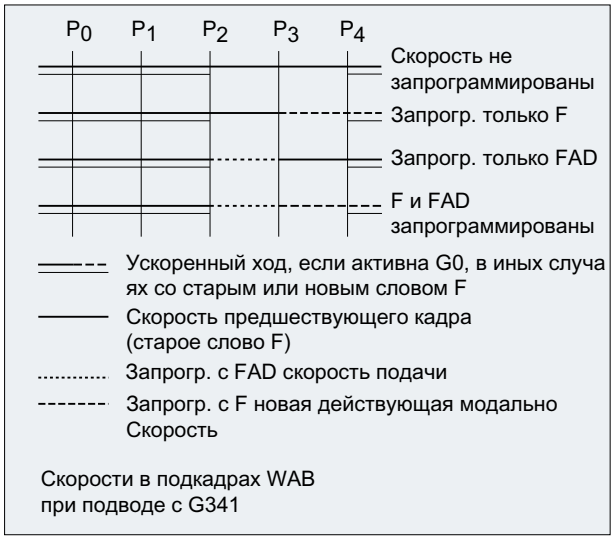
Программный код	Комментарий
`$TC_DP1[1,1]=120`	; фрезерный инструмент T1/D1
`$TC_DP6[1,1]=7`	; инструмент с радиусом 7 мм
`N10 G90 G0 X0 Y0 Z20 D1 T1`	
`N20 G41 G341 G247 DISCL=AC(5) DISR=13 FAD 500 X40 Y-10 Z=0 F200`	
`N30 X50`	
`N40 X60`	
...	



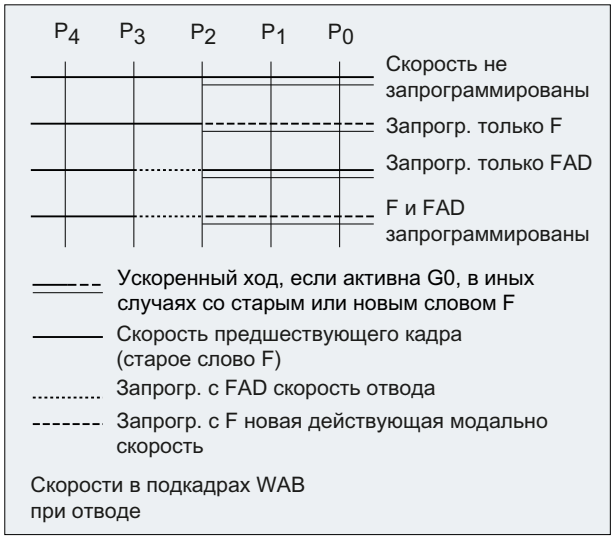
При отводе роли модально действующей подачи из предыдущего кадра и запрограммированного в кадре SAR значения подачи меняются местами, т. е. перемещение по самому контуру отвода осуществляется со старой подачей, заново запрограммированная с помощью слова F скорость действует соответственно от P_2 до P_0.



Скорости в подкадрах WAB
при подводе с G340



Скорости в подкадрах WAB
при подводе с G341



Чтение позиций

Точки P_3 и P_4 при подводе могут считываться как системные переменные в WCS.

- $P_APR: чтение P
- $_3$ (точка старта)
- $P_AEP: чтение P
- $_4$ (начальная точка контура)
- $P_APDV: чтение, содержат ли $P_APR и $P_AEP действительные значения

3.10.4.2 Подвод и отвод с расширенными стратегиями отвода (G460, G461, G462)

В определенных геометрических особых случаях, в отличие от прежнего выполнения с включенным контролем столкновений для кадра подвода и отвода, необходимы специальные расширенные стратегии подвода и отвода при активации или деактивации коррекции радиуса инструмента. Так, к примеру, контроль столкновения может привести к тому, что участок на контуре будет обработан не полностью, см. рисунок ниже:

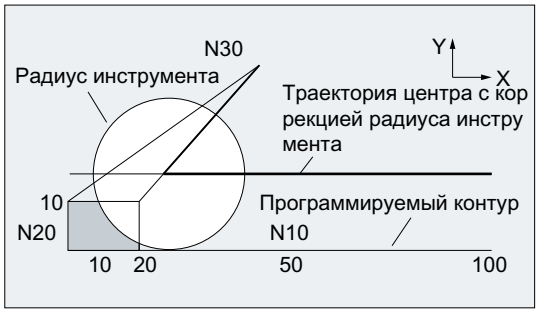


Рисунок 3-21 Характеристика отвода при G460

Синтаксис

```
G460
G461
G462
```

Значение

G460:	Как раньше (включение контроля столкновения для кадра подвода и отвода)
G461:	Вставка окружности в кадре КРИ, если невозможна точка пересечения, центр которой находится в конечной точке кадра без коррекции, и чей радиус равен радиусу инструмента. До точки пересечения обработка осуществляется по **вспомогательной окружности** вокруг конечной точки контура (т. е. до конца контура).
G462:	Вставка прямой в кадре КРИ, если точка пересечения невозможна, кадр удлиняется через его конечную касательную (стандартная установка) Обработка осуществляется до **удлинения** последнего элемента контура (т. е. почти до конца контура).

Примечание

Характеристика подвода симметрична характеристике отвода.

Характеристика подвода или отвода определяется состоянием G-команды в кадре подвода или отвода. Таким образом, характеристика повода может быть установлена независимо от характеристики отвода.

Примеры

Пример 1: Характеристика отвода при G460

В дальнейшем описывается ситуация при деактивации коррекции радиуса инструмента. Поведение при подводе полностью аналогично.

Программный код	Комментарий
G42 D1 T1	; Радиус инструмента 20 мм
...	
G1 X110 Y0	
N10 X0	
N20 Y10	
N30 G40 X50 Y50	

Пример 2: Подвод при G461

Программный код	Комментарий
N10 $TC_DP1[1,1]=120	; Тип инструмента "фреза"
N20 $TC_DP6[1,1]=10	; Радиус инструмента

Программный код	Комментарий
N30 X0 Y0 F10000 T1 D1	
N40 Y20	
N50 G42 X50 Y5 G461	
N60 Y0 F600	
N70 X30	
N80 X20 Y-5	
N90 X0 Y0 G40	
N100 M30	

Дополнительная информация

G461

Если точка пересечения последнего кадра КРИ с предыдущим кадром невозможна, то кривая смещения этого кадра продлевается на окружность, центр которой находится в конечной точке кадра без коррекции и радиус которой равен радиусу инструмента.

СЧПУ пытается рассечь эту окружность одним из предыдущих кадров.

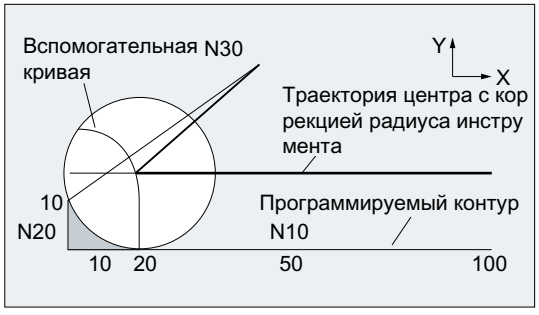


Рисунок 3-22 Характеристика отвода при G461

Контроль столкновений CDON, CDOF

Здесь при активной CDOF (см. раздел "Контроль столкновений, CDON, CDOF) поиск отменяется, если точка пересечения была найдена, т.е. не проверяется, существуют ли точки пересечения с более старыми кадрами.

При активной CDON и в том случае, если точка пересечения была найдена, продолжается поиск других точек пересечения.

Найденная таким образом точка пересечения является новой конечной точкой предыдущего кадра и стартовой точкой кадра деактивации. Вставленная окружность служит только для вычисления точки пересечения и не вызывает движения перемещения.

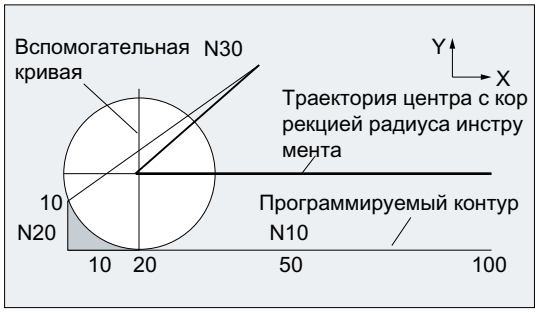
Примечание

Если точка пересечения не найдена, то выводится ошибка 10751 (опасность столкновения).

G462

Если точка пересечения последнего кадра КРИ с предыдущим кадром невозможна, то при отводе с G462 (по умолчанию) в конечной точке последнего кадра с коррекцией радиуса инструмента вставляется прямая (кадр продлевается своей конечной касательной).

Поиск точки пересечения тогда осуществляется идентично G461.



Поведение отвода при G462 (см. пример)

При G462 образованный в программе-образце из N10 и N20 угол выбирается не настолько, как это было бы возможно с используемым инструментом. Но это поведение может быть все же необходимым, чтобы не повредить подконтур (отличный от запрограммированного контура) в примере слева от N20 и при значениях y, больших 10 мм.

Угловые параметры при KONT

Если KONT активна (обход контура в стартовой или конечной точке), то различается, находится ли конечная точка перед или за контуром.

- **Конечная тока перед контуром**
  Если конечная точка находится перед контуром, то характеристика отвода аналогична NORM. Это свойство не изменяется и в том случае, если последний кадр контура при G451 продлевается прямой или окружностью. Поэтому дополнительные стратегии обхода, чтобы избежать повреждения контура вблизи конечной точки контура, не нужны.

- **Конечная тока за контуром**
  Если конечная точка находится за контуром, то всегда, в зависимости от G450 / G451, вставляется окружность или прямая. G460 - G462 тогда не имеет значения. Если последний кадр перемещения в этой ситуации не имеет точки пересечения с предшествующим кадром, то теперь может получиться точка пересечения со вставленным элементом контура или с участком прямой от конечной точки обходной окружности до запрограммированной конечной точки.
  Если вставленным элементом контура является окружность (G450), и она образует с предшествующим кадром точку пересечения, то она идентична точке пересечения, которая получилась бы при NORM и G461. В остальном нужно пройти дополнительный сегмент круга. Вычисления точки пересечения для линейной части кадра отвода более не требуется.
  Во втором случае, когда точка пересечения вставленного элемента контура с предшествующими кадрами не найдена, то выполняется перемещение на точку пересечения между прямой отвода и предшествующим кадром.
  Таким образом, при активной G461 или G462 поведение отличается от такового при G460 только в том случае, если либо активна NORM, либо поведение при KONT, обусловленное геометрически, идентично таковому при NORM.

3.10.5 Включение/выключение контроля столкновений ("Распознавание бутылочного горлышка") (CDON, CDOF, CDOF2)

Контроль столкновений ("Распознавание бутылочного горлышка") при активной коррекции радиуса инструмента включается/выключается в программе ЧПУ командами группы G 23.

Синтаксис

```
G41/G42 CDON
...
CDOF/CDOF2
```

Объяснение

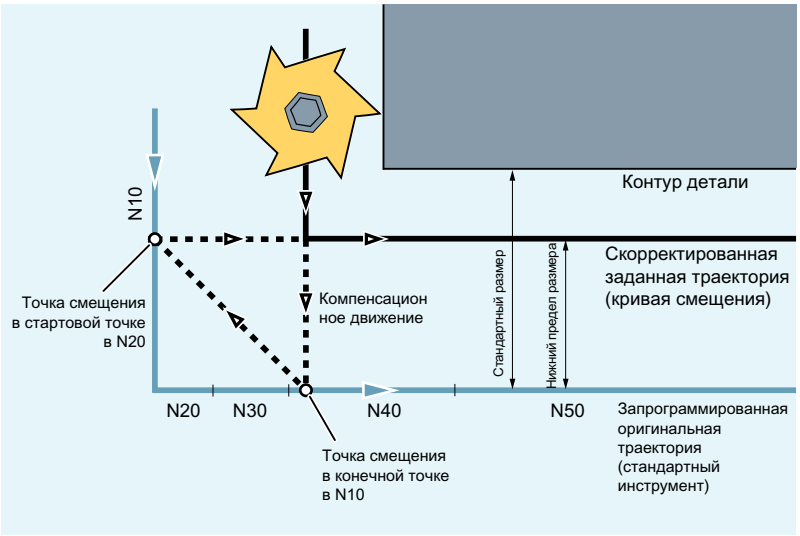
CDON:	Включение контроля столкновений ("Распознавание бутылочного горлышка") Посредством CDON через настраиваемое (MD20240) количество кадров проверяется, пересекаются ли пути инструментов не соседних или не последовательных кадров. Благодаря этому удается своевременно распознавать столкновения и активно предотвращать их при помощи СЧПУ.
CDOF:	Выключение контроля столкновений ("Распознавание бутылочного горлышка") Посредством CDOF для текущего кадра выполняется поиск общей точки пересечения с предыдущим кадром перемещения (на внутренних углах), при необходимости и в более старых кадрах. Если обнаруживается точка пересечения между соседними кадрами, дальнейшие кадры не рассматриваются. На наружных углах точка пересечения между двумя следующими друг за другом кадрами обнаруживается всегда. **Указание:** Посредством CDOF удается избежать ошибочного распознавания сужений, вызванного, например, отсутствующей информацией в программе ЧПУ.
CDOF2:	Выключение контроля столкновения для периферийного фрезерования 3D Посредством CDOF2 определяется направление коррекции инструмента из соседних частей кадра. CDOF2 действует только при периферийном фрезеровании 3D (CUT3Dx) и имеет и при всех других режимах обработки (например, CUT2D или CUT3DFD) значение, идентичное CDOF.

Пример действия контроля столкновений

Программа ЧПУ описывает траекторию центра стандартного инструмента. Контур для используемого на соответствующий момент времени инструмента дает нижний предел размера: он увеличен на следующем изображении для лучшего пояснения геометрических отношений.

Кроме того, в примере СЧПУ осуществляет опережающее рассмотрение только на три кадра.

MD20240 $MC_CUTCOM_MAXNUM_CHECK_BLOCKS = 3



Поскольку точка пересечения существует только между кривыми смещения двух кадров N10 и N40, оба кадра N20 и N30 следовало бы исключить. В примере кадр N40 еще не известен СЧПУ, если в заключении должен обрабатываться N10. Тем самым может быть пропущен только один единственный кадр.

При активной CDOF2 выполняется представленное на изображении движение компенсации, останова нет. В этой ситуации активная CDOF или CDON вызвали бы появление аварийного сообщения.

3.10.6 Коррекция инструмента 2 1/2 D (CUT2D, CUT2DD, CUT2DF, CUT2DFD)

Коррекция радиуса инструмента 2½ D применяется в том случае, если для обработки наклонных поверхностей выполняется не центровка инструмента, а поворот **заготовки**. Активацию осуществляют командами `CUT2D`, `CUT2DD`, `CUT2DF` или `CUT2DFD`.

Коррекция длины инструмента

Коррекция длины инструмента всегда вычисляется относительно зафиксированной в пространстве, не повернутой рабочей плоскости.

Коррекция радиуса контурного инструмента 2½ D

Коррекцию радиуса контурного инструмента 2½ D активируют в том случае, если вместе с `CUT2D`, `CUT2DD`, `CUT2DF` или `CUT2DFD` запрограммирована одна из команд `G41` (коррекция радиуса инструмента слева от контура) или `G42` (коррекция радиуса инструмента справа от контура). Она служит для автоматического выбора резцов для не вращательно-симметричных инструментов, с помощью которых возможна посегментная обработка отдельных участков контура.

Примечание

При не активной коррекции радиуса инструмента 2½ D поведение контурного инструмента идентично обычному инструменту, состоящему только из первого резца.

Коррекция радиуса инструмента 2½ D относительно дифференциального инструмента

Коррекция радиуса инструмента 2½ D относительно дифференциального инструмента активируется командой `CUT2DD` или `CUT2DFD`. Ее следует применять, если контур запрограммирован относительно траектории центра дифференциального инструмента, а обработка осуществляется отличающимся инструментом. В этом случае при расчете коррекции радиуса инструмента 2½ D учитывают только значение износа радиуса активного инструмента ($TC_DP_15) и, возможно, запрограммированного смещения коррекции инструмента OFFN (Страница 260) и TOFFR (Страница 88). Базовый радиус ($TC_DP6) активного инструмента **не** учитывается.

Синтаксис

```
CUT2D
CUT2DD
CUT2DF
CUT2DFD
```

Объяснение

`CUT2D`:	Активация коррекции радиуса 2½ D
`CUT2DD`:	Активация коррекции радиуса 2½ D относительно дифференциального инструмента
`CUT2DF`:	Активация коррекции радиуса 2½ D, коррекция радиуса инструмента относительно текущего фрейма или наклонной плоскости
`CUT2DFD`:	Активация коррекции радиуса 2½ D по дифференциальному инструменту относительно текущего фрейма или наклонной плоскости

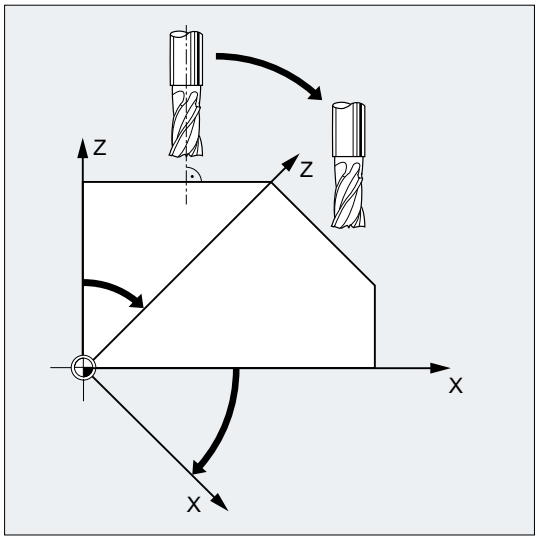
Дополнительная информация

Контурные инструменты

- Разрешение
  Деблокирование коррекции радиуса контурного инструмента осуществляется для отдельного канала через:
  MD28290 $MC_MM_SHAPED_TOOLS_ENABLE
- Тип инструмента
  Типы контурных инструментов задаются для отдельного канала через:
  MD20370 $MC_SHAPED_TOOL_TYPE_NO
- Резцы
  Каждому контурному инструменту в произвольной последовательности можно назначить некоторое количество резцов (D-номера). Максимальное количество резцов на инструмент задается параметром:
  MD18106 $MN_MM_MAX_CUTTING_EDGE_PERTOOL
  Первым резцом контурного инструмента является резец, выбираемый при активации инструмента. Если, например, в программе командами `T3 D5` активирован пятый резец (D5) третьего инструмента (T3), то D5 и последующие резцы будут определять контурный инструмент либо частично, либо все вместе. Находящиеся перед D5 резцы будут проигнорированы.

Коррекция радиуса инструмента 2½ D без поворота плоскости коррекции (CUT2D, CUT2DD)

В случае программирования фрейма с одним поворотом при выполнении `CUT2D` или `CUT2DD` плоскость, в которой происходит коррекция радиуса инструмента (плоскость коррекции), **не поворачивается вместе с ним**. Коррекция радиуса инструмента рассчитывается относительно **не повернутой** рабочей плоскости (`G17`, `G18`, `G19`). Коррекция длины инструмента работает относительно плоскости коррекции.

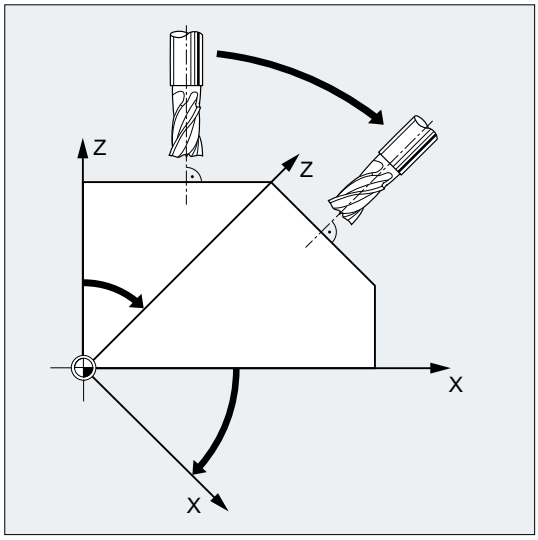


Для обработки наклонных поверхностей значения коррекции инструмента должны быть соответственно определены или вычислены с использованием функционала для "Коррекции длин инструмента для ориентируемых инструментов".

Коррекция радиуса инструмента 2½ D с поворотом плоскости коррекции (CUT2DF, CUT2DFD)

В случае программирования фрейма с одним поворотом при выполнении `CUT2DF` или `CUT2DFD` плоскость, в которой происходит коррекция радиуса инструмента (плоскость коррекции), **поворачивается вместе с ним**. Коррекция радиуса инструмента рассчитывается относительно **повернутой** рабочей плоскости (`G17`, `G18`, `G19`). Однако коррекция длин инструмента продолжает действовать относительно **не повернутой** рабочей плоскости.

Условие: Машина должна допускать ориентацию инструмента перпендикулярно повернутой рабочей плоскости и настройку для обработки.



Примечание

Коррекция длин инструмента продолжает действовать относительно не повернутой рабочей плоскости.

Дополнительная информация: см.: Описание функций – Инструменты.

3.10.7 Постоянная коррекция радиуса инструмента (CUTCONON, CUTCONOF)

Функция "Постоянная коррекция радиуса инструмента" служит для подавления коррекции радиуса инструмента для определенного числа кадров, но при этом как смещение сохраняется образованная через коррекцию радиуса инструмента в предшествующих кадрах разница между запрограммированной и фактически пройденной траекторией центра инструмента. Преимущества этой функции проявляются тогда, когда, к примеру, при строчечном фрезеровании в точках возврата необходимо несколько кадров перемещения, но созданные коррекцией радиуса инструмента контуры (стратегии обхода) являются нежелательными. Она может использоваться независимо от типа коррекции радиуса инструмента ($2^1/_2$D, торцовое фрезерование 3D, периферийное фрезерование 3D).

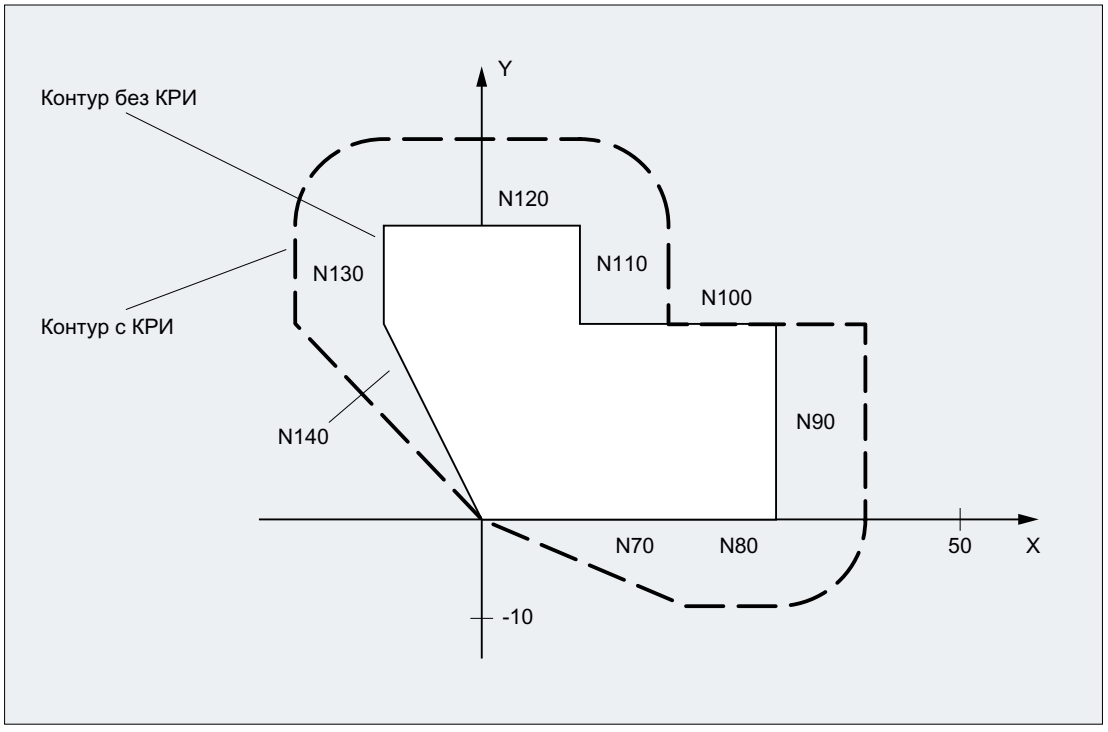
Синтаксис

```
CUTCONON
CUTCONOF
```

Значение

CUTCONON:	Команда для включения функции "Постоянная коррекция радиуса инструмента"
CUTCONOF:	Команда для выключения функции "Постоянная коррекция радиуса инструмента"

Пример



Программный код	Комментарий
N10	; Определение инструмента d1.
N20 $TC_DP1[1,1]= 110	; тип
N30 $TC_DP6[1,1]= 10.	; радиус
N40	
N50 X0 Y0 Z0 G1 G17 T1 D1 F10000	
N60	
N70 X20 G42 NORM	
N80 X30	
N90 Y20	
N100 X10 CUTCONON	; Включение подавления коррекции.
N110 Y30 KONT	; При отключении подавления контура при необходимости вставить обходную окружность.
N120 X-10 CUTCONOF	
N130 Y20 NORM	; Нет обходной окружности при отключении КРИ.
N140 X0 Y0 G40	

Программный код	Комментарий
`N150 M30`	

Дополнительная информация

Обычно перед активацией подавления коррекции коррекция радиуса инструмента уже активна, и она еще активна, когда подавление коррекции снова деактивируется. В последнем кадре перемещения перед `CUTCONON` выполняется движение на точку смещения в конечной точке кадра. Все последующие кадры, в которых активно подавление коррекции, проходятся без коррекции. Но при этом они смещаются на вектор от конечной точки последнего кадра коррекции к его точке смещения. Тип интерполяции этих кадров (линейная, круговая, полиномиальная) может быть любым.

Кадр деактивации подавления коррекции, т.е. кадр, содержащий `CUTCONOF`, подвергается обычной коррекции. Он начинается в точке смещения стартовой точки. Между конечной точкой предшествующего кадра, т.е. последнего запрограммированного кадра перемещения с активной `CUTCONON`, и этой точкой вставляется линейный кадр.

Круговые кадры, у которых плоскость окружности стоит вертикально на плоскости коррекции (вертикальные окружности), обрабатываются так, как если бы в них была бы запрограммирована `CUTCONON`. Эта не явная активация подавления коррекции автоматически отменяется в первом кадре перемещения, содержащим движение перемещения в плоскости коррекции и не являющимся такой окружностью. Вертикальные окружности такого плана могут встречаться только при периферийном фрезеровании.

3.10.8 Инструменты с релевантным положением резцов

У инструментов с релевантным положением резцов (токарный и шлифовальный инструмент, типы инструмента 400–599; см. главу "Нормирование знака износа") переход с G40 в G41/G42 или наоборот рассматривается как смена инструмента. Это приводит при активной трансформации (к примеру, TRANSMIT) к остановке предварительной обработки (остановка декодирования) и тем самым, при определенных обстоятельствах, к отклонениям от предполагаемого контура детали.

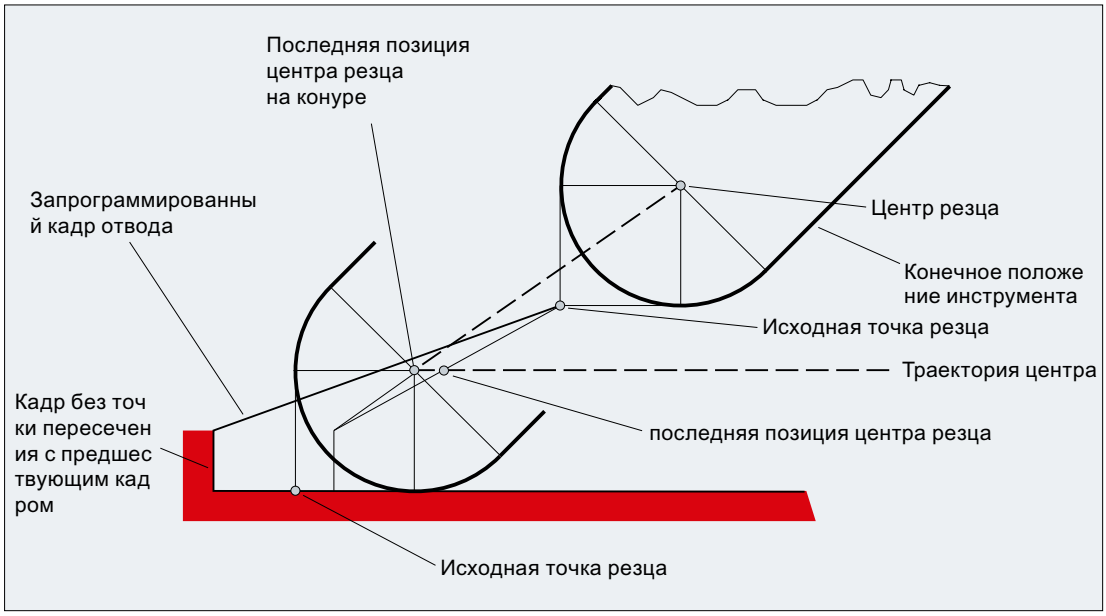
Эта первоначальная функциональность изменяется относительно:

1. Остановка предварительной обработки при TRANSMIT
2. Вычисление точек пересечения при подводе или отводе с KONT
3. Смена инструмента при активной коррекции радиуса инструмента
4. Коррекция радиуса инструмента с переменной ориентацией инструмента при трансформации

Дополнительная информация

Первоначальная функциональность была изменена следующим образом:

- Переход с G40 на G41/G42 и наоборот более не рассматривается как смена инструмента. Поэтому TRANSMIT более не вызывает остановки предварительной обработки.
- Для вычисления точек пересечения с кадром подвода или отвода используется прямая между центрами резцов в начале и конце кадра. Разница между исходной точкой резцов и центром резцов накладывается на это движение.
  При подводе или отводе с KONT (инструмент обходит точку контура; см. предыдущий раздел "Подвод к контуру и отвод") наложение осуществляется в линейном подкадре движения подвода или отвода. Поэтому геометрические отношения идентичны для инструментов с и без релевантного положения резцов. Отличия от данного поведения возникают только в относительно редких случаях, когда кадр подвода или отвода образует точку пересечения с не соседним кадром перемещения, см. следующий рисунок:



- Смена инструмента при активной коррекции радиуса инструмента, при которой изменяется расстояние между центром резца и исходной точкой резца, в круговых кадрах с кадрах перемещения с рациональными полиномами с номинальным порядком > 4 запрещена. Для других типов интерполяции, смена, в отличии от прежнего состояния, допускается и при активной трансформации (к примеру, TRANSMIT).
- При коррекции радиуса инструмента с переменной ориентацией инструмента трансформация исходной точки резца на центр резца более не может быть реализована через простое смещение нулевой точки. Поэтому инструменты с релевантным положением резцов запрещены при периферийном фрезеровании 3D (ошибка).

Примечание

Тема не является релевантной для торцового фрезерования, так как здесь и раньше допускались только определенные типы инструмента без релевантного положения резцов. (Инструменты с не явно допущенным типом инструмента рассматриваются как сферическая фреза с указанным радиусом. Указание положения резцов игнорируется.)

3.11 Параметры движения по траектории

3.11.1 Точный останов (G60, G9, G601, G602, G603)

Точный останов это режим перемещения, при котором в конце каждого кадра перемещения все участвующие в движении перемещения траекторные оси и дополнительные оси, которые не перемещаются за границы кадров, затормаживаются до состояния покоя.

Точный останов используются тогда, когда необходимо изготовление острых наружных углов или чистовая обработка внутренних углов по размеру.

С помощью критерия точного останова определяется, как точно будет выполнен подвод к угловой точке и когда будет выполнено переключение на следующий кадр:

- "Точный останов точный"
  Смена кадра выполняется, как только для всех участвующих в движении перемещения осей достигнуты спец. для осей границы допуска для "Точного останова точного".
  "точный останов точный" настраивается через: MD36010 $MA_STOP_LIMIT_FINE[<ось>]
- "Точный останов грубый"
  Смена кадра выполняется, как только для всех участвующих в движении перемещения осей достигнуты спец. для осей границы допуска для "Точного останова грубого".
  "точный останов грубый" настраивается через: MD36000 $MA_STOP_LIMIT_COARSE[<ось>]
- "Завершение интерполятора"
  Смена кадра выполняется, как только СЧПУ для всех участвующих в движении перемещения осей вычислила заданную скорость ноль. Фактическая позиция или отклонение, обусловленное запаздыванием, участвующих осей не рассматриваются.

Синтаксис

```
G60 ...
G9 ...
G601/G602/G603 ...
```

Объяснение

G60:	Команда для включения действующего **модально** точного останова
G9:	Команда для включения действующего **покадрово** точного останова
G601:	Команда для активации критерия точного останова "**Точный останов точный**"
G602:	Команда для активации критерия точного останова "**Точный останов грубый**"
G603:	Команда для активации критерия точного останова "**Завершение интерполятора**"

Примечание

Команды для активации критериев точного останова (G601 / G602 / G603) действуют только при активной G60 или G9.

Пример

Программный код	Комментарий
N5 G602	; Выбран критерий "Точный останов грубый".
N10 G0 G60 Z...	; Модальный точный останов активен.
N20 X... Z...	; G60 продолжает действовать.
...	
N50 G1 G601	; Выбран критерий "Точный останов точный".

Программный код	Комментарий
`N80 G64 Z...`	; Переключение на режим управления траекторией.
...	
`N100 G0 G9`	; Точный останов действует только в этом кадре.
`N110 ...`	; Режим управления траекторией снова активен.

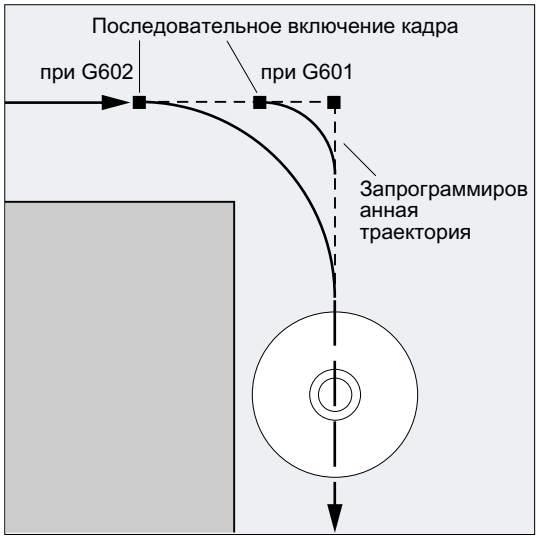
Дополнительная информация

G60, G9

G9 создает точный останов в текущем кадре, G60 – в актуальном кадре и во всех последующих кадрах.

Команды режима управления траекторией G64 или G641 - G645 отключают G60.

G601, G602



Движение притормаживается и кратковременно останавливается на угловой точке.

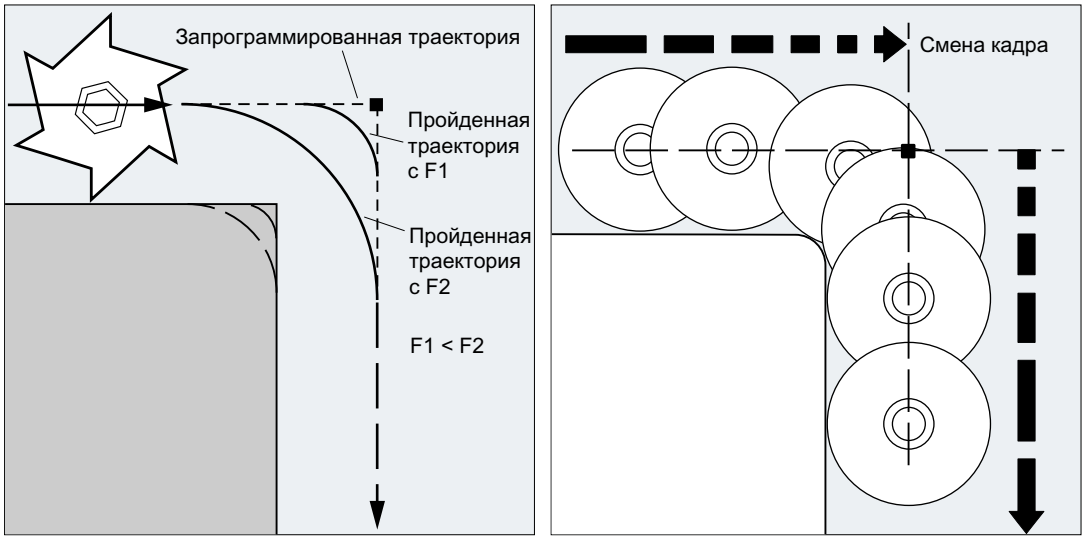
Примечание

Устанавливать только необходимую мин. ширину критериев точного останова. Чем ближе друг другу зафиксированы границы, тем дольше длится компенсация положения и переход к заданному конечному положению.

G603

Смена кадра запускается при вычислении СЧПУ заданной скорости для участвующих осей равной нулю. На этот момент времени фактическое значение – в зависимости от

динамики и скорости движения по траектории – отстает на участок выбега. Благодаря этому возможна шлифовка углов детали.



Сконфигурированный критерий точного останова

Для G0 и прочих команд 1-й группы G спец. для канала можно зафиксировать, что вместо запрограммированного критерия точного останова автоматически используется предустановленный критерий (см. данные изготовителя станка!).

3.11.2 Режим управления траекторией (G64, G641, G642, G643, G644, G645, ADIS, ADISPOS)

В режиме управления траекторией скорость движения по траектории на конце кадра для его смены не снижается до скорости, обеспечивающей достижение критерия точного останова. Целью же, напротив, является избежание сильного торможения траекторных осей в точке смены кадра, чтобы перейти в следующий кадр по возможности с той же скоростью движения по траектории. Для достижения этой цели, при выборе режима управления траекторией дополнительно активируется функция "Опережающее управление скоростью (LookAhead)".

Режим управления траекторией с перешлифовкой означает, что ломанные переходы кадров через локальные изменения запрограммированной характеристики делаются тангенциальными или сглаживаются.

Следствием режима управления траекторией являются:

- Закругление контура
- Сокращение времени обработки из-за отсутствия процессов торможения и разгона, необходимых для достижения критерия точного останова.
- Улучшение условий резания благодаря более равномерной эпюре скоростей.

Режим управления траекторией имеет смысл, если:

- Контур должен быть пройден по возможности быстро (к примеру, с ускоренным ходом).
- Точная характеристика в рамках критерия ошибки может отличаться от запрограммированной, чтобы создать непрерывную постоянную характеристику.

Режим управления траекторией не имеет смысла, если:

- Контур должен быть пройден точно.
- Требуется абсолютная стабильность скорости.

Примечание

Режим управления траекторией прерывается кадрами, не явно запускающими остановку предварительной обработки, к примеру, через:

- Обращение к определенным данным состояния станка ($A...)
- Вывод вспомогательных функций

Синтаксис

```
G64 ...
G641 ADIS=...
G641 ADISPOS=…
G642 ...
G643 ...
G644 ...
G645 ...
```

Значение

G64:	Режим управления траекторией со снижением скорости согласно коэффициенту перегрузки
G641:	Режим управления траекторией с перешлифовкой по критерию пути
ADIS=...:	Критерий пути при G641 для траекторных функций G1, G2, G3, ...
ADISPOS=...:	Критерий пути при G641 для ускоренного хода G0
	Критерий пути (= интервал перешлифовки) ADIS или ADISPOS описывает участок, который кадр перешлифовки может начать самое раннее перед концом кадра или участок после конца кадра, на котором должен быть завершен кадр перешлифовки. **Примечание:** Если ADIS/ADISPOS не программируется, то действует значение "ноль" и параметры движения как при G64. При коротких путях перемещения интервал перешлифовки уменьшается автоматически (до макс. 36 %).

G642:	Режим управления траекторией с перешлифовкой с соблюдением определенных допусков В этом режиме перешлифовка обычно выполняется с соблюдением макс. допустимой погрешности траектории. Но вместо этого спец. для оси допуска может быть сконфигурировано и соблюдение макс. погрешности контура (допуск контура) или макс. углового отклонения ориентации инструмента (допуск ориентации. **Примечание:** Расширение на допуск контура и ориентации существует только в системах с имеющейся опцией "Полиномиальная интерполяция".
G643:	Режим управления траекторией с перешлифовкой с соблюдением определенных допусков (внутри кадра) С помощью G643, в отличие от G642, не создается отдельный кадр перешлифовки, а вставляются специфические для осей внутрикадровые движения перешлифовки. Путь перешлифовки может быть различным для каждой оси.
G644:	Режим управления траекторией с перешлифовкой с макс. возможной динамикой **Примечание:** G644 при активной кинематической трансформации невозможна. Происходит внутреннее переключение на G642.
G645:	Режим управления траекторией с перешлифовкой углов с тангенциальными переходами кадров с соблюдением определенных допусков G645 работает на углах идентично G642. Но с G645 кадры перешлифовки создаются и на тангенциальных переходах кадров только тогда, когда ход изгиба оригинального контура минимум в одной оси имеет скачок.

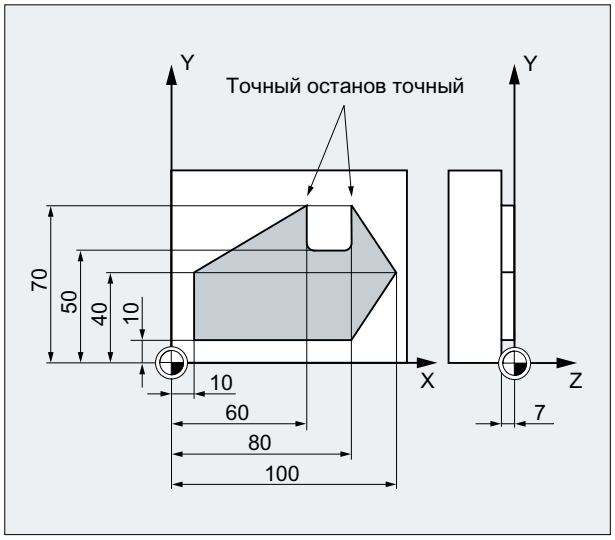
Примечание

Перешлифовка не является заменой для закругления углов (RND). Пользователь не должен предполагать, как будет выглядеть контур внутри зоны перешлифовки. Тип перешлифовки может зависеть и от динамических свойств, к примеру, скорости движения по траектории. Поэтому перешлифовка на контуре имеет смысл только с маленькими значениями ADIS. Если необходимо прохождение определенного контура на углу, то надо использовать RND.

Примечание

Если созданное через G641, G642, G643, G644 или G645 движение перешлифовки прерывается, то при последующем репозиционировании (REPOS) выполняется подвод не к точке прерывания, а к начальной или угловой точке оригинального кадра перемещения (в зависимости от режима REPOS).

Пример



Требуется точный подвод к обоим наружным углам на пазу. Остальной процесс изготовления должен происходить в режиме управления траекторией.

Программный код	Комментарий
`N05 DIAMOF`	; Радиус как указание размера.
`N10 G17 T1 G41 G0 X10 Y10 Z2 S300 M3`	; Переход к стартовой позиции, включение шпинделя, коррекция траектории.
`N20 G1 Z-7 F8000`	; Подача инструмента.
`N30 G641 ADIS=0.5`	; Зашлифовка переходов контура.
`N40 Y40`	
`N50 X60 Y70 G60 G601`	; Точный подвод к позиции с точным остановом точным.
`N60 Y50`	
`N70 X80`	
`N80 Y70`	
`N90 G641 ADIS=0.5 X100 Y40`	; Зашлифовка переходов контура.
`N100 X80 Y10`	
`N110 X10`	
`N120 G40 G0 X-20`	; Выключение коррекции траектории.
`N130 Z10 M30`	; Отвод инструмента, конец программы.

Дополнительная информация

Режим управления траекторией G64

В режиме управления траекторией инструмент проходит тангенциальные контурные переходы с возможной постоянной скоростью движения по траектории (без притормаживания на границах кадра). Перед углами и кадрами с точным остановом выполняется опережающее торможение (LookAhead).

Обход углов осуществляется также с постоянной скоростью. Для уменьшения ошибок контура скорость соответственно снижается с учетом предела ускорения и коэффициента перегрузки.

Примечание

Степень шлифовки переходов контура зависит от скорости подачи и коэффициента перегрузки. Коэффициент перегрузки может быть настроен в MD32310 $MA_MAX_ACCEL_OVL_FACTOR.

Через установку MD20490 $MC_IGNORE_OVL_FACTOR_FOR_ADIS перешлифовка переходов кадров всегда выполняется независимо от установленного коэффициента перегрузки.

Во избежание нежелательной остановки движения по траектории (свободное резание!) следует учитывать следующие моменты:

- Вспомогательные функции, включаемые после окончания движения или перед следующим движением, прерывают режим управления траекторией (исключение: быстрые вспомогательные функции).
- Позиционирующие оси всегда перемещаются по принципу точного останова, окно позиционирования точное (как G601). Если в кадре ЧПУ необходимо ждать позиционирующие оси, то режим управления траекторией траекторных осей прерывается.

Промежуточно запрограммированные кадры только с комментариями, кадрами вычисления или вызовами подпрограмм, напротив, не приводят к помехам.

Примечание

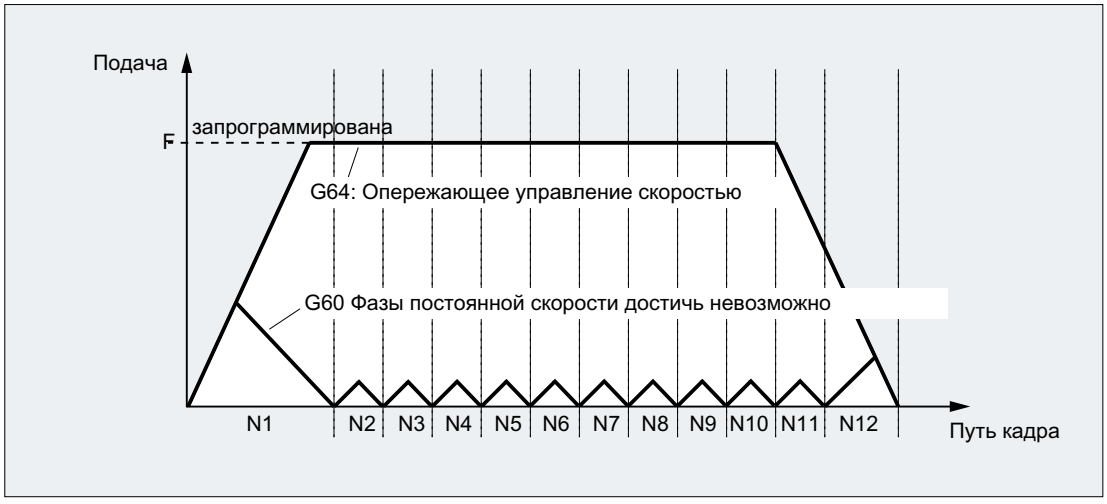
Если не все траекторные оси включены в FGROUP, то часто на переходах кадра для не включенных осей происходит скачок скорости, который СЧПУ ограничивает посредством уменьшения скорости на смене кадров до разрешенного через MD32300 $MA_MAX_AX_ACCEL и MD32310 $MA_MAX_ACCEL_OVL_FACTOR значения. Этого притормаживания можно избежать, смягчив заданную связь позиций траекторных осей через перешлифовку.

Опережающее управление скоростью LookAhead

В режиме управления траекторией с СЧПУ заранее автоматически определяет управление скоростью для нескольких кадров ЧПУ. Благодаря этому ускорение и торможение для аппроксимирующих тангенциальных переходов может осуществляться через несколько кадров.

Прежде всего благодаря опережающему управлению скоростью с высокими траекторными подачами можно создавать цепочки движений, которые состоят из коротких сегментов перемещения.

Максимальное количество кадров ЧПУ, на которое может осуществляться опережение, может быть настроено через машинные данные.



Режим управления траекторией с перешлифовкой по критерию пути (G641)

При G641 СЧПУ вставляет переходные элементы на переходах контура. С помощью интервала перешлифовки ADIS (или ADISPOS при G0) указывается макс. разрешенная зашлифовка углов. В пределах интервала перешлифовки СЧПУ может разрывать траекторную связь и заменять ее на динамически-оптимальный путь.

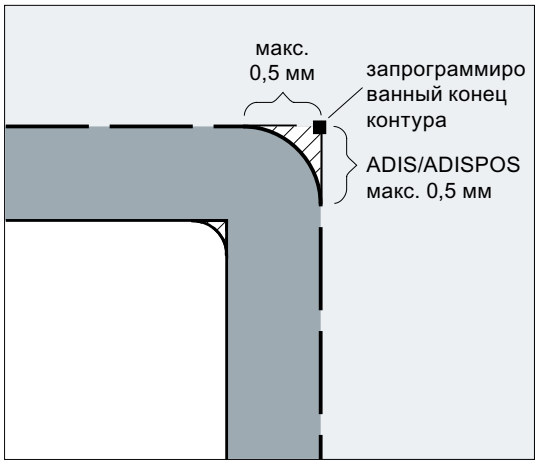
Недостаток: Для всех осей доступно только одно значение ADIS.

G641 действует подобно RNDM, но не ограничена осями рабочей плоскости.

Как и G64, G641 работает с опережающим управлением скоростью LookAhead. Подвод к кадрам перешлифовки с сильным изгибом осуществляется с уменьшенной скоростью.

Пример:

Программный код	Комментарий
`N10 G641 ADIS=0.5 G1 X... Y...`	; Кадр перешлифовки может начинаться самое меньшее за 0,5 мм перед запрограммированным концом кадра и должен заканчиваться через 0,5 мм после конца кадра. Эта настройка действует модально.



Примечание

Перешлифовка не может и не должна подменять функции для определенного сглаживания (RND, RNDM, ASPLINE, BSPLINE, CSPLINE).

Перешлифовка с осевой точностью при G642

При G642 перешлифовка осуществляется не в пределах определенной области ADIS, а соблюдаются определенные с MD33100 $MA_COMPRESS_POS_TOL осевые допуски. Путь перешлифовки определяется из кратчайшего пути перешлифовки всех осей. Это значение учитывается при создании кадра перешлифовки.

Внутрикадровая перешлифовка при G643

Максимальные отклонения от точного контура устанавливаются при перешлифовке с G643 через машинные данные MD33100 $MA_COMPRESS_POS_TOL для каждой оси.

С помощью G643 не создается свой кадр перешлифовки, а вставляются специфические для осей внутрикадровые движения перешлифовки. При G643 путь перешлифовки каждой оси может быть различным.

Перешлифовка с допуском контура и ориентации при G642/G643

С помощью MD20480 $MC_SMOOTHING_MODE перешлифовка с G642 и G643 может быть сконфигурирована таким образом, что вместо специфических для осей допусков будут действовать допуск контура и допуск ориентации.

Допуск контура и ориентации задаются в спец. для канала установочных данных:

SD42465 $SC_SMOOTH_CONTUR_TOL (макс. погрешность контура)

SD42466 $SC_SMOOTH_ORI_TOL (макс. угловое отклонение ориентации инструмента)

Установочные данные могут программироваться и программе ЧПУ и тем самым задаваться разными для каждого перехода кадра. Очень разные параметры для допуска контура и допуска ориентации инструмента могут сказываться только при G643.

Примечание

Расширение на допуск контура и ориентации существует только в системах с имеющейся опцией "Полиномиальная интерполяция".

Примечание

Для перешлифовки с соблюдением допуска ориентации, должна быть активна трансформация ориентации.

Перешлифовка с макс. возможной динамикой при G644

Перешлифовка с макс. возможной динамикой конфигурируется в MD20480 $MC_SMOOTHING_MODE на четвертой позиции:

Значение	Объяснение
0	Задача макс. осевых погрешностей с: MD33100 $MA_COMPRESS_POS_TOL
1	Задача макс. пути перешлифовки через программирование: ADIS=... или ADISPOS=...
2	Задача макс. возможных частот каждой оси в области перешлифовки с: MD32440 $MA_LOOKAH_FREQUENCY Область перешлифовки устанавливается таким образом, чтобы при движении перешлифовки не возникали частоты, превышающие заданную макс. частоту.
3	При перешлифовке с G644 не контролируется ни допуск, ни интервал перешлифовки. Каждая ось движется с макс. возможной динамикой вокруг угла. При SOFT соблюдается и макс. ускорение, и макс. рывок каждой оси. При BRISK рывок не ограничивается, а каждая ось движется с макс. возможным ускорением.

Перешлифовка тангенциальных переходов кадров при G645

Движение перешлифовки при G645 определяется так, что все участвующие оси не получают скачка в ускорении, а спараметрированные макс. отклонения от оригинального контура (MD33120 $MA_PATH_TRANS_POS_TOL) не превышаются.

На ломанных, не тангенциальных переходах кадров характеристика перешлифовки такая же, как при G642.

Без промежуточных кадров перешлифовки

В следующих случаях промежуточный кадр перешлифовки не вставляется:

- Между обеими кадрами осуществляется остановка.
  Это происходит, если:
  - Вывод вспомогательной функции стоит перед движением в последующем кадре.
  - Последующий кадр не содержит движения по траектории.
  - Для последующего кадра в первый раз ось, которая до этого была позиционирующей осью, перемещается как траекторная ось.
  - Для последующего кадра в первый раз ось, которая до этого была траекторной осью, перемещается как позиционирующая ось.
  - Предшествующий кадр перемещает гео-оси, а последующий кадр нет.
  - Последующий кадр перемещает гео-оси, а предыдущий кадр нет.
  - Перед резьбонарезанием последующий кадр имеет G33 как функцию перемещения, а предшествующий кадр нет.
  - Осуществляется переключение между BRISK и SOFT.
  - Значимые для трансформации оси не полностью согласованы с движением по траектории (например, при качании, позиционирующие оси).
- Кадр перешлифовки замедлил бы выполнение программы обработки деталей.
  Это происходит:
  - Между очень кроткими кадрами.
    Так как для каждого кадра необходимо минимум один такт интерполятора, то вставленный промежуточный кадр удвоил бы время обработки.
  - Если переход кадра с G64 (режим управления траекторией без перешлифовки) может быть пройден без уменьшения скорости.
    Перешлифовка увеличила бы время обработки. Т. е. значение разрешенного коэффициента перегрузки (MD32310 $MA_MAX_ACCEL_OVL_FACTOR) влияет на то, будет ли выполнена перешлифовка перехода кадра или нет. Коэффициент перегрузки учитывается только при перешлифовке с G641 / G642. При перешлифовке с G643 коэффициент перегрузки не влияет (такое поведение может быть установлено и для G641 и G642 , для этого установить MD20490 $MC_IGNORE_OVL_FACTOR_FOR_ADIS = TRUE).

- Перешлифовка не спараметрирована.
  Это происходит, если:
  - При G641 в кадрах G0 ADISPOS=0 (предустановка!).
  - При G641 в кадрах не-G0 ADISPOS=0 (предустановка!).
  - При G641 при переходе между G0 и не-G0 или не-G0 и G0 действует меньшее значение из ADISPOS и ADIS.
  - При G642/G643 все специфические для осей допуски равны нулю.
- Кадр не содержит движения перемещения (нулевой кадр).
  Это происходит, если:
  - Активны синхронные действия.
    Обычно нулевые кадры удаляются интерпретатором. Но если синхронные действия активны, то этот нулевой кадр вставляется в цепочку и выполняется. При этом запускается точный останов согласно активному программированию. Тем самым синхронное действие должно получить возможность для переключения.
  - Через переходы в программе создаются нулевые кадры.

Режим управления траекторией при ускоренном ходе G0

И для движения ускоренным ходом должна быть указана одна из названных функций G60/G9 или G64 или G641 - G645. В иных случаях действует введенная через машинные данные предустановка.

3.12 Трансформации координат (фреймы)

3.12.1 Фреймы

Фрейм

Фрейм – это автономное правило вычисления, которое переводит одну декартову систему координат в другую декартову систему координат.

Базовый фрейм (базовое смещение)

Базовый фрейм описывает трансформацию координат из базовой кинематической системы (BKS) в базовую систему нулевой точки (BNS) и действует как устанавливаемые фреймы.

См. Базовая кинематическая система (BKS) (Страница 42) .

Устанавливаемые фреймы

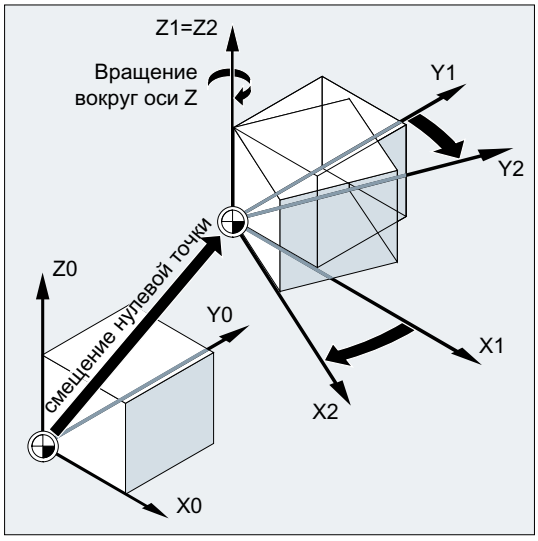
Устанавливаемые фреймы — это вызываемые с помощью команд G54 до G57 и G505 до G599 из любой программы ЧПУ устанавливаемые смещения нулевой точки. Значения смещения предварительно устанавливаются оператором и сохраняются в памяти нулевой точки СЧПУ. С их помощью определяется настраиваемая система нулевой точки (ENS).

См.:

- Настраиваемая система нулевой точки (ENS) (Страница 43)
- Устанавливаемое смещение нулевой точки (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153) (Страница 150)

Программируемые фреймы

Иногда имеет смысл или требуется переместить в программе ЧПУ первоначально выбранную систему координат детали (или "Настраиваемую систему нулевой точки") на другую позицию и при необходимости повернуть, отразить и/или масштабировать. Это осуществляется через программируемые фреймы.



См. Базовая кинематическая система (BKS) (Страница 42) .

См. также

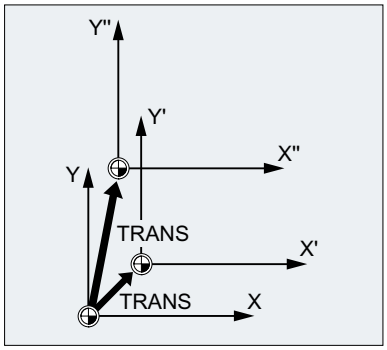
Фрейм-операторы (Страница 316)

3.12.2 Фрейм-операторы

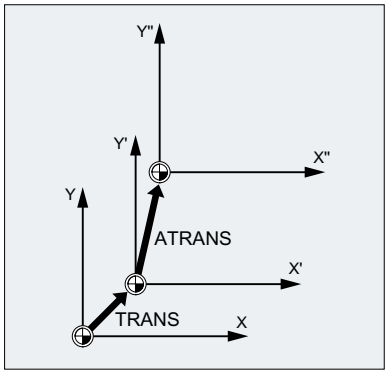
Функция

Операторы для программируемых фреймов действуют в актуальной программе ЧПУ. Они действуют либо аддитивно, либо как замещение:

- Замещающий оператор
  Удаляет все запрограммированные прежде фрейм-операторы. Исходной точкой служит последнее настраиваемое смещение нулевой точки (G54 ... G57, G505 ... G599).

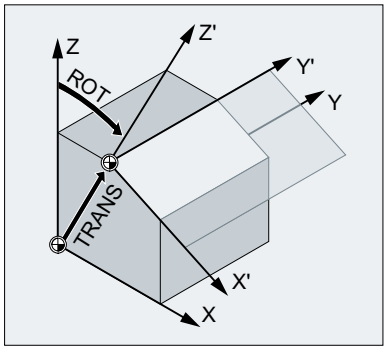


- Аддитивный оператор
  Устанавливается на уже существующий фрейм. Исходной точкой служит актуальная установленная или последняя запрограммированная через фрейм-оператор нулевая точка детали.



Пример использования

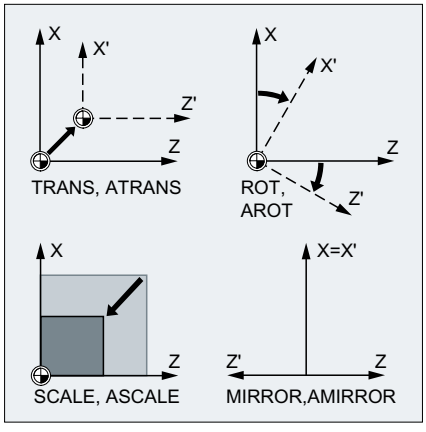
1. Смещение нулевой точки системы координат детали (WCS).
2. Поворот системы координат детали (WCS) для выравнивания плоскости параллельно нужной рабочей плоскости.

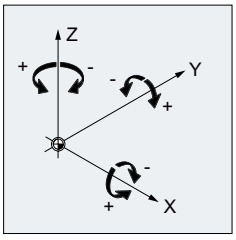


Синтаксис

Замещающие операторы	Аддитивные операторы
TRANS X… Y… Z…	ATRANS X… Y… Z…
ROT X… Y… Z…	AROT X… Y… Z…
ROT RPL=…	AROT RPL=…
ROTS/CROTS X... Y...	AROTS X... Y...
SCALE X… Y… Z…	ASCALE X… Y… Z…
MIRROR X0/Y0/Z0	AMIRROR X0/Y0/Z0

Объяснение



TRANS/ATRANS:	Смещение WCS в направлении указанной гео-оси(ей)		
ROT/AROT:	Вращение WCS: • через соединение отдельных вращений вокруг указанной гео-оси(ей) или • на угол RPL=... в актуальной рабочей плоскости (G17/G18/G19)		
	Направление вращения:		
	Последовательность вращений:	с представлением RPY:	Z, Y', X''
		с углом Эйлера:	Z, X', Z''
	Диапазон значений:	Углы поворота определены однозначно только в следующих диапазонах:	
		с представлением RPY:	-180 ≤ x ≤ 180
			-90 < y < 90
			-180 ≤ z ≤ 180
		с углом Эйлера:	0 ≤ x < 180
			-180 ≤ y ≤ 180
			-180 ≤ z ≤ 180
ROTS/AROTS:	Вращение WCS через указание пространственных углов Ориентация плоскости в пространстве однозначно определена через указание двух пространственных углов. Поэтому может быть запрограммировано макс. 2 пространственных угла: ROTS/AROTS X... Y... / Z... X... / Y... Z...		
CROTS:	CROTS действует как ROTS, но относится к действующему фрейму в системе УД.		
SCALE/ASCALE:	Масштабирование в направлении указанной гео-оси(ей) для увеличения/уменьшения контура		
MIRROR/AMIRROR:	Отражение WCS через отражение (перемена направления) указанной гео-оси		
	Значение:	выбирается свободно (здесь: "0")	

Граничные условия

- Фрейм-операторы нужно программировать в отдельном кадре ЧПУ.
- Фрейм-операторы могут использоваться по отдельности или произвольно комбинироваться друг с другом.

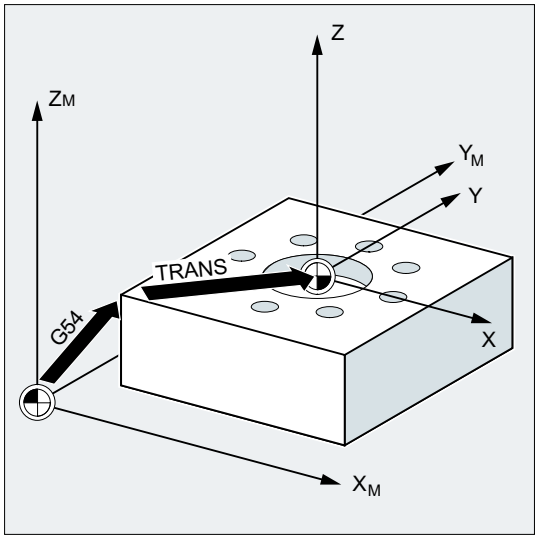
- Фрейм-операторы выполняются в запрограммированной последовательности.
- Аддитивные операторы часто используются в подпрограммах. Определенные в главной программе базовые операторы сохраняются после завершения подпрограммы, если подпрограмма была запрограммирована с атрибутом SAVE.

3.12.3 программируемое смещение нулевой точки (TRANS, ATRANS)

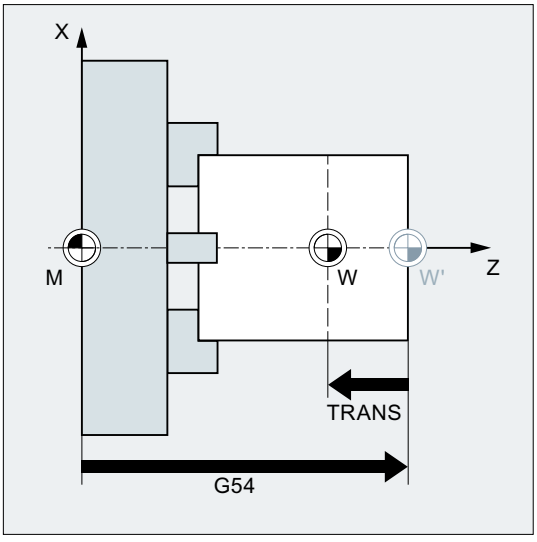
С помощью команды `TRANS` выполняется абсолютное смещение WCS от системы ENS, созданной через настраиваемое смещение нулевой точки (G54 ... G57, G505 ... G599).

С помощью команды `ATRANS` выполняется аддитивное смещение WCS, созданной через `TRANS`.

Фрезерование:



Токарная обработка:



Синтаксис

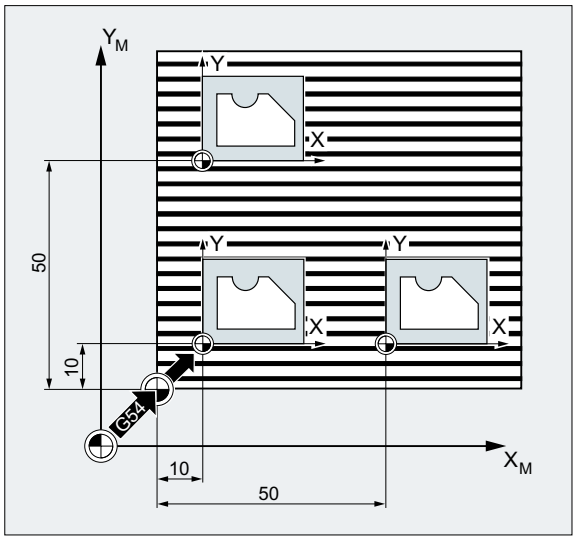
```
TRANS X... Y... Z...
ATRANS X... Y... Z...
```

Объяснение

`TRANS`:	Абсолютное смещение WCS от нулевой точки детали (ENS), установленной через настраиваемое смещение нулевой точки (G54 ... G57, G505 ... G599)	
	Автономно в кадре:	да
`ATRANS`:	Аддитивное смещение нулевой точки WCS от нулевой точки детали, установленной через `TRANS`	
	Автономно в кадре:	да
`X... Y... Z...`:	Значения смещения в направлении указанных гео-осей	

Примеры

Пример 1: Фрезерная обработка



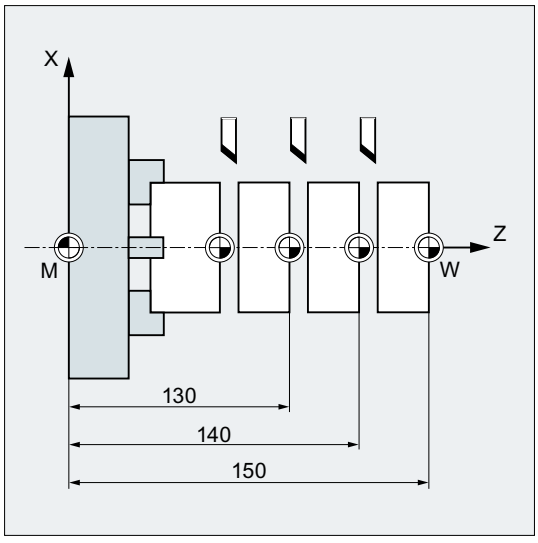
У этой детали показанные формы встречаются несколько раз в одной программе.

Последовательность обработки для этой формы зафиксирована в подпрограмме.

Через смещение нулевой точки устанавливаются соответствующие требуемые нулевые точки детали и после вызывается подпрограмма.

Программный код	Комментарий
`N10 G1 G54`	; Рабочая плоскость `X/Y`, нулевая точка детали
`N20 G0 X0 Y0 Z2`	; движение к стартовой точке
`N30 TRANS X10 Y10`	; Абсолютное смещение
`N40 L10`	; Вызов подпрограммы
`N50 TRANS X50 Y10`	; Абсолютное смещение
`N60 L10`	; Вызов подпрограммы
`N70 M30`	; конец программы

Пример 2: Токарная обработка



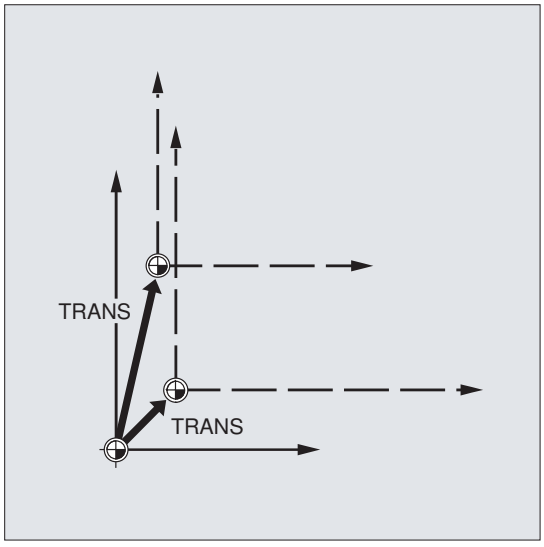
Программный код	Комментарий
...	
`N10 TRANS X0 Z150`	; Абсолютное смещение
`N15 L20`	; Вызов подпрограммы
`N20 TRANS X0 Z140 (или ATRANS Z-10)`	; Абсолютное смещение
`N25 L20`	; Вызов подпрограммы
`N30 TRANS X0 Z130 (или ATRANS Z-10)`	; Абсолютное смещение
`N35 L20`	; Вызов подпрограммы
...	

Дополнительная информация

TRANS X... Y... Z...

Смещение нулевой точки на запрограммированные в соответствующих указанных осевых направлениях (траекторные, синхронные и позиционирующие оси) значения смещения. Исходной точкой служит последнее указанное устанавливаемое смещение нулевой точки (G54 ... G57, G505 ... G599).

ВНИМАНИЕ
Не исходный фрейм Команда `TRANS` сбрасывает все фрейм-компоненты установленного до этого программируемого фрейма.

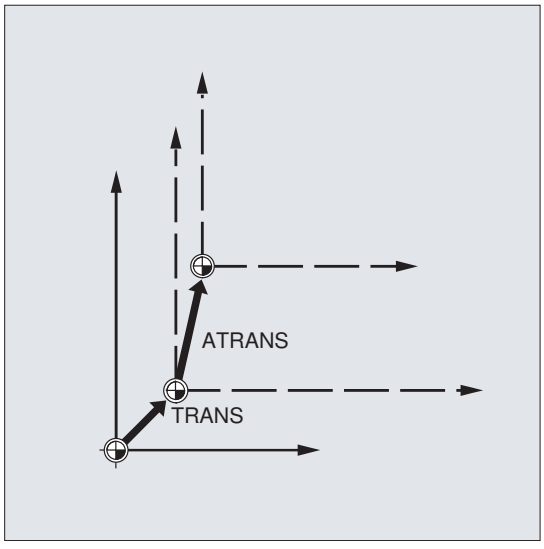


Примечание

Смещение, надстраиваемое на уже существующие фреймы, должно быть запрограммировано с `ATRANS`.

ATRANS X... Y... Z...

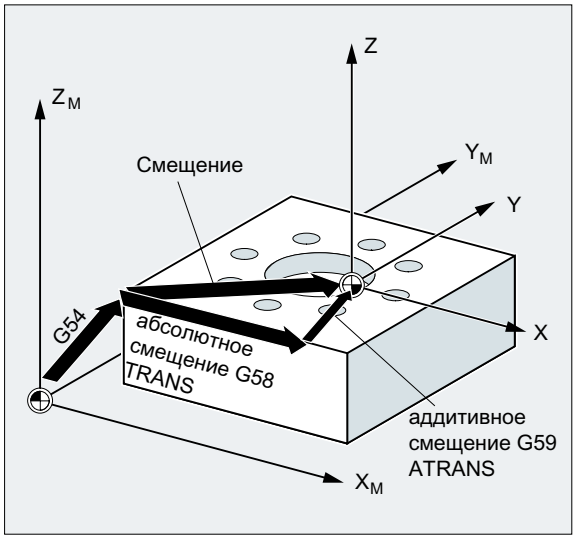
Смещение нулевой точки на запрограммированные в соответствующих указанных осевых направлениях значения смещения. Исходной точкой служит актуальная установленная или последняя запрограммированная нулевая точка.



3.12.4 Программируемое смещение нулевой точки (G58, G59)

С помощью функций G58 и G59 может быть осуществлено осевое замещение долей программируемого смещения нулевой точки (TRANS/ATRANS) (Страница 319):

- G58: абсолютная доля смещения (грубое смещение)
- G59: аддитивная доля смещения (точное смещение)



Условия

Функции G58 и G59 могут использоваться только при сконфигурированном точном смещении (MD24000 $MC_FRAME_ADD_COMPONENTS = 1).

Синтаксис

```
G58 <ось_1><значение_1> ... <ось_3><значение_3>
G59 <ось_1><значение_1> ... <ось_3><значение_3>
```

Объяснение

G58:	G58 заменяет абсолютную долю смещения программируемого смещения нулевой точки для указанной оси, аддитивно запрограммированное смещение сохраняется. Исходной точкой служит последнее настраиваемое смещение нулевой точки (G54 ... G57, G505 ... G599).
	Автономно в кадре: да
G59:	G59 заменяет аддитивную долю смещения программируемого смещения нулевой точки для указанной оси, абсолютно запрограммированное смещение сохраняется.
	Автономно в кадре: да
<ось_n>:	Геометрическая ось в канале
<значение_n>:	Значения смещения в направлении указанных геометрических осей

Пример

Программный код	Комментарий
...	
`N50 TRANS X10 Y10 Z10`	; Абсолютная доля смещения `X10 Y10 Z10`
`N60 ATRANS X5 Y5`	; Аддитивная доля смещения `X5 Y5`
	→ Общее смещение: `X15 Y15 Z10`
`N70 G58 X20`	; абсолютная доля смещения `X20`
	→ Общее смещение `X25 Y15 Z10`
`N80 G59 X10 Y10`	; Аддитивная доля смещения `X10 Y10`
	→ Общее смещение `X30 Y20 Z10`
...	

Дополнительная информация

Абсолютная доля смещения (**грубое смещение**) изменяется через следующие команды:

- `TRANS`
- `G58`
- `CTRANS`
- `CFINE`
- `$P_PFRAME[x,TR]`

Аддитивная доля смещения (**точное смещение**) изменяется через следующие команды:

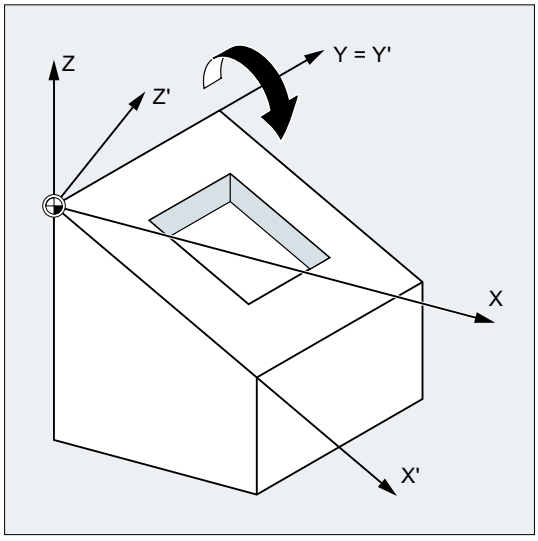
- `ATRANS`
- `G59`
- `CTRANS`
- `CFINE`
- `$P_PFRAME[X,FI]`

Примеры

Команда	Грубое смещение V_C	Точное смещение V_F
`TRANS X10`	$V_C = 10$	Без изменений
`G58 X10`	$V_C = 10$	Без изменений
`$P_PFRAME[X,TR] = 10`	$V_C = 10$	Без изменений
`ATRANS X10`	Без изменений	$V_F = V_F + 10$
`G59 X10`	Без изменений	$V_F = 10$
`$P_PFRAME[X,FI] = 10`	Без изменений	$V_F = 10$
`CTRANS(X,10)`	$V_C = 10$	$V_F = 0$
`CTRANS()`	$V_C = 0$	$V_F = 0$
`CFINE(X,10)`	$V_C = 0$	$V_F = 10$

3.12.5 Программируемое вращение (ROT, AROT, RPL)

С помощью команд ROT / AROT возможен поворот системы координат детали в пространстве. Команды относятся только к программируемому фрейму $P_PFRAME.



Синтаксис

```
ROT <1-я гео-ось><угол> <2-я гео-ось><угол> <3-я гео-ось><угол>
ROT RPL=<угол>
AROT <1-я гео-ось><угол> <2-я гео-ось><угол> <3-я гео-ось><угол>
AROT RPL=<угол>
```

Примечание

Вращения системы координат детали выполняются через угол Эйлера.

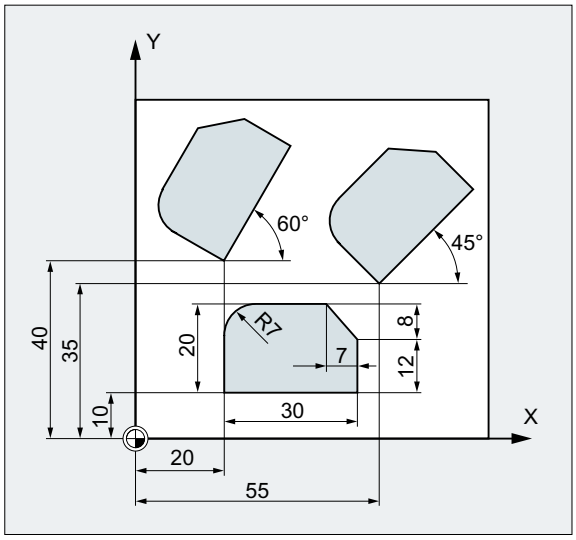
Объяснение

ROT:	Абсолютное вращение	
	Исходный фрейм:	Программируемый фрейм $P_PFRAME
	Исходная точка:	Нулевая точка актуальной установленной с G54 ... G57, G505 ... G599 системы координат детали
AROT:	Аддитивное вращение	
	Исходный фрейм:	Программируемый фрейм $P_PFRAME
	Исходная точка:	Нулевая точка актуальной установленной с G54 ... G57, G505 ... G599 системы координат детали
<n-ная гео-ось>:	Идентификатор n-ной гео-оси, вокруг которой должен быть выполнен поворот на указанный угол. Для не запрограммированной гео-оси в качестве угла поворота не явно устанавливается значение 0°.	

RPL:	Поворот вокруг расположенной вертикально к активной плоскости (G17, G18, G19) гео-оси на указанный угол	
	Исходный фрейм:	Программируемый фрейм $P_PFRAME
	Исходная точка:	Нулевая точка актуальной установленной с G54 ... G57, G505 ... G599 системы координат детали
<угол>	Указание угла в градусах	
	Диапазон значений:	-360° ≤ угол ≤ 360°

Примеры

Пример 1: Вращение в плоскости G17

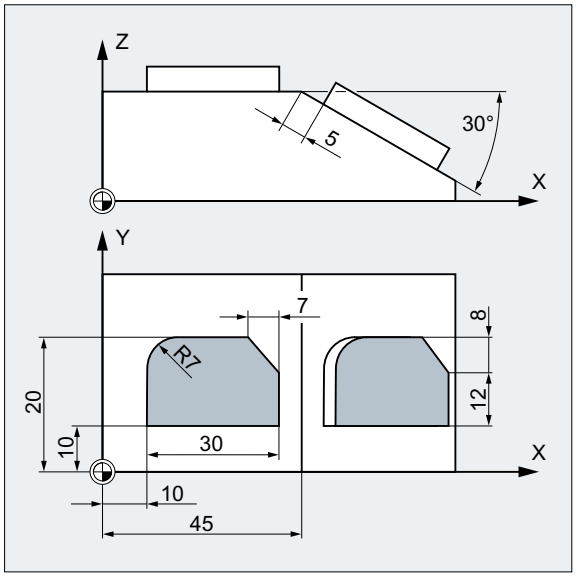


У этой детали показанные формы встречаются несколько раз в одной программе. Дополнительно к смещению нулевой точки должны быть осуществлены вращения, так как формы расположены не параллельно осям.

Программный код	Комментарий
N10 G17 G54	; Рабочая плоскость X/Y, нулевая точка детали
N20 TRANS X20 Y10	; Абсолютное смещение
N30 L10	; Вызов подпрограммы
N40 TRANS X55 Y35	; Абсолютное смещение
N50 AROT RPL=45	; Аддитивный поворот вокруг расположенной вертикально к плоскости G17
	; оси Z на 45°
N60 L10	; Вызов подпрограммы
N70 TRANS X20 Y40	; Абсолютное смещение ; (сбрасывает все предыдущие смещения)
N80 AROT RPL=60	; Аддитивный поворот вокруг расположенной вертикально к плоскости G17
	; оси Z на 60°
N90 L10	; Вызов подпрограммы
N100 G0 X100 Y100	; Отвод

Программный код	Комментарий
`N110 M30`	; конец программы

Пример 2: Вращение в пространстве вокруг оси Y

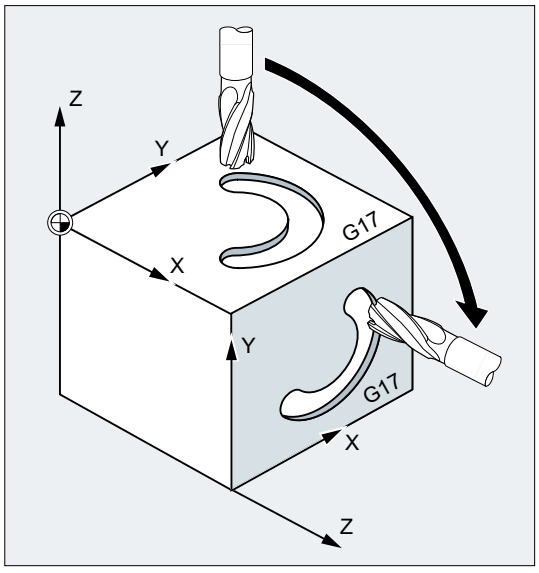


В этом примере в одном зажиме должны быть обработаны параллельные осям и наклонные поверхности детали.

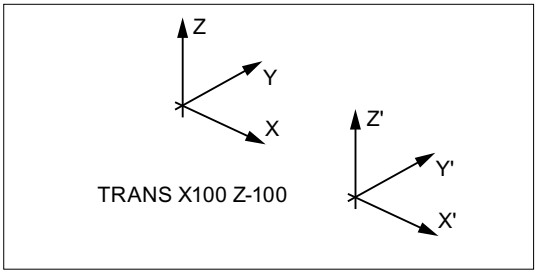
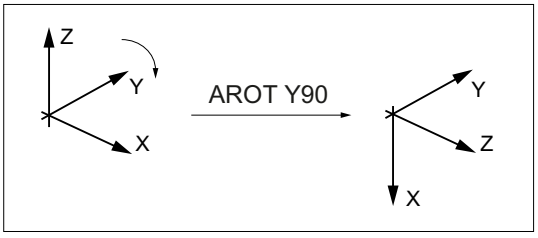
Условие:
Инструмент должен быть точно установлен вертикально к наклонной плоскости в повернутом направлении Z.

Программный код	Комментарий
`N10 G17 G54`	; Рабочая плоскость `X/Y`, нулевая точка детали
`N20 TRANS X10 Y10`	; Абсолютное смещение
`N30 L10`	; Вызов подпрограммы
`N40 ATRANS X35`	; Аддитивное смещение
`N50 AROT Y30`	; Аддитивное вращение вокруг оси `Y`
`N60 ATRANS X5`	; Аддитивное смещение
`N70 L10`	; Вызов подпрограммы
`N80 G0 X300 Y100 M30`	; отвод, конец программы

Пример 3: Многосторонняя обработка



В этом примере в двух расположенных вертикально друг к другу поверхностях детали через подпрограммы изготовляются идентичные формы. В новой системе координат на правой поверхности детали направление подачи, рабочая плоскость и нулевая точка установлены как в верхней поверхности. При этом дальше действуют необходимые для выполнения подпрограммы условия: рабочая плоскость `G17`, плоскость координат X/Y, направление подачи Z.

Программный код	Комментарий
`N10 G17 G54`	; Рабочая плоскость `X/Y`, нулевая точка детали
`N20 L10`	; Вызов подпрограммы
`N30 TRANS X100 Z-100`	; Абсолютное смещение `WCS`
	
`N40 AROT Y90`	; Аддитивный поворот `WCS` вокруг `Y` на 90°
	

Программный код	Комментарий
`N50 AROT Z90`	; Аддитивный поворот `WCS` вокруг `Z` на 90°
`N60 L10`	; Вызов подпрограммы
`N70 G0 X300 Y100 M30`	; отвод, конец программы

Дополнительная информация

Вращение в активной плоскости

При программировании с помощью `RPL=...` WCS поворачивается вокруг расположенной вертикально к активной плоскости оси.

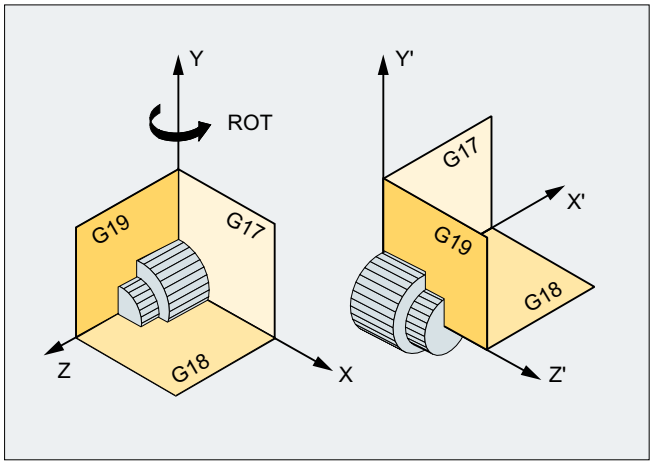


Рисунок 3-23 Вращение вокруг оси Y или в плоскости G18

⚠ ПРЕДУПРЕЖДЕНИЕ

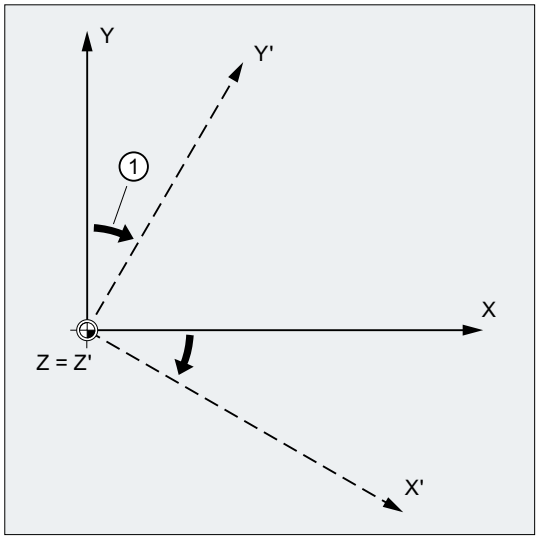
Смена плоскостей

Если после вращения программируется смена плоскостей (`G17`, `G18`, `G19`), то текущие углы поворота соответствующих осей сохраняются и действуют и в новой плоскости. Поэтому настоятельно рекомендуется перед сменой плоскостей сбросить текущие углы поворота на 0:

- `N100 ROT X0 Y0 Z0 ;` явное программирование угла
- `N100 ROT ;` не явное программирование угла

Абсолютное вращение с ROT X... Y... Z...

WCS поворачивается вокруг указанных осей на запрограммированный угол поворота.

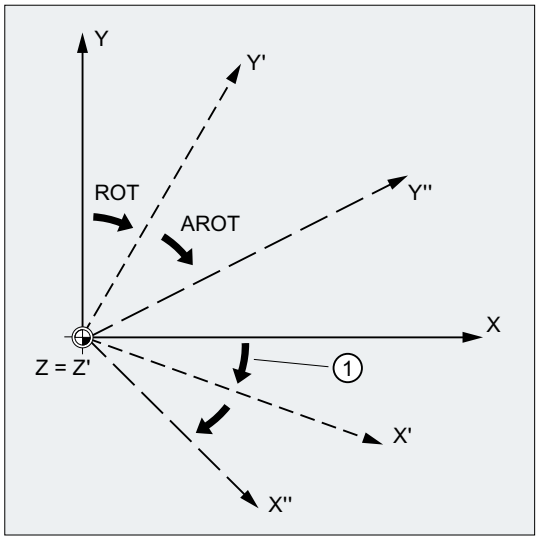


① Угол поворота

Рисунок 3-24 Абсолютное вращение вокруг оси Z

Аддитивное вращение с AROT X... Y... Z...

WCS продолжает вращение вокруг указанных осей на запрограммированный угол поворота.



① Угол поворота

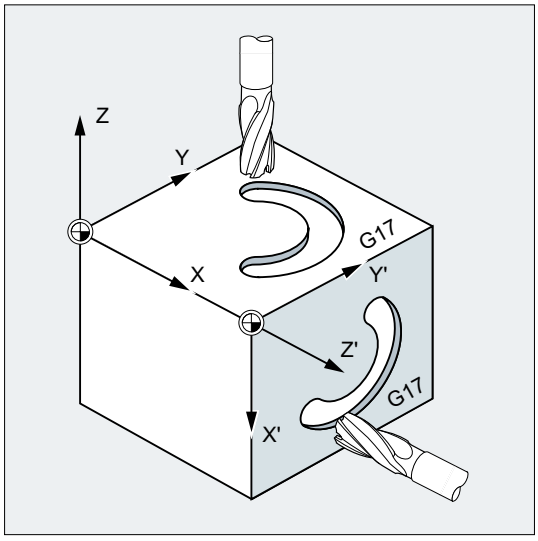
Рисунок 3-25 Абсолютное и аддитивное вращение вокруг оси Z

Вращение рабочей плоскости

При вращении с `ROT` / `AROT` также поворачивается и рабочая плоскость (`G17`, `G18`, `G19`).

Пример: Рабочая плоскость G17
WCS лежит на торце детали. Посредством смещения и вращения система координат смещается в одну из боковых поверхностей. Рабочая плоскость G17 также вращается. Тем самым движения перемещения в плоскости G17 могут и дальше программироваться через X и Y, а подачи через Z.

Условие:
Инструмент должен располагаться вертикально к рабочей плоскости, положительное направление оси подачи показывает в направлении зажима инструмента.
Посредством указания `CUT2DF` коррекция на радиус инструмента действует в повернутой плоскости.



3.12.6 Программируемые вращения фреймов с пространственными углами (ROTS, AROTS, CROTS)

С помощью команд `ROTS`, `AROTS` и `CROTS` вращения систем координат детали могут указываться в пространственных углах. Пространственные углы это углы, которые образуют прямые пересечения желаемой, повернутой в пространстве плоскости, с главными плоскостями еще не повернутой WCS.

Примечание

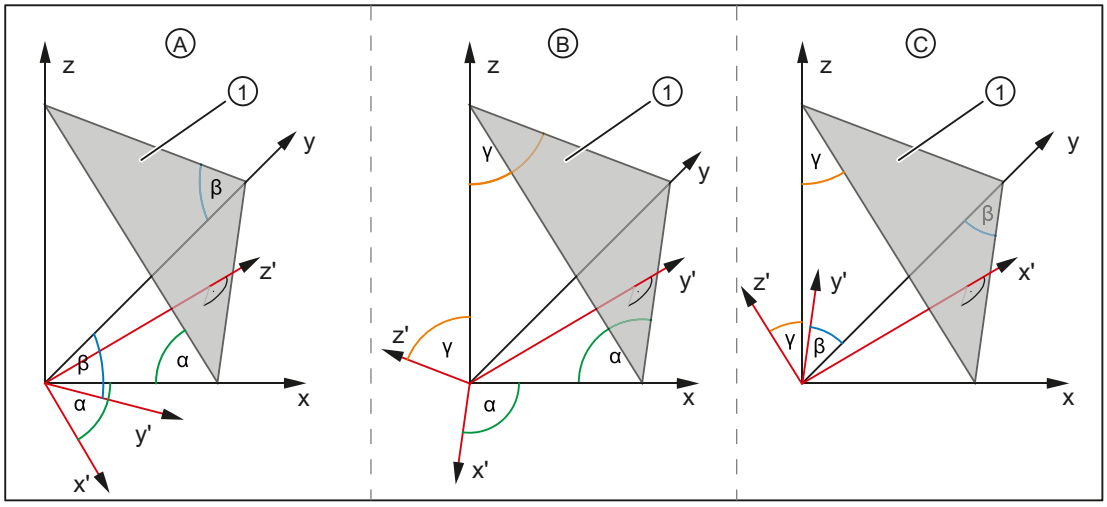
Идентификаторы гео-осей

В качестве примера для дальнейшего описания используется следующее определение:

- 1-я гео-ось: X
- 2-я гео-ось: Y
- 3-я гео-ось: Z

Из примера на рисунке ниже следует, что программирование `ROTS` `X`α `Y`β вызывает точную установку плоскости G17 WCS параллельно представленной наклонной плоскости. При этом положение нулевой точки WCS остается неизменным.

Ориентация повернутой WCS устанавливается таким образом, что первая повернутая ось лежит в плоскости, образуемой этой и 3-й осью исходной системы координат. В примере: X' лежит в исходной плоскости X/Z.



① Наклонная плоскость

α, β, γ Пространственный угол

A Точная установка плоскости G17 параллельно наклонной плоскости:

- 1-е вращение
  Поворот x вокруг y на угол α ⇒
  ось x' параллельно наклонной плоскости
- 2-е вращение
  Поворот y' вокруг x' на β ⇒
  ось y' параллельно наклонной плоскости
  ось ⇒ z' вертикально к наклонной плоскости
  ⇒ G17 параллельно наклонной плоскости

B Точная установка плоскости G18 параллельно наклонной плоскости:

- 1-е вращение
  Поворот z вокруг x на угол γ ⇒
  ось z' параллельно наклонной плоскости
- 2-е вращение
  Поворот x' вокруг z' на угол α ⇒
  ось x' параллельно наклонной плоскости
  ось ⇒ y' вертикально к наклонной плоскости
  ⇒ G18 параллельно наклонной плоскости

C Точная установка плоскости G19 параллельно наклонной плоскости:

- 1-е вращение
  Поворот y вокруг z на угол β ⇒
  ось y' параллельно наклонной плоскости
- 2-е вращение
  Поворот z' вокруг y' на угол γ ⇒
  ось z' параллельно наклонной плоскости
  ось ⇒ x' вертикально к наклонной плоскости
  ⇒ G19 параллельно наклонной плоскости

Синтаксис

Определения

Положение плоскости в пространстве однозначно определено через два пространственных угла. Указание третьего пространственного угла привело бы к переопределению плоскости. Это не допускается.

При программировании только одного пространственного угла выполняется поворот WCS, идентичный `ROT`, `AROT` (см. главу "Программируемое вращение (ROT, AROT, RPL) (Страница 325)").

Через обе запрограммированные оси плоскость устанавливается согласно определению областей для `G17`, `G18`, `G19`. Это устанавливает последовательность осей координат (1-я ось / 2-я ось плоскости) или последовательность поворотов на пространственный угол:

Плоскость	1-я ось	2-я ось
G17	X	Y
G18	Z	X
G19	Y	Z

Установка плоскости G17 ⇒ пространственный угол для X и Y

- 1-е вращение: X вокруг Y на угол α
- 2-е вращение: Y вокруг X' на угол β
- Ориентация: X' лежит в исходной плоскости Z/X.

```
ROTS  X<α> Y<β>
AROTS X<α> Y<β>
CROTS X<α> Y<β>
```

Установка плоскости G18 ⇒ пространственный угол для Z и X

- 1-е вращение: Z вокруг X на угол γ
- 2-е вращение: X вокруг Z' на угол α
- Ориентация: Z' лежит в исходной плоскости Y/Z.

```
ROTS  Z<γ> X<α>
AROTS Z<γ> X<α>
CROTS Z<γ> X<α>
```

Установка плоскости G19 ⇒ пространственный угол для Y и Z

- 1-е вращение: Y вокруг Z на угол β
- 2-е вращение: Z вокруг Y' на угол γ
- Ориентация: Y' лежит в исходной плоскости X/Y.

```
ROTS  Y<β> Z<γ>
AROTS Y<β> Z<γ>
CROTS Y<β> Z<γ>
```

Объяснение

ROTS:	Абсолютные вращения фреймов с пространственными углами, исходный фрейм: программируемый фрейм $P_PFRAME
AROTS:	Аддитивные вращения фреймов с пространственными углами; исходный фрейм: программируемый фрейм $P_PFRAME
CROTS:	Абсолютные вращения фреймов с пространственными углами, исходный фрейм: запрограммированный фрейм $P_...
X, Y, Z:	идентификаторы гео-осей (см. указание выше: Идентификаторы гео-осей)
α, β, γ:	пространственный угол относительно соответствующей гео-оси: • α → X • β → Y • γ → Z

3.12.7 Программируемый коэффициент масштабирования (SCALE, ASCALE)

Со SCALE/ASCALE для всех траекторных, синхронных и позиционирующих осей могут быть запрограммированы коэффициенты масштабирования для увеличения или уменьшения в направлении соответствующих указанных осей. Благодаря этому можно учитывать схожие геометрические формы или различные размеры усадки при программировании.

Синтаксис

```
SCALE X... Y... Z...
ASCALE X... Y... Z...
```

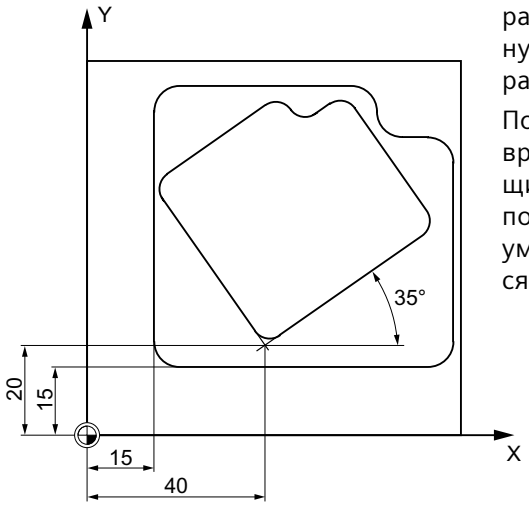
Примечание

Каждый фрейм-оператор программируется в отдельном кадре ЧПУ.

Значение

SCALE:	Абсолютное увеличение/уменьшение, относительно актуальной действующей, установленной с помощью G54 ... G57, G505 ... G599, системы координат
ASCALE:	Аддитивное увеличение/уменьшение, относительно актуальной действующей, установленной или запрограммированной системы координат
X... Y... Z...:	Коэффициенты масштабирования в направлении указанных гео-осей

Пример



У этой детали оба кармана встречаются два раза, но с различными размерами и повернутые друг к другу. Последовательность обработки зафиксирована в подпрограмме.

Посредством смещения нулевой точки и вращения устанавливаются соответствующие необходимые нулевые точки детали, посредством масштабирования контур уменьшается и после этого снова вызывается подпрограмма.

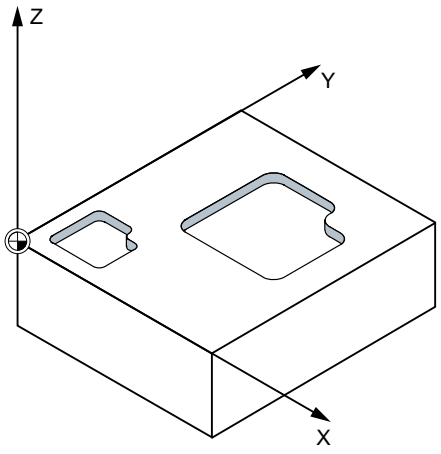
Программный код	Комментарий
`N10 G17 G54`	; Рабочая плоскость `X/Y`, нулевая точка детали
`N20 TRANS X15 Y15`	; Абсолютное смещение
`N30 L10`	; Изготовление большого кармана
`N40 TRANS X40 Y20`	; Абсолютное смещение
`N50 AROT RPL=35`	; Поворот в плоскости на 35°
`N60 ASCALE X0.7 Y0.7`	; Коэффициент масштабирования для маленького кармана
`N70 L10`	; Изготовление маленького кармана
`N80G0 X300 Y100 M30`	; отвод, конец программы

Дополнительная информация

SCALE X... Y... Z...

Для каждой оси может быть указан свой коэффициент масштабирования, на который может быть осуществлено увеличение или уменьшение. Масштабирование относится к установленной с помощью G54 ... G57, G505 ... G599 системе координат детали.

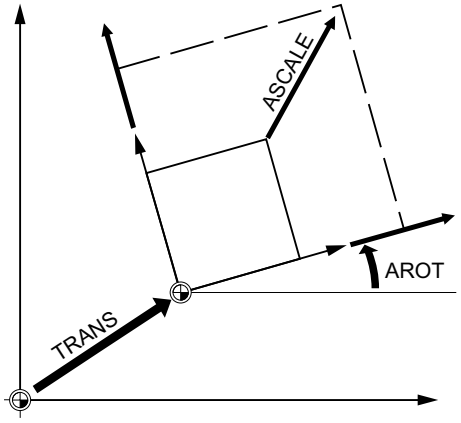
ВНИМАНИЕ
Не исходный фрейм Команда `SCALE` сбрасывает все фрейм-компоненты установленного до этого программируемого фрейма.



ASCALE X... Y... Z...

Изменение масштаба, которое должно надстраиваться на уже существующие фреймы, программируется с помощью ASCALE. В этом случае последний действующий коэффициент масштабирования умножается на новый.

Исходной точкой для изменения масштаба служит актуальная установленная или последняя запрограммированная система координат.



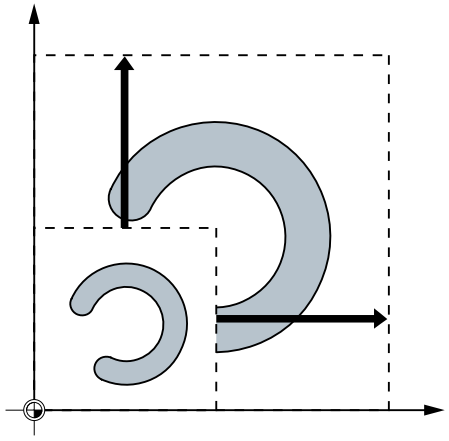
Масштабирование и смещение

Примечание

Если после SCALE программируется смещение с ATRANS, то значения смещения также масштабируются.

Различные коэффициенты масштабирования

ВНИМАНИЕ
Опасность столкновений Внимание при различных коэффициентах масштабирования! Круговые интерполяции могут, к примеру, масштабироваться только с одинаковыми коэффициентами.



Примечание

Но для программирования искаженных окружностей различные коэффициенты масштабирования могут использоваться целенаправленно.

3.12.8 Программируемое отражение (MIRROR, AMIRROR)

С помощью `MIRROR`/`AMIRROR` формы детали могут отражаться на оси координат. Все движения перемещения, запрограммированные после, к примеру, в подпрограмме, выполняются в отраженном виде.

Синтаксис

```
MIRROR X... Y... Z...
AMIRROR X... Y... Z...
```

Примечание

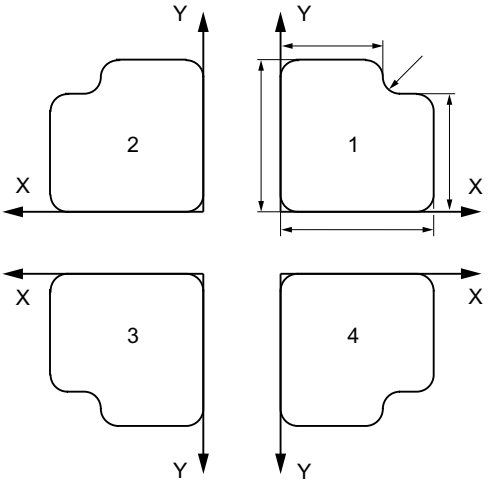
Каждый фрейм-оператор программируется в отдельном кадре ЧПУ.

Значение

MIRROR:	Абсолютное отражение, относительно актуальной действующей, установленной с помощью G54 ... G57, G505 ... G599, системы координат
AMIRROR:	Аддитивное отражение, относительно актуальной действующей, установленной или запрограммированной системы координат
X... Y... Z...:	Геометрическая ось, направление которой должно быть изменено. Указанное здесь значение выбирается свободно, к примеру, X0 Y0 Z0.

Примеры

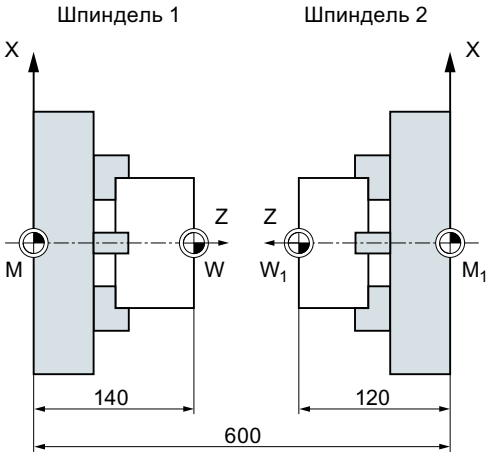
Пример 1: Фрезерная обработка



Показанный здесь контур программируется один раз как подпрограмма. Три следующих контура создаются через отражение. Нулевая точка детали располагается по центру к контурам.

Программный код	Комментарий
N10 G17 G54	; Рабочая плоскость X/Y, нулевая точка детали
N20 L10	; Изготовление первого контура справа вверху
N30 MIRROR X0	; Отражение оси X (в X изменяется направление)
N40 L10	; Изготовление второго контура слева вверху
N50 AMIRROR Y0	; Отражение оси Y (в Y изменяется направление)
N60 L10	; Изготовление третьего контура слева внизу
N70 MIRROR Y0	; MIRROR сбрасывает предыдущие фреймы. Отражение оси Y (в Y изменяется направление)
N80 L10	; Изготовление четвертого контура справа внизу
N90 MIRROR	; выключение отражения
N100 G0 X300 Y100 M30	; отвод, конец программы

Пример 2: Токарная обработка



Сама обработка сохраняется как подпрограмма, выполнение на соответствующем шпинделе реализуется через отражения и смещения.

Программный код	Комментарий
`N10 TRANS X0 Z140`	; Смещение нулевой точки на `W`
...	; Обработка 1-й стороны шпинделем 1
`N30 TRANS X0 Z600`	; Смещение нулевой точки на шпиндель 2
`N40 AMIRROR Z0`	; Отражение оси `Z`
`N50 ATRANS Z120`	; Смещение нулевой точки на `W1`
...	; Обработка 2-й стороны шпинделем 2

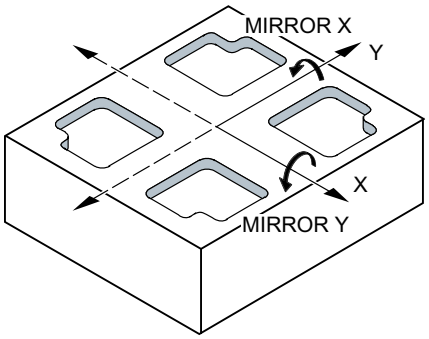
Дополнительная информация

MIRROR X... Y... Z...

Отражение программируется через осевую смену направления в выбранной рабочей плоскости.

Пример: рабочая плоскость G17 X/Y

Отражение (на оси Y) требует смены направления в X и поэтому программируется с `MIRROR X0`. Контур обрабатывается зеркально на противоположной стороне оси отражения Y.

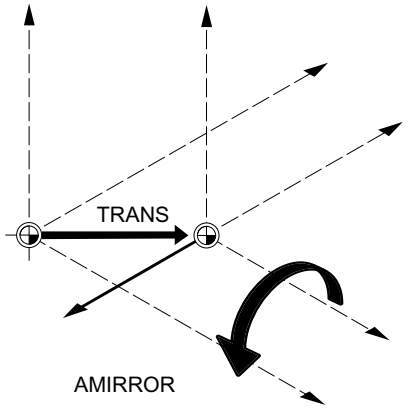


Отражение относится к актуальной действующей, установленной с помощью G54 ... G57, G505 ... G599, системе координат.

ВНИМАНИЕ
Не исходный фрейм Команда `MIRROR` сбрасывает все фрейм-компоненты установленного до этого программируемого фрейма.

AMIRROR X... Y... Z...

Отражение, которое должно надстраиваться на уже существующие трансформации, программируется с `AMIRROR`. Исходной точкой служит актуальная установленная или последняя запрограммированная система координат.



Выключение отражения

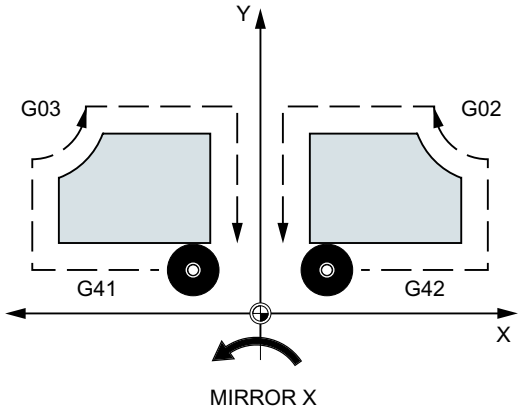
Для всех осей: `MIRROR` (без указания оси)

При этом сбрасываются все фрейм-компоненты запрограммированного до этого фрейма.

Коррекцией на радиус инструмента

Примечание

При команде отражения СЧПУ автоматически изменяет команды коррекции траектории (G41/G42 bzw. G42/G41) в соответствии с измененным направлением обработки.



Это же относится и к направлению вращения окружности (G2/G3 или G3/G2)

Примечание

Если после MIRROR программируется аддитивное вращение с AROT, то возможно придется работать с обратными направлениями вращения (положительное/отрицательное или отрицательное/положительное). Отражения в геометрических осях автоматически пересчитываются СЧПУ во вращения и при необходимости в отражения устанавливаемой с помощью машинных данных оси отражения. Это же относится и к устанавливаемым смещениям нулевой точки

Ось отражения

Через машинные данные можно установить, через какую ось будет выполнено отражение:

MD10610 $MN_MIRROR_REF_AX = <значение>

Величина	Объяснение
0	Отражение через запрограммированную ось (инверсия значений).
1	Ось X это ось отсчета.
2	Ось Y это ось отсчета.
3	Ось Z это ось отсчета.

Интерпретация запрограммированных значений

Через машинные данные можно установить, как должны быть интерпретированы запрограммированные значения:

MD10612 $MN_MIRROR_TOGGLE = <значение>

Величина	Объяснение
0	Запрограммированные значения осей не обрабатываются.
1	Запрограммированные значения осей обрабатываются: • При запрограммированных значениях осей ≠ 0 ось отражается, если она еще не отражена. • При запрограммированном значении оси = 0 отражение отключается.

3.12.9 Создание фрейма по точной установке инструмента (TOFRAME, TOROT, PAROT)

TOFRAME создает прямоугольную систему координат, ось Z которой совпадает с актуальной точной установкой инструмента. Благодаря этому пользователь может выполнить свободный ход инструмента в направлении Z без столкновений (к примеру, после поломки инструмента в 5-осевой программе).

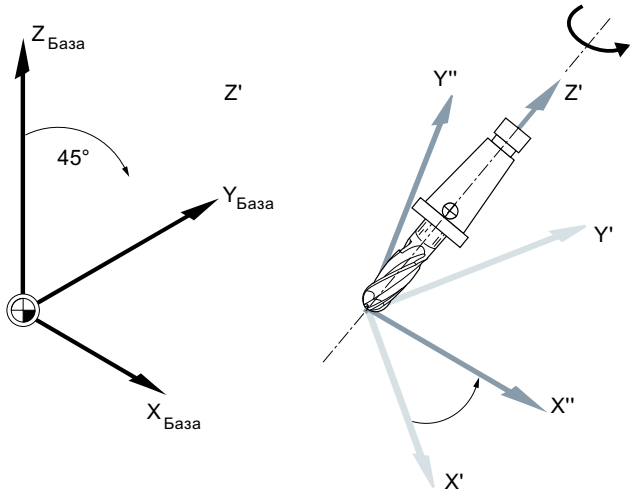
Положение обоих осей X и Y при этом зависит от установки в машинных данных MD21110 $MC_X_AXES_IN_OLD_X_Z_PLANE (система координат при автоматическом определении фрейма). Новая система координат либо остается такой, какой она получается из кинематики станка, либо дополнительно выполняется поворот вокруг новой оси Z таким образом, что новая ось X лежит в старой плоскости Z-X (см. указания изготовителя станка).

Результирующий фрейм, описывающий ориентацию, стоит в системной переменной для программируемого фрейма ($P_PFRAME).

С помощью TOROT в запрограммированном фрейме переписывается только доля вращения. Все остальные компоненты остаются неизменными.

TOFRAME и TOROT оптимизированы для фрезерных обработок, в которых обычно активна G17 (рабочая плоскость X/Y). Напротив, для токарных обработок или в общих случаях при активной G18 или G19 необходимы фреймы, у которых ось X или Y совпадает с точной установкой инструмента. Эти фреймы программируются с помощью команд TOFRAMEX/TOROTX или TOFRAMEY/TOROTY.

С PAROT система координат детали (WCS) точно устанавливается на детали.



Синтаксис

```
TOFRAME/TOFRAMEZ/TOFRAMEY/TOFRAMEX
...
TOROTOF

TOROT/TOROTZ/TOROTY/TOROTX
...
TOROTOF

PAROT
...
PAROTOF
```

Значение

TOFRAME:	Точно установить ось Z WCS через вращение фрейма параллельно ориентации инструмента
TOFRAMEZ:	как TOFRAME
TOFRAMEY:	Точно установить ось Y WCS через вращение фрейма параллельно ориентации инструмента
TOFRAMEX:	Точно установить ось X WCS через вращение фрейма параллельно ориентации инструмента
TOROT:	Точно установить ось Z WCS через вращение фрейма параллельно ориентации инструмента Определенное через TOROT вращение идентично таковому при TOFRAME.
TOROTZ:	как TOROT

TOROTY:	Точно установить ось Y WCS через вращение фрейма параллельно ориентации инструмента
TOROTX:	Точно установить ось X WCS через вращение фрейма параллельно ориентации инструмента
TOROTOF:	Отключить точную установку параллельно ориентации инструмента
PAROT:	Точной установить WCS через вращение фрейма на детали Смещения, масштабирования и отражения в активном фрейме сохраняются.
PAROTOF:	Активированное с PAROT относящееся к детали вращение фрейма отключается с PAROTOF .

Примечание

С помощью команды TOROT достигается связное программирование для активных ориентируемых инструментальных суппортов для любого типа кинематики.

Аналогично ситуации для вращающегося инструментального суппорта, с помощью PAROT можно активировать вращение инструментального стола. Таким образом, определяется фрейм, который изменяет положение системы координат детали так, что движение компенсации станка не возникает. Языковая команда PAROT не отклоняется, если нет активного ориентируемого инструментального суппорта.

Пример

Программный код	Комментарий
N100 G0 G53 X100 Z100 D0	
N120 TOFRAME	
N140 G91 Z20	; TOFRAME учитывается, все запрограммированные движения гео-осей относятся к новой системе координат.
N160 X50	
...	

Дополнительная информация

Согласование осевого направления

Если вместо TOFRAME / TOFRAMEZ или TOROT / TOROTZ программируется одна из команд TOFRAMEX, TOFRAMEY, TOROTX, TOROTY, то действуют согласования направлений осей согласно этой таблице:

Команда	Направление инструмента (аппликата)	Вспомогательная ось (абсцисса)	Вспомогательная ось (ордината)
TOFRAME / TOFRAMEZ / TOROT / TOROTZ	Z	X	Y
TOFRAMEY / TOROTY	Y	Z	X
TOFRAMEX / TOROTX	X	Y	Z

Собственный системный фрейм для TOFRAME или TOROT

Созданные через TOFRAME или TOROT фреймы могут быть записаны в собственный системный фрейм $P_TOOLFRAME. Для этого должен быть установлен бит 3 в машинных данных MD28082 $MC_MM_SYSTEM_FRAME_MASK. Программируемый фрейм при этом сохраняется неизменным. Различия получаются при дальнейшей обработке программируемого фрейма.

3.12.10 Отключение фрейма (G53, G153, SUPA, G500)

При выполнении определенных процессов, к примеру, подвод к точке смены инструмента, необходимо определение и точное по времени подавление по времени различных фрейм-компонентов.

Настраиваемые фреймы могут либо отключаться модально, либо подавляться покадрово.

Программируемые фреймы могут подавляться покадрово или удаляться.

Синтаксис

```
G53
G153
SUPA
G500
TRANS
ROT
SCALE
MIRROR
```

Значение

G53:	Действующее покадрово подавление всех программируемых и настраиваемых фреймов
G153:	G153 действует как G53 и дополнительно подавляет общий базовый фрейм ($P_ACTBFRAME)
SUPA:	SUPA действует как G153 и дополнительно подавляет: • смещения маховичком (DRF) • наложенные движения • внешнее смещение нулевой точки • смещение PRESET
G500:	Действующее модально выключение всех настраиваемых фреймов (G54 ... G57, G505 ... G599), если в G500 не стоит значение.
TRANS ROT SCALE MIRROR:	без указания оси вызывает удаление программируемых фреймов.

3.12.11 Программирование: Отключение наложений (спец. для оси) (CORROF)

Процедура CORROF удаляет следующие спец. для оси наложения:

- Настраиваемые движением маховичка аддитивные смещения нулевой точки (смещения DRF)
- Запрограммированные системной переменной $AA_OFF смещения позиции

Удаление значения наложения вызывает остановку предварительной обработки, и компонент позиции отключенного наложенного движения передается в позицию в базовой кинематической системе. При этом ни одна из осей не перемещается.

Считываемое через системную переменную $AA_IM (текущее заданное значение **MCS** оси) значение позиции в системе координат **станка не изменяется**.

Считываемое через системную переменную $AA_IW (актуальное заданное значение **WCS**оси) значение позиции в системе координат **детали изменяется**, поскольку теперь оно содержит отключенную долю наложенного движения.

Примечание

В **программе ЧПУ** разрешено программировать `CORROF`.

В **синхронном действии** программировать `CORROF` **не разрешено**.

Синтаксис

`CORROF` (<ось>,"<строка>"[,<ось>,"<строка>"])

Объяснение

`CORROF`:	Процедура отключения следующих смещений или наложений оси: • Смещение DRF • Смещения позиции ($AA_OFF)	
	Активность:	модально
<ось>:	Идентификатор оси (идентификатор оси канала, геометрической оси или оси станка)	
	Тип данных:	AXIS
<строка>:	строка символов для определения типа наложения	
	Тип данных:	BOOL
	Значение	**Объяснение**
	`DRF`	Смещение DRF
	`AA_OFF`	Смещение позиции ($AA_OFF)

Примеры

Пример 1: Спец. для оси отключение смещения DRF (1)

Через перемещение маховичком DRF создается смещение DRF в оси X. Для всех других осей канала смещения DRF не действуют.

Программный код	Комментарий
`N10 CORROF(X,"DRF")`	`; CORROF` здесь действует как `DRFOF`.
...	

Пример 2: Спец. для оси отключение смещения DRF (2)

Через перемещение маховичком DRF создается смещение DRF в оси X и в оси Y. Для всех других осей канала смещения DRF не действуют.

Программный код	Комментарий
; Отключается только смещение `DRF` оси `X`, смещение `DRF` оси `Y` сохраняется.	
; При `DRFOF( )` оба смешения были бы отключены.	
`N10 CORROF(X,"DRF")`	
...	

Пример 3: Спец. для оси отключение смещения позиции $AA_OFF

Программный код	Комментарий
; Для оси X интерполируется смещение позиции == 10.	
`N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5`	
...	
; Смещение позиции оси `X` отключается: `$AA_OFF[X]=0`	
; Ось `X` не перемещается.	
; К актуальной позиции оси `X` прибавляется смещение позиции.	
`N80 CORROF(X,"AA_OFF")`	
...	

Пример 4: Спец. для оси отключение смещения DRF и смещения позиции $AA_OFF (1)

Через перемещение маховичком DRF создается смещение DRF в оси X. Для всех других осей канала смещения DRF не действуют.

Программный код	Комментарий
; Для оси X интерполируется смещение позиции 10.	
`N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5`	
...	
; Отключается только смещение `DRF` и смещение позиции оси `X`.	
; Смещение `DRF` оси `Y` сохраняется.	
`N70 CORROF(X,"DRF",X,"AA_OFF")`	
...	

Пример 5: Спец. для оси отключение смещения DRF и смещения позиции $AA_OFF (2)

Через перемещение маховичком DRF создается смещение DRF в оси X и в оси Y. Для всех других осей канала смещения DRF не действуют.

Программный код	Комментарий
; Для оси X интерполируется смещение позиции == 10.	
`N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5`	
...	
; Отключается смещение `DRF` оси `Y` и смещение позиции оси `X`.	
; Смещение `DRF` оси `X` сохраняется.	
`N70 CORROF(Y,"DRF",X,"AA_OFF")`	
...	

Дополнительная информация

$AA_OFF_VAL

После отключения смещения позиции из-за $AA_OFF, системная переменная $AA_OFF_VAL (интегрированный путь наложения оси) соответствующей оси равна нулю.

$AA_OFF в режиме работы JOG

И в режиме работы JOG при изменении $AA_OFF происходит интерполяция смещения позиции как наложенное движение, если эта функция разрешена через машинные данные MD36750 $MA_AA_OFF_MODE.

$AA_OFF в синхронном действии

Если при отключении смещения позиции через `CORROF(`<ось>`,"AA_OFF")` активно синхронное действие, которое немедленно снова устанавливает $AA_OFF (`DO $AA_OFF[`<ось>]=<значение>), то $AA_OFF отключается и больше не устанавливается, и отображается аварийное сообщение 21660. Если же синхронное действие активируется позднее, к примеру, в кадре после `CORROF`, то $AA_OFF устанавливается и выполняется интерполяция смещения позиции.

Автоматический переход оси канала

Если ось, для которой была запрограммирована `CORROF`, активна в другом канале, то через переход оси она передается в канал (условие: MD30552 $MA_AUTO_GET_TYPE > 0) и после отключение смещения позиции и/или смещения DRF.

3.12.12 Отключение аддитивных смещений нулевой точки (DRFOF)

Настраиваемые движением маховичка аддитивные смещения нулевой точки (смещения DRF) отключаются посредством процедуры `DRFOF`.

Отключение вызывает остановку предварительной обработки, и доля позиции отключенного смещения DRF передается в позицию в базовой кинематической системе, т. е. при этом оси не перемещаются. Значение системной переменной $AA_IM (актуальное заданное значение MCS оси) не изменяется, значение системной

переменной $AA_IW (актуальное заданное значение WCS оси) изменяется, т. к. оно теперь содержит отключенную долю из наложенного движения.

Синтаксис

```
DRFOF
```

Значение

DRFOF:	Процедура выключения смещений DRF для всех активных осей канала	
	Активность:	модально

Примеры

Пример 1: Спец. для оси отключение смещения DRF (1)

Через перемещение маховичком DRF создается смещение DRF в оси X. Для всех других осей канала смещения DRF не действуют.

Программный код	Комментарий
`N10 CORROF(X,"DRF")`	`; CORROF` здесь действует как `DRFOF`.
...	

Пример 2: Спец. для оси отключение смещения DRF (2)

Через перемещение маховичком DRF создается смещение DRF в оси X и в оси Y. Для всех других осей канала смещения DRF не действуют.

Программный код	Комментарий
`N10 CORROF(X,"DRF")`	; Отключается только смещение `DRF` оси `X`, смещение `DRF` оси `Y` сохраняется (при `DRFOF` были бы отключены оба смещения).
...	

Пример 3: Спец. для оси отключение смещения DRF и смещения позиции $AA_OFF (1)

Через перемещение маховичком DRF создается смещение DRF в оси X. Для всех других осей канала смещения DRF не действуют.

Программный код	Комментарий
`N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5`	; Для оси X интерполируется смещение позиции == 10.
...	
`N70 CORROF(X,"DRF",X,"AA_OFF")`	; Отключается только смещение `DRF` и смещение позиции оси `X`, смещение `DRF` оси `Y` сохраняется.
...	

Пример 4: Спец. для оси отключение смещения DRF и смещения позиции $AA_OFF (2)

Через перемещение маховичком DRF создается смещение DRF в оси X и в оси Y. Для всех других осей канала смещения DRF не действуют.

Программный код	Комментарий
`N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5`	; Для оси X интерполируется смещение позиции == 10.
...	
`N70 CORROF(Y,"DRF",X,"AA_OFF")`	; Отключается смещение `DRF` оси `Y` и смещение позиции оси `X`, смещение `DRF` оси `X` сохраняется.
...	

3.12.13 Специфическое для шлифования смещение нулевой точки (GFRAME0, GFRAME1 ... GFRAME100)

Команда активации циклового фрейма в канале

Программирование команды `GFRAME<n>` активирует соответствующий цикловый фрейм хранения данных $P_GFR[<n>] в канале. Для этого активный цикловый фрейм $P_GFRAME приравнивается к цикловому фрейму хранения данных $P_GFR[<n>]:

`GFRAME<n>` ⇒ $P_GFRAME = $P_GFR[<n>]

Команда	Цикловый фрейм, активированный в канале
`GFRAME0`	$P_GFR[0] (**нулевой фрейм**)
`GFRAME1`	$P_GFR[1]
...	...
`GFRAME100`	$P_GFR[100]

Синтаксис

`GFRAME<n>`

Объяснение

`GFRAME<n>`:	Активация циклового фрейма <n> хранения данных	
	G-группа:	64
	Первичная установка:	MD20150 $MC_GCODE_RESET_VALUES[**63**]
	Активность:	Модально
<n>:	Номер циклового фрейма	
	Диапазон значений:	0, 1, 2, ... 100

3.13 Вывод вспомогательных функций

Функция

С помощью вывода вспомогательных функций осуществляется своевременное сообщение на PLC, когда программа обработки деталей хочет совершить определенные действия по переключению станка через PLC. Это осуществляется посредством передачи соответствующих вспомогательных функций с их параметрами на интерфейс PLC. Обработка переданных значений и сигналов должна осуществляться через программу электроавтоматики.

Вспомогательные функции

Следующие вспомогательные функции могут быть переданы на PLC:

Вспомогательная функция	**Адрес**
Выбор инструмента	T
Коррекция инструмента	D, DL
Подача	F / FA
Число оборотов шпинделя	S
Функции M	M
Функции H	H

Для каждой группы функций или отдельной функции с помощью машинных данных устанавливается, будет ли вывод запущен **перед**, **при** или **после** движения перемещения.

PLC может использоваться для выполнения различных процессов квитирования для вывода вспомогательных функций.

Свойства

Важные свойства вспомогательных функций приведены в следующей обзорной таблице:

Функция	Расширение адреса		Значение			Объяснения	Макс. кол-во на кадр
	Объяснение	Диапазон	Диапазон	Тип	Объяснение		
M	-	0 (не явно)	0 ... 99	INT	Функция	Для диапазона значений между 0 и 99 расширение адреса 0. Принудительно без расширения адреса: M0, M1, M2, M17, M30	5
	№ шпинделя	1 ... 12	1 ... 99	INT	Функция	M3, M4, M5, M19, M70 с расширением адреса шпиндель № (например, `M2=5` ; останов шпинделя для шпинделя 2). Без Nr. шпинделя функция действует для мастер-шпинделя.	
	Любое	0 – 99	100 ... 2147483647	INT	Функция	Функция M пользователя*	
S	№ шпинделя	1 ... 12	0 ... ± 1,8*10^{308}	REAL	Скорость	Без Nr. шпинделя функция действует для мастер-шпинделя.	3
H	Любое	0 – 99	0 ... ± 2147483647 ± 1,8*10^{308}	 INT REAL	Любое	Функции не действуют в ЧПУ, реализуются только через PLC.*	3
T	Номер шпинделя (при активном управлении инструментом)	1 ... 12	0 - 32000 (и имена инструмента при активном управлении инструментом)	INT	Выбор инструмента	Имена инструмента не передаются на интерфейс PLC.	1
D	-	-	0 – 12	INT	Выбор коррекции инструмента	D0: Отмена Предустановка: D1	1
DL	Зависящая от места коррекция	1 – 6	0 ... ± 1,8*10^{308}	REAL	Выбор коррекции инструмента точной	Относится к выбранному до этого номеру D	1
F	-	-	0,001 – 999 999,999	REAL	Подача по траектории		6
FA	Номер оси	1 – 31	0,001 – 999 999,999	REAL	Осевая подача		
* Значение функций определяется изготовителем станка (**см. Указания изготовителя станка!**).							

Дополнительная информация

Количество выводимых функций на кадр ЧПУ

В одном кадре ЧПУ может быть запрограммировано максимум 10 выводов функций. Вспомогательные функции могут выводиться и из компонента действия **синхронных действий**.

Группировка

Названные функции могут объединяться в группы. Для некоторых команд M подразделение групп уже задано. С помощью группировки может устанавливаться характеристика квитирования.

Быстрый вывод функций (QU)

Функции, не сконфигурированные для быстрого вывода, могут быть определены для отдельных выводов с помощью кодового слова `QU` в качестве быстрого вывода. Выполнение программы продолжается без ожидания квитирования исполнения дополнительной функции (ожидание квитирования передачи осуществляется). Благодаря этому удается избежать ненужных точек остановки и прерываний движений перемещения.

Примечание

Для функции "Быстрые выводы функций" должны быть установлены соответствующие машинные данные (→ **Изготовитель станка!**).

Вывод функций при движениях перемещения

Передача информации и ожидание соответствующих реакций занимает время, влияя тем самым на движения перемещения.

Быстрое квитирование без задержки смены кадров

Управление поведением при смене кадров возможно через машинные данные. При установке "без задержки смены кадров" получаются следующие параметры для быстрых вспомогательных функций:

Вывод вспомогательной функции	**Поведение**
Перед движением	Переход кадров между кадрами с быстрыми вспомогательными функциями осуществляется **без** прерывания и **без** уменьшения скорости. Вывод вспомогательных функций осуществляется в первом такте интерполятора кадра. Последующий кадр выполняется без задержки квитирования.
При движении	Переход кадров между кадрами с быстрыми вспомогательными функциями осуществляется **без** прерывания и **без** уменьшения скорости. Вывод вспомогательных функций осуществляется в ходе кадра. Последующий кадр выполняется без задержки квитирования.
После движения	Движение останавливается в конце кадра. Вывод вспомогательных функций осуществляется в конце кадра. Последующий кадр выполняется без задержки квитирования.

⚠ ОСТОРОЖНО

Вывод функций в режиме управления траекторией

Вывод функций **перед** движениями перемещения прерывает режим управления траекторией (G64 / G641) и вызывает точный останов для предыдущего кадра.

Вывод функций **после** движений перемещения прерывает режим управления траекторией (G64 / G641) и создает точный останов для текущего кадра.

Важно: Ожидание необходимого сигнала квитирования от PLC также может привести к прерыванию режима управления траекторией, к примеру, цепочки команд M в кадрах с очень короткими длинами ходами траектории.

3.13.1 Функции M

С помощью функций M запускаются действия по переключению, например, "ВКЛ/ВЫКЛ СОЖ" и прочие функции на станке.

Синтаксис

M<значение>
M[<расширение адреса>]=<значение>

Объяснение

M :	Адрес для программирования функций M	
<расширение адреса>:	Для некоторых функций M действует расширенное написание адреса (например, указание номера шпинделя для функций шпинделя).	
<значение>:	Посредством присваивания значения (номер функции M) осуществляется согласование с определенной функцией станка.	
	Тип:	INT
	Диапазон значений:	0 ... 2147483647 (макс. значение INT)

Предопределенные функции M

Некоторым важным для выполнения программы функциям M в стандартном исполнении СЧПУ уже присвоены значения:

Функция M	Объяснение
M0*	Запрограммированная остановка
M1*	Остановка по выбору
M2*	Конец главной программы (как M30)
M3	Правое вращение шпинделя
M4	Левое вращение шпинделя

Функция M	Объяснение
M5	Остановка шпинделя
M6	Смена инструмента (стандартная настройка)
M17*	Конец подпрограммы
M19	Позиционировать шпиндель
M30*	Конец главной программы (как M2)
M40	Автоматическое переключение редуктора
M41	Ступень редуктора 1
M42	Ступень редуктора 2
M43	Ступень редуктора 3
M44	Ступень редуктора 4
M45	Ступень редуктора 5
M70	Шпиндель переключается в осевой режим

Примечание

Для функций, обозначенных *, расширенное написание адреса не допускается.

Функции M0, M1, M2, M17 и M30 всегда запускаются **после** движения перемещения.

Определенные изготовителем станка функции M

Все свободные номера функций M могут быть заняты изготовителем станка, например, функциями переключения для управления зажимными приспособлениями или для включения/выключения других функций станка.

Примечание

Назначенные свободным номерам функций M функциональности являются спец. для станка. Поэтому определенная функция M может иметь различную функциональность на различных станках.

Доступные на станке функции M и их функциональность см. Данные изготовителя станка.

Примеры

Пример 1: Макс. число функций M в кадре

Программный код	Комментарий
`N10 S...`	
`N20 X... M3`	`;` M-функция в кадре с движением оси,
	; шпиндель разгоняется до движения оси `X`.
`N180 M789 M1767 M100 M102 M376`	; Макс. `5` M-функций в кадре.

Пример 2: Функция M как быстрый вывод

Программный код	Комментарий
`N10 H=QU(735)`	; Быстрый вывод для `H735`.
`N10 G1 F300 X10 Y20 G64`	
`N20 X8 Y90 M=QU(7)`	; Быстрый вывод для `M7`.

M7 была запрограммирована как быстрый вывод, поэтому режим управления траекторией (G64) не прерывается.

Примечание

Использовать эту функцию только в отдельных случаях, так как при взаимодействии с выводом других функций изменяется временное согласование.

Дополнительная информация по предопределенным командам M

Запрограммированная остановка: M0

В кадре ЧПУ с `M0` обработка останавливается. Теперь, к примеру, можно удалить стружку, осуществить дополнительное измерение и т.д.

Запрограммированная остановка 1 - остановка по выбору: M1

M1 может быть установлена через:

- HMI/диалог "Управление программой"
  или
- Интерфейс ЧПУ/PLC

Обработка программы ЧПУ останавливается на запрограммированных кадрах.

Запрограммированная остановка 2 - ассоциированная с M1 вспомогательная функция с остановом в ходе программы

Запрограммированная остановка 2 может быть установлена через HMI/диалог "Управление программой" и позволяет прерывать технологические процессы в любое время на конце обрабатываемого сегмента. Тем самым оператор может вмешиваться в текущее производство, к примеру, для удаления сливной стружки.

Конец программы: M2, M17, M30

Программа завершается с `M2`, `M17` или `M30`. Если главная программа вызывается из другой программы (как подпрограмма), то `M2` / `M30` действует как `M17` и наоборот, т. е. `M17` действует в главной программе как `M2` / `M30`.

Функции шпинделя: M3, M4, M5, M19, M70

Для всех функцией шпинделя действует расширенное написание адреса с указанием номера шпинделя.

Пример:

Программный код	Комментарий
`M2=3`	; Правое вращение для второго шпинделя

Если расширение адреса не запрограммировано, то функция действует для мастер-шпинделя.

3.14 Дополнительные команды

3.14.1 Сообщения (MSG)

С помощью команды `MSG()` любая строка символов может быть выведена как сообщение для оператора из программы обработки детали.

Синтаксис

```
MSG("<текст сообщения>"[,<исполнение>])
...
MSG ()
```

Объяснение

`MSG`:	Предопределенный вызов подпрограммы для вывода сообщения		
<текст сообщения>:	Любая строка символов для индикации в качестве сообщения		
	Тип:	STRING	
	Максимальная длина:	124 символов; двухстрочная индикация (2*62 символов)	
	В тексте сообщения благодаря использованию связывающего оператора "<<" могут выводиться и переменные.		
<исполнение>:	Параметр для установки момента времени для записи сообщения (опция)		
	Тип:	INT	
	Значение:	0 (первичная установка)	Отдельный кадр главного хода для записи сообщения не создается. Она выполняется в следующем исполняемом кадре ЧПУ. Активный режим управления траекторией не прерывается.
		1	Создается отдельный кадр главного хода для записи сообщения. Активный режим управления траекторией прерывается.
`MSG()`:	Посредством программирования `MSG()` без текста сообщения, актуальное сообщение снова удаляется. Без удавления индикация сохраняется до появления следующего сообщения.		

Примечание

Если сообщение должно быть выведено на активном языке интерфейса, то пользователю необходима информация о текущем установленном на HMI языке. Эта информация запрашивается в программе обработки детали и в синхронных действиях через системную переменную $AN_LANGUAGE_ON_HMI (Страница 1364).

Примеры

Пример 1: Вывести / удалить сообщение

Программный код	Комментарий
`N10 G91 G64 F100`	; режим управления траекторией
`N20 X1 Y1`	
`N... X... Y...`	
`N20 MSG ("Обработка Часть 1")`	; Сообщение выводится только с `N30`.
	; Режим управления траекторией сохраняется.
`N30 X... Y...`	
`N... X... Y...`	
`N400 X1 Y1`	
`N410 MSG ("Обработка Часть 2",1)`	; Сообщение выводится с `N410`.
	; Режим управления траекторией прерывается.
`N420 X1 Y1`	
`N... X... Y...`	
`N900 MSG ()`	; Удалить сообщение.

Пример 2: Текст сообщения с переменной

Программный код	Комментарий
`N10 R12=$AA_IW[X]`	; Актуальная позиция оси X в `R12`.
`N20 MSG("Проверить позицию оси X"<<R12<<"")`	; Вывести сообщение с переменной `R12`.
...	
`N90 MSG ()`	; Удалить сообщение из `N20`.

3.14.2 Запись строки символов в переменную BTSS (WRTPR)

С помощью функции `WRTPR()` можно записать любую строку символов из программы обработки детали в переменную BTSS progProtText.

Синтаксис

`WRTPR(`<строка>`[,`<ExecTime>`])`

Значение

WRTPR:	вызов функции для вывода строки символов.		
<строка>:	Любая строка символов, записываемая в переменную BTSS progProtText.		
	Тип:	String	
	Максимальная длина:	128 символов	
<ExecTime>:	опциональный параметр для установки момента времени записи строки символов.		
	Тип:	INT	
	Диапазон значений:	0, 1	
		0 (по умолчанию)	Для записи строки символов отдельный кадр главного хода не создается. Она выполняется в следующем исполняемом кадре ЧПУ. Активный режим управления траекторией не прерывается.
		1	Для записи строки символов создается отдельный кадр главного хода. Активный режим управления траекторией прерывается.

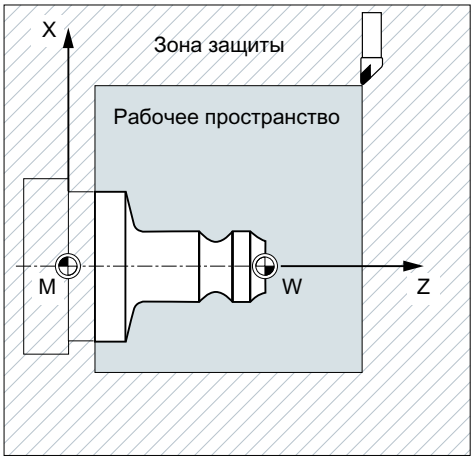
Примеры

Программный код	Комментарий
`N10 G91 G64 F100`	; режим управления траекторией
`N20 X1 Y1`	
`N30 WRTPR("N30")`	; Строка символов «N30» записывается только в N40.
	; Режим управления траекторией сохраняется.
`N40 X1 Y1`	
`N50 WRTPR("N50",1)`	; Строка символов «N50» записывается в N50.
	; Режим управления траекторией прерывается.
`N60 X1 Y1`	

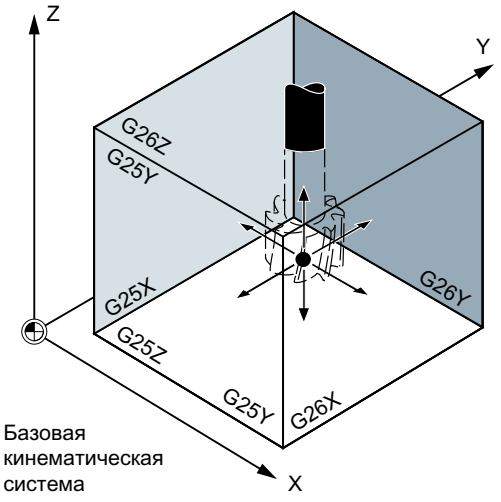
3.14.3 Ограничение рабочего поля

3.14.3.1 Ограничение рабочего поля в BKS (G25/G26, WALIMON, WALIMOF)

С помощью G25/G26 можно ограничить рабочее пространство (рабочее поле, рабочую зону), в котором должен перемещаться инструмент, во всех осях канала. Области вне определенных с помощью G25/G26 границ рабочего поля заблокированы для движений инструмента.



Указания координат для отдельных осей действуют в базовой кинематической системе:



Ограничение рабочего поля для всех установленных действующими осей должно быть запрограммировано с помощью команды WALMON. WALIMOF делает ограничение рабочего поля не действительным. WALIMON это стандартная установка, поэтому программирование необходимо только в том случае, если до этого ограничение рабочего поля было выключено.

Синтаксис

```
G25 X…Y…Z…
G26 X… Y… Z…
WALIMON
...
WALIMOF
```

Значение

G25:	**Нижнее** ограничение рабочего поля Присваивание значений в осях канала в базовой кинематической системе
G26:	**Верхнее** ограничение рабочего поля Присваивание значений в осях канала в базовой кинематической системе
X… Y… Z…:	Нижние или верхние границы рабочего поля для отдельных осей канала Данные относятся к базовой кинематической системе.
WALIMON:	**Вкл**ючить ограничение рабочего поля для всех осей
WALIMOF:	**Выкл**ючить ограничение рабочего поля для всех осей

Наряду с программируемым вводом значений через G25/G26 возможен и ввод через спец. для оси установочные данные:

SD43420 $SA_WORKAREA_LIMIT_PLUS (ограничение рабочего поля плюс)

SD43430 $SA_WORKAREA_LIMIT_MINUS (ограничение рабочего поля минус)

Активация и деактивация спараметрированного через SD43420 и SD43430 ограничения рабочего поля осуществляется спец. для направления через начинающие действовать сразу же спец. для оси установочные данные:

SD43400 $SA_WORKAREA_PLUS_ENABLE (ограничение рабочего поля в положительном направлении активно)

SD43410 $SA_WORKAREA_MINUS_ENABLE (ограничение рабочего поля в отрицательном направлении активно)

Посредством спец. для направления активации/деактивации можно ограничить рабочую зону для оси только в одном направлении.

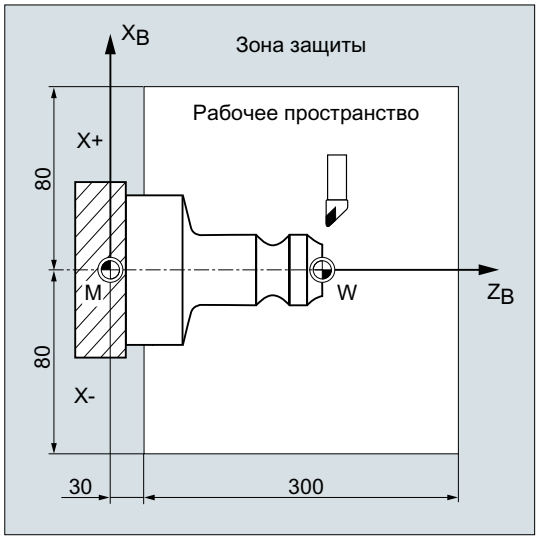
Примечание

Запрограммированное с G25/G26 ограничение рабочего поля имеет приоритет и переписывает введенные в SD43420 и SD43430 значения.

Примечание

С помощью G25/G26 по адресу S могут быть запрограммированы и предельные значения для скорости шпинделя. Подробности об этом см. " Программируемое ограничение числа оборотов шпинделя (G25, G26) (Страница 109) ".

Пример



Посредством ограничения рабочего поля с G25/26 рабочее пространство токарного станка ограничивается таким образом, что окружающие устройства, как то револьвер, измерительная станция и т.д. защищены от повреждений.

Основная установка: WALIMON

Программный код	Комментарий
N10 G0 G90 F0.5 T1	
N20 G25 X-80 Z30	; Определение нижнего ограничения для отдельных осей координат
N30 G26 X80 Z330	; Определение верхнего ограничения
N40 L22	; Программа обработки резаньем
N50 G0 G90 Z102 T2	; К точке смены инструмента
N60 X0	
N70 WALIMOF	; Выключение ограничения рабочего поля
N80 G1 Z-2 F0.5	; Сверление
N90 G0 Z200	; Назад
N100 WALIMON	; Включение ограничения рабочего поля
N110 X70 M30	; конец программы

Дополнительная информация

Опорная точка на инструменте

При активной коррекции длин инструмента в качестве исходной точки контролируется острие инструмента, в иных случаях исходная точка инструментального суппорта.

Учет радиуса инструмента должен быть активирован отдельно. Это осуществляется через спец. для канала машинные данные:

MD21020 $MC_WORKAREA_WITH_TOOL_RADIUS

Если исходная точка инструмента находится вне определенного через ограничение рабочего поля рабочего пространства или выходит из этой зоны, то выполнение программы останавливается.

Примечание

Если активны трансформации, то учет данных инструмента (длина инструмента и радиус инструмента) может отличаться от описанного поведения.

Программируемое ограничение рабочего поля, G25/G26

Для каждой оси можно установить верхнее (G26) и нижнее (G25) ограничение рабочего поля. Эти значения начинают действовать сразу же и сохраняются при соответствующей установке MD (→ MD10710 $MN_PROG_SD_RESET_SAVE_TAB) после RESET и повторного включения.

Примечание

С помощью предопределенной функции CALCPOSI (Страница 602) перед движениями перемещения можно проверить, будет ли предусмотренный путь пройден с учетом ограничений рабочего поля и/или защищенных областей.

3.14.3.2 Ограничение рабочего поля в WCS/ENS (WALCS0 ... WALCS10)

«Ограничение рабочего поля в WCS/ENS» обеспечивает гибкое, специфическое для детали ограничение диапазона перемещения осей канала в системе координат детали (WCS) или настраиваемой системе нулевой точки (ENS). Оно предназначено главным образом для применения в сфере обычных токарных станков.

Условие

Оси канала должны быть реферированы.

Группа ограничений рабочего поля

Чтобы при переключении согласований осей, например при включении/выключении трансформаций или активного фрейма, не потребовалось заново записывать специфические для оси границы рабочего поля для всех осей канала, предусмотрены группы ограничений рабочего поля.

Группа ограничений рабочего поля включает в себя следующие данные:

- Границы рабочего поля для всех осей канала
- Базовая система ограничения рабочего поля

Синтаксис

```
...
$P_WORKAREA_CS_COORD_SYSTEM[<WALimNo>]=<Value>
```

```
$P_WORKAREA_CS_PLUS_ENABLE[<WALimNo>,<Ax>]=<Value>
$P_WORKAREA_CS_LIMIT_PLUS[<WALimNo>,<Ax>]=<Value>
$P_WORKAREA_CS_MINUS_ENABLE[<WALimNo>,<Ax>]=<Value>
$P_WORKAREA_CS_LIMIT_MINUS[<WALimNo>,<Ax>]=<Value>
...
WALCS<n>
...
WALCS0
```

Значение

$P_WORKAREA_CS_COORD_SYSTEM[<WALimNo>]=<Value>		
Система координат, к которой относится ограничение рабочего поля какой-либо группы		
<WALimNo>:	Группа ограничений рабочего поля	
	Тип:	INT
	Диапазон значений:	0 (группа 1) ... 9 (группа 10)
<VALUE>:	Значение типа INT	
	1	Система координат детали (WCS)
	3	Настраиваемая система нулевой точки (ENS)

$$P_WORKAREA_CS_PLUS_ENABLE[<WALimNo>,<Ax>]=<Value>		
Разрешить ограничение рабочего поля в **положительном** направлении оси для указанной оси канала		
<WALimNo>:	Группа ограничений рабочего поля	
	Тип:	INT
	Диапазон значений:	0 (группа 1) ... 9 (группа 10)
<Ax>:	имя оси канала	
<VALUE>:	Значение типа BOOL	
	0 (FALSE)	Нет разрешения
	1 (TRUE)	Разрешение

$P_WORKAREA_CS_MINUS_ENABLE[<WALimNo>,<Ax>]=<Value>		
Разрешить ограничение рабочего поля в **отрицательном** направлении оси для указанной оси канала		
<WALimNo>:	Группа ограничений рабочего поля	
	Тип:	INT
	Диапазон значений:	0 (группа 1) ... 9 (группа 10)
<Ax>:	имя оси канала	
<VALUE>:	Значение типа BOOL	
	0 (FALSE)	Нет разрешения
	1 (TRUE)	Разрешение

$P_WORKAREA_CS_LIMIT_PLUS[<WALimNo>,<Ax>]=<Value>		
Ограничение рабочего поля в **положительном** направлении указанной оси канала		
<WALimNo>:	Группа ограничений рабочего поля	
	Тип:	INT
	Диапазон значений:	0 (группа 1) ... 9 (группа 10)
<Ax>:	имя оси канала	
<VALUE>:	Значение типа REAL	

$P_WORKAREA_CS_LIMIT_MINUS[<WALimNo>,<Ax>]=<Value>		
Ограничение рабочего поля в **отрицательном** направлении указанной оси канала		
<WALimNo>:	Группа ограничений рабочего поля	
	Тип:	INT
	Диапазон значений:	0 (группа 1) ... 9 (группа 10)
<Ax>:	имя оси канала	
<VALUE>:	Значение типа REAL	

WALCS<n>:	Включение ограничений рабочего поля из группы ограничений рабочего поля	
<n>:	номер группы ограничений рабочего поля	
	Диапазон значений:	1 ... 10

WALCS0:	Выключение активных в канале ограничений рабочего поля

Примечание

Фактически доступное число групп ограничений рабочего поля зависит от конфигурирования (→ см. данные изготовителя станка).

Пример

В канале определено 3 оси: X, Y и Z

Необходимо определить и после активировать группу ограничений рабочего поля Nr. 2, в которой оси в WCS ограничиваются по следующим параметрам:

- Ось X в плюсовом направлении: 10 мм
- Ось X в минусовом направлении: нет ограничения
- Ось Y в плюсовом направлении: 34 мм
- Ось Y в минусовом направлении: -25 мм

- Ось Z в плюсовом направлении: нет ограничения
- Ось Z в минусовом направлении: -600 мм

Программный код	Комментарий
...	
N51 $P_WORKAREA_CS_COORD_SYSTEM[1]=1	; Ограничение рабочего поля группы ограничений рабочего поля 2 действует в WCS.
N60 $P_WORKAREA_CS_PLUS_ENABLE[1,X]=TRUE	
N61 $P_WORKAREA_CS_LIMIT_PLUS[1,X]=10	
N62 $P_WORKAREA_CS_MINUS_ENABLE[1,X]=FALSE	
N70 $P_WORKAREA_CS_PLUS_ENABLE[1,Y]=TRUE	
N73 $P_WORKAREA_CS_LIMIT_PLUS[1,Y]=34	
N72 $P_WORKAREA_CS_MINUS_ENABLE[1,Y]=TRUE	
N73 $P_WORKAREA_CS_LIMIT_MINUS[1,Y]=-25	
N80 $P_WORKAREA_CS_PLUS_ENABLE[1,Z]=FALSE	
N82 $P_WORKAREA_CS_MINUS_ENABLE[1,Z]=TRUE	
N83 $P_WORKAREA_CS_LIMIT_PLUS[1,Z]=-600	
...	
N90 WALCS2	; Включение группы ограничений рабочего поля 2.
...	

Дополнительная информация

Активность

Ограничение рабочего поля с WALCS1 - WALCS10 действует независимо от ограничения рабочего поля с WALIMON. Если активны обе функции, то действует первое ограничение на пути движения оси.

Исходная точка на инструменте

Учет данных инструмента (длина инструмента и радиус инструмента) и тем самым исходной точки на инструменте при контроле ограничения рабочего поля соответствует поведению при ограничении рабочего поля с WALIMON.

3.14.4 Реферирование (G74)

После включения станка все осевые салазки (при использовании инкрементной системы измерения перемещения) должны быть перемещены на свою референтную метку. Лишь после этого могут программироваться движения перемещения.

С помощью G74 можно выполнить реферирование в программе ЧПУ.

Синтаксис

```
G74 X1=0 Y1=0 Z1=0 A1=0 ... ; Программирование в своем кадре ЧПУ
```

Объяснение

G74:	Вызов G-команд: реферирование
X1=0 Y1=0 Z1=0 … :	Указанный адрес оси станка X1, Y1, Z1 ... для **линейных осей**перемещается в референтную точку
A1=0 B1=0 C1=0 … :	Указанный адрес оси станка A1, B1, C1 ... для **круговых осей**перемещается в референтную точку

Примечание

Перед реферированием нельзя программировать трансформацию для оси, которая должна с помощью G74 должна быть перемещена на референтную метку.

Трансформация выключается командой TRAFOOF.

Пример

При смене измерительной системы осуществляется переход к референтной точке и устанавливается нулевая точка детали.

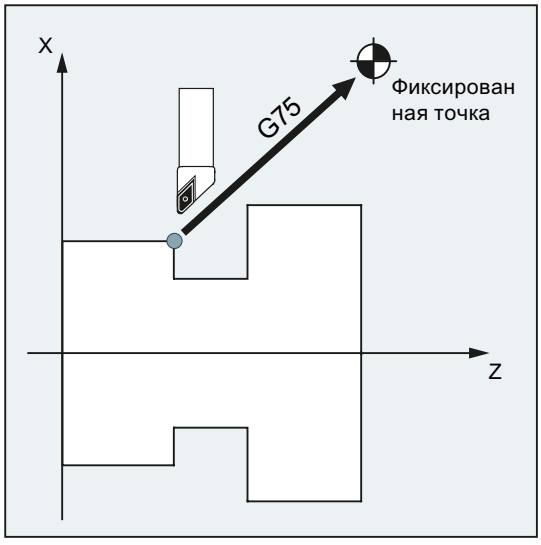
Программный код	Комментарий
N10 SPOS=0	; Шпиндель в режиме управления по положению
N20 G74 X1=0 Y1=0 Z1=0 C1=0	; Реферирование для линейных и круговых осей
N30 G54	; Смещение нулевой точки
N40 L47	; Программа обработки резаньем
N50 M30	; конец программы

3.14.5 Движение к фиксированной точке (G75)

С помощью действующей не модально команды G75 оси по отдельности и независимо друг от друга могут перемещаться на базисные точки в станочной области, к примеру, на точки смены инструмента, точки загрузки, точки автоматической смены палет и т.д.

Фиксированные точки это позиции в системе координат станка, сохраненные в машинных данных (MD30600 $MA_FIX_POINT_POS[n]). На ось может быть определено макс. 4 фиксированные точки.

Подвод к фиксированным точкам может быть осуществлен из любой программы ЧПУ, независимо от актуальной позиции инструмента или детали. Перед движением осей выполняется внутренняя остановка предварительной обработки.



Условия

Для перехода к фиксированным точка с G75 должны быть выполнены следующие условия:

- Координаты фиксированных точек должны быть точно определены и зафиксированы в машинных данных.
- Фиксированные точки должны находится в пределах действующего диапазона перемещения (→ Соблюдать границы программных конечных выключателей!)
- Перемещаемые оси должны быть реферированы.
- Не должно быть активной коррекции радиуса инструмента.
- Не должно быть активной кинематической трансформации.
- Перемещаемые оси не должны участвовать в активной трансформации.
- Ни одна из перемещаемых осей не может быть ведомой осью активного соединения.
- Ни одна из перемещаемых осей не может быть осью структуры Gantry.
- Компилируемые циклы не могут подключить компонент движения.

Синтаксис

```
G75 <имя оси><позиция оси> ... FP=<n>
```

Объяснение

G75:	Подвод к фиксированной точке
<имя оси>:	Имя оси станка, которые должны быть перемещены к фиксированной точке Допускаются все идентификаторы осей.

<позиция оси>:	Значение позиции не важно. Поэтому, как правило, указывается значение "0".		
FP=:	Фиксированная точка, к которой должен быть выполнен подвод		
	<n>:	Номер фиксированной точки	
		Диапазон значений:	1, 2, 3, 4
	Примечание: Если FP=<n> или номер фиксированной точки не запрограммированы или если запрограммировано FP=0, то это интерпретируется как FP=1 и выполняется подвод к фиксированной точке 1.		

Примечание

В одном кадре G75 может быть запрограммировано и несколько осей. В этом случае оси перемещаются к указанной фиксированной точке одновременно.

Примечание

Значение адреса FP не должно превышать число определенных фиксированных точек для каждой запрограммированной оси (MD30610 $MA_NUM_FIX_POINT_POS).

Пример

Для смены инструмента оси X (= AX1) и Z (= AX3) должны быть перемещены на фиксированную позицию осей станка 1 с X = 151,6 и Z = -17,3.

Машинные данные:

- MD30600 $MA_FIX_POINT_POS[AX1,0] = 151.6
- MD30600 $MA_FIX_POINT[AX3,0] = 17.3

Программа ЧПУ:

Программный код	Комментарий
...	
N100 G55	; Активация настраиваемого смещения нулевой точки.
N110 X10 Y30 Z40	; Подвод к позиции в WCS.
N120 G75 X0 Z0 FP=1 M0	; Ось X движется на 151,6
	; и ось Z движется на 17,3 (в MCS).
	; Каждая ось движется со своей макс. скоростью.
	; В этом кадре не должно быть активных дополнительных движений.
	; Что и после достижения конечных позиций
	; дополнительные движения также не выполнялись,
	; здесь вставлен Стоп.
N130 X10 Y30 Z40	; Снова выполняется подвод к позиции из N110.
	; Смещение нулевой точки снова активно.
...	

Примечание

Если функция "Управление инструментом с магазинами" активна, то вспомогательной функции T... или M... (обычно M6) не достаточно для инициирования блокировки смены кадра на конце движения G75.

Причина: При установке "Управление инструментом с магазинами активно" вспомогательные функции для смены инструмента не выводятся на PLC.

Дополнительная информация

G75

Оси перемещаются как оси станка ускоренным ходом. Выполняется внутренняя эмуляция движения через функции "SUPA" (подавление всех фреймов) и "G0 RTLIOF" (движение ускоренного хода с интерполяцией отдельной оси).

Если условия для "RTLIOF" (интерполяция отдельной оси) не выполнены, то подвод к фиксированной точке выполняется как траектория.

При достижении фиксированной точки оси останавливаются в пределах окна допуска "Точный останов точный".

Параметрируемая динамика для G75

Для движений позиционирования на фиксированные позиции (G75) через следующие машинные данные можно установить требуемый динамический режим:

MD18960 $MN_POS_DYN_MODE (тип динамики позиционирующей оси)

Осевые дополнительные движения

Следующие осевые доп. движения учитываются на момент интерпретации кадра G75:

- внешнее смещение нулевой точки
- DRF
- смещение синхронизации ($AA_OFF)

После этого доп. движения осей не могут изменяться до достижения конца движения перемещения кадром G75.

Доп. движения после интерпретации кадра G75 ведут к соответствующему смещению фиксированной точки подвода.

Следующие дополнительные движения не учитываются независимо от момента интерпретации и приводят к соответствующему смещению заданного конечного положения:

- Коррекция инструмента Online
- Дополнительные движения из компилируемых циклов в BKS как MCS

Активные фреймы

Все активные фреймы игнорируются. Перемещение выполняется в системе координат станка.

Ограничение рабочего поля в WCS/ENS

Ограничение рабочего поля, спец. для системы координат (WALCS0 ... WALCS10) не действует в кадре с G75. Заданная точка контролируется как стартовая точка следующего кадра.

Движения осей/шпинделей с POSA/SPOSA

Если запрограммированные оси/шпиндели перемещались прежде с POSA или SPOSA, то перед подводом к фиксированной точке эти движения сначала доводятся до конца.

Функции шпинделя в кадре G75

Если шпиндель исключен из "Подвода к фиксированной точке", то в кадре G75 могут быть дополнительно запрограммированы функции шпинделя (например, позиционирование с SPOS/SPOSA).

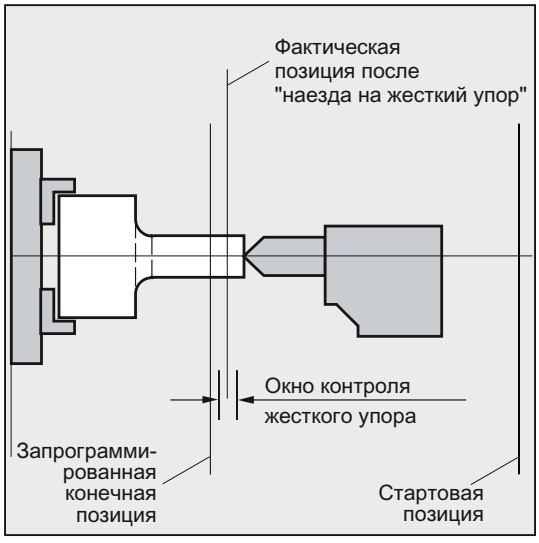
Оси модуло

У осей модуло подвод к фиксированной точке выполняется по кратчайшему пути.

3.14.6 Наезд на жесткий упор (FXS, FXST, FXSW)

Функция

С помощью функции "Наезд на жесткий упор" можно создать определенные усилия для зажима деталей, которые необходимы, к примеру, для задних бабок, пинолей и захватов. Кроме этого с помощью функции можно переходить к механическим референтным точкам.



При достаточно уменьшенном моменте возможны и простые процессы измерения без обязательного подключения щупа. Функция "Наезд на жесткий упор" может использоваться для осей и перемещаемых в качестве осей шпинделей.

Синтаксис

```
FXS[<ось>]=...
FXST[<ось>]=...
FXSW[<ось>]=...
FXS[<ось>]=... FXST[<ось>]=...
FXS[<ось>]=... FXST[<ось>]=... FXSW[<ось>]=...
```

Объяснение

FXS:	Команда для включения и выключения функции "Наезд на жесткий упор"
	FXS[<ось>]=1: Включить функцию
	FXS=[<ось>]=0: Выключить функцию
FXST:	Опционная команда для установки зажимного момента
	Данные в % от максимального момента привода.
FXSW:	Опционная команда для установки ширины окна для контроля жесткого упора
	Данные в мм, дюймах или градусах.
<ось>:	Имена осей станка
	Программируются оси станка (X1, Y1, Z1 и т.д.).

Примечание

Команды FXS, FXST и FXSW действуют модально.

Программирование FXST и FXSW является опционным: если данные отсутствуют, то действует последнее запрограммированное значение или установленное в соответствующих машинных данных значение.

Активировать наезд на жесткий упор: FXS[<ось>] = 1

Движение к заданной точке может быть описано как движение траекторных или позиционирующих осей. Для позиционирующих осей функция возможна и за границы кадра.

Наезд на жесткий упор может осуществляться и для нескольких осей одновременно и параллельно движению других осей. Жесткий упор должен находиться между стартовой позицией и заданным конечным положением.

ВНИМАНИЕ

Опасность столкновений

Сразу же после активации функции "Наезд на жесткий упор" для оси/шпинделя более нельзя запрограммировать новую позицию для этой оси.

Перед выбором функции шпиндели должны быть переведены в режим управления положением.

Пример:

Программный код	Комментарий
`X250 Y100 F100 FXS[X1]=1 FXST[X1]=12.3 FXSW[X1]=2`	; Ось `X1` движется с подачей `F100` (не обязательные данные) на заданную позицию `X=250` мм. Зажимной момент составляет 12.3% от максимального движущего момента, контроль осуществляется в окне шириной 2 мм.
...	

Деактивировать наезд на жесткий упор: FXS[<ось>] = 0

Выключение функции вызывает остановку предварительной обработки.

В кадре с `FXS[`<ось>`]`=0 могут и должны стоять движения перемещения.

ВНИМАНИЕ

Опасность столкновений

Движения перемещения на позицию отвода должны осуществляться от жесткого упора, в ином случае возможны повреждения упора или станка.

Смена кадра осуществляется после достижения позиции отвода. Если позиция отвода не указана, то смена кадра осуществляется сразу же после отключения ограничения моментов.

Пример:

Программный код	Комментарий
`X200 Y400 G01 G94 F2000 FXS[X1]=0`	; Ось `X1` отводится от жесткого упора на позицию `X=200` мм. Все другие данные являются опционными.
...	

Зажимной момент (FXST) и окно контроля (FXSW)

Запрограммированное ограничение моментов `FXST` действует с начала кадра, т.е. и наезд на упор осуществляется с уменьшенным моментом. `FXST` и `FXSW` могут быть запрограммированы или изменены в любой момент времени в программе обработки деталей. Изменения начинают действовать перед движениями перемещения, стоящими в том же кадре.

ВНИМАНИЕ
Опасность столкновений Если программируется новое окно контроля жесткого упора, то изменяется не только ширина окна, но и исходная точка для центра окна, если ось до этого двигалась. Фактическая позиция оси станка при изменении окна является новым центром окна. Окно должно быть выбрано таким образом, чтобы только наезд на упор приводил к срабатыванию контроля жесткого упора.

Дополнительная информация

Рампа нарастания

Через машинные данные можно определить рампу нарастания для новой границы момента, чтобы предотвратить скачкообразную установку границы момента (например, при зажиме детали пинолью).

Подавление ошибок

В приложениях ошибка упора может быть подавлена из программы обработки деталей, при этом в машинных данных создается маска ошибки и новая настройка MD активируется с помощью NEW_CONF.

Активация

Команды для наезда на жесткий упор могут вызываться из синхронных действий/ технологических циклов. Активация может осуществляться и без движения, ограничение момента осуществляется сразу же. Как только ось начинает двигаться со стороны заданного значения, выполняется контроль упора.

Активация из синхронных действий

Пример:

Если ожидаемое событие ($R1) наступает, а наезд на жесткий упор еще не выполняется, то необходимо активировать `FXS` для оси Y. Момент должен составлять 10 % номинального момента. Для ширины окна контроля действует значение предварительной установки.

Программный код

```
N10 IDS=1 WHENEVER (($R1=1) AND ($AA_FXS[Y]==0)) DO $R1=0 FXS[Y]=1
FXST[Y]=10
```

Обычная программа обработки деталей должна обеспечить установку $R1 к желаемому моменту времени.

Деактивация из синхронных действий

Пример:

Если имеется ожидаемое событие ($R3) и существует состояние "Наезд на упор" (системная переменная $AA_FXS), то FXS должна быть отключена.

Программный код

```
IDS=4 WHENEVER (($R3==1) AND ($AA_FXS[Y]==1)) DO FXS[Y]=0
FA[Y]=1000 POS[Y]=0
```

Жесткий упор был достигнут

После достижения жесткого упора:

- остаточный путь удаляется, и отслеживается заданное значение положения.
- Движущий момент возрастает до запрограммированного предельного значения FXSW и после этого остается постоянным.
- Контроль жесткого упора активируется в пределах указанной ширины окна.

Граничные условия

- Измерение со стиранием остатка пути
  "Измерение со стиранием остаточного пути" (команда MEAS) и "Наезд на жесткий упор" не могут одновременно программироваться в одном кадре.
  Исключение: Одна функция воздействует на траекторную ось, а другая – на позиционирующую ось, или обе воздействуют на позиционирующие оси.
- Контроль контура
  При активной функции "Наезд на жесткий упор" контроль контура не осуществляется.
- Позиционирующие оси
  При "Наезде на жесткий упор" с позиционирующими осями смена кадров осуществляется независимо от движения наезда на жесткий упор.
- Оси Link и оси контейнера
  Наезд на жесткий упор допускается и для осей Link и осей контейнера.
  Состояние зависимой оси станка сохраняется и после вращений контейнера. Это же относится и к модальному ограничению моментов с FOCON.
- Наезд на жесткий упор **невозможен**:
  - для осей Gantry
  - Для конкурирующих позиционирующих осей, управляемых исключительно с PLC (выбор FXS должен осуществляться из программы ЧПУ).
- Если граница моментов слишком уменьшается, то ось более не может следовать установке заданного значения, регулятор положения переходит в ограничение и увеличивается погрешность контура. В этом рабочем состоянии при увеличении границы моментов могут возникнуть толчкообразные движения. Для обеспечения следования оси необходимо контролировать, чтобы погрешность контура не была бы больше, чем при неограниченном моменте.

3.14.7 Время ожидания (G4)

Команда G4 в кадре программирует время (задержку), отсчитываемое с момента обработки кадра в главном ходе. Смена кадров происходит в следующем кадре, после полного истечения времени.

Примечание

G4 прерывает режим управления траекторией.

Синтаксис

```
G4 F<Time>
G4 S<NumSpi>
G4 S<n> = <NumSpi>
```

Значение

G4:	Активировать время ожидания	
	Автономно в кадре:	да
F<Time>:	По адресу F задается задержка <Time> в секундах.	
S<NumSpi>:	По адресу S задается задержка в оборотах шпинделя <NumSpi> относительно текущего главного шпинделя.	
S<n>=NumSpi>:	По адресу S задается задержка в оборотах шпинделя <NumSpi> относительно шпинделя с расширением адреса <n>.	

Примечание

Используемые в кадре задержки G4 для указания времени адреса F и S не влияют на подачу F... и частоту вращения шпинделя S... программы.

Граничные условия

Синхронные действия

Два синхронных действия запрограммированы в программе таким образом, чтобы следующий кадр выполнялся с задержкой относительно кадра действия, в котором выполняются синхронные действия. Одно синхронное действие представляет собой модальное синхронное действие. Другое синхронное действие представляет собой покадровое синхронное действие. Если нужно чтобы покадровое синхронное действие влияло на модальное синхронное действие, например, разрешало обработку через UNLOCK, то в качестве действующей задержки должны быть предусмотрены, **по меньшей мере, два такта интерполятора** например, G4 F<такт интерполятора * 2>.

Действующая задержка зависит от настроенной машинной даты MD10280 $MN_PROG_FUNCTION_MASK, бит 4 = <значение>

Величина	Значение
0	Действующая задержка равна запрограммированной задержке
1	Действующая задержка равна запрограммированной задержке, округленной до ближайшего (в сторону увеличения) кратного такта интерполятора (MD10071 $MN_IPO_CYCLE_TIME)

Пример программы:

- MD10071 $MN_IPO_CYCLE_TIME == 8 мс
- MD10280 $MN_PROG_FUNCTION_MASK, бит 4 = 1

Программный код	Комментарий
`N10 WHEN TRUE DO LOCK(1)`	; покадровый `SynAct: LOCK`
	; модальный `SynAct. ID=1`
`N20 G4 F2`	; кадр действия для `SynAct` из `N10`
`N30 WHEN TRUE DO UNLOCK(1)`	; покадровый `SynAct: UNLOCK`
	; модальный `SynAct. ID=1`
`N40 ID=1 WHENEVER TRUE DO $R0=1 RDISABLE`	; модальный `SynAct ID=1`
	`;` R-параметр `R0=1`
	; установка блокировки загрузки
`N50 G4 F0.012`	; кадр действия для `SynAct` из `N40` и `N50`
	; см. ниже абзац «Описание»
`N60 G4 F10`	

Описание

Цель заключается в том, чтобы покадровое синхронное действие из N30 отменяло активную блокировку (LOCK) модального синхронного действия с ID=1 из N40 и, тем самым, в N50 записывался R-параметр и включалась блокировка загрузки. Эта цель может быть достигнута только в том случае, если длительность действующей задержки составляет, по меньшей мере, два такта интерполятора.

Действующая задержка складывается из запрограммированной задержки, такта интерполятора и значения параметра MD10280 $MN_PROG_FUNCTION_MASK, бит 4. Чтобы длительность действующей задержки составляла, по меньшей мере, два такта интерполятора, необходимо запрограммировать следующую задержку:

- бит 4 == 0: запрограммированная задержка ≥ 2 * такта интерполятора
- бит 4 == 1: запрограммированная задержка ≥ 1,5 * такта интерполятора

Если действующая задержка меньше двух тактов интерполятора, запись R-параметра и блокировка загрузки будут выполнены только в кадре N60.

Пример

Программный код	Комментарий
`N10 G1 F200 Z-5 S300 M3`	; подача `F`, скорость шпинделя `S`

Программный код	Комментарий
`N20 G4 F3`	; Время ожидания: 3с
`N30 X40 Y10`	
`N40 G4 S30`	; Ожидать 30 оборотов шпинделя (соответствует при `S=300` об/мин и процентовке скорости `100 %: t = 0,1` мин).
`N50 X...`	; Запрограммированная в `N10` подача и скорость шпинделя продолжают действовать.

3.14.8 Внутренняя остановка предварительной обработки

Функция

При обращении к данным состояния станка ($A...) СЧПУ осуществляет внутреннюю остановку предварительной обработки. Следующий кадр выполняется только тогда, когда все подготовленные и сохраненные прежде кадры полностью выполнены. Предыдущий кадр останавливается с точным остановом (как `G9`).

Пример

Программный код	Комментарий
...	
`N40 POSA[X]=100`	
`N50 IF $AA_IM[X]==R100 GOTOF MARKE1`	; Обращение к данным состояния станка (`$A...`), СЧПУ осуществляет внутреннюю остановку предварительной обработки.
`N60 G0 Y100`	
`N70 WAITP(X)`	
`N80 MARKE1:`	
...	

3.15 Прочая информация

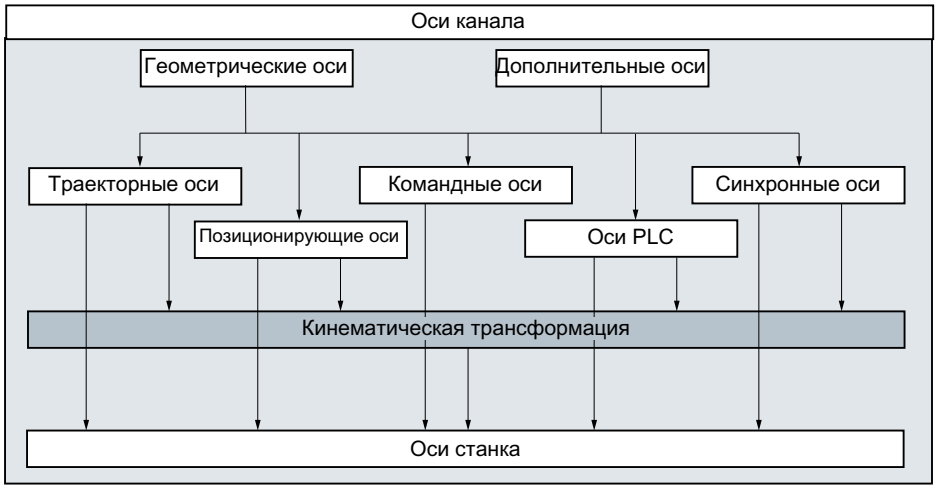
3.15.1 Оси

Типы осей

В рамках программирования различают следующие типы осей:

- Главные оси / гео-оси
- Дополнительные оси
- Главный шпиндель, мастер-шпиндель

- Оси станка
- Оси канала
- Траекторные оси
- Позиционирующие оси
- Синхронные оси
- Командные оси
- Оси PLC / конкурирующие позиционирующие оси
- Link-оси
- Оси Lead-Link



3.15.1.1 Главные оси/геометрические оси

Главные оси определяют прямоугольную, правовращающуюся систему координат. В этой системе координат программируются движения инструмента.

В технике ЧПУ главные оси обозначаются как геометрические оси. В данном руководстве по программированию также будет использоваться это понятие.

Переключаемые гео-оси

С помощью функции "Переключаемые геометрические оси" (см. описание функций "Расширенное программирование") можно изменять сконфигурированное через машинные данные соединение геометрических осей из программы обработки деталей. При этом одна из определенных в качестве синхронной дополнительной оси ось канала может заменять любую геометрическую ось.

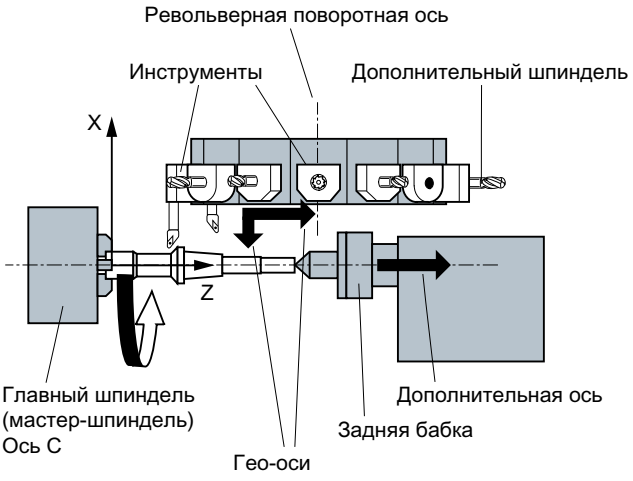
Идентификатор оси

Имя/идентификатор гео-оси можно задавать через следующие машинные данные:

MD20060 $MC_AXCONF_GEOAX_NAME_TAB (имя гео-оси в канале)

Стандартное обозначение на токарных станках:

1. Гео-ось: X
2. Гео-ось: Z



Стандартное обозначение на фрезерных станках:

1. Гео-ось: X
2. Гео-ось: Y
3. Гео-ось: Z

Дополнительная информация

Для программирования фреймов и геометрии детали (контура) используются максимум три геометрические оси.

Идентификаторы для геометрических осей и осей канала могут быть одинаковыми, если возможно отображение.

Имена гео-осей и осей канала могут быть одинаковыми во всех каналах. Таким образом, любая программа может обрабатываться в любом канале.

3.15.1.2 Дополнительные оси

В отличие от геометрических осей, для дополнительных осей не определено геометрической связи между осями.

Типичными дополнительными осями являются:

- оси револьверной головки
- оси поворотного стола
- оси качающейся головки
- оси загрузчика

Идентификатор оси

У токарного станка с поворотным магазином, к примеру,:

- Позиция револьверной головки U
- Задняя бабка V

Пример программирования

Программный код	Комментарий
`N10 G1 X100 Y20 Z30 A40 F300`	; Движения траекторных осей.
`N20 POS[U]=10POS[X]=20 FA[U]=200 FA[X]=350`	; Движения позиционирующих осей.
`N30 G1 X500 Y80 POS[U]=150FA[U]=300 F550`	; Траекторная и позиционирующая ось.
`N40 G74 X1=0 Z1=0`	; Подвод в референтную точку.

3.15.1.3 Главный шпиндель, мастер-шпиндель

Какой из шпинделей является главным, определяется кинематикой станка. Этот шпиндель, как правило, через машинные данные назначается мастер-шпинделем.

Это соответствие может быть изменено программной командой `SETMS` (<номер шпинделя>). С помощью `SETMS` без указания номера шпинделя можно вернуться к определенному в машинных данных мастер-шпинделю.

Для мастер-шпинделя действуют специальные функции, к примеру, резьбонарезание.

Идентификатор шпинделя

S или S0

3.15.1.4 Оси станка

Оси станка это физически имеющиеся на станке оси.

Запрограммированное движение траекторной или дополнительной оси может через активную в канале трансформацию (TRANSMIT, TRACYL или TRAORI) действовать на несколько осей станка.

Оси станка запрашиваются прямо в программе только в особых случаях (например, при реферировании или движении в фиксированную точку).

Идентификатор оси

Имя/идентификатор оси станка можно задавать через следующие спец. для ЧПУ машинные данные:

MD10000 $MN_AXCONF_MACHAX_NAME_TAB (имя оси станка)

Установка по умолчанию: X1, Y1, Z1, A1, B1, C1, U1, V1

Кроме того, оси станка имеют фиксированные идентификаторы осей, которые всегда можно использовать независимо от имен, установленных в машинных данных:

AX1, AX2, ..., AX<n>

3.15.1.5 Оси канала

Осями канала называются все геометрические, дополнительные оси и оси станка, назначенные каналу.

Идентификатор оси

Спец. для канала имя/идентификатор одной из геометрических и дополнительных осей можно задавать через следующие машинные данные:

MD20080 $MC_AXCONF_CHANAX_NAME_TAB (имя оси канала)

Установка по умолчанию: X, Y, Z, A, B, C, U, V

Сопоставление, на какую ось станка будет отображаться та или иная гео-ось или дополнительная ось в канале, задается в следующих машинных данных:

MD20070 $MC_AXCONF_MACHAX_USED (используемые оси станка)

3.15.1.6 Траекторные оси

Траекторные оси описывают ход траектории и тем самым движение инструмента в пространстве.

Запрограммированная подача действует вдоль этой траектории. Участвующие в этой траектории оси достигают своих позиций одновременно. Как правило, это геометрические оси.

Какие оси являются траекторными осями и тем самым определяют скорость, задается через предварительные установки.

В программе ЧПУ траекторные оси могут указываться с помощью `FGROUP`.

Дополнительную информацию по `FGROUP` см. "Подача (G93, G94, G95, F, FGROUP, FL, FGREF) (Страница 110)".

3.15.1.7 Позиционирующие оси

Позиционирующие оси интерполируются раздельно, т. е. каждая позиционирующая ось имеет свой осевой интерполятор и свою подачу. Позиционирующие оси не интерполируют с траекторными осями.

Позиционирующие оси перемещаются из программы ЧПУ или с PLC. Если одна ось одновременно должна перемещаться и программой ЧПУ, и PLC, то появляется сообщение об ошибке.

Типичными позиционирующими осями являются:

- загрузчик для транспортировки детали
- загрузчик для отгрузки детали
- инструментальный магазин/револьвер

Типы

Различаются позиционирующие оси с синхронизацией к концу кадра или через несколько кадров.

Оси POS

Смена кадра осуществляется к концу кадра, если все запрограммированные в этом кадре траекторные и позиционирующие оси достигли своей запрограммированной конечной точки.

Оси POSA

Движения этих позиционирующих осей могут осуществляться через несколько кадров.

Оси POSP

Движение этих позиционирующих осей для подвода к конечной позиции осуществляется поэтапно.

Примечание

Позиционирующие оси становятся синхронными осями, если они перемещаются без особой идентификации POS/POSA.

Режим управления траекторией (G64) для траекторных осей возможен только тогда, когда позиционирующие оси (POS) достигли своей конечной позиции перед траекторными осями.

Траекторные оси, программируемые с помощью `POS`/`POSA`, для этого кадра изымаются из соединения траекторных осей.

Дополнительную информацию по `POS`, `POSA` и `POSP` см. "Перемещение позиционирующих осей (POS, POSA, POSP, FA, WAITP, WAITMC) (Страница 118)".

3.15.1.8 Синхронные оси

Синхронные оси движутся синхронно с ходом траектории от начальной к запрограммированной конечной позиции.

Запрограммированная в `F` подача действует для всех запрограммированных в кадре траекторных осей, но не для синхронных осей. Для хода синхронных осей требуется то же время, что и для траекторных осей.

Синхронной осью может быть, к примеру, круговая ось, перемещаемая синхронно с траекторной интерполяцией.

3.15.1.9 Командные оси

Командные оси запускаются из синхронных действий на основе события (команды). Они могут позиционироваться, запускаться и останавливаться полностью асинхронно с программой обработки деталей. Ось не может приводиться в движение одновременно из программы обработки деталей и из синхронных действий.

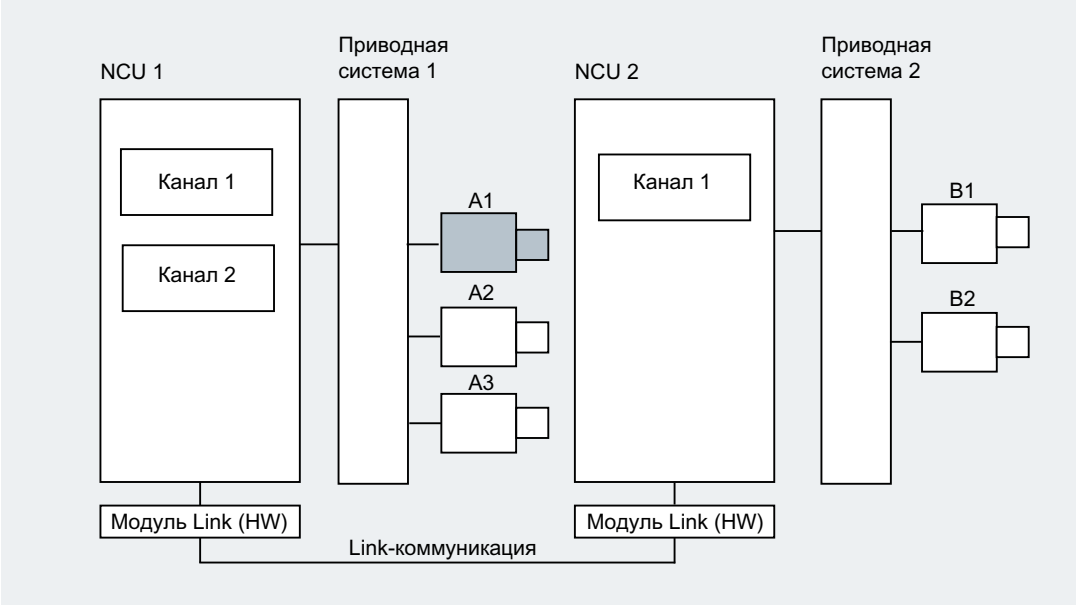
Командные оси интерполируются раздельно, т.е. каждая командная ось имеет свой осевой интерполятор и свою подачу.

3.15.1.10 Оси PLC

Оси PLC перемещаются с PLC через специальные функциональные блоки в главной программе и могут двигаться асинхронно со всеми прочими осями. Движения перемещения запускаются траекторными и синхронными движениями.

3.15.1.11 Оси Link

Link-оси это оси, физически подключенные к другому NCU и подчиняющиеся его управлению по положению. Link-оси могут быть динамически согласованы с каналами **другого** NCU. С точки зрения определенного NCU, Link-оси это не локальные оси.



Для динамического изменения согласования с NCU служит концепция **осевого контейнера**. Переход осей с `GET` и `RELEASE` из программы обработки детали для Link-осей **не** доступен.

Дополнительная информация

Условия

- Участвующие NCU1 и NCU2 должны быть соединены через Link-модуль с быстрой Link-коммуникацией.
- Ось должна быть соответственно сконфигурирована через машинные данные.
- Необходимо наличие опции "Link-ось".

Описание

Управление по положению осуществляется на NCU, на котором ось физически соединена с приводом. Там находится соответствующий осевой интерфейс VDI.

Заданные значения положения создаются для Link-осей на другом NCU и коммуникация осуществляется через NCU-Link.

Link-коммуникация должна отвечать за синхронизацию между интерполяторами и регулятором положения или интерфейсом PLC. Вычисленные интерполяторами заданные значения должны быть переданы на контур управления положением на основном NCU, или фактические значения должны быть снова переданы обратно.

Осевой контейнер

Осевой контейнер — это структура данных кольцевого буфера, в которой осуществляется согласование локальных осей и/или Link-осей с каналами. Записи в кольцевом буфере могут иметь **циклическое смещение**.

Конфигурация Link-осей допускает в логическом образе осей станка наряду с прямой ссылкой на локальные оси или Link-оси ссылку на осевой контейнер. Такая ссылка состоит из:

- номера контейнера **и**
- слота (место в кольцевом буфере внутри соответствующего контейнера)

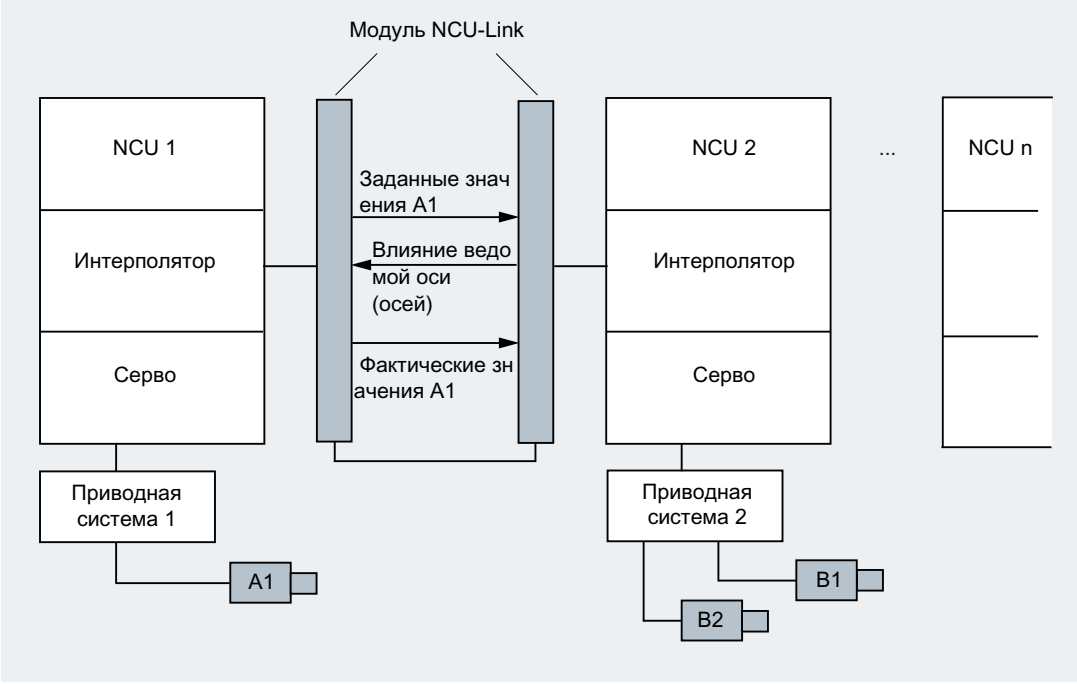
В качестве элемента на месте в кольцевом буфере стоит:

- локальная ось **или**
- Link-ось

Элементы в осевом контейнере содержат локальные оси станка или Link-оси с точки зрения отдельного NCU. Элементы в логическом образе осей станка (MD10002 $MN_AXCONF_LOGIC_MACHAX_TAB) отдельного NCU являются фиксированными.

3.15.1.12 Оси Lead-Link

Lead-Link-ось это ось, которая интерполируется NCU и используется одним или несколькими другими NCU в качестве ведущей оси для управления ведомыми осями.



Осевая ошибка регулятора положения передается на все другие NCU, которые через Lead-Link-ось имеют отношение к затронутой оси.

Зависимые от Lead-Link-оси NCU могут использовать следующие соединения с Lead-Link-осью:

- главное значение(заданное, фактическое, симулированное главное значение)
- буксировка
- тангенциальное слежение
- электронный редуктор (ELG)
- синхронный шпиндель

Программирование

Главный NCU:

Только NCU, с которым физически согласована ось главного значения, может программировать движения перемещения для этой оси. Исходя из этого программирование не должно учитывать особенностей.

NCU ведомых осей:

Программирование на NCU ведомых осей не может содержать команд перемещения для Lead-Link-оси (ось главного значения). Нарушения этого правила вызывают ошибку.

Обращение к Lead-Link-оси осуществляется через идентификатор осей канала обычным способом. Доступ к состояниям Lead-Link-оси возможен через выбранные системные переменные.

Дополнительная информация

Условия

- Участвующие NCU1 до NCU<n> (<n> макс. 8) должны быть соединены через Link-модуль с быстрой Link-коммуникацией.
  Литература:
  Справочник по оборудованию "Проектирование NCU"
- Ось должна быть соответственно сконфигурирована через машинные данные.
- Необходимо наличие опции "Link-ось".
- Для всех участвующих NCU должен быть сконфигурирован идентичный такт интерполятора.

Ограничения

- Ведущая ось в качестве Lead-Link-оси не может быть Link-осью, т. е. перемещаться другим NCU как основным NCU.
- Ведущая ось в качестве Lead-Link-оси не может быть осью контейнера, т. е. осью, к которой попеременно обращаются различные NCU.
- Lead-Link-ось не может быть запрограммированной ведущей осью соединения Gantry.
- Соединения с Lead-Link-осями не могут иметь многоступенчатого последовательного включения (каскадирование).
- Переход оси возможен только внутри основного NCU Lead-Link-оси.

Системные переменные

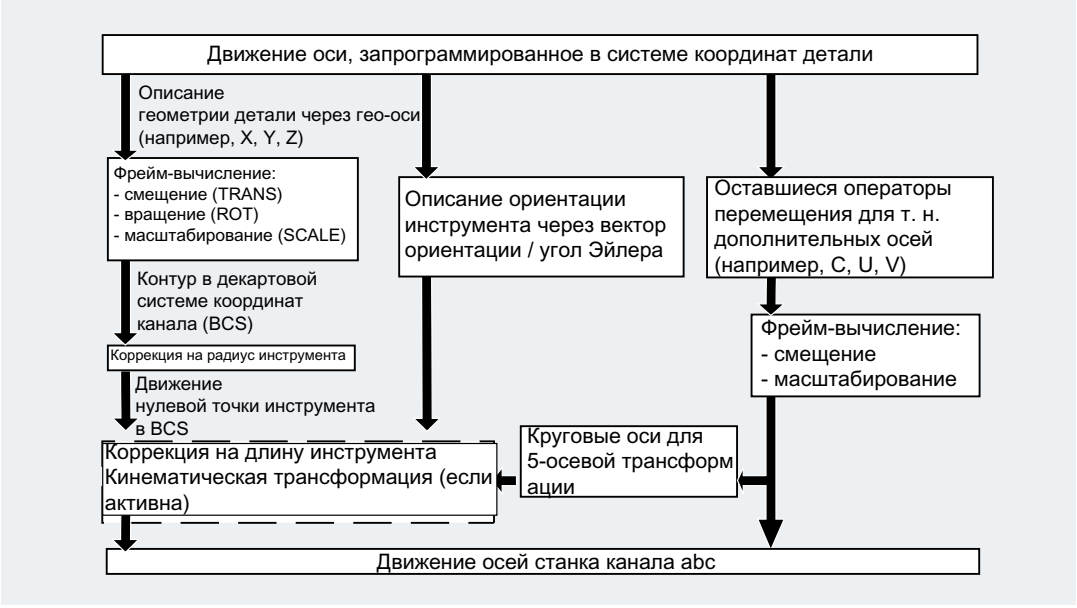
Следующие системные переменные могут использоваться с идентификатором осей канала Lead-Link-оси:

Системная переменная	Объяснение
$AA_LEAD_SP	Симулированное главное значение – позиция
$AA_LEAD_SV	Симулированное главное значение – скорость

Если эти системные переменные актуализируются через NCU ведущей оси, то новые значения передаются и на NCU, которые хотят перемещать ведомые оси в зависимости от этой ведущей оси.

3.15.2 От команды движения до движения станка

Связь между запрограммированными движениями осей (командами движения) и следующими из них движениями станка должен пояснить рисунок ниже:

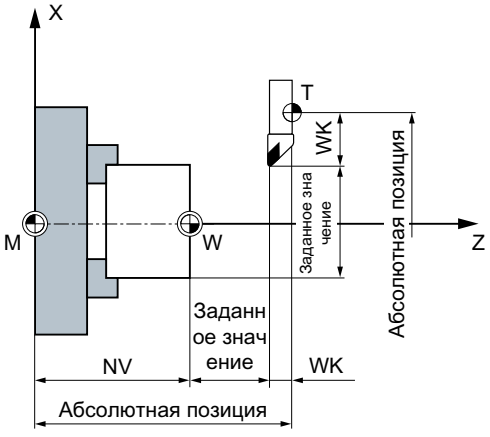


3.15.3 Вычисление хода

Вычисление хода получает участок перемещения, который должен быть пройден в одном кадре, с учетом всех смещений и коррекций.

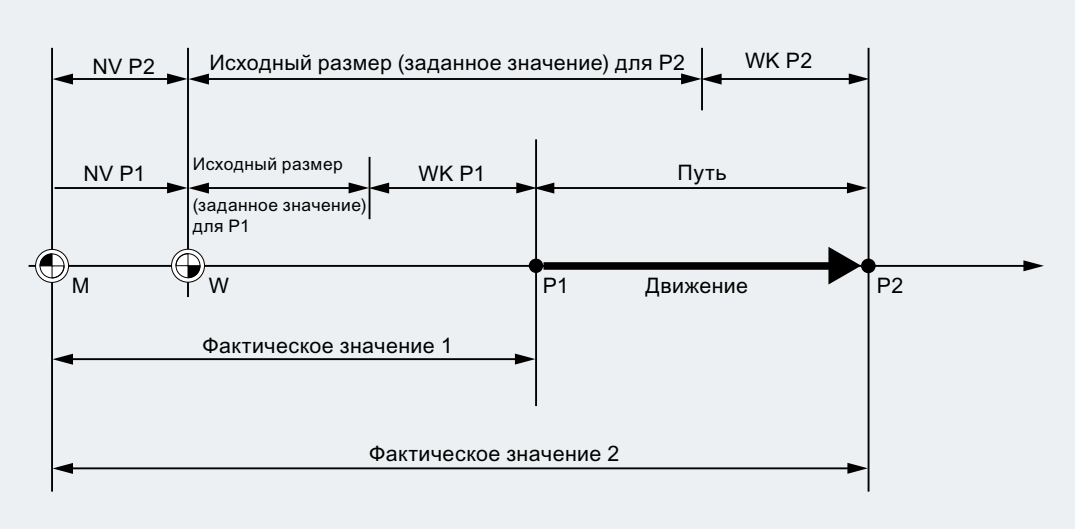
Общее правило:

Ход =
заданное значение – фактическое значение + смещение нулевой точки (ZO) + коррекция инструмента (WK)



Если в новом программном кадре программируется новое смещение нулевой точки и новая коррекция инструмента, то:

- при вводе исходного размера:
  Ход = (исходный размер P2 - исходный размер P1) + (NV P2 - NV P1) + (WK P2 - WK P1).
- при вводе составного размера:
  Ход = составной размер + (NV P2 - NV P1) + (WK P2 - WK P1).



3.15.4 Адреса

Постоянные адреса

Эти адреса установлены фиксировано, т. е. символы адреса изменить невозможно.

Перечень представлен в таблице "Постоянные адреса (Страница 1290)".

Настраиваемые адреса

Этим адресам изготовитель станка через машинные данные может присвоить другое имя.

Примечание

Настраиваемые адреса должны быть однозначными в СЧПУ, т. е. нельзя использовать одно и то же имя адреса для адресов разного типа (значения осей и конечные точки, ориентация инструмента, параметры интерполяции, ...).

Перечень представлен в таблице "Устанавливаемые адреса (Страница 1296)".

Адреса, действующие модально/покадрово

Модально действующие адреса сохраняют свою значимость с запрограммированным значением до тех пор (во всех последующих кадрах), пока по тому же адресу не будет запрограммировано новое значение.

Действующие покадрово адреса сохраняют свою значимость только в том кадре, в котором они были запрограммированы.

Пример:

Программный код	Комментарий
`N10 G01 F500 X10`	
`N20 X10`	; Подача `F` из `N10` действует до ввода новой подачи.

Адреса с осевым расширением

У адресов с осевым расширением имя оси стоит в квадратных скобках после адреса, который устанавливает соответствие осям.

Пример:

Программный код	Комментарий
`FA[U]=400`	; Специфическая подача для оси `U`.

См. также таблицу "Постоянные адреса (Страница 1290)".

Расширенное написание адреса

Расширенное написание адресов позволяет систематизировать большее количество осей и шпинделей.

Расширенный адрес состоит из числового расширения и присвоенного посредством символа "=" арифметического выражения. Цифровое расширение имеет одну или две позиции и всегда является положительным.

Расширенное написание адреса допускается только для следующих простых адресов:

Адрес	Объяснение
X,Y,Z,...	Адреса осей
I, J, K	Параметры интерполяции
S	Число оборотов шпинделя
SPOS, SPOSA	Позиция шпинделя
M	Дополнительные функции
H	Вспомогательные функции
T	Номер инструмента
F	Подача

Примеры:

Программный код	Комментарий
`X7`	; "=" не требуется; 7 это значение; но символ "=" возможен и здесь
`X4=20`	; Ось `X4; "="` требуется
`CR=7.3`	; 2 буквы; "=" требуется
`S1=470`	; скорость 1-го шпинделя: 470 об/мин
`M3=5`	; остановка 3-го шпинделя

Для адресов M, H, S а также для SPOS и SPOSA, числовое расширение может быть заменено на переменную. При этом идентификатор переменных стоит в квадратных скобках.

Примеры:

Программный код	Комментарий
`S[SPINU]=470`	; Число оборотов для шпинделя, номер которого зафиксирован в переменной `SPINU`
`M[SPINU]=3`	; Правое вращение для шпинделя, номер которого зафиксирован в переменной `SPINU`
`T[SPINU]=7`	; Предварительный выбор инструмента для шпинделя, номер которого зафиксирован в переменной `SPINU`

3.15.5 Имя

Команды согласно DIN 66025 дополняются на высокоуровневом языке программирования ЧПУ, среди прочего, названными объектами.

Названными объектами могут быть, например:

- системные переменные
- определяемые пользователем переменные
- оси / шпиндели
- подпрограммы
- кодовые слова
- метки перехода
- макросы

Примечание

Идентификаторы должны быть однозначными. Один и тот же идентификатор **не может** использоваться для различных объектов.

Правила именования

Имя можно выбирать произвольно, соблюдая следующие правила:

- Разрешенные символы:
  - Буквы: A ... Z, a ... z
  - Цифры: 0 ... 9
  - Знак подчеркивания: _
- Первыми двумя символами должны быть буквы или символы подчеркивания.
- Максимальная длина:
  - Имя программы (Страница 46): 24 символа
  - Имена осей: 8 символов
  - Имена переменных: 31 символ

Примечание

Зарезервированные кодовые слова запрещено использовать в качестве идентификаторов.

Циклы

Во избежание конфликтов именования рекомендуется учитывать следующие правила присваивания имен циклам пользователя:

Строка	Зарезервировано для имен
• CYCLE • CUST_ • GROUP_ • _ • S_ • E_ • F_	Циклы SIEMENS
• CCS_	Компилируемые циклы SIEMENS
• CC_	Компилируемые циклы пользователя

Циклы пользователя

Имена циклов пользователя рекомендуется начинать с "U_".

Переменные

Информация именования переменных представлена в следующих главах:

- Системные данные (Страница 400)
- Определение переменных пользователя (DEF) (Страница 408)

3.15.6 Постоянные

Постоянная (общ.)

Постоянная это элемент данных, значение которого не изменяется при выполнении программы, к примеру, присваивание значения адресу.

Десятичная постоянная

Числовое значение десятичной постоянной отображается в десятичной систем.

Постоянная INTEGER

Постоянная INTEGER это целое значение, т. е. последовательность цифр без десятичной точки с или без знака.

Примеры:

`X10`	Присваивание значения +10 адресу X
`X-35`	Присваивание значения -35 адресу X
`X0`	Присваивание значения 0 адресу X **Примечание:** (X0 нельзя заменить на X)

Постоянная REAL

Постоянная REAL это последовательность цифр с десятичной точкой с или без знака, а также с или без экспоненты.

Примеры:

`X10.25`	Присваивание значения +10.25 адресу X
`X-10.25`	Присваивание значения -10.25 адресу X
`X0.25`	Присваивание значения +0.25 адресу X
`X.25`	Присваивание значения +0.25 адресу X, без вводного "0"
`X=-.1EX-3`	Присваивание значения -0,1*10^{-3} адресу X

Примечание

Если для адреса с допустимым вводом десятичной точки после десятичной точки записано больше мест, чем предусмотрено для этого адреса, то он округляется до предусмотренного числа мест.

Шестнадцатеричная постоянная

Также возможны постоянные, интерпретируемые шестнадцатерично. При этом буквы "A" до "F" служат шестнадцатеричными цифрами с десятичными значениями 10 до 15.

Шестнадцатеричные постоянные заключаются между двумя апострофами и начинаются с буквы "H" с последующим шестнадцатеричным значением. Разрешаются разделительные символы между буквами и цифрами.

Пример:

Программный код	Комментарий
`$MC_TOOL_MANAGEMENT_MASK='H7F'`	; Через присвоение шестн. постоянной в машинных данных устанавливается бит 0-7.

Примечание

Максимальное количество символов ограничивается диапазоном значений целочисленного типа данных.

Двоичная постоянная

Возможны и постоянные, которые интерпретируются двоично. При этом используются только цифры "0" и "1".

Двоичные постоянные заключаются между апострофами и начинаются с буквы "B" с последующим двоичным значением. Разделительные символы между цифрами разрешаются.

Пример:

Программный код	Комментарий
`$MN_AUXFU_GROUP_SPEC='B10000001'`	; Через присвоение двоичной постоянной в машинных данных устанавливается бит 0 и бит 7.

Примечание

Максимальное количество символов ограничивается диапазоном значений целочисленного типа данных.

3.15.7 Операторы и функции вычисления

Операторы

Операторы вычисления

Системные переменные типа REAL или INT могут быть связаны следующими операторами:

Оператор	Объяснение
+	Сложение
-	Вычитание
*	Умножение

Оператор	Объяснение
/	• Деление в синхронных действиях INT / INT ⇒ **INT** • Деление в синхронных действиях с результатом REAL посредством использования функции ITOR(): ITOR(INT) / ITOR(INT) ⇒ **REAL** • Деление в программах ЧПУ: INT / INT ⇒ **REAL**
DIV	Целочисленное деление: INT / INT ⇒ **INT**
MOD	Выделение дробной части (только для типа INT) выводит остаток деления INT Пример: 3 MOD 4 = 3

Примечание

Можно соединять только переменные одного типа.

Операторы сравнения

Оператор	Объяснение
==	равно
>	не равно
<	меньше
>	больше
<=	меньше или равно
>=	больше или равно

Логические операторы

Оператор	Объяснение
NOT	НЕТ
AND	И
OR	ИЛИ
XOR	Исключающее ИЛИ

Побитовые логические операторы

Оператор	Объяснение
B_OR	побитовое ИЛИ
B_AND	побитовое И
B_XOR	побитовое исключающее ИЛИ
B_NOT	побитовое отрицание

Приоритет операторов

При выполнении в синхронном действии у операторов следующий приоритет (самый высокий приоритет: 1):

Прио.	Операторы	Объяснение
1	NOT, B_NOT	Отрицание, побитовое отрицание
2	*, /, DIV, MOD	Умножение, деление
3	+, -	Сложение, вычитание
4	B_AND	побитовое И
5	B_XOR	побитовое исключающее ИЛИ
6	B_OR	побитовое ИЛИ
7	AND	И
8	XOR	исключающее ИЛИ
9	OR	ИЛИ
10	<<	Соединение строк, тип результата STRING
11	==, <>, <, >, >=, <=	Операторы сравнения

Примечание

При использовании нескольких операторов в одном выражении настоятельно рекомендуется устанавливать однозначный приоритет отдельных операторов посредством расстановки скобок "(...)".

Пример условия в одном выражении с несколькими операторами:

Программный код

```
... WHEN ($AA_IM[X] > WERT) AND ($AA_IM[Y] > WERT1) DO ...
```

Функции вычисления

Оператор	Объяснение
SIN()	Синус
COS()	Косинус
TAN()	Тангенс
ASIN()	Арксинус
ACOS()	Арккосинус
ATAN2(,)	Арктангенс2
SQRT()	Квадратный корень
ABS()	Величина
POT()	2-я степень (квадрат)
TRUNC()	Целочисленная часть Настраиваемая точность для команд сравнения с TRUNC
ROUND ()	Округление до целого числа
LN()	натуральный логарифм
EXP()	Показательная функция (экспонента)

Индексирование

В свою очередь, индекс системных переменных типа "Поле ..." также может быть системной переменной. При этом индекс также обрабатывается в главном ходе в такте интерполяции.

Пример

Программный код

```
... WHEN ... DO $AC_PARAM[ $AC_MARKER[1] ] = 3
```

Ограничения

- Вложения индексирования другими системными переменными запрещены.
- Индекс запрещено образовывать через переменные предварительной обработки. Следовательно, следующий пример **не** разрешен, поскольку $P_EP – переменная предварительной обработки:
  $AC_PARAM[1] = $P_EP[$AC_MARKER[0]]

См. также

Функции вычисления (Страница 454)

Операции (Страница 454)

Расширенное программирование

4

4.1 Гибкое программирование ЧПУ

4.1.1 Переменные

Благодаря использованию переменных, относящихся к системным и пользовательским данным, в первую очередь в комбинации с функциями вычисления и управляющими структурами, можно гибко оформлять программы ЧПУ и циклы.

⚠ ПРЕДУПРЕЖДЕНИЕ

Опасность для персонала и оборудования из-за изменения переменных

При использовании переменных в программе ЧПУ необходимо учитывать, что операторы станков или неуполномоченные лица могут при наличии соответствующего права доступа изменить переменные и тем самым повлиять на выполнение программы. Это может стать причиной травм персонала и материального ущерба.

- Во избежание негативного влияния изменения переменных на выполнение программы необходимо предусмотреть в программе ЧПУ соответствующие проверки данных ("Input validation").

- Системные данные
  К системным данным относятся переменные, предопределенные в системе. Эти переменные имеют определенное значение. Они используются системными программами в первую очередь. Пользователь может считывать и записывать эти переменные в программах ЧПУ и циклах. Пример: Характеристики машины, настройки, системные переменные **.**
  Хотя значение системной даты не может редактироваться, пользователь может в известной мере изменить свойства путем переопределения.
  См. "Переопределение системных данных, пользовательских данных и языковых команд ЧПУ (REDEF) (Страница 414)"
- Данные пользователя
  К пользовательским данным относятся определенные пользователем переменные, значение которых задается исключительно пользователем. Система не анализирует эти переменные.
  Пользовательские данные делятся на:
  - Предопределенные переменные пользователя
    Предопределенные переменные пользователя представляют собой уже определенные в системе переменные, количество которых параметрировано в характеристиках машины. Свойства этих переменных могут настраиваться пользователем.
    См. "Переопределение системных данных, пользовательских данных и языковых команд ЧПУ (REDEF) (Страница 414)".
  - Определенных пользователем переменных
    Определенные пользователем переменные представляют собой переменные, определенные пользователем и создаваемые системой для рабочего цикла. Их число, тип данных, видимость и все другие свойства устанавливаются только пользователем.
    См. "Определение переменных пользователя (DEF) (Страница 408)"

4.1.1.1 Системные данные

Системные данные содержат предопределенные в системе переменные, обеспечивающие в программах ЦПУ и циклах доступ к текущему параметрированию СЧПУ, а также состояниям станка, СЧПУ и процессов.

Переменные предварительной обработки

Переменные предварительной обработки представляют собой системные данные, которые считываются и записываются в контексте предварительной обработки, т. е. на момент интерпретации кадра, в котором запрограммирована переменная. Переменная предварительной обработки не создает остановки предварительной обработки.

Переменные главного хода

Переменные главного хода представляют собой системные данные, которые считываются и записываются в контексте главного хода, т. е. на момент исполнения кадра, в котором запрограммирована переменная. Переменными главного хода являются:

- Переменные, которые могут быть запрограммированы в синхронных действиях (чтение/запись)
- Переменные, которые могут быть запрограммированы в программе ЧПУ и вызывают остановку предварительной обработки (чтение/запись)
- Переменные, которые могут быть запрограммированы в программе ЧПУ и получают значение при предварительной обработке, но записываются только на главном ходе (синхронно с главным ходом: только запись)

Систематика префиксов

Для особого обозначения системных данных, перед именем обычно стоит префикс, состоящий из символа $, одной или двух букв и символа подчеркивания:

$ + 1. Буква	**Объяснение: Тип данных**
Данные предварительной обработки (системные данные, считываемые / записываемые при предварительной обработке)	
$M	Машинные данные [1)]
$S	Установочные данные, защищенные области [1)]
$T	Данные управления инструментом
$P	Запрограммированные значения
$C	Переменные оболочек циклов ISO
$O	Опциональные данные
R	R-параметры (параметры для расчета) [2)]
Данные главного хода (системные данные, считываемые / записываемые при главном ходе)	
$$M	Машинные данные [1)]
$$S	Установочные данные [1)]
$A	Текущие данные главного хода
$V	Данные регулятора положения
$R	R-параметры (параметры для расчета) [2)]

[1)] Будут ли машинные и установочные данные обрабатываться как переменные предварительной обработки или переменные главного хода, зависит от того, записываются ли они с одним или двумя символами $. Форма записи может выбираться свободно в зависимости от приложения.

[2)] При использовании R-параметра в программе обработки детали / цикле как переменной предварительной обработки, префикс не записывается, например, R10. При использовании в синхронном действии как переменной главного хода в качестве префикса записывается символ $, например, $R10.

2-ая буква	Объяснение: Видимость
N	Глобальная переменная NC (**N**C)
C	Спец. для канала переменные (**C**hannel)
A	Спец. для оси переменные (**A**xis)

Граничные условия

Исключение в систематике префиксов

Следующие системные переменные отличаются от в.н. систематики префиксов:

- $TC_...: 2-ая буква C ссылается не на специфические для канала, а на специфические для зажима инструмента системные переменные (TC = Tool Carrier)
- $P_ ...: Спец. для канала системные переменные

Использование машинных и установочных данных в синхронных действиях

При использовании машинных и установочных данных в синхронных действиях через префикс можно определить, будут ли машинные или установочные данные считываться/записываться синхронно с предварительной обработкой или главным ходом.

Если данные при обработке не изменяются, то возможно считывание синхронно с предварительной обработкой. Для этого префикс машинных или установочных данных записывается с символом $:

```
ID=1 WHENEVER $AA_IM[z] < $SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Если данные при обработке изменяются, то считывание/запись должны выполняться синхронно с главным ходом. Для этого префикс машинных или установочных данных записывается с двумя символами $:

```
ID=1 WHENEVER $AA_IM[z] < $$SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Примечание

Запись машинных и установочных данных

При записи машинных или установочных данных следить за тем, чтобы активный уровень доступа при выполнении программы обработки детали / цикла разрешал бы доступ по записи и чтобы активность данных была бы "IMMEDIATE".

См. также

Переменные (Страница 399)

4.1.1.2 Предопределенные переменные пользователя: Специфические для канала R-параметры (R)

Спец. для канала R-параметры это предопределенные переменные пользователя с обозначением R, определенные как поле типа данных REAL. Традиционно для R-параметров наряду с написанием с индексом поля, например, `R[10]`, разрешено и написание без индекса поля, например, `R10`.

При использовании в синхронных действиях должна быть предустановленна буква $, например, `$R10`.

Синтаксис

При использовании в качестве переменной предварительной обработки:
`R<n>`
`R[`<выражение>]

При использовании в качестве переменной главного хода:
`$R<n>`
`$R[`<выражение>]

Объяснение

`R`:	Идентификатор при использовании в качестве переменной предварительной обработки, например, в программе обработки детали	
`$R`:	Идентификатор при использовании в качестве переменной главного хода, например, в синхронных действиях	
	Тип:	REAL
	Диапазон значений:	При не экспоненциальном представлении: ± (0.000 0001 ... 9999 9999) **Примечание:** Разрешено макс. 8 десятичных разрядов
		При экспоненциальном представлении: ± ($1*10^{-300}$ ... $1*10^{+300}$) **Примечание:** • Форма записи: <Mantisse>EX<Exponent>, например, 8.2EX-3 • Разрешено макс. 10 символов, включая знак и десятичную запятую.

<n>:	Номер R-параметра	
	Тип:	INT
	Диапазон значений:	0 - MAX_INDEX **Указание** MAX_INDEX получается из спараметрированного числа R-параметров: MAX_INDEX = (MD28050 $MN_MM_NUM_R_PARAM) - 1
<выражение>:	Индекс поля В качестве индекса поля может быть выведено любое выражение, пока результат выражения может быть преобразован в тип данных INT (INT, REAL, BOOL, CHAR)	

Пример

Присваивания R-параметрам и использование R-параметров в математических функциях:

Программный код	Комментарий
`R0=3.5678`	; Присваивание при предварительной обработке
`R[1]=-37.3`	; Присваивание при предварительной обработке
`R3=-7`	; Присваивание при предварительной обработке
`$R4=-0.1EX-5`	; Присваивание при главном ходе: `R4 = -0.1 * 10^-5`
`$R[6]=1.874EX8`	; Присваивание при главном ходе: `R6 = 1.874 * 10^8`
`R7=SIN(25.3)`	; Присваивание при предварительной обработке
`R[R2]=R10`	; Косвенная адресация через `R`-параметры
`R[(R1+R2)*R3]=5`	; Косвенная адресация через матем. выражение
`X=(R1+R2)`	; Переместить ось `X` на позицию, получаемую из суммы `R1` и `R2`
`Z=SQRT(R1*R1+R2*R2)`	; Переместить ось `Z` на позицию квадратного корня(`R1^2 + R2^2`)

См. также

Переменные (Страница 399)

4.1.1.3 Предопределенные переменные пользователя: Глобальные R-параметры (RG)

Функция

Наряду со специфическими для канала R-параметрами пользователю доступны и глобальные R-параметры. Они существуют внутри СЧПУ однократно и могут считываться/записываться из всех каналов.

Глобальные R-параметры используются, например, для получения информации из одного канала в следующий. Другой пример – глобальные настройки, которые должны обрабатываться для всех каналов, например вылет заготовки из шпинделя.

Считывание и запись глобальных R-параметров осуществляется через пользовательский интерфейс или в программе ЧПУ при предварительной обработке. Использование в синхронных действиях или технологических циклах невозможно.

Примечание

При считывании и записи глобальных R-параметров синхронизация между каналами **не** осуществляется.

Поскольку считывание и запись осуществляются при предварительной обработке, момент времени, в который записанное значение из одного канала активируется в другом канале, не определен.

Пример:

В канале 1 цикл с одним глобальным R-параметром выполняется как счетчик циклов. Канал 2 записывает в этот глобальный R-параметр значение, которое приводит к отмене цикла в канале 1. Однако все циклы, интерпретированные к этому моменту в канале 1 при предварительной обработке, еще выполняются. Число таких циклов не определено и зависит среди прочего от загрузки канала.

Синхронизацию между каналами пользователь должен реализовать самостоятельно на прикладном уровне, например, с помощью меток WAIT!

Синтаксис

Запись в программе ЧПУ
`RG[<n>]=`<значение>
`RG[`<выражение>`]=`<значение>

Чтение в программе ЧПУ
`R...=RG[<n>]`
`R...=RG[`<выражение>`]`

Объяснение

`RG`:	Стандартное имя адреса ЧПУ для **глобальных** R-параметров **Примечание:** Имя адреса ЧПУ можно настраивать через MD15800 $MN_R_PARAM_NCK_NAME	
`<n>`:	Номер глобального R-параметра	
	Тип:	INT
	Диапазон значений:	0 ... MAX_INDEX **Примечание** MAX_INDEX получается из спараметрированного числа глобальных R-параметров: MAX_INDEX = (MD18156 $MN_MM_NUM_R_PARAM_NCK) - 1
<выражение>:	В качестве индекса поля может быть выведено любое выражение, пока результат выражения может быть преобразован в тип данных INT (INT, REAL, BOOL, CHAR)	

<значение>:	Значение глобального R-параметра	
	Тип:	REAL
	Диапазон значений:	При не экспоненциальном представлении: ± (0.000 0001 ... 9999 9999) **Примечание:** Разрешено макс. 8 десятичных разрядов
		При экспоненциальном представлении: ± ($1*10^{-300}$ ... $1*10^{+300}$) **Примечание:** • Форма записи: <мантисса>EX<показатель степени>, например 8.2EX-3 • Разрешено макс. 10 символов, включая знак и десятичную запятую.

4.1.1.4 Предопределенные переменные пользователя: Link-переменные

Через Link-переменные в рамках функции "NCU-Link" возможен обмен циклическими данными между NCU, соединенными друг с другом в одной сети. При этом они обеспечивают спец. для формата данных доступ к памяти Link-переменных. Как размер, так и структура данных память Link-переменных, определяется пользователем / изготовителем станка спец. для установки.

Link-переменные это глобальные системные переменные пользователя, которые при сконфигурированной Link-коммуникации могут считываться и записываться всеми NCU структуры связи в программах обработки деталей и циклах. В отличие от глобальных переменных пользователя (GUD), Link-переменные могут использоваться и в синхронных действиях.

На установках без активных NCU-Link, Link-переменные могут использоваться локально для СЧПУ наряду с глобальными переменными пользователя (GUD), как дополнительные глобальные переменные пользователя.

Синтаксис

```
$A_DLB[<индекс>]
$A_DLW[<индекс>]
$A_DLD[<индекс>]
$A_DLR[<индекс>]
```

Объяснение

$A_DLB:	Link-переменная для формата данных BYTE (1 байт)	
	Тип данных:	UINT
	Диапазон значений:	0 ... 255
$A_DLW:	Link-переменная для формата данных WORD (2 байта)	
	Тип данных:	INT
	Диапазон значений:	-32768 ... 32767

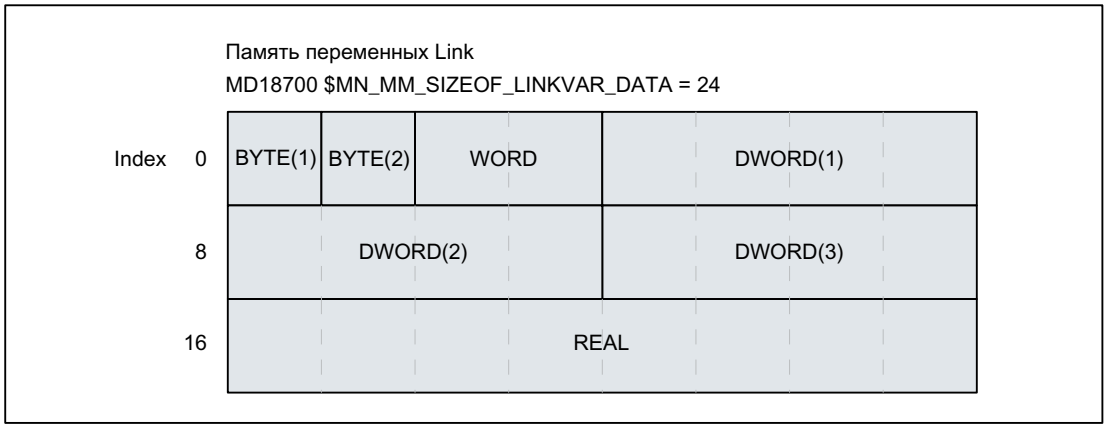
$A_DLD:	Link-переменная для формата данных DWORD (4 байта)	
	Тип данных:	INT
	Диапазон значений:	-2147483648 ... 2147483647
$A_DLR:	Link-переменная для формата данных REAL (8 байт)	
	Тип данных:	REAL
	Диапазон значений:	$\pm(2{,}2*10^{-308} \ldots 1{,}8*10^{+308})$
<индекс>:	Индекс адреса в байтах, рассчитанный от начала памяти Link-переменных	
	Тип данных:	INT
	Диапазон значений:	0 - MAX_INDEX **Указание** • MAX_INDEX получается из спараметрированного размера памяти Link-переменных: MAX_INDEX = (MD18700 $MN_MM_SIZEOF_LINKVAR_DATA) - 1 • Могут программироваться только индексы, чтобы адресованные в памяти Link-переменных байты лежали бы в границах одного формата данных ⇒ Индекс = n * байт, где n = 0, 1, 2, ... – $A_DLB[i]: i = 0, 1, 2, ... – $A_DLW[i]: i = 0, 2, 4, ... – $A_DLD[i]: i = 0, 4, 8, ... – $A_DLR[i]: i = 0, 8, 16, ...

Пример

В системе автоматики имеется 2 NCU (NCU1 и NCU2). К NCU1 подключена ось станка AX2, которая перемещается как Link-ось NCU2.

NCU1 циклически записывает фактическое значение тока ($VA_CURR) оси AX2 в память Link-переменных. NCU2 циклически считывает переданное через Link-коммуникацию фактическое значение тока и при превышении предельного значения выводит ошибку 61000.

Структура данных в памяти Link-переменных представлена на следующем рисунке. Фактическое значение тока передается через значение REAL.



УЧПУ1

NCU1 циклически записывает в статическом синхронном действии в такте IPO фактическое значение тока оси AX2 через Link-переменную $A_DLR[16] в память Link-переменных.

Программный код

```
N111 IDS=1 WHENEVER TRUE DO $A_DLR[16]=$VA_CURR[AX2]
```

УЧПУ2

NCU2 циклически считывает в статическом синхронном действии в такте IPO фактическое значение тока оси AX2 через Link-переменную $A_DLR[16] из памяти Link-переменных. Если фактическое значение тока больше 23.0 A, то отображается ошибка 61000.

Программный код

```
N222 IDS=1 WHEN $A_DLR[16] > 23.0 DO SETAL(61000)
```

4.1.1.5 Определение переменных пользователя (DEF)

Команда `DEF` позволяет определять пользовательские переменные (User Data), и присваивать им значения.

Согласно области действия, т. е. области, в которой переменная отображается, существуют следующие категории переменных пользователя:

- Локальные переменные пользователя (LUD)
  Локальные переменные пользователя (LUD) — переменные, определенные в программе ЧПУ, которая на момент выполнения не является главной программой. Они создаются при вызове программы ЧПУ и удаляются при сбросе (Reset) конца программы или следующем запуске СЧПУ. Обращение к LUD возможно только внутри программы ЧПУ, в которой они определены.
- Глобальные программные переменные пользователя (PUD)
  Глобальные программные переменные пользователя (PUD) — переменные пользователя, определенные в используемой в качестве главной программы программе ЧПУ. Они создаются при вызове программы ЧПУ и удаляются при сбросе (Reset) конца программы или следующем запуске СЧПУ. Обращение к PUD возможно в главной программе и во всех подпрограммах.

  Примечание

  Доступность глобальных программных переменных пользователя (PUD)

  Определенные в главной программе глобальные программные переменные пользователя (PUD) доступны в подпрограммах только при следующих установленных машинных данных:

  MD11120 $MN_LUD_EXTENDED_SCOPE = 1

  При MD11120 = 0 определенные в главной программе глобальные программные переменные пользователя доступны только в главной программе.

- Глобальные переменные пользователя (GUD)
  Глобальные переменные пользователя (GUD) – это спец. для ЧПУ или канала глобальные переменные, определенные в блоке данных (SGUD, MGUD, UGUD, GUD4 ... GUD9) и сохраняющиеся после сброса (Reset) конца программы или следующего запуска СЧПУ. Обращение к GUD возможно во всех программах ЧПУ.

Переменные пользователя перед их использованием (чтение / запись) должны быть определены. Соблюдать следующие правила:

- GUD должны быть определены в файле определений, например, _N_DEF_DIR/ _N_UGUD_DEF.
- PUD и LUD должны быть определены в разделе описаний программы ЧПУ.
- Определение данных должно выполняться в отдельном кадре.
- Для описания данных может использоваться только один тип данных.
- Для описания данных может быть определено несколько переменных одного типа данных.

Синтаксис

LUD и PUD
`DEF` <тип> <физ_единица> <предельные значения> <имя>[<значение_1>, <значение_2>, <значение_3>]=<иниц_значение>

GUD
DEF <область> <остановка предварительной обработки> <права доступа> <класс данных> <тип> <физ_единица> <предельные значения> <имя>[<значение_1>, <значение_2>, <значение_3>]=<иниц_значение>

Объяснение

DEF:	Команда определения переменных пользователя GUD, PUD, LUD	
<область>:	Область действия, релевантно только для GUD:	
	ЧПУ:	Глобальная для ЧПУ переменная пользователя
	CHAN:	Глобальная для канала переменная пользователя
<остановка предварительной обработки>:	Остановка предварительной обработки, релевантно только для GUD (опция)	
	SYNR:	Остановка предварительной обработки при чтении
	SYNW:	Остановка предварительной обработки при записи
	SYNRW:	Остановка предварительной обработки при чтении/записи
<права доступа>:	Степень защиты для чтения/записи GUD через программу ЧПУ или BTSS (опция)	
	APRP <степень защиты>:	Чтение: Программа ЧПУ
	APWP <степень защиты>:	Запись: Программа ЧПУ
	APRB <степень защиты>:	Чтение: BTSS
	APWB <степень защиты>:	Запись: BTSS
	<степень защиты>:	Диапазон значений: 0 ... 7
	См. "Атрибут: Права доступа (APR, APW, APRP, APWP, APRB, APWB) (Страница 425)"	
<класс данных>:	Назначение класса данных (только SINUMERIK 828D!)	
	DCM:	класс данных M (= Manufacturer)
	DCI:	класс данных I (= Individual)
	DCU:	класс данных U (= User)
<тип>:	Тип данных:	
	INT:	Целочисленное значение со знаком
	REAL:	Действительное число (LONG REAL по IEEE)
	BOOL:	Значение истинности TRUE (1) / FALSE (0)
	CHAR:	Символы ASCII
	STRING[<макс.длина>]:	Строка символов определенной длины
	AXIS:	Идентификатор оси/шпинделя
	FRAME:	Геометрические данные для статической трансформации координат
	См. "Типы данных (Страница 437)"	

<физическая единица>:	Физическая единица (опция)	
	PHU <единица>:	физическая единица
	См. "Атрибут: Физическая единица (PHU) (Страница 422)"	
<предельные значения>:	нижнее / верхнее предельное значение (опция)	
	LLI <предельное значение>:	Нижнее предельное значение (lower limit)
	ULI <предельное значение>:	Верхнее предельное значение (upper limit)
	См. "Атрибут: предельные значения (LLI, ULI) (Страница 421)"	
<имя>:	Имя переменной **Указание** • Макс. 31 символ • Первыми двумя символами должны быть буква и/или символ подчеркивания. • Символ "$" зарезервирован для системных переменных и не может использоваться.	
[<значение_1>, <значение_2>, <значение_3>]:	Указание размеров поля для 1- до макс. 3-мерных переменных поля (опция) По инициализации переменных поля см. "Определение и инициализация переменных поля (DEF, SET, REP) (Страница 431)"	
<иниц_значение>:	Инициализирующее значение (опция) См. "Атрибут: Инициализирующее значение (Страница 418)" По инициализации переменных поля см. "Определение и инициализация переменных поля (DEF, SET, REP) (Страница 431)"	

Примеры

Пример 1: Определения переменных пользователя в блоке данных для изготовителя станка

Программный код	Комментарий
`%_N_MGUD_DEF`	`;` GUD-блок: Изготовитель станка
`$PATH=/_N_DEF_DIR`	
`DEF CHAN REAL PHU 24 LLI 0 ULI 10 STROM_1, STROM_2`	
; Описание	
; Определение двух `GUD: STROM_1, STROM_2`	
; Область действия: в канале	
; Тип данных: `REAL`	
; Остановка предварительной обработки: не запрограммировано => значение по умолчанию = нет остановки предварительной обработки	
; Физ. Единица: 24 = [`A`]	
; Предельные значения: `Low = 0.0, High = 10.0`	
; Права доступа: не запрограммировано => значение по умолчанию = 7 = положение кодового переключателя 0	
; Значение инициализации: не запрограммировано => значение по умолчанию = 0.0	
`DEF NCK REAL PHU 13 LLI 10 APWP 3 APRP 3 APWB 0 APRB 2 ZEIT_1=12, ZEIT_2=45`	

Программный код	Комментарий
; Описание	
; Определение двух GUD: ZEIT_1, ZEIT_2	
; Область действия: по ЧПУ	
; Тип данных: REAL	
; Остановка предварительной обработки: не запрограммировано => значение по умолчанию = нет остановки предварительной обработки	
; Физ. Единица: 13 = [сек]	
; Предельные значения: Low = 10.0, High = не запрограммировано => верхняя граница области определения	
; Права доступа:	
; программа ЧПУ: Запись/чтение = 3 = пользователь	
; BTSS: Запись = 0 = Siemens, чтение = 3 = пользователь	
; Значение инициализации: ZEIT_1 = 12.0, ZEIT_2 = 45.0	
DEF NCK APWP 3 APRP 3 APWB 0 APRB 3 STRING[5] GUD5_NAME = "COUNTER"	
; Описание	
; Определение одной GUD: GUD5_NAME	
; Область действия: по ЧПУ	
; Тип данных: STRING, макс. 5 символов	
; Остановка предварительной обработки: не запрограммировано => значение по умолчанию = нет остановки предварительной обработки	
; Физ. Единица: не запрограммировано => значение по умолчанию = 0 = нет физической единицы	
; Предельные значения: не запрограммировано => границы области определения: Low = 0, High = 255	
; Права доступа:	
; программа ЧПУ: Запись/чтение = 3 = пользователь	
; BTSS: Запись = 0 = Siemens, чтение = 3 = пользователь	
; Значение инициализации: "COUNTER"	
M30	

Пример 2: Глобальные и локальные программные переменные пользователя (PUD / LUD)

Программный код	Комментарий
PROC MAIN	; Главная программа
DEF INT VAR1	; PUD-определение
...	
SUB2	; Вызов подпрограммы
...	
M30	

Программный код	Комментарий
PROC SUB2	; Подпрограмма SUB2
DEF INT VAR2	; LUD-ОПРЕДЕЛЕНИЕ
...	
IF (VAR1==1)	; Чтение PUD
VAR1=VAR1+1	; Чтение и запись PUD

Программный код	Комментарий
VAR2=1	; Запись LUD
ENDIF	
SUB3	; Вызов подпрограммы
...	
M17	

Программный код	Комментарий
PROC SUB3	; Подпрограмма SUB3
...	
IF (VAR1==1)	; Чтение PUD
VAR1=VAR1+1	; Чтение и запись PUD
VAR2=1	; Ошибка: LUD из SUB2 не известна
ENDIF	
...	
M17	

Пример 3: Определение и использование переменных пользователя типа данных AXIS

Программный код	Комментарий
DEF AXIS ABSZISSE	; 1-я Гео-ось
DEF AXIS SPINDLE	; Шпиндель
...	
IF ISAXIS(1) == FALSE GOTOF WEITER	
ABSZISSE = $P_AXN1	
WEITER:	
...	
SPINDLE=(S1)	; 1-я Шпиндель
OVRA[SPINDLE]=80	; Процентовка шпинделя = 80 %
SPINDLE=(S3)	; 3-й шпиндель

Граничные условия

Глобальные переменные пользователя (GUD)

В рамках определения глобальных переменных пользователя (GUD) должны быть учтены следующие машинные данные:

№	Идентификатор: $MN_	Объяснение
11140	GUD_AREA_ SAVE_TAB	Дополнительная резервная копия для блоков GUD
18118 [1)]	MM_NUM_GUD_MODULES	Число файлов GUD в активной файловой системе
18120 [1)]	MM_NUM_GUD_NAMES_NCK	Число глобальных имен GUD
18130 [1)]	MM_NUM_GUD_NAMES_CHAN	Число спец. для канала имен GUD

№	Идентификатор: $MN_	Объяснение
18150 [1)]	MM_GUD_VALUES_MEM	Место в памяти для глобальных значений GUD
18660 [1)]	MM_NUM_SYNACT_GUD_REAL	Число конфигурируемых GUD тип данных REAL
18661 [1)]	MM_NUM_SYNACT_GUD_INT	Число конфигурируемых GUD тип данных INT
18662 [1)]	MM_NUM_SYNACT_GUD_BOOL	Число конфигурируемых GUD тип данных BOOL
18663 [1)]	MM_NUM_SYNACT_GUD_AXIS	Число конфигурируемых GUD тип данных AXIS
18664 [1)]	MM_NUM_SYNACT_GUD_CHAR	Число конфигурируемых GUD тип данных CHAR
18665 [1)]	MM_NUM_SYNACT_GUD_STRING	Число конфигурируемых GUD тип данных STRING

[1)] MD у SINUMERIK 828D только для чтения!

Выходящее за рамки канала использование глобальной переменной пользователя ЧПУ типа данных AXIS

Глобальная переменная пользователя ЧПУ типа данных `AXIS`, инициализированная при определении в блоке данных с идентификатором оси, может использоваться в различных каналах ЧПУ, только если ось в этих каналах имеет тот же номер оси канала.

Если это не так, то необходимо загрузить переменную в начале программы ЧПУ или, как в примере ниже, использовать функцию AXNAME(...) (см. "Осевые функции (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Страница 914)").

Программный код	Комментарий
`DEF NCK STRING[5] ACHSE="X"`	; Определение в блоке данных
...	
`N100 AX[AXNAME(ACHSE)]=111 G00`	; использование в программе ЧПУ

4.1.1.6 Переопределение системных данных, пользовательских данных и языковых команд ЧПУ (REDEF)

С помощью команды `REDEF` можно изменять атрибуты системных данных, пользовательских данных и языковых команд ЧПУ. Базовым условием для переопределения является выполнение по времени после соответствующего определения.

Одновременное изменение нескольких атрибутов при переопределении невозможно. Для каждого изменяемого атрибута должен быть запрограммирован отдельный оператор `REDEF`.

Если программируется несколько конкурирующих изменений атрибутов, то всегда активируется последнее изменение.

Сброс величин атрибутов

Измененные с REDEF атрибуты для прав доступа и момент инициализации через повторное программирование REDEF с последующим именем переменной или языковой команды ЧПУ могут быть сброшены на свое значение по умолчанию:

- Права доступа: Степень защиты 7
- Момент инициализации: нет инициализации или сохранение текущего значения

Переопределяемые атрибуты

См. "Обзор определяемых и переопределяемых атрибутов (Страница 429)".

Локальные переменные пользователя (PUD / LUD)

Переопределения для локальных переменных пользователя (PUD / LUD) невозможны.

Синтаксис

```
REDEF <имя> <остановка предварительной обработки>
REDEF <имя> <физическая единица>
REDEF <имя> <предельные значения>
REDEF <имя> <права доступа>
REDEF <имя> <момент инициализации>
REDEF <имя> <момент инициализации> <иниц_значение>
REDEF <имя> <класс данных>
REDEF <имя>
```

Объяснение

REDEF:	Команда для переопределения определенного атрибута или сброса атрибута "Права доступа" и/или "Момент инициализации" системных переменных, переменных пользователя и языковых команд ЧПУ	
<имя>:	Имя уже определенной переменной или языковой команды ЧПУ	
<VL-Stop>:	Остановка предварительной обработки	
	SYNR:	Остановка предварительной обработки при чтении
	SYNW:	Остановка предварительной обработки при записи
	SYNRW:	Остановка предварительной обработки при чтении/записи

<физическая единица>:	Физическая единица	
	PHU <единица>:	физическая единица
	См. "Атрибут: Физическая единица (PHU) (Страница 422)". **Указание** Переопределение невозможно для: • Системные переменные • Глобальные данные пользователя (GUD) типов данных: BOOL, AXIS, STRING, FRAME	
<предельные значения>:	нижнее /верхнее предельное значение	
	LLI <предельное значение>:	Нижнее предельное значение (lower limit)
	ULI <предельное значение>:	Верхнее предельное значение (upper limit)
	См. "Атрибут: предельные значения (LLI, ULI) (Страница 421)". **Указание** Переопределение невозможно для: • Системные переменные • Глобальные данные пользователя (GUD) типов данных: BOOL, AXIS, STRING, FRAME	
<права доступа>:	Права доступа для чтения / записи через программу обработки детали или BTSS	
	APX <степень защиты>:	Выполнение: Языковой элемент ЧПУ
	APRP <степень защиты>:	Чтение: Программа обработки детали
	APWP <степень защиты>:	Запись: Программа обработки детали
	APRB <степень защиты>:	Чтение: OPI
	APWB <степень защиты>:	Запись: OPI
	<степень защиты>:	Диапазон значений: 0 ... 7
	См. "Атрибут: Права доступа (APR, APW, APRP, APWP, APRB, APWB) (Страница 425)".	
<момент инициализации>:	Момент времени повторной инициализации переменной	
	INIPO:	Power On
	INIRE:	Конец главной программы, NC-Reset или Power On
	INICF:	NEWCONF или конец главной программы, сброс ЧПУ или Power On
	PRLOC:	Конец главной программы, NC-Reset после локального изменения или Power On
	См. "Атрибут: Инициализирующее значение (Страница 418)".	

<иниц_значение>:	Инициализирующее значение При переопределении инициализирующего значения всегда должен быть указан и момент инициализации (см. <момент инициализации>). См. "Атрибут: Инициализирующее значение (Страница 418)". По инициализации переменных поля см. "Определение и инициализация переменных поля (DEF, SET, REP) (Страница 431)". **Указание** Переопределение невозможно для системных переменных, кроме установочных данных.	
<класс данных>:	Назначение класса данных (только SINUMERIK 828D!)	
	DCM:	класс данных M (= Manufacturer)
	DCI:	класс данных I (= Individual)
	DCU:	класс данных U (= User)

Пример

Переопределение системной переменной $TC_DPCx в блоке данных для изготовителя станка

Программный код

```
%_N_MGUD_DEF                                    ; GUD-блок: Изготовитель станка
N100 REDEF $TC_DPC1 APWB 2 APWP 3
N200 REDEF $TC_DPC2 PHU 21
N300 REDEF $TC_DPC3 LLI 0 ULI 200
N400 REDEF $TC_DPC4 INIPO (100, 101, 102, 103)
N800 REDEF $TC_DPC1
N900 REDEF $TC_DPC4
M30
```

к N100: Право доступа по записи: BTSS = степень защиты 2, программа обработки детали = степень защиты 3

к N200: Физическая единица [%]

к N300: нижнее предельное значение = 0, верхнее предельное значение = 200

к N400: Переменная поля инициализируется при PowerOn с четырьмя значениями.

к N800 / N900: Сброс величин атрибута "Права доступа" и/или "Момент инициализации"

Примечание

Использование файлов ACCESS

При использовании файлов ACCESS переопределение прав доступа должно быть передано из _N_MGUD_DEF в _N_MACCESS_DEF.

Граничные условия

Гранулярность

Переопределение всегда относится к целой, однозначно обозначенной своим именем переменной. Невозможно, например, присваивать отдельным элементам поля переменных поля различные значения атрибутов.

4.1.1.7 Атрибут: Инициализирующее значение

Определение (DEF) переменных пользователя

При определении для следующих переменных пользователя может быть задано инициализирующее значение:

- Глобальные переменные пользователя (GUD)
- Глобальные программные переменные пользователя (PUD)
- Локальные переменные пользователя (LUD)

Переопределение (REDEF) системных переменных и переменных пользователя

При переопределении для следующих переменных может быть задано инициализирующее значение:

- Системные данные
  - Установочные данные
- Данные пользователя
  - R-параметры
  - Переменные синхронных действий ($AC_MARKER, $AC_PARAM, $AC_TIMER)
  - GUD синхронных действий (SYG_xy[ ], где x=R, I, B, A, C, S и y=S, M, U, 4, ..., 9)
  - Параметры EPS
  - Данные инструмента OEM
  - Данные магазина OEM
  - Глобальные переменные пользователя (GUD)

Момент реинициализации

При переопределении можно задать момент времени, когда переменная должна быть реинициализирована, т. е. снова установлена на инициализирующее значение:

- `INIPO` (Power On)
  Реинициализация переменной при PowerOn.
- `INIRE` (Reset)
  Реинициализация переменной при NC-Reset, ГРР-Reset, завершении программы обработки детали (`M02` / `M30`) или Power On.

- `INICF` (NEWCONF)
  Реинициализация переменной выполняется при функции "Активация машинных данных" через HMI, команду программы обработки детали `NEWCONF` или сброс ЧПУ, сброс ГРР, завершении программы обработки детали (M02 / M30) или Power On.
- `PRLOC` (локальное программное изменение)
  Реинициализация переменной при NC-Reset, ГРР-Reset или завершении программы обработки детали (M02 / M30) только в том случае, если она изменена в рамках текущей программы обработки детали.
  Атрибут `PRLOC` может использоваться только в комбинации с программируемыми установочными данными (см. таблицу ниже).

Таблица 4-1 Программируемые установочные данные

Номер	Идентификатор	G-команда [1)]
42000	$SC_THREAD_START_ANGLE	`SF`
42010	$SC_THREAD_RAMP_DISP	`DITS` / `DITE`
42400	$SA_PUNCH_DWELLTIME	`PDELAYON`
42800	$SA_SPIND_ASSIGN_TAB	`SETMS`
43210	$SA_SPIND_MIN_VELO_G25	`G25`
43220	$SA_SPIND_MAX_VELO_G26	`G26`
43230	$SA_SPIND_MAX_VELO_LIMS	`LIMS`
43300	$SA_ASSIGN_FEED_PER_REV_SOURCE	`FPRAON`
43420	$SA_WORKAREA_LIMIT_PLUS	`G26`
43430	$SA_WORKAREA_LIMIT_MINUS	`G25`
43510	$SA_FIXED_STOP_TORQUE	`FXST`
43520	$SA_FIXED_STOP_WINDOW	`FXSW`
43700	$SA_OSCILL_REVERSE_POS1	`OSP1`
43710	$SA_OSCILL_REVERSE_POS2	`OSP2`
43720	$SA_OSCILL_DWELL_TIME1	`OST1`
43730	$SA_OSCILL_DWELL_TIME2	`OST2`
43740	$SA_OSCILL_VELO	`FA`
43750	$SA_OSCILL_NUM_SPARK_CYCLES	`OSNSC`
43760	$SA_OSCILL_END_POS	`OSE`
43770	$SA_OSCILL_CTRL_MASK	`OSCTRL`
43780	$SA_OSCILL_IS_ACTIVE	`OS`
43790	$SA_OSCILL_START_POS	`OSB`
1) С этой G-командой происходит обращение к установочным данным		

Граничные условия

Инициализирующее значение: Глобальные переменные пользователя (GUD)

- Для глобальных переменных пользователя (GUD) с областью действия ЧПУ в качестве момента инициализации можно задавать только `INIPO` (Power On).
- Для глобальных переменных пользователя (GUD) с областью действия `CHAN` в качестве момента инициализации наряду с `INIPO` (Power On), можно задавать и `INIRE` (Reset) или `INICF` (NEWCONF).
- У глобальных переменных пользователя (GUD) с областью действия `CHAN` и моментом инициализации `INIRE` (Reset) или `INICF` (NEWCONF) при сбросе ЧПУ, сбросе ГРР и "Активация машинных данных" новая инициализация переменных выполняется только в каналах, в которых были запущены названные события.

Инициализирующее значение: Тип данных FRAME

Для переменных типа данных `FRAME` инициализирующее значение не указывается. Переменные типа данных `FRAME` всегда инициализируются не явно с фреймом по умолчанию.

Инициализирующее значение: Тип данных CHAR

Для переменных типа данных `CHAR` вместо кода ASCII (0...255) может быть запрограммирован и соответствующий символ ASCII в кавычках, например, "A"

Инициализирующее значение: Тип данных STRING

У переменных типа данных `STRING` строка символов должна быть заключена в кавычки, например: ...= "MASCHINE_1"

Инициализирующее значение: Тип данных AXIS

Для переменных типа данных `AXIS` при расширенном написании адреса идентификатор оси должен быть заключен в скобки, например: ...=(X3)

Инициализирующее значение: Системная переменная

Задача спец. инициализирующих значений пользователя для системных переменных через переопределение невозможна. Инициализирующие значения системных переменных фиксировано заданы системой. Но через переопределение можно изменить момент времени (`INIRE`, `INICF`) реинициализации системной переменной.

Не явное инициализирующее значение: Тип данных AXIS

Для переменных типа данных `AXIS` используется следующее не явное инициализирующее значение:

- Системные данные: "первая гео-ось"
- GUD синхронных действий (обозначение: SYG_A*), PUD, LUD: идентификатор оси из машинных данных: MD20082 $MC_AXCONF_CHANAX_DEFAULT_NAME

Не явное инициализирующее значение: Данные инструментов и магазинов

Для данных инструментов и магазинов инициализирующие значения могут быть заданы через следующие машинные данные: MD17520 $MN_TOOL_DEFAULT_DATA_MASK

Примечание

Синхронизация

За синхронизацию событий, запускающих реинициализацию глобальной переменной, с чтением этой переменной в другом месте, отвечает исключительно пользователь / изготовитель станка.

См. также

Переменные (Страница 399)

4.1.1.8 Атрибут: предельные значения (LLI, ULI)

Верхнее и нижнее предельное значение области определения может быть задано только для следующих типов данных:

- INT
- REAL
- CHAR

Определение (DEF) переменных пользователя: Предельные значения и не явные инициализирующие значения

Если при определении переменной пользователя одного из названных выше типов данных явное инициализирующее значение не определяется, то переменная устанавливается на не явное инициализирующее значение типа данных:

- INT: 0
- REAL: 0.0
- CHAR: 0

Если не явное инициализирующее значение выходит за пределы установленной через запрограммированные предельные значения области определения, то переменная инициализируется с предельным значением, наиболее близким к не явному инициализирующему значению:

- Не явное инициализирующее значение < Нижнее предельное значение (LLI) ⇒ Инициализирующее значение = Нижнее предельное значение
- Не явное инициализирующее значение > Верхнее предельное значение (ULI) ⇒ Инициализирующее значение = Верхнее предельное значение

Примеры:

Программный код	Комментарий
`DEF REAL GUD1`	; Нижнее предельное значение = Граница области определения ; Верхнее предельное значение = Граница области определения ; Инициализирующее значение не запрограммировано ; => Не явное инициализирующее значение = 0.0
`DEF REAL LLI 5.0 GUD2`	; Нижнее предельное значение = 5.0 ; Верхнее предельное значение = Граница области определения ; => Инициализирующее значение = 5.0
`DEF REAL ULI -5 GUD3`	; Нижнее предельное значение = Граница области определения ; Верхнее предельное значение = -5.0 ; => Инициализирующее значение = -5.0

Переопределение (REDEF) переменных пользователя: Предельные значения и актуальные фактические значения

Если при переопределении предельных значений переменной пользователя они изменяются таким образом, что актуальное фактическое значение лежит за пределами новой области определения, то выводится ошибка и предельные значения не применяются.

Примечание

Переопределение (REDEF) переменных пользователя

При переопределении предельных значений переменной пользователя следить за непротиворечивым изменением следующих значений:

- Предельные значения
- Фактическое значение
- Инициализирующее значение при переопределении и при автоматической реинициализации из-за INIPO, INIRE или INICF

См. также

Переменные (Страница 399)

4.1.1.9 Атрибут: Физическая единица (PHU)

Физическая единица может задаваться только для переменных следующих типов данных:

- INT
- REAL

Программируемые физические единицы (PHU)

Физическая единица указывается как число с фиксированной запятой: `PHU` <единица>

Могут программироваться следующие физические единицы:

<Единица>	Значение	Физическая единица
0	Нет физической единицы	-
1	Линейная или угловая позиция [1)2)]	[мм], [дюйм], [градус]
2	Линейная позиция [2)]	[мм], [дюйм]
3	Угловая позиция	[градус]
4	Линейная или угловая скорость [1)2)]	[мм/мин], [дюймов/мин], [об/мин]
5	Линейная скорость [2)]	[мм/мин]
6	Угловая скорость	[об/мин]
7	Линейное или угловое ускорение [1)2)]	[м/сек^2], [дюймов/сек^2], [об/сек^2]
8	Линейное ускорение [2)]	[м/сек^2], [дюймов/сек^2]
9	Угловое ускорение	[об/сек^2]
10	Линейный или угловой рывок [1)2)]	[м/сек^3], [дюймов/сек^3], [об/сек^3]
11	Линейный рывок [2)]	[м/сек^3], [дюймов/сек^3]
12	Угловой рывок	[об/сек^3]
13	Время	[сек]
14	Усиление регулятора положения	[16.667/сек]
15	Окружная подача [2)]	[мм/об], [дюймов/об]
16	Температурная компенсация [1)2)]	[мм], [дюйм]
18	Сила	[Н]
19	Масса	[кг]
20	Момент инерции [3)]	[кгм^2]
21	Процент	[%]
22	Частота	[Гц]
23	Напряжение	[В]
24	Ток	[А]
25	Температура	[°C]
26	Угол	[градус]
27	KV	[1000/мин]
28	Линейная или угловая позиция [3)]	[мм], [дюйм], [градус]
29	Скорость резания [2)]	[м/мин], [футов/мин]
30	Окружная скорость [2)]	[м/сек], [футов/сек]
31	Сопротивление	[Ом]
32	Индуктивность	[мГн]
33	Момент вращения [3)]	[Нм]
34	Постоянная момента вращения [3)]	[Нм/А]
35	Усиление регулятора тока	[В/А]
36	Усиление регулятора скорости [3)]	[Нм/(рад*сек)]
37	Скорость	[об/мин]
42	Мощность	[кВт]
43	Ток, низкий	[мкА]

<Единица>	Значение	Физическая единица
46	Момент вращения, низкий [3)]	[мкНм]
48	Промилле	-
49	-	[Гц/сек]
65	Расход	[л/мин]
66	Давление	[бар]
67	Объем [3)]	[$см^3$]
68	Петлевой коэффициент [3)]	[мм/(В*мин)]
69	Петлевой коэффициент регулятора усилия	[Н/В]
155	Шаг резьбы [3)]	[мм/об], [дюймов/об]
156	Изменение шага резьбы [3)]	[мм/об / об], [дюймов/об / об]
1) Физическая единица зависит от типа оси: линейная или круговая ось		
2) Переключение системы единиц G70/G71(дюймовая/метрическая) После переключения основной системы (MD10240 $MN_SCALING_SYSTEM_IS_METRIC) с G70/G71 при доступах по записи/чтению к содержащим длины системным переменным и переменным пользователя пересчет значений (фактическое значение, значение по умолчанию и предельные значения **не** выполняется G700/G710(дюймовая/метрическая) После переключения основной системы (MD10240 $MN_SCALING_SYSTEM_IS_METRIC) с G700/G710 при доступах по записи/чтению к содержащим длины системным переменным и переменным пользователя пересчет значений (фактическое значение, значение по умолчанию и предельные значения **выполняется**		
3) Переменная **не** переводится автоматически в актуальную систему единиц ЧПУ (дюймовая/метрическая). За перевод отвечает только пользователь / изготовитель станка.		

Примечание

Переполнение уровней из-за преобразования форматов

Внутренним форматом хранения для переменных пользователя (GUD / PUD / LUD) с содержащими длины физическими единицами является метрический формат. Избыточное использование таких переменных в главном ходе NCK, к примеру, в синхронных действиях, при переключении системы единиц может вызвать перегрузку процессора на уровне интерполятора, ошибка 4240.

Примечание

Совместимость единиц

При использовании переменных (присваивание, сравнение, вычисление и т. п.), контроль совместимости используемых единиц не осуществляется. За возможно необходимый перевод отвечает только пользователь / изготовитель станка.

См. также

Переменные (Страница 399)

4.1.1.10 Атрибут: Права доступа (APR, APW, APRP, APWP, APRB, APWB)

Обозначение

Обозначение атрибутов доступа AP... состоит из:

1. A: **A**ccess (доступ)
2. P: **P**rotection (защита)
3. R / W: **R**ead / **W**rite (чтение / запись)
4. P / B: **P**rogram / **B**TSS (программа / BTSS)

Права доступа / степени защиты

Правам доступа соответствуют следующие, указываемые при программировании степени защиты:

Право доступа	Степень защиты
Системный пароль	0
Пароль изготовителя станка	1
Сервисный пароль	2
Пароль пользователя	3
Положение кодового переключателя 3	4
Положение кодового переключателя 2	5
Положение кодового переключателя 1	6
Положение кодового переключателя 0	7

Определение (DEF) данных пользователя

Права доступа (APR... / APW...) могут быть определены для следующих данных:

- Глобальные данные пользователя (GUD)

Переопределение (REDEF) системных и пользовательских данных

Права доступа (APR... / APW...) могут быть переопределены для следующих данных:

- Системные данные
  - Машинные данные

    Примечание

    Переопределение прав чтения машинных данных

    Степень защиты для чтения машинных данных можно установить только с кодовым словом APR совместно для программы обработки детали и BTSS.

    При переопределении прав чтения кодовые слова APRP и APRB не поддерживаются и вызывают вывод аварийного сообщения 12490 "Право доступа APRP/APRB <уровень защиты> не настроено".
  - Установочные данные
  - Системная переменная
  - Данные процесса
  - Данные магазина
  - Данные инструмента
- Данные пользователя
  - R-параметры
  - Переменные синхронных действий ($AC_MARKER, $AC_PARAM, $AC_TIMER)
  - GUD синхронных действий (SYG_xy[ ], где x=R, I, B, A, C, S и y=S, M, U, 4, ..., 9)
  - Параметры EPS
  - Данные инструмента OEM
  - Данные магазина OEM
  - Глобальные переменные пользователя (GUD)

    Примечание

    При переопределении право доступа к переменной между самой низкой степенью защиты 7 и собственной степенью защиты, например, 1 (изготовитель станка) может присваиваться свободно.

Переопределение (REDEF) языковых команд ЧПУ

Право доступа и выполнения (APX) может быть переопределено для следующих языковых команд ЧПУ:

- G-команды / функции перемещения (Страница 177)
- Предопределенные функции (Страница 1350)
- Предопределенные вызовы подпрограмм

- Оператор DO для синхронных действий
- Программные идентификаторы циклов
  Цикл должен находиться в директории циклов и содержать оператор PROC.

Права доступа касательно программ ЧПУ и циклов (APRP, APWP)

Следствиями различных прав доступа для доступа в программе ЧПУ или цикле являются:

- APRP 0 / APWP 0
  - При выполнении программы ЧПУ должен быть установлен системный пароль
  - Цикл должен находиться в директории _N_CST_DIR (система)
  - Для директории _N_CST_DIR в MD11160 $MN_ACCESS_EXEC_CST право выполнения должно быть установлено на систему
- APRP 1 / APWP 1 или APRP 2 / APWP 2
  - При выполнении программы ЧПУ должен быть установлен пароль изготовителя станка или сервисный пароль
  - Цикл должен находиться в директории _N_CMA_DIR (изготовитель станка) или _N_CST_DIR
  - Для директорий _N_CMA_DIR или _N_CST_DIR в машинных данных MD11161 $MN_ACCESS_EXEC_CMA или MD11160 $MN_ACCESS_EXEC_CST права выполнения должны быть установлены как минимум на изготовителя станка
- APRP 3 / APWP 3
  - При выполнении программы ЧПУ должен быть установлен пароль пользователя
  - Цикл должен находиться в директории _N_CUS_DIR (пользователь) или _N_CST_DIR
  - Для директорий _N_CUS_DIR, _N_CMA_DIR или _N_CST_DIR в машинных данных MD11162 $MN_ACCESS_EXEC_CUS, MD11161 $MN_ACCESS_EXEC_CMA или MD11160 $MN_ACCESS_EXEC_CST права выполнения должны быть установлены как минимум на пользователя
- APRP 4...7 / APWP 4...7
  - При выполнении программы ЧПУ должно быть установлено положение кодового переключателя 3 ... 0
  - Цикл должен находиться в директории _N_CUS_DIR, _N_CMA_DIR или _N_CST_DIR
  - для директорий _N_CUS_DIR или _N_CMA_DIR или _N_CST_DIR в машинных данных MD11162 $MN_ACCESS_EXEC_CUS, MD11161 $MN_ACCESS_EXEC_CMA или MD11160 $MN_ACCESS_EXEC_CST права выполнения должны быть установлены как минимум на соответствующее положение кодового переключателя

Права доступа касательно BTSS (APRB, APWB)

Права доступа (APRB, APWB) ограничивают в равной мере доступ к системным переменным и переменным пользователя через BTSS для всех системных компонентов (HMI, PLC, внешние процессоры, службы EPS и т. д.).

Примечание

Локальные права доступа HMI

При изменениях прав доступа к системным данным необходимо следить за тем, чтобы это не входило в противоречие с установленными через механизмы HMI правами доступа.

Атрибуты доступа APR / APW

Из соображений совместимости атрибуты APR и APW не явно отображаются на атрибуты APRP / APRB и APWP / APWB:

- APR x ⇒ APRP x APRB x
- APW y ⇒ APWP y APWB y

Права доступа через файлы ACCESS

При использовании файлов ACCESS для присваивания прав доступа, переопределения прав доступа для системных данных, данных пользователя и языковых команд ЧПУ могут осуществляться исключительно в файлах ACCESS. Исключением являются глобальные данные пользователя (GUD). Для них переопределение прав доступа должно по-прежнему выполняться в соответствующих файлах определений *_DEF.

Для сквозной защиты доступа необходимо непротиворечивое согласование машинных данных для прав выполнения и защиты доступа соответствующих директорий.

Общий принцип действий:

1. Создание требуемых файлов определений:
   - _N_DEF_DIR/_N_SACCESS_DEF
   - _N_DEF_DIR/_N_MACCESS_DEF
   - _N_DEF_DIR/_N_UACCESS_DEF
2. Параметрирование права записи для файлов определений на требуемое для переопределения значение:
   - MD11170 $MN_ACCESS_WRITE_SACCESS = <уровень защиты>
   - MD11171 $MN_ACCESS_WRITE_MACCESS = <уровень защиты>
   - MD11172 $MN_ACCESS_WRITE_UACCESS = <уровень защиты>

3. Для доступа к защищенным элементам из циклов, необходимо согласовать права выполнения и записи директорий циклов _N_CST_DIR, _N_CMA_DIR и _N_CST_DIR:
   Права выполнения
   - MD11160 $MN_ACCESS_EXEC_CST = <уровень защиты>
   - MD11161 $MN_ACCESS_EXEC_CMA = <уровень защиты>
   - MD11162 $MN_ACCESS_EXEC_CUS = <уровень защиты>

   Права записи
   - MD11165 $MN_ACCESS_WRITE_CST = <уровень защиты>
   - MD11166 $MN_ACCESS_WRITE_CMA = <уровень защиты>
   - MD11167 MN_ACCESS_WRITE_CUS = <уровень защиты>

   Право выполнения должно быть установлен, на, по меньшей мере, ту же степень защиты, что и макс. степень защиты используемого элемента.
   Право записи как минимум должно быть установлено на ту же степень защиты, что и право выполнения.
4. Права записи локальных директорий циклов HMI должны быть установлены на ту же степень защиты, что и у локальных директорий циклов ЧПУ.

Вызовы подпрограмм в файлах ACCESS

Для дальнейшего структурирования защиты доступа, в файлах ACCESS возможен вызов и подпрограмм (идентификатор SPF или MPF). При этом подпрограммы получают права выполнения вызывающего файла ACCESS.

Примечание

В файлах ACCESS возможно переопределение только прав доступа. Все другие атрибуты, как и прежде, должны программироваться или переопределяться в соответствующих файлах определений.

См. также

Переменные (Страница 399)

4.1.1.11 Обзор определяемых и переопределяемых атрибутов

Таблицы ниже показывают, для каких типов данных какие атрибуты могут быть определены (DEF) и/или переопределены (REDEF).

Системные данные

Тип данных	Иниц.значение	Предельные значения	Физ. единица	Права доступа	Класс данных (только 828D)
Машинные данные	---	---	---	REDEF	REDEF
Установочные данные	REDEF	---	---	REDEF	---
FRAME-данные	---	---	---	REDEF	---

Тип данных	Иниц.значение	Предельные значения	Физ. единица	Права доступа	Класс данных (только 828D)
Данные процесса	---	---	---	REDEF	---
Комп. погрешности ходового винта (EEC)	---	---	---	REDEF	---
Компенсация провисания (CEC)	---	---	---	REDEF	---
Компенсация квадрантных ошибок (QEC)	---	---	---	REDEF	---
Данные магазина	---	---	---	REDEF	---
Данные инструмента	---	---	---	REDEF	---
Защищенные области	---	---	---	REDEF	---
Ориентируемые инструментальные суппорта	---	---	---	REDEF	---
Кинематические цепочки	---	---	---	REDEF	---
Защищенные области 3D	---	---	---	REDEF	---
Ограничение рабочего поля	---	---	---	REDEF	---

Данные пользователя

Тип данных	Иниц.значение	Предельные значения	Физ. единица	Права доступа	Класс данных
R-параметры	REDEF	REDEF	REDEF	REDEF	---
Переменные синхронных действий ($AC_...)	REDEF	REDEF	REDEF	REDEF	---
GUD синхронных действий ((SYG_...)	REDEF	REDEF	REDEF	REDEF	---
Параметры EPS	REDEF	REDEF	REDEF	REDEF	---
Данные инструмента OEM	REDEF	REDEF	REDEF	REDEF	---
Данные магазина OEM	REDEF	REDEF	REDEF	REDEF	---
Глобальные переменные пользователя (GUD)	DEF / REDEF	DEF	DEF	DEF / REDEF	DEF / REDEF
Локальные переменные пользователя (PUD / LUD)	DEF	DEF	DEF	---	---

См. также

Переменные (Страница 399)

4.1.1.12 Определение и инициализация переменных поля (DEF, SET, REP)

Переменная пользователя может быть определена как 1- до макс. 3-мерное поле (массив):

- 1-мерное: `DEF` <тип данных> <имя переменной>[<n>]
- 2-мерное: `DEF` <тип данных> <имя переменной>[<n>,<m>]
- 3-мерное: `DEF` <тип данных> <имя переменной>[<n>,<m>,<o>]

Примечание

Переменные пользователя типа данных STRING могут быть определены макс. как 2-мерное поле.

Типы данных

Переменные пользователя могут быть определены как поля для следующих типов данных: BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME

Присваивание значения элементам поля

Значения элементам поля могут присваиваться в следующие моменты времени:

- при определении поля (инициализирующие значения)
- при выполнении программы

При этом значения могут присваиваться через:

- явное указание элемента поля
- явное указание элемента поля как стартового элемента и указание списка значений (`SET`)
- явное указание элемента поля как стартового элемента и указание значения и частоты его повторения (`REP`)

Примечание

Переменным пользователя типа данных FRAME не могут быть присвоены инициализирующие значения.

Синтаксис (`DEF`)

```
DEF <тип данных> <имя переменной>[<n>,<m>,<o>]
DEF STRING[<длина строки>] <имя переменной>[<n>,<m>]
```

Синтаксис (DEF...=SET...)

Использование списка значений:

- при определении:
  DEF <тип данных> <имя переменной>[<n>,<m>,<o>] =
  SET(<значение1>,<значение2>,...)
  равнозначно:
  DEF <тип данных> <имя переменной>[<n>,<m>,<o>] =
  (<значение1>,<значение2>,...)

  Примечание

  При инициализации через список значений указание SET является опциональным.

- при присваивании значения:
  <имя переменной>[<n>,<m>,<o>]=SET(<ЗНАЧЕНИЕ1>,<значение2>,...)

Синтаксис (DEF...=REP...)

Использование значения с повторением

- при определении:
  DEF <тип данных> <имя переменной>[<n>,<m>,<o>]=REP(<значение>)
  DEF <тип данных> <имя переменной>[<n>,<m>,<o>] = REP(<значение><число элементов поля>)
- при присваивании значения:
  <имя переменной>[<n>,<m>,<o>]=REP(<значение>)
  <имя переменной>[<n>,<m>,<o>]=REP(<значение>,<число элементов поля>)

Значение

DEF:	Команда для определения переменных	
<тип данных>:	Тип данных переменных	
	Диапазон значений: • Для системных переменных: BOOL, CHAR, INT, REAL, STRING, AXIS • Для переменных GUD или LUD: BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME	
<длина строки>:	Макс. число символов для типа данных STRING	
<имя переменной>:	Имя переменной	
[<n>,<m>,<o>]:	Размеры поля или индексы поля	
<n>:	Размер поля или индекс поля для 1-й размерности	
	Тип:	INT (для системных переменных также AXIS)
	Диапазон значений:	Макс. размер поля: 65535 Индекс поля: 0 ≤ n ≤ 65534
<m>:	Размер поля или индекс поля для 2-й размерности	
	Тип:	INT (для системных переменных также AXIS)
	Диапазон значений:	Макс. размер поля: 65535 Индекс поля: 0 ≤ m ≤ 65534

<o>:	Размер поля или индекс поля для 3-й Размерность	
	Тип:	INT (для системных переменных также AXIS)
	Диапазон значений:	Макс. размер поля: 65535 Индекс поля: 0 ≤ o ≤ 65534
SET:	Присваивание значения через указанный список значений	
(<значение1>,<значение2>,...):	Список значений	
REP:	Присваивание значения через указанное <значение>	
<значение>:	Значение, с которым элементы поля должны записываться при инициализации с REP.	
<число элементов поля>:	Число элементов поля, которые должны записываться с указанным <значение>. Для оставшихся элементов поля, в зависимости от момента времени, действует: • Инициализация при определении поля: → Оставшиеся элементы поля записываются с нулем • Присваивание при выполнении программы: → Актуальные значения элементов поля остаются без изменений. Если параметр не запрограммирован, то все элементы поля записываются с <значение>. Если параметр равен нулю, то в зависимости от момента времени действует: • Инициализация при определении поля: → Все элементы предустанавливаются на ноль • Присваивание при выполнении программы: → Актуальные значения элементов поля остаются без изменений.	

Индекс поля

Не явная последовательность элементов поля, к примеру, при присваивании значения через SET или REP, достигается через итерацию индексов поля справа налево.

Пример: Инициализация 3-мерного поля с 24 элементами поля:

```
DEF INT FELD[2,3,4] = REP(1,24)
  FELD[0,0,0] = 1      1. элемент поля
  FELD[0,0,1] = 1      2. элемент поля
  FELD[0,0,2] = 1      3. элемент поля
  FELD[0,0,3] = 1      4. элемент поля
  ...
  FELD[0,1,0] = 1      5. элемент поля
  FELD[0,1,1] = 1      6. элемент поля
  ...
  FELD[0,2,3] = 1      12. элемент поля
  FELD[1,0,0] = 1      13. элемент поля
  FELD[1,0,1] = 1      14. элемент поля
```

```
  ...
  FELD[1,2,3] = 1      24. элемент поля
```

соответственно:

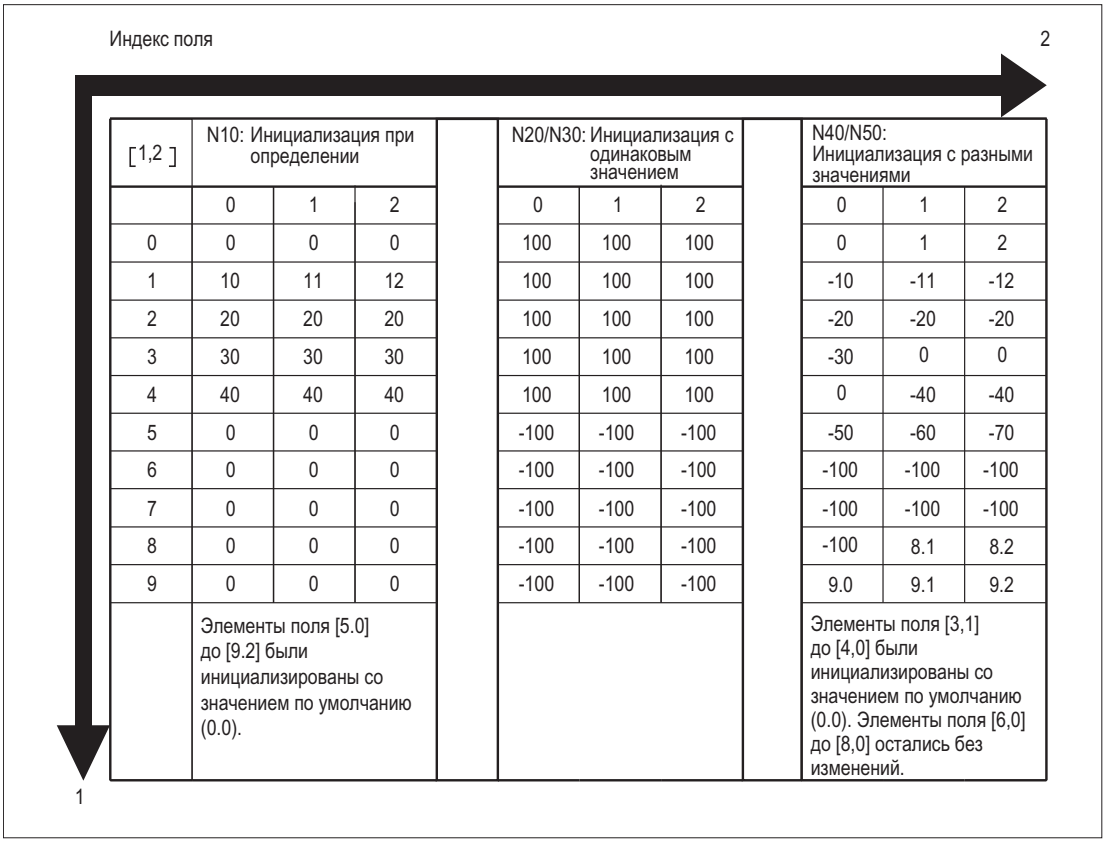
```
FOR n=0 TO 1
  FOR m=0 TO 2
     FOR o=0 TO 3
       FELD[n,m,o] = 1
     ENDFOR
  ENDFOR
ENDFOR
```

Пример: Инициализация целых полей переменных

Актуальные значения см. рисунок.

Программный код

```
N10 DEF REAL FELD1[10,3]=SET(0,0,0,10,11,12,20,20,20,30,30,30,40,40,40,)
N20 FELD1[0,0]=REP(100)
N30 FELD1[5,0]=REP(-100)
N40 FELD1[0,0]=SET(0,1,2,-10,-11,-12,-20,-20,-20,-30, , , ,-40,-40,-50,-60,-70)
N50 FELD1[8,1]=SET(8.1,8.2,9.0,9.1,9.2)
```



[1,2]	N10: Инициализация при определении			N20/N30: Инициализация с одинаковым значением			N40/N50: Инициализация с разными значениями		
	0	1	2	0	1	2	0	1	2
0	0	0	0	100	100	100	0	1	2
1	10	11	12	100	100	100	-10	-11	-12
2	20	20	20	100	100	100	-20	-20	-20
3	30	30	30	100	100	100	-30	0	0
4	40	40	40	100	100	100	0	-40	-40
5	0	0	0	-100	-100	-100	-50	-60	-70
6	0	0	0	-100	-100	-100	-100	-100	-100
7	0	0	0	-100	-100	-100	-100	-100	-100
8	0	0	0	-100	-100	-100	-100	8.1	8.2
9	0	0	0	-100	-100	-100	9.0	9.1	9.2
	Элементы поля [5.0] до [9.2] были инициализированы со значением по умолчанию (0.0).						Элементы поля [3,1] до [4,0] были инициализированы со значением по умолчанию (0.0). Элементы поля [6,0] до [8,0] остались без изменений.		

См. также

Определение и инициализация переменных поля (DEF, SET, REP) (Страница 431)

Переменные (Страница 399)

Дополнительная информация (SET)

Инициализация при определении

- Начиная с 1-го элемента поля, со значениями из списка инициализируется столько элементов поля, сколько элементов запрограммировано в списке значений.
- Элементам поля без явно указанных значений в списке значений (пропуски в списке значений) присваивается 0.
- Для переменных типа данных AXIS пропуски в списке значений запрещены.
- Если список значений содержит больше значений, чем определено элементов поля, то отображается ошибка.

Присваивание значения при выполнении программы

При присваивании значения при выполнении программы действуют описанные выше для определения правила. Дополнительно существуют следующие возможности:

- В качестве элементов в списке значений разрешены и выражения.
- Присваивание значения начинается на запрограммированном индексе поля. Благодаря этому возможно целенаправленное присваивание значений подполям.

Пример:

Программный код	Комментарий
`DEF INT FELD[5,5]`	; Определение поля
`FELD[0,0]=SET(1,2,3,4,5)`	; Присваивание значения первым 5 элементам поля [0,0] - [0,4]
`FELD[0,0]=SET(1,2, , ,5)`	; Присваивание значения с пропуском первым 5 элементам поля [0,0] - [0,4], элементам поля [0,2] и [0,3] = 0

`FELD[2,3]=SET(VARIABLE,4*5.6)`	; Присваивание значения с переменной и выражением от индекса поля [2,3]:
	`[2,3] = VARIABLE`
	`[2,4] = 4 * 5.6 = 22.4`

Дополнительная информация (REP)

Инициализация при определении

- Все или указанное опционально число элементов поля инициализируются с указанным значением (постоянная).
- Переменные типа данных FRAME не могут быть инициализированы.

Пример:

Программный код	Комментарий
`DEF REAL varName[10]=REP(3.5,4)`	; Определение поля и инициализация элементов поля [0] до [3] со значением 3,5

Присваивание значения при выполнении программы

При присваивании значения при выполнении программы действуют описанные выше для определения правила. Дополнительно существуют следующие возможности:

- В качестве элементов в списке значений разрешены и выражения.
- Присваивание значения начинается на запрограммированном индексе поля. Благодаря этому возможно целенаправленное присваивание значений подполям.

Примеры:

Программный код	Комментарий
`DEF REAL varName[10]`	; Определение поля
`varName[5]=REP(4.5,3)`	; Элементы поля от [5] до [7] = 4,5
`R10=REP(2.4,3)`	`;` R-параметры от `R10` до `R12 = 2,4`
`DEF FRAME FRM[10]`	; Определение поля
`FRM[5]=REP(CTRANS(X,5))`	; Элементы поля от [5] до [`9] = CTRANS(X,5)`

См. также

Определение и инициализация переменных поля (DEF, SET, REP) (Страница 431)

4.1.1.13 Типы данных

Следующие типы данных имеются в ЧПУ:

Тип данных	Значение	Диапазон значений
INT	Целочисленное значение со знаком	-2147483648 ... +2147483647
REAL	Действительное число (LONG REAL по IEEE)	±(~2,2*10^{-308} ... ~1,8*10^{+308})
BOOL	Значение истинности TRUE (1) и FALSE (0)	1, 0
CHAR	Символы ASCII	Код ASCII 0 ... 255
STRING	Строка символов определенной длины	Макс. 200 символов (без специальных символов)
AXIS	Идентификатор оси/шпинделя	Идентификатор оси канала
FRAME	Геометрические данные для статической трансформации координат (смещения, вращения, масштабирования, отражения)	---

Не явные преобразования типов данных

Следующие преобразования типов данных возможны и не явно выполняются при присваиваниях и передачах параметров:

из ↓/ в →	REAL	INT	BOOL
REAL	x	o	&
INT	x	x	&
BOOL	x	x	x
x: возможно без ограничений O: возможна потеря данных из-за превышения диапазона значений ⇒ ошибка; округление: значение после запятой ≥ 0,5 ⇒ округлять в сторону увеличения, значение после запятой < 0,5 ⇒ округлять в сторону уменьшения &: значение ≠ 0 ⇒ TRUE, значение == 0 ⇒ FALSE			

См. также

Переменные (Страница 399)

4.1.1.14 Минимум, максимум и диапазон переменных (MINVAL, MAXVAL и BOUND)

С помощью команд `MINVAL` и `MAXVAL` можно сравнить между собой значения двух переменных. Как результат возвращается меньшее значение (при `MINVAL`) или большее значение (при `MAXVAL`).

С помощью функции `BOUND` можно проверить, лежит ли значение контрольной переменной в пределах определенного диапазона значений.

Синтаксис

<меньшее значение>=MINVAL(<переменная1>,<переменная2>)
<большее значение>=MAXVAL(<переменная1>,<переменная2>)
<возвращаемое значение>=<BOUND>(<минимум>,<максимум>,<контрольная переменная>)

Значение

MINVAL:	Определяет **меньшее** значение двух переменных (<переменная1>, <переменная2>)
<меньшее значение>:	Переменная результата для команды MINVAL Устанавливается на меньшее значение переменной.
MAXVAL:	Определяет **большее** значение двух переменных (<переменная1>, <переменная2>)
<большее значение>:	Переменная результата для команды MAXVAL Устанавливается на большее значение переменной.
BOUND:	Проверяет, лежит ли переменная (<контрольная переменная>) в пределах определенного диапазона значений.
<минимум>:	Переменная, определяющая мин. значение диапазона значений
<максимум>:	Переменная, определяющая макс. значение диапазона значений
<возвращаемое значение>:	Переменная результата для команды BOUND Если значение контрольной переменной лежит в пределах определенного диапазона значений, то переменная результата устанавливается на значение контрольной переменной. Если значение контрольной переменной больше макс. значения, то переменная результата устанавливается на макс. значение области определения. Если значение контрольной переменной меньше мин. значения, то переменная результата устанавливается на мин. значение области определения.

Примечание

MINVAL, MAXVAL и BOUND могут программироваться и в синхронных действиях.

Примечание

Поведение при равенстве

При равенстве для MINVAL/MAXVAL выводится это равное значение. При BOUND снова возвращается значение проверяемой переменной.

Пример

Программный код	Комментарий
`DEF REAL rVar1=10.5, rVar2=33.7, rVar3, rVar4, rVar5, rValMin, rValMax, rRetVar`	
`rValMin=MINVAL(rVar1,rVar2)`	`; rValMin` устанавливается на значение 10.5.
`rValMax=MAXVAL(rVar1,rVar2)`	`; rValMax` устанавливается на значение 33.7.

Программный код	Комментарий
`rVar3=19.7`	
`rRetVar=BOUND(rVar1,rVar2,rVar3)`	`;` `rVar3` лежит в пределах границ, `rRetVar` устанавливается на 19.7.
`rVar3=1.8`	
`rRetVar=BOUND(rVar1,rVar2,rVar3)`	`;` `rVar3` лежит ниже мин. границы, `rRetVar` устанавливается на 10.5.
`rVar3=45.2`	
`rRetVar=BOUND(rVar1,rVar2,rVar3)`	`;` `rVar3` лежит выше макс. границы, `rRetVar` устанавливается на 33.7.

4.1.1.15 Проверка наличия переменной (ISVAR)

С помощью предварительно определенной функции ISVAR можно проверить, известна ли в ЧПУ системная переменная / переменная пользователя (например, машинные данные, установочные данные, системная переменная, общая переменная, например, GUD).

Переменная

Запрашиваемая переменная может иметь следующее строение:

Безразмерная переменная:	<переменная>
Одноразмерная переменная без индекса поля:	<переменная>[ ]
Одноразмерная переменная с индексом поля n:	<переменная>[<n>]
Двухразмерная переменная без индекса поля:	<переменная>[,]
Двухразмерная переменная с индексами поля n и m:	<переменная>[<n>,<m>]

Синтаксис

<результат>=`ISVAR(`<переменная>[`<n>,<m>])`

Объяснение

<результат>:	Возвращаемое значение		
	Тип данных:	BOOL	
	Диапазон значений:	1	Имеется переменная
		0	Неизвестная переменная
`ISVAR`:	Проверяет, известна ли в ЧПУ указанная системная переменная / переменная пользователя.		
<переменная>:	Имя системной переменной / переменной пользователя		
	Тип данных:	STRING	
<n>:	Индекс поля первой размерности (**опция**)		
	Тип данных:	INT	

<m>:	Индекс поля второй размерности (**опция**)	
	Тип данных:	INT

Согласно передаваемому параметру, выполняются следующие проверки:

- Известно ли имя;
- Является ли переменная полем,
- Это одно- или двухразмерное поле
- Находится ли соответствующий индекс поля в разрешенном диапазоне.

Если все эти проверки положительные, возвращается TRUE (1).

Если одна из проверок отрицательна, или если обнаруживается синтаксическая ошибка, возвращается FALSE (0).

Примеры

Программный код	Комментарий
`DEF INT VAR1`	
`DEF BOOL IS_VAR=FALSE`	
`N10 IS_VAR=ISVAR("VAR1")`	`; IS_VAR` в этом случае `TRUE`.

Программный код	Комментарий
`DEF REAL VARARRAY[10,10]`	
`DEF BOOL IS_VAR=FALSE`	
`N10 IS_VAR=ISVAR("VARARRAY[,]")`	`; IS_VAR` в этом случае `TRUE`, это двухразмерное поле.
`N20 IS_VAR=ISVAR("VARARRAY")`	`; IS_VAR = TRUE`, переменная существует.
`N30 IS_VAR=ISVAR("VARARRAY[8,11]")`	`; IS_VAR = FALSE`, индекс поля запрещен.
`N40 IS_VAR=ISVAR("VARARRAY[8,8")`	`; IS_VAR = FALSE, "]"` отсутствует (синтаксическая ошибка).
`N50 IS_VAR=ISVAR("VARARRAY[,8]")`	`; IS_VAR = TRUE`, индекс поля разрешен.
`N60 IS_VAR=ISVAR("VARARRAY[8,]")`	`; IS_VAR = TRUE`, индекс поля разрешен.

Программный код	Комментарий
`DEF BOOL IS_VAR=FALSE`	
`N100 IS_VAR=ISVAR("$MC_GCODE_RESET_VALUES[1]"`	; Передаваемый параметр это машинные данные, `IS_VAR = TRUE`.

Программный код	Комментарий
`DEF BOOL IS_VAR=FALSE`	
`N10 IS_VAR=ISVAR("$P_EP")`	`; IS_VAR` в этом случае `TRUE`.
`N20 IS_VAR=ISVAR("$P_EP[X]")`	`; IS_VAR` в этом случае `TRUE`.

4.1.1.16 Чтение значений атрибута / типа данных (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP)

С помощью предопределенных функций GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM и GETVARDFT можно считывать значения атрибутов системных переменных/переменных пользователя, с помощью GETVARTYP можно считывать тип данных системной переменной/переменной пользователя.

Чтение физической единицы

Синтаксис:
<результат>=GETVARPHU(<имя>)

Объяснение:

<результат>:	Числовое значение физической единицы		
	Тип данных:	INT	
	Диапазон значений:	см. таблицу в "Атрибут: Физическая единица (PHU) (Страница 422)"	
		В случае ошибки:	
		- 2	Указанное имя переменной не присвоено никакому системному параметру и никакой переменной пользователя.
GETVARPHU:	Чтение физической единицы системной переменной/переменной пользователя		
<имя>:	Имя системной переменной/переменной пользователя		
	Тип данных:	STRING	

Пример:

ЧПУ содержит следующую GUD-переменную:

DEF CHAN REAL PHU 42 LLI 0 ULI 10000 electric

Программный код	Комментарий
`DEF INT result=0`	
`result=GETVARPHU("electric")`	; Определить физическую единицу GUD-переменной.
`IF (result < 0) GOTOF error`	

Как результат возвращается значение 42. Это соответствует физической единице [кВт].

Примечание

С GETVARPHU можно, например, проверить, имеют при присваивании переменной a = b обе переменные ожидаемые физические единицы.

Чтение права доступа

Синтаксис:
<результат>=GETVARAP(<имя>,<доступ>)

Объяснение:

<результат>:	Степень защиты для указанного <доступа>		
	Тип данных:	INT	
	Диапазон значений:	0 ... 7	См. "Атрибут: Права доступа (APR, APW, APRP, APWP, APRB, APWB) (Страница 425)".
		В случае ошибки:	
		- 1	не поддается записи (имеет значение только для типов доступа "WP" и "WB")
		- 2	Указанное имя переменной не присвоено никакому системному параметру и никакой переменной пользователя.
		- 3	неправильное значение параметра <доступ>
GETVARAP:	Чтение права доступа к системной переменной/переменной пользователя		
<имя>:	Имя системной переменной/переменной пользователя		
	Тип данных:	STRING	
<доступ>:	Тип доступа		
	Тип данных:	STRING	
	Диапазон значений:	"RP"	Чтение через программу обработки детали
		"WP"	Запись через программу обработки детали
		"RB"	Чтение через BTSS
		"WB"	Запись через BTSS

Пример:

Программный код	Комментарий
`DEF INT result=0`	
`result=GETVARAP("$TC_MAP8","WB")`	; Определить защиту доступа для системного параметра "Позиция магазина" касательно записи через BTSS.
`IF (result < 0) GOTOF error`	

Как результат возвращается значение 7. Это соответствует положению кодового переключателя 0 (= нет защиты доступа).

Примечание

С GETVARAP может быть, например, реализована проверочная программа, проверяющая ожидаемые от приложения права доступа.

Чтение предельных значений

Синтаксис:
<состояние>=GETVARLIM(<имя>,<предельное значение>,<результат>)

Объяснение:

<состояние>:	Состояние функции		
	Тип данных:	INT	
	Диапазон значений:	1	o. k.
		-1	Предельное значение не определено (для переменных типа AXIS, STRING, FRAME)
		-2	Указанное имя переменной не присвоено никакому системному параметру и никакой переменной пользователя.
		-3	Неправильное значение параметра <предельное значение>
GETVARLIM:	Чтение нижнего/верхнего предельного значения системной переменной/переменной пользователя		
<имя>:	Имя системной переменной/переменной пользователя		
	Тип данных:	STRING	
<предельное значение>:	Указывает, какое предельное значение должно быть выгружено		
	Тип данных:	CHAR	
	Диапазон значений:	"L"	Нижнее предельное значение
		"U"	Верхнее предельное значение
<результат>:	Возврат предельного значения		
	Тип данных:	VAR REAL	

Пример:

Программный код	Комментарий
`DEF INT state=0`	
`DEF REAL result=0`	
`state=GETVARLIM("$MA_MAX_AX_VELO","L",result)`	; Определить нижнее предельное значение для `MD32000 $MA_MAX_AX_VELO`.
`IF (result < 0) GOTOF error`	

Считать значение размерности

Синтаксис:
<результат>=GETVARDIM(<имя>,<индекс>)

Объяснение:

<результат>:	Размер указанной в параметре <индекс> размерности	
	Тип данных:	INT
GETVARDIM:	Чтение размера 1-й, 2-й или 3-й размерности поля системной переменной/переменной пользователя	

<имя>:	Имя системной переменной/переменной пользователя		
	Тип данных:	STRING	
<индекс>:	Индекс поля		
	Тип данных:	INT	
	Диапазон значений:	1 ... 3	
		1	Индекс для 1-й размерности поля
		2	Индекс для 2-й размерности поля
		3	Индекс для 3-й размерности поля

Пример:

Программный код	**Комментарий**
`N5 DEF REAL myReal[5,4]`	
`N10 R1=GETVARDIM("myReal",1)`	; Определение размера 1-й размерности поля
	; результат: `R1 = 5`
`N15 R2=GETVARDIM("myReal",2)`	; Определение размера 2-й размерности поля
	; результат: `R2 = 4`

Чтение стандартного значения

Синтаксис:
<состояние>=`GETVARDFT(`<имя>,<результат>[,<индекс_1>,<индекс_2>, <индекс_3>])

Объяснение:

<состояние>:	Состояние функции		
	Тип данных:	INT	
	Диапазон значений:	1	o. k.
		-1	Стандартное значение недоступно (например, т. к. определение типа данных переменной результата не подходит для типа данных системной переменной/переменной пользователя)
		-2	Указанное имя переменной не присвоено никакому системному параметру и никакой переменной пользователя.
		-3	Неправильное значение параметра <индекс_1>, размерность меньше единицы (⇒ не поле, а скалярная переменная)
		-4	Неправильное значение параметра <индекс_2>
		-5	Неправильное значение параметра <индекс_3>
`GETVARDFT`:	Чтение стандартного значения системной переменной/переменной пользователя		
<имя>:	Имя системной переменной/переменной пользователя		
	Тип данных:	STRING	

<результат>:	Возврат стандартного значения	
	Тип данных:	VAR REAL (при чтении стандартного значения переменных типов INT, REAL, BOOL, AXIS)
		VAR STRING (при чтении стандартного значения переменных типов STRING и CHAR)
		VAR FRAME (при чтении стандартного значения переменных типа FRAME)
<индекс_1>:	Индекс для 1-й размерности поля (опционально)	
	Тип данных:	INT
	Не запрограммировано означает = 0	
<индекс_2>:	Индекс для 2-й размерности поля (опционально)	
	Тип данных:	INT
	Не запрограммировано означает = 0	
<индекс_3>:	Индекс для 3-й размерности поля (опционально)	
	Тип данных:	INT
	Не запрограммировано означает = 0	

Пример:

Программный код	**Комментарий**
`DEF INT state=0`	
`DEF REAL resultR=0`	; Переменная для записи стандартных значений типов `INT, REAL, BOOL, AXIS.`
`DEF FRAME resultF=0`	; Переменная для записи стандартных значений типов `FRAME`
`IF (GETVARTYP("$MA_MAX_AX_VELO") <> 4) GOTOF error`	
`state=GETVARDFT("$MA_MAX_AX_VELO",resultR, AXTOINT(X))`	; Определить стандартное значение оси `"X".`
`IF (resultR < 0) GOTOF error`	
`IF (GETVARTYP("$TC_TP8") <> 3) GOTOF error`	
`state=GETVARDFT("$TC_TP8", resultR)`	
`IF (GETVARTYP("$P_UBFR")  <> 7) GOTOF error`	
`state=GETVARDFT("$P_UBFR", resultF )`	

Чтение типа данных

Синтаксис:
<результат>=`GETVARTYP(`<имя>)

Объяснение:

<результат>:	Тип данных указанной системной переменной/переменной пользователя		
	Тип данных:	INT	
	Диапазон значений:	1	= BOOL
		2	= CHAR
		3	= INT
		4	= REAL
		5	= STRING
		6	= AXIS
		7	= FRAME
		В случае ошибки:	
		< 0	Указанное имя переменной не присвоено никакому системному параметру и никакой переменной пользователя.
GETVARTYP:	Чтение типа данных системной переменной/переменной пользователя		
<имя>:	Имя системной переменной/переменной пользователя		
	Тип данных:	STRING	

Пример:

Программный код	Комментарий
`DEF INT result=0`	
`DEF STRING name="R"`	
`result=GETVARTYP(name)`	; Определить тип `R`-параметра.
`IF (result < 0) GOTOF error`	

Как результат возвращается значение 4. Это соответствует типу данных REAL.

4.1.1.17 Возможные преобразования типов

Постоянное числовое значение, переменная или выражение, присваиваемое переменной, должно быть совместимо с типом этой переменной. Если это так, то при присваивании тип преобразуется автоматически.

Возможные преобразования типов

к позиции	REAL	INT	BOOL	CHAR	STRING	AXIS	FRAME
из							
REAL	да	да*	да[1])	да*	–	–	–
INT	да	да	да[1])	да[2])	–	–	–
BOOL	да	да	да	да	да	–	–
CHAR	да	да	да[1])	да	да	–	–
STRING	–	–	да[4])	да[3])	да	–	–
AXIS	–	–	–	–	–	да	–
FRAME	–	–	–	–	–	–	да

Пояснения

* При преобразовании типа из REAL в INT дробное значение >= 0,5 округляется в большую сторону, иные в меньшую сторону (см. функцию ROUND)

1) Значение <> 0 соответствует TRUE, значение == 0 соответствует FALSE

2) Если значение находится в допустимом числовом диапазоне

3) Если только 1 символ

4) Длина строки 0 = >FALSE, в иных случаях TRUE

Примечание

Если при преобразовании значение превышает конечный диапазон, то следует сообщение об ошибке.

Если в одном выражении встречаются различные типы, то согласование типов выполняется автоматически. Преобразования типов возможны и в синхронных действиях, см. главу "Синхронные действия движения, не явное преобразование типов".

4.1.2 Косвенное программирование

4.1.2.1 Косвенное программирование адресов

При косвенном программировании адресов (Страница 1289) расширенный адрес (<индекс>) заменяется на переменную подходящего типа.

Примечание

Косвенное программирование адресов невозможно для:

- N (номер кадра)
- L (подпрограмма)
- Настраиваемых адресов
  (к примеру, X[1] вместо X1 запрещено)

Синтаксис

<АДРЕС>[<индекс>]

Объяснение

<АДРЕС>[...]:	Фиксированный адрес с расширением (индекс)
<индекс>:	Переменная, к примеру, для номера шпинделя, оси, ...

Примеры

Пример 1: Косвенное программирование номера шпинделя

Прямое программирование:

Программный код	Комментарий
`S1=300`	; Число оборотов 300 об/мин для шпинделя с номером 1.

Косвенное программирование:

Программный код	Комментарий
`DEF INT SPINU=1`	; Определение переменной типа `INT` и присваивание значения.
`S[SPINU]=300`	; Число оборотов 300 об/мин для шпинделя, номер которого сохранен в переменной `SPINU` (в этом примере это шпиндель с номером 1).

Пример 2: Косвенное программирование оси

Прямое программирование:

Программный код	Комментарий
`FA[U]=300`	; Подача 300 для оси `"U"`.

Косвенное программирование:

Программный код	Комментарий
`DEF AXIS AXVAR2=U`	; Определение переменной типа `AXIS` и присваивание значения.
`FA[AXVAR2]=300`	; Подача 300 для оси , имя адреса которой сохранено в переменной с именем `AXVAR2`.

Пример 3: Косвенное программирование оси

Прямое программирование:

Программный код	Комментарий
`$AA_MM[X]`	; Чтение измеренного значения щупа (`MCS`) оси `"X"`.

Косвенное программирование:

Программный код	Комментарий
`DEF AXIS AXVAR3=X`	; Определение переменной типа `AXIS` и присваивание значения.
`$AA_MM[AXVAR3]`	; Чтение измеренного значения щупа (`MCS`) для оси, имя которой сохранено в переменной `AXVAR3`.

Пример 4: Косвенное программирование оси

Прямое программирование:

Программный код
`X1=100 X2=200`

Косвенное программирование:

Программный код	Комментарий
`DEF AXIS AXVAR1 AXVAR2`	; Определение двух переменных типа `AXIS`.

Программный код	Комментарий
`AXVAR1=(X1) AXVAR2=(X2)`	; Присваивание имен осей.
`AX[AXVAR1]=100 AX[AXVAR2]=200`	; Перемещение осей, имена адресов которых сохранены в переменных с именами `AXVAR1` и `AXVAR2`.

Пример 5: Косвенное программирование оси

Прямое программирование:

Программный код
`G2 X100 I20`

Косвенное программирование:

Программный код	Комментарий
`DEF AXIS AXVAR1=X`	; Определение переменной типа `AXIS` и присваивание значения.
`G2 X100 IP[AXVAR1]=20`	; Косвенное программирование указания центра для оси , имя адреса которой сохранено в переменной с именем `AXVAR1`

Пример 6: Косвенное программирование элементов поля

Прямое программирование:

Программный код	Комментарий
`DEF INT FELD1[4,5]`	; Определение поля 1.

Косвенное программирование:

Программный код	Комментарий
`DEFINE DIM1 AS 4`	; Для размерностей поля размеры поля должны указываться как фиксированные значения.
`DEFINE DIM2 AS 5`	
`DEF INT FELD[DIM1,DIM2]`	
`FELD[DIM1-1,DIM2-1]=5`	

Пример 7: Косвенный вызов подпрограммы

Программный код	Комментарий
`CALL "L" << R10`	; Вызов программы, номер которой стоит в `R10` (соединение строк).

4.1.2.2 Косвенное программирование G-команд

Косвенное программирование G-команд (Страница 1302) обеспечивает эффективное программирование циклов.

Синтаксис

`G[`<группа>`]=`<номер>

Объяснение

G[...]:	G-команда с расширением (индекс)	
<группа>:	Параметры индекса: G-группа	
	Тип:	INT
<номер>:	Переменная для номера G-команды	
	Тип:	INT или REAL

Примечание

Как правило, возможно косвенное программирование только не определяющих синтаксис G-команд.

Из определяющих синтаксис G-команд возможна только группа G-команд 1. Определяющие синтаксис G-команды групп G-команд 2, 3 и 4 невозможны.

Примечание

При косвенном программировании G-команды арифметические функции запрещены. Необходимое вычисление номера G-команды должно быть выполнено в отдельной строке программы обработки детали перед косвенным программированием в G-командах.

Примеры

Пример 1: Настраиваемое смещение нулевой точки (группа G-команд 8)

Программный код	Комментарий
N1010 DEF INT INT_VAR	
N1020 INT_VAR=2	
...	
N1090 G[8]=INT_VAR G1 X0 Y0	; G54
N1100 INT_VAR=INT_VAR+1	; Вычисление G-команды
N1110 G[8]=INT_VAR G1 X0 Y0	; G55

Пример 2: Выбор плоскости (группа G-команд 6)

Программный код	Комментарий
N2010 R10=$P_GG[6]	; Считывание активной G-команды группы G-команд 6
...	
N2090 G[6]=R10	

4.1.2.3 Косвенное программирование атрибутов позиций (GP)

Атрибуты позиций, например, инкрементальное или абсолютное программирование позиции оси, в комбинации с кодовым словом GP могут программироваться косвенно как переменные.

Использование

Косвенное программирование атрибутов позиций используется в **циклах замещения**, т. к. здесь существует следующее преимущество по сравнению с программированием атрибутов позиций как кодового слова (например, IC, AC, ...):

Благодаря косвенному программированию как переменной **не нужен** оператор CASE, который выполняет ветвление через все возможные атрибуты позиций.

Синтаксис

<КОМАНДА ПОЗИЦИОНИРОВАНИЯ>[<ось/шпиндель>]=
GP(<позиция>,<атрибут позиции>)
<ось/шпиндель>=GP(<позиция>,<атрибут позиции>)

Объяснение

<КОМАНДА ПОЗИЦИОНИРОВАНИЯ>[]:	Следующие команды позиционирования могут быть запрограммированы с кодовым словом GP: POS, POSA, SPOS, SPOSA Кроме этого возможны: • Все имеющиеся в канале идентификаторы осей/шпинделей: <ось/шпиндель> • Переменный идентификатор оси/шпинделя AX
<ось/шпиндель>:	Ось/шпиндель, которая должна быть позиционирована
GP():	Кодовое слово для позиционирования
<позиция>:	Параметр 1 Позиция оси/шпинделя как постоянная или переменная
<атрибут позиции>:	Параметр 2 Атрибут позиции (к примеру, режим подвода к позиции) как переменная (к примеру, $P_SUB_SPOSMODE) или как кодовое слово (IC, AC, ...)

Выводимые переменными величины имеют следующее значение:

Величина	**Объяснение**	**Допускается для:**
0	Нет изменений атрибута позиции	
1	AC	POS, POSA,SPOS, SPOSA,AX, адрес оси
2	IC	POS, POSA,SPOS, SPOSA,AX, адрес оси
3	DC	POS, POSA,SPOS, SPOSA,AX, адрес оси
4	ACP	POS, POSA,SPOS, SPOSA,AX, адрес оси
5	ACN	POS, POSA,SPOS, SPOSA,AX, адрес оси
6	OC	-
7	PC	-
8	DAC	POS, POSA,AX, адрес оси
9	DIC	POS, POSA,AX, адрес оси
10	RAC	POS, POSA,AX, адрес оси

Величина	Объяснение	Допускается для:
11	RIC	POS, POSA,AX, адрес оси
12	CAC	POS, POSA
13	CIC	POS, POSA
14	CDC	POS, POSA
15	CACP	POS, POSA
16	CACN	POS, POSA

Пример

При активном соединении синхронных шпинделей между ходовым винтом S1 и ведомым шпинделем S2 через команду SPOS в главной программе вызывается следующий цикл замещения для позиционирования шпинделей.

Позиционирование осуществляется через оператор в N2230:
```
SPOS[1]=GP($P_SUB_SPOSIT,$P_SUB_SPOSMODE)
SPOS[2]=GP($P_SUB_SPOSIT,$P_SUB_SPOSMODE)
```

Позиция подвода считывается из системной переменной $P_SUB_SPOSIT, режим подвода к позиции - из системной переменной $P_SUB_SPOSMODE.

Программный код	Комментарий
N1000 PROC LANG_SUB DISPLOF SBLOF	
...	
N2100 IF($P_SUB_AXFCT==2)	
N2110	; Замещение команды SPOS / SPOSA / M19 при активном соединении синхронных шпинделей
N2185 DELAYFSTON	; Начало области задержки останова
N2190 COUPOF(S2,S1)	; Деактивировать соединение синхронных шпинделей
N2200	; Позиционирование ведущего и ведомого шпинделя
N2210 IF($P_SUB_SPOS==TRUE) OR ($P_SUB_SPOSA==TRUE)	
N2220	; Позиционировать шпиндель со SPOS:
N2230 SPOS[1]=GP($P_SUB_SPOSIT, $P_SUB_SPOSMODE)	
SPOS[2]=GP($P_SUB_SPOSIT, $P_SUB_SPOSMODE)	
N2250 ELSE	
N2260	; Позиционировать шпиндель с M19:
N2270 M1=19 M2=19	; Позиционирование ведущего и ведомого шпинделя
N2280 ENDIF	
N2285 DELAYFSTOF	; Конец области задержки останова
N2290 COUPON(S2,S1)	; Активировать соединение синхронных шпинделей
N2410 ELSE	
N2420	; Опрос на предмет других замещений
...	

Программный код	Комментарий
N3300 ENDIF	
...	
N9999 RET	

Граничные условия

В синхронных действиях косвенное программирование атрибутов позиций невозможно.

4.1.2.4 Косвенное программирование строк программы обработки детали (EXECSTRING)

С помощью команды программы обработки детали EXECSTRING можно выполнить созданную прежде переменную String как строку программы обработки детали.

Синтаксис

EXECSTRING программируется в отдельной строке программы обработки детали:
EXECSTRING (<переменная String>)

Значение

EXECSTRING:	Команда для выполнения переменной String как строки программы обработки детали
<переменная String>:	Переменная типа STRING, содержащая саму исполняемую строку программы обработки детали

Примечание

С помощью EXECSTRING могут устанавливаться, за исключением управляющих структур (Страница 491), все конструкции программы обработки детали, которые могут быть запрограммированы **в программном блоке программы обработки детали**. Исключением являются операторы PROC и DEF, а также использование в файлах INI и DEF.

Пример

Программный код	Комментарий
N100 DEF STRING[100] MY_BLOCK	; Определение переменной String для записи исполняемой строки программы обработки детали.
N110 DEF STRING[10] MFCT1="M7"	
...	
N200 EXECSTRING(MFCT1 << "M4711")	; Выполнить строку программы обработки детали "M7 M4711".
...	
N300 R10=1	
N310 MY_BLOCK="M3"	

Программный код	Комментарий
N320 IF(R10)	
N330 MY_BLOCK = MY_BLOCK << MFCT1	
N340 ENDIF	
N350 EXECSTRING(MY_BLOCK)	; Выполнить строку программы обработки детали "M3 M7".

4.1.3 Операции

4.1.3.1 Функции вычисления

Оператор / функция вычисления	Значение
+	Сложение
-	Вычитание
*	Умножение
/ [1]	Деление [1]
DIV [1]	Целочисленное деление [1]
MOD [1]	Деление по модулю (выдает остаток целочисленного деления) [1]
:	Связывающий оператор для переменных FRAME
SIN( )	Синус
COS()	Косинус
TAN()	Тангенс
ASIN()	Арксинус
ACOS()	Арккосинус
ATAN2(,) [1]	Арктангенс2 [1]
SQRT()	Квадратный корень
ABS()	Значение
POT()	2-я степень (квадрат)
TRUNC()	Целая часть Настраиваемая точность для команд сравнения с TRUNC (см. "Коррекция точности при ошибках сравнения (TRUNC) (Страница 459)")
ROUND ()	Округление до целого
LN()	Натуральный логарифм
EXP()	Показательная функция
MINVAL()	Меньшее значение двух переменных (см. "Минимум, максимум и диапазон переменных (MINVAL, MAXVAL и BOUND) (Страница 437)")
MAXVAL()	Большее значение двух переменных (см. "Минимум, максимум и диапазон переменных (MINVAL, MAXVAL и BOUND) (Страница 437)")

BOUND()	Значение переменной, лежащее в определенном диапазоне значений (см. "Минимум, максимум и диапазон переменных (MINVAL, MAXVAL и BOUND) (Страница 437)")
CTRANS()	Смещение
CROT()	Вращение
CSCALE()	Изменение масштаба
CMIRROR()	Отражение
1) См. абзац "Примеры"	

Программирование

Для функций вычисления действует обычное математическое представление. Приоритеты в выполнении устанавливаются посредством круглых скобок. Для тригонометрических и их обратных функций действует указание градуса (прямой угол = 90°).

Примеры

Деление: /

(тип REAL) = тип INT или тип REAL) / (тип INT или тип REAL);

Пример: 3 / 4 = 0.75

Целочисленное деление: DIV

(тип INT) = (тип INT или REAL) / (тип INT или REAL);

Пример: 7 DIV 4.1 = 1

Деление по модулю (выдает остаток целочисленного деления): MOD

(тип REAL) = (тип INT или REAL) MOD (тип INT или REAL);

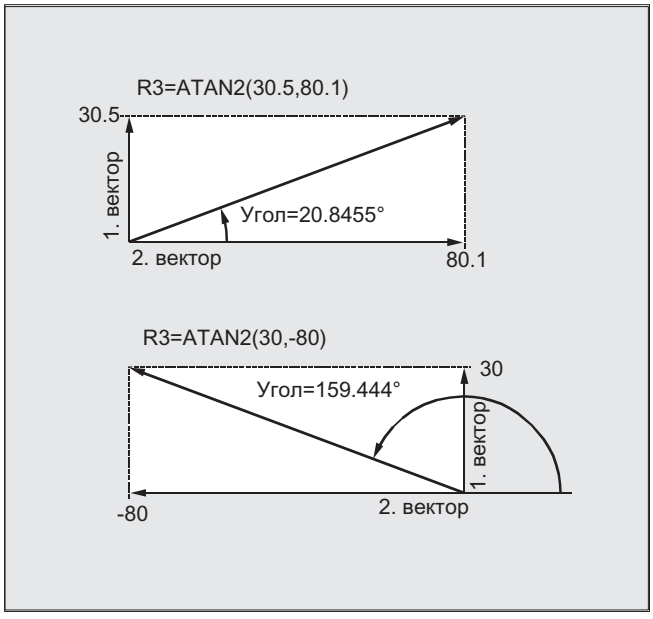
Пример: 7 MOD 4.1 = 2.9

Арктангенс2: ATAN2

Функция вычисляет ATAN2 из двух расположенных вертикально друг на друге векторов угол суммарного вектора.

Результат находится в диапазоне четырех квадрантов (-180° < 0 < +180°).

Основой для угловой корреляции всегда является 2-е значение в положительном направлении.



Примеры программирования

Программный код	Комментарий
`R1=R1+1`	; Новое `R1 = старое R1 + 1`
`R1=R2+R3 R4=R5-R6 R7=R8*R9`	
`R10=R11/R12 R13=SIN(25.3)`	
`R14=R1*R2+R3`	; Вычисление по правилам арифметики.
`R14=(R1+R2)*R3`	; Сначала вычисляются выражения в скобках.
`R15=SQRT(POT(R1)+POT(R2))`	; Сначала раскрываются внутренние скобки:
	`R15 = квадратный корень( (R1^2 + R2^2) )`
`RESFRAME=FRAME1:FRAME2`	`;` FRAME-соединение со связывающим оператором
`FRAME3=CTRANS(…):CROT(…)`	Присваивание значения FRAME-компоненту

4.1.3.2 Операторы сравнения и логические операторы

Операции сравнения могут использоваться, к примеру, для формулировки условия перехода. При этом возможно сравнение и сложных выражений.

Операции сравнения могут использоваться для переменных типа CHAR, INT, REAL и BOOL . Для типа CHAR сравнивается кодовое значение.
Для типов STRING, AXIS и FRAME возможны: == и <>, которые для операций типа STRING могут применяться и в синхронных действиях.

Результат сравнительных операций всегда имеет тип BOOL.

Логические операторы служат для связи значений истинности.

Логические операции могут применяться только для переменных типа BOOL. Через внутреннее преобразование типов они могут использоваться и для типов данных CHAR, INT, и REAL.

Для логических (булевых) операций для типов данных BOOL, CHAR, INT и REAL действует:

- 0 соответствует: false
- Отлично от 0 соответствует: TRUE

Побитовые логические операторы

С помощью переменных типа CHAR и INT могут выполняться и побитовые логические операции. При необходимости преобразование типов выполняется автоматически.

Программирование

Оператор сравнения	Значение
==	равно
<>	не равно
>	больше
<	меньше
>=	больше или равно
<=	меньше или равно

Логический оператор	Значение
AND	И
OR	ИЛИ
NOT	Отрицание
XOR	исключающее ИЛИ

Побитовый логический оператор	Значение
B_AND	побитовый И
B_OR	побитовый ИЛИ
B_NOT	побитовое отрицание
B_XOR	побитовое исключающее ИЛИ

Примечание

В арифметических выражениях посредством круглых скобок можно устанавливать последовательность выполнения всех операторов, отличную от обычных правил приоритета.

Примечание

Между ЛОГИЧЕСКИМИ операндами и операторами необходимо вставлять пробелы.

Примечание

Оператор B_NOT относится только к одному операнду. Он стоит после оператора.

Примеры

Пример 1: Операторы сравнения
```
IF R10>=100 GOTOF ZIEL
```

или
```
R11=R10>=100
IF R11 GOTOF ZIEL
```

Результат сравнения `R10>=100` сначала буферизируется в `R11`.

Пример 2: Логические операторы
```
IF (R10<50) AND ($AA_IM[X]>=17.5) GOTOF ZIEL
```

или
```
IF NOT R10 GOTOB START
```

`NOT` относится только к одному операнду.

Пример 3: Побитовые логические операторы
```
IF $MC_RESET_MODE_MASK B_AND 'B10000' GOTOF ACT_PLANE
```

4.1.3.3 Приоритет операций

Каждому оператору соответствует приоритет. При обработке выражения всегда сначала используются операторы с более высоким приоритетом. В случае равнозначных операторов обработка выполняется слева направо.

В арифметических выражениях посредством круглых скобок можно устанавливать последовательность выполнения всех операторов, отличную от обычных правил приоритета.

Последовательность операторов

От высшего к низшему приоритету

1.	NOT, B_NOT	Отрицание, побитовое отрицание
2.	*, /, DIV, MOD	Умножение, деление
3.	+, –	Сложение, вычитание
4.	B_AND	Побитовое И
5.	B_XOR	Побитовое исключающее ИЛИ
6.	B_OR	Побитовое ИЛИ
7.	AND	И
8.	XOR	Исключающее ИЛИ
9.	OR	ИЛИ
10.	<<	Соединение строк, тип результата STRING
11.	==, <>, >, <, >=, <=	Операторы сравнения

Примечание

Связывающий оператор ":" для фреймов не может использоваться с другими операторами в одном выражении. Поэтому градация приоритетов для этого оператора не требуется.

Пример оператора If

```
If (otto==10) and (anna==20) gotof end
```

4.1.3.4 Коррекция точности при ошибках сравнения (TRUNC)

Команда TRUNC отрезает умноженный на коэффициент точности операнд.

Настраиваемая точность для команд сравнения

Внутреннее отображение данных программы обработки детали типа REAL осуществляется в формате IEEE с 64 битами. Из-за такого формата десятичные числа могут отображаться неточно, что при сравнении с идеально вычисленными значениями может привести к нежелательным результатам.

Относительное равенство

Для того, чтобы обусловленные форматом отображения неточности, не сказывались отрицательно на ходе программы, команды сравнения контролируются на предмет на абсолютного, а относительно равенства.

Синтаксис

Коррекция точности при ошибках сравнения

```
TRUNC (R1*1000)
```

Значение

TRUNC:	Обрезание мест после запятой

Учтенное относительное равенство 10^{-12} при операциях

- Равенство: (==)
- Неравенство: (<>)
- Больше или равно: (>=)
- Меньше или равно: (<=)
- Больше/Меньше: (><) с абсолютным равенством
- Больше: (>)
- Меньше: (<)

Совместимость

По соображениям совместимости проверка на предмет относительного равенства при (>) и (<) может быть деактивирована через установку машинных данных MD10280 $MN_PROG_FUNCTION_MASK Бит0 = 1.

Примечание

Поэтому сравнения с данными типа REAL по названным причинам всегда содержат определенную неточность. В случае неприемлемых погрешностей необходимо перейти на вычисление INTEGER, при этом операнды умножаются на коэффициент точности и после обрезаются с TRUNC.

Синхронные действия

Описанное поведение команд сравнения распространяется и на синхронные действия.

Примеры

Пример 1: Анализ точности

Программный код	Комментарий
`N40 R1=61.01 R2=61.02 R3=0.01`	; присваивание начальных значений
`N41 IF ABS(R2-R1) > R3 GOTOF FEHLER`	; прежде переход был бы выполнен
`N42 M30`	; конец программы
`N43 FEHLER: SETAL(66000)`	
`R1=61.01 R2=61.02 R3=0.01`	; присваивание начальных значений
`R11=TRUNC(R1*1000) R12=TRUNC(R2*1000) R13=TRUNC(R3*1000)`	; коррекция точности
`IF ABS(R12-R11) > R13 GOTOF FEHLER`	; переход более не выполняется
`M30`	; конец программы
`FEHLER: SETAL(66000)`	

Пример 2: Образование и обработка коэффициента двух операндов

Программный код	Комментарий
`R1=61.01 R2=61.02 R3=0.01`	; присваивание начальных значений
`IF ABS((R2-R1)/R3)-1) > 10EX-5 GOTOF FEHLER`	; переход не выполняется
`M30`	; конец программы
`FEHLER: SETAL(66000)`	

4.1.3.5 Округления с избытком (ROUNDUP)

С помощью функции "ROUNDUP" вводные значения типа REAL (дробные числа с десятичной точкой) могут быть округлены до следующего большего целого числа.

Синтаксис

`ROUNDUP(<значение>)`

Значение

ROUNDUP:	Команда для округления вводного значения
<значение>:	Вводное значение типа REAL

Примечание

Вводные значения типа INTEGER (целое число) возвращаются без изменений.

Примеры

Пример 1: Различные вводные значения и результаты их округления

Пример	Результат округления
ROUNDUP(3.1)	4.0
ROUNDUP(3.6)	4.0
ROUNDUP(-3.1)	-3.0
ROUNDUP(-3.6)	-3.0
ROUNDUP(3.0)	3.0
ROUNDUP(3)	3.0

Пример 2: ROUNDUP в программе обработки детали ЧПУ

Программный код

```
N10 X=ROUNDUP(3.5) Y=ROUNDUP(R2+2)
N15 R2=ROUNDUP($AA_IM[Y])
N20 WHEN X=100 DO Y=ROUNDUP($AA_IM[X])
...
```

4.1.4 Операции со строкой

Операции со строкой

Наряду с классическими операциями "Присваивание значения" и "Сравнение" возможны следующие операции со строкой:

- Преобразование типов в STRING (AXSTRING) (Страница 462)
- Преобразование типов из STRING (NUMBER, ISNUMBER, AXNAME) (Страница 463)
- Соединение строк (<<) (Страница 464)
- Преобразование в строчные/прописные буквы (TOLOWER, TOUPPER) (Страница 465)
- Определение длины строки (STRLEN) (Страница 466)

- Поиск символа/строки в строке (INDEX, RINDEX, MINDEX, MATCH) (Страница 466)
- Выбор части строки (SUBSTR) (Страница 467)
- Чтение и запись отдельных символов (Страница 468)
- Форматирование строки (SPRINT) (Страница 469)

Специальное значение символа 0

Символ 0 внутренне интерпретируется на идентификатор конца строки. При замене символа на символ 0 строка сокращается.

Пример:

Программный код	Комментарий
`DEF STRING[20] STRG="Ось . остановлена"`	
`STRG[6]="X"`	
`MSG(STRG)`	; Выводит сообщение "Ось X остановлена".
`STRG[6]=0`	
`MSG(STRG)`	; Выводит сообщение "Ось".

4.1.4.1 Преобразование типов в STRING (AXSTRING)

Благодаря функции "Преобразование типа в STRING" возможно использование переменных различного типа в качестве составной части сообщения (MSG).

Выполняется не явно при использовании оператора << для типов данных INT, REAL, CHAR и BOOL (см. " Соединение строк (<<) (Страница 464) ").

Значение INT преобразуется в обычную читабельную форму. Для значений REAL указывается до 10 мест после запятой.

С помощью команды `AXSTRING` возможно преобразование переменных типа AXIS в STRING.

Синтаксис

```
<STRING_ERG> = << <bel._Typ>
<STRING_ERG> = AXSTRING(<идентификатор оси>)
```

Значение

<STRING_ERG>:	Переменная для результата преобразования типа	
	Тип:	STRING
<люб._тип>:	Типы переменных INT, REAL, CHAR, STRING и BOOL	
AXSTRING:	Команда AXSTRING выводит указанный идентификатор оси как строку.	
<идентификатор оси>:	Переменная для идентификатора оси	
	Тип:	AXIS

Примечание

Переменные FRAME не могут быть преобразованы.

4.1.4.2 Преобразование типов из STRING (NUMBER, ISNUMBER, AXNAME)

С помощью команды NUMBER выполняется преобразование из STRING в REAL. Возможность преобразования можно проверить с помощью команды ISNUMBER .

С помощью команды AXNAME строка преобразуется в тип данных AXIS.

Синтаксис

```
<REAL_ERG>=NUMBER("<строка>")
<BOOL_ERG>=ISNUMBER("<строка>")
<AXIS_ERG>=AXNAME("<строка>")
```

Значение

NUMBER:	Команда NUMBER возвращает представленное через <строка> число как значение REAL.		
<строка>:	Преобразуемая переменная типа STRING		
<REAL_ERG>:	Переменная для результата преобразования типа с NUMBER		
	Тип:	REAL	
ISNUMBER:	С помощью команды ISNUMBER можно проверить, может ли <строка> быть преобразован в действительное число.		
<BOOL_ERG>:	Переменная для результата опроса с ISNUMBER		
	Тип:	BOOL	
	Значение:	TRUE	ISNUMBER выводит значение TRUE, если <строка> представляет действительное по правилам языка число REAL.
		false	Если ISNUMBER выводит значение FALSE, то при вызове NUMBER с той же <строка> следует ошибка.
AXNAME:	Команда AXNAME преобразует указанную <строку> в идентификатор оси. **Примечание:** Если <строка> не может быть согласована с сконфигурированным идентификатором оси, то следует ошибка.		
<AXIS_ERG>:	Переменная для результата преобразования типа с AXNAME		
	Тип:	AXIS	

Пример

Программный код	Комментарий
DEF BOOL BOOL_ERG	
DEF REAL REAL_ERG	
DEF AXIS AXIS_ERG	

Программный код	Комментарий
`BOOL_ERG=ISNUMBER("1234.9876Ex-7")`	`; BOOL_ERG == TRUE`
`BOOL_ERG=ISNUMBER("1234XYZ")`	`; BOOL_ERG == FALSE`
`REAL_ERG=NUMBER("1234.9876Ex-7")`	`; REAL_ERG == 1234.9876Ex-7`
`AXIS_ERG=AXNAME("X")`	`; AXIS_ERG == X`

4.1.4.3 Соединение строк (<<)

Функция "Соединение строк" позволяет составить строку из отдельных частей.

Связь реализуется через оператор "<<". Конечным типом этого оператора для всех комбинаций основных типов CHAR, BOOL, INT, REAL и STRING является STRING. Возможно необходимое преобразование выполняется по существующим правилам.

Синтаксис

<люб._тип> << <люб._тип>

Объяснение

<люб._тип>:	Переменные типа CHAR, BOOL, INT, REAL или STRING
<< :	Оператор для соединения переменных (<люб._тип>) в составную строку символов (тип STRING). Этот оператор доступен отдельно в качестве так называемого "унарного" варианта. Таким образом возможно явное преобразование типов в STRING (не для FRAME и AXIS): << <люб._тип>

К примеру, таким образом можно составить сообщение или команду из текстовых списков и вставить параметры (около одного имени блока):
`MSG(STRG_TAB[LOAD_IDX]<<`ИМЯ БЛОКА)

Примечание

Промежуточные результаты при соединении строк не могут превышать макс. длины строки.

Примечание

Типы FRAME и AXIS не могут использоваться с оператором "<<".

Примеры

Пример 1: Соединение строк

Программный код	Комментарий
`DEF INT IDX=2`	
`DEF REAL VALUE=9.654`	

Программный код	Комментарий
`DEF STRING[20] STRG="INDEX:2"`	
`IF STRG=="Индекс:"<<IDX GOTOF NO_MSG`	
`MSG("Индекс:"<<IDX<<"/значение:"<<VALUE)`	; Индикация: "Индекс:2/значение:9.654"
`NO_MSG:`	

Пример 2: Явное преобразование типа с <<

Программный код	Комментарий
`DEF REAL VALUE=3.5`	
`<<VALUE`	; Указанная переменная типа `REAL` преобразуется в тип `STRING`.

4.1.4.4 Преобразование в строчные/прописные буквы (TOLOWER, TOUPPER)

Функция "Преобразование в строчные/прописные буквы" позволяет преобразовать все буквы строки символов в унифицированное представление.

Синтаксис

```
<STRING_ERG>=TOUPPER("<строка>")
<STRING_ERG>=TOLOWER("<строка>")
```

Значение

`TOUPPER`:	С помощью команды `TOUPPER` все буквы строки символов преобразуются в **прописные** буквы.	
`TOLOWER`:	С помощью команды `TOLOWER` все буквы строки символов преобразуются в **строчные** буквы.	
<строка>:	Строка символов, которая должна быть преобразована	
	Тип:	STRING
`<STRING_ERG>`:	Переменная для результата преобразования	
	Тип:	STRING

Пример

Так как также существует возможность запуска данных пользователя на интерфейсе, то может быть достигнуто унифицированное представление со строчными или прописными буквами:

Программный код

```
DEF STRING [29] STRG
...
IF "LEARN.CNC"==TOUPPER(STRG) GOTOF LOAD_LEARN
```

4.1.4.5 Определение длины строки (STRLEN)

С помощью команды STRLEN можно определить длину строки символов.

Синтаксис

<INT_ERG>=STRLEN("<СТРОКА>")

Значение

STRLEN:	С помощью команды STRLEN определяется длина указанной строки символов. Возвращается число символов, которые – считая от начала строки символов - не являются символами 0.	
<строка>:	Строка символов, длина которой должна быть определена	
	Тип:	STRING
<INT_ERG>:	Переменная для результата определения	
	Тип:	INT

Пример

Функция в комбинации с доступом к отдельным символам позволяет определить конец строки символов:

Программный код

```
IF (STRLEN(BAUSTEIN_NAME)>10) GOTOF FEHLER
```

4.1.4.6 Поиск символа/строки в строке (INDEX, RINDEX, MINDEX, MATCH)

Эта функция позволяет выполнять поиск отдельных символов или строки в другой строке. Результат функции показывает, на какой позиции строки был найден символ/строка в рассматриваемой строке.

Синтаксис

INT_ERG=INDEX(STRING,CHAR) ; **Тип результата: INT**

INT_ERG=RINDEX(STRING,CHAR) ; **Тип результата: INT**

INT_ERG=MINDEX(STRING,STRING) ; **Тип результата: INT**

INT_ERG=MATCH(STRING,STRING) ; **Тип результата: INT**

Семантика

Функции поиска: Они возвращают позицию в строке (первый параметр), где поиск был успешным. Если символ/строка не найдены, то возвращается значение -1. При этом первый символ имеет позицию 0.

Значение

INDEX:	Поиск указанного вторым параметром символа (сначала) в первом параметре.
RINDEX:	Поиск указанного вторым параметром символа (с конца) в первом параметре.
MINDEX:	Соответствует функции INDEX, за исключением того, что передается список символов (как строка), из которых возвращается индекс первого найденного символа.
MATCH:	Поиск строки в строке.

Таким образом, строки могут разбиваться по определенным критериям, к примеру, по позициям с пробелом или разделительному символу пути ("/").

Пример

Разделение введенных данных по имени пути и блока

Программный код	Комментарий
`DEF INT PFADIDX, PROGIDX`	
`DEF STRING[26] EINGABE`	
`DEF INT LISTIDX`	
`EINGABE = "/_N_MPF_DIR/ _N_EXECUTE_MPF"`	
`LISTIDX = MINDEX (EINGABE, "M,N,O,P") + 1`	; Как значение в `LISTIDX` возвращается 3; т.к. `"N"` это первый символ в параметре `EINGABE`, из списка выбора сначала.
`PFADIDX = INDEX (EINGABE, "/") +1`	; при этом: `PFADIDX = 1`
`PROGIDX = RINDEX (EINGABE, "/") +1`	; при этом: `PROGIDX = 12`
	; С помощью введенной в следующем разделе функции `SUBSTR` переменная `EINGABE` может быть разбита на компоненты ;"Путь";и "Блок":
`VARIABLE = SUBSTR (EINGABE, PFADIDX, PROGIDX-PFADIDX-1)`	; выводит `"_N_MPF_DIR"`
`VARIABLE = SUBSTR (EINGABE, PROGIDX)`	; выводит `"_N_EXECUTE_MPF"`

4.1.4.7 Выбор части строки (SUBSTR)

С помощью функции SUBSTRING возможно чтение любых частей в рамках строки.

Синтаксис

`<STRING_ERG>=SUBSTR(<`строка`>,<`индекс`>,<`длина`>)`

`<STRING_ERG>=SUBSTR(<`строка`>,<`индекс`>)`

Значение

SUBSTR:	Функция выводит из <строка> часть строки, исходя из <индекс> с указанной <длина>. Если параметр <длина> не указан, то функция выводит часть строки, исходя из <индекс> до конца строки.
<индекс>:	Начальная позиция части строки в рамках строки. Если начальная позиция лежит за концом строки, то возвращается пустая строка (" "). Первый символ строки: Индекс = 0 Диапазон значений: 0 ... (длина строки - 1)
<длина>:	Длина части строки. Если указывается слишком большая длина, то возвращается только часть строки до конца строки. Диапазон значений: 1 ... (длина строки - 1)

Пример

Программный код	Комментарий
`DEF STRING[29] ERG`	
`;  1`	
`;  0123456789012345678`	
`ERG = SUBSTR("QUITTUNG: 10 до 99", 10, 2)`	`; ERG == "10"`
`ERG = SUBSTR("QUITTUNG: 10 до 99", 10)`	`; ERG == "10 до 99"`

4.1.4.8 Чтение и запись отдельных символов

В рамках строки возможно чтение и запись отдельных символов.

При этом необходимо соблюдать следующие граничные условия:

- возможно только для определенных пользователем переменных, не для системных переменных
- отдельные символы строки при вызовах подпрограмм передаются только "call by value"

Синтаксис

<символ>=<строка>[<индекс>]
<символ>=<строка_массив>[<массив_индекс>,<индекс>]
<строка>[<индекс>]=<символ>
<строка_массив>[<массив_индекс>,<индекс>]=<символ>

Значение

<строка>:	Любая строка
<символ>:	Переменная типа CHAR
<индекс>:	Позиция символа в рамках строки. Первый символ строки: Индекс = 0 Диапазон значений: 0 ... (длина строки - 1)

Примеры

Пример 1: Переменное сообщение

Программный код	Комментарий
; 0123456789	
DEF STRING [50] MELDUNG = "Ось n достигла позиции"	
MELDUNG [6] = "X"	
MSG (MELDUNG)	; "Ось X достигла позиции"

Пример 2: Обработка системной переменной

Программный код	Комментарий
DEF STRING [50] STRG	; Буфер для системных переменных
...	
STRG = $P_MMCA	; Загрузка системных переменных
IF STRG[0] == "E" GOTO ...	; Обработка системных переменных

Пример 3: Передача параметров "вызов по значению" и "вызов по ссылке"

Программный код	Комментарий
; 0123456	
DEF STRING[50] STRG = "ось X"	
DEF CHAR CHR	
...	
EXTERN UP_VAL(ACHSE)	; Определение подпрограммы с параметром "вызов по значению"
EXTERN UP_REF(VAR ACHSE)	; Определение подпрограммы с параметром "вызов по ссылке"
...	
UP_VAL(STRG[6])	; Передача параметров "по значению"
...	
CHR = STRG[6]	; Буферизация
UP_REF(CHR)	; Передача параметров "по ссылке"

4.1.4.9 Форматирование строки (SPRINT)

С помощью предопределенной функции SPRINT можно форматировать и, к примеру, подготавливать строки символов для вывода на внешние устройства (см. также "Process DataShare - Вывод на внешнее устройство/файл (EXTOPEN, WRITE, EXTCLOSE) (Страница 1039)").

Синтаксис

"<результат_строка>"=SPRINT("<формат_строка>",<значение_1>,<значение_2>,..., <значение_n>)

Значение

SPRINT:	Идентификатор для предопределенной функции, выводящей значение типа STRING.
"<формат_строка>":	Строка символов, содержащая постоянные и переменные части. Переменные части определяются знаком управления форматом % и последующим описанием формата.
< значение_1>,< значение_2>,...,< значение_n>:	Значение в форме постоянной или переменной ЧПУ, вставляемое в месте, в котором стоит n-ный знак управления форматом %, согласно описанию формата в <формат_строка>.
"<результат_строка>":	Форматированная строка символов (макс. 400 байт)

Имеющиеся описания форматов

%B:	Преобразование в строку "TRUE", если преобразуемое значение:
	• отлично от 0.
	• не является пустой строкой (для значений String).
	Преобразование в строку "FALSE", если преобразуемое значение:
	• равно 0.
	• является пустой строкой.
	Пример:
	`N10 DEF BOOL BOOL_VAR=1`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF BOOL_VAR:%B", BOOL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF BOOL_VAR:TRUE".

%C:	Преобразование с символ ASCII.
	Пример:
	`N10 DEF CHAR CHAR_VAR="X"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF CHAR_VAR:%C",CHAR_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF CHAR_VAR:X".

%D:	Преобразование в строку с целочисленным значением (INTEGER).
	Пример:
	`N10 DEF INT INT_VAR=123`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF INT_VAR:%D",INT_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF INT_VAR:123".

%<m>D:	Преобразование в строку с целочисленным значением (INTEGER). Строка имеет мин. длину в <m> символов. Отсутствующие места заполняются выровненными по левым разрядам символами пробела.
	Пример:
	`N10 DEF INT INT_VAR=-123`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF INT_VAR:%6D",INT_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF INT_VAR:xx-123" ("x" в примере замещает символ пробела).

%F:	Преобразование в строку с десятичным числом с 6 местами после запятой. Места после запятой при необходимости округляются или заполняются 0.
	Пример:
	`N10 DEF REAL REAL_VAR=-1.2341234EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%F",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:-1234.123400".

%<m>F:	Преобразование в строку с десятичным числом с 6 местами после запятой и общей длиной мин. в <m> символов. Места после запятой при необходимости округляются или заполняются 0. Отсутствующие символы на всю длину <m> заполняются выровненными по левым разрядам символами пробела.
	Пример:
	`N10 DEF REAL REAL_VAR=-1.23412345678EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15F",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR": xxx-1234.123457" ("x" в примере замещает символ пробела).

%.<n>F:	Преобразование в строку с десятичным числом с <n> местами после запятой. Места после запятой при необходимости округляются или заполняются 0.
	Пример:
	`N10 DEF REAL REAL_VAR=-1.2345678EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3F",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:-1234.568".

%<m>.<n>F:	Преобразование в строку с десятичным числом с <n> местами после запятой и общей длиной мин. в <m> символов. Места после запятой при необходимости округляются или заполняются 0. Отсутствующие символы на всю длину <m> заполняются выровненными по левым разрядам символами пробела.
	Пример:
	`N10 DEF REAL REAL_VAR=-1.2341234567890EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%10.2F",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:xx-1234.12" ("x" в примере замещает символ пробела).

%E:	Преобразование в строку с десятичным числом в экспоненциальном представлении. Мантисса сохраняется нормализированной с одним местом перед запятой и 6 местами после запятой. Места после запятой при необходимости округляются или заполняются 0. Показатель начинается с кодового слова "EX". За ним следует знак ("+" или "-") и двух- или трехзначное число.
	Пример:
	`N10 DEF REAL REAL_VAR=-1234.567890`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%E",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:-1.234568EX+03".

%<m>E:	Преобразование в строку с десятичным числом в экспоненциальном представлении с общей длиной мин. в <m> символов. Отсутствующие символы заполняются выровненными по левым разрядам символами пробела. Мантисса сохраняется нормализированной с одним местом перед запятой и 6 местами после запятой. Места после запятой при необходимости округляются или заполняются 0. Показатель начинается с кодового слова "EX". За ним следует знак ("+" или "-") и двух- или трехзначное число.
	Пример:
	`N10 DEF REAL REAL_VAR=-1234,5`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%20E",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:xxxxxx-1.234500EX+03" ("x" в примере замещает символ пробела)

%.<n>E:	Преобразование в строку с десятичным числом в экспоненциальном представлении. Мантисса сохраняется нормализированной с одним местом перед запятой и <n> местами после запятой. Места после запятой при необходимости округляются или заполняются 0. Показатель начинается с кодового слова "EX". За ним следует знак ("+" или "-") и двух- или трехзначное число.
	Пример:
	`N10 DEF REAL REAL_VAR=-1234,5678`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.2E",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:-1.23EX+03".

%<m>.<n>E:	Преобразование в строку с десятичным числом в экспоненциальном представлении с общей длиной мин. в <m> символов. Отсутствующие символы заполняются выровненными по левым разрядам символами пробела. Мантисса сохраняется нормализированной с одним местом перед запятой и <n> местами после запятой. Места после запятой при необходимости округляются или заполняются 0. Показатель начинается с кодового слова "EX". За ним следует знак ("+" или "-") и двух- или трехзначное число.
	Пример:
	`N10 DEF REAL REAL_VAR=-1234,5678`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.2E", REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:xx-1.23EX+03" ("x" в примере замещает символ пробела).

%G:	Преобразование в строку с десятичным числом в десятичном или экспоненциальном представлении в зависимости от диапазона значений: если представляемое значение по величине меньше 1.0EX-04 или больше/равно 1.0EX+06, то выбирается экспоненциальное представление, в остальных случаях десятичное представление. Отображается макс. шесть значимых мест, при необходимости выполняется округление.
	Пример с десятичным представлением:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:0,000123457".
	Пример с экспоненциальным представлением:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+06`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:1.23457EX+06".

%<m>G:	Преобразование в строку с десятичным числом в десятичном или экспоненциальном представлении в зависимости от диапазона значений (как %G). Строка имеет общую длину мин. в <m> символов. Отсутствующие символы заполняются выровненными по левым разрядам символами пробела.
	Пример с десятичным представлением:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:xxxx0.000123457" ("x" в примере замещает символ пробела).
	Пример с экспоненциальным представлением:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+06`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:xxx1.23457EX+06" ("x" в примере замещает символ пробела).

%.<n>G:	Преобразование в строку с десятичным числом в десятичном или экспоненциальном представлении в зависимости от диапазона значений. Отображается макс. <n> значимых мест, при необходимости выполняется округление. если представляемое значение по величине меньше 1.0EX-04 или больше/равно 1.0EX(+<n>), то выбирается экспоненциальное представление, в остальных случаях десятичное представление.
	Пример с десятичным представлением:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:0,000123".
	Пример с экспоненциальным представлением:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT = SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:1.23EX+03".

%<m>.<n>G:	Преобразование в строку с десятичным числом в десятичном или экспоненциальном представлении в зависимости от диапазона значений (как %.<n>G). Строка имеет общую длину мин. в <m> символов. Отсутствующие символы заполняются выровненными по левым разрядам символами пробела.
	Пример с десятичным представлением:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:xxx0.0001235" ("x" в примере замещает символ пробела).
	Пример с экспоненциальным представлением:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF REAL_VAR:xx1.235EX+06" ("x" в примере замещает символ пробела).

%.<n>P:	Преобразование REAL-значения в INTEGER-значение с учетом <n> мест после запятой. INTEGER-значение выводится как 32-битное двоичное число. Если преобразуемое значение не может быть представлено в 32 бит, то обработка отменяется с ошибкой. Т.к. созданная с оператором спецификации формата %.<n>P байтовая последовательность может содержать и двоичные нули, то созданная таким образом общая строка более не соответствует условным обозначениям типа данных ЧПУ STRING. Поэтому она не может быть ни сохранена в переменную типа STRING, ни подвергнута дальнейшей обработке с помощью строковых команд языка ЧПУ. Единственным возможным применением является передача параметров на команду WRITE с выводом на соответствующее внешнее устройство (см. пример ниже). Как только <формат_строка> содержит описание формата типа %P, вся строка, за исключением сгенерированного с %.<n>P двоичного числа, выводится согласно MD10750 $MN_SPRINT_FORMAT_P_CODE в символьном коде ASCII, ISO (DIN6024) или EIA (RS244). При программировании символа, который не может быть преобразован, обработка отменяется с ошибкой. **Пример:** N10 DEF REAL REAL_VAR=123.45 N20 DEF INT ERROR N30 DEF STRING[20] EXT_DEVICE="/ext/dev/1" ... N100 EXTOPEN(ERROR,EXT_DEVICE) N110 IF ERROR <> 0 ... ; обработка ошибок N200 WRITE(ERROR,EXT_DEVICE,SPRINT("INTEGER BINARY CODED:%.3P",REAL_VAR) N210 IF ERROR <> 0 ... ; обработка ошибок Результат: Строка "INTEGER BINARY CODED: 'H0001E23A'" передается на устройство вывода /ext/dev/1. Шестнадцатеричное значение 0x0001E23A соответствует десятичному значению 123450.

%<m>.<n>P:	Преобразование REAL-значения согласно установке в машинных данных MD10751 $MN_SPRINT_FORMAT_P_DECIMAL в строку с:
	• целым числом из <m> + <n> мест **или**
	• десятичным числом макс. с <m> местами перед запятой и точно <n> местами после запятой.
	Как у описания формата %.<n>P вся строка сохраняется в определенный через MD10750 $MN_SPRINT_FORMAT_P_CODE код символа.
	Преобразование при MD10751 = 0:
	Значение REAL преобразуется в строку с целым числом из <m> + <n> мест. При необходимости места после запятой округляются до <n> мест или заполняются нулями. Отсутствующие места перед запятой заполняются символами пробела. Знак минус добавляется выровненным слева, вместо знака плюс вставляется пробел.
	Пример:
	`N10 DEF REAL REAL_VAR=-123.45`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("PUNCHED TAPE FORMAT:%5.3P",REAL_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "PUNCHED TAPE FORMAT:-xx123450" ("x" в примере замещает символ пробела).
	Преобразование при MD10751 = 1:
	Значение REAL преобразуется в строку с десятичным числом макс. <m> местами перед запятой и точно <n> местами после запятой. При необходимости места перед запятой обрезаются, а места после запятой округляются или заполняются нулями. Если <n> равно 0, то десятичная точка также не нужна.
	Пример:
	`N10 DEF REAL REAL_VAR1=-123.45`
	`N20 DEF REAL REAL_VAR2=123.45`
	`N30 DEF STRING[80] RESULT`
	`N40 RESULT=SPRINT("PUNCHED TAPE FORMAT:%5.3P VAR2:%2.0P", REAL_VAR1,REAL_VAR2)`
	Результат: Строковая переменная RESULT описывается строкой символов "PUNCHED TAPE FORMAT:-123.450 VAR2:23".

%S:	Вставка строки.
	Пример:
	`N10 DEF STRING[16] STRING_VAR="ABCDEFG"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%S",STRING_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF STRING_VAR:ABCDEFG".

%<m>S:	Вставка строки мин. с <m> символами. Отсутствующие места заполняются символами пробела.
	Пример:
	`N10 DEF STRING[16] STRING_VAR="ABCDEFG"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10S",STRING_VAR)`
	Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF STRING_VAR:xxxABCDEFG" ("x" в примере замещает символ пробела).

%.<n>S:	Вставка <n> символов строки (начиная с первого символа). **Пример:** N10 DEF STRING[16] STRING_VAR="ABCDEFG" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%.3S",STRING_VAR) Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF STRING_VAR:ABC".
%<m>.<n>S:	Вставка <n> символов строки (начиная с первого символа). Созданная строка имеет общую длину мин. в <m> символов. Отсутствующие места заполняются символами пробела. **Пример:** N10 DEF STRING[16] STRING_VAR="ABCDEFG" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10.5S", STRING_VAR) Результат: Строковая переменная RESULT описывается строкой символов "CONTENT OF STRING_VAR:xxxxxABCDE" ("x" в примере замещает символ пробела).
%X:	Преобразование значения INTEGER в строку с шестнадцатеричным представлением. **Пример:** N10 DEF INT INT_VAR='HA5B8' N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("INTEGER HEXADECIMAL:%X",INT_VAR) Результат: Строковая переменная RESULT описывается строкой символов "INTEGER HEXADECIMAL:A5B8".

Примечание

Свойство языка ЧПУ, не делающего различий между прописным и строчным написанием идентификаторов и кодовых слов, действует и для описаний форматов. Поэтому они могут программироваться прописными или строчными буквами без различия в функциях.

Возможности образования комбинаций

Таблица ниже показывает, какие типы данных ЧПУ могут комбинироваться с каким описанием формата. Действуют правила для не явного преобразования типа данных (см. "Типы данных (Страница 437)").

	Типы данных ЧПУ						
	BOOL	**CHAR**	**INT**	**REAL**	**STRING**	**AXIS**	**FRAME**
%B	+	+	+	+	+	-	-
%C	-	+	-	-	+	-	-
%D	+	+	+	+	-	-	-
%F	-	-	+	+	-	-	-
%E	-	-	+	+	-	-	-
%G	-	-	+	+	-	-	-
%S	-	+	-	-	+	-	-

	Типы данных ЧПУ						
	BOOL	**CHAR**	**INT**	**REAL**	**STRING**	**AXIS**	**FRAME**
%X	+	+	+	-	-	-	-
%P	-	-	+	+	-	-	-

Примечание

Из таблицы видно, что типы данных ЧПУ AXIS и FRAME не могут использоваться напрямую в функции SPRINT. Но можно:

- преобразовать тип данных AXIS с помощью функции AXSTRING в строку, которая после может быть обработана со SPRINT.
- прочитать отдельные значения типа данных FRAME через доступ к фрейм-компонентам. Благодаря этому получаются данные типа REAL, которые могут быть подвергнуты дальнейшей обработке со SPRINT.

4.1.5 Переходы и ветвления в программе

4.1.5.1 Возврат на начало программы (GOTOS)

С помощью команды GOTOS для повторения программы можно вернуться на начало главной или подпрограммы.

Через машинные данные можно установить, чтобы при каждом возврате на начало программы:

- время выполнения программы устанавливалось бы на "0".
- подсчет деталей увеличивался бы на значение "1".

Синтаксис

```
GOTOS
```

Значение

GOTOS:	Оператор перехода с целью перехода "начало программы".	
	Выполнение управляется через интерфейсный сигнал ЧПУ/PLC: DB21, ... DBX384.0 (управление ветвлением программы)	
	Значение:	Значение:
	0	Нет возврата на начало программы. Выполнение программы продолжается на следующем после GOTOS кадре программы обработки детали.
	1	Возврат на начало программы. Программа обработки детали повторяется.

Граничные условия

- GOTOS вызывает внутреннюю STOPRE (остановка предварительной обработки).
- В программе обработки детали с определениями данных (переменные LUD) с GOTOS выполняется переход на первый кадр программы после раздела определения, т.е. определения данных не осуществляются повторно. Таким образом, определенные переменные сохраняют достигнутое в кадре GOTOS значение и не сбрасываются на запрограммированные в разделе определения стандартные значения.
- Если программы с операторами перехода должны выполняться через функцию "Выполнение с внешнего устройства" из внешней программной памяти, то цели перехода должны находиться в пределах памяти догрузки.
  Это условие, как правило, является проблемой для операторов перехода на начало программы (GOTOS), так как программы обычно имеют большой размер для помещения в память догрузки. При первой догрузке начало программы удаляется из памяти догрузки. Если при дальнейшем выполнении программы потребуется обработать оператор перехода на начало программы, тогда функция будет не в состоянии найти цель перехода. Происходит отмена программы с выводом аварийного сообщения 14000.

  Примечание

  Для выполнения внешних программ без ограничений в части запрограммированных операторов перехода, рекомендуется вместо функции "Выполнение с внешнего устройства" использовать опцию "Выполнение из внешней памяти (EES)".
- В синхронных действиях и технологических циклах команда GOTOS недоступна.

Пример

Программный код	Комментарий
`N10 ...`	; Начало программы.
...	
`N90 GOTOS`	; Переход на начало программы.
...	

4.1.5.2 Переходы на метки перехода (GOTOB, GOTOF, GOTO, GOTOC)

В программе могут быть установлены метки перехода (метки), на которые можно переходить из других мест внутри одной программе с помощью команд GOTOF, GOTOB, GOTO или GOTOC. После выполнение программы начинается на операторе, который следует непосредственно за меткой перехода. Благодаря этому могут быть реализованы ветвления внутри программы.

Наряду с метками перехода в качестве целей перехода возможны и номера главных и вспомогательных кадров.

Если перед оператором перехода сформулировано условие перехода (IF ...), то переход в программе осуществляется только при выполнении условия перехода.

Синтаксис

```
GOTOB <цель перехода>
IF <условие перехода> == TRUE GOTOB <цель перехода>

GOTOF <цель перехода>
IF <условие перехода> == TRUE GOTOF <цель перехода>

GOTO <цель перехода>
IF <условие перехода> == TRUE GOTO <цель перехода>

GOTOC <цель перехода>
IF <условие перехода> == TRUE GOTOC <цель перехода>
```

Значение

GOTOB:	Оператор перехода с целью перехода в направлении начала программы.	
GOTOF:	Оператор перехода с целью перехода в направлении конца программы.	
GOTO:	Оператор перехода с поиском цели перехода. Поиск сначала выполняется в направлении конца программы, потом в направлении начала программы.	
GOTOC:	Действие аналогично GOTO с той разницей, что ошибка 14080 "Цель перехода не найдена" подавляется. Это означает, что выполнение программы в случае безрезультатного поиска цели перехода не отменяется, а продолжается со следующей за командой GOTOC строки программы.	
<цель перехода>:	Параметры цели перехода Возможными данными являются:	
	<метка перехода>:	Цель перехода это установленная в программе метка перехода с пользовательским именем: <метка перехода>:
	<номер кадра>:	Цель перехода это номер главного или вспомогательного кадра (к примеру: 200, N300)
	Переменная типа STRING:	Переменная цель перехода. Переменная обозначает метку перехода или номер кадра.
IF:	Кодовое слово для формулирования условия перехода. Условие перехода допускает все операции сравнения и логические операции (результат: TRUE или FALSE). Переход в программе выполняется, если результат этой операции TRUE.	

Примечание

Метки перехода (метки)

Метки перехода всегда стоят в начале кадра. Если имеется номер программы, то метка перехода стоит непосредственно за номером кадра.

Для наименования меток перехода действуют следующие правила:

- Число символов:
  - минимум 2
  - максимум 32
- Разрешенными символами являются:
  - буквы
  - цифры
  - символы подчеркивания
- Первыми двумя символами должны быть буквы или символы подчеркивания.
- После имени метки перехода следует двоеточие (":").

Граничные условия

- Целью перехода может быть только кадр с меткой перехода или номером кадра, лежащий **внутри** программы.
- Оператор перехода без условия перехода должен быть запрограммирован в отдельном кадре. Для операторов перехода с условиями перехода это ограничение не действует. Здесь можно сформулировать несколько операторов перехода в одном кадре.
- Для программ с операторами перехода без условий перехода конец программы M2/M30 не обязательно должен стоять на конце программы.
- Если программы с операторами перехода должны выполняться через функцию "Выполнение с внешнего устройства" из внешней программной памяти, то цели перехода должны находиться в пределах памяти догрузки. В ином случае цель перехода не будет найдена и происходит отмена программы с выводом аварийного сообщения 14000.

  Примечание

  Для выполнения внешних программ без ограничений в части запрограммированных операторов перехода, рекомендуется вместо функции "Выполнение с внешнего устройства" использовать опцию "Выполнение из внешней памяти (EES)".

Примеры

Пример 1: Переходы на метки перехода

Программный код	Комментарий
`N10 …`	

Программный код	Комментарий
N20 GOTOF Label_1	; переход в направлении конца программы к
	; метка перехода «Label_1».
N30 …	
N40 Label_0: R1=R2+R3	; Метка перехода "Label_0" установлена.
N50 ...	
N60 Label_1:	; Метка перехода "Label_1" установлена.
N70 …	
N80 GOTOB Label_0	; переход в направлении начала программы
	; к метке перехода «Label_0».
N90 …	

Пример 2: Косвенный переход на номер кадра

Программный код	Комментарий
IF <условие> == TRUE	
R10=100	; назначение цели перехода
ELSE	
R10=110	; назначение цели перехода
ENDIF	
; переход в направлении конца программы на кадр, номер которого стоит в R10	
N10 GOTOF "N"<<R10	
…	
N90 ...	
N100 ...	; Цель перехода
N110 ...	
…	

Пример 3: Переход на переменную цель перехода

Программный код	Комментарий
DEF STRING[20] ZIEL	
IF <условие> == TRUE	
ZIEL = "Marke1"	; назначение цели перехода
ELSE	
ZIEL = "Marke2"	; назначение цели перехода
ENDIF	
; переход в направлении конца программы к переменной цели перехода «Содержание ZIEL»	
GOTOF ZIEL	
Marke1: T="Bohrer1"	; цель перехода 1
…	
Marke2: T="Bohrer2"	; цель перехода 2
…	

Пример 4: Переход с условием перехода

Программный код	Комментарий
`N40 R1=30 R2=60 R3=10 R4=11 R5=50 R6=20`	; присваивание начальных значений
`N41 LA1: G0 X=R2*COS(R1)+R5 Y=R2*SIN(R1)+R6`	; метка перехода `LA1`
`N42 R1=R1+R3 R4=R4-1`	
`; IF` условие перехода `== TRUE`	
`; THEN` переход в направлении начала программы к метке перехода `LA1`	
`N43 IF R4>0 GOTOB LA1`	
`N44 M30`	; конец программы

4.1.5.3 Ветвление программы (CASE ... OF ... DEFAULT ...)

Функция CASE предлагает возможность проверки актуального значения (тип: INT) переменной или функции вычисления и в зависимости от результата перейти на различные места в программе.

Синтаксис

`CASE(`<выражение>`) OF` <постоянная_1> `GOTOF` <цель перехода_1>
<постоянная_2> `GOTOF` <цель перехода_2> `...` `DEFAULT GOTOF` <цель перехода_n>

Значение

`CASE`:	Оператор перехода	
<выражение>:	Переменная или функция вычисления	
`OF`:	Кодовое слово для формулировки условных ветвлений программы	
<постоянная_1>:	Первое указанное постоянное значение для переменной или функции вычисления	
	Тип:	INT
<постоянная_2>:	Второе указанное постоянное значение для переменной или функции вычисления	
	Тип:	INT
`DEFAULT`:	На тот случай, когда переменная или функция вычисления не принимает ни одного из указанных постоянных значений, с помощью оператора DEFAULT можно определить цель перехода. **Примечание:** Если оператор DEFAULT не запрограммирован, то в этом случае целью перехода становится следующий за оператором CASE кадр.	
`GOTOF`:	Оператор перехода с целью перехода в направлении конца программы. Вместо `GOTOF` могут быть запрограммированы и все другие команды GOTO (см. тему "Переходы в программе на метки перехода").	

<цель перехода_1>:	Выполняется ветвление на эту цель перехода, если значение переменной или функции вычисления соответствует первой указанной постоянной. Цель перехода может быть указана следующим образом:	
	<метка перехода>:	Цель перехода это установленная в программе метка перехода с пользовательским именем: <метка перехода>:
	<номер кадра>:	Цель перехода это номер главного или вспомогательного кадра (к примеру: 200, N300)
	Переменная типа STRING:	Переменная цель перехода. Переменная обозначает метку перехода или номер кадра.
<цель перехода_2>:	Выполняется ветвление на эту цель перехода, если значение переменной или функции вычисления соответствует второй указанной постоянной.	
<цель перехода_n>:	Выполняется ветвление на эту цель перехода, если значение переменной не принимает ни одного из указанных постоянных значений.	

Граничные условия

Выполнение с внешнего устройства

Если программы с функцией CASE должны выполняться через функцию "Выполнение с внешнего устройства" из внешней программной памяти, то цели перехода должны находиться в пределах памяти догрузки. В ином случае цель перехода не будет найдена и происходит отмена программы с выводом аварийного сообщения 14000.

Примечание

Для выполнения внешних программ без ограничений в части запрограммированных операторов перехода, рекомендуется вместо функции "Выполнение с внешнего устройства" использовать опцию "Выполнение из внешней памяти (EES)".

Пример

Программный код

```
...
N20 DEF INT VAR1 VAR2 VAR3
N30 CASE(VAR1+VAR2-VAR3) OF 7 GOTOF Label_1 9 GOTOF
Label_2 DEFAULT GOTOF Label_3
N40 Label_1: G0 X1 Y1
N50 Label_2: G0 X2 Y2
N60 Label_3: G0 X3 Y3
...
```

Оператор CASE из N30 определяет следующие возможности ветвления программы:

1. Если значение функции вычисления VAR1+VAR2-VAR3 = 7, то перейти на кадр с определением метки перехода "Label_1" (→ N40).
2. Если значение функции вычисления VAR1+VAR2-VAR3 = 9, то перейти на кадр с определением метки перехода "Label_2" (→ N50).
3. Если значение функции вычисления VAR1+VAR2-VAR3 ни 7, ни 9, то перейти на кадр с определением метки перехода "Label_3" (→ N60).

4.1.6 Повторение блока программы (REPEAT, REPEATB, ENDLABEL, P)

Повторение программного блока позволяет повторять уже записанные блоки программы внутри программы в любом составе.

Повторяемые строки или области программы обозначаются метками перехода (метками).

Примечание

Метки перехода (метки)

Метки перехода всегда стоят в начале кадра. Если имеется номер программы, то метка перехода стоит непосредственно за номером кадра.

Для наименования меток перехода действуют следующие правила:

- Число символов:
  - минимум 2
  - максимум 32
- Разрешенными символами являются:
  - буквы
  - цифры
  - символы подчеркивания
- Первыми двумя символами должны быть буквы или символы подчеркивания.
- После имени метки перехода следует двоеточие (":").

Синтаксис

1. Повторение отдельной строки программы:

```
<метка перехода>: ...
...
REPEATB <метка перехода> P=<n>
...
```

2. Повторение области программы между меткой перехода и оператором REPEAT:

```
<метка перехода>: ...
```

```
...
REPEAT <метка перехода> P=<n>
...
```

3. Повторение области между двумя метками перехода:

```
<стартовая метка перехода>: ...
...
<конечная метка перехода>: ...
...
REPEAT <стартовая метка перехода> <конечная метка перехода>
P=<n>
...
```

Примечание

Заключение оператора REPEAT с обеими метками перехода в скобки невозможно. Если <стартовая метка перехода> найдена перед оператором REPEAT и <конечная метка перехода> не достигается до оператора REPEAT, то выполняется повторение между <стартовая метка перехода> и оператором REPEAT.

4. Повторение области между меткой перехода и ENDLABEL:

```
<метка перехода>: ...
...
ENDLABEL: ...
...
REPEAT <метка перехода> P=<n>
...
```

Примечание

Заключение оператора REPEAT с <метка перехода> и ENDLABEL в скобки невозможно. Если <метка перехода> найдена перед оператором REPEAT и ENDLABEL не достигается до оператора REPEAT, то выполняется повторение между <метка перехода> и оператором REPEAT.

Значение

REPEATB:	Команда повторения строки программы
REPEAT:	Команда повторения области программы

<метка перехода>:	<метка перехода> обозначает: • повторяемую строку программы (при REPEATB) или • начало повторяемой области программы (при REPEAT)
	Обозначенная с <метка перехода> строка программы может стоять до или после оператора REPEAT/REPEATB. Поиск сначала выполняется в направлении начала программы. Если метка перехода в этом направлении не найдена, то выполняется поиск в направлении конца программы. **Исключение:** Если необходимо повторить область программы между меткой перехода и оператором REPEAT (см. 2. в Синтаксисе), то обозначенная с <метка перехода> строка программы должна стоять **до** оператора REPEAT, т.к. в этом случае поиск выполняется **только** в направлении начала программы. Если в кадре с <метка перехода> имеются дополнительные операторы, то при каждом повторении они выполняются заново.
ENDLABEL:	Кодовое слово, обозначающее конец повторяемой области программы Если строка с ENDLABEL содержит другие операторы, то при каждом повторении они выполняются заново. ENDLABEL может использоваться в программе многократно.
P:	Адрес для указания числа повторов
<n>:	Число повторов блока программы
	Тип: INT
	Повторяемый блок программы повторяется <n> раз. После последнего повторения программа продолжается на следующей за строкой REPEAT/REPEATB строке. **Примечание:** Если P=<n> не указано, то повторяемый блок программы повторяется точно один раз.

Примеры

Пример 1: Повторение отдельной строки программы

Программный код	Комментарий
N10 POSITION1: X10 Y20	
N20 POSITION2: CYCLE(0,,9,8)	; Позиционный цикл
N30 …	
N40 REPEATB POSITION1 P=5	; Выполнить КАДР N10 пять раз.
N50 REPEATB POSITION2	; Выполнить кадр N20 один раз.
N60 ...	
N70 M30	

Пример 2: Повторение области программы между меткой перехода и оператором REPEAT

Программный код	Комментарий
N5 R10=15	
N10 BEGIN: R10=R10+1	; Ширина

Программный код	Комментарий
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	
N50 X=-R10	
N60 Y=-R10	
N70 Z=10+R10	
N80 REPEAT BEGIN P=4	; Выполнить область N10 до N70 четыре раза.
N90 Z10	
N100 M30	

Пример 3: Повторение области между двумя метками перехода

Программный код	Комментарий
N5 R10=15	
N10 BEGIN: R10=R10+1	; Ширина
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	
N50 X=-R10	
N60 Y=-R10	
N70 END: Z=10	
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	; Выполнить область N10 до N70 три раза.
N110 Z10	
N120 M30	

Пример 4: Повторение области между меткой перехода и ENDLABEL

Программный код	Комментарий
N10 G1 F300 Z-10	
N20 BEGIN1:	
N30 X10	
N40 Y10	
N50 BEGIN2:	
N60 X20	
N70 Y30	
N80 ENDLABEL: Z10	
N90 X0 Y0 Z0	
N100 Z-10	
N110 BEGIN3: X20	
N120 Y30	
N130 REPEAT BEGIN3 P=3	; Выполнить область N110 до N120 три раза.
N140 REPEAT BEGIN2 P=2	; Выполнить область N50 до N80 два раза.
N150 M100	

Программный код	Комментарий
N160 REPEAT BEGIN1 P=2	; Выполнить область N20 до N80 два раза.
N170 Z10	
N180 X0 Y0	
N190 M30	

Пример 5: Фрезерная обработка, обработка позиции сверления с помощью различных технологий

Программный код	Комментарий
N10 ZENTRIERBOHRER()	; Установить центровое сверло.
N20 POS_1:	; Позиции сверления 1
N30 X1 Y1	
N40 X2	
N50 Y2	
N60 X3 Y3	
N70 ENDLABEL:	
N80 POS_2:	; Позиции сверления 2
N90 X10 Y5	
N100 X9 Y-5	
N110 X3 Y3	
N120 ENDLABEL:	
N130 BOHRER()	; Сменить сверло и цикл сверления.
N140 GEWINDE(6)	; Установить метчик M6 и цикл резьбы.
N150 REPEAT POS_1	; Повторить сегмент программы от POS_1 один раз до ENDLABEL.
N160 BOHRER()	; Сменить сверло и цикл сверления.
N170 GEWINDE(8)	; Установить метчик M8 и цикл резьбы.
N180 REPEAT POS_2	; Повторить сегмент программы от POS_2 один раз до ENDLABEL.
N190 M30	

Дополнительная информация

Вложенность

Возможен вложенный вызов повторения блока программы. Каждый вызов занимает один уровень подпрограммы.

M17 / RET

Если при обработке повторения блока программы программируется M17 или RET, то повторение блока программы отменяется. Программа продолжается на следующем за строкой REPEAT кадре.

Индикация программы

На актуальной индикации программы повторение блока программы отображается как отдельный уровень подпрограммы.

Разрыв уровней

Если при обработке блока программы запускается разрыв уровней, то программа продолжается после вызова обработки блока программы.

Пример:

Программный код	Комментарий
N5 R10=15	
N10 BEGIN: R10=R10+1	; Ширина
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	; Разрыв уровней
N50 X=-R10	
N60 Y=-R10	
N70 END: Z10	
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	
N120 Z10	; Продолжить обработку программы.
N130 M30	

Управляющие структуры и повторения блока программы

Возможно комбинированное использование управляющих структур и повторения блока программы. Но не должно быть дублирований. Повторение блока программы должно лежать внутри ветви управляющей структуры или управляющая структура внутри повторения блока программы.

Переходы повторения блока программы

При комбинации переходов и повторения блока программы кадры обрабатываются чисто последовательно. Если, к примеру, осуществляется переход из повторения блока программы, то обработка продолжается до тех пор, пока запрограммированный конец блока программы не будет найден.

Пример:

Программный код
N10 G1 F300 Z-10
N20 BEGIN1:
N30 X=10
N40 Y=10
N50 GOTOF BEGIN2
N60 ENDLABEL:
N70 BEGIN2:
N80 X20
N90 Y30
N100 ENDLABEL: Z10
N110 X0 Y0 Z0
N120 Z-10

Программный код
`N130 REPEAT BEGIN1 P=2`
`N140 Z10`
`N150 X0 Y0`
`N160 M30`

Примечание

Оператор `REPEAT` должен стоять за кадрами перемещения.

Выполнение с внешнего устройства и повторение части программы

В случае внешних программ с повторением части программы повторяемая строка программы (при REPEATB) или начало повторяемой области программы (при REPEAT) должны находиться в памяти догрузки. В ином случае цель перехода не будет найдена и происходит отмена программы с выводом аварийного сообщения 14000.

Примечание

Для выполнения внешних программ без ограничений в части повторений части программы, рекомендуется вместо функции "Выполнение с внешнего устройства" использовать функцию "Выполнение из внешней памяти (EES)".

4.1.7 Управляющие структуры

СЧПУ стандартно обрабатывает кадры ЧПУ в запрограммированной последовательности.

Эта последовательность может изменяться через программирование программных блоков и программных циклов. Эти управляющие структуры программируются с кодовыми словами IF, ELSE, ENDIF, LOOP, FOR, WHILE и REPEAT.

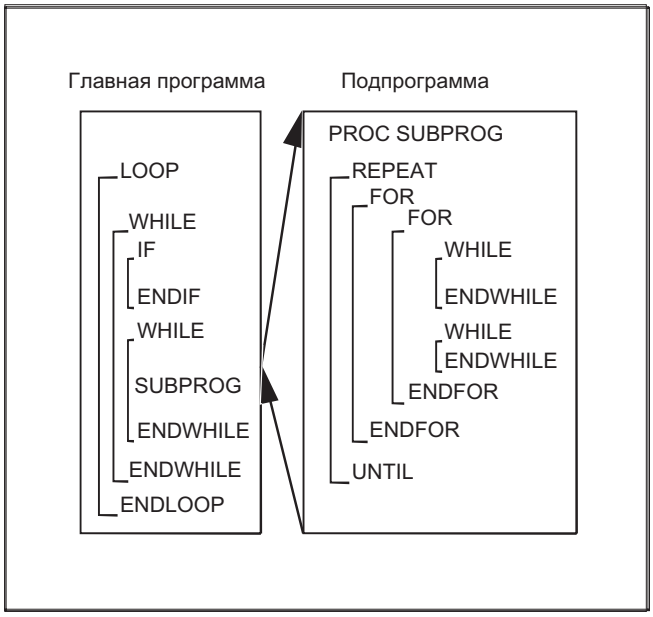
ВНИМАНИЕ
Ошибки программирования Управляющие структуры возможны только внутри командной части программы. Условное или повторное выполнение определений в заголовке программы невозможно. Кодовые слова для управляющих структур, как и цели перехода, не могут перекрываться макросами. Проверка для макроопределения не выполняется.

Активность

Управляющая структура не может использоваться между программами.

Глубина вложенности

Внутри каждого уровня подпрограммы возможна глубина вложенности до 16 управляющих структур.



Динамическая характеристика

В стандартно активном режиме интерпретатора посредством использования переходов в программе может быть достигнуто более быстрое выполнение программы, чем при использовании управляющих структур.

В предварительно скомпилированных циклах различие между переходами в программе и управляющими структурами отсутствует.

Текущая индикация кадров в программных циклах

Если в программном цикле выполняются только кадры предварительной обработки, то в текущей индикации кадров отображается последний кадр главного хода **перед** программным циклом.

Для отображения, например, с целью диагностики, и выполненных кадров предварительной обработки в текущей индикации кадров, необходимо активировать отдельный кадр декодирования SBL2.

Шлифование без кадра главного хода

Если в программном цикле не запрограммирован кадр главного хода, цикл обрабатывается в предварительной обработке, пока не выполнено условие цикла.

Тем самым возникает высокая загруженность, и индикация может оказаться под угрозой.

Способом устранения может быть добавление команды STOPRE или времени ожидания G04 в 0 секунд **в** цикл.

Граничные условия

Индикация кадра

Пропуск кадров с элементами управляющих структур невозможен.

Управляющие структуры и переходы в программе

Метки перехода (метки) не разрешены в кадрах с элементами управляющих структур.

Примечание

Не рекомендуется комбинированное использование управляющих структур и переходов в программе.

Выполнение с внешнего устройства

Для внешних программ с управляющими структурами начало цикла должно находиться в памяти догрузки. В ином случае цель перехода не будет найдена и происходит отмена программы с выводом аварийного сообщения 14000.

Примечание

Для выполнения внешних программ без ограничений в части управляющих структур, рекомендуется вместо функции "Выполнение с внешнего устройства" использовать функцию "Выполнение из внешней памяти (EES)".

Режим интерпретатора

Управляющие структуры обрабатываются интерпретатором. При определении конца цикла, с учетом найденных при этом управляющих структур, выполняется поиск начала цикла. Поэтому в режиме интерпретатора блочная структура программы не проверяется полностью.

Предварительная обработка циклов

При предварительной обработке циклов возможна проверка правильности вложенности управляющих структур.

4.1.7.1 Условный оператора и ветвление (IF, ELSE, ENDIF)

Условный оператор: IF - программный блок - ENDIF

В случае условного оператора стоящий между `IF` и `ENDIF` программный блок выполняется, только если условие выполнено.

Ветвление: IF - программный блок_1 - ELSE - программный блок_2 - ENDIF

В случае ветвления всегда выполняется один из двух программных блоков.

Стоящий между `IF` и `ELSE` программный блок_1 выполняется, если условие выполнено.

Выполняется стоящий между ELSE и ENDIF программный блок_2 выполняется, если условие **не** выполнено.

Примечание

ELSE в синхронных действиях

Кодовое слово ELSE может быть запрограммировано и в синхронных действиях. Это позволяет добавить в синхронное действие операции, которые должны выполняться при невыполненном условии.

Синтаксис

Условный оператор

```
IF <условие>
  Блок программы                          ; Выполнение при: <условие> == TRUE
ENDIF
```

Ветвление

```
IF <условие>
  Программный блок_1                      ; Выполнение при: <условие> == TRUE
ELSE
  Программный блок_2                      ; Выполнение при: <условие> == FALSE
ENDIF
```

Объяснение

IF:	Запускает условный оператор или ветвление.
ELSE:	Запускает альтернативный программный блок.
ENDIF:	Отмечает конец условного оператора или ветвления.
<условие>:	Логическое выражение, обработка которого дает TRUE или FALSE.

Пример: Подпрограмма смены инструмента

Программный код	Комментарий
PROC L6	Программа смены инструмента
N500 DEF INT TNR_AKTUELL	Переменная для активного номера T
N510 DEF INT TNR_VORWAHL	Переменная для предварительно выбранного номера T
	Определить актуальный инструмент
N520 STOPRE	
N530 IF $P_ISTEST	В режиме тестирования программы ...
N540 TNR_AKTUELL = $P_TOOLNO	... из программного контекста считывается "актуальный" инструмент.
N550 ELSE	В ином случае ...

Программный код	Комментарий
N560 TNR_AKTUELL = $TC_MPP6[9998,1]	... выгружается инструмент шпинделя.
N570 ENDIF	
N580 GETSELT(TNR_VORWAHL)	Чтение номера T предварительно выбранного инструмента на шпинделе.
N590 IF TNR_AKTUELL <> TNR_VORWAHL	Если предварительный инструмент это еще не актуальный инструмент, то ...
N600 G0 G40 G60 G90 SUPA X450 Y300 Z300 D0	... подвод к точке смены инструмента ...
N610 M206	... и выполнить смену инструмента.
N620 ENDIF	
N630 M17	

4.1.7.2 Бесконечный программный цикл (LOOP, ENDLOOP)

Бесконечный цикл используется в бесконечных программах. Концом цикла всегда является возврат в начало цикла.

Синтаксис

```
LOOP
...
ENDLOOP
```

Значение

LOOP:	Запускает бесконечный цикл.
ENDLOOP:	Обозначает конец цикла и вызывает возврат на начало цикла.

Пример

Программный код
...
LOOP
MSG("нет активного резца инструмента")
M0
stopre
ENDLOOP
...

4.1.7.3 Счетный цикл (FOR ... TO ..., ENDFOR)

Счетный цикл используется в тех случаях, когда процесс работы должен быть повторен за фиксированное число проходов.

Синтаксис

```
FOR <переменная> = <начальное значение> TO <конечное значение>
...
ENDFOR
```

Значение

FOR:	Запускает счетный цикл.	
ENDFOR:	Обозначает конец цикла и вызывает возврат на начало цикла, пока конечное значение подсчета еще не достигнуто.	
<переменная>:	Счетная переменная, которая увеличивается от начального к конечному значению и приращается при каждом проходе на значение "1".	
	Тип	INT или REAL **Примечание:** Используется тип REAL, если, к примеру, для счетного цикла программируются R-параметры. Если счетная переменная имеет тип REAL, то ее значение округляется до целочисленного значения.
<начальное значение>:	Начальное значение подсчета Условие: Начальное значение должно быть меньше конечного значения.	
<конечное значение>:	Конечное значение подсчета	

Примеры

Пример 1: Переменная INTEGER или R-параметр как счетная переменная

Переменная INTEGER как счетная переменная:

Программный код	Комментарий
`DEF INT iVARIABLE1`	
`R10=R12-R20*R1 R11=6`	
`FOR iVARIABLE1= R10 TO R11`	; Счетная переменная = переменная INTEGER
`R20=R21*R22+R33`	
`ENDFOR`	
`M30`	

Параметр R как счетная переменная:

Программный код	Комментарий
`R11=6`	
`FOR R10=R12-R20*R1 TO R11`	; Счетная переменная = R-параметр (переменная Real)
`R20=R21*R22+R33`	
`ENDFOR`	
`M30`	

Пример 2: Изготовление фиксированного числа деталей

Программный код	Комментарий
`DEF INT STUECKZAHL`	; Определяет переменную типа `INT` с именем `"STUECKZAHL"`.
`FOR STUECKZAHL = 0 TO 100`	; Запускает счетный цикл. Переменная `"STUECKZAHL"` увеличивается от начального значения "0" до конечного значения "100".
`G01 …`	
`ENDFOR`	; Конец счетного цикла.
`M30`	

4.1.7.4 Программный цикл с условием в начале цикла (WHILE, ENDWHILE)

В цикле WHILE условие стоит в начале цикла. Пока условие выполнено, цикл WHILE выполняется.

Синтаксис

```
WHILE <условие>
...
ENDWHILE
```

Значение

`WHILE`:	Запускает программный цикл.
`ENDWHILE`:	Обозначает конец цикла и вызывает возврат на начало цикла.
<условие>:	Условие, которое должно быть выполнено, чтобы выполнить цикл WHILE.

Пример

Программный код	Комментарий
…	
`WHILE $AA_IW[BOHRACHSE] > -10`	; Вызов цикла `WHILE` при следующем условии: актуальное заданное значение `WCS` для оси сверления должно быть больше -10.
`G1 G91 F250 AX[BOHRACHSE] = -1`	
`ENDWHILE`	
…	

4.1.7.5 Программный цикл с условием на конце цикла (REPEAT, UNTIL)

В цикле REPEAT условие стоит в конце цикла. Цикл REPEAT проходится один раз и повторяется до тех пор, пока выполнено условие.

Синтаксис

```
REPEAT
...
UNTIL <условие>
```

Значение

REPEAT:	Запускает программный цикл.
UNTIL:	Обозначает конец цикла и вызывает возврат на начало цикла.
<условие>:	Условие, которое должно быть выполнено, чтобы цикл WHILE более не выполнялся.

Пример

Программный код	Комментарий
...	
REPEAT	; Вызов цикла REPEAT.
...	
UNTIL ...	; Проверка, выполнено ли условие.
...	

4.1.7.6 Пример программы со вложенными управляющими структурами

Программный код	Комментарий
LOOP	
IF NOT $P_SEARCH	; IF нет поиска кадра
G1 G90 X0 Z10 F1000	
WHILE $AA_IM[X] <= 100	; WHILE (заданное значение X-ось <= 100)
G1 G91 X10 F500	; схема сверления
Z-5 F100	
Z5	
ENDWHILE	
ELSE	; ELSE поиск кадра
MSG("При поиске сверление не выполняется")	
ENDIF	; ENDIF
$A_OUT[1] = 1	; следующая кондукторная плита
G4 F2	
ENDLOOP	
M30	

4.1.8 Выходящая за рамки канала координация программ (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)

Любой канал ЧПУ принципиально может выполнять запущенную в нем программу независимо от других каналов своей группы режимов работы (ГРР). Но если в изготовлении детали одновременно задействовано несколько программ в нескольких каналах ГРР, программные процессы в разных каналах необходимо координировать посредством следующих координирующих команд.

Условие

Участвующие в координации программы каналы должны относиться к **одной и той же** группе режимов работы (ГРР).

MD10010 $MC_ASSIGN_CHAN_TO_MODE_GROUP[<канал>] = <номер ГРР>

Имя канала вместо номера канала

В качестве параметра предварительно определенных процедур координации программ можно вместо номеров каналов использовать и внесенные в MD20000 $MC_CHAN_NAME[<индекс канала>] имена каналов. Использование имен каналов в программах ЧПУ требует разблокирования:

MD10280 $MN_PROG_FUNCTION_MASK, бит 1 = TRUE

Примечание

Минимальный интервал между командами

Между командами `INIT`, `START`, `WAITE`, `WAITM`, `SETM`, `CLEARM` и командой `WAITMC` необходимо выдерживать интервал не менее двух кадров перемещения. WAITMC — исполняемый кадр, однако для оптимизации он смещается в предыдущий кадр и потом удаляется как кадр. Для примера: `SETM` — неисполняемый кадр и смещается в следующий кадр, так что в случае расстояния в один кадр обе команды находились бы в среднем кадре. Поскольку возможен только один кадр, в случае расстояния в один кадр для `WAITMC` оптимизация не выполняется.

Для этого программа затормаживается, и обработка ненадолго останавливается.

Синтаксис

```
INIT(<ChanNr>, <Prog>, <AckMode>)
START(<ChanNr>, <ChanNr>, ...)
WAITM(<MarkNr>, <ChanNr>, <ChanNr>, ...)
WAITE(<ChanNr>, <ChanNr>, ...)
WAITMC(<MarkNr>, <ChanNr>, <ChanNr>, ...)
SETM(<MarkNr>, <MarkNr>, ...)
CLEARM(<MarkNr>, <MarkNr>, ...)
```

Объяснение

INIT():	Предварительно определенная процедура для выбора программы ЧПУ, которая должна выполняться в указанном канале		
START():	Предварительно определенная процедура пуска выбранной в соответствующем канале программы		
WAITM():	Предопределенная процедура ожидания достижения метки ожидания в указанных каналах В собственном канале указанная метка ожидания устанавливается через WAITM. Предшествующий кадр завершается с точным остановом. После синхронизации метка ожидания удаляется. Одновременно возможна установка не более 10 меток на канал.		
WAITE():	Предварительно определенная процедура для ожидания завершения программы в одном или нескольких других каналах		
WAITMC():[1)]	Предопределенная процедура ожидания достижения метки ожидания в указанных каналах В отличие от WAITM торможение осей до точного останова инициируется, только если другие каналы еще не достигли метки ожидания.		
SETM():[1)]	Предварительно определенная процедура для установки одной или нескольких меток ожидания для координации каналов Это не затрагивает обработку в собственном канале. SETM остается действительным и после сброса (Reset) канала и старта ЧПУ.		
CLEARM():[1)]	Предварительно определенная процедура для удаления одной или нескольких меток ожидания координации каналов Это не затрагивает обработку в собственном канале. CLEARM() удаляет все метки ожидания в канале. CLEARM(0) удаляет только метку ожидания "0". CLEARM остается действительным и после сброса (Reset) канала и старта ЧПУ.		
<ChanNr>:	номер канала Номер собственного канала указывать не требуется.		
	Тип:	INT	
<Prog>:	Абсолютное или относительное указание пути (**опция**) + имя программы		
	Тип:	STRING	
	Информация по указанию пути представлена в Руководство по программированию — Программирование ЧПУ, глава "Адресация файлов программной памяти" (Страница 574).		
<AckMode>:	Режим квитирования (**опция**)		
	Тип:	CHAR	
	Значения:	"N"	Без квитирования Обработка программы продолжается после отправки команды. Отправитель не уведомляется о том, что команда не может быть выполнена успешно.
		"S"	Синхронное квитирование Выполнение программы приостанавливается до тех пор, пока принимающий компонент не подтвердит команду. При положительном квитировании выполняется следующая команда. При отрицательном квитировании выводится сообщение об ошибке.

<MarkNr>:	Номер метки ожидания **Указание** В многоканальной системе в распоряжении есть не более 100 меток ожидания (метка ожидания 0 ... 99). В одноканальной системе в распоряжении есть только метка ожидания 0.
1) Для спец. для пользователя коммуникации и/или координации каналов можно вставлять метки ожидания посредством SETM / CLEARM и без использования условной команды ожидания WAITMC. При этом метки ожидания сохраняют свои значения и после сброса (Reset) канала и старта ЧПУ.	

Примеры

START посредством имен каналов из MD20000

- Параметрирование
```
MD10280 $MN_PROG_FUNCTION_MASK, бит 1 = TRUE
$MC_CHAN_NAME[ 0 ] = "BEARBEITUNG"  ; Имя канала 1
$MC_CHAN_NAME[ 1 ] = "ZUFUEHRUNG"   ; Имя канала 2
```
- Программирование

Программный код	Комментарий
START(BEARBEITUNG)	; Пуск канала 1
START(ZUFUEHRUNG)	; Пуск канала 2

START через локальные "имена каналов" и переменную пользователя

Программный код	Комментарий
DEF INT MASCHINE = 1	; Определение переменной пользователя для канала 1
DEF INT LADER = 2	; Определение переменной пользователя для канала 2
...	
START(MASCHINE)	; Пуск канала 1
START(LADER)	; Пуск канала 2

START через локальные "имена каналов", переменную пользователя и спараметрированные имена каналов

Программный код	Комментарий
DEF INT chanNo1	; Определение переменной пользователя для канала 1
DEF INT chanNo2	; Определение переменной пользователя для канала 2
chanNo1 = CHAN_1	; Присваивание спараметрированного имени канала каналу 1
chanNo2 = CHAN_2	; Присваивание спараметрированного имени канала каналу 2
...	
START(chanNo1)	; Пуск канала 1
START(chanNo2)	; Пуск канала 2

Команда INIT с абсолютным указанием пути

Выбор программы /_N_MPF_DIR/_N_ABSPAN1_MPF в канале 2.

Программный код

```
INIT(2,"/_N_WKS_DIR/_N_WELLE1_WPD/_N_ABSPAN1_MPF")
```

Команда INIT с именем программы

Выбор программы с именем "MYPROG". СЧПУ ищет программу по маршруту поиска.

Программный код

```
INIT(2,"MYPROG")
```

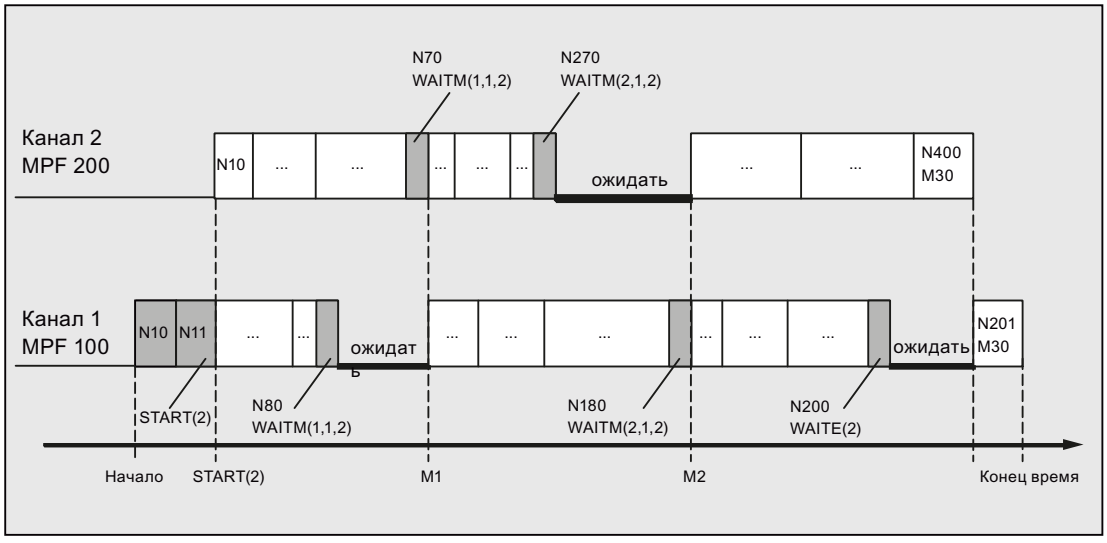
Координация программ с WAITM

- Канал 1: Программа /_N_MPF_DIR/_N_MPF100_MPF уже выбрана и запущена.

Программный код	Комментарий
	; Программа MPF100
N10 INIT(2,"MPF200","N")	; Выбор программы MPF200, канал 2
N11 START(2)	; Пуск канала 2
...	
N80 WAITM(1,1,2)	; Ожидание метки WAIT 1 в канале 1 и 2
N81 ...	; Канал 1, N81 и канал 2, N71 начинаются ; синхронно
...	
N180 WAITM(2,1,2)	; Ожидание метки WAIT 2 в канале 1 и 2
N181 ...	; Канал 1, N181 и канал 2, N271 начинаются ; синхронно
...	
N200 WAITE(2)	; Ожидание конца программы в канале 2
N201 ...	; N201 начинается только после конца программы ; MPF200 в канале 2 начата
N201 M30	; Конец программы канала 1

- Канал 2: В канале 1 через кадр N10 и N20 выбирается и запускается программа MPF200_MPF для канала 2.

Программный код	Комментарий
;$PATH=/_N_MPF_DIR	; Программа MPF200
...	
N70 WAITM(1,1,2)	Ожидание метки WAIT 1 в канале 1 и 2
N71 ...	; Канал 1, N81 и канал 2, N71 начинаются ; синхронно
...	
N270 WAITM(2,1,2)	Ожидание метки WAIT 2 в канале 1 и 2
N271 ...	; Канал 1, N181 и канал 2, N271 начинаются ; синхронно
...	
N400 M30	Конец программы канала 2



Граничные условия

Асинхронное начало выполнения последующих кадров после меток WAIT

При координации каналов с помощью меток WAIT может произойти асинхронное начало выполнения последующих кадров. Это происходит, если непосредственно перед достижением общей метки WAIT в одном из синхронизируемых каналов вызывается действие, приводящее к удалению остатка пути с не явным репозиционированием (REPOSA) в этом канале.

Допущение: Текущее соотнесение осей в каналах 1 и 2

- Канал 1: оси X1 и U
- Канал 2: ось X2

Таблица 4-2 Процесс во времени в каналах 1 и 2

Канал 1	Канал 2	Описание
...	...	Любая обработка в канале 1 и 2
`N100 WAITM(20,1,2)`		Канал 1: достигает метки `WAIT` и ожидает синхронизации с каналом 2
Начало обработки GETD(U): • *Переход оси* • *Стирание остатка пути* • *REPOSA* *Конец*	`N200 GETD(U)`	Канал 2: запрашивает ось U из канала 1 Канал 1: обработка `GET(U)` в фоновом режиме
	`N210 WAITM(20,1,2)`	Канал 2: достигает метки `WAIT`. ⇒ На этом синхронизация каналов 1 и 2 завершена
	`N220 G0 X2=100`	Канал 2: начало обработки N220
`N110 G0 X1=100`		Канал 1: Смещенное по времени начало обработки N110

4.1.9 Техника макросов (DEFINE ... AS)

ВНИМАНИЕ
Техника макросов повышает комплексность программирования. С помощью техники макросов язык программирования СЧПУ может быть сильно изменен. При использовании техники макросов необходимо соблюдать осторожность.

Макросом обозначается объединение отдельных операторов в новый общий оператор с собственным именем. При вызове макроса при выполнении программы запрограммированные под макроименем операторы выполняются последовательно.

Согласно области действия, т. е. области, в которой действует макроопределение, существуют следующие категории макросов:

- Локальные макросы
  Локальные макросы — это макросы, которые определены в начале программы ЧПУ, которая на момент выполнения не является главной программой. Они создаются при вызове программы ЧПУ и удаляются при сбросе (Reset) конца программы или следующем запуске СЧПУ. Обращение к локальным макросам возможно только внутри программы ЧПУ, в которой они определены.
- Глобальные программные макросы
  Глобальные программные макросы — это макросы, которые определены в начале программы ЧПУ, используемой как главная программа. Они создаются при вызове программы ЧПУ и удаляются при сбросе (Reset) конца программы или следующем запуске СЧПУ. Обращение к глобальным программным макросам возможно в главной программе и во всех подпрограммах.

  Примечание

  Доступность глобальных программных макросов

  Определенные в главной программе глобальные программные макросы доступны в подпрограммах только при следующих установленных машинных данных:

  MD11120 $MN_LUD_EXTENDED_SCOPE = 1

  При MD11120 = 0 определенные в главной программе глобальные программные макросы доступны только в главной программе.

- Глобальные макросы
  Глобальные макросы — это спец. для ЧПУ или канала глобальные макросы, определенные в файле определений (макрофайл) и сохраняющиеся после сброса (Reset) конца программы или следующего запуска СЧПУ. Глобальные макросы могут быть вызваны и выполнены в любой главной программе или подпрограмме.

  Примечание

  Для использования макросов **внешнего** макрофайла в программе ЧПУ необходимо загрузить макрофайл в ЧПУ.

Перед использованием макросы должны быть определены. Соблюдать следующие правила:

- В макросе могут быть определены любые идентификаторы, функции G, M, H и имена подпрограмм L.
- Макроопределение может происходить в начале программы или в отдельном файле определений (макрофайл).
- Локальные глобальные программные макросы определены в начале программы.
- Глобальные макросы должны быть определены в макрофайле, например, _N_DEF_DIR/_N_UMAC_DEF.
- Макросы G-команд могут быть определены как глобальные макросы.
- Функции H и L могут программироваться 2-позиционно.
- Команды M и G могут программироваться 3-позиционно.

Примечание

Запрещено перезаписывать кодовые слова и зарезервированные имена с помощью макросов. Это действительно и для всех целей перехода внутри команды GOTO и для всех кодовых слов в программных циклах, таких как FOR, WHILE, LOOP, REPEAT.

Синтаксис

Определение макроса:
`DEFINE <Macro_Name> AS <Operation_1> <Operation_2> ...`

Вызов в программе ЧПУ:
<Макрос_имя>

Объяснение

`DEFINE ... AS`:	Комбинация кодовых слов для определения макроса
<Макрос_имя>:	Имя макроса В качестве макроимен разрешены только идентификаторы. С помощью макроимени макрос вызывается из программы ЧПУ.
<Операция_1>:	Первая операция программирования в макросе
<Операция_2>:	Вторая операция программирования в макросе

Примеры

Пример 1: Макроопределение в начале программы

Программный код	Комментарий
`DEFINE LINIE AS G1 G94 F300`	; макроопределение
...	
`N70 LINIE X10 Y20`	; макровызов

Программный код	Комментарий
...	

Пример 2: Макроопределения в макрофайле

Программный код	Комментарий
`DEFINE M6 AS L6`	; При смене инструмента вызывается подпрограмма, берущая на себе необходимую передачу данных. В подпрограмме выводится сама функция M смены инструмента (например, `M106`).
`DEFINE G81 AS DRILL(81)`	; Эмуляция `DIN-G`-команды.
`DEFINE G33 AS M333 G333`	; При резьбонарезании запрашивается синхронизация с `PLC`. Первоначальная `G`-команда 33 была через `MD` переименована в `G333`, для пользователя программирование остается прежним.

Пример 3: Внешний макрофайл

После загрузки внешнего макрофайла в СЧПУ, макрофайл должен быть загружен в ЧПУ. Только после этого можно использовать макросы в программе ЧПУ.

Программный код	Комментарий
`%_N_UMAC_DEF`	
`;$PATH=/_N_DEF_DIR`	; Спец. для заказчика макросы
`DEFINE PI AS 3.14`	
`DEFINE TC1 AS M3 S1000`	
`DEFINE M13 AS M3 M7`	; Правое вращение шпинделя, включение СОЖ
`DEFINE M14 AS M4 M7`	; Левое вращение шпинделя, включение СОЖ
`DEFINE M15 AS M5 M9`	; Останов шпинделя, выключение СОЖ
`DEFINE M6 AS L6`	; Вызов программы смены инструмента
`DEFINE G80 AS MCALL`	; Отмена цикла сверления
`M30`	

4.2 Техника подпрограмм

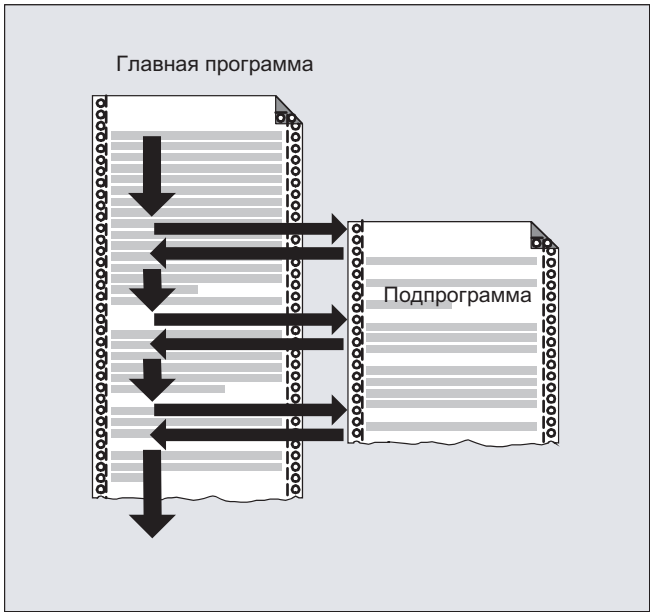
4.2.1 Общая информация

4.2.1.1 Подпрограмма

Обозначение "Подпрограмма" относится к тому времени, когда программы обработки деталей были жестко разделены на главные и подпрограммы. При этом главными программами были программы обработки деталей, которые выбирались на СЧПУ для выполнения и после запускались. Подпрограммами были программы обработки деталей, вызываемые из главной программы.

Такого жесткого разделения в сегодняшнем языке ЧПУ SINUMERIK больше нет. Любая программа обработки детали в принципе может быть выбрана как главная программа и запущена или вызвана как подпрограмма из другой программы обработки детали.

Таким образом, в дальнейшем подпрограммой будет обозначаться программа обработки детали, которая вызывается из другой программы обработки детали.



Использование

Как и во всех высокоуровневых языках программирования, и в языке ЧПУ подпрограммы используются для того, чтобы поместить многократно используемые программные блоки в самостоятельные, завершенные программы.

Преимуществами подпрограмм являются:

- Увеличение наглядности и читабельности программ
- Увеличение качества через повторное использование протестированных блоков программы
- Возможность создания специфических библиотек обработки
- Экономия места в памяти

4.2.1.2 Имена подпрограмм

Правила именования

Имя подпрограммы можно выбирать произвольно, соблюдая следующие правила:

- Разрешенные символы:
  - Буквы: A ... Z, a ... z
  - Цифры: 0 ... 9
  - Знак подчеркивания: _
- Первыми двумя символами должны быть две буквы или знак подчеркивания с буквой после него.

  Примечание

  Если это условие выполнено, программа ЧПУ может быть вызвана из другой программы как подпрограмма путем указания имени программы . Если имя программы начинается с цифр, вызов подпрограммы возможен только через оператор CALL.

- Максимальная длина: 24 символа

Примечание

Прописные / строчные буквы

В языке ЧПУ SINUMERIK **нет** различия между прописными и строчными буквами.

Примечание

Недопустимые имена программ

Во избежание проблем с приложениями Windows **запрещено** использовать следующие имена программ:

- CON, PRN, AUX, NUL
- COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9
- LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8, LPT9

Расширения внутри СЧПУ

К присвоенному при создании программы имени подпрограммы СЧПУ добавляет префикс и постфикс:

- Префикс: `_N_`
- Постфикс: `_SPF`

Использование имени программы

При использовании имени программы, например, для вызова подпрограммы, возможны любые комбинации префикса, имени программы и постфикса.

Пример:

Подпрограмма с именем программы SUB_PROG может вызываться через следующие идентификаторы:

1. SUB_PROG
2. _N_SUB_PROG
3. SUB_PROG_SPF
4. _N_SUB_PROG_SPF

Идентичность имен главных программ и подпрограмм

Если существуют главная программа (.MPF) и подпрограмма (.SPF) с одинаковым именем программы, то для однозначной идентификации при использовании имени программы в программе ЧПУ необходимо указывать и соответствующее расширение файла. В противном случае используется программа с указанным именем программы, которая будет найдена по маршруту поиска первой.

Имя файла и имя программы

Имя файла и имя содержащейся в нем программы должны быть одинаковыми.

В файле содержится только одна программа.

4.2.1.3 Вложенность подпрограмм

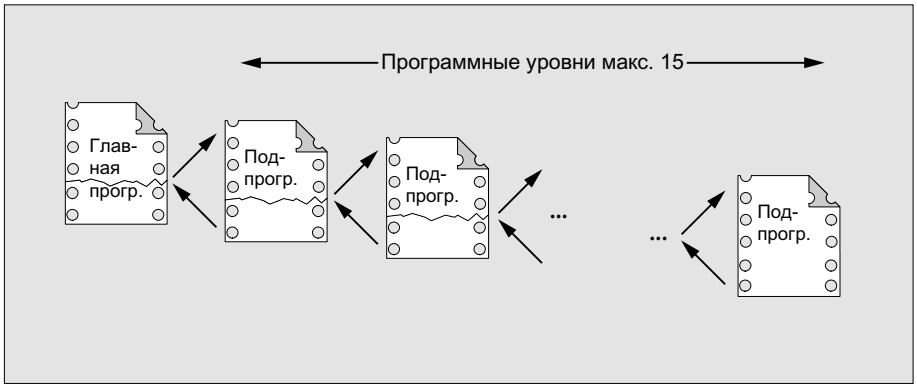
Главная программа может вызывать подпрограммы, которые в свою очередь вызывают подпрограммы. Таким образом, процессы программ вложены друг в друга. При этом каждая программа выполняется на своем программном уровне.

Глубина вложенности

Язык ЧПУ в настоящее время предлагает 16 программных уровней. Главная программа всегда выполняется на самом верхнем программном уровне 0. Подпрограмма всегда выполняется на следующем за таковым в вызове более низком программном уровне. Таким образом, программный уровень 1 это первый уровень подпрограмм.

Подразделение программных уровней:

- Программный уровень 0: уровень главной программы
- Программный уровень 1 - 15: уровень подпрограмм 1 - 15



Обработчики прерываний (ASUP)

Если в рамках обработчика прерываний вызывается подпрограмма, то она выполняется не на актуальном активном в канале программном уровне (n), а также на следующем более низком программном уровне (n+1). Для обеспечения этой возможности и на самом низком программном уровне, в комбинации с обработчиками прерываний предлагаются 2 дополнительных программных уровня (16 и 17).

Если требуется более 2 программных уровней, то это должно быть явно учтено в структурировании выполняемой в канале программы обработки детали. Т.е. в этом случае может быть задействовано макс. столько программных уровней, чтобы осталось еще достаточно программных уровней для обработки прерываний.

Если, к примеру, обработке прерываний требуется 4 программных уровня, то программа обработки детали должна быть структурирована таким образом, чтобы она занимала макс. программный уровень 13. Если после происходит прерывание, то для него доступны 4 требуемых программных уровня (14 до 17).

Циклы Siemens

Для циклов Siemens требуется 3 программных уровня. Поэтому вызов цикла Siemens должен быть выполнен самое позднее на:

- Выполнение программы обработки детали: программный уровень 12
- Обработчик прерываний: программный уровень 14

4.2.1.4 Маршрут поиска

При вызове подпрограммы без указания пути система управления выполняет поиск в имеющейся программной памяти в предопределенной последовательности (см. "Маршрут поиска при вызове подпрограммы (Страница 579)").

4.2.1.5 Формальные и фактические параметры

Речь о формальных и фактических параметрах связана с определением и вызовом подпрограмм с передачей параметров.

Формальный параметр

При определении подпрограммы, передаваемые подпрограмме параметры, т.н. формальные параметры, должны быть определены с типом и именем параметра.

Тем самым формальные параметры определяют интерфейс подпрограммы.

Пример:

Программный код	Комментарий
`PROC KONTUR (REAL X, REAL Y)`	; Формальные параметры: `X` и `Y` оба типа `REAL`
`N20 X1=X Y1=Y`	; Перемещение оси `X1` на позицию `X` и оси `Y1` на позицию `Y`
...	
`N100 RET`	

Фактический параметр

При вызове подпрограммы ей должны быть переданы абсолютные значения или переменные, т.н. фактические параметры.

Тем самым фактические параметры при вызове заполняют интерфейс подпрограммы актуальными значениями.

Пример:

Программный код	Комментарий
`N10 DEF REAL BREITE`	; определение переменных
`N20 BREITE=20.0`	; Присваивание переменной
`N30 KONTUR(5.5, BREITE)`	; Вызов подпрограммы с фактическими параметрами: 5.5 и `BREITE`
...	
`N100 M30`	

4.2.1.6 Передача параметров

Определение подпрограммы с передачей параметров

Определение подпрограммы с передачей параметров осуществляется с кодовым словом `PROC` и полным перечнем ожидаемых от подпрограммы параметров.

Неполная передача параметров

При вызове подпрограммы не всегда должны быть явно переданы все определенные на интерфейсе подпрограммы параметры. Если параметр опускается, то для этого параметра передается стандартное значение "0".

Но для однозначного обозначения последовательности параметров, запятые в качестве разделительных символов параметров все же должны указываться всегда. Исключением является последний параметр. Если он опускается при вызове, может быть опущена и последняя запятая.

Пример:

Подпрограмма:

Программный код	Комментарий
`PROC SUB_PROG (REAL X, REAL Y, REAL Z)`	; Формальные параметры: X, Y и Z
...	
`N100 RET`	

Главная программа:

Программный код	Комментарий
`PROC MAIN_PROG`	
...	
`N30 SUB_PROG(1.0,2.0,3.0)`	; Вызов подпрограммы с полной передачей параметров: X=1.0, Y=2.0, Z=3.0
…	
`N100 M30`	

Примеры вызова подпрограммы в `N30` с неполной передачей параметров:

```
N30 SUB_PROG( ,2.0,3.0)          ; X=0.0, Y=2.0, Z=3.0
N30 SUB_PROG(1.0, ,3.0)          ; X=1.0, Y=0.0, Z=3.0
N30 SUB_PROG(1.0,2.0)            ; X=1.0, Y=2.0, Z=0.0
N30 SUB_PROG( , ,3.0)            ; X=0.0, Y=0.0, Z=3.0
N30 SUB_PROG( , , )              ; X=0.0, Y=0.0, Z=0.0
```

ВНИМАНИЕ

Передача параметров, вызываемых по ссылке

Параметры, передаваемые по ссылке, не могут быть опущены при вызове подпрограммы.

ВНИМАНИЕ

Тип данных AXIS

Параметры тира данных AXIS не могут быть опущены при вызове подпрограммы.

Проверка передаваемых параметров

Через системную переменную $P_SUBPAR [n] где n = 1, 2, ... в подпрограмме можно проверить, был ли параметр явно передан или опущен. Индекс n относится к последовательности формальных параметров. Индекс n = 1 относится к 1-му формальному параметру, индекс n = 2 ко 2-му формальному параметру и т.д.

Следующий сегмент программы показывает на примере для 1-го формального параметра, как может быть реализована проверка:

Программирование	Комментарий
`PROC SUB_PROG (REAL X, REAL Y, REAL Z)`	; Формальные параметры: `X, Y и Z`
`N20 IF $P_SUBPAR[1]==TRUE`	; Проверка 1-го формального параметра `X`.
...	; Эти операции выполняются, если формальный параметр `X` был передан явно.
`N40 ELSE`	
...	; Эти операции выполняются, если формальный параметр `X` не был передан.
`N60 ENDIF`	
...	; Общие операции
`N100 RET`	

4.2.2 Определение подпрограммы

4.2.2.1 Подпрограмма без передачи параметров

При определении подпрограмм без передачи параметров строка определения в начале программы не нужна.

Синтаксис

`[PROC` <имя программы>]
...

Значение

`PROC`:	Оператор определения в начале программы
<имя программы>:	Имя программы

Пример

Пример 1: Подпрограмма с оператором `PROC`

Программный код	Комментарий
`PROC SUB_PROG`	; Строка определения
`N10 G01 G90 G64 F1000`	

Программный код	Комментарий
N20 X10 Y20	
...	
N100 RET	; возврат из подпрограммы

Пример 2: Подпрограмма без оператора PROC

Программный код	Комментарий
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; возврат из подпрограммы

См. также

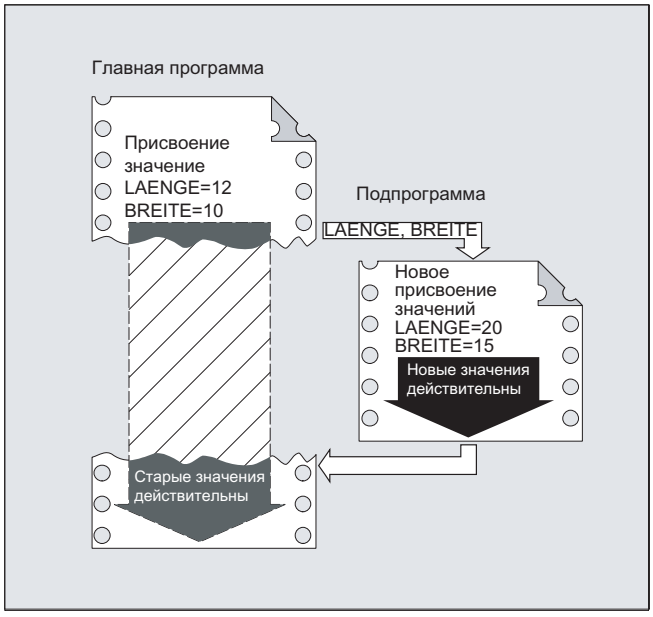
Вызовы подпрограмм без передачи параметров (Страница 542)

4.2.2.2 Подпрограмма с передачей параметров по значению (PROC)

Определение подпрограммы с передачей параметров, вызываемых по значению, выполняется кодовым словом PROC с последующим именем программы и полным перечнем параметров с типом и именем. Оператор определения должен стоять в первой строке программы.

Параметры, вызываемые по значению

Вызывающая программа при передаче параметра, вызываемого по значению, передает в подпрограмму только значение переменной. Так подпрограмма не получает прямого доступа к переменной. В результате при изменении значения параметра изменяется только значение, видимое в подпрограмме. Значение переменной, определенной в вызывающей программе, остается неизменным. Так передача параметров, вызываемых по значению, не оказывает обратного воздействия на вызывающую программу.



Синтаксис

```
PROC <имя программы> (<тип параметра> <имя параметра>=<Init_Wert>, ...)
```

Примечание

Всего может быть передано до 127 параметров.

Объяснение

PROC:	Оператор определения в начале программы
<имя программы>:	Имя программы
<тип параметра>:	Тип данных параметра (например, REAL, INT, BOOL)
<имя параметра>:	Имя параметра
<иниц_значение>:	Опциональное значение для инициализации параметра (**опция**) Если при вызове подпрограммы параметр не указывается, то параметру присваивается инициализирующее значение.

Примеры

Пример 1

Определение подпрограммы SUB_PROG с тремя параметрами типа REAL со значениями по умолчанию:

Программный код

```
PROC SUB_PROG(REAL LENGTH=10.0, REAL WIDTH=20.0, REAL HIGHT=30.0)
```

Пример 2

Различные варианты вызова

Программный код

```
PROC MAIN_PROG
  REAL PAR_1 = 100
  REAL PAR_2 = 200
  REAL PAR_3 = 300
 ; варианты вызова
  SUB_PROG
  SUB_PROG(PAR_1, PAR_2, PAR_3)
  SUB_PROG(PAR_1)
  SUB_PROG(PAR_1, , PAR_3)
  SUB_PROG( , , PAR_3)
N100 RET
```

См. также

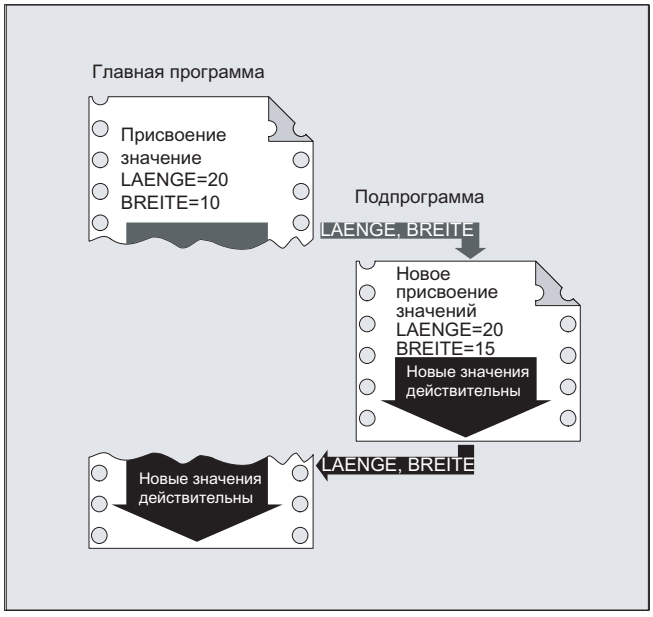
Вызов подпрограммы с передачей параметров (EXTERN) (Страница 544)

4.2.2.3 Подпрограмма с передачей параметров по ссылке (PROC, VAR)

Определение подпрограммы с передачей параметров по ссылке осуществляется с кодовым словом `PROC`, с последующим именем программы и полным перечнем параметров с кодовым словом `VAR`, с типом и именем. Оператор определения должен стоять в первой строке программы. В качестве параметра можно также передавать ссылки на поля.

Параметры, вызываемые по ссылке

Вызывающая программа при передаче параметра по ссылке передает в подпрограмму не значение переменной, а ссылку (указатель) на переменную. Таким образом, подпрограмма получает прямой доступ к переменной. В результате при изменении значения параметра изменяется не только значение, видимое в подпрограмме, но и значение переменной, определенной в вызывающей программе. Таким образом, передача параметров по ссылке оказывает обратное воздействие на вызывающую программу, в том числе, после завершения подпрограммы.



Примечание

Передача параметров по ссылке необходима только тогда, когда переданная переменная была локально определена в вызывающей программе (LUD). Глобальные переменные канала или ЧПУ не должны передаваться, т.к. к ним возможен и прямой доступ из подпрограммы.

Синтаксис

```
PROC <имя программы> (VAR <тип параметра> <имя параметра>, ...)
PROC <имя программы> (VAR <тип поля> <имя поля> [<m>,<n>,<o>], ...)
```

Примечание

Всего может быть передано до 127 параметров.

Объяснение

PROC:	Оператор определения в начале программы
VAR:	Кодовое слово для передачи параметров по ссылке
<имя программы>:	Имя программы
<тип параметра>:	Тип данных параметра (например, REAL, INT, BOOL)
<имя параметра>:	Имя параметра
<тип поля>:	Тип данных элементов поля (например, REAL, INT, BOOL)
<имя поля>:	Имя поля

[<m>,<n>,<o>]:	Размер поля Сейчас возможны макс. 3-мерные поля:	
	<m>:	Размер поля для 1-й размерности
	<n>:	Размер поля для 2-й размерности
	<o>:	Размер поля для 3-й размерности

Примечание

- Указанное после кодового слова PROC имя параметра должно совпадать с присвоенным на интерфейсе именем программы.
- С помощью полей неопределенной длины как формальных параметров подпрограммы могут обрабатывать поля переменной длины. Для этого при определении, например, двухмерного поля как формального параметра длина 1-й размерности не указывается. Но запятая должна быть записана.
  Пример: PROC <имя программы> (VAR REAL FELD[,5])

Пример

Определение подпрограммы с двумя параметрами как ссылка на тип REAL:

Программный код

```
; Параметр 1: Ссылка на тип: REAL, имя: LAENGE
; Параметр 2: Ссылка на тип: REAL, имя: BREITE
PROC SUB_PROG(VAR REAL LAENGE, VAR REAL BREITE)
```

См. также

Вызов подпрограммы с передачей параметров (EXTERN) (Страница 544)

4.2.2.4 Сохранить модальные функции G (SAVE)

Следствием атрибута SAVE является то, что активные перед вызовом подпрограммы модальные G-команды сохраняются и после завершения подпрограммы снова активируются.

ВНИМАНИЕ
Прерывание режима управления траекторией Если при активном режиме управления траекторией вызывается подпрограмма с атрибутом SAVE, то режим управления траекторией прерывается на конце подпрограммы (возврат).

Синтаксис

PROC <имя подпрограммы> SAVE

Объяснение

SAVE:	Сохранение модальных G-команд перед вызовом подпрограммы и восстановление после завершения подпрограммы

Пример

В подпрограмме KONTUR действует модальная команда G91 (составной размер). В главной программе действует модальная команда G90 (абсолютный размер). За счет определения подпрограммы с SAVE после завершения подпрограммы в главной программе снова действует G90.

Определение подпрограммы:

Программный код	Комментарий
`PROC KONTUR (REAL WERT1) SAVE`	; Определение подпрограммы с параметром `SAVE`
`N10 G91 ...`	; Модальная `G`-команда `G91`: Составной размер
`N100 M17`	; Конец подпрограммы

Главная программа:

Программный код	Комментарий
`N10 G0 X... Y... G90`	; Модальная `G`-команда `G90`: Абсолютный размер
`N20 ...`	
...	
`N50 KONTUR (12.4)`	;Вызов подпрограммы
`N60 X... Y...`	; Модальная `G`-команда `G90` реактивирована через `SAVE`

Граничные условия

Фреймы

Поведение фреймов касательно подпрограмм с атрибутом SAVE зависит от типа фрейма и может быть установлено через машинные данные.

4.2.2.5 Подавление покадровой обработки (SBLOF, SBLON)

При активной покадровой обработке пользователь может полностью или частично обработать программу ЧПУ, не останавливаясь. Подавление покадровой обработки выполняется командой SBLOF, повторное включение выполняется командой SBLON.

Подавление покадровой обработки для всей программы ЧПУ

Если выключение покадровой обработки (SBLOF) запрограммировано в первой строке (PROC ...) **главной программы**, то оно действует до конца или до отмены программы ЧПУ. В этом случае программа ЧПУ выполняется при отдельном кадре без остановки.

Если выключение покадровой обработки (SBLOF) запрограммировано в первой строке (PROC ...) **подпрограммы**, то оно действует до конца или до отмены программы ЧПУ. С

помощью запрограммированной команды возврата принимается решение, произойдет ли остановка на конце подпрограммы или нет:

- Возврат с M17: Стоп на конце подпрограммы
- Возврат с RET: **Нет** останова в конце подпрограммы

Подавление покадровой обработки внутри программы ЧПУ

Если выключение покадровой обработки (SBLOF) запрограммировано в кадре внутри программы ЧПУ, то покадровая обработка выключается с этого кадра и до следующего запрограммированного включения покадровой обработки (SBLON) или в конце активного уровня подпрограммы.

Синтаксис

Подавление покадровой обработки для всей программы ЧПУ:

```
PROC ... SBLOF
...
```

Подавление покадровой обработки внутри программы ЧПУ:

```
...
SBLOF
...
SBLON
...
```

Объяснение

PROC:	Первый оператор программы	
SBLOF:	Предопределенная процедура отключения покадровой обработки	
	Автономно в кадре:	да, возможно в кадре PROC
	Активность:	модально
SBLON:	Предопределенная процедура включения покадровой обработки	
	Автономно в кадре:	да
	Активность:	модально

Особенности

- **Индикация кадра при подавленной покадровой обработке**
  Актуальная индикация кадра может быть подавлена в подпрограммах с DISPLOF. Если DISPLOF программируется вместе с SBLOF, то при остановах покадровой обработки внутри подпрограммы отображается вызов подпрограммы.
- **Подавление покадровой обработки в случае асинхронных подпрограмм (ASUP)**
  Для выполнения ASUP при активной покадровой обработке за один шаг необходимо запрограммировать в ASUP оператор PROC с SBLOF. Это же относится и к функции "Редактируемая системная ASUP" (MD11610 $MN_ASUP_EDITABLE).
  Если останов покадровой обработки подавляется в системной ASUP или ASUP пользователя через программирование SBLOF в строке PROC или через установки в машинных данных MD10702 $MN_IGNORE_SINGLEBLOCK_MASK (Бит0 = 1 или Бит1 = 1), то останов покадровой обработки через программирование SBLON в ASUP снова может быть активирован.
  Если останов покадровой обработки подавляется в ASUP пользователя через установки в машинных данных MD20117 $MC_IGNORE_SINGLEBLOCK_ASUP, то останов покадровой обработки через программирование SBLON в ASUP **не** может быть снова активирован.
- **Особенности для различных типов покадровой обработки**
  - "SB2: Кадр вычисления" UND MD10702 $MN_IGNORE_SINGLEBLOCK_MASK, бит 12 = 1:
    → Останова в кадре SBLON нет.
  - "SB3: Отдельный кадр точный":
    → Команда SBLOF подавляется**.**
- **Подавление покадровой обработки во вложенных программах**
  Если в подпрограмме была запрограммирована SBLOF в операторе PROC, то происходит останов на возврате из подпрограммы с M17. Тем самым не допускается исполнение уже следующего кадра в вызывающей программе. Если в подпрограмме с SBLOF, без SBLOF в операторе PROC, активируется подавление покадровой обработки, то останов происходит только после следующего кадра функций станка вызывающей программы. Если это не желательно, то в подпрограмме еще до возврата (M17) снова необходимо запрограммировать SBLON. При возврате с RET в программу верхнего уровня останов не происходит.

Примеры

Пример 1: Подавление покадровой обработки внутри программы ЧПУ

Исходная ситуация: Покадровая обработка активна.

Программный код	Комментарий
N10 G1 X100 F1000	
N20 **SBLOF**	; Выключение покадровой обработки
N30 Y20	
N40 M100	
N50 R10=90	
N60 **SBLON**	; Снова включить покадровую обработку
N70 M110	

Программный код	Комментарий
N80 ...	

Область между N20 и N60 в режиме покадровой обработки обрабатывается как один шаг.

Пример 2:
Цикл должен действовать для пользователя как одна команда

Исходная ситуация: Покадровая обработка активна.

Главная программа:

Программный код
...
N100 G1 X10 G90 F200
N120 X-4 Y6
N130 **CYCLE1**
N140 G1 X0
N150 M30

Цикл CYCLE1:

Программный код	Комментарий
N100 **PROC** CYCLE1 DISPLOF **SBLOF**	; Подавление покадровой обработки для всей программы.
N110 R10=3*SIN(R20)+5	
N120 IF (R11 <= 0)	
N130 SETAL(61000)	
N140 ENDIF	
N150 G1 G91 Z=R10 F=R11	
N160 M17	

Цикл CYCLE1 выполняется при активной покадровой обработке. Это означает, что для обработки CYCLE1 необходимо один раз нажать клавишу Start.

Пример 3: Запущенная с PLC ASUP для активации измененного смещения нулевой точки и коррекций инструмента должна быть скрыта.

Программный код	Комментарий
N100 **PROC** NV **SBLOF DISPLOF**	; Подавление покадровой обработки и индикации кадра.
N110 CASE $P_UIFRNUM OF	
0 GOTOF _G500	
1 GOTOF _G54	
2 GOTOF _G55	
3 GOTOF _G56	
4 GOTOF _G57	
DEFAULT GOTOF END	
N120 _G54: G54 D=$P_TOOL T=$P_TOOLNO	
N130 RET	
N140 _G54: G55 D=$P_TOOL T=$P_TOOLNO	
N150 RET	
N160 _G56: G56 D=$P_TOOL T=$P_TOOLNO	

Программный код	Комментарий
N170 RET	
N180 _G57: G57 D=$P_TOOL T=$P_TOOLNO	
N190 RET	
N200 END: D=$P_TOOL T=$P_TOOLNO	
N210 RET	

Пример 4: Целенаправленный останов в подпрограмме

Исходная ситуация:

- Покадровая обработка активна.
- MD10702 $MN_IGNORE_SINGLEBLOCK_MASK, бит 12 = 1

Главная программа:

Программный код	Комментарий
N10 G0 X0	; Останов покадровой обработки
N20 X10	; Останов покадровой обработки
N30 **CYCLE**	; Созданный циклом кадр перемещения.
N50 G90 X20	; Останов покадровой обработки
M30	

Цикл CYCLE:

Программный код	Комментарий
PROC CYCLE **SBLOF**	; Подавить покадровую обработку
N100 R0=1	
N110 **SBLON**	; Нет останова покадровой обработки из-за MD10702 бит 12 = 1
N120 X1	; Останов покадровой обработки
N140 **SBLOF**	
N150 R0=2	
RET	

Пример 5: Подавление покадровой обработки при вложенности программы

Исходная ситуация:

- Покадровая обработка тип 2 активна.
- В кадре SBLON останов запрещен (MD10702 $MN_IGNORE_SINGLEBLOCK_MASK, бит 12 = 1)

Программный код	Комментарий
N10 X0 F1000	; Останов покадровой обработки
N20 UP1(0)	
PROC UP1(INT _NR) **SBLOF**	; Выключение покадровой обработки для UP1
N100 X10	
N110 UP2(0)	
PROC UP2(INT _NR)	
N200 X20	

Программный код	Комментарий
N210 **SBLON**	; Включение покадровой обработки
N220 X22	; Останов покадровой обработки
N230 UP3(0)	
PROC UP3(INT _NR)	
N300 **SBLOF**	; Выключение покадровой обработки
N305 X30	
N310 **SBLON**	; Включение покадровой обработки
N320 X32	; Останов покадровой обработки
N330 **SBLOF**	; Выключение покадровой обработки
N340 X34	
N350 M17	; Останов покадровой обработки (M17)
N240 X24	; Останов покадровой обработки (N210)
N250 M17	; Останов покадровой обработки (M17)
N120 X12	
N130 M17	; Останов покадровой обработки (M17)
N30 X0	; Останов покадровой обработки
N40 M30	; Останов покадровой обработки

4.2.2.6 Подавление актуальной индикации кадра (DISPLOF, DISPLON, ACTBLOCNO)

На индикации кадра стандартно индицируется актуальный программный кадр. В циклах или подпрограммах индикация актуального кадра может быть подавлена с помощью команды DISPLOF. Вместо актуального кадра тогда индицируется вызов цикла или подпрограммы. С помощью команды DISPLON подавление индикации кадров может быть снова отменено.

DISPLOF или DISPLON программируется в строке программы с оператором PROC и действует для всей подпрограммы и не явно для всех вызванных из этой подпрограммы подпрограмм, не содержащих команды DISPLON или DISPLOF. Это же поведение действует и для ASUP.

Синтаксис

```
PROC … DISPLOF
PROC … DISPLOF ACTBLOCNO
PROC … DISPLOF
```

Значение

DISPLOF:	Команда для подавления актуальной индикации кадра.	
	Размещение:	На конце строки программы с оператором PROC
	Активность:	До возврата из подпрограммы или завершения программы.
	Примечание: Если из подпрограммы с помощью команды DISPLOF вызываются другие подпрограммы, то и в этих подпрограммах актуальная индикация кадра подавляется, если в них явно не запрограммирована DISPLON.	

DISPLON:	Команда отмены подавления актуальной индикации кадров	
	Размещение:	На конце строки программы с оператором PROC
	Активность:	До возврата из подпрограммы или завершения программы.
	Примечание: Если из подпрограммы с помощью команды DISPLON вызываются другие подпрограммы, то и в этих подпрограммах отображается актуальный кадр программы, если в них явно не запрограммирована DISPLOF.	
ACTBLOCNO:	DISPLOF вместе с атрибутом ACTBLOCNO приводит к тому, что в случае ошибки выводится номер актуального кадра, в котором возникла ошибка. Это действует и в том случае, когда на более низком программном уровне запрограммирована только DISPLOF. При DISPLOF без ACTBLOCNO, напротив, индицируется номер кадра вызова цикла или подпрограммы из последнего не обозначенного с DISPLOF программного уровня.	

Примеры

Пример 1: Подавление актуальной индикации кадра в цикле

Программный код	Комментарий
PROC CYCLE(AXIS TOMOV, REAL POSITION) SAVE DISPLOF	; Подавление актуальной индикации кадра. Вместо этого должен быть показан вызов цикла, к примеру: CYCLE(X,100.0)
DEF REAL DIFF	; Содержание циклов
G01 ...	
...	
RET	; Возврат из подпрограммы. На индикации кадра появляется следующий за вызовом цикла кадр.

Пример 2: Индикация кадра при выводе ошибки

Подпрограмма SUBPROG1 (с ACTBLOCNO):

Программный код	Комментарий
PROC SUBPROG1 DISPLOF ACTBLOCNO	
N8000 R10 = R33 + R44	
...	
N9040 R10 = 66 X100	; Запустить ошибку 12080
...	
N10000 M17	

Подпрограмма SUBPROG2 (без ACTBLOCNO):

Программный код	Комментарий
PROC SUBPROG2 DISPLOF	

Программный код	Комментарий
N5000 R10 = R33 + R44	
...	
N6040 R10 = 66 X100	; Запустить ошибку 12080
...	
N7000 M17	

Главная программа:

Программный код	Комментарий
N1000 G0 X0 Y0 Z0	
N1010 ...	
...	
N2050 SUBPROG1	; Вывод ошибки = "12080 канал K1 кадр N9040 синтаксическая ошибка в тексте R10="
N2060 ...	
N2350 SUBPROG2	; Вывод ошибки = "12080 канал K1 кадр N2350 синтаксическая ошибка в тексте R10="
...	
N3000 M30	

Пример 3: Отменить подавление актуальной индикации кадра

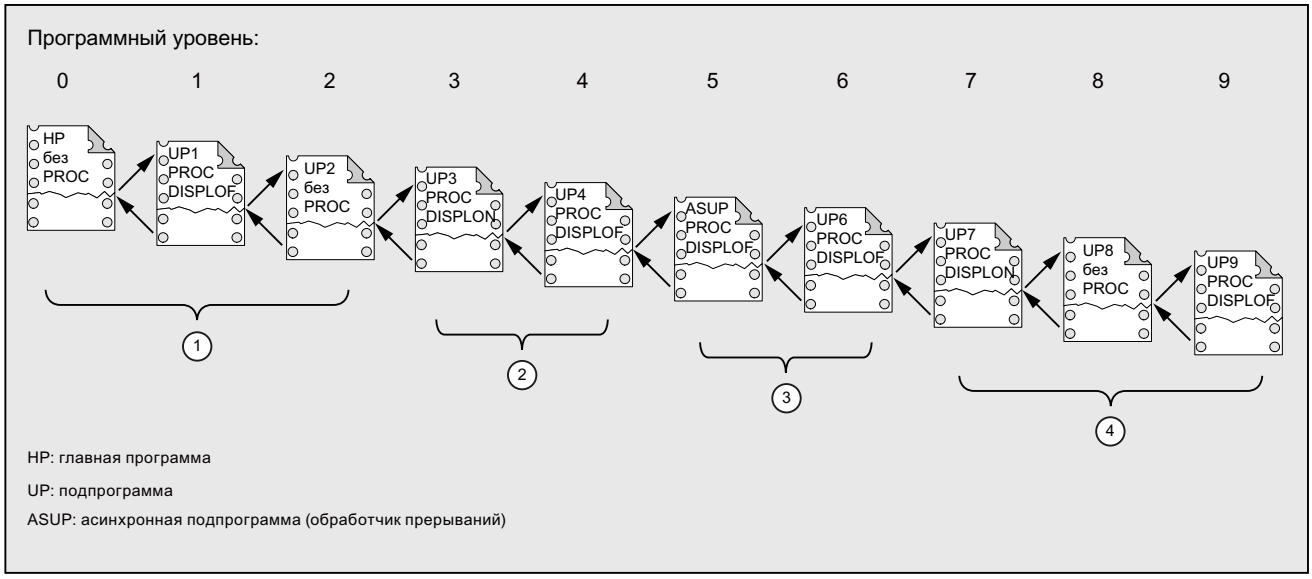
Подпрограмма SUB1 с подавлением:

Программный код	Комментарий
PROC SUB1 DISPLOF	; Подавление актуальной индикации кадров в подпрограмме SUB1. Вместо этого должен быть показан кадр с вызовом SUB1.
...	
N300 SUB2	; Вызвать подпрограмму SUB2.
...	
N500 M17	

Подпрограмма SUB2 без подавления:

Программный код	Комментарий
PROC SUB2 DISPLON	; Отменить подавление актуальной индикации кадров в подпрограмме SUB2.
...	
N200 M17	; Возврат в подпрограмму SUB1. В SUB1 актуальная индикация кадров снова подавляется.

Пример 4: Параметры индикации при различных комбинациях DISPLON/DISPLOF



① На актуальной индикации кадров отображаются строки программы обработки детали из программного уровня 0.

② На актуальной индикации кадров отображаются строки программы обработки детали из программного уровня 3.

③ На актуальной индикации кадров отображаются строки программы обработки детали из программного уровня 3.

④ На актуальной индикации кадров отображаются строки программы обработки детали из программного уровня 7/8.

4.2.2.7 Обозначить подпрограммы с подготовкой (PREPRO)

С помощью кодового слова `PREPRO` при запуске в конце строки оператора `PROC` можно обозначить все файлы.

Примечание

Такой тип подготовки программы зависит от настройки машинных данных MD10700 $MN_PREPROCESSING_LEVEL. Просьба следовать указаниям изготовителя станка.

Синтаксис

```
PROC … PREPRO
```

Объяснение

`PREPRO`:	Кодовое слово для обозначения всех подготовленных при запуске файлов, сохраненных в директориях циклов программ ЧПУ

Загрузка подпрограмм с подготовкой и вызов подпрограмм

Как подготовленные при запуске подпрограммы с параметрами, так и при вызове подпрограммы, обработка директорий циклов выполняется в той же последовательности:

1. _N_CUS_DIR циклы пользователя
2. _N_CMA_DIR циклы изготовителя
3. _N_CST_DIR стандартные циклы

В случае одноименных программ ЧПУ с различным оформлением активируется последний найденный оператор PROC, а другой оператор PROC игнорируется без сообщения об ошибке.

4.2.2.8 Возврат из подпрограммы M17

В конце подпрограммы стоит команда возврата M17 (или команда программы обработки детали M30). Она вызывает возврат в вызывающую программу на кадр программы обработки детали после вызова подпрограммы.

Примечание

M17 и M30 обрабатываются в языке ЧПУ равнозначно.

Синтаксис

```
PROC <имя программы>
...
M17/M30
```

Граничные условия

Влияние возврата из подпрограммы на режим управления траекторией

Если M17 (или M30) стоит одна в кадре программы обработки детали, то из-за этого активный в канале режим управления траекторией прерывается.

Во избежание прерывания режима управления траекторией, записать M17 (или M30) и в последний кадр перемещения. Дополнительно следующие машинные данные должны быть установлены на "0":

MD20800 $MC_SPF_END_TO_VDI = 0 (нет вывода M30/M17 на интерфейс ЧПУ/PLC)

Пример

1. подпрограмма с M17 в отдельном кадре

Программный код	Комментарий
N10 G64 F2000 G91 X10 Y10	
N20 X10 Z10	

Программный код	Комментарий
N30 M17	; Возврат с прерыванием режима управления траекторией.

2. подпрограмма с M17 в последнем кадре перемещения

Программный код	Комментарий
N10 G64 F2000 G91 X10 Y10	
N20 X10 Z10 M17	; Возврат без прерывания режима управления траекторией.

4.2.2.9 Возврат из подпрограммы RET

В качестве эквивалента для команды возврата с M17, в подпрограмме можно использовать и команду RET . RET должна быть запрограммирована в отдельном кадре программы обработки детали. Как и M17 RET вызывает возврат в вызывающую программу на кадр программы обработки детали после вызова подпрограммы.

Примечание

Через программирование параметров, параметры возврата RET могут быть изменены (см. "Параметрируемый возврат из подпрограммы (RET ...) (Страница 531)").

Использование

Использовать оператор RET тогда, когда режим управления траекторией G64 (G641 ... G645) не должен прерываться возвратом.

Условие

Команда RET может использоваться только в подпрограммах, которые не были определены с атрибутом SAVE.

Синтаксис

```
PROC <имя программы>
...
RET
```

Пример

Главная программа:

Программный код	Комментарий
PROC MAIN_PROGRAM	; Начало программы
...	
N50 SUB_PROG	; Вызов подпрограммы: SUB_PROG

Программный код	Комментарий
N60 ...	
...	
N100 M30	; конец программы

Подпрограмма:

Программный код	Комментарий
PROC SUB_PROG	
...	
N100 RET	; Возврат выполняется на кадр N60 в главной программе.

4.2.2.10 Параметрируемый возврат из подпрограммы (RET ...)

Вообще с помощью команды RET выполняется возврат из подпрограммы в вызывающую программу. Затем обработка продолжается с программной строки, следующей за вызовом подпрограммы. Если обработка программы должна продолжаться с другого места, имеются следующие возможности:

- Продолжение обработки программы после вызова циклов обработки резаньем в режиме диалекта ISO (после описания контура).
- Возврат в главную программу из любого уровня подпрограммы (и после ASUP) при обработке ошибок.
- Возврат через несколько программных уровней для специальных приложений в компилируемых циклах и в режиме диалекта ISO.

Для этого команду RET нужно запрограммировать с дополнительными параметрами.

Направление поиска

При указании параметра <конечный кадр> возврат осуществляется сначала на кадр после кадра вызова. Затем выполняется поиск цели в направлении **конца** программы, в которую осуществляется возврат. Если этот поиск не дал результата, далее выполняется поиск в направлении начала программы.

Синтаксис

RET("<конечный кадр>")
RET("<конечный кадр>",<кадр после конечного кадра>)
RET("<конечный кадр>",<кадр после конечного кадра>,<число уровней возврата>)
RET("<конечный кадр>", ,<число уровней возврата>)
RET("<конечный кадр>",<кадр после конечного кадра>,<число уровней возврата>,
<возврат на начало программы>)
RET(, ,<число уровней возврата>,<возврат на начало программы>)

Значение

RET:	Конец подпрограммы		
<конечный кадр>:	В этом параметре в качестве цели перехода указывается кадр, на котором обработка программы должна быть продолжена. Если параметр <число уровней возврата> не запрограммирован, то цель перехода находится в программе, из которой была вызвана актуальная подпрограмма. Возможными данными являются:		
	<номер кадра>	Номер конечного кадра. Поиск номера кадра в программе, в которую выполняется возврат, сначала выполняется в направлении конца программы.	
	<метка перехода>	Метка перехода, которая должна быть в программе, куда выполняется возврат. Поиск метки перехода в программе, в которую выполняется возврат, сначала выполняется в направлении конца программы.	
	<строка символов>	Строка символов, которая должна быть в программе, куда выполняется возврат (например, имя программы или переменной). Поиск строки символов в программе, в которую выполняется возврат, сначала выполняется в направлении конца программы. Для программирования строки символов в конечном кадре действуют следующие правила: • **Пробел на конце** (в отличие от метки перехода, которая обозначается через ":" на конце). • **Перед строкой символов** может быть установлен только номер кадра и/или метка перехода, **не программные команды**.	
<кадр после конечного кадра>:	В этом параметре указывается, должна ли обработка программы продолжаться в кадре, указанном в параметре <конечный кадр>, или в следующем за ним кадре.		
	Тип:	INT	
	Значение:	0	Возврат осуществляется на кадр, указанный в параметре <конечный кадр>.
		> 0	Возврат осуществляется на кадр, следующий за указанным в параметре <конечный кадр>.

<число уровней возврата>:	В этом параметре указывается число программных уровней, которые нужно пройти назад, чтобы найти там конечный кадр и продолжить обработку программы.		
	Тип:	INT	
	Значение:	1	Программа продолжается на "актуальном программном уровне -1" (как `RET` без параметров).
		2	Программа продолжается на "актуальном программном уровне -2", т. е. пропускается один уровень.
		3	Программа продолжается на "актуальном программном уровне -3", т. е. пропускается два уровня.
		...	
	Диапазон значений:	1 ... 15	
<возврат на начало программы>:	В этом параметре указывается, нужно ли при возврате в главную программу и активном там **режиме диалекта ISO** продолжать программу с самого начала.		
	Тип:	BOOL	
	Значение:	1	Если возврат осуществляется в главную программу и там активен **режим диалекта ISO** , то выполняется ветвление на начало программы.

Примечание

При возврате из подпрограммы со строкой символов в качестве данных для поиска конечного кадра, в вызывающей программе сначала всегда выполняется поиск метки перехода.

Поэтому, если цель перехода должна быть однозначно определена строкой символов, то строка символов не должна совпадать с именем метки перехода, т.к. в ином случае возврат из подпрограммы всегда будет выполняться на метку перехода, а не на строку символов (см. пример 2).

Граничные условия

Возврат через несколько программных уровней

При возврате через несколько программных уровней обрабатываются операторы SAVE отдельных программных уровней.

Если при возврате через несколько программных уровней активна модальная подпрограмма и если в одной из пропущенных подпрограмм запрограммирована

команда отмены MCALL для модальной подпрограммы, то модальная подпрограмма остается активной.

ВНИМАНИЕ
Ошибки программирования В случае возврата через несколько программных уровней пользователь сам должен позаботиться о том, чтобы обработка продолжалась с необходимыми модальными настройками. Это обеспечивается, например, программированием соответствующего главного кадра.

Выполнение с внешнего устройства

Если главная и подпрограммы выполняются через функцию "Выполнение с внешнего устройства" из внешней программной памяти, то запрограммированная с RET(...) цель перехода должны находиться в памяти догрузки. В ином случае цель перехода не будет найдена и происходит отмена программы с выводом аварийного сообщения 14000.

Примечание

Для выполнения внешних программ без ограничений в части запрограммированных операторов перехода, рекомендуется вместо функции "Выполнение с внешнего устройства" использовать опцию "Выполнение из внешней памяти (EES)".

Примеры

Пример 1: Повторная установка в главной программе после обработки ASUP

Программирование	Комментарий
`N10010 CALL "UP1"`	; программный уровень 0 (главная программа)
`  N11000 PROC UP1`	; программный уровень 1
`  N11010 CALL "UP2"`	
`  N12000 PROC UP2`	; программный уровень 2
...	
`  N19000 PROC ASUP`	; программный уровень 3 (обработка ASUP)
...	
`  N19100 RET("N10900", ,$P_STACK)`	; Возврат из подпрограммы в главную программу
	; $P_STACK: актуальный программный уровень
`N10900`	; Конечный кадр в главной программе
`N10910 MCALL`	; Выключить модальный вызов подпрограммы
`N10920 G0 G60 G40 M5`	; Инициализировать другие модальные настройки

Пример 2: Строка символов (<строка>) как данные для поиска конечного кадра

Главная программа:

Программный код	Комментарий
PROC MAIN_PROGRAM	
N1000 DEF INT iVar1=1, iVar2=4	
N1010 ...	
N1200 subProg1	; Вызов подпрограммы "subProg1"
N1210 M2 S1000 X10 F1000	
N1220	
N1400 subProg2	; Вызов подпрограммы "subProg2"
N1410 M3 S500 Y20	
N1420 ..	
N1500 lab1: iVar1=R10*44	
N1510 F500 X5	
N1520 ...	
N1550 subprog1: G1 X30	; "subProg1" определена здесь как метка перехода.
N1560 ...	
N1600 subProg3	; Вызов подпрограммы "subProg3"
N1610 ...	
N1900 M30	

Подпрограмма subProg1:

Программный код	Комментарий
PROC subProg1	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg2")	; Возврат в главную программу на кадр N1400

Подпрограмма subProg2:

Программный код	Комментарий
PROC subProg2	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("iVar1")	; Возврат в главную программу на кадр N1500

Подпрограмма subProg3:

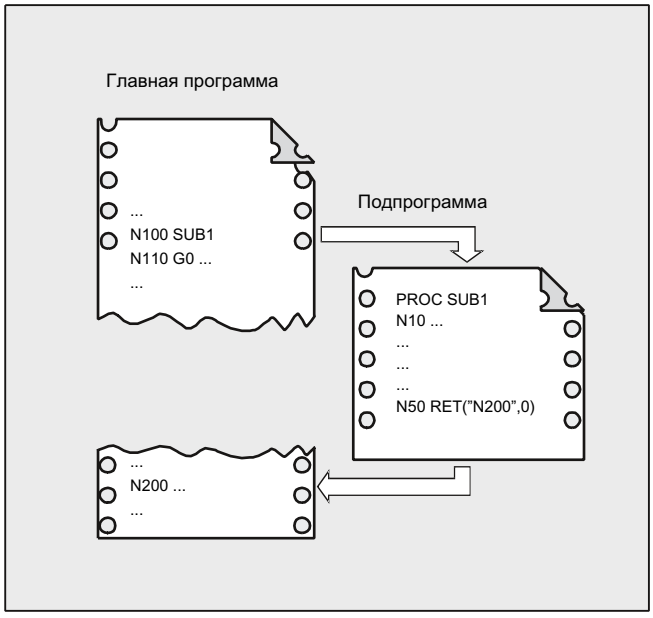
Программный код	Комментарий
PROC subProg3	
N2000 R10=R20+100	
N2010 ...	

Программный код	Комментарий
`N2200 RET("subProg1")`	; Возврат в главную программу на кадр `N1550`

Дополнительная информация

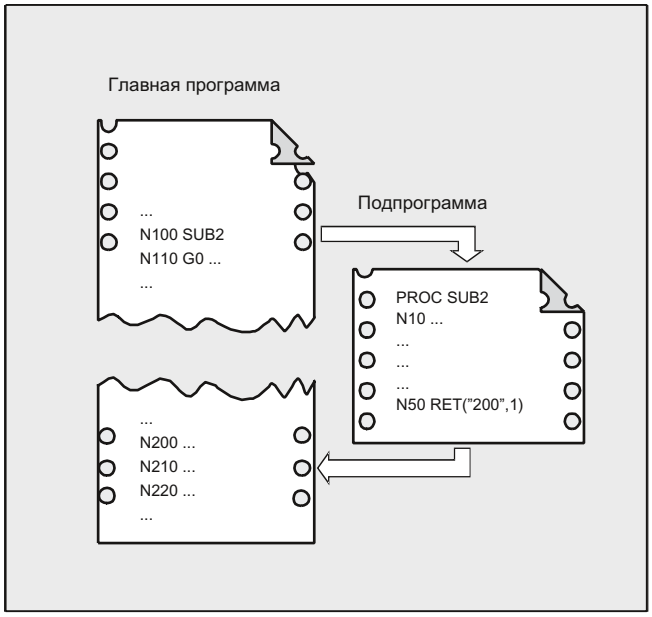
На следующих рисунках показаны различные действия параметров возврата

1. <конечный кадр> = "N200", <кадр после конечного кадра> = 0



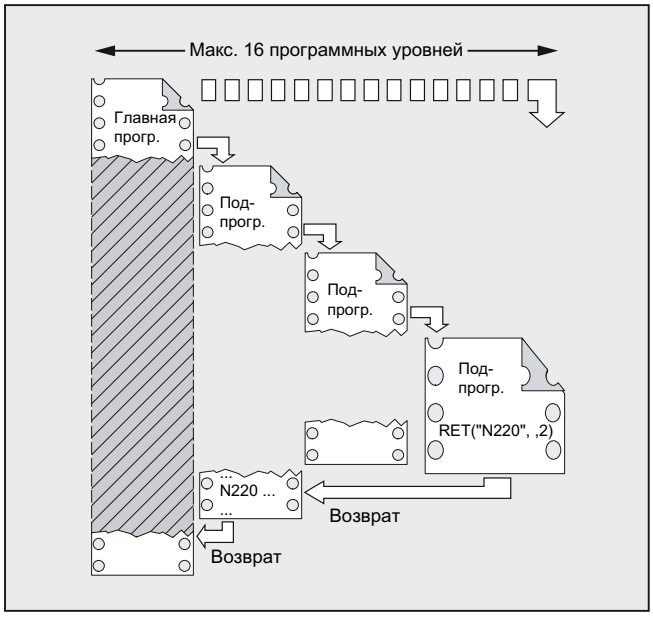
После команды `RET` обработка программы продолжается на кадре `N200` в главной программе.

2. <конечный кадр> = "N200", <кадр после конечного кадра> = 1



После команды `RET` обработка программы продолжается на кадре (`N210`), следующим за кадром `N200` в главной программе.

3. <конечный кадр> = "N220", <число уровней возврата> = 2



После команды `RET` два программных уровня пропускается и обработка программы продолжается на кадре `N220` .

4.2.2.11 Параметрируемый возврат из подпрограммы (RETB ...)

Вообще с помощью команды RETB выполняется возврат из подпрограммы в вызывающую программу. Затем обработка продолжается с программной строки, следующей за вызовом подпрограммы. Если обработка программы должна продолжаться с другого места, имеются следующие возможности:

- Продолжение обработки программы после вызова циклов обработки резаньем в режиме диалекта ISO (после описания контура).
- Возврат в главную программу из любого уровня подпрограммы (и после ASUP) при обработке ошибок.
- Возврат через несколько программных уровней для специальных приложений в компилируемых циклах и в режиме диалекта ISO.

Для этого команду RETB нужно запрограммировать с дополнительными параметрами.

Направление поиска

При указании параметра <конечный кадр> возврат осуществляется сначала на кадр после кадра вызова. Затем выполняется поиск цели в направлении **начала** программы, в которую осуществлен возврат. Если этот поиск не дал результата, далее выполняется поиск в направлении конца программы.

Синтаксис

```
RETB("<конечный кадр>")
RETB("<конечный кадр>",<кадр после конечного кадра>)
RETB("<конечный кадр>",<кадр после конечного кадра>,<число уровней возврата>)
RETB("<конечный кадр>", ,<число уровней возврата>)
RETB("<конечный кадр>",<кадр после конечного кадра>,<число уровней возврата>,
<возврат на начало программы>)
RETB( , ,<число уровней возврата>,<возврат на начало программы>)
```

Значение

RETB:	Конец подпрограммы		
<конечный кадр>:	В этом параметре в качестве цели перехода указывается кадр, на котором обработка программы должна быть продолжена. Если параметр <число уровней возврата> не запрограммирован, то цель перехода находится в программе, из которой была вызвана актуальная подпрограмма. Возможными данными являются:		
	<номер кадра>	Номер конечного кадра. Поиск номера кадра в программе, куда выполняется возврат, сначала осуществляется в направлении начала программы.	
	<метка перехода>	Метка перехода, которая должна быть в программе, куда выполняется возврат. Поиск метки перехода в программе, куда выполняется возврат, сначала осуществляется в направлении начала программы.	
	<строка символов>	Строка символов, которая должна быть в программе, куда выполняется возврат (например, имя программы или переменной). Поиск строки символов в программе, куда выполняется возврат, сначала осуществляется в направлении начала программы. Для программирования строки символов в конечном кадре действуют следующие правила: • **Пробел на конце** (в отличие от метки перехода, которая обозначается через ":" на конце). • **Перед строкой символов** может быть установлен только номер кадра и/или метка перехода, **не программные команды**.	
<кадр после конечного кадра>:	В этом параметре указывается, должна ли обработка программы продолжаться в кадре, указанном в параметре <конечный кадр>, или в следующем за ним кадре.		
	Тип:	INT	
	Значение:	0	Возврат осуществляется на кадр, указанный в параметре <конечный кадр>.
		> 0	Возврат осуществляется на кадр, следующий за указанным в параметре <конечный кадр>.

<число уровней возврата>:	В этом параметре указывается число программных уровней, которые нужно пройти назад, чтобы найти там конечный кадр и продолжить обработку программы.		
	Тип:	INT	
	Значение:	1	Программа продолжается на "актуальном программном уровне -1" (как `RET` без параметров).
		2	Программа продолжается на "актуальном программном уровне -2", т. е. пропускается один уровень.
		3	Программа продолжается на "актуальном программном уровне -3", т. е. пропускается два уровня.
		...	
	Диапазон значений:	1 ... 15	
<возврат на начало программы>:	В этом параметре указывается, нужно ли при возврате в главную программу и активном там **режиме диалекта ISO** продолжать программу с самого начала.		
	Тип:	BOOL	
	Значение:	1	Если возврат осуществляется в главную программу и там активен **режим диалекта ISO** , то выполняется ветвление на начало программы.

Примечание

При возврате из подпрограммы со строкой символов в качестве данных для поиска конечного кадра, в вызывающей программе сначала всегда выполняется поиск метки перехода.

Поэтому, если цель перехода должна быть однозначно определена строкой символов, то строка символов не должна совпадать с именем метки перехода, т.к. в ином случае возврат из подпрограммы всегда будет выполняться на метку перехода, а не на строку символов (см. пример 2).

Граничные условия

Возврат через несколько программных уровней

При возврате через несколько программных уровней обрабатываются операторы SAVE отдельных программных уровней.

Если при возврате через несколько программных уровней активна модальная подпрограмма и если в одной из пропущенных подпрограмм запрограммирована

команда отмены MCALL для модальной подпрограммы, то модальная подпрограмма остается активной.

ВНИМАНИЕ
Ошибки программирования В случае возврата через несколько программных уровней пользователь сам должен позаботиться о том, чтобы обработка продолжалась с необходимыми модальными настройками. Это обеспечивается, например, программированием соответствующего главного кадра.

Выполнение с внешнего устройства

Если главная и подпрограммы выполняются через функцию "Выполнение с внешнего устройства" из внешней программной памяти, то запрограммированная с RETB(...) цель перехода должна находиться в памяти догрузки. В ином случае цель перехода не будет найдена и происходит отмена программы с выводом аварийного сообщения 14000.

Примечание

Для выполнения внешних программ без ограничений в части запрограммированных операторов перехода, рекомендуется вместо функции "Выполнение с внешнего устройства" использовать опцию "Выполнение из внешней памяти (EES)".

Пример

Программный код	Комментарий
BEISPIEL.MPF	
…	
N3000 START_CYC(param1, param2, …)	
N3010 TECH_CYC1(param1, param2, …)	
N3020 TECH_CYC2(param1, param2, …)	
N3030 TECH_CYC3(param1, param2, …)	
N3040 END_CYC(param1, param2, …)	
N3040 **END_CYC**(param1, param2, …)	
N3050 …	
N4500 START_CYC(param11, param12, …)	
N4510 …	
N4590 END_CYC(param11, param12, ..)	
N5000 …	
…	
N6000 M30	

Программный код	Комментарий
PROC END_CYC(…)	; Вызов в главной программе строки N3040
N10000 …	

Программный код	Комментарий
N15000 if status == 1	
N15010 RETB("START_CYC")	; Возврат в вызывающую программу BEISPIEL.MPF
	; Поиск строки символов "START_CYC"
	; Направление поиска: назад, в направлении начала программы
	; Продолжение обработки программы со строки N3000
N15020 ENDIF	
N15030 if status == 0	
N15040 RET	; Возврат в вызывающую программу BEISPIEL.MPF
	; Продолжение обработки программы со строки N3050
N15050 ENDIF	
N16000 RET("START_CYC")	; Возврат в вызывающую программу BEISPIEL.MPF
	; Поиск строки символов "START_CYC"
	; Направление поиска: вперед, в направлении конца программы
	; Продолжение обработки программы со строки N4500
N17060 RETB	; Возврат в вызывающую программу BEISPIEL.MPF
	; Продолжение обработки программы со строки N3050
	; RETB без параметров идентична RET

4.2.3 Вызов подпрограммы

4.2.3.1 Вызовы подпрограмм без передачи параметров

Вызов подпрограммы выполняется либо с адресом L и номером подпрограммы, либо через указание имени программы.

Главная программа также может быть вызвана как подпрограмма. В этом случае установленный в главной программе конец программы M2 или M30 интерпретируется как M17 (конец программы с возвратом в вызывающую программу).

Примечание

Подпрограмма также может быть запущена как главная программа.

Стратегия поиска СЧПУ:

Существует ли *_MPF?

Существует ли *_SPF?

Из этого следует: Если имя вызываемой подпрограммы идентично имени главной программы, то снова вызывается вызывающая главная программа. Этого, как правило, нежелательного эффекта можно избежать через выбор однозначных имен для подпрограмм и главных программ.

Примечание

Подпрограммы, не требующие передачи параметров, могут вызываться и из файла инициализации.

Синтаксис

L<номер>/<имя программы>

Примечание

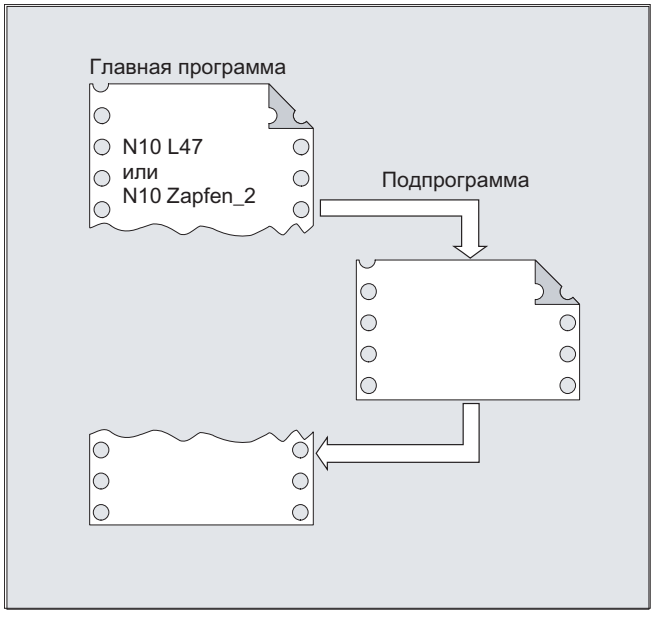
Вызов подпрограммы всегда должен быть запрограммирован в отдельном кадре ЧПУ.

Значение

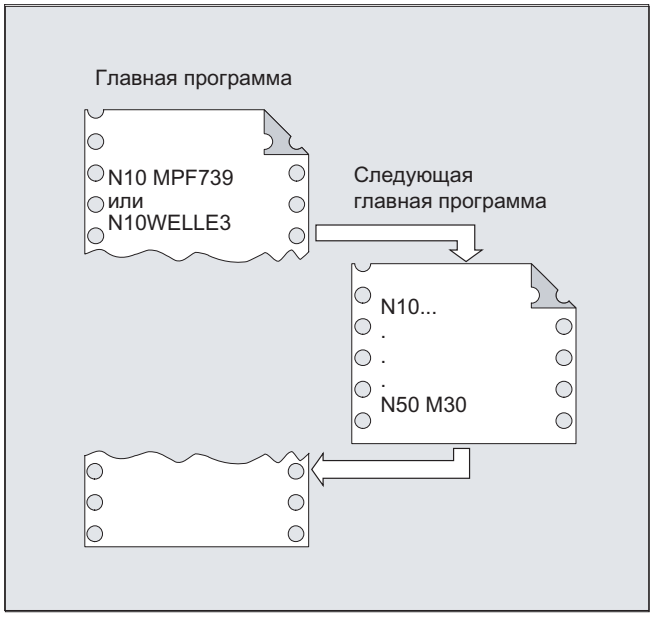
L:	Адрес для вызова подпрограммы	
<номер>:	Номер подпрограммы	
	Тип:	INT
	Значение:	Макс. 7 десятичных позиций **Внимание:** Вводные нули имеют значение при присваивании имени (⇒ L123, L0123 и L00123 это три различные подпрограммы).
<имя программы>:	Имя подпрограммы (или главной программы)	

Примеры

Пример 1: Вызов подпрограммы без передачи параметров



Пример 2: Вызов главной программы как подпрограммы



См. также

Подпрограмма без передачи параметров (Страница 514)

4.2.3.2 Вызов подпрограммы с передачей параметров (EXTERN)

При вызове подпрограммы с передачей параметров переменные или значения могут передаваться напрямую (не для параметров VAR).

Подпрограммы с передачей параметров перед вызовом в главной программе должны быть указаны с EXTERN (к примеру, в начале программы). При этом указывается имя подпрограммы и типы переменных в порядке передачи.

ВНИМАНИЕ
Опасность спутывания Как типы переменных, так и последовательность передачи, должны совпадать с определениями, согласованными в подпрограмме в PROC. Имена параметров в главной и подпрограмме могут быть различными.

Синтаксис

```
EXTERN <имя программы>(<тип_пар1>,<тип_пар2>,<тип_пар3>)
...
<имя программы>(<значение_пар1>,<значение_пар2>,<значение_пар3>)
```

Примечание

Вызов подпрограммы всегда должен быть запрограммирован в отдельном кадре ЧПУ.

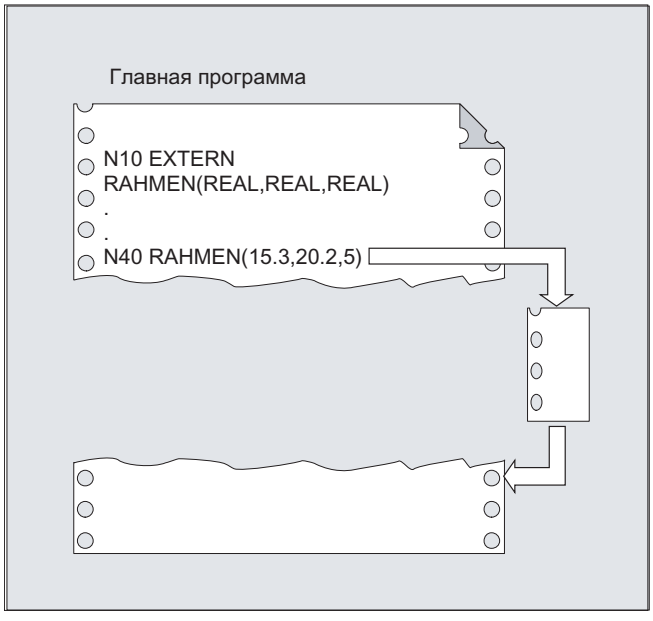
Объяснение

<имя программы>:	Имя подпрограммы
EXTERN:	Кодовое слово для указания подпрограммы с передачей параметров **Примечание:** EXTERN должен указываться только тогда, когда подпрограмма стоит в директории детали или в глобальной директории подпрограмм. Циклы не должны указываться как EXTERN .
<тип_пар1>,<тип_пар2>,<тип_пар3>:	Типы переменных передаваемых параметров в последовательности передачи
<значение_пар1>,<значение_пар2>,<значение_пар3>:	Значения переменных для передаваемых параметров

Примеры

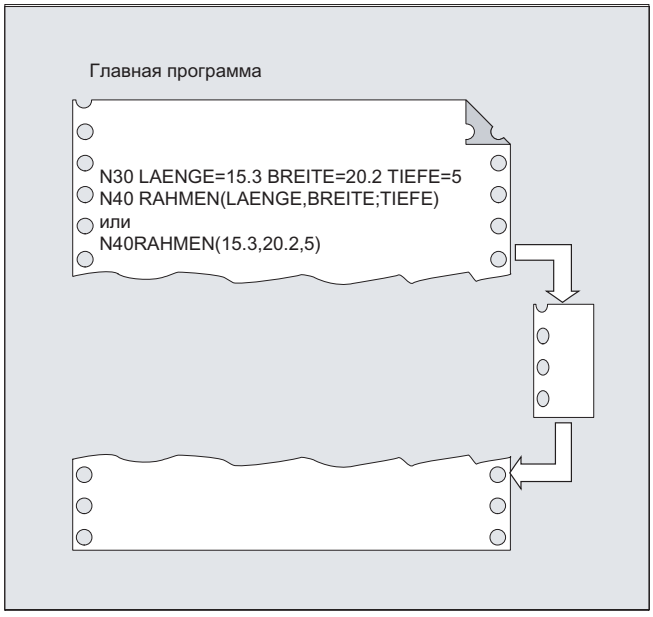
Пример 1: Вызов подпрограммы с предшествующим извещением

Программный код	Комментарий
N10 EXTERN RAHMEN(REAL,REAL,REAL)	; Указание подпрограммы.
...	
N40 RAHMEN(15.3,20.2,5)	; Вызов подпрограммы с передачей параметров.



Пример 2: Вызов подпрограммы без извещения

Программный код	Комментарий
`N10 DEF REAL LAENGE, BREITE, TIEFE`	
`N20 …`	
`N30 LAENGE=15.3 BREITE=20.2 TIEFE=5`	
`N40 RAHMEN(LAENGE,BREITE,TIEFE)`	`;или: N40 RAHMEN(15.3,20.2,5)`



См. также

Подпрограмма с передачей параметров по значению (PROC) (Страница 515)

Подпрограмма с передачей параметров по ссылке (PROC, VAR) (Страница 517)

4.2.3.3 Число повторений программы (P)

Если необходимо последовательное выполнение подпрограммы несколько раз, то в кадре с вызовом подпрограммы по адресу P можно запрограммировать желаемое число повторений программы.

⚠ ОСТОРОЖНО

Вызов подпрограммы с повторением программы и передачей параметров

Параметры передаются только при вызове или первом проходе программы. Для следующих повторений параметры остаются неизменными. Если необходимо при повторениях программы изменить параметры, то следует установить в подпрограмме соответствующие согласования.

Синтаксис

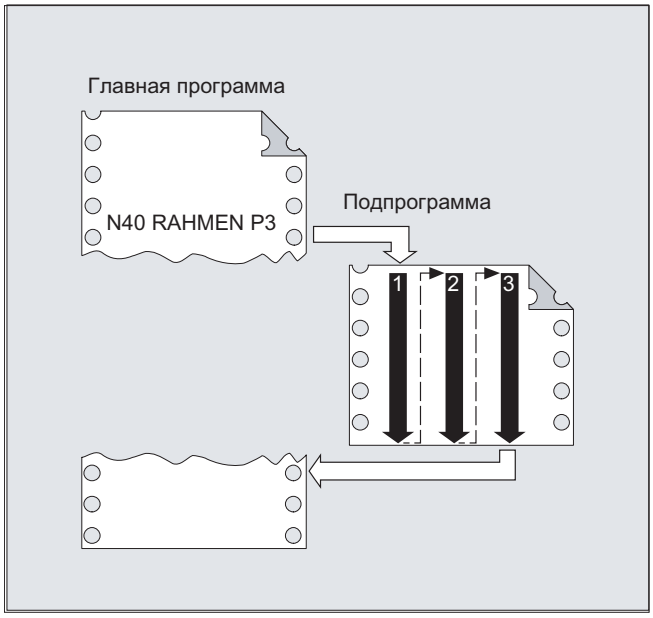
<имя программы> P<значение>

Значение

<имя программы>:	Вызов подпрограммы	
P:	Адрес для программирования повторений программы	
<значение>:	Число повторений программы	
	Тип:	INT
	Диапазон значений:	1 ... 9999 (без знака)

Пример

Программный код	Комментарий
...	
N40 RAHMEN P3	; Подпрограмма RAHMEN должна быть выполнена последовательно три раза.
...	



4.2.3.4 Модальный вызов подпрограммы (MCALL)

При модальном вызове подпрограммы `MCALL(<ProgName>)` указанная подпрограмма вызывается не сразу. Вместо этого вызов с этого момента в программе обработки детали осуществляется автоматически после каждого кадра перемещения с движениями по траектории. В том числе и с переходом через программные уровни.

Примечание

При выполнении программы действует всегда только последний модальный вызов подпрограммы `MCALL(<ProgName>)`. Текущий модальный вызов подпрограммы заменяет активный прежде.

Если в подпрограмму передаются параметры, то передача параметров осуществляется только при вызове:

```
MCALL(<ProgName>(Par1, Par2, ...))
```

Примечание

Область определения данных в начале подпрограммы проходится только один раз при выполнении кадра с запрограммированным вызовом MCALL. При следующих вызовах подпрограммы после кадров перемещения область определения данных больше не проходится. Вследствие этого значение, присвоенное при определении локальной переменной пользователя (LUD), после первого вызова более недоступно: доступно значение, записанное в цикле последним. После первого вызова переменная не создается заново, а сохраняет последнее значение предыдущего вызова. Во избежание такого поведения рекомендуется разделить определение LUD и присвоение значений и программировать присвоение значений в собственном кадре после области определения данных.

ВНИМАНИЕ

Модальные вызовы подпрограммы без движения по траектории

Модальная подпрограмма в следующих ситуациях вызывается и без программирования движения по траектории:

- Программирование адресов S или F, если активна G0 или G1
- Если G0 или G1 стоит в кадре одна или была запрограммирована вместе с другими G-командами.

Синтаксис

```
MCALL <ProgName>
...
MCALL
```

Объяснение

MCALL <ProgName>:	Включение функции "Модальный вызов подпрограммы"
<ProgName>:	Имя подпрограммы
MCALL:	С помощью MCALL без указания имени программы функция "Модальный вызов подпрограммы" выключается.

Граничные условия

ASUP

Если обработка программы обработки детали прерывается программой ASUP (см. главу "Обработчик прерываний (ASUP) (Страница 559)"), то в этой ASUP модальные вызовы подпрограммы не выполняются.

Если ASUP запускается в состоянии канала "Reset", она ведет себя относительно модальных вызовов подпрограммы, как обычная программа обработки детали.

Цикл смены инструмента

Если в цикле смены инструмента происходит отмена функции "Модальный вызов подпрограммы", следует учитывать, что при необходимости цикл смены инструмента не явно вызывается и после поиска кадра через поисковую ASUP или вручную через пересохранение. В этой ситуации отменять функцию "Модальный вызов подпрограммы" не следует, потому что иначе результат поиска будет ошибочным. Поэтому отмену функции "Модальный вызов подпрограммы" в цикле смены инструмента рекомендуется программировать следующим образом:

```
...
  IF $AC_ASUP == 0        ; Вызов осуществляется не через поисковую ASUP или пересохранение.
    MCALL                 ; Выключение функции "Модальный вызов подпрограммы".
  ENDIF
...
```

Примеры

Пример 1

```
N10 G0 X0 Y0
N20 MCALL L70             ; Включение модального вызова подпрограммы для L70.
N30 X10 Y10               ; Выполняется подвод в X10 Y10, затем вызывается L70.
N40 X20 Y20               ; Выполняется подвод в X20 Y20, затем вызывается L70.
...
N100 MCALL                ; Выключение функции "Модальный вызов подпрограммы".
N110 X0 Y0                ; Выполняется подвод в X0 Y0, а L70 не вызывается.
```

Пример 2

Программный код
N10 G0 X0 Y0
N20 MCALL L70
N30 L80

В этом примере последующие кадры ЧПУ с запрограммированными траекторными осями стоят в подпрограмме L80. L70 вызывается через L80.

4.2.3.5 Косвенный вызов подпрограммы (CALL)

В зависимости от имеющихся условий в одном месте могут вызываться различные подпрограммы. Для этого имя подпрограммы сохраняется в переменной типа STRING. Вызов подпрограммы осуществляется с CALL и именем переменной.

Примечание

Косвенный вызов подпрограммы возможен только для подпрограмм без передачи параметров. Для прямого вызова подпрограммы сохранить имя в постоянной STRING.

Синтаксис

CALL <имя программы>

Объяснение

CALL:	Команда для косвенного вызова подпрограммы	
<имя программы>:	Имя подпрограммы (переменная или постоянная)	
	Тип:	STRING

Пример

Прямой вызов с постоянной STRING:

Программный код	Комментарий
…	
CALL "/_N_WKS_DIR/_N_SUBPROG_WPD/_N_TEIL1_SPF"	; Вызвать подпрограмму TEIL1 с CALL напрямую.
…	

Косвенный вызов через переменную:

Программный код	Комментарий
…	
DEF STRING[100] PROGNAME	; Определение переменных.

Программный код	Комментарий
`PROGNAME="/_N_WKS_DIR/_N_SUBPROG_WPD/_N_TEIL1_SPF"`	; Согласовать подпрограмму `TEIL1` с переменной `PROGNAME`.
`CALL PROGNAME`	; Вызвать подпрограмму `TEIL1` через `CALL` и переменную `PROGNAME` косвенно.
…	

4.2.3.6 Косвенный вызов подпрограммы с указанием выполняемого программного блока (CALL BLOCK ... TO ...)

С помощью `CALL` и комбинации кодовых слов `BLOCK ... TO` подпрограмма вызывается косвенно и выполняется обозначенный стартовой меткой и меткой конца программный блок.

Синтаксис

`CALL` <имя программы> `BLOCK` <стартовая метка> `TO` <метка конца>
`CALL BLOCK` <стартовая метка> `TO` <метка конца>

Значение

`CALL`:	Команда для косвенного вызова подпрограммы	
<имя программы>:	Имя подпрограммы (переменная или постоянная), содержащей обрабатываемый программный блок (**опциональные данные**).	
	Тип:	STRING
	Примечание: Если <имя программы> не запрограммировано, то поиск и выполнение обозначенного <стартовой меткой> и <меткой конца> программного блока осуществляется в актуальной программе.	
`BLOCK ... TO ... :`	Комбинация кодовых слов для косвенного выполнения программного блока	
<стартовая метка>:	Переменная, указывающая на начало обрабатываемого программного блока.	
	Тип:	STRING
<метка конца>:	Переменная, указывающая на конец обрабатываемого программного блока.	
	Тип:	STRING

Пример

Главная программа:

Программный код	Комментарий
…	
`DEF STRING[20] STARTLABEL, ENDLABEL`	; Определение переменных для стартовой метки и метки конца.
`STARTLABEL="LABEL_1"`	

Программный код	Комментарий
`ENDLABEL="LABEL_2"`	
...	
`CALL "CONTUR_1" BLOCK STARTLABEL TO ENDLABEL`	; Косвенный вызов подпрограммы и обозначение выполняемого программного блока.
...	

Подпрограмма:

Программный код	Комментарий
`PROC CONTUR_1 ...`	
`LABEL_1`	; Стартовая метка: Начало выполнения программного блока
`N1000 G1 ...`	
...	
`LABEL_2`	; Метка конца: Конец выполнения программного блока
...	

4.2.3.7 Косвенный вызов запрограммированной на языке ISO программы (ISOCALL)

С помощью косвенного вызова программы ISOCALL можно вызвать запрограммированную на языке ISO программу. При этом активируется установленный в машинных данных режим ISO. В конце программы снова активируется первоначальный режим обработки. Если в машинных данных режим ISO не установлен, то вызов подпрограммы выполняется в режиме Siemens.

Дополнительная информация по режиму ISO Описание функций "Диалекты ISO"

Синтаксис

`ISOCALL` <имя программы>

Объяснение

`ISOCALL`:	Кодовое слово для косвенного вызова подпрограммы, при котором активируется установленный в машинных данных режим ISO.
<имя программы>:	Имя запрограммированной на языке ISO программы (переменная или постоянная типа STRING)

Пример: Пример вызова контура с программированием циклов из режима ISO

Программный код	Комментарий
`0122_SPF`	; Описание контура в режиме `ISO`
`N1010 G1 X10 Z20`	
`N1020 X30 R5`	
`N1030 Z50 C10`	
`N1040 X50`	
`N1050 M99`	

Программный код	Комментарий
N0010 DEF STRING[5] PROGNAME = "0122"	; Программа (цикл) обработки детали Siemens
...	
N2000 R11 = $AA_IW[X]	
N2010 ISOCALL PROGNAME	
N2020 R10 = R10+1	; Выполнить программу 0122.spf в режиме ISO
...	
N2400 M30	

4.2.3.8 Вызов подпрограммы с указанием пути и параметрами (PCALL)

С помощью PCALL возможен вызов подпрограмм с абсолютным указанием пути и передачей параметров.

Синтаксис

PCALL <путь/имя программы>(<параметр 1>,..., <параметр n>)

Значение

PCALL:	Кодовое слово для вызова подпрограммы с абсолютным указанием пути.
<путь/имя программы>:	Абсолютное указание пути включая имя подпрограммы. Правила указания пути см. "Адресация файлов программной памяти (Страница 574)". Если абсолютный путь не был указан, то поведение PCALL идентично стандартному вызову подпрограммы с идентификатором программы. Имя программы указывается без префикса и без идентификатора файла. Если имя программы нужно запрограммировать с префиксом и идентификатором файла, то оно должно быть явно интерпретировано с префиксом и идентификатором файла с помощью команды EXTERN.
<параметр 1>, ...,:	Актуальные параметры согласно оператору PROC подпрограммы.

Пример

Программный код
PCALL/_N_WKS_DIR/_N_WELLE_WPD/WELLE(параметр1, параметр2,...)

4.2.3.9 Расширить маршрут поиска для вызовов подпрограмм (CALLPATH)

С помощью команды CALLPATH можно расширить маршрут поиска для вызовов подпрограммы. Таким образом, могут вызываться и подпрограммы из не выбранной директории детали без указания полного, абсолютного имени пути подпрограммы.

Другой способ применения предлагается в режиме работы EES "EES без GDIR", когда для хранения глобальных подпрограмм используется директория во внешней программной памяти. В этом случае с помощью CALLPATH можно расширить маршрут поиска, включив в него эту директорию подпрограммы.

Расширение маршрута поиска выполняется перед строкой для циклов пользователя (_N_CUS-DIR).

Следующие события снова отменяют расширение маршрута поиска:

- CALLPATH с пробелом
- CALLPATH без параметров
- Конец программы обработки детали
- Reset

Синтаксис

```
CALLPATH ("<имя маршрута>")
```

Объяснение

CALLPATH:	Кодовое слово для программируемого расширения маршрута поиска. Программируется в собственной строке программы обработки детали.
<имя маршрута>:	Постоянная или переменная типа STRING. Содержит абсолютное указание пути к директории, на которую должен быть расширен маршрут поиска. Правила указания пути см. "Адресация файлов программной памяти (Страница 574)".

Пример

Маршрут поиска нужно расширить на определенную директорию детали:

Программный код

```
...
CALLPATH ("/_N_WKS_DIR/_N_MYWPD_WPD")
...
```

Здесь устанавливается следующий маршрут поиска (позиция 5 является новой):

1. Актуальная директория/*name*
2. Актуальная директория/*name_SPF*
3. Актуальная директория/*name_MPF*
4. //NC:/_N_SPF_DIR/*name_SPF*
5. **/_N_WKS_DIR/_N_MYWPD_WPD/name_SPF**

6. /N_CUS_DIR/*name_SPF*
7. /_N_CMA_DIR / *name_SPF*
8. /_N_CST_DIR / *name_SPF*

Граничные условия

- CALLPATH проверяет, существует ли запрограммированное имя маршрута в действительности. В случае ошибки выполнение программы обработки детали отменяется с ошибкой кадра коррекции 14009.
- CALLPATH может быть запрограммирована и в файлах INI. В этом случае он действует в течение срока обработки файла INI (файл WPD-INI или программа инициализации для активных данных ЧПУ, к примеру, фреймы в 1-ом канале _N_CH1_UFR_INI). После маршрут поиска снова сбрасывается.

4.2.3.10 Выполнение внешней подпрограммы (EXTCALL)

С помощью команды EXTCALL можно догрузить и выполнить программу обработки детали из внешнего ЗУ.

В качестве внешнего ЗУ предлагаются:

- Локальный диск
- Сетевой диск
- Диск USB

Примечание

В качестве интерфейса для обработки находящейся на USB-диске внешней программы можно использовать только интерфейсы USB на лицевой стороне пульта управления или TCU.

ВНИМАНИЕ

Повреждение инструмента/детали при использовании USB-флэшки

Не рекомендуется использовать USB-флэшку для обработки внешней подпрограммы. Нарушение коммуникации с USB-флэшкой при выполнении программы обработки детали из-за плохого контакта, толчков или случайного извлечения приводит к немедленной остановке обработки. При этом инструмент и/или деталь могут быть повреждены.

Предустановка ветви программы

Путь к внешней директории программы может быть предустановлен с помощью установочных данных:

SD42700 $SC_EXT_PROG_PATH

В комбинации с указанной при вызове EXTCALL ветвью программы и идентификатором подпрограммы из этого получается общий путь к вызываемой программе обработки детали.

Примечание

Если ветвь программы указывается только через вызов EXTCALL, в SD42700 должно быть пусто!

Примечание

Параметр

При вызове внешней программы передача в нее параметров невозможна.

Синтаксис

EXTCALL ("<путь/><имя программы>")

Значение

EXTCALL:	Команда для вызова внешней подпрограммы	
"<путь/><имя программы>":	Постоянная/переменная типа STRING	
	<путь/>:	Абсолютное или относительное указание пути (**опция**)
	<имя программы>:	Имя программы указывается без префикса "_N_". Расширение файла ("MPF", "SPF") может быть добавлено с символом "_" или "." к имени программы (**опция**). Пример: "WELLE" "WELLE_SPF" "WELLE.SPF"

Указание пути: Краткие обозначения

При указании пути можно использовать следующие условные обозначения:

- Локальный диск: "**LOCAL_DRIVE:**"
- Карта памяти: "**CF_CARD:**"
- USB-диск (панель оператора): "**USB:**"

Краткие обозначения "CF_CARD:" и "LOCAL_DRIVE:" можно использовать в качестве альтернативы.

Пример

Выполнение с локального диска

Главная программа "MAIN.MPF" находится в памяти ЧПУ и выбрана для выполнения.

Подпрограмма "SP_1"

Внешняя подпрограмма "SP_1.SPF" или "SP_1.MPF" находится на локальном диске в директории "/user/sinumerik/data/prog/WKS.DIR/WST1.WPD".

Путь к внешней директории программ устанавливается с:

SD42700 $SC_EXT_PROG_PATH = LOCAL_DRIVE:WKS.DIR/WST1.WPD

Примечание

Указание пути для вызова внешней подпрограммы:

- Без использования предустановки: "LOCAL_DRIVE:WKS.DIR/WST1.WPD/SP_1"
- С использованием предустановки: "SP_1"

Подпрограмма "SP_2"

Внешняя подпрограмма "SP_2.SPF" или "SP_2.MPF" находится в директории WKS.DIR/WST1.WPD диска USB. Предустановка пути к внешней директории программ используется для пути подпрограммы "SP_1" и также не переписывается в главной программе. Поэтому при вызове подпрограммы "SP_2" должен быть указан полный путь.

Главная программа "MAIN"

Программный код

```
N010 PROC MAIN
N020 ...
N030 EXTCALL("SP_1")
N030 EXTCALL("USB:WKS.DIR/WST1.WPD/SP_2")
N050 ...
N060 M30
```

Дополнительная информация

Вызов EXTCALL с абсолютным указанием пути

Если подпрограмма существует по указанному пути, то она выполняется с вызовом EXTCALL. Если подпрограмма отсутствует по указанному пути, то выполнение программы отменяется с вызовом EXTCALL.

Вызов EXTCALL с относительным указанием пути / без указания пути

При вызове EXTCALL с относительным указанием пути или без указания пути поиск в имеющейся программной памяти осуществляется по следующему образцу:

1. Если в SD42700 $SC_EXT_PROG_PATH предустановленно указание пути, то сначала, исходя из этого пути, выполняется поиск данных в вызове EXTCALL (имя программы возможно с относительным указанием пути). В этом случае абсолютный путь получается через соединение символов:
   - Предустановленное указание пути в SD42700 $SC_EXT_PROG_PATH
   - Разделительный символ "/"
   - Указание пути и имя подпрограммы в команде EXTCALL
2. Если подпрограмма не была найдена в 1, то поиск выполняется в директориях памяти пользователя.

Поиск прекращается при первом нахождении подпрограммы. Если подпрограмма не найдена, то выполнение программы отменяется с вызовом EXTCALL.

Настраиваемая память догрузки (буфер FIFO)

Для выполнения внешней подпрограммы потребуется память догрузки. Размер памяти догрузки предустановлен на 30 кБ и может быть изменен только изготовителем станка (через MD18360 MM_EXT_PROG_BUFFER_SIZE).

Примечание

Подпрограммы с операторами переходов

Для внешних подпрограмм, содержащих операторы перехода (GOTOF, GOTOB, CASE, FOR, LOOP, WHILE, REPEAT, IF, ELSE, ENDIF и т. п.), цели перехода должны находиться в памяти догрузки.

Это условие, как правило, является проблемой для операторов перехода на начало программы (GOTOS), так как программы обычно имеют большой размер для помещения в память догрузки. При первой догрузке начало программы удаляется из памяти догрузки. Если при дальнейшем выполнении программы потребуется обработать оператор перехода на начало программы, тогда функция будет не в состоянии найти цель перехода. Происходит отмена программы с выводом аварийного сообщения 14000.

Для выполнения внешних подпрограмм без ограничений в части запрограммированных операторов перехода, рекомендуется вместо функции "Выполнение внешних подпрограмм (EXTCALL)" использовать функцию "Выполнение из внешней памяти (EES)".

Примечание

Программы ShopMill-/ShopTurn

Программы ShopMill и ShopTurn из-за прикрепленных на конце файла описаний контура должны целиком лежать в памяти догрузки.

Для выполняемых параллельно внешних подпрограмм требуется своя память догрузки.

Reset / конец программы / POWER ON

RESET и POWER ON отменяют внешние вызовы подпрограмм и соответствующая память догрузки стирается.

Выбранная для "Выполнения с внешнего устройства" подпрограмма остается выбранной для "Выполнения с внешнего устройства" и после Reset, завершения программы или POWER ON. Поведение не отличается от внутренне выбранных программ, если внешняя программная память остается доступной.

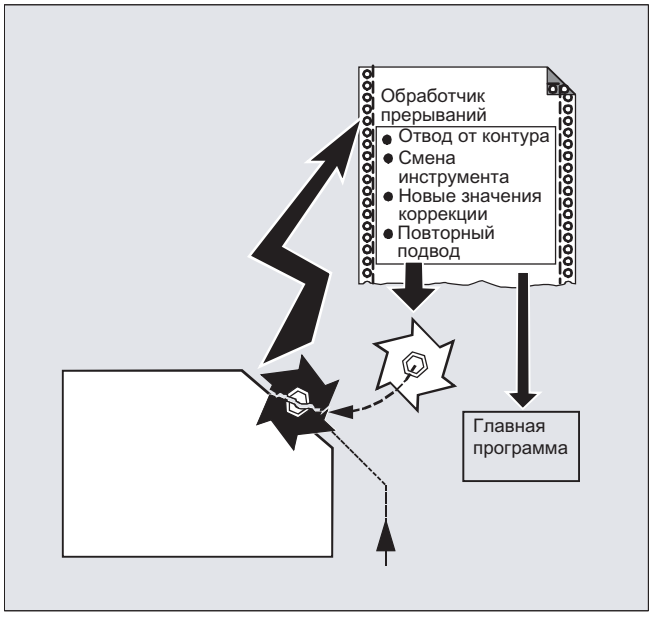
4.3 Обработчик прерываний (ASUP)

4.3.1 Функция обработчика прерываний

Примечание

Попеременно встречающиеся в описании ниже понятия "Асинхронная подпрограмма (ASUP)" и "Обработчик прерываний" обозначают одну и ту же функциональность.

Функция обработчика прерываний будет пояснена на типичном примере:



При обработке происходит поломка инструмента. Следствием этого является сигнал, останавливающий текущий процесс обработки и одновременно запускающий подпрограмму – так называемый обработчик прерываний. В этой подпрограмме находятся все операторы, которые должны быть выполнены в этом случае.

После выполнения подпрограммы (и восстановления готовности к работе) СЧПУ возвращается к главной программе и продолжает обработку – в зависимости от команды REPOS – в месте прерывания (см. "Повторный подвод к контуру (Страница 893)").

⚠ ОСТОРОЖНО

Опасность столкновений

Если в подпрограмме команда REPOS не запрограммирована, то позиционирование выполняется на конечную точку кадра, следующего за прерванным кадром.

4.3.2 Создание обработчика прерываний

Создание обработчика прерываний как подпрограммы

Обработчик прерываний при определении обозначается как подпрограмма.

Пример:

Программный код	Комментарий
`PROC ABHEB_Z`	; Имя программы "ABHEB_Z"
`N10 ...`	; Затем следуют кадры ЧПУ.
...	
`N50 M17`	; В заключении конец программы и возврат в главную программу.

Архивирование модальных G-команд (Save)

При определении обработчик прерываний может быть обозначен посредством SAVE.

Следствием атрибута SAVE является сохранение активных модальных G-команд перед вызовом обработчика прерываний и их повторная активация после завершения обработчика прерываний (см. " Подпрограммы с механизмом SAVE (SAVE) (Страница 519) ").

Таким образом можно продолжить обработку после выполнения обработчика прерываний в месте прерывания.

Пример:

Программный код
`PROC ABHEB_Z SAVE`
`N10 ...`
...
`N50 M17`

Согласование дополнительных обработчиков прерываний (SETINT)

Внутри обработчика прерываний могут быть запрограммированы операторы SETINT (см. "Согласование и запуск обработчика прерываний (SETINT)" (Страница 561)), тем самым предварительно активируя дополнительные обработчики прерываний. Запуск выполняется только через вход.

4.3.3 Согласование и запуск обработчика прерываний (SETINT, PRIO, BLSYNC)

СЧПУ имеет несколько быстрых входов (вход 1 ... 8), каждый из которых вызывает прерывание (1 ... 8). Каждому прерыванию можно с помощью команды `SETINT` сопоставить приоритет и обработчик прерываний. Если прерывание вызывается установкой быстрого входа, то текущая обработка в канале прерывается и запускается обработчик прерываний.

Приоритет прерывания

Если в какой-либо программе обработки детали прерывания назначаются нескольким входам, нужно присвоить прерываниям разные приоритеты.

Прерыванию можно присвоить одно из значений приоритета 1 ... 128. Значение приоритета 1 соответствует высшему, 128 – низшему приоритету.

Синтаксис

```
SETINT(<n>) <ИМЯ>
SETINT(<n>) PRIO=<значение> <ИМЯ>
SETINT(<n>) PRIO=<значение> <ИМЯ> BLSYNC
SETINT(<n>) PRIO=<значение> <ИМЯ> LIFTFAST
```

Значение

`SETINT(<n>):`	Сигналу прерывания <n> назначается программа ЧПУ (ASUP) <имя>. Назначенный обработчик прерываний запускается, как только распознается сигнал прерывания <n> == 1. **Примечание:** Если сигналу прерывания <n> назначается другой обработчик прерываний, предыдущее назначение перестает действовать.	
`<n>`:	Номер сигнала прерывания	
	Тип:	INT
	Диапазон значений:	1 ... 32
`PRIO=`:	Приоритет прерывания (опционально)	
<значение>:	(**опция**) Значение приоритета	
	Тип:	INT
	Диапазон значений:	1 ... 128 (1 ⇒ высший приоритет)
<ИМЯ>:	Имя программы ЧПУ (ASUP)	

BLSYNC:	(**опция**) BLSYNC действует таким образом, что после вызова прерывания сначала ожидается завершение обработки актуального кадра. Только после этого выполняется обработчик прерываний.
LIFTFAST:	(**опция**) LIFTFAST действует таким образом, что после вызова прерывания сначала происходит быстрый отвод (см. главу «Быстрый отвод от контура (SETINT LIFTFAST, ALF) (Страница 564)»). Только после этого выполняется обработчик прерываний.

Граничные условия

Правила обработки прерываний

1. Для каждого прерывания, которое не может обрабатываться сразу или в данный момент уже находится в обработке, сохраняется один дополнительный запрос прерывания. Остальные запросы этого прерывания утрачиваются.
2. Если в данный момент обрабатывается прерывание и вызывается еще одно прерывание с более высоким приоритетом, то оно прерывает менее приоритетное прерывание. После завершения более приоритетного прерывания продолжается обработка менее приоритетного прерывания. Если во время обработки более приоритетного прерывания поступают ещё запросы на менее приоритетное прерывание, то сохраняется один запрос. Остальные утрачиваются.
3. Если в данный момент обрабатывается прерывание и вызывается еще одно прерывание с более высоким приоритетом, то оно прерывает менее приоритетное прерывание. Более приоритетное прерывание обрабатывается. Если снова вызывается более приоритетное прерывание, то текущее прерывание прерывается, и обрабатывается более приоритетное прерывание. Максимально возможны шесть активных уровней прерывания. Один обрабатываемый в данный момент уровень прерывания и пять ожидающих уровней прерывания. Для каждого активного уровня прерывания сохраняется не более одного дополнительного запроса прерывания. Все остальные запросы прерывания утрачиваются. Запрос прерывания утрачивается и в том случае, если он направляется на остальные уровни прерывания (уровень прерывания ≥ 7).

Примеры

Пример 1: Согласование обработчиков прерываний и установка приоритета

Программный код	Комментарий
N20 SETINT(3) PRIO=1 ABHEB_Z	; IF вход 3 == 1
	; THEN запуск обработчика прерываний «ABHEB_Z»
N30 SETINT(2) PRIO=2 ABHEB_X	; IF вход 2 == 1
	; THEN запуск обработчика прерываний «ABHEB_X»

Программы последовательно выполняются в очередности значений приоритета при одновременной готовности входов: сначала "ABHEB_Z", потом "ABHEB_X".

Пример 2: Новое согласование обработчика прерываний

Программный код	Комментарий
`N20 SETINT(3) PRIO=2 ABHEB_Z`	`;` `IF` вход 3 == 1
	`;` `THEN` запуск обработчика прерываний «**`ABHEB_Z`**»
...	
`N80 SETINT(3) PRIO=1 ABHEB_X`	`;` `IF` вход 3 == 1
	`;` `THEN` запуск обработчика прерываний «**`ABHEB_X`**»

4.3.4 Деактивация/повторная активация согласования обработчика прерываний (DISABLE, ENABLE)

Оператор `SETINT` может быть деактивирован с помощью `DISABLE` и снова активирован с помощью `ENABLE` без потери согласования "Вход → Обработчик прерываний".

Синтаксис

```
DISABLE(<n>)
ENABLE(<n>)
```

Значение

`DISABLE(<n>)`:	Команда: **Деактивация** согласования обработчика прерываний входа <n>	
`ENABLE(<n>)`:	Команда: **Реактивация** согласования обработчика прерываний входа <n>	
<n>:	Параметры: Номер сигнала прерывания	
	Тип:	INT
	Диапазон значений:	1 ... 32

Пример

Программный код	Комментарий
`N20 SETINT(3) PRIO=1 ABHEB_Z`	; при включении входа 3 должен запуститься
	; обработчик прерываний «`ABHEB_Z`».
...	
`N90 DISABLE(3)`	; Оператор `SETINT` из `N20` деактивируется.
...	
`N130 ENABLE(3)`	; Оператор `SETINT` из `N20` снова активируется.
...	

4.3.5 Удаление согласования обработчика прерываний (CLRINT)

Заданное через `SETINT` назначение сигнала прерывания программе ЧПУ (ASUP) можно удалить с помощью `CLRINT`.

Синтаксис

```
CLRINT(<n>)
```

Значение

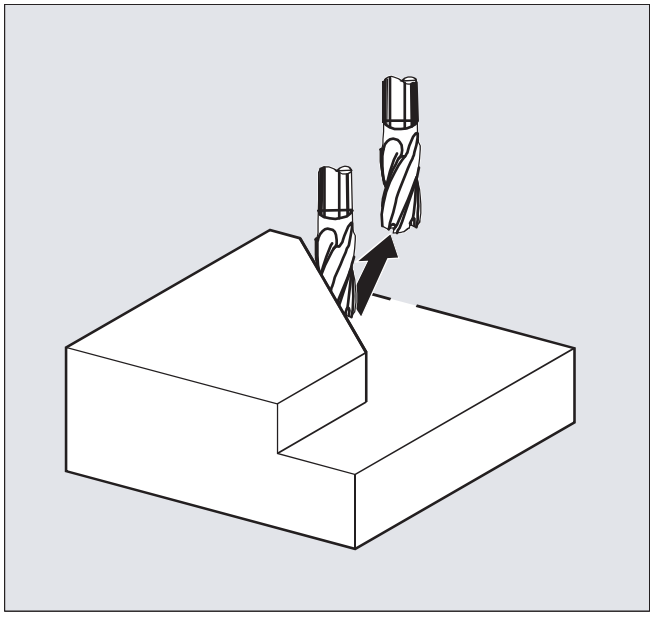
CLRINT(<n>):	Команда: Удаление назначения сигнала прерывания <n> программе ЧПУ (ASUP) <n>, определенной с помощью SETINT	
<n>:	Параметры: Номер сигнала прерывания	
	Тип:	INT
	Диапазон значений:	1 ... 32

Пример

Программный код	Комментарий
N20 SETINT(3) PRIO=2 ABHEB_Z	
...	
N50 CLRINT(3)	; Согласование между входом "3" и обработчиком прерываний "ABHEB_Z" удалено.

4.3.6 Быстрый отвод от контура (SETINT LIFTFAST, ALF)

При операторе SETINT с LIFTFAST при срабатывании входа инструмент движется от контура детали с быстрым отводом.



Дальнейший процесс зависит от того, содержит ли оператор SETINT наряду с LIFTFAST обработчик прерываний:

С обработчиком прерываний:	**После** быстрого отвода выполняется обработчик прерываний.
Без обработчика прерываний:	Обработка после быстрого отвода останавливается с ошибкой.

Синтаксис

```
SETINT(<n>) PRIO=1 LIFTFAST
SETINT(<n>) PRIO=1 <ИМЯ> LIFTFAST
```

Объяснение

SETINT(<n>):	Команда: Согласовать вход <n> с обработчиком прерываний. Согласованный обработчик прерываний запускается при срабатывании входа <n>.	
<n>:	Параметры: Номер входа	
	Тип:	INT
	Диапазон значений:	1 ... 8
PRIO= :	Определение приоритета	
<значение>:	Значение приоритета	
	Диапазон значений:	1 ... 128
	Приоритет 1 соответствует высшему приоритету.	
<ИМЯ>:	Имя подпрограммы (обработчика прерываний), которая должна быть выполнена.	
LIFTFAST:	Команда: Быстрый отвод от контура	
ALF=... :	Команда: Программируемое направление перемещения (стоит в кадре движения) По возможностям программирования с ALF см. тему " Направление перемещения при быстром отводе от контура (Страница 567) ".	

Граничные условия

Поведение при активном фрейме с отражением

При определении направления отвода проверяется, активен ли фрейм с отражением. В этом случае направление отвода относительно направления касательной изменяется справа налево. Составляющие направления в направлении инструмента не отражаются. Такое поведение активируется через установку MD:

MD21202 $MC_LIFTFAST_WITH_MIRROR = TRUE

Пример

Сломанный инструмент должен быть автоматически заменен однотипным инструментом. После обработка продолжается с новым инструментом.

Главная программа:

Главная программа	Комментарий
N10 SETINT(1) PRIO=1 W_WECHS LIFTFAST	; При включении входа 1 инструмент сразу же отводится быстрым отводом (код Nr. 7 для коррекции радиуса инструмента G41) от контура. После выполняется обработчик прерываний "W_WECHS".
N20 G0 Z100 G17 T1 ALF=7 D1	
N30 G0 X-5 Y-22 Z2 M3 S300	
N40 Z-7	
N50 G41 G1 X16 Y16 F200	
N60 Y35	
N70 X53 Y65	
N90 X71.5 Y16	
N100 X16	
N110 G40 G0 Z100 M30	

Подпрограмма:

Подпрограмма	Комментарий
PROC W_WECHS SAVE	; Подпрограмма с сохранением актуального рабочего состояния
N10 G0 Z100 M5	; Позиция смены инструмента, стоп шпинделя
N20 T11 M6 D1 G41	; Смена инструмента
N30 REPOSL RMBBL M3	; Повторный подвод к контуру и возврат в главную программу (программируется в одном кадре)

4.3.7 Направление перемещения при быстром отводе от контура

Движение отвода

Плоскость движения отвода определяется следующими G-командами:

- LFTXT
  Плоскость движения отвода определяется из касательной к траектории и направления инструмента (стандартная настройка).
- LFWP
  Плоскость движения отвода — это активная рабочая плоскость, выбираемая G-командами G17, G18 или G19. Направление движения отвода не зависит от касательной к траектории. Таким образом можно запрограммировать параллельный оси быстрый отвод.
- LFPOS
  Отвод названной с помощью POLFMASK / POLFMLIN оси на запрограммированную с POLF абсолютную позицию оси.
  ALF не влияет на направление отвода для нескольких осей, а также для нескольких осей в линейной связи.

Программируемое направление перемещения (ALF=...)

В плоскости движения отвода с помощью ALF программируется направление с дискретным шагом в 45 градусов.

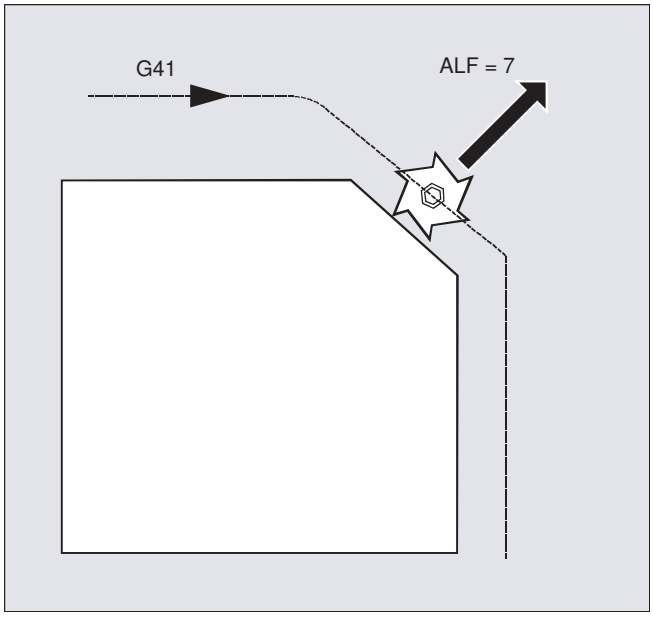
Возможные направления перемещения сохранены в СЧПУ под специальными кодовыми номерами, по которым они могут вызываться.

Пример:

Программный код

```
N10 SETINT(2) PRIO=1 ABHEB_Z LIFTFAST
ALF=7
```

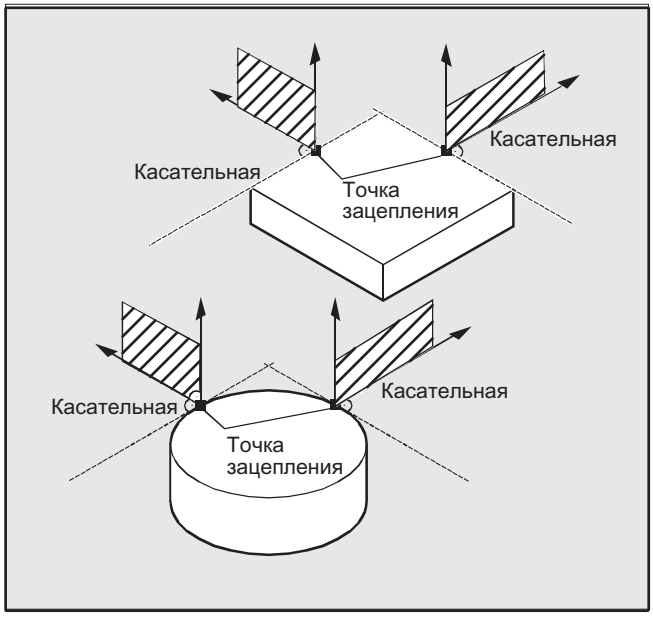
Инструмент движется с включенной G41 (направление обработки слева от контура) вертикально от контура.



Базовая плоскость для описания направлений перемещения при LFTXT

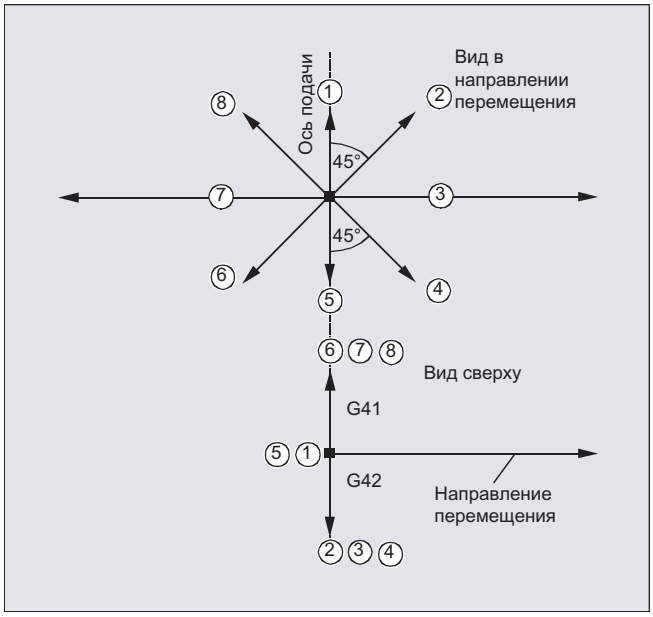
В точке касания инструмента на запрограммированном контуре создается плоскость, служащая базой для указания движения отвода с соответствующим кодовым номером.

Базовая плоскость создается из продольной оси инструмента (направление подачи) и вектора к этой оси вертикального к касательной в точке касания инструмента на контуре.



Кодовые номера с направлениям перемещения при LFTXT

Исходя из базовой плоскости, на рисунке ниже указаны кодовые номера с направлениями перемещения.



Для ALF=1 определен отвод в направлении инструмента.

При ALF=0 функция "Быстрый отвод" отключена.

⚠ ОСТОРОЖНО

Опасность столкновений

При включенной коррекции радиуса инструмента:

- при G41 кодировки 2, 3, 4
- при G42 кодировки 6, 7, 8

не должны использоваться, т.к. в этих случаях произошел бы подвод инструмента к контуру и столкновение с деталью.

Кодовые номера с направлениям перемещения при LFWP

При LFWP направление в рабочей плоскости получается согласно следующему согласованию:

- G17: плоскость X/Y
  ALF=1: отвод в направлении X
  ALF=3: отвод в направлении Y
- G18: плоскость Z/X
  ALF=1: отвод в направлении Z
  ALF=3: отвод в направлении X
- G19: плоскость Y/Z
  ALF=1: отвод в направлении Y
  ALF=3: отвод в направлении Z

См. также

Быстрый обратный ход при резьбонарезании (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN) (Страница 239)

4.3.8 Процесс движения для обработчиков прерываний

Обработчик прерываний без LIFTFAST

Движения осей затормаживаются на траектории до состояния покоя. После запускается обработчик прерываний.

Позиция покоя сохраняется как позиция прерывания и при `REPOS` с `RMIBL` в конце обработчика прерываний к ней выполняется подвод.

Обработчик прерываний с LIFTFAST

Движения осей затормаживаются на траектории. Одновременно движение `LIFTFAST` выполняется как наложенное движение. После остановки движения по траектории и движения `LIFTFAST` запускается обработчик прерываний.

В качестве позиции прерывания сохраняется позиция на контуре, на которой было запущено движение `LIFTFAST` и из-за этого произошел выход из траектории.

Поведение обработчика прерываний с `LIFTFAST` и `ALF=0` идентично обработчику прерываний без `LIFTFAST`.

Примечание

Значение, на которое геометрические оси при быстром отводе отводятся от контура, может быть установлено через машинные данные.

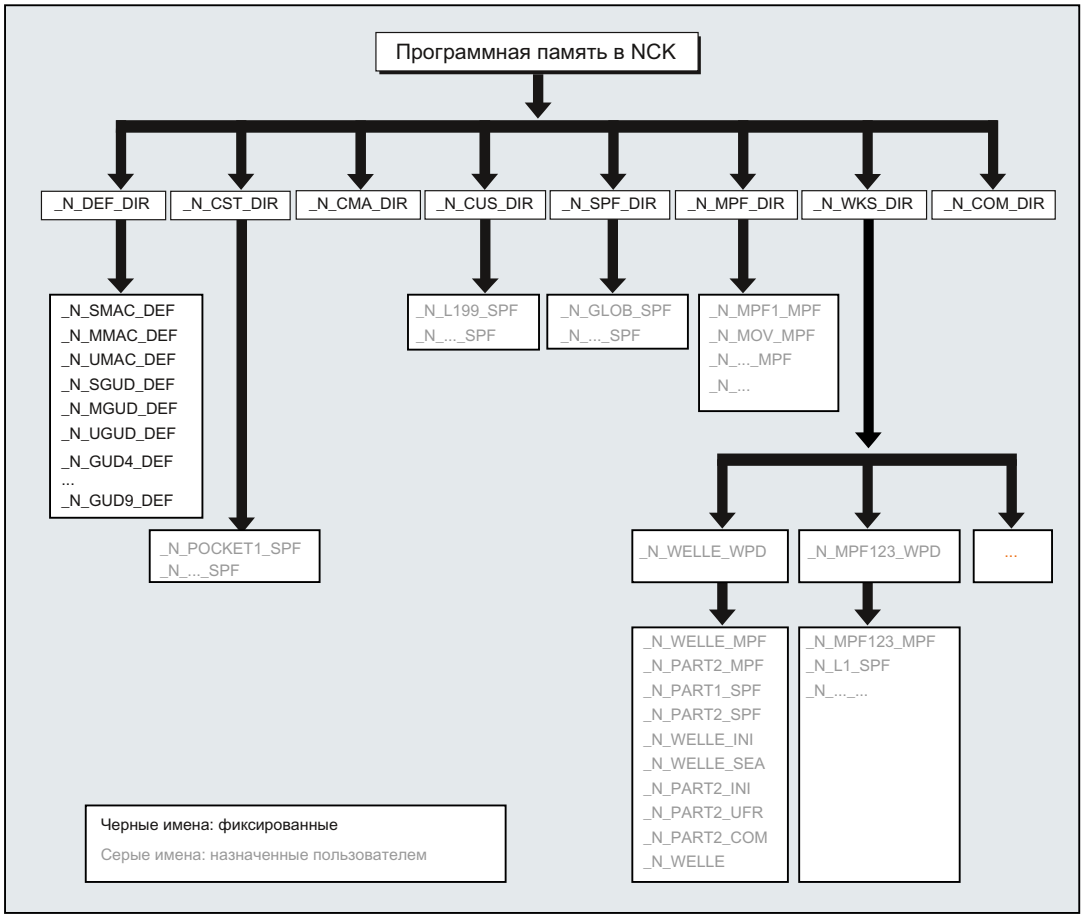
4.4 Управление файлами и программами

4.4.1 Программная память

4.4.1.1 Программная память в NCK

В программной памяти файлы и программы (к примеру, главные и подпрограммы, макроопределения) сохраняются постоянно (→ пассивная файловая система).

Наряду с этим существует часть типов файлов, которые могут быть здесь буферизированы и при необходимости (к примеру, при обработке определенной детали) передаваться в оперативной память (к примеру, для инициализации).



Стандартные директории

Следующие директории имеются стандартно:

Директория	Содержание
_N_DEF_DIR	Блоки данных и макроблоки
_N_CST_DIR	Стандартные циклы
_N_CMA_DIR	Циклы изготовителя
_N_CUS_DIR	Циклы пользователя
_N_WKS_DIR	Детали
_N_SPF_DIR	Глобальные подпрограммы
_N_MPF_DIR	Главные программы
_N_COM_DIR	Комментарии

Типы файлов

В программной памяти могут быть размещены следующие типы файлов:

Тип файла	Описание
<имя>_MPF	Главная программа
<имя>_SPF	Подпрограмма
<имя>_TEA	Машинные данные
<имя>_SEA	Установочные данные
<имя>_TOA	Коррекции инструмента
<имя>_UFR	Смещения нулевой точки/фрейм
<имя>_INI	Файл инициализации
<имя>_GUD	Глобальные данные пользователя
<имя>_RPA	R-параметры
<имя>_COM	Комментарий
<имя>_DEF	Определения для глобальных данных пользователя и макросов

Главная директория деталей (_N_WKS_DIR)

Главная директория деталей по умолчанию называется _N_WKS_DIR и находится в программной памяти. Главная директория детали содержит для всех запрограммированных деталей соответствующие каталоги деталей.

Директории деталей (..._WPD)

Директория детали содержит все файлы, необходимые для обработки детали. Это могут быть главные программы, подпрограммы, любые программы инициализации и файлы комментариев.

Программы инициализации выполняются один раз после выбора программы при первом запуске программы обработки детали (согласно машинным данным MD11280 $MN_WPD_INI_MODE).

Пример:

Директория детали _N_WELLE_WPD, созданная для детали WELLE, содержит следующие файлы:

Файл	Описание
_N_WELLE_MPF	Главная программа
_N_PART2_MPF	Главная программа
_N_PART1_SPF	Подпрограмма
_N_PART2_SPF	Подпрограмма
_N_WELLE_INI	Общая программа инициализации данных для детали
_N_WELLE_SEA	Программа инициализации установочных данных
_N_PART2_INI	Общая программа инициализации данных для программы Part 2

Файл	Описание
_N_PART2_UFR	Программа инициализации для фрейм-данных для программы Part 2
_N_WELLE_COM	Файл комментария

Дополнительно в директории детали можно хранить и такие данные, которые не требуются ЧПУ для непосредственной обработки. Это могут быть не только файлы ASCII, но и двоичные файлы, например изображения в формате JPG или описания в формате PDF. Чтобы ЧПУ могло интерпретировать их как двоичные файлы, расширения этих файлов должны быть известны в ЧПУ (настройка при вводе в эксплуатацию через MD17000 $MN_ EXTENSIONS_OF_BIN_FILES; по умолчанию предустановлены следующие расширения файлов: JPG, GIF, PNG, BMP, PDF, ICO, HTM).

Выбрать деталь для обработки

Директория детали может быть выбрана для выполнения в одном канале. Если в этой директории находится главная программа **с тем же именем** или только единственная главная программа (_MPF), то она автоматически выбирается для обработки.

4.4.1.2 Внешняя программная память

Наряду с пассивной файловой системой в ЧПУ на станке может быть доступна и внешняя программная память (например, на локальном или сетевом диске).

С помощью функций "Выполнение с внешнего устройства" или "EES (Execution from External Storage)" возможно выполнение программ обработки детали **напрямую** из внешней программной памяти.

Дополнительная информация: Описание функций – Основные функции

Глобальная память программ обработки деталей (GDIR)

При описании дисков один из них можно отметить как глобальную память программ обработки деталей (GDIR).

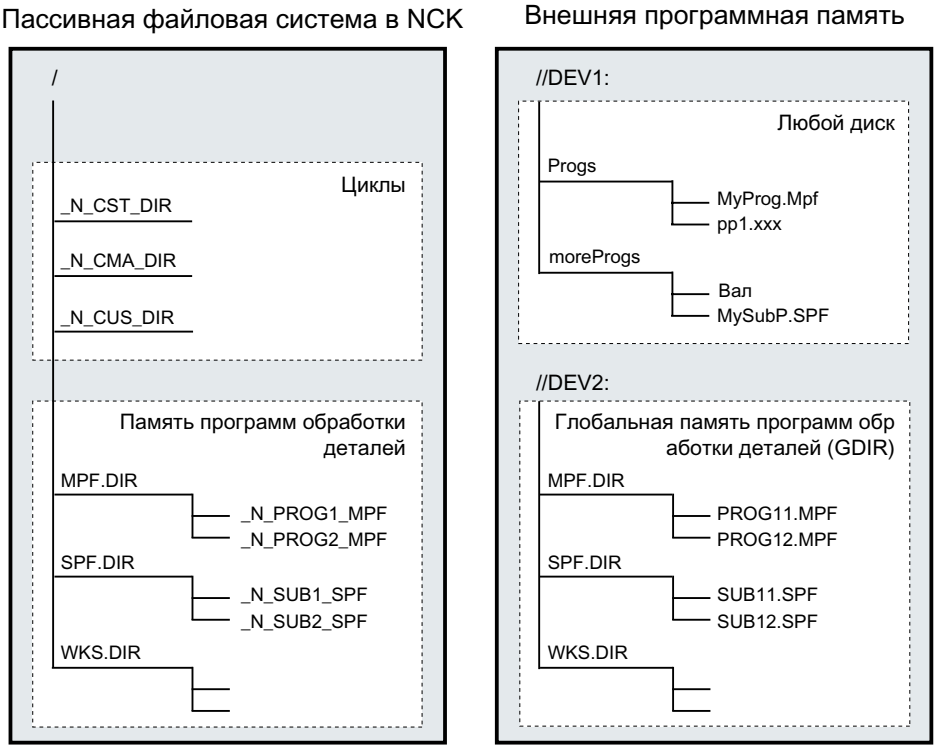
В этом случае система автоматически создает на диске директории MPF.DIR, SPF.DIR и WKS.DIR. Эти три директории образуют память GDIR.

Память GDIR играет роль исключительно для функции EES. В зависимости от конфигурации диска память GDIR заменяет или расширяет память программ обработки деталей ЧПУ. Тем не менее, настройка памяти GDIR для режима EES не обязательна.

Директории и файлы памяти GDIR могут адресоваться в программе обработки детали точно так же, как в пассивной файловой системе. При этом возможен совместимый перенос программы ЧПУ с указанием пути из пассивной файловой системы в память GDIR. Директория SPF.DIR памяти GDIR содержится в маршруте поиска для подпрограмм.

Организация программы

На следующем рисунке показана организация программы во внешней программной памяти:



Не зависящие от регистра файловые системы

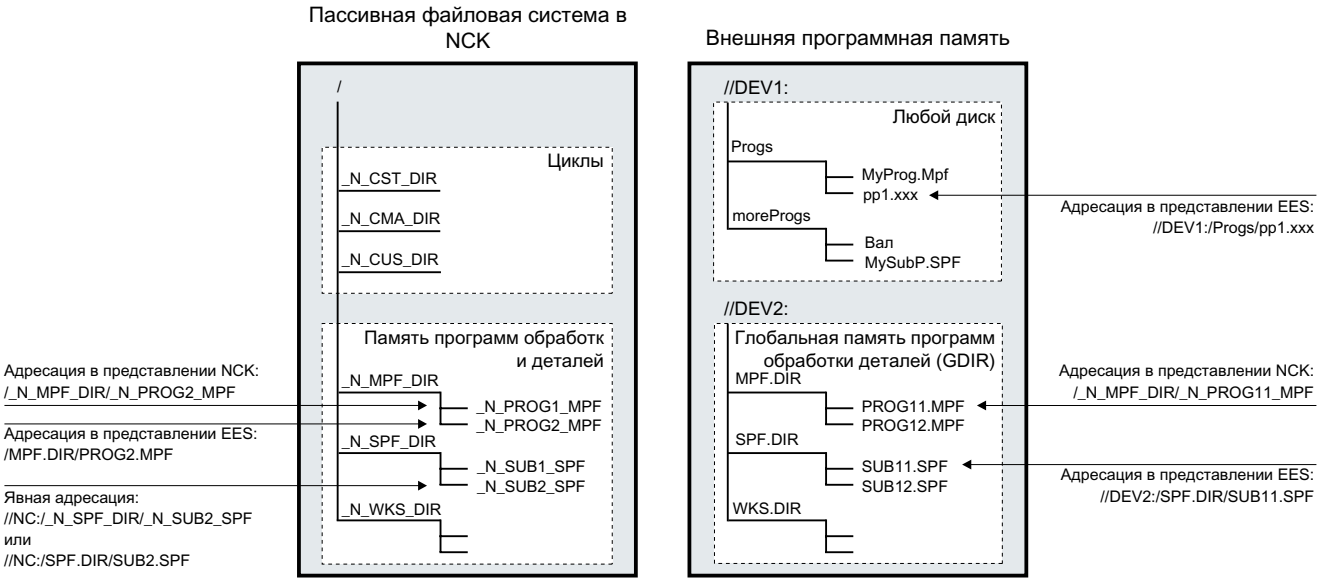
Примечание

Чтобы избежать проблем с прописным и строчным написанием при адресации файлов (см. "Адресация файлов программной памяти (Страница 574)"), в качестве внешней программной памяти следует использовать файловые системы, **не зависящие от регистра**.

4.4.1.3 Адресация файлов программной памяти

На файл в программной памяти, который запрашивается через команду для работы с файлами (например WRITE, DELETE, READ, ISFILE, FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO), дается отсылка либо через абсолютный путь плюс имя файла, либо только через имя файла. Во втором случае в качестве пути файла используется путь выбранной программы.

Адресация в представлении ЧПУ/EES



Адресация файлов пассивной файловой системы

Адресация файлов пассивной файловой системы обычно осуществляется в **представлении ЧПУ** (имена директорий и файлов начинаются с идентификатора домена "_N_", разделительным символом для расширения директории/файла является "_") без указания имени диска. Но и адресация в **представлении EES** (без идентификатора домена "_N_", разделительным символом для расширения директории/файла является ".") тоже допустима.

Пример:

- Представление ЧПУ: "/_N_SPF_DIR/_N_SUB1_SPF"
- Представление EES: "/SPF.DIR/SUB1.SPF"

Примечание

Адресации файлов пассивной файловой системы в представлении EES преобразуются внутри в представление ЧПУ по следующим правилам:

- К именам директорий и файлов добавляется идентификатор домена "_N_".
- Если четвертым с конца символом в имени директории или файла является точка ("."), она преобразуется в символ подчеркивания ("_").

Через предопределенное имя диска "//NC:" в пассивной файловой системе тоже возможна целенаправленная адресация.

Пример:

- Представление ЧПУ: "//NC:/_N_SPF_DIR/_N_SUB1_SPF"
- Представление EES: "//NC:/SPF.DIR/SUB1.SPF"

Адресация файлов внешней программной памяти

Адресация файлов внешней программной памяти, не отмеченной как память GDIR, должна осуществляться в представлении EES. В начале пути адресации необходимо указать имя диска (например, "//DEV1:"). Допустимы все символические имена устройств, сконфигурированные в /user/sinumerik/hmi/cfg/logdrive.ini.

Пример:

- Представление EES: "//DEV1:/MyProgDir/pp1.xxx"
- Представление ЧПУ: недопустимо

Адресация файлов глобальной памяти программ обработки деталей (GDIR)

При адресации файлов памяти GDIR наряду с указанием пути в представлении EES допускается и указание пути в представлении ЧПУ.

Пример:

- Представление EES: "//DEV2:/MPF.DIR/PROG11.MPF"
- Представление ЧПУ: "/_N_MPF_DIR/_N_PROG11_MPF"

Примечание

Адресации файлов памяти GDIR в представлении ЧПУ преобразуются внутри в представление EES по следующим правилам:

- Идентификатор домена "_N_" в имени директории и файла удаляется.
- Если четвертым с конца символом в имени директории или файла является символ подчеркивания ("_"), он преобразуется в точку (".").

Правила указания пути

Полное указание пути состоит из имени диска, пути директории и имени файла.

Имя диска

Для указания имени диска действуют следующие правила:

- Допустимы все символические имена устройств, сконфигурированные в /user/sinumerik/hmi/cfg/logdrive.ini.
- В начале стоит символ "//", за ним следует как минимум одна буква или цифра.
- Последующими символами могут быть буквы, цифры, "_" и пробел в любой комбинации.
- Завершается имя буквой или цифрой, за которой следует двоеточие ":".
- Прочие специальные символы запрещены.

Примечание

Для пассивной файловой системы предопределено имя диска "//NC:".

Примеры:

- Внешняя программная память:
  - //Drive1:
  - //Drive_1:
  - //Drive 1:
  - //A B:
  - //1 B C 2:

Путь директории

Для указания пути директории действуют следующие правила:

- В начале и в конце пути директории и в качестве разделительного символа для отдельных участков пути используется "/".

  Примечание

  Двойная косая черта ("//") внутри пути директории **недопустима**!

- Имена директорий:
  - Имена директорий должны начинаться с буквы или цифры. Только при адресации в представлении ЧПУ имена директорий начинаются с идентификатора домена "_N_".
  - Последующими символами могут быть буквы, цифры и "_" в любой комбинации.

    Примечание

    В случае внешней программной памяти в имени директории допускаются и пробелы. Но это не действительно, если внешняя программная память настроена как глобальная память программ обработки деталей (GDIR).

  - Прочие специальные символы запрещены.
- Расширения директорий:
  - Расширения директорий должны состоять точно из трех букв/цифр.
  - Они отделяются от имени директории символом "_" (представление ЧПУ) или "." (представление EES).

  Примечание

  В пассивной файловой системе есть только расширения директорий _DIR и _WPD.

Примеры:

- Пассивная файловая система или память GDIR:
  - Представление ЧПУ: _N_WKS_DIR/_N_MYNCPROGS_WPD/...
  - Представление EES: WKS.DIR/MYPROGS.WPD/...
- Внешняя программная память:
  - /abc
  - /ab_c.def
  - /ab c1.def
  - /a b c .d11
  - /abc.def/ghi.klm

Имя файла

Для имен файлов действуют следующие правила:

- Только при адресации в представлении ЧПУ имена файлов начинаются с идентификатора домена "_N_".
- Оба следующих символа должны быть две буквы или знак подчеркивания с буквой после него.

  Примечание

  Если это условие выполнено, программа ЧПУ может быть вызвана из другой программы как подпрограмма путем указания имени программы . Если имя программы начинается с цифр, вызов подпрограммы возможен только через оператор CALL.

- Последующими символами могут быть буквы, цифры и "_" в любой комбинации.
- Расширение файла:
  - Расширение файла должно состоять точно из трех букв/цифр.

    Примечание

    Допустимые расширения файлов в пассивной файловой системе представлены в "Программная память в NCK (Страница 570)".

  - Оно отделяется от имени файла символом "_" (представление ЧПУ) или "." (представление EES).

Примеры:

- Пассивная файловая система или память GDIR:
  - Представление ЧПУ: _N_SUB1_SPF
  - Представление EES: SUB1.SPF
- Внешняя программная память:
  - Teil1
  - _Teil1
  - Teil_1.spf
  - Teil1.mpf

Имя подпрограммы в коде DIN

Для имен подпрограмм в коде DIN действуют следующие правила:

- Первый символ – буква "L".
- Следующие символы – цифры (не менее одной).
- Расширение файла:
  - Расширение файла должно состоять точно из трех букв.
  - Оно отделяется от имени файла символом "_" (представление ЧПУ) или "." (представление EES).

Примеры:

- L123
- L1_SPF (представление ЧПУ) или L1.SPF (представление EES)

Максимальная длина пути

Для указания имени диска и пути директории доступно не более 128 байт, длина имени файла должна составлять не более 31 байт. Следовательно, общая длина пути не должна превышать 159 байт.

4.4.1.4 Маршрут поиска при вызове подпрограммы

При вызовах подпрограмм без указания пути абсолютный путь определяется через выполнение фиксированного маршрута поиска.

При этом поиск в программной памяти осуществляется в следующей последовательности:

		Директория	Описание
1		текущая директория / *name*	Текущая директория – это директория, в которой осуществлен выбор программы. Это может быть: • директория детали или стандартная директория _N_MPF_DIR в памяти программ обработки деталей ЧПУ или в глобальной памяти программ обработки деталей или • любая директория внешней программной памяти
2		текущая директория / *name_SPF*	
3		текущая директория / *name_MPF*	
4	a	//NC:/_N_SPF_DIR / *name_SPF*	Директория подпрограммы в памяти программ обработки деталей ЧПУ
	B	//DEV2:/_N_SPF_DIR / name_SPF [1]	Директория подпрограммы в глобальной памяти программ обработки деталей **Указание:** Этот этап поиска отсутствует, если глобальная память программ обработки деталей не создана или выбор программы осуществлен в памяти программ обработки деталей ЧПУ.
5		Расширение маршрута поиска, запрограммированное с `CALLPATH` (см. "Расширить маршрут поиска для вызовов подпрограмм (CALLPATH) (Страница 553)"). **Примечание:** Этот этап поиска отсутствует, если команда `CALLPATH` не запрограммирована.	
6		/_N_CUS_DIR / *name_SPF*	Директория циклов пользователя
7		/_N_CMA_DIR / *name_SPF*	Директория циклов изготовителя
8		/_N_CST_DIR / *name_SPF*	Директория стандартных циклов

[1] //DEV2:" означает, например, диск, на котором создана глобальная память программ обработки деталей.

Для поиска действуют следующие правила:

- Маршрут поиска проходится для каждого отдельного вызова подпрограммы, т. е. не важно, где находится программа верхнего уровня.
- В зависимости от директории учитываются разные типы файлов.
- Поиск принципиально не выполняется в директориях нижнего уровня, т. е. во вложенных директориях.

4.4.1.5 Запрос пути и имени файла

Для запроса пути и имени файла программы ЧПУ предусмотрены следующие системные переменные, считываемые в программе обработки детали:

Системная переменная	Тип	Объяснение
$P_STACK	INT	Выводит программный уровень, на котором обрабатывается текущая программа ЧПУ.
$P_PATH[<n>]	STRING	Выводит путь программы ЧПУ, которая обрабатывается на программном уровне, выбранном через индекс поля <n>. Примеры: $P_PATH[0] выводит путь для главной программы, например "/_N_WKS_DIR/_N_WELLE_WPD/". $P_PATH[$P_STACK - 1] выводит путь вызывающей программы. Если путь относится к программе ЧПУ, которая заложена в пассивной файловой системе ЧПУ или в глобальной памяти программ обработки деталей (GDIR), то этот путь выводится в представлении ЧПУ. Если путь относится к программе ЧПУ, которая обрабатывается какой-либо другой внешней программной памятью, но не глобальной памятью программ обработки деталей, то $P_PATH выводит этот путь в представлении EES.
$P_PROG[<n>]	STRING	Выводит имя программы ЧПУ, которая обрабатывается на программном уровне, выбранном через индекс поля <n>. Если программа ЧПУ заложена в пассивной файловой системе ЧПУ или в глобальной памяти программ обработки деталей, имя программы выводится в представлении ЧПУ. Если программа ЧПУ обрабатывается с какого-либо другого внешнего диска, но не глобальной памятью программ обработки деталей, то $P_PROG выводит имя в представлении EES.
$P_PROGPATH	STRING	Выводит путь программы ЧПУ, которая обрабатывается в данный момент. Вызов $P_PROGPATH идентичен $P_PATH[$P_STACK].

Системная переменная	Тип	Объяснение	
$P_IS_EES_PATH[<n>]	BOOL	Запрос, соответствует ли выведенный через $P_PATH[<n>] путь или выведенное через $P_PROG[<n>] имя программы представлению ЧПУ или представлению EES.	
		= FALSE	$P_PATH[<n>] и $P_PROG[<n>] обеспечивают представление ЧПУ. Т. е. перед каждым идентификатором стоит префикс "_N_". Разделительным символом для расширения файла является "_". Примеры: • Путь в представлении ЧПУ: "/_N_WKS_DIR/_N_MYWPD_WPD/" • Имя программы в представлении ЧПУ: "_N_MYPROG_MPF" Путь в представлении ЧПУ может относиться как к пассивной файловой системе в ЧПУ, так и к глобальной памяти программ обработки деталей.
		=TRUE	$P_PATH[<n>] и $P_PROG[<n>] выводят представление EES. Т. е. перед идентификаторами префикса "_N_" нет. Разделительным символом для расширения файла является ".". Примеры: • Путь в представлении EES: "//DEV1:/WKS.DIR/MYWPD.WPD/" • Имя программы в представлении EES: "MYPROG.MPF"

<n>: Индекс <n> определяет программный уровень, с которого должна считываться информация о пути (диапазон значений: 0 ... 17)

Примечание

В режиме EES вне глобальной памяти программ обработки деталей (GDIR) системными переменными $P_PROG, $P_PATH, и $P_PROGPATH имена пути выводятся в представлении EES. Поэтому программы пользователя, которые используют и далее обрабатывают эти имена пути, нужно расширить для режима EES таким образом, чтобы они могли обрабатывать и имена пути в представлении EES.

4.4.2 Оперативная память (CHANDATA, COMPLETE, INITIAL)

Функция

Оперативная память содержит текущие системные данные и данные пользователя, с которыми работает СЧПУ (активная файловая система), например:

- Активные машинные данные
- Данные коррекции инструмента

- Смещения нулевой точки
- ...

Программы инициализации

Здесь речь идет о программах, с которыми предустанавливаются (инициализируются) данные оперативной памяти. Для этого могут использоваться следующие типы файлов:

Тип файла	Описание
name_TEA	Машинные данные
name_SEA	Установочные данные
name_TOA	Коррекции инструмента
name_UFR	Смещения нулевой точки/фрейм
name_INI	Файл инициализации
name_GUD	Глобальные данные пользователя
name_RPA	R-параметры

Области данных

Данные могут быть разбиты на различные области, в которых они должны действовать. Например, СЧПУ может иметь несколько каналов или обычно и несколько осей.

Существует:

Идентификатор	Области данных
ЧПУ	Специфические данные ЧПУ
CH<n>	Специфические данные канала (<n> указывает номер канала)
AX<n>	Специфические данные осей (n указывает номер оси станка)
TO	Данные инструмента
COMPLETE	Все данные

Создание программы инициализации на внешнем PC

С помощью идентификатора области данных и идентификатора типа данных можно определить области, которые при сохранении данных рассматриваются как блок:

_N_AX5_TEA_INI	Машинные данные для оси 5
_N_CH2_UFR_INI	Фреймы канала 2
_N_COMPLETE_TEA_INI	Все машинные данные

После ввода в эксплуатацию СЧПУ в оперативной памяти имеется блок данных, обеспечивающий правильную работу СЧПУ.

Принцип действий для многоканальных СЧПУ (CHANDATA)

CHANDATA (номер канала) для нескольких каналов разрешен только в файле _N_INITIAL_INI. Это файл ввода в эксплуатацию, с помощью которого инициализируются все данные СЧПУ.

Программный код	Комментарий
%_N_INITIAL_INI	
CHANDATA(1)	
	; согласование осей станка канал 1:
$MC_AXCONF_MACHAX_USED[0]=1	
$MC_AXCONF_MACHAX_USED[1]=2	
$MC_AXCONF_MACHAX_USED[2]=3	
CHANDATA(2)	
	; согласование осей станка канал 2:
$MC_AXCONF_MACHAX_USED[0]=4	
$MC_AXCONF_MACHAX_USED[1]=5	
CHANDATA(1)	
	: осевые машинные данные:
	; окно точного останова грубого:
$MA_STOP_LIMIT_COARSE[AX1]=0.2	; ось 1
$MA_STOP_LIMIT_COARSE[AX2]=0.2	; ось 2
	; окно точного останова точного:
$MA_STOP_LIMIT_FINE[AX1]=0.01	; ось 1
$MA_STOP_LIMIT_FINE[AX1]=0.01	; ось 2

ВНИМАНИЕ

Оператор CHANDATA

В программе обработки детали оператор CHANDATA может быть установлен только для канала, в котором выполняется программа ЧПУ. Это означает, что оператор может использоваться для защиты программ ЧПУ от выполнения в не предусмотренном для этого канале.

В случае ошибки выполнение программы отменяется.

Примечание

Файлы INI в списках заданий не содержат операторов CHANDATA.

Сохранение программ инициализации (COMPLETE, INITIAL)

Файлы оперативной памяти могут быть сохранены на внешний PC и снова загружены с него.

- Файлы сохраняются с помощью COMPLETE .
- С помощью INITIAL для всех областей создается файл INI (_N_INITIAL_INI).

Загрузка программ инициализации

ВНИМАНИЕ
Потеря данных Если загружается файл с именем "INITIAL_INI", то все данные, которые не обеспечиваются в файле, инициализируются со стандартными данными. Исключением являются только машинные данные. Т.е **установочные данные, данные инструмента, WO, значения GUD, ...** обеспечиваются со стандартными данными (обычно "НОЛЬ").

Для загрузки отдельных машинных данных, подходит, например, файл COMPLETE_TEA_INI. В этом файле СЧПУ ожидает только машинные данные. Тем самым другие области данных в этом случае не затрагиваются.

Загрузка программ инициализации

Программы INI могут выбираться и вызываться и как программы обработки детали, если они используют только данные одного канала. Таким образом также возможна инициализация управляемых программой данных.

4.5 Обработка файла

4.5.1 Записать файл (WRITE)

С помощью команды `WRITE` кадры/данные из программы ЧПУ могут быть записаны в конец файла (журнала), находящегося в пассивной файловой системе или во внешней программной памяти. Это может быть и обрабатываемая в настоящий момент программа.

Примечание

Записываемый посредством команды `WRITE` файл создается заново, если он не существует в программной памяти.

Условие

Актуальная установленная степень защиты должна быть равна или быть выше права WRITE файла. Если это не так, то доступ отклоняется с сообщением об ошибке (возвращаемое значение переменной ошибки = 13).

Синтаксис

```
DEF INT <ошибка>
...
```

```
WRITE(<ошибка>,"<имя файла>"/"<ExtG>","<кадр/данные>")
```

Значение

WRITE:	Команда добавления кадра или данных в конце указанного файла		
<ошибка>:	**Параметр 1:** Переменная для возврата значения ошибки		
	Тип:	INT	
	Значение:	0	нет ошибок
		1	путь не разрешен
		2	путь не найден
		3	файл не найден
		4	неправильный тип файла
		10	файл заполнен
		11	файл используется
		12	нет свободных ресурсов
		13	нет прав доступа
		14	отсутствие или неудача EXTOPEN для устройства вывода
		15	ошибка записи на внешнее устройство
		16	запрограммирован недействительный внешний путь
<имя файла>:	**Параметр 2:** Имя файла, в котором должен быть добавлен указанный кадр или указанные данные		
	Тип:	STRING	
	Перед собственно именем файла можно указывать абсолютный путь. Если путь не указывается, то поиск файла осуществляется в актуальной директории (=директория выбранной программы). Правила указания пути см. "Адресация файлов программной памяти (Страница 574)".		
<ExtG>:	Если данные с помощью функции "Process DataShare" должны быть выведены на внешнее устройство/файл, то необходимо указать вместо имени файла символический идентификатор для открываемого внешнего устройства/файла.		
	Тип:	STRING	
	Дополнительную информацию см. "Process DataShare - Вывод на внешнее устройство/файл (EXTOPEN, WRITE, EXTCLOSE) (Страница 1039)". **Примечание:** Идентификатор должен быть идентичен указанному в команде EXTOPEN идентификатору.		
<кадр/данные>:	**Параметр 3:** Кадр или данные, которые должны быть добавлены в указанный файл.		
	Тип:	STRING	

Примечание

При записи в пассивную файловую систему или во внешнюю программную память команда WRITE не явно добавляет символ "LF" (LINE FEED = новая строка) в конце строки вывода.

Для вывода на внешнее устройство/файл через функцию "Process DataShare" это поведение не действует. Если "LF" также должен быть выведен, то это должно быть указано явно в строке вывода.

→ См. пример 3: не явный/явный "LF"!

Граничные условия

- **Макс. размер файла (→ изготовитель станка!)**
  Макс. возможный размер файлов журнала в пассивной файловой системе устанавливается с машинными данными:
  MD11420 $MN_LEN_PROTOCOL_FILE
  Макс. размер файла действует для всех файлов, создаваемых с помощью команды WRITE в пассивной файловой системе. При превышении выводится сообщение об ошибке и кадр или данные не сохраняются. Если памяти достаточно, то может быть создан новый файл.

Примеры

Пример 1: Команда WRITE в пассивную файловую систему без абсолютного указания пути

Программный код	Комментарий
`N10 DEF INT ERROR`	; Определение переменных ошибок.
`N20 WRITE(ERROR,"PROT","PROTOKOLL VOM 7.2.97")`	; Записать текст "PROTOKOLL VOM 7.2.97" в файл _N_PROT_MPF.
`N30 IF ERROR`	;Обработка ошибок.
`N40 MSG ("Ошибка команды WRITE:" << ERROR)`	
`N50 M0`	
`N60 ENDIF`	
...	

Пример 2: Команда WRITE в пассивную файловую систему с абсолютным указанием пути

Программный код
...
`WRITE(ERROR,"/_N_WKS_DIR/_N_PROT_WPD/_N_PROT_MPF","PROTOKOLL VOM 7.2.97")`
...

Пример 3: Не явный/явный "LF"

a, запись в пассивную файловую систему с не явно созданным "LF"

Программный код

```
...
N110 DEF INT ERROR
N120 WRITE(ERROR,"/_N_MPF_DIR/_N_MYPROTFILE_MPF","MY_STRING")
N130 WRITE(ERROR,"/_N_MPF_DIR/_N_MYPROTFILE_MPF","MY_STRING")
N140 M30
```

Результат вывода:

MY_STRING

MY_STRING

b, запись во внешний файл без не явного созданного "LF"

Программный код

```
...
N200 DEF STRING[30] DEV_1
N210 DEF INT ERROR
N220 DEV_1="LOCAL_DRIVE/myprotfile.mpf"
N230 EXTOPEN(ERROR,DEV_1)
N240 WRITE(ERROR,DEV_1,"MY_STRING")
N250 WRITE(ERROR,DEV_1,"MY_STRING")
N260 EXTCLOSE(ERROR,DEV_1)
N270 M30
```

Результат вывода:

MY_STRINGMY_STRING

c, запись во внешний файл с явно запрограммированным "LF"

Для получения результата, идентичного a, запрограммировать следующее:

Программный код

```
...
N200 DEF STRING[30] DEV_1
N210 DEF INT ERROR
N220 DEV_1="LOCAL_DRIVE/myprotfile.mpf"
N230 EXTOPEN(ERROR,DEV_1)
N240 WRITE(ERROR,DEV_1,"MY_STRING'H0A'")
N250 WRITE(ERROR,DEV_1,"MY_STRING'H0A'")
N260 EXTCLOSE(ERROR,DEV_1)
N270 M30
```

Результат вывода:

MY_STRING

MY_STRING

4.5.2 Удалить файл (DELETE)

С помощью команды DELETE могут удаляться любые файлы, независимо от того, созданы ли они через команду WRITE или нет. Файлы, созданные при более высоком уровне доступа, также могут быть удалены с DELETE.

Синтаксис

```
DEF INT <ошибка>
DELETE(<ошибка>,"<имя файла>")
```

Значение

DELETE:	Команда удаления указанного файла		
<ошибка>:	Переменная для возврата значения ошибки		
	Тип.	INT	
	Значение:	0	нет ошибок
		1	путь не разрешен
		2	путь не найден
		3	файл не найден
		4	неправильный тип файла
		11	файл используется
		12	нет свободных ресурсов
		20	иная ошибка
<имя файла>:	Имя удаляемого файла		
	Тип:	STRING	
	Перед собственно именем файла можно указывать абсолютный путь. Если путь не указывается, то поиск файла осуществляется в актуальной директории (=директория выбранной программы). Правила указания пути см. "Адресация файлов программной памяти (Страница 574)".		

Пример

Программный код	Комментарий
N10 DEF INT ERROR	; Определение переменных ошибок.
N15 STOPRE	; Остановка предварительной обработки.
N20 DELETE(ERROR,"/_N_SPF_DIR/_N_TEST1_SPF")	; Удалить файл TEST1 в директории подпрограммы.
N30 IF ERROR	;Обработка ошибок.
N40 MSG("Ошибка команды DELETE:" <<ERROR)	

Программный код	Комментарий
N50 M0	
N60 ENDIF	

4.5.3 Чтение строк в файле (READ)

Команда READ считывает в указанном файле одну или несколько строк и помещает считанную информацию в поле типа STRING. Каждая считанная строка в этом поле занимает один элемент поля.

Условие

Актуальная установленная степень защиты должна быть равна или быть выше права READ файла. Если это не так, то доступ отклоняется с сообщением об ошибке (возвращаемое значение переменной ошибки = 13).

Синтаксис

```
DEF INT <ошибка>
DEF STRING[<длина строки>] <результат>[<n>,<m>]
READ(<ошибка>,"<имя файла>",<начальная строка>,<число строк>,<результат>)
```

Объяснение

READ:	Команда по чтению строк указанного файла и по сохранению этих строк в поле переменной.		
<ошибка>:	Переменная для возврата значения ошибки (параметры, вызываемые по ссылке)		
	Тип.	INT	
	Значение:	0	нет ошибок
		1	путь не разрешен
		2	путь не найден
		3	файл не найден
		4	неправильный тип файла
		11	файл используется
		13	прав доступа недостаточно
		21	Строка отсутствует (параметр <начальная строка> или <число строк> больше, чем число строк в указанном файле).
		22	Длина поля переменной результата (<результат>) слишком маленькая.
		23	Слишком большой диапазон строк (параметр <число строк> выбран таким большим, что считывается за конец файла).

<имя файла>:	Имя считываемого файла (параметр, вызываемый по значению)		
	Тип:	STRING	
	Перед собственно именем файла можно указывать абсолютный путь. Если путь не указывается, то поиск файла осуществляется в актуальной директории (=директория выбранной программы). Правила указания пути см. "Адресация файлов программной памяти (Страница 574)".		
<начальная строка>:	Начальная строка считываемой области файла (параметр, вызываемый по значению)		
	Тип:	INT	
	Значение:	0	Считывается указанное с помощью параметра <число строк> число строк перед концом файла.
		1 ... n	Номер первой считываемой строки.
<число строк>:	Число считываемых строк (параметр, вызываемый по значению)		
	Тип:	INT	
<результат>:	Переменная результата (параметр, вызываемый по ссылке) Поле переменной, в которое помещается считанный текст.		
	Тип:	STRING (макс. длина: 255)	
	Если в параметре <число строк> указано меньше строк, чем составляет размер поля `[<n>,<m>]` переменной результата, то оставшиеся элементы поля не изменяются. Конец строки через управляющий символ "LF" (Line Feed) или "CR LF" (Carrige Return Line Feed) **не** помещается в переменную результата. Считанные строки обрезаются, если строка длиннее, чем определенная длина строки. Сообщение об ошибке не появляется.		

Примечание

Загрузка двоичных файлов невозможна. Выводится ошибка "неправильный тип файла" (возвращаемое значение переменной ошибки = 4). Чтение следующих типов файлов невозможно: _BIN, _EXE, _OBJ, _LIB, _BOT, _TRC, _ACC, _CYC, _NCK.

Пример

Программный код	Комментарий
`N10 DEF INT ERROR`	; Определение переменных ошибок.
`N20 DEF STRING[255] RESULT[5]`	; Определение переменной результата.
`N30 READ(ERROR,"/_N_CST_DIR/_N_TESTFILE_MPF",1,5,RESULT)`	;Имя файла с идентификатором домена, расширением файла и указанием пути.
`N40 IF ERROR <>0`	;Обработка ошибок.
`N50 MSG("FEHLER"<<ERROR<<"BEI READ-BEFEHL")`	
`N60 M0`	
`N70 ENDIF`	
...	

4.5.4 Проверка наличия файла (ISFILE)

Командой `ISFILE` можно проверить, существует ли файл в программной памяти.

Синтаксис

<Результат>=`ISFILE("`<Имя файла>")

Объяснение

`ISFILE`:	Команда для проверки наличия файла		
<Имя файла>:	Имя файла, наличие которого нужно проверить.		
	Тип:	STRING	
	Перед собственно именем файла можно указывать абсолютный путь. Если путь не указывается, то поиск файла осуществляется в текущей директории (=директория выбранной программы). Правила указания пути см. "Адресация файлов программной памяти (Страница 574)".		
<Результат>:	Переменная результата для записи результата проверки		
	Тип.	BOOL	
	Значение:	TRUE	Файл имеется
		FALSE	Файл отсутствует

Примеры

Пример 1

Программный код	Комментарий
`N10 DEF BOOL RESULT`	; Определение переменной результата.
`N20 RESULT=ISFILE("TESTFILE")`	
`N30 IF(RESULT==FALSE)`	
`N40 MSG("`▸ФАЙЛ ОТСУТСТВУЕТ")	
`N50 M0`	
`N60 ENDIF`	
...	

Пример 2

Программный код	Комментарий
`N10 DEF BOOL RESULT`	; Определение переменной результата.
`N20 RESULT=ISFILE("TESTFILE")`	
`N30 IF(NOT ISFILE("TESTFILE"))`	
`N40 MSG("`ФАЙЛ ОТСУТСТВУЕТ")	
`N50 M0`	
`N60 ENDIF`	
...	

4.5.5 Выгрузка файловой информации (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO)

Через команды FILEDATE, FILETIME, FILESIZE, FILESTAT и FILEINFO можно выгрузить определенную файловую информацию, например, дату / время последнего обращения по записи, текущий размер файла, состояние файла или сумму этих информаций.

Условие

Текущая установленная степень защиты должна быть равна или быть выше права просмотра директории верхнего уровня. Если это не так, то доступ отклоняется с сообщением об ошибке (возвращаемое значение переменной ошибки = 13).

Синтаксис

FILE.... (<Ошибка>,"<Имя файла>",<Результат>)

Объяснение

FILEDATE:	Выводит **дату** последнего обращения к файлу по записи		
FILETIME:	Выводит **время** последнего обращения к файлу по записи		
FILESIZE:	Выводит **текущий размер** файла		
FILESTAT:	Выводит **состояние** файла касательно следующих **прав**: • Чтение (r: read) • Запись (w: write) • Исполнение (x: execute) • Индикация (s: show) • Удаление (d: delete) **Примечание:** Эти уровни защиты являются специальными свойствами пассивной файловой системы. Поэтому при доступе к внешней программной памяти FILESTAT предоставляет только права доступа по умолчанию (77777).		
FILEINFO:	Выводит для файла **суммарную информацию**, которая может быть выгружена через FILEDATE, FILETIME, FILESIZE и FILESTAT.		
<ошибка>:	Переменная для возврата значения ошибки (параметры, вызываемые по ссылке)		
	Тип.	VAR INT	
	Значение:	0	нет ошибок
		1	путь не разрешен
		2	путь не найден
		3	файл не найден
		4	неправильный тип файла
		13	прав доступа недостаточно
		22	Длина строки переменной результата (<результат>) слишком маленькая.

<Имя файла>:	Имя файла, из которого должна быть выгружена файловая информация			
	Тип:	CHAR[160]		
	Перед собственно именем файла можно указывать абсолютный путь. Если путь не указывается, то поиск файла осуществляется в текущей директории (=директория выбранной программы). Правила указания пути см. "Адресация файлов программной памяти (Страница 574)".			
<Результат>:	Переменная результата (параметр Call-By-Reference) Переменная, в которую помещается запрошенная файловая информация.			
	Тип:	VAR CHAR[8]	при	FILEDATE формат: "dd.mm.yy"
		VAR CHAR[8]	при	FILETIME формат: "hh:mm.ss"
		VAR INT	при	FILESIZE Размер файлы выводится в байтах.
		VAR CHAR[5]	при	FILESTAT формат: "rwxsd" (r: **r**ead, w: **w**rite, x: e**x**ecute, s: **s**how, d: **d**elete)
		VAR CHAR[32]	при	FILEINFO Формат: "rwxsd nnnnnnnn dd.mm.yy hh:mm:ss"

Пример

Программный код	Комментарий
`N10 DEF INT ERROR`	; Определение переменных ошибок.
`N20 STRING[32] RESULT`	; Определение переменной результата.
`N30 FILEINFO(ERROR,"/_N_MPF_DIR/ _N_TESTFILE_MPF",RESULT)`	; Имя файла с идентификатором домена, расширением файла и указанием пути.
`N40 IF ERROR <> 0`	; Обработка ошибок
`N50 MSG("ОШИБКА"<<ERROR<<"В СЛУЧАЕ КОМАНДЫ FILEINFO")`	
`N60 M0`	
`N70 ENDIF`	
...	

Пример мог бы вывести в переменной результата RESULT, например, следующий результат:

"77777 12345678 26.05.00 13:51:30"

4.6 Защищенные области

4.6.1 Определение защищенных областей (CPROTDEF, NPROTDEF)

Защищенные области, которые должны защищать элементы станка от столкновений, определены соответственно в блоках в программе обработки детали. Они включают следующие элементы:

1. Определение рабочей плоскости
   Перед собственно началом определения защищенной области необходимо выбрать рабочую плоскость, к которой должно относиться описание контура защищенной области.
2. Начало определения
   В зависимости от команды ЧПУ защищенная область создается либо спец. для канала, либо спец. для станка.
3. Описание контура защищенной области
   Контур защищенной области описывается посредством движений перемещения. Они не выполняются и не связаны с предшествующими или последующими геометрическими описаниями. Они определяют только защищенную область.
4. Конец определения

Синтаксис

```
DEF INT <Var>
G17/G18/G19
CPROTDEF/NPROTDEF(<n>,<t>,<AppLim>,<AppPlus>,<AppMinus>)
G0/G1/... X/Y/Z...
...
EXECUTE(<Var>)
```

Объяснение

`DEF INT <Var>`:	Определение локальной вспомогательной переменной (тип данных: INTEGER)
`<Var>`:	Имя вспомогательной переменной
`G17/G18/G19`:	Рабочая плоскость **Указание:** Рабочая плоскость не может изменяться до конца определения. Программирование аппликаты между началом и концом определения не допускается.
`CPROTDEF()`:	Предопределенная процедура определения спец. для **канала** защищенной области
`NPROTDEF()`:	Предопределенная процедура определения спец. для **станка** защищенной области

<n>:	Номер определенной защищенной области		
	Тип данных:	INT	
<t>:	Тип защищенной области		
	Тип данных:	BOOL	
	Значение:	TRUE	Относящаяся к **инструменту** защищенная область
		FALSE	Относящаяся к **детали** защищенная область
<AppLim>:	Тип ограничения в 3-м измерении		
	Тип данных:	INT	
	Значение:	0	нет ограничения
		1	ограничение в плюсовом направлении
		2	ограничение в минусовом направлении
		3	Ограничение в плюсовом и минусовом направлении
<AppPlus>:	Значение ограничения в плюсовом направлении 3-й размерности		
	Тип данных:	REAL	
<AppMinus>:	Значение ограничения в минусовом направлении 3-й размерности		
	Тип данных:	REAL	

`G0/G1/... X/Y/Z... ...:`	Контур защищенной области описывается макс. 11 движениями перемещения в выбранной рабочей плоскости. При этом первое движение перемещения — это движение к контуру. Последняя точка описания контура всегда должна совпадать с первой точкой описания контура. Защищенной областью считается область слева от контура: • внутренняя защищенная область Контур для внутренней защищенной области следует описывать против часовой стрелки. • Наружная защищенная область (допускается только для относящихся к детали защищенных областей) Контур для наружной защищенной области следует описывать по часовой стрелке. Допустимы следующие элементы контура: • `G0`, `G1` для прямых элементов контура • `G2` для сегментов окружности по часовой стрелке Допустимо только для относящихся к детали защищенных областей! Не допустимо для относящихся к инструменту защищенных областей, т. к. они могут быть только выпуклыми. • `G3` для круговых сегментов против часовой стрелки **Указание:** Защищенная область не может быть описана полным кругом. Полный круг должен быть разбит на две делительные окружности. **Указание:** Последовательность `G2` → `G3` или `G3` → `G2` не допустима! Между двумя круговыми кадрами должен быть вставлен короткий кадр `G1`.
`EXECUTE(<Var>):`	Предопределенная процедура, отмечающая конец определения `EXECUTE` возвращает к обычной обработке программы.

Пример

Пример см. в "Активация/деактивация защищенных областей (CPROT, NPROT) (Страница 599)".

Дополнительная информация

Спец. для станка защищенные области

Относящаяся к станку защищенная область или ее контур определяется посредством гео-осей, т. е. относительно базовой кинематической системы (BKS) канала. Чтобы был возможен правильный контроль защищенных областей во всех каналах, в которых активна спец. для станка защищенная область, базовые кинематические системы (BKS) всех участвующих каналов должны быть идентичны:

- положение начала координат относительно нулевой точки станка
- Ориентация осей координат

Исходная точка описания контура

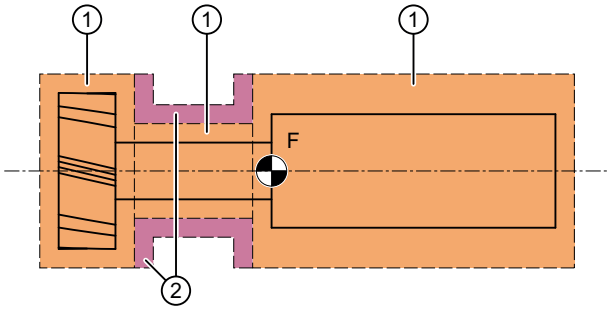
- Относящиеся к инструменту защищенные области
  Координаты для относящихся к **инструменту** защищенных областей указываются абсолютно, относительно **опорной точки инструментального суппорта F**.
- Относящиеся к детали защищенные области
  Координаты для относящихся к **детали** защищенных областей указываются абсолютно, относительно **нулевой точки базовой кинематической системы (BKS)**.

Вращательно-симметричные защищенные области

Для вращательно-симметричных защищенных областей (например, патрон шпинделя) необходимо описание всего контура, а не только до центра вращения.

Относящиеся к инструменту защищенные области

Относящиеся к инструменту защищенные области всегда должны быть выпуклыми. Если необходима вогнутая защищенная область, то необходимо разбить его на несколько выпуклых защищенных областей.



① Выпуклые защищенные области
② Вогнутые защищенные области (**недопустимы!**)
F Исходная точка инструментального суппорта

Граничные условия

При определении защищенной области следующие функции не могут быть активны и запрещены к использованию:

- коррекция радиуса инструмента (коррекция радиуса фрезы, коррекция радиуса резца)
- трансформация
- движение к референтной точке (`G74`)
- движение к фиксированной точке (`G75`)
- время ожидания (`G4`)
- останов предварительной обработки кадра (`STOPRE`)
- конец программы (`M17`, `M30`)
- M-функции: `M0`, `M1`, `M2`

4.6.2 Активация/деактивация защищенных областей (CPROT, NPROT)

Предварительно определенные в программе обработки детали защищенные области можно в любой момент активировать либо предварительно активировать для последующей активации через программу электроавтоматики. Активные защищенные области можно деактивировать в любой момент.

При активации либо предварительной активации возможно относительное смещение исходной точки защищенной области.

Примечание

Защищенная область учитывается только после реферирования всех гео-осей канала, в котором она активирована.

Примечание

Контроль защищенных областей

Если ни одна относящаяся к инструменту защищенная область не активна, то траектория инструмента проверяется по отношению к относящимся к детали защищенным областям.

Если ни одна относящаяся к детали защищенная область не активна, контроль защищенных областей не выполняется.

Синтаксис

```
CPROT(<n>,<Status>,<XMov>,<YMov>,<ZMov>)
NPROT(<n>,<Status>,<XMov>,<YMov>,<ZMov>)
```

Объяснение

CPROT:	Предопределенная процедура активации спец. для **канала** защищенной области		
NPROT:	Предопределенная процедура активации спец. для **станка** защищенной области		
<n>:	Номер защищенной области		
	Тип данных:	INT	
<состояние>:	С помощью этого параметра устанавливается спец. для канала статус активации.		
	Тип данных:	INT	
	Значение:	0	Деактивировать защищенную область
		1	предварительная активация защищенной области
		2	Активировать защищенную область
		3	Предварительно активировать защищенную область с условным остановом

<XMov>,<YMov>,<ZMov>:	Значения аддитивного смещения в направлении X/Y/Z Смещение может выполняться в 1, 2 или 3 размерностях. Значения смещения относятся к: • нулевой точке станка при относящейся к **детали** защищенной области • исходной точке инструментального суппорта F для спец. для **инструмента** защищенной области.	
	Тип данных:	REAL

Пример

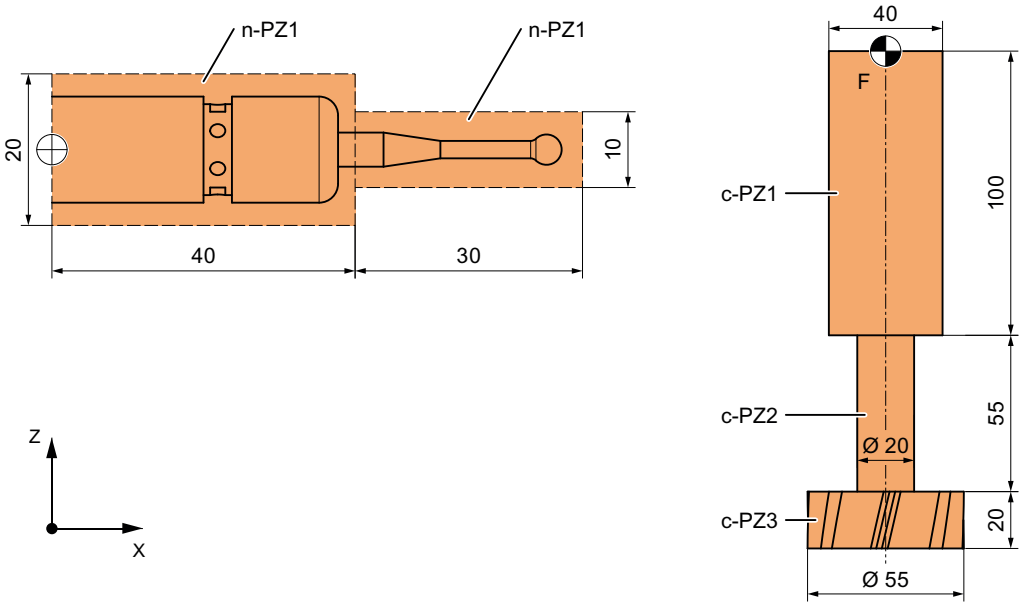
Для фрезерного станка необходимо контролировать возможное столкновение фрезы с измерительным щупом. Положение измерительного щупа при активации должно быть указано через смещение.

Для этого определяются следующие защищенные области:

- Соответственно по одной спец. для станка и относящейся к детали защищенной области для держателя измерительного щупа (n-PZ1) и самого измерительного щупа (n-PZ2).
- Соответственно по одной спец. для станка и относящейся к детали защищенной области для держателя фрезы (c-PZ1), хвостовика фрезы (c-PZ2) и самой фрезы (c-PZ3).

Ориентация всех защищенных областей лежит в направлении Z.

Положение исходной точки измерительного щупа при активации должно находиться на X = -120, Y = 60 и Z = 80.



① Исходная точка для защищенной области измерительного щупа

F Исходная точка инструментального суппорта

Программный код	**Комментарий**
DEF INT PROTZONE	; Определение вспомогательной переменной
G17	; Рабочая плоскость XY
; Определение защищенных областей:	
NPROTDEF(1,FALSE,3,10,-10)	; Защищенная область n-PZ1
G01 X0 Y-10	
X40	
Y10	
X0	
Y-10	
EXECUTE(PROTZONE)	
NPROTDEF(2,FALSE,3,5,-5)	; Защищенная область n-PZ2
G01 X40 Y-5	
X70	
Y5	
X40	
Y-5	
EXECUTE(PROTZONE)	
CPROTDEF(1,TRUE,3,0,-100)	; Защищенная область c-PZ1
G01 X-20 Y-20	
X20	
Y20	
X-20	
Y-20	
EXECUTE(PROTZONE)	
CPROTDEF(2,TRUE,3,-100,-150)	; Защищенная область c-PZ2
G01 X0 Y-10	
G03 X0 Y10 J10	
X0 Y-10 J-10	
EXECUTE(PROTZONE)	
CPROTDEF(3,TRUE,3,-150,-170)	; Защищенная область c-PZ3
G01 X0 Y-27.5	
G03 X0 Y27.5 J27.5	
X0 Y27.5 J-27.5	
EXECUTE(PROTZONE)	
; Активация защищенных областей:	
NPROT(1,2,-120,60,80)	; Активация защищенной области n-PZ1 со смещением
NPROT(2,2,-120,60,80)	; Активация защищенной области n-PZ2 со смещением
CPROT(1,2,0,0,0)	; Активация защищенной области c-PZ1
CPROT(2,2,0,0,0)	; Активация защищенной области c-PZ2
CPROT(3,2,0,0,0)	; Активация защищенной области c-PZ3

Дополнительная информация

Статус активации после запуска СЧПУ

Защищенная область может быть активна сразу же после запуска СЧПУ и реферирования осей. Это случай, когда для защищенной области следующая системная переменная установлена на TRUE:

- $SN_PA_ACTIV_IMMED[<n>] (для спец. для станка защищенной области) или
- $SC_PA_ACTIV_IMMED[<n>] (для спец. для канала защищенной области)
  Индекс "<n>" соответствует номеру защищенной области: 0 = 1 Защищенная область

Защищенная область активируется статусом = 2 и без смещения.

Многократная активация защищенной области

Одна спец. для станка защищенная область может одновременно действовать в нескольких каналах (например, защищенная область пиноли при двух противолежащих салазках). Контроль защищенных областей выполняется только в том случае, если все гео-оси реферированы.

Защищенная область в одном канале не может быть одновременно активирована с различными смещениями.

Контроль защищенной области при активной коррекции радиуса инструмента

При активной коррекции радиуса инструмента работоспособный контроль защищенной области возможен только в том случае, если плоскость коррекции радиуса инструмента идентична плоскости определения защищенной области.

4.6.3 Проверка нарушения защищенной области, ограничения рабочего поля и программных конечных выключателей (CALCPOSI)

Функция

Функция CALCPOSI проверяет в системе координат детали (WCS), можно ли перемещать **гео-оси** из стартовой позиции по заданному пути перемещения, не нарушая активных ограничений. Если путь перемещения из-за ограничений не может быть пройден полностью, то возвращается положительное значение состояния в десятичной кодировке и макс. возможный путь перемещения.

Определение

```
INT CALCPOSI(VAR REAL[3] <Start>, VAR REAL[3] <Dist>, VAR REAL[5]
<Limit>, VAR REAL[3] <MaxDist>, BOOL <MeasSys>, INT <TestLim>)
```

Синтаксис

```
<состояние> = CALCPOSI(VAR <Start>, VAR <Dist>, VAR <Limit>, VAR
<MaxDist>, <MeasSys>, <TestLim>)
```

Объяснение

`CALCPOSI(...)`:	Предопределенная функция для теста на нарушение ограничений в отношении гео-осей		
	Остановка предварительной обработки:		нет
	Автономно в кадре:		да
<состояние>: (часть 1)	Возвращаемое значение функции. Отрицательные значения показывают состояния ошибки.		
	Тип данных:		INT
	Диапазон значений:		-8 ≤ x ≤ 100000
	Значение:	0	Путь перемещения может быть пройден полностью
		-1	В <Limit>, по крайней мере, один компонент — отрицательный.
		-2	Ошибка при описании трансформации. Пример: Путь перемещения проходит через сингулярность, так что позиции осей не определены.
		-3	Указанный путь перемещения <Dist> и максимально возможный путь перемещения <MaxDist> находятся в линейной зависимости. **Указание** Возможно только в сочетании с <TestLim>, бит 4 == 1.
		-4	Проекция направления перемещения из <Dist> на ограничивающую поверхность — это нулевой вектор, или направление перемещения располагается вертикально на нарушенной ограничивающей поверхности. **Указание** Возможно только в сочетании с <TestLim>, бит 5 == 1.
		-5	В <TestLim> бит 4 == 1 И бит 5 == 1
		-6	По крайней мере, одна ось станка, которую необходимо рассмотреть для проверки границ перемещения, не реферирована.
		-7	Функция предотвращения столкновений: Недействительное определение кинематической цепочки или защищенных областей.
		-8	Функция предотвращения столкновений: Функция не может быть выполнена из-за недостаточности памяти.

<состояние>: (часть 2)	Первая позиция **Указание** Если одновременно нарушено несколько границ, то сигнал поступает по той границе, которая вызывает наибольшее ограничение заданного пути перемещения.		
	Значение:	1	Программные конечные выключатели ограничивают путь перемещения
		2	Ограничение рабочего поля ограничивает путь перемещения
		3	Защищенные области ограничивают путь перемещения
		4	Функция предотвращения столкновений: Защищенные области ограничивают путь перемещения
	Вторая позиция		
	Значение:	1x	начальное значение нарушает границу
		2x	Заданная прямая нарушает границу. Это значение возвращается и тогда, когда сама конечная точка не нарушает границы, но на пути от стартовой к конечной точке произошло бы нарушение предельного значения (например, прохождение через защищенную область, изогнутые программные конечные выключатели в WCS при нелинейных трансформациях, например, Transmit).
<состояние>: (часть 3)	Третья позиция		
	Значение:	1xx	И первая позиция == 1 или 2: Положительное предельное значение нарушено.
			И первая позиция == 3 [1]: Нарушена спец. для ЧПУ защищенная область.
		2xx	И первая позиция == 1 или 2: Отрицательное предельное значение нарушено.
			И первая позиция == 3 [1]: Нарушена спец. для канала защищенная область.
<состояние>: (часть 4)	Четвертая позиция		
	Значение:	1xxx	И первая позиция == 1 или 2: Коэффициент, на который умножается номер оси, нарушающей границу. Счет осей начинается с 1. Отношение: • Программные конечные выключатели: Оси станка • Ограничение рабочего поля: Геометрические оси И первая позиция == 3 [1]: Коэффициент, на который умножается номер нарушенной защищенной области.

<состояние>: (часть 5)	Шестая позиция		
	Значение:	0xxxxx	Шестая позиция == 0: <Dist> не изменяется
		1xxxxx	В <Dist> возвращается вектор направления, определяющий дальнейшее направление движения на ограничивающей поверхности. Возможно только при следующих граничных условиях: • Программные конечные выключатели или ограничение рабочего поля нарушены (не в стартовой точке) • Трансформация **не** активна • <TestID>, бит 4 или бит 5 == 1
`<Start>`:	Ссылка на вектор со стартовыми позициями: • <Start> [0]: 1-й гео-оси • <Start> [1]: 2-й гео-оси • <Start> [2]: 3-й гео-оси		
	Тип параметра:		Вход
	Тип данных:		VAR REAL[3]
	Диапазон значений:		-макс. REAL-значение ≤ x[<n>] ≤ +макс. REAL-значение
`<Dist>`:	Ссылка на вектор.		
	Вход: инкрементальная величина пути перемещения • <Dist> [0]: 1-й гео-оси • <Dist> [1]: 2-й гео-оси • <Dist> [2]: 3-й гео-оси		
	Выход (только при установленной шестой позиции в <состояние>):		
		В качестве выходного значения <Dist> содержит единичный вектор **v**, определяющий дальнейшее направление перемещения в WCS.	
		Случай 1: Формирование вектора **v** для <TestID>, бит 4 == 1 Входные векторы <Dist> и <MaxDist> образуют плоскость движения. Эта плоскость пересекается с нарушенной ограничивающей поверхностью. Прямая пересечения обеих плоскостей определяет направление вектора **v**. При этом ориентация (знак) выбирается таким образом, чтобы угол между входным вектором <MaxDist> и **v** не превышал 90 градусов.	
		Случай 2: Формирование вектора **v** для <TestID>, бит 5 == 1 Вектор **v** — единичный вектор в направлении проекции содержащегося в <Dist> вектора перемещения на ограничивающую поверхность. Если проекция вектора перемещения на ограничивающую поверхность представляет собой нулевой вектор, то возвращается ошибка.	
	Тип параметра:		Вход/выход
	Тип данных:		VAR REAL[3]
	Диапазон значений:		-макс. REAL-значение ≤ x[<n>] ≤ +макс. REAL-значение

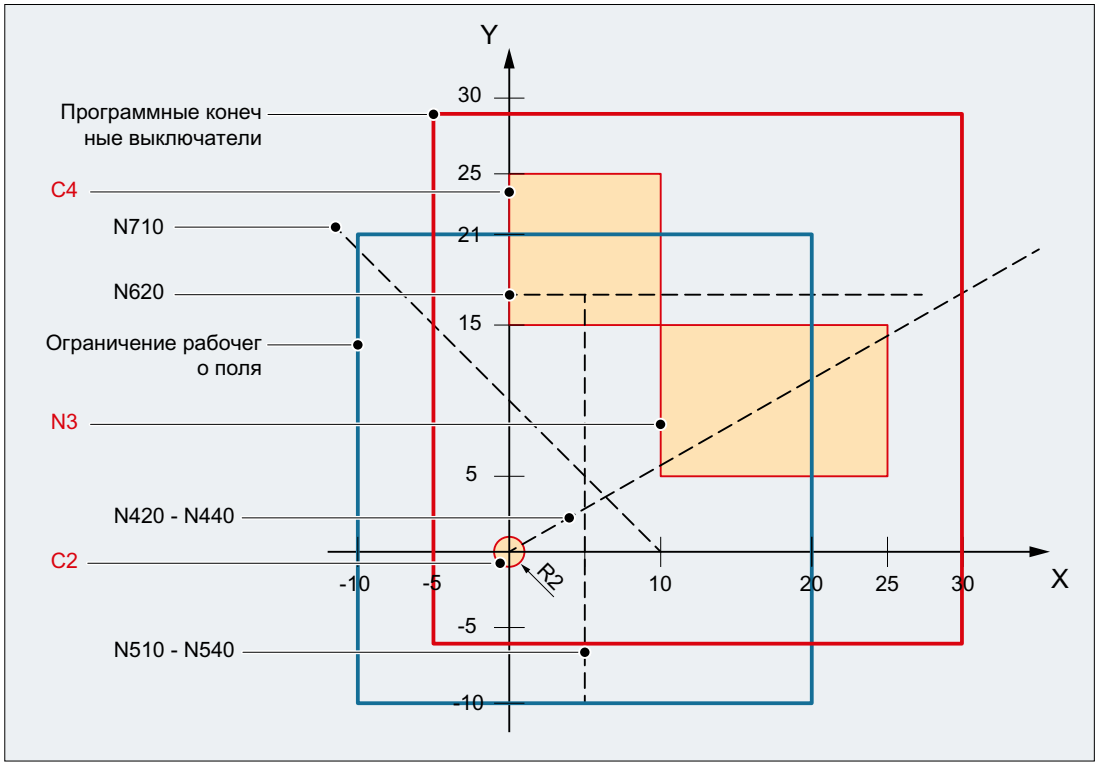
<Limit>:	Ссылка на поле длиной 5. • <Limit> [0 - 2]: Минимальные отступы гео-осей от ограничений: – <Limit> [0]: 1-й гео-оси – <Limit> [1]: 2-й гео-оси – <Limit> [2]: 3-й гео-оси Все минимальные отступы соблюдены при: – Ограничение рабочего поля: без ограничений – Программные конечные выключатели: Если нет активной трансформации или если активна трансформация, при которой возможно однозначное соотнесение гео-осей с линейными осями станка, например, 5-осевые трансформации. • <Limit> [3]: Содержит минимальный отступ линейных осей станка, согласование которых с гео-осью невозможно, например, из-за нелинейной трансформации. Это значение используется также как предельное значение при контроле обычных защищенных областей и защищенных областей предотвращения столкновений. • <Limit> [4]: Содержит минимальный отступ круговых осей станка, согласование которых с гео-осью невозможно, например, из-за нелинейной трансформации. **Указание** Это значение действует только при контроле программных конечных выключателей специальных трансформаций.	
	Тип параметра:	Вход
	Тип данных:	VAR REAL[5]
	Диапазон значений:	-макс. REAL-значение ≤ x[n] ≤ макс. REAL-значение
<MaxDist>:	Ссылка на вектор с инкрементальным значением пути перемещения, при котором заданное минимальное расстояние границы оси не нарушает ни одна из задействованных осей: • <Dist> [0]: 1-й гео-оси • <Dist> [1]: 2-й гео-оси • <Dist> [2]: 3-й гео-оси Если путь перемещения не ограничен, то содержание этого возвращаемого параметра идентично содержанию <Dist>.	
	Для <TestID>, бит 4 == 1: <Dist> и <MaxDist> В качестве входных значений <MaxDist> и <Dist> должны содержать векторы, образующие плоскость движения. Оба вектора не должны иметь линейной зависимости друг от друга. Величина <MaxDist> произвольная. Для расчета направления движения см. описание <Dist>.	
	Тип параметра:	Выход
	Тип данных:	VAR REAL[3]
	Диапазон значений:	-макс. REAL-значение ≤ x[<n>] ≤ +макс. REAL-значение

<MeasSys>:	Система единиц (дюймовая / метрическая) для позиционных и линейных данных (**опция**)		
	Тип данных:		BOOL
	Значение:	FALSE (по умолчанию)	Система единиц согласно текущей активной G-команде из G-группы 13 (G70, G71, G700, G710). **Указание** При активной G70 и метрической основной системе или активной G71 и дюймовой основной системе системные переменные $AA_IW и $AA_MW выводятся в основной системе и при использовании для CALCPOSI при необходимости должны быть пересчитаны.
		TRUE	Система единиц согласно установленной основной системе: MD52806 $MN_ISO_SCALING_SYSTEM
<TestLim>:	Биткодированный выбор контролируемых ограничений (**опция**)		
	Тип данных:		INT
	Значение по умолчанию:		Бит 0, 1, 2, 3, 6, 7 == 1 (207)
	Бит	**Дес.**	**Объяснение**
	0	1	Программные конечные выключатели
	1	2	Ограничение рабочего поля
	2	4	Активированные обычные защищенные области
	3	8	Предактивированные обычные защищенные области
	4	16	При нарушении программных конечных выключателей или ограничений рабочего поля в <Dist> возвращать направление перемещения согласно **случаю 1** (см. выше).
	5	32	При нарушении программных конечных выключателей или ограничений рабочего поля в <Dist> возвращать направление перемещения согласно **случаю 2** (см. выше).
	6	64	Активированные защищенные области недопущения столкновений
	7	128	Предактивированные защищенные области предотвращения столкновений
	8	256	Пары активированных и предактивированных защищенных областей предотвращения столкновений

[1] Если нарушено несколько защищенных областей, то сигнализируется та, которая вызывает наибольшее ограничение заданного пути перемещения.

Пример

Ограничения



В примере представлены действующие программные конечные выключатели и ограничения рабочего поля в плоскости X-Y и три следующих защищенных области:

- C2: Относящаяся к инструменту спец. для канала защищенная область, активная, круговая, радиус = 2 мм
- C4: Относящаяся к детали спец. для канала защищенная область, предварительно активирована, квадратная, длина стороны = 10 мм
- N3: Относящаяся к станку защищенная область, активная, прямоугольная, длина стороны = 10 мм x 15 мм

Программа ЧПУ

В программе ЧПУ определяются защищенные области и ограничения рабочего поля. Затем вызывается функция `CALCPOSI()` с различным параметрированием.

Программный код

```
N10 DEF REAL _START[3]
N20 DEF REAL _DIST[3]
N30 DEF REAL _LIMIT[5]
N40 DEF REAL _MAXDIST[3]
N50 DEF INT _PA
N60 DEF INT _STATUS
```

Программный код

```
; Относящаяся к инструменту защищенная область C2
N70 CPROTDEF(2, TRUE, 0)
N80 G17 G1 X-2 Y0
N90 G3 I2 X2
N100 I-2 X-2
N110 EXECUTE(_PA)
; Относящаяся к детали защищенная область C4
N120 CPROTDEF(4, FALSE, 0)
N130 G17 G1 X0 Y15
N140 X10
N150 Y25
N160 X0
N170 Y15
N180 EXECUTE(_PA)
; Спец. для станка защищенная область N3
N190 NPROTDEF(3, FALSE, 0)
N200 G17 G1 X10 Y5
N210 X25
N220 Y15
N230 X10
N240 Y5
N250 EXECUTE(_PA)
; Активировать или предварительно активировать защищенные области
N260 CPROT(2, 2, 0, 0, 0)
N270 CPROT(4, 1, 0, 0, 0)
N280 NPROT(3, 2, 0, 0, 0)
; Определить ограничения рабочего поля
N290 G25 XX=-10 YY=-10
N300 G26 XX=20 YY=21
N310 _START[0] = 0.
N320 _START[1] = 0.
N330 _START[2] = 0.
N340 _DIST[0] = 35.
N350 _DIST[1] = 20.
N360 _DIST[2] =  0.
N370 _LIMIT[0] = 0.
N380 _LIMIT[1] = 0.
N390 _LIMIT[2] = 0.
N400 _LIMIT[3] = 0.
N410 _LIMIT[4] = 0.
N420 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST)
N430 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,3)
N440 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,1)
N450 _START[0] =  5.
N460 _START[1] = 17.
N470 _START[2] =  0.
N480 _DIST[0] =  0.
N490 _DIST[1] =-27.
N500 _DIST[2] =  0.
N510 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,14)
N520 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,, 6)
N530 _LIMIT[1] = 2.
```

Программный код

```
N540 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,, 6)
N550 _START[0] = 27.
N560 _START[1] = 17.1
N570 _START[2] =  0.
N580 _DIST[0] =-27.
N590 _DIST[1] =  0.
N600 _DIST[2] =  0.
N610 _LIMIT[3] = 2.
N620 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,12)
N630 _START[0] = 0.
N640 _START[1] = 0.
N650 _START[2] = 0.
N660 _DIST[0] =  0.
N670 _DIST[1] = 30.
N680 _DIST[2] =  0.
N690 TRANS X10
N700 AROT Z45
N710 _STATUS = CALCPOSI(_START,_DIST, _LIMIT, _MAXDIST)
; Снова удалить фреймы из N690 и N700
N720 TRANS
N730 _START[0] =  0.
N740 _START[1] = 10.
N750 _START[2] =  0.
; Векторы _DIST и _MAXDIST определяют плоскость движения
N760 _DIST[0] = 30.
N770 _DIST[1] = 30.
N780 _DIST[2] =  0.
N790 _MAXDIST[0] = 1.
N800 _MAXDIST[1] = 0.
N810 _MAXDIST[2] = 1.
N820 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,17)
N830 M30
```

Результаты CALCPOSI()

N...	<состояние>	<MaxDist>[0] ≙ X	<MaxDist>[1] ≙ Y	Примечания
420	3123	8.040	4.594	N3 нарушается.
430	1122	20.000	11.429	Контроль защищенных областей отсутствует, ограничение рабочего поля нарушается.
440	1121	30.000	17.143	Активен только контроль программных конечных выключателей.
510	4213	0.000	0.000	Нарушение стартовой точки C4
520	0000	0.000	-27.000	Предварительно активированная C4 не контролируется. Заданный путь может быть пройден полностью.
540	2222	0.000	-25.000	Из-за _LIMIT[1] = 2 путь перемещения сокращается ограничением рабочего поля.

N...	<состояние>	<MaxDist>[0] ≙ X	<MaxDist>[1] ≙ Y	Примечания
620	4223	-13.000	0.000	Интервал до C4 из-за C2 и _LIMIT[3] в общей сложности 4 мм. Интервал C2 → N3 в 0,1 мм не вызывает ограничения пути перемещения.
710	1221	0.000	21.213	Фрейм со смещением и вращением активен. Допустимый путь перемещения в _DIST действителен в смещенной и повернутой WCS.
820	102121	18.000	18.000	Программный конечный выключатель оси Y нарушается. При <_TESTLIM> = 17 запрашивается вычисление направления дальнейшего движения. Это направление указано в <_DIST> (0.707, 0.0, 0.707). Оно действительно, поскольку в <_STATUS> установлена шестая позиция.

Дополнительная информация

Состояние оси "Реферирована"

Все рассматриваемые `CALCPOSI()` оси станка должны быть реферированы.

Характеристики перемещения относительно окружности

Характеристики перемещения относительно окружности **всегда** интерпретируются как характеристики радиуса. Это следует учитывать, в частности, для поперечных осей с активированным программированием диаметра (`DIAMON` / `DIAM90`).

Сокращение пути перемещения

Если указанный путь перемещения оси ограничивается, то в возвращаемом значении `<MaxDist>` пропорционально сокращаются и пути перемещения других осей. Тем самым результирующая конечная точка и далее расположена на заданной траектории.

Круговые оси

Круговые оси контролируются только в том случае, если они не являются осями по модулю.

Для одной или нескольких участвующих осей допустимо отсутствие определений программных конечных выключателей, ограничений рабочего поля или защищенных областей.

Состояние программного конечного выключателя и ограничения рабочего поля

Программные конечные выключатели и ограничения рабочего поля учитываются, только если они при выполнении `CALCPOSI()` активны. На состояние можно повлиять, например, через:

- Машинные данные: MD21020 $MC_WORKAREA_WITH_TOOL_RADIUS
- Установочные данные: $AC_WORKAREA_CS_...

- Интерфейсные сигналы ЧПУ/PLC: DB31, ... DBX12.2 / 3
- Команды: WALIMON / WALIMOF

Программные конечные выключатели и трансформации

В случае CALCPOSI() при различных кинематических трансформациях (например, TRANSMIT) на основе многозначности в определенных местах диапазона перемещения позиции осей станка (MCS) не всегда можно однозначно определить по позициям геометрических осей (WCS). В обычном режиме перемещения однозначность следует, как правило, из предыстории и условия, что непрерывному движению в WCS должно соответствовать непрерывное движение осей станка в MCS. Поэтому в подобных случаях при контроле программных конечных выключателей для разрешения многозначности используется текущая позиция станка на момент выполнения функции CALCPOSI().

Примечание

Остановка предварительной обработки

При использовании CALCPOSI() в сочетании с трансформацией только пользователь несет ответственность за программирование для синхронизации позиций станка с предварительной обработкой перед CALCPOSI() остановки предварительной обработки (STOPRE).

Отступ защищенных областей и обычные защищенные области

В случае обычных защищенных областей **не** гарантируется, что при движении перемещения на заданном пути перемещения указанный в параметре <Limit>[3] безопасный отступ соблюдается в отношении всех защищенных областей. Гарантируется лишь, что при удлинении возвращаемой в <Dist> конечной точки на безопасный отступ в направлении перемещения ни одна защищенная область не нарушается. Но ход прямой может пролегать на любом расстоянии от защищенной области.

Отступ защищенных областей и защищенные области предотвращения столкновений

В случае защищенных областей предотвращения столкновения гарантируется, что при движении перемещения на заданном пути перемещения указанный в параметре <Limit>[3] безопасный отступ соблюдается в отношении всех защищенных областей.

Указанный в параметре <Limit>[3] безопасный отступ действует, только если:

<Limit>[3] > (MD10619 $MN_COLLISION_TOLERANCE)

Если в параметре <TestLim> установлен бит 4 (вычисление направления дальнейшего перемещения), то содержащийся в <DIST> вектор направления действителен, только если установлена шестая позиция в возвращаемом значении функции (<состояние>). Если такое направление определить невозможно, поскольку либо были нарушены защищенные области, либо активна какая-либо трансформация, входное значение в <DIST> остается неизменным. Дополнительное сообщение об ошибке не выводится.

4.7 Специальные команды перемещения

4.7.1 Подвод к кодированным позициям (CAC, CIC, CDC, CACP, CACN)

Команды пути для "Подвод к кодированным позициям" позволяют перемещать линейные и круговые оси (при указании номеров позиций) на постоянные помещенные в таблицы машинных данных позиции осей.

Синтаксис

```
CAC(<n>)
CIC(<n>)
CACP(<n>)
CACN(<n>)
```

Объяснение

`CAC(<n>)`:	Выполнить подвод к кодированной позиции с номером позиции n
`CIC(<n>)`:	Выполнить подвод к кодированной позиции, исходя из актуального номера позиции, на n-количество позиций вперед (+) или назад (–)
`CDC(<n>)`:	Выполнить подвод к кодированной позиции с номером позиции n по кратчайшему пути (только для круговых осей)
`CACP(<n>)`:	Выполнить подвод к кодированной позиции с номером позиции n в положительном направлении (только для круговых осей)
`CACN(<n>)`:	Выполнить подвод к кодированной позиции с номером позиции n в отрицательном направлении (только для круговых осей)
`<n>`:	Номер позиции в таблице машинных данных Диапазон значений: 0, 1, ... (макс. число мест в таблице - 1)

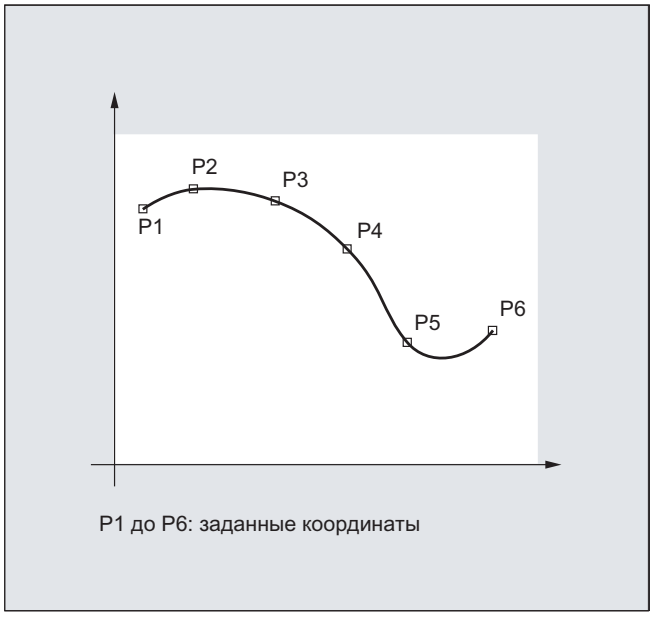
Пример: Подвод к кодированным позициям позиционирующей оси

Программный код	Комментарий
`N10 FA[B]=300`	; Подача для позиционирующей оси `B`
`N20 POS[B]=CAC(10)`	; Выполнить подвод к кодированной позиции с номером позиции 10
`N30 POS[B]=CIC(-4)`	; Выполнить подвод к кодированной позиции с "актуальным номером позиции" - 4

4.7.2 Сплайн-интерполяция (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Аналитически точное описание произвольно изогнутых контуров на деталях невозможно. Поэтому аппроксимация таких контуров выполняется через ограниченное число опорных точек, к примеру, при оцифровке поверхностей. Для создания оцифрованной поверхности на детали опорные точки должны быть объединены в одно описание контура. Это обеспечивает сплайн-интерполяция.

Сплайн определяет кривую, состоящую из полиномов 2-го или 3-го порядка. Свойства на опорных точках сплайна могут определяться **в зависимости от используемого типа сплайна** .



Следующие типы сплайнов доступны для SINUMERIK solution line:

- A-сплайн
- B-сплайн
- C-сплайн

Синтаксис

Общая информация:

```
ASPLINE X... Y... Z... A... B ... C ...
BSPLINE X... Y... Z... A... B ... C ...
CSPLINE X... Y... Z... A... B ... C ...
```

Для B-сплайна дополнительно можно запрограммировать:

```
PW=<n>
SD=2
PL=<значение>
```

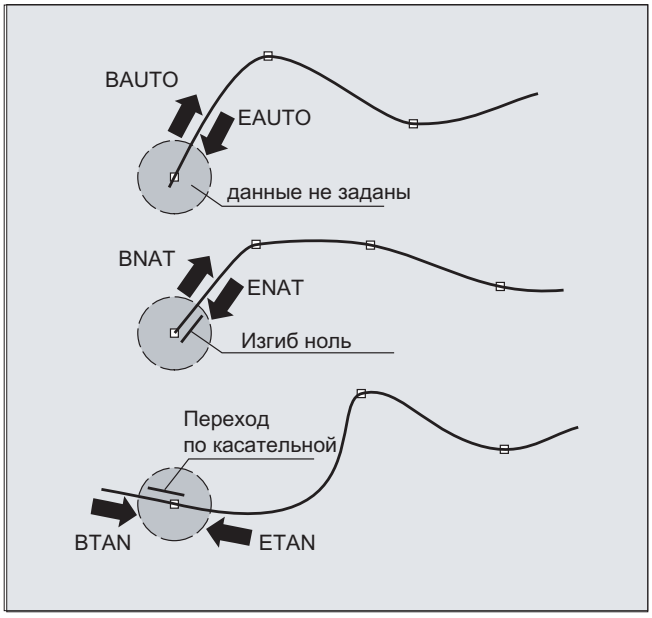
Для A- и C-сплайна дополнительно можно запрограммировать:

BAUTO / BNAT / BTAN

EAUTO / ENAT / ETAN

Значение

Тип сплайн-интерполяции:			
ASPLINE:	Команда для включения A-сплайн-интерполяции		
BSPLINE:	Команда для включения B-сплайн-интерполяции		
CSPLINE:	Команда для включения C-сплайн-интерполяции		
	Команды ASPLINE, BSPLINE и CSPLINE действуют модально и относятся к группе команд перемещения.		
Опорные точки или контрольные точки:			
X... Y... Z... A... B ... C...	Позиции в декартовых координатах		
Вес точки (только B-сплайн):			
PW:	С помощью команды PW для каждой опорной точки можно запрограммировать т.н. "Вес точки".		
<n>:	"Вес точки"		
	Диапазон значений:	0 ≤ n ≤ 3	
	Размер шага:	0.0001	
	Действие:	n > 1	Кривая сильнее притягивается к контрольной точке.
		n < 1	Кривая слабее притягивается к контрольной точке.
Порядок сплайна (только B-сплайн):			
SD:	Стандартно используется полигон 3-го порядка. Но через программирование SD=2 возможно и использование полигона 2-го порядка.		
Расстояние между узловыми точками (только B-сплайн):			
PL :	Выполняется внутреннее подходящее вычисление расстояний между узловыми точками. Но СЧПУ может обрабатывать и заданные расстояния между узловыми точками, которые с помощью команды PL указываются как т.н. длине интервалов параметров.		
<значение>:	Длина интервала параметров		
	Диапазон значений:	Как размер перемещения	
Переходная характеристика в начале сплайна (только A- или C-сплайн):			
BAUTO:	Данные для переходной характеристики не задаются. Начало следует из положения первой точки.		
BNAT:	Изгиб ноль		
BTAN:	Тангенциальный переход к предшествующему кадру (позиция стирания)		

Переходная характеристика в конце сплайна (только A- или C-сплайн):	
EAUTO:	Данные для переходной характеристики не задаются. Конец следует из положения последней точки.
ENAT:	Изгиб ноль
ETAN:	Тангенциальный переход к предшествующему кадру (позиция стирания)
	

Примечание

Программируемая переходная характеристика не влияет на B-сплайн. B-сплайн в стартовой и конечной точке всегда является тангенциальным к контрольному полигону.

Граничные условия

- Возможно использование коррекции радиуса инструмента.
- Контроль столкновений выполняется в проекции на плоскость.

Примеры

Пример 1: B-сплайн

Программный код 1 (все веса 1)

```
N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20
```

Программный код 1 (все веса 1)

```
N40 X20 Y40
N50 X30 Y30
N60 X40 Y45
N70 X50 Y0
```

Программный код 2 (различный вес)

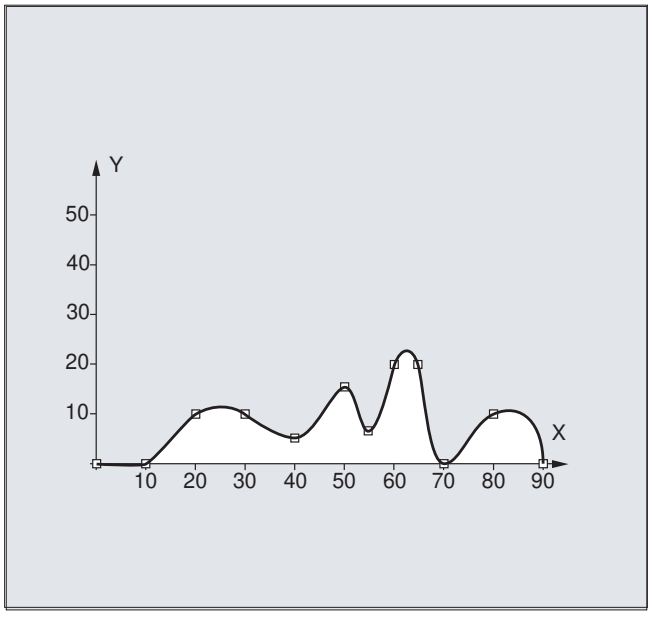
```
N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20 PW=2
N40 X20 Y40
N50 X30 Y30 PW=0.5
N60 X40 Y45
N70 X50 Y0
```

Программный код 3 (контрольный полигон)	Комментарий
`N10 G1 X0 Y0 F300 G64`	
`N20`	; отсутствует
`N30 X10 Y20`	
`N40 X20 Y40`	
`N50 X30 Y30`	
`N60 X40 Y45`	
`N70 X50 Y0`	

Пример 2: C-сплайн, изгиб ноль в начале и в конце

Программный код

```
N10 G1 X0 Y0 F300
N15 X10
N20 BNAT ENAT
N30 CSPLINE X20 Y10
N40 X30
N50 X40 Y5
N60 X50 Y15
N70 X55 Y7
N80 X60 Y20
N90 X65 Y20
N100 X70 Y0
N110 X80 Y10
N120 X90 Y0
N130 M30
```



Пример 3: сплайн-интерполяция (A-сплайн) и трансформация координат (ROT)

Главная программа:

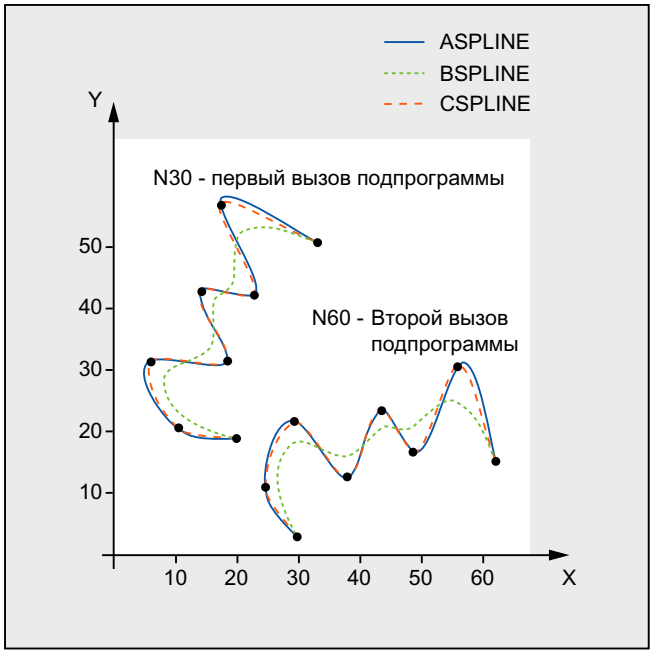
Программный код	Комментарий
N10 G00 X20 Y18 F300 G64	; Подвод в стартовую точку.
N20 ASPLINE	; Активировать тип интерполяции A–сплайна.
N30 KONTUR	; Первый вызов подпрограммы.
N40 ROT Z–45	; Трансформация координат: Поворот WCS на -45° вокруг оси Z.

Программный код	Комментарий
N50 G00 X20 Y18	Подвод в стартовую точку контура.
N60 KONTUR	; Второй вызов подпрограммы.
N70 M30	; конец программы

Подпрограмма "Kontur" (содержит координаты опорных точек):

Программный код
N10 X20 Y18
N20 X10 Y21
N30 X6 Y31
N40 X18 Y31
N50 X13 Y43
N60 X22 Y42
N70 X16 Y58
N80 X33 Y51
N90 M1

На рисунке ниже, наряду со сплайном, полученным из примера программирования (ASPLINE), содержаться и сплайны, которые получились бы при активации B- или C-сплайн-интерполяции (BSPLINE, CSPLINE):



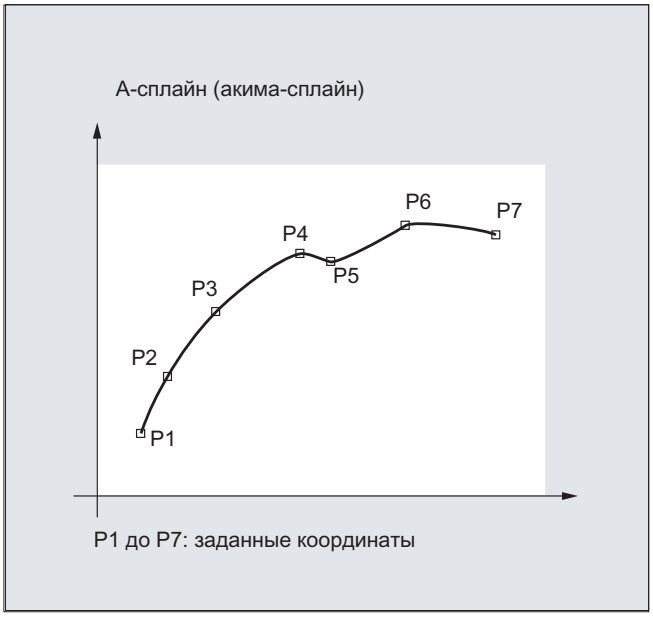
Дополнительная информация

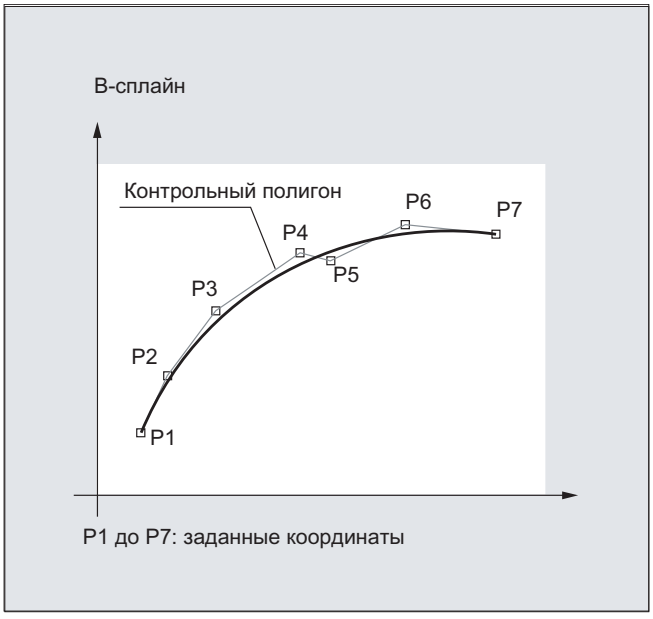
Преимущества сплайн-интерполяции

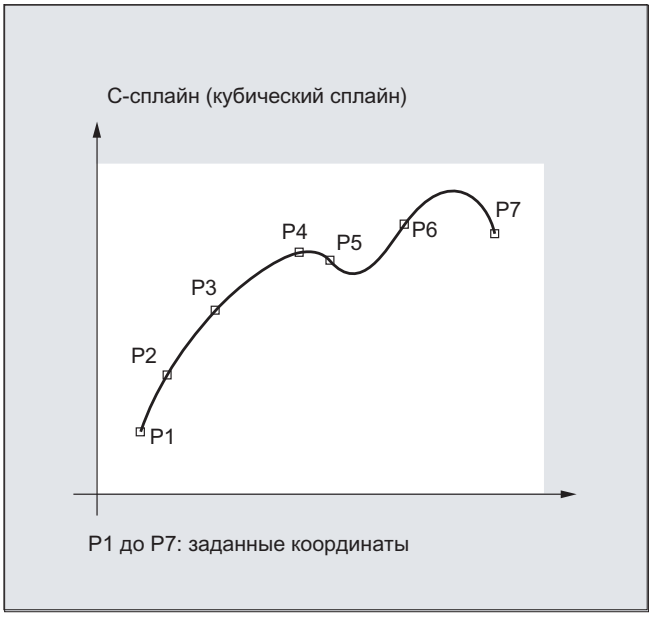
Использование сплайн-интерполяции позволяет, в отличие от использования линейных кадров G01, получить следующие преимущества:

- Уменьшение числа необходимых для описания контура кадров программы обработки детали
- Мягкий, щадящий для механики ход кривой на переходах между кадрами программы обработки детали

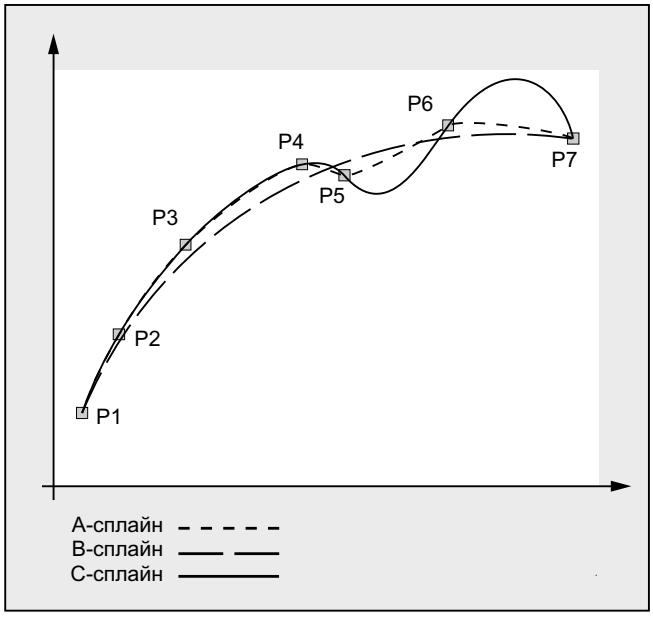
Свойства и использование различных типов сплайнов

Тип сплайна	Свойства и использование
A-сплайн	 **Свойства:** • Проходит точно через заданные опорные точки. • Ход кривой стабильный по касательной и кривизне. • Практически не создает нежелательных колебаний. • Сфера влияния изменений опорных точек является локальной, т. е. изменение одной опорной точки сказывается только макс. на 6 соседних опорных точках. **Использование:** A-сплайн подходит прежде всего для интерполяции ходов кривой с большими изменениями наклона (например, ступенчатые ходы кривой).

Тип сплайна	Свойства и использование
B-сплайн	 **Свойства:** • Проходит не через заданные опорные точки, а только вблизи. Кривая притягивается опорными точками. Посредством коэффициента эквивалентности опорных точек возможно дополнительное управление ходом кривой. • Ход кривой стабильный по касательной и кривизне. • Практически не создает нежелательных колебаний. • Сфера влияния изменений опорных точек является локальной, т. е. изменение одной опорной точки сказывается только макс. на 6 соседних опорных точках. **Использование:** Первоначально B-сплайн задумывался как интерфейс к системам CAD.

Тип сплайна	Свойства и использование
C-сплайн	 **Свойства:** • Проходит точно через заданные опорные точки. • Ход кривой стабильный по касательной и кривизне. • Часто создает нежелательные колебания, особенно в местах с большими изменениями наклона. • Сфера влияния изменений опорных точек является глобальной, т. е. изменение одной опорной точки сказывается на всем ходе кривой. **Использование:** C-сплайн может успешно использоваться тогда, когда опорные точки лежат на аналитически известной кривой (окружность, парабола, гипербола)

Сравнение трех типов сплайнов в случае одинаковых опорных точек



Мин. число сплайн-кадров

G-команды ASPLINE, BSPLINE, CSPLINE соединяют конечные точки кадров сплайнами. Для этого на стадии предварительной обработки необходимо одновременное вычисление ряда кадров (конечных точек). Размер буфера для вычисления стандартно составляет 10 кадров. Не любая информация кадра является конечной точкой сплайна. Но СЧПУ из 10 кадров требуется определенное число кадров конечных точек сплайна:

Тип сплайна	Мин. число сплайн-кадров
A-сплайн:	Из каждых 10 кадров как минимум **4** должны быть сплайн-кадрами. Кадры комментариев и вычисления параметров при этом не учитываются.
B-сплайн:	Из каждых 10 кадров как минимум **6** должны быть сплайн-кадрами. Кадры комментариев и вычисления параметров при этом не учитываются.
C-сплайн:	Необходимое мин. число сплайн-кадров получается из следующей суммы: Значение из MD20160 $MC_CUBIC_SPLINE_BLOCKS + 1 В MD20160 вносится число точек, через которые вычисляется сегмент сплайна. Стандартная настройка 8. Поэтому из каждых 10 кадров в стандартном случае как минимум **9** должны быть сплайн-кадрами.

Примечание

При значении, ниже допустимого, выводится ошибка, как и в том случае, когда участвующая в сплайне ось программируется как позиционирующая ось.

Объединение коротких сплайн-кадров

При сплайн-интерполяции могут возникать короткие сплайн-кадры, которые приводят не ненужному уменьшению скорости движения по траектории. С помощью функции "Соединение коротких сплайн-кадров" эти кадры могут быть соединены таким образом, что полученная длина кадра является достаточной и не приводит к уменьшению скорости движения по траектории.

Функция активируется через спец. для канала машинные данные:

MD20488 $MC_SPLINE_MODE (настройка для сплайн-интерполяции)

Дополнительная информация: Описание функций – Основные функции

4.7.3 Соединение сплайнов (SPLINEPATH)

Интерполирующие в соединении сплайнов оси выбираются с помощью команды `SPLINEPATH`. До восьми траекторных осей возможно при сплайн-интерполяции.

Примечание

Если `SPLINEPATH` не программируется явно, то первые три оси канала перемещаются как соединение сплайнов.

Синтаксис

Определение соединения сплайнов выполняется в отдельном кадре:

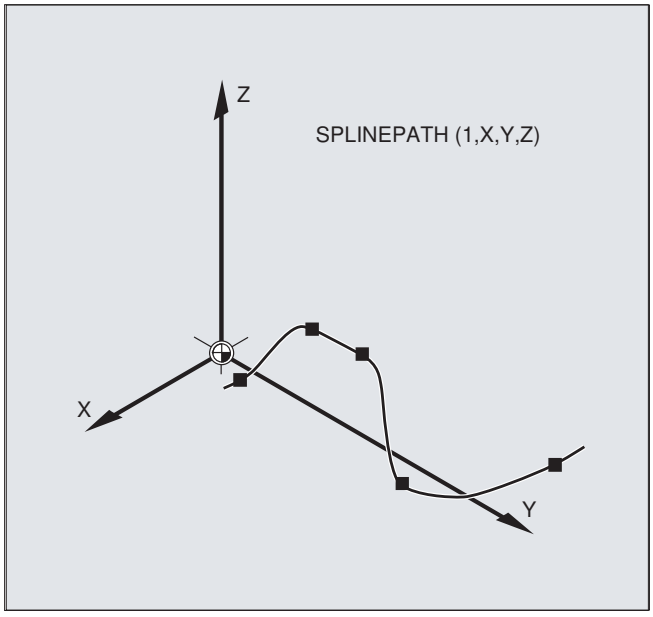
```
SPLINEPATH(n,X,Y,Z,…)
```

Значение

`SPLINEPATH`:	Команда для определения соединения сплайнов
`n`:	=1 (фиксированное значение)
`X,Y,Z,…` :	Идентификаторы интерполирующих в соединении сплайнов траекторных осей

Пример: Соединение сплайнов с тремя траекторными осями

Программный код	Комментарий
`N10 G1 X10 Y20 Z30 A40 B50 F350`	
`N11 SPLINEPATH(1,X,Y,Z)`	; Соединение сплайнов
`N13 CSPLINE BAUTO EAUTO X20 Y30 Z40 A50 B60`	`;` C-сплайн
`N14 X30 Y40 Z50 A60 B70`	; Опорные точки
…	
`N100 G1 X… Y…`	; Отмена сплайн-интерполяции



4.7.4 Включение / выключение сжатия кадра ЧПУ (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF)

Функции компрессора линейных кадров (и в зависимости от параметрирования также круговых кадров и/или кадров ускоренного хода) включаются / отключаются G-командами из группы G 30. Команды действуют модально.

Синтаксис

```
COMPON / COMPCURV / COMPCAD / COMPSURF
...
COMPOF
```

Объяснение

COMPON:	Включение функции компрессора COMPON
COMPCURV:	Включение функции компрессора COMPCURV
COMPCAD:	Включение функции компрессора COMPCAD
COMPSURF:	Включение функции компрессора COMPSURF
COMPOF:	Выключение текущей активной функции компрессора

Примечание

Для дополнительного улучшения качества поверхности можно использовать функцию перешлифовки G642 и ограничение рывка SOFT . Эти команды записываются в начале программы.

Пример: COMPCAD

Программный код	Комментарий
N10 G00 X30 Y6 Z40	
N20 G1 F10000 G642	; Включение: Функция перешлифовки G642
N30 SOFT	; Включение: Ограничение рывка SOFT
N40 **COMPCAD**	; Включение: Функция компрессора COMPCAD
N50 FIFOCTRL	
N24050 Z32.499	; 1-й кадр перемещения
N24051 X41.365 Z32.500	; 2-й кадр перемещения
...	
N99999 X... Z...	; Последний кадр перемещения
COMPOF	; Функция компрессора выкл.
...	

4.7.5 Полиномиальная интерполяция (POLY, POLYPATH, PO, PL)

В прямом смысле в случае полиномиальной интерполяции (POLY) речь идет не о виде сплайн-интерполяции. В первую очередь она задумана как интерфейс для программирования созданных на внешних устройствах сплайн-кривых. При этом сегменты сплайна могут программироваться напрямую.

Такой тип интерполяции снимает с ЧПУ задачу по вычислению коэффициентов полиномов. Ее оптимальное использование возможно тогда, когда коэффициенты поступают напрямую от системы CAD или постпроцессора.

Синтаксис

Полином 3-го порядка:
```
POLY PO[X]=(xe,a2,a3) PO[Y]=(ye,b2,b3) PO[Z]=(ze,c2,c3) PL=n
```

Полиномы 5-го порядка и новый синтаксис полинома:
```
POLY X=PO(xe,a2,a3,a4,a5) Y=PO(ye,b2,b3,b4,b5) Z=PO(ze,c2,c3,c4,c5)
PL=n
POLYPATH("AXES","VECT")
```

Примечание

Сумма запрограммированных в одном кадре ЧПУ коэффициентов полиномов и осей не должна превышать макс. разрешенного числа осей на кадр.

Значение

POLY :	Включение полиномиальной интерполяции с помощью кадра с POLY.
POLYPATH :	Полиномиальная интерполяция с возможностью выбора для обоих групп осей AXIS или VECT

PO[идентификатор оси/переменная] :	Конечный точки и коэффициенты полиномов
X, Y, Z :	Идентификатор оси
xe, ye, ze :	Указание конечной позиции для соответствующей оси; диапазон значений как размер перемещения
a2, a3, a4, a5 :	Коэффициенты a_2, a_3, a_4, и a_5 записываются с их значением; диапазон значений как размер перемещения. Соответствующий последний коэффициент не нужен, если он имеет значение ноль.
PL :	Длина интервала параметров, на котором определены полиномы (область определения функции f(p)). Интервал всегда начинается на 0, p может принимать значения от 0 до PL. Теоретический диапазон значений для PL: 0,0001 ... 99 999,9999 **Примечание:** Значение PL действует для кадра, в котором оно стоит. Если PL не запрограммировано, то действует PL=1.

Включение/выключение полиномиальной интерполяции

Полиномиальная интерполяция включается в программе обработки детали через G-команду POLY.

G-команда POLY вместе с G0, G1, G2, G3, ASPLINE, BSPLINE и CSPLINE относится к 1-й G-группе.

Оси, запрограммированные только с именем и конечной точкой (к примеру, X10), перемещаются линейно. Если таким образом запрограммированы все оси, то поведение СЧПУ как при G1.

Полиномиальная интерполяция снова не явно отключается через программирование другой команды 1-й G-группы (к примеру, G0, G1).

Коэффициент полинома

Значение PO (PO[]=) или ...=PO(...) указывает все коэффициенты полиномов для оси. Согласно порядку полинома несколько значений указываются разделенными запятой. Внутри одного кадра возможны различные порядки полинома для различных осей.

Подпрограмма POLYPATH

С `POLYPATH(...)` полиномиальная интерполяция может разрешаться выборочно для определенных групп осей:

Только траекторные и дополнительные оси:	`POLYPATH("AXES")`
Только оси ориентации: (при перемещении с трансформацией ориентации)	`POLYPATH("VECT")`

Соответствующие не разрешенные оси перемещаются линейно.

Стандартно полиномиальная интерполяция разрешена для обоих групп осей.

Через программирование без параметра `POLYPATH( )` полиномиальная интерполяция деактивируется для всех осей.

Пример

Программный код	Комментарий
`N10 G1 X… Y… Z… F600`	
`N11 POLY PO[X]=(1,2.5,0.7) PO[Y]=(0.3,1,3.2) PL=1.5`	; Полиномиальная интерполяция вкл
`N12 PO[X]=(0,2.5,1.7) PO[Y]=(2.3,1.7) PL=3`	
…	
`N20 M8 H126 …`	
`N25 X70 PO[Y]=(9.3,1,7.67) PL=5`	; Смешанные данные для осей
`N27 PO[X]=(10,2.5) PO[Y]=(2.3)`	`; PL` не запрограммирован; действует `PL=1`
`N30 G1 X… Y… Z.`	; Полиномиальная интерполяция выкл
…	

Пример: Новый синтаксис полинома

Продолжающий действовать синтаксис полинома	Новый синтаксис полинома
`PO[`идентификатор оси`]=(.. , ..)`	Идентификатор оси`=PO(.. , ..)`
`PO[PHI]=(.. , ..)`	`PHI=PO(.. , ..)`
`PO[PSI]=(.. , ..)`	`PSI=PO(.. , ..)`
`PO[THT]=(.. , ..)`	`THT=PO(.. , ..)`
`PO[]=(.. , ..)`	`PO(.. , ..)`
`PO[`переменная`]=IC(.. , ..)`	Переменная`=PO IC(.. , ..)`

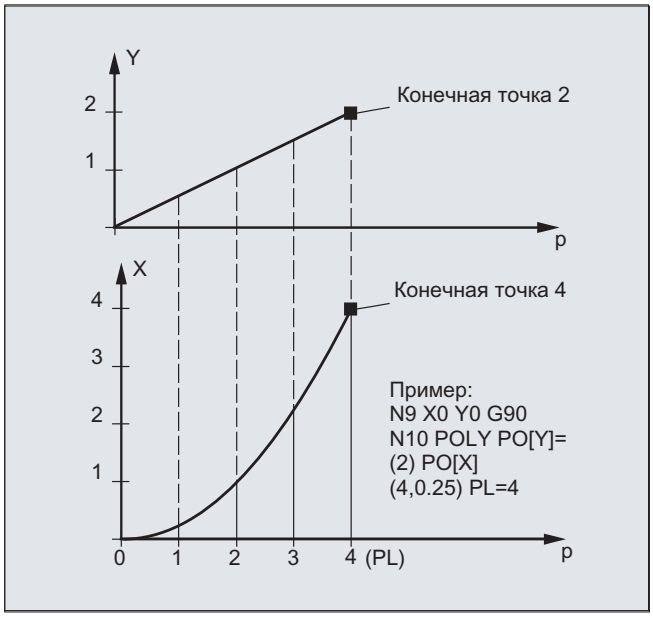
Пример: Кривая в плоскости X/Y

Программирование

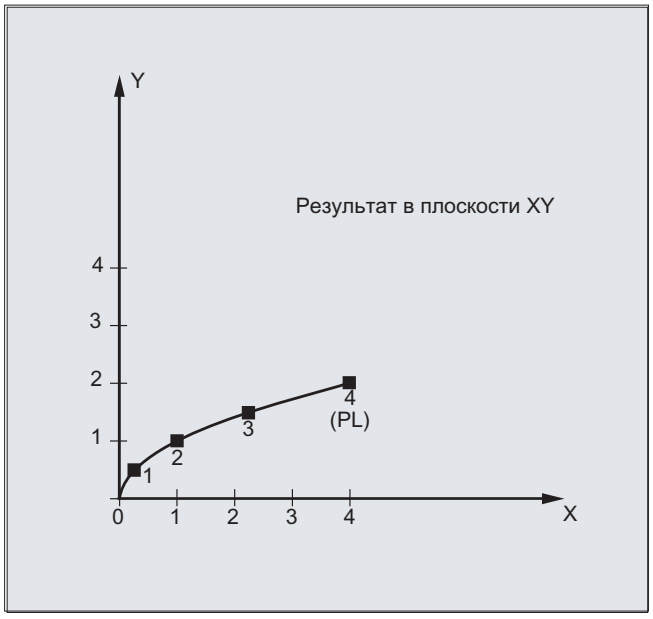
Программный код
`N9 X0 Y0 G90 F100`

Программный код
`N10 POLY PO[Y]=(2) PO[X]=(4,0.25) PL=4`

Ход кривых X(p) и Y(p)



Ход кривой в плоскости XY



Описание

В общей форме функция полинома выглядит следующим образом:

$f(p)= a_0 + a_1p + a_2p^2 + . . . + a_np^n$

где: a_i: постоянные коэффициенты (i = 0, 1, ..., n)
p: параметры

В СЧПУ можно программировать полиномы не выше 5-го порядка:

$f(p)= a_0 + a_1p + a_2p^2 + a_3p^3 + a_4p^4 + a_5p^5$

Посредством присваивания конкретных значений коэффициентам можно создавать различные ходы кривой, например, линейные, параболические, степенные функции.

Прямая создается через $a_2 = a_3 = a_4 = a_5 = 0$:

$f(p) = a_0 + a_1p$

Дальше действует:

$a_{0:}$ Позиция оси на конце предшествующего кадра

$p = PL$

$a_1 = (x_E - a_0 - a_2*p^2 - a_3*p^3) / p$

Можно программировать полиномы **без** активации полиномиальной интерполяции через G-команду `POLY`. Но в этом случае выполняется интерполяция не запрограммированных полиномов, а осуществляется линейный подвод к запрограммированным конечным точкам осей (`G1`). Только после явной активации полиномиальной интерполяции в программе обработки детали (`POLY`) запрограммированные полиномы начинают перемещаться как таковые.

Особенность: Полином-знаменатель

Для геометрических осей с помощь `PO[]=(...)` без указания имени оси также можно запрограммировать общий полином-знаменатель, т. е. движение геометрических осей интерполируется как частное двух полиномов.

Таким образом, могут быть точно представлены, например, конические сечения (круг, эллипс, парабола, гипербола).

Пример:

Программный код	Комментарий
`POLY G90 X10 Y0 F100`	; Гео-оси перемещаются линейно на позицию `X10, Y0`.
`PO[X]=(0,-10) PO[Y]=(10) PO[]=(2,1)`	; Гео-оси перемещаются по четверти круга на `X0 Y10`.

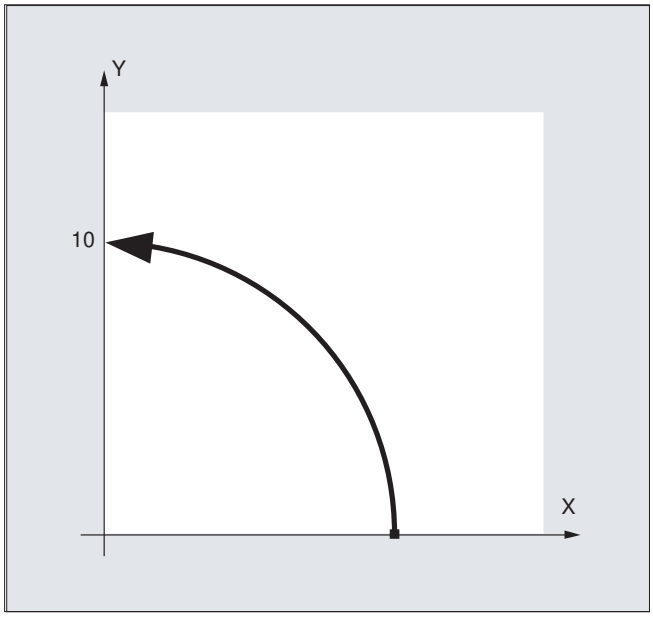
Постоянный коэффициент (a_0) полинома-знаменателя всегда принимается за 1. Запрограммированная конечная точка не зависит от `G90` / `G91`.

Из запрограммированных значений X(p) и Y(p) вычисляются как:

$X(p) = (10 - 10 * p^2) / (1 + p^2)$
$Y(p) = 20 * p / (1 + p^2)$
где $0 \leq p \leq 1$

На основе запрограммированных начальных точек, конечных точек, коэффициента a_2 и PL=1 получаются следующие промежуточные результаты:

Числитель (X) = $10 + 0 * p - 10 * p^2$
Числитель (Y) = $0 + 20 * p + 0 * p^2$
Знаменатель = $1 + p^2$



При включенной полиномиальной интерполяции программирование полинома-знаменателя с нулями внутри интервала `[0,PL]` отклоняется с ошибкой. Полином-знаменатель не влияет на движение дополнительных осей.

Примечание

Коррекция радиуса инструмента для полиномиальной интерполяции может включаться с `G41`, `G42` и использоваться как для линейной или круговой интерполяции.

4.7.6 Устанавливаемое соотношение траекторий (SPATH, UPATH)

При полиномиальной интерполяции (`POLY`, `ASPLINE`, `BSPLINE`, `CSPLINE`, `COMP...`) позиции траекторных осей i задаются через полиномы $p_i(U)$. Параметр кривой U при этом перемещается внутри кадра ЧПУ от 0 до 1.

С помощью FGROUP выбираются оси (оси FGROUP), к которым относится подача по траектории F. Но интерполяция с постоянной скоростью на пути S осей FGROUP означает при полиномиальной интерполяции, как правило, не постоянное изменение параметра кривой U. Поэтому для осей, не содержащихся в FGROUP, возможен выбор из двух способов следования этих осей за осями FGROUP:

- Синхронно ходу траектории S (SPATH)
- Синхронно параметру кривой U (UPATH)

Синтаксис

```
SPATH
UPATH
```

Объяснение

SPATH:	Не содержащиеся в FGROUP оси перемещаются относительно хода траектории S
UPATH:	Не содержащиеся в FGROUP оси перемещаются относительно параметра кривой U

Примечание

UPATH и SPATH определяют и связь полинома слова F (FPOLY, FCUB, FLIN) с движением по траектории.

Граничные условия

SPATH или UPATH не имеют значения в следующих случаях:

- линейная интерполяция (G1)
- круговая интерполяция (G2, G3)
- кадры резьбы (G33, G34, G35, G33x, G63)
- все траекторные оси имеются в FGROUP

Пример

Пример ниже показывает разницу между двумя типами управления движением.

Программный код

```
N10 FGROUP(X,Y,Z)
N15 G1 X0 A0 F1000 SPATH                ; SPATH
N20 POLY PO[X]=(10,10) A10
```

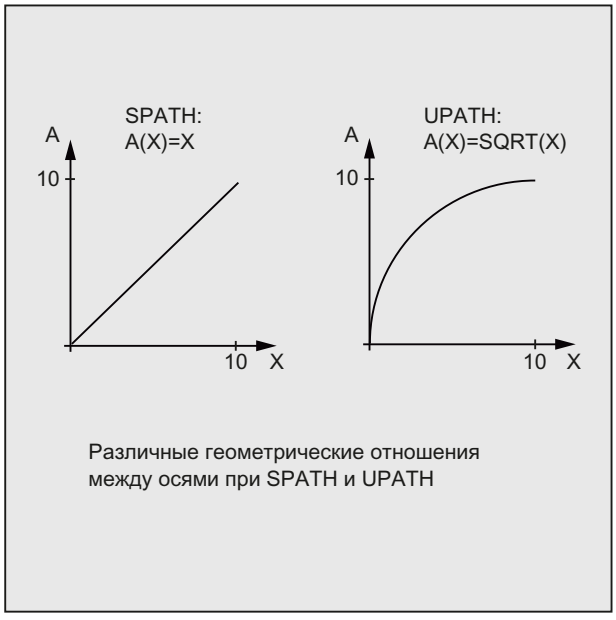
Программный код

```
N10 FGROUP(X,Y,Z)
N15 G1 X0 A0 F1000 UPATH                ; UPATH
```

Программный код

```
N20 POLY PO[X]=(10,10) A10
```

В обоих сегментах программы ход S осей FGROUP в `N20` зависит от квадрата параметра кривой U. Поэтому вдоль хода X получаются различные позиции синхронной оси A, в зависимости от того, активна ли `SPATH` или `UPATH`.



Различные геометрические отношения между осями при SPATH и UPATH

Дополнительная информация

Поведение СЧПУ при Reset и машинные/опциональные данные

После Reset действует определенный посредством MD20150 $MC_GCODE_RESET_VALUES[44] G-команда (45-я G-группа).

Значение по умолчанию для типа перешлифовки определяется посредством MD20150 $MC_GCODE_RESET_VALUES[9] (10-я G-группа).

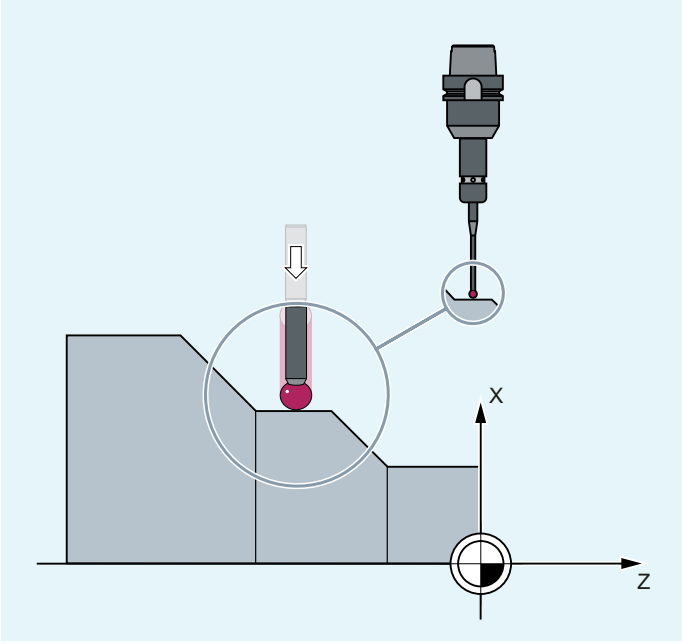
Спец. для оси машинные данные MD33100 $MA_COMPRESS_POS_TOL[<n>] имеют расширенное значение: они содержат допуски для функции компрессора и для перешлифовки с G642.

4.7.7 Специфическое для канала измерение (MEAS, MEAW)

При специфическом для канала измерении активация процесса измерения для канала ЧПУ постоянно выполняется из программы обработки детали, выполняемой в соответствующем канале. В кадре измерения программируется соответственно **одно** пусковое событие (передний или задний фронт измерительного щупа) и **один** режим измерения со стиранием остатка пути (MEAS) или без стирания остатка пути (MEAW). В этом случае в процессе измерения участвуют **все** запрограммированные в кадре измерения оси.

Как только кадр измерения активен, измерительный щуп подводится к детали. При контактном фронте измерительного щупа для всех запрограммированных в кадре измерения осей измеряются позиции и записываются для каждой оси в соответствующую ячейку памяти.

Результаты измерения читаются в программе обработки детали или с синхронными действиями, как в системе координат станка, так и детали.



Синтаксис

```
MEAS=<TE> G... X... Y... Z...
MEAW=<TE> G... X... Y... Z...
```

Объяснение

MEAS:	Измерение **со** стиранием остатка пути	
	Активность:	покадрово
MEAW:	Измерение **без** стирания остатка пути	
	Активность:	покадрово
	Использование:	Для задач измерения, в которых в любом случае необходим подвод к запрограммированной позиции.

<TE>:	Пусковое событие для запуска измерения		
	Тип:	INT	
	Диапазон значений:		-2, -1, 1, 2
	Значение:	(+)1	передний фронт измерительного щупа 1 (на входе измерительной системы 1)
		-1	задний фронт измерительного щупа 1 (на входе измерительной системы 1)
		(+)2	передний фронт измерительного щупа 2 (на входе измерительной системы 2)
		-2	задний фронт измерительного щупа 2 (на входе измерительной системы 2)
	Примечание: Существует макс. 2 измерительных щупа (в зависимости от конфигурации).		
G...:	Тип интерполяции, к примеру, G0, G1, G2 или G3		
X... Y... Z...:	Конечные точки в декартовых координатах		

Примечание

MEAS и MEAW действуют покадрово и программируются вместе с операторами движения. При этом подача и тип интерполяции (G0, G1, ...), как и число осей, должны быть согласованы с соответствующей задачей измерения.

Пример

Программный код	Комментарий
...	
N10 MEAS=1 G1 F1000 X100 Y730 Z40	; Кадр измерения со щупом первого измерительного входа и линейная интерполяция. Остановка предварительной обработки создается автоматически.
...	

Дополнительная информация

Запрос состояния

Если в программе необходимо определить, был ли измерительный щуп отклонен или контактировал, то это можно запросить через следующие системные переменные:

Системная переменная	Объяснение	Тип данных	Значение	
$A_PROBE[<n>]	Состояние измерительного щупа	INT	0	Измерительный щуп не отклонен.
			1	Измерительный щуп отклонен.

Системная переменная	Объяснение	Тип данных	Значение	
$AC_MEA[<n>]	Состояние контактирования измерительного щупа $AC_MEA[<n>] автоматически сбрасывается в начале измерения.	INT	0	Измерительный щуп не контактировал:
			1	Контакт измерительного щупа произошел.

<n> = номер измерительного щупа

Запись измеренных значений

При спец. для канала измерении регистрируются позиции всех перемещенных траекторных и позиционирующих осей кадра (макс. число осей в зависимости от конфигурации СЧПУ). При MEAS после контакта измерительного щупа происходит определенное притормаживание.

Примечание

Если в кадре измерения запрограммирована гео-ось, то сохраняются измеренные значения для всех текущих гео-осей.

Если в кадре измерения запрограммирована участвующая в трансформации ось, то сохраняются измеренные значения всех участвующих в этой трансформации осей.

Чтение результатов измерения

Результаты измерения щупом для обработанных с помощью щупа осей могут быть считаны в программе обработки детали и в синхронных действиях через следующие системные переменные:

Системная переменная	Объяснение
$AA_MM[<ось>]	Результаты измерения щупом в системе координат станка.
$AA_MW[<ось>]	Результаты измерения щупом в системе координат детали.

4.7.8 Спец. для оси измерение (MEASA, MEAWA, MEAC) (опция):

При спец. для оси измерении активация может выполняться в программе обработки детали **или** в синхронных действиях. При наличии двух измерительных системы для оси можно использовать обе.

Доступны следующие методы измерений:

- Измерение со стиранием остатка пути (MEASA)
- Измерение без стирания остатка пути (MEAWA)
- Непрерывное измерение без стирания остатка пути (MEAC)

MEASA, MEAWA для соответствующей запрограммированной оси позволяют регистрировать до четырех измеренных значений на измерение и сохранять их согласно пусковому событию в системных переменных.

При непрерывном измерении с MEAC результаты измерения сохраняются в переменных FIFO.

Синтаксис

```
MEASA[<Axis>]=(<Mode>,<TE1>,...,<TE4>)
MEAWA[<Axis>]=(<Mode>,<TE1>,...,<TE4>)
MEAC[<Axis>]=(<Mode>,<MeasMem>,<TE1>,...,<TE4>)
```

Примечание

MEASA и MEAWA действуют покадрово и могут быть запрограммированы вместе в одном кадре. Если, напротив, MEASA/MEAWA вместе с MEAS/MEAW программируются в одном кадре, то следует сообщение об ошибке.

Объяснение

MEASA:	Спец. для оси измерение со стиранием остатка пути	
	Активность:	покадрово
MEAWA:	Спец. для оси измерение без стирания остатка пути	
	Активность:	покадрово
MEAC:	Спец. для оси непрерывное измерение без стирания остатка пути	
	Активность:	покадрово
<ось>:	Имя оси канала, использованной для измерения	
<Mode>:	Двухзначное (xx) число для указания режима работы (режим измерения и измерительная система)	
	Первый десятичный разряд: режим измерения Задает, как должны активироваться пусковые события: во временной или в запрограммированной последовательности.	
	x**0**	Отменить задание измерения.
	x**1**	До 4 **различных** одновременно активируемых пусковых событий.
	x**2**	До 4 последовательно активируемых пусковых событий.
	x**3**	До 4 последовательно активируемых пусковых событий, но без контроля пускового события 1 при запуске (аварийные сообщения 21700/21703 подавляются). **Примечание:** Этот режим при MEAC невозможен.
	Второй десятичный разряд: измерительная система Задает измерительную систему для выполнения измерения.	
	0x (или не указано)	Активная измерительная система
	1x	Измерительная система 1
	2x	Измерительная система 2
	3x	Обе измерительные системы

<TE>:	Пусковое событие для запуска измерения	
	Тип:	INT
	Диапазон значений:	-2, -1, 1, 2
	(+)1	передний фронт измерительного щупа 1
	-1	задний фронт измерительного щупа 1
	(+)2	передний фронт измерительного щупа 2
	-2	задний фронт измерительного щупа 2
	Примечание: Если процесс измерения выполняется с двумя измерительными системами, то можно запрограммировать не более двух пусковых событий (передний или задний фронт). Для каждого из обоих пусковых событий регистрируются измеренные значения обеих систем измерения.	
<MeasMem>:	Номер FIFO (динамическая память)	

Примеры

Пример 1: Спец. для оси измерение со стиранием остатка пути в режиме 1 (обработка во временной последовательности)

a) измерение с одной измерительной системой

Программный код	Комментарий
...	
N100 **MEASA[X]=(1,1,-1)** G01 X100 F100	; Измерение в режиме 1 с активной измерительной системой. Ожидание сигнала измерения с передним/задним фронтом измерительного щупа 1 на пути перемещения к X=100.
N110 IF $AC_MEA[1]==FALSE GOTOF ENDE	; Проверить успех измерения.
N120 R10=$AA_MM1[X]	; Сохранить относящееся к первому запрограммированному пусковому событию (передний фронт) измеренное значение.
N130 R11=$AA_MM2[X]	; Сохранить относящееся ко второму запрограммированному пусковому событию (задний фронт) измеренное значение.
N140 ENDE:	

b) измерение с двумя измерительными системами

Программный код	Комментарий
...	
N200 **MEASA[X]=(31,1,-1)** G01 X100 F100	; Измерение в режиме 1 с обеими измерительными системами. Ожидание сигнала измерения с передним/задним фронтом измерительного щупа 1 на пути перемещения к X=100.
N210 IF $AC_MEA[1]==FALSE GOTOF ENDE	; Проверить успех измерения.
N220 R10=$AA_MM1[X]	; Сохранить измеренное значение измерительной системы 1 при переднем фронте.

Программный код	Комментарий
N230 R11=$AA_MM2[X]	; Сохранить измеренное значение измерительной системы 2 при переднем фронте.
N240 R12=$AA_MM3[X]	; Сохранить измеренное значение измерительной системы 1 при заднем фронте.
N250 R13=$AA_MM4[X]	; Сохранить измеренное значение измерительной системы 2 при заднем фронте.
N260 ENDE:	

Пример 2: Спец. для оси измерение со стиранием остатка пути в режиме 2 (обработка в запрограммированной последовательности)

Программный код	Комментарий
...	
N100 **MEASA[X]=(2,1,-1,2,-2)** G01 X100 F100	; Измерение в режиме 2 с активной измерительной системой. Ожидание сигнала измерения в последовательности передний фронт измерительного щупа 1, задний фронт измерительного щупа 1, передний фронт измерительного щупа 2, задний фронт измерительного щупа 2 на пути перемещения к X=100.
N110 IF $AC_MEA[1]==FALSE GOTOF MESSTASTER2	; Проверить успех измерения с измерительным щупом 1.
N120 R10=$AA_MM1[X]	; Сохранить относящееся к первому запрограммированному пусковому событию (передний фронт щупа 1) измеренное значение.
N130 R11=$AA_MM2[X]	; Сохранить относящееся ко второму запрограммированному пусковому событию (передний фронт щупа 1) измеренное значение.
N140 MESSTASTER2:	
N150 IF $AC_MEA[2]==FALSE GOTOF ENDE	; Проверить успех измерения с измерительным щупом 2.
N160 R12=$AA_MM3[X]	; Сохранить относящееся к третьему запрограммированному пусковому событию (передний фронт щупа 2) измеренное значение.
N170 R13=$AA_MM4[X]	; Сохранить относящееся к четвертому запрограммированному пусковому событию (передний фронт щупа 2) измеренное значение.
N180 ENDE:	

Пример 3: Спец. для оси непрерывное измерение в режиме 1 (обработка во временной последовательности)

а) Измерение до 100 значений

Программный код	Комментарий
...	
N110 DEF REAL MESSWERT[100]	
N120 DEF INT Schleife=0	

Программный код	Комментарий
N130 **MEAC[X]=(1,1,-1)** G01 X1000 F100	; Измерение в режиме 1 с активной измерительной системой, сохранение измеренных значений в $AC_FIFO1, ожидание сигнала измерения с задним фронтом измерительного щупа 1 на пути перемещения к X=1000.
N135 STOPRE	
N140 MEAC[X]=(0)	; Отменить измерение после достижения позиции оси.
N150 R1=$AC_FIFO1[4]	; Сохранить число полученных измеренных значений в параметре R1.
N160 FOR Schleife=0 TO R1-1	
N170 MESSWERT[Schleife]=$AC_FIFO1[0]	; Выгрузить и сохранить измеренные значения из $AC_FIFO1.
N180 ENDFOR	

b) Измерение со стиранием остатка пути после 10 измеренных значений

Программный код	Комментарий
...	
N10 WHEN $AC_FIFO1[4]>=10 DO MEAC[x]=(0) DELDTG(x)	; Стирание остаточного пути.
N20 **MEAC[x]=(1,1,1,-1)** G01 X100 F500	
N30 MEAC[X]=(0)	
N40 R1=$AC_FIFO1[4]	; Число измеренных значений.
...	

c) Измерение заднего/переднего профиля зуба с 2 измерительными щупами

Программный код	Комментарий
...	
N110 DEF REAL MESSWERT[16]	
N120 DEF INT Schleife=0	
N130 **MEAC[X]=(1,1,-1,2)** G01 X100 F100	; Измерение в режиме 1 с активной измерительной системой, сохранение измеренных значений в $AC_FIFO1, ожидание сигнала измерения в последовательности заднего фронта измерительного щупа 1, переднего фронта измерительного щупа 2, на пути перемещения к X=100.
N140 STOPRE	; Остановка предварительной обработки.
N150 MEAC[X]=(0)	; Отменить измерение после достижения позиции оси.
N160 R1=$AC_FIFO1[4]	; Сохранить число полученных измеренных значений в параметре R1.
N170 FOR Schleife=0 TO R1-1	
N180 MESSWERT[Schleife]=$AC_FIFO1[0]	; Выгрузить и сохранить измеренные значения из $AC_FIFO1.
N190 ENDFOR	

Дополнительная информация

Задание измерения

Программирование задания измерения может осуществляться в программе обработки детали или из синхронного действия. Для каждой оси в один и тот же момент времени может быть активно только одно задание измерения.

Примечание

Подача должна быть согласована с соответствующим заданием измерения.

Для MEASA и MEAWA правильные результаты могут быть обеспечены только для подач, при которых возникает не более одного одинакового и не более 4 различных пусковых события на такт регулятора положения.

При непрерывном измерении с MEAC отношение между тактом интерполятора и тактом регулятора положения не может быть больше, чем 1 : 8.

Пусковое событие

Пусковое событие состоит из номера измерительного щупа и критерия запуска (передний или задний фронт) сигнала измерения.

Для каждого измерения может быть обработано до 4 пусковых событий сработавших измерительных щупов соответственно, т. е. до двух измерительных щупов с двумя измерительными фронтами каждый. Последовательность обработки, а также макс. число пусковых событий, зависят при этом от выбранного режима.

Если процесс измерения выполняется с двумя измерительными системами, то можно запрограммировать не более двух пусковых событий (передний или задний фронт). Для каждого из обоих пусковых событий регистрируются измеренные значения обоих измерительных щупов.

При использовании телеграммы PROFIBUS 391 (стандартная настройка для коммуникации PROFIBUS) возможно только одно измеренное значение на каждое пусковое событие и такт регулятора положения.

При использовании телеграммы PROFIBUS 395 количество измеренных значений на каждое пусковое событие можно при MEAC в режиме работы 1 увеличить, в общей сложности, до 8 измеренных значений при переднем фронте и до 8 при заднем фронте на каждое пусковое событие и такт регулятора:

- Один измерительный щуп: 8 измеренных значений при переднем фронте и 8 при заднем фронте
- Два измерительных щупа: 4 измеренных значения при переднем фронте и 4 при заднем фронте на каждый измерительный щуп

Благодаря этому при использовании телеграммы PROFIBUS 395 можно реализовать более высокие значения подачи или скорости вращения.

Режим работы

С помощью первой цифры (второй десятичный разряд) режима выбирается желаемая измерительная система. Если имеется только одна измерительная система, но запрограммирована вторая, то автоматически используется имеющаяся.

С помощью второй цифры (первый десятичный разряд) выбирается желаемый режим измерения. Тем самым процесс измерения адаптируется к возможностям соответствующей СЧПУ:

- **Режим 1**
  Обработка пусковых событий выполняется во временной последовательности их возникновения.
- **Режим 2**
  Обработка пусковых событий выполняется в запрограммированной последовательности.
- **Режим 3**
  Обработка пусковых событий выполняется в запрограммированной последовательности, но без контроля пускового события 1 при старте.

Измерение с и без стирания остатка пути

При программировании MEASA стирание остатка пути выполняется только после регистрации всех требуемых измеренных значений.

Для специальных задач измерения, в которых в любом случае необходим подвод к запрограммированной позиции, используется MEAWA.

Примечание

MEASA не может быть запрограммирована в синхронных действиях. В качестве эквивалента MEAWA плюс стирание остатка пути можно запрограммировать как синхронное действие.

Если задание измерения запускается с MEAWA из синхронных действий, то измеренные значения доступны только в системе координат станка.

Гео-оси / трансформации

Если спец. для оси измерение должно быть запущено для гео-оси, то такое же задание измерения должно быть явно запрограммировано для всех оставшихся гео-осей. Это же относится и к осям, участвующим в трансформации.

Примеры:
```
N10 MEASA[Z]=(1,1) MEASA[Y]=(1,1) MEASA[X]=(1,1) G0 Z100
```
или
```
N10 MEASA[Z]=(1,1) POS[Z]=100
```

Запрос состояния

Если в программе необходимо определить, был ли измерительный щуп отклонен или контактировал, то это можно запросить через следующие системные переменные:

Системная переменная	Объяснение	Тип данных	Значение	
$A_PROBE[<n>]	Состояние измерительного щупа	INT	0	Измерительный щуп не отклонен.
			1	Измерительный щуп отклонен.

Системная переменная	Объяснение	Тип данных	Значение	
$AC_MEA[<n>]	Состояние контактирования измерительного щупа $AC_MEA[<n>] автоматически сбрасывается в начале измерения.	INT	0	Измерительный щуп не контактировал:
			1	Контакт измерительного щупа произошел (все пусковые события, запрограммированные в кадре измерения, произошли).

<n> = номер измерительного щупа

Примечание

Если измерение запускается из синхронных действий, то $AC_MEA более не обновляется. В этом случае необходимо запросить интерфейсный сигнал ЧПУ/PLC DB31, ... DBX62.3 или равнозначную переменную $AA_MEAACT[<Axis>]:

$AA_MEAACT==1: Измерение активно

$AA_MEAACT==0: Измерение не активно

Ограничение измерительного щупа

Системная переменная $A_PROBE_LIMITED позволяет считывать в программе ЧПУ или синхронном действии состояние ограничения измерительного щупа при использовании телеграммы PROFIBUS 395:

$A_PROBE_LIMITED[<n>] == 0: Ограничение измерительного щупа не активно / сброшено

$A_PROBE_LIMITED[<n>] == 1: Ограничение измерительного щупа активно

<n> = номер измерительного щупа

Результаты измерения для MEASA / MEAWA

Результаты измерения щупом могут быть считаны в программе обработки детали и в синхронных действиях через следующие системные переменные:

Системная переменная	Объяснение
$AA_MM1[<ось>] ... $AA_MM4[<ось>]	Значение, измеренное щупом, при пусковом событии 1 в MCS ... Значение, измеренное щупом, при пусковом событии 4 в MCS
$AA_MW1[<ось>] ... $AA_MW4[<ось>]	Значение, измеренное щупом, при пусковом событии 1 в WCS ... Значение, измеренное щупом, при пусковом событии 4 в WCS

<ось> = измерительная ось

Если задание измерения выполняется с двумя измерительными системами, то каждое из двух возможных пусковых событий регистрируется обеими измерительными системами. В этом случае назначение системных переменных следующее:

$AA_MM1[<ось>]	или	$AA_MW1[<ось>]	Измеренное значение измерительной системы 1 при пусковом событии 1
$AA_MM2[<ось>]	или	$AA_MW2[<ось>]	Измеренное значение измерительной системы 2 при пусковом событии 1
$AA_MM3[<ось>]	или	$AA_MW3[<ось>]	Измеренное значение измерительной системы 1 при пусковом событии 2
$AA_MM4[<ось>]	или	$AA_MW4[<ось>]	Измеренное значение измерительной системы 2 при пусковом событии 2

Непрерывное измерение (MEAC)

В случае MEAC измеренные значения доступны в системе координат станка и помещаются в указанную память FIFO[<n>] (динамическая память). Если для измерения сконфигурировано два измерительных щупа, то измеренные значения второго щупа отдельно помещаются в дополнительно сконфигурированную для этого (настройка через MD) память FIFO[<n>+1].

Память FIFO — это динамическая память, в которой измеренные значения по динамическому принципу вносятся в переменную $AC_FIFO.

Примечание

Содержание FIFO может быть выгружено из динамической памяти только один раз. Для многократного использования измеренных данных они должны быть промежуточно сохранены в данных пользователя.

При превышении числом измеренных значений макс. установленного в машинных данных для памяти FIFO числа, измерение автоматически завершается.

Бесконечное измерение может быть реализовано посредством циклической выгрузки измеренных значений. При этом выгрузка должна осуществляться как минимум также часто, как поступление новых измеренных значений.

Дополнительная информация: Описание функций "Синхронные действия"

Защита от ошибок программирования

Следующие ошибочные программирования определяются и индицируются с ошибкой:

- MEASA/MEAWA вместе с MEAS/MEAW программируются в одном кадре.
  Пример:
  ```
  N01 MEAS=1 MEASA[X]=(1,1) G01 F100 POS[X]=100
  ```
- MEASA/MEAWA с числом параметров <2 или >5.
  Пример:
  ```
  N01 MEAWA[X]=(1) G01 F100 POS[X]=100
  ```

- MEASA/MEAWA с пусковым событием, отличным от 1/ -1/ 2/ -2
  Пример:
  `N01 MEASA[B]=(1,1,3) B100`
- MEASA/MEAWA с неправильным режимом
  Пример:
  `N01 MEAWA[B]=(4,1) B100`
- MEASA/MEAWA с дважды запрограммированным пусковым событием
  Пример:
  `N01 MEASA[B]=(1,1,-1,2,-1) B100`
- MEASA/MEAWA и отсутствующая гео-ось
  Пример:
  `N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) G01 X50 Y50 Z50 F100` ;ГЕО-ось `X/Y/Z`
- Несогласованное задание измерения для гео-осей
  Пример:
  `N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) MEASA[Z]=(1,1,2) G01 X50 Y50 Z50 F100`

См. также

Синхронные действия (Страница 992)

4.7.9 Специальные функции для OEM-пользователя (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829)

Адреса OEM

Значение адресов OEM определяет пользователь OEM. Функция реализуется через компилированные циклы. 5 адресов OEM зарезервировано (`OMA1` ... `OMA5`). Идентификаторы адресов настраиваемые. Адреса OEM разрешены в каждом кадре.

Зарезервированные вызовы G-команд:

Для пользователя OEM зарезервированы следующие вызовы G-команд:

- `OEMIPO1`, `OEMIPO2` (из группы G 1)
- `G810` ... `G819` (G-группа 31)
- `G820` ... `G829` (G-группа 32)

Функция реализуется через компилированные циклы.

Функции и подпрограммы

Дополнительно пользователи OEM могут создавать и предопределенные функции и подпрограммы с передачей параметров.

Примечание

Симуляция детали

При симуляции детали до ПО 4.4 не поддерживаются, от ПО 4.4 поддерживаются только некоторые компилируемые циклы (CC).

Поэтому языковые команды в программе обработки детали из не поддерживаемых компилируемых циклов (`OMA1` ... `OMA5`, `OEMIPO1/2`, `G810` ... `G829`, некоторые процессы и функции) приводят без индивидуальной обработки с сигнализации ошибки и отмене симуляции.

Решение: В индивидуальном порядке обработать отсутствующие CC-спец. языковые элементы в программе обработки детали ($P_SIM-запрос).
Пример:

```
N1 G01 X200 F500
IF (1==$P_SIM)
N5 X300 ;при симуляции CC не активен
ELSE
N5 X300 OMA1=10
ENDIF
```

4.7.10 Уменьшение подачи с замедлением на углах (FENDNORM, G62, G621)

При автоматическом замедлении на углах подача уменьшается колоколообразно непосредственно перед соответствующим углом. Кроме этого, объем релевантных для обработки параметров инструмента может быть спараметрирован через установочные данные.

- Начало и конец уменьшения подачи
- Процентовка, с которой уменьшается подача
- Определение релевантного угла

В качестве релевантных углов учитываются такие углы, внутренний угол которых меньше, чем спараметрированный через установочные данные угол.

При значении по умолчанию FENDNORM функция автоматической угловой процентовки отключается.

Синтаксис

```
FENDNORM
G62 G41
```

```
G621
```

Объяснение

FENDNORM:	Автоматическое замедление на углах выкл
G62:	Замедление **на внутренних углах** при активной коррекции радиуса инструмента
G621:	Замедление **на всех углах** при активной коррекции радиуса инструмента

G62 действует только на внутренних углах с активной коррекцией радиуса инструмента G41/G42 и активным режимом управления траекторией G64/G641.

Подвод к соответствующему углу выполняется с уменьшенной подачей, которая рассчитывается следующим образом:

F * (процентовка для уменьшения подачи) * процентовка подачи

Макс. возможное уменьшение подачи достигается тогда, когда инструмент, относительно траектории центра, должен выполнить изменение направление в соответствующем углу.

G621 действует аналогично G62 на каждом углу, определенных через FGROUP осей.

4.7.11 Программируемый критерий окончания движения (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)

Подобно критерию смены кадра при траекторной интерполяции (G601, G602 и G603), для командных осей/осей PLC в программе обработки детали или в синхронных действиях можно запрограммировать критерий окончания движения для интерполяции отдельной оси.

В зависимости от того, какой критерий окончания движения установлен, кадры программы обработки детали или кадры технологического цикла с движениями отдельных осей завершаются с различной скоростью. Это же относится и к PLC через FC15/16/18.

Синтаксис

```
FINEA[<ось>]
COARSEA[<ось>]
IPOENDA[<ось>]
IPOBRKA(<ось>[,<момент времени>])
ADISPOSA(<ось>[,<режим>,<размер окна>])
```

Значение

FINEA:	Критерий окончания движения: "Точный останов точный"	
	Активность:	модально
COARSEA:	Критерий окончания движения: "Точный останов грубый"	
	Активность:	модально
IPOENDA:	Критерий окончания движения:"Стоп интерполятора"	

	Активность:	модально	
IPOBRKA:	Критерий смены кадра: Рампа торможения		
	Активность:	модально	
ADISPOSA:	Окно допуска для критерия окончания движения		
	Активность:	модально	
<ось>:	Имя оси канала (X, Y, ...)		
<момент времени>:	Момент времени смены кадра, относительно рампы торможения в %: • 100% = начало рампы торможения • 0% = конец рампы торможения, однозначно с IPOENDA		
	Тип:	REAL	
<режим>:	Отношение окна допуска		
	Диапазон значений:	0	Окно допуска не активно
		1	Окном допуска относительно заданной позиции
		2	Окном допуска относительно фактической позиции
	Тип:	INT	
<размер окна>:	Размер окна допуска		
	Тип:	REAL	

Примеры

Пример 1: Критерий окончания движения: "Стоп интерполятора"

Программный код
; переместить позиционирующую ось X на 100, скорость 200 м/мин, ускорение 90%,
; критерий окончания движения: Стоп интерполятора
N110 G01 POS[X]=100 FA[X]=200 ACC[X]=90 IPOENDA[X]
; синхронное действие:
; ВСЕГДА КОГДА: вход 1 установлен
; ТОГДА переместить позиционирующую ось X на 50, скорость 200 м/мин, ускорение 140%,
; критерий окончания движения: Стоп интерполятора
N120 EVERY $A_IN[1] DO POS[X]=50 FA[X]=200 ACC[X]=140 IPOENDA[X]

Пример 2: Критерий смены кадра: "Рампа торможения"

Программный код	Комментарий
	; установка по умолчанию действует
N40 POS[X]=100	; Движение позиционирования X на позицию 100
	Критерий смены кадра: точный останов точный

Программный код	Комментарий
`N20 IPOBRKA(X,100)`	; Критерий смены кадра: "Рампа торможения",
	100% = начало рампы торможения
`N30 POS[X]=200`	; Смена кадра выполняется, как только ось X начинает торможение
`N40 POS[X]=250`	; Ось `X` не продолжает торможение на позицию 200, а продолжает движение на позицию 250.
	Как только ось начинает торможение, происходит смена кадра.
`N50 POS[X]=0`	; Ось `X` тормозит и возвращается на позицию 0.
	Смена кадра выполняется на позиции 0 и при "Точном останове точном"
`N60 X10 F100`	; Ось `X` движется как траекторная ось на позицию 10

Дополнительная информация

Системная переменная для критерия окончания движения

Активный критерий окончания движения может быть считан через системную переменную $AA_MOTEND.

Критерий смены кадра: "Рампа торможения" (IPOBRKA)

Если при активации критерия смены кадра "Рампа торможения" для опционального момента времени смены кадры, но начинает действовать для следующего движения позиционирования и записывается синхронно с главным ходом в установочные данные. Если значение для момента времени смены кадра не указано, то активируется актуальное значение установочных данных.

SD43600 $SA_IPOBRAKE_BLOCK_EXCHANGE

При следующем программировании осевого критерия окончания движения FINEA, COARSEA , IPOENDA) IPOBRKA для соответствующей оси деактивируется.

Дополнительный критерий смены кадра: "Окно допуска" (ADISPOSA)

С ADISPOSA в качестве дополнительного критерия смены кадра можно определить окно допуска вокруг конечной точки кадра (по выбору фактическая или заданная позиция). Тогда для смены кадра должны быть выполнены оба условия:

- Критерий смены кадра: "Рампа торможения"
- Критерий смены кадра: "Окно допуска"

4.8 Трансформации координат (фреймы)

4.8.1 Трансформация координат через фрейм-переменные

Наряду с фрейм-операторами (Страница 316), например, ROT, AROT, SCALE и т. д., систему координат детали (WCS) можно трансформировать и через фрейм-переменные $P_...FR (фреймы УД) и $P_...FRAME (активные фреймы).

На следующем рисунке показана схема структурирования фрейм-переменных:

- Фреймы УД
- Активные фреймы
- Активный общий фрейм: соединение всех активных фреймов
- Глобальные фреймы ЧПУ
- Специфические для канала фреймы

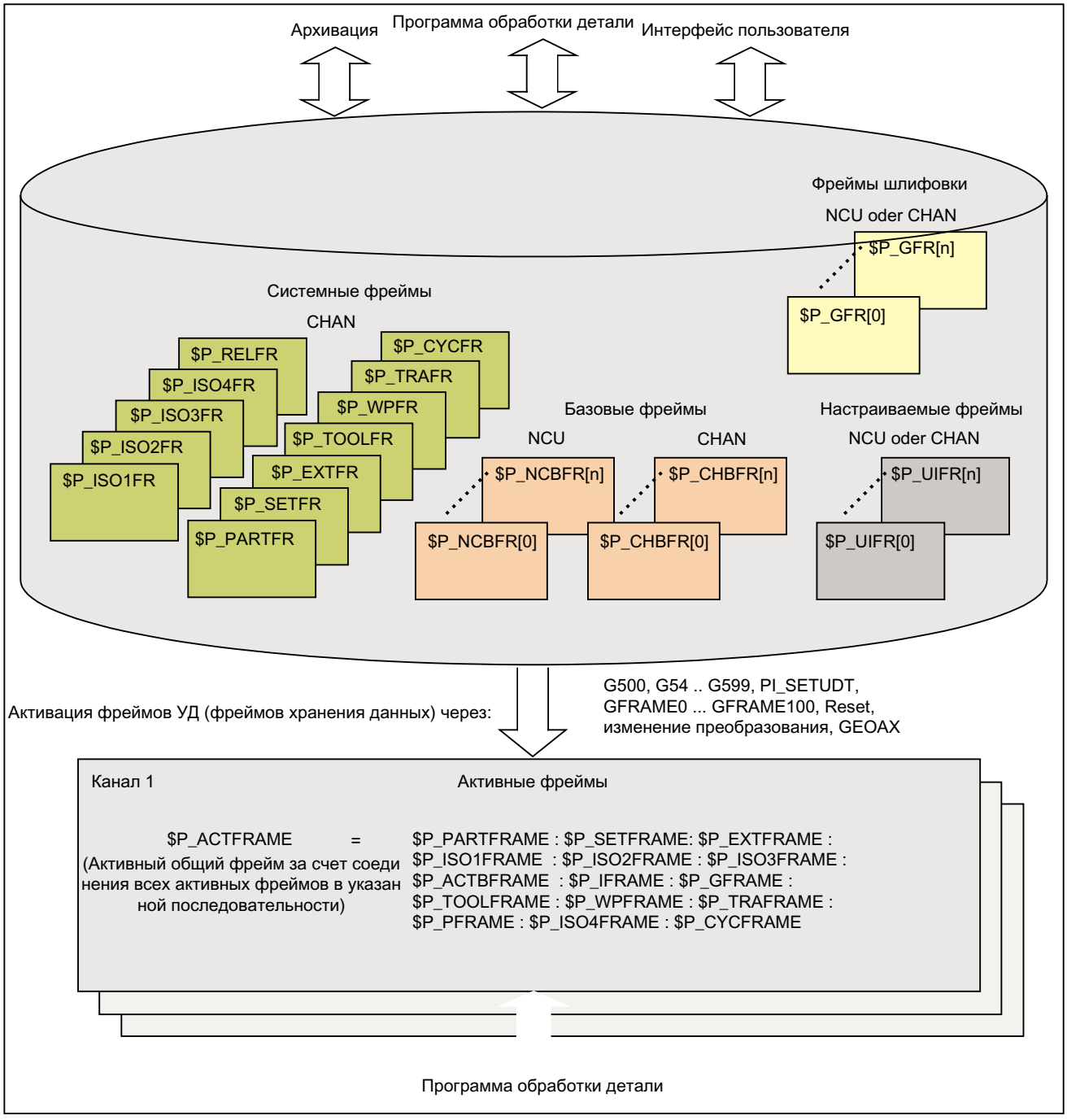


Рисунок 4-1 Обзор фрейм-переменных

4.8.1.1 Предопределенные фрейм-переменные ($P_CHBFRAME, $P_IFRAME, $P_PFRAME, $P_ACTFRAME)

Активно: Специфические для канала фреймы $P_CHBFRAME[<n>] ($P_BFRAME)

Примечание

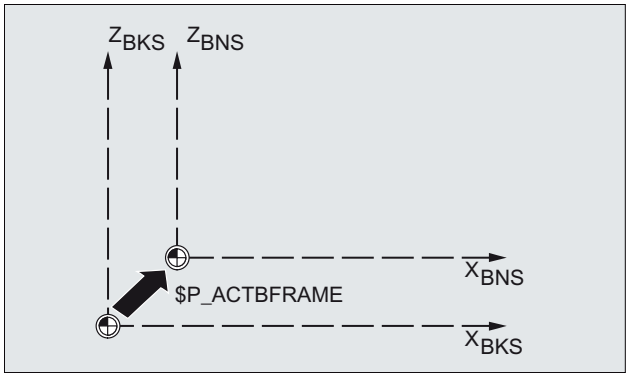
Актуальный базовый фрейм `$P_BFRAME` и базовый фрейм УД `$P_UBFR` сохраняются из соображений совместимости.

- `$P_BFRAME` ≙ `$P_CHBFRAME[0]`
- `$P_UBFR` ≙ `$P_CHBFR[0]`.

Фрейм-переменные $P_CHBFRAME[<n>] определяют отношение между базовой кинематической системой (BKS) и базовой системой нулевой точки (BNS).

Если актуальный специфический для канала базовый фрейм $P_CHBFRAME[`<n>`] должен активироваться в программе ЧПУ немедленно, имеются следующие возможности

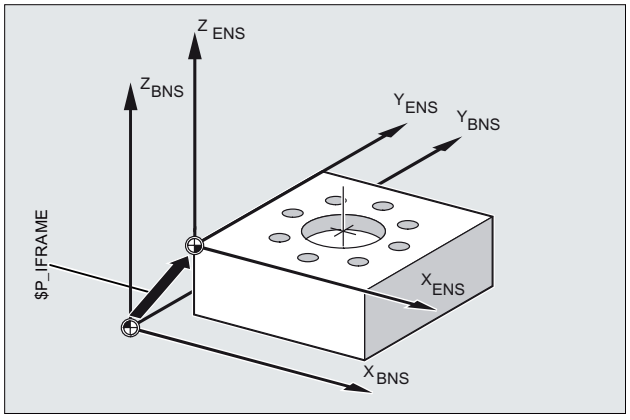
- Команды:
  - `G500` (выключение всех настраиваемых фреймов, базовые фреймы остаются активными)
  - `G54` ... `G599` (настраиваемые смещения нулевой точки)
- Присваивание специфического для канала базового фрейма системы УД актуальному специфическому для канала базовому фрейму:
  `$P_CHBFRAME[<n>] = $P_CHBFR[<m>]`



Активно: специфический для канала фрейм $P_IFRAME

Фрейм-переменная $P_IFRAME, определяет отношение между базовой системой нулевой точки (BNS) и настраиваемой системой нулевой точки (ENS).

- `$P_IFRAME` соответствует `$P_UIFR[$P_IFRNUM]`
- `$P_IFRAME` содержит после программирования, к примеру, `G54` определенные через `G54` смещение, вращение, масштабирование и отражение.

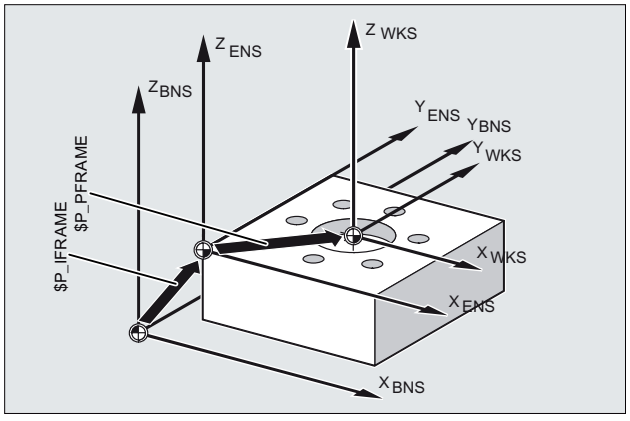


Активно: специфический для канала программируемый фрейм $P_PFRAME

Фрейм-переменная $P_PFRAME, определяет отношение между настраиваемой системой нулевой точки (ENS) и системой координат детали (WCS).

`$P_PFRAME` содержит результирующий фрейм, полученный

- **из программирования** `TRANS/ATRANS`, `ROT/AROT`, `SCALE/ASCALE`, `MIRROR/AMIRROR` или
- **из присваивания** `CTRANS`, `CROT`, `CMIRROR`, `CSCALE` программируемому FRAME



Активно: Общий фрейм $P_ACTFRAME

Действующий в канале общий фрейм, получается из соединения всех действующих в канале фреймов.

```
$P_ACTFRAME =    $P_PARTFRAME : $P_SETFRAME  : $P_EXTFRAME :
                 $P_ISO1FRAME : $P_ISO2FRAME : $P_ISO3FRAME :
                 $P_ACTBFRAME : $P_IFRAME    : $P_GFRAME :
                 $P_TOOLFRAME : $P_WPFRAME   : $P_TRAFRAME :
                 $P_PFRAME    : $P_ISO4FRAME : $P_CYCFRAME
```

`$P_ACTFRAME` описывает нулевую точку актуальной системы координат детали.

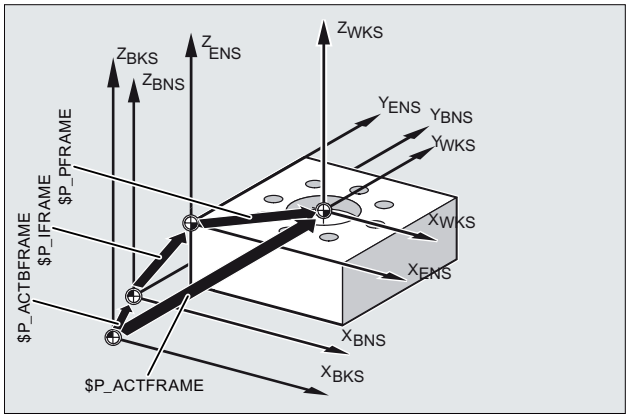
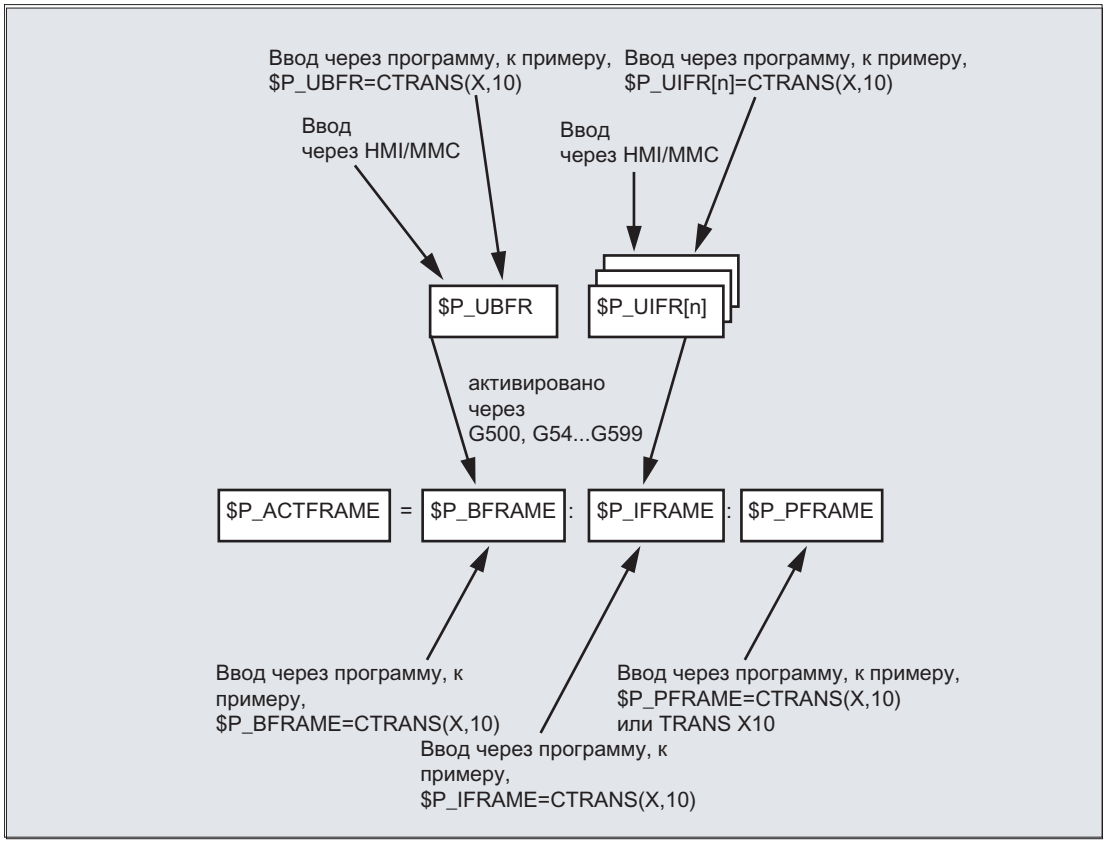


Рисунок 4-2 Фрейм-переменная $P_ACTFRAME

Если один из следующих фреймов `$P_BFRAME` / `$P_CHBFRAME[<n>]`, `$P_IFRAME` или `$P_PFRAME` изменяется, то актуальный общий фрейм `$P_ACTFRAME` пересчитывается заново.



Базовый фрейм и настраиваемый фрейм действуют после Reset, если MD 20110 RESET_MODE_MASK установлены следующим образом:

Бит0=1, Бит14=1 --> `$P_UBFR` (базовый фрейм) действует

Бит0=1, Бит5=1 --> `$P_UIFR[$P_UIFRNUM]` (настр. фрейм) действует

Хранение данных: Специфические для канала базовые фреймы $P_CHBFR[<n>]

Через фрейм-переменные $P_CHBFR[<n>] базовые фреймы в системе УД считываются / записываются. Фрейм УД при записи активируется в канале не сразу. Активация записанного фрейма происходит при следующих условиях:

- Reset канала и MD20110 $MC_RESET_MODE_MASK,Бит0 == 1 и Бит14 == 1
- Команда `G500`, `G54` ... `G57`, `G505` ... `G599` (включение/выключение базовых фреймов с последующим пересчетом актуального общего фрейма)

Хранение данных: Специфические для канала настраиваемые фреймы $P_UIFR[<n>]

Через фрейм-переменные $P_UIFR[<n>] настраиваемые фреймы в системе УД считываются / записываются. Фрейм при записи активируется в канале не сразу. Записанный фрейм учитывается в канале при следующих условиях:

- Команда `G500` (выключение всех настраиваемых фреймов или смещений нулевой точки)
- Команда `G54` ... `G57`, `G505` ... `G599` (включение настраиваемого фрейма или смещения нулевой точки)

Активный настраиваемый фрейм	Фрейм УД	(соответствует команде)
$P_IFRAME =	$P_UIFR[0]	G500
	$P_UIFR[1]	G54
	$P_UIFR[2]	G55
	$P_UIFR[3]	G56
	$P_UIFR[4]	G57
	$P_UIFR[5]	G505
	$P_UIFR[6]	G506
	...	...
	$P_UIFR[99]	G599

4.8.2 Присвоение значений фреймам

4.8.2.1 Присвоение прямых значений (значение оси, угол, масштаб)

В программе ЧПУ возможно прямое присваивание значений фреймам или фрейм-переменным.

Синтаксис

Синтаксис

`$P_PFRAME = CTRANS(X, <значение смещения>, Y, <значение смещения>, Z, <значение смещения>, ...)`

```
$P_PFRAME = ROT(X, <угол>, Y, <угол>, Z, <угол>, ...)
$P_UIFR[..] = CROT(X, <угол>, Y, <угол>, Z, <угол>, ...
$P_PFRAME = CSCALE(X, <масштаб>, Y, <масштаб>, Z, <масштаб>, ...)
$P_PFRAME = CMIRROR(X, Y, Z)
```

Синтаксис для `$P_CHBFRAME[<n>]` идентичен `$P_PFRAME`.

Значение

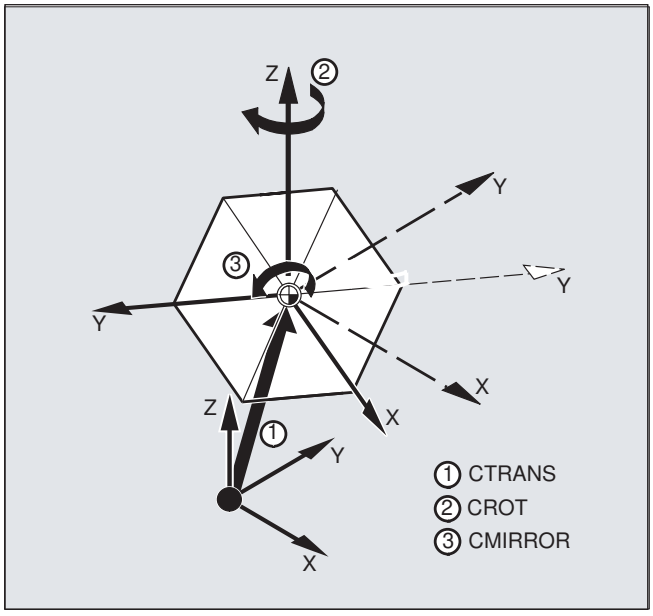
`CTRANS`:	Смещение в указанных осях
`CROT`:	Вращение вокруг указанных осей
`CSCALE`:	Изменение масштаба в указанных осях
`CMIRROR`:	Реверсирование указанной оси
`X, Y, Z`:	Значение смещения в направлении указанной гео-оси
<значение смещения>:	Значение смещения
<угол>:	Угол, на который выполняется поворот
<масштаб>:	Указание масштаба

Примеры

Присваивание значений компонентам актуального программируемого фрейма

Присваивание значений компонентам смещения, поворота и отражения актуального программируемого фрейма:

```
$P_PFRAME = CTRANS(X,10,Y,20,Z,5) : CROT(Z,45) : CMIRROR(y)
```



Запись компонентов поворота фрейма

Присваивание значений всем трем осям компонента поворота настраиваемого фрейма УД $P_UIFR с помощью CROT :

```
$P_UIFR[5]=CROT (X, 0, Y, 0, Z, 0)
```

В качестве альтернативы этому прямое присваивание отдельных значений соответствующим осям компонента поворота фрейма УД:

```
$P_UIFR[5, Y, RT]=0
$P_UIFR[5, X, RT]=0
$P_UIFR[5, Z, RT]=0
```

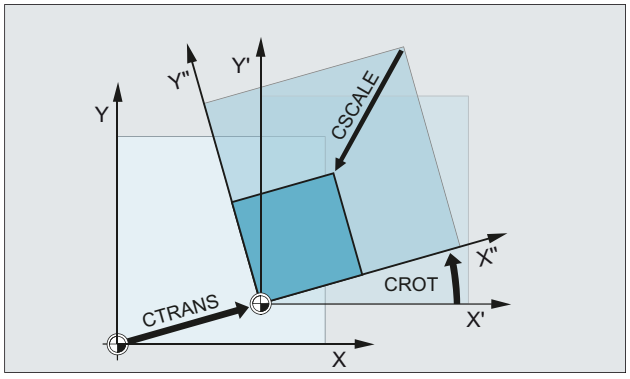
Описание

Несколько операций на одном фрейме можно соединить через связывающий оператор : друг с другом. В этом случае операции выполняются последовательно слева направо.

Пример

Связанные операции на $P_PFRAME со смещением, поворотом и масштабированием:

```
$P_PFRAME = CTRANS(...) : CROT(...) : CSCALE...
```



4.8.2.2 Чтение и изменение компонентов фрейма (TR, FI, RT, SC, MI)

Существует возможность обращения к **отдельным** данным фрейма, к примеру, к определенному значению смещения или углу поворота. Эти значения могут изменяться или присваиваться другой переменной.

Синтаксис

`R10=$P_UIFR[$P_UIFNUM,X,RT]`	Угол поворота RT вокруг оси X из актуального действующего настраиваемого смещения нулевой точки $P_UIFRNUM должен быть присвоен переменной R10.
`R12=$P_UIFR[25,Z,TR]`	Значение смещения TR в Z из блока данных установленного фрейма Nr. 25 должно быть присвоено переменной R12.
`R15=$P_PFRAME[Y,TR]`	Значение смещения TR в Y актуального программируемого фрейма должно быть присвоено переменной R15.
`$P_PFRAME[X,TR] = 25`	Значение смещения TR в X актуального программируемого фрейма должно быть изменено. С этого момента действует X25.

Значение

`$P_UIFRNUM`:	С помощью этой переменной автоматически создается связь с актуальным действительным настраиваемым смещением нулевой точки.
`P_UIFR[n,…,…]`:	Посредством указания номера фрейма `n` осуществляется обращение к настраиваемому фрейму `Nr. n`.
	Указание компонента, который должен быть считан или изменен:
`TR`:	`TR` смещение
`FI`:	`FI` смещение точное
`RT`:	`RT` вращение
`SC`:	`SC Scale` изменение масштаба
`MI`:	`MI` отражение
`X, Y, Z`:	Дополнительно (см. примеры) указывается соответствующая ось `X, Y, Z`.

Диапазон значений для вращения RT

Вращение вокруг 1-й гео-оси: -180° до +180°
Вращение вокруг 2-й гео-оси: -90° до +90°
Вращение вокруг 3-й гео-оси: -180° до +180°

Описание

Вызов фрейма

Через указание системной переменной $P_UIFRNUM можно напрямую обращаться к актуальному, установленному с помощью $P_UIFR или G54, G55, ... смещению нулевой точки
($P_UIFRNUM содержит номер актуального установленного фрейма).

Все другие сохраненные настраиваемые фреймы $P_UIFR вызываются через указание соответствующего номера $P_UIFR[n].

Для предопределенных фрейм-переменных и определенных пользователем фреймов указывается имя, к примеру, $P_IFRAME.

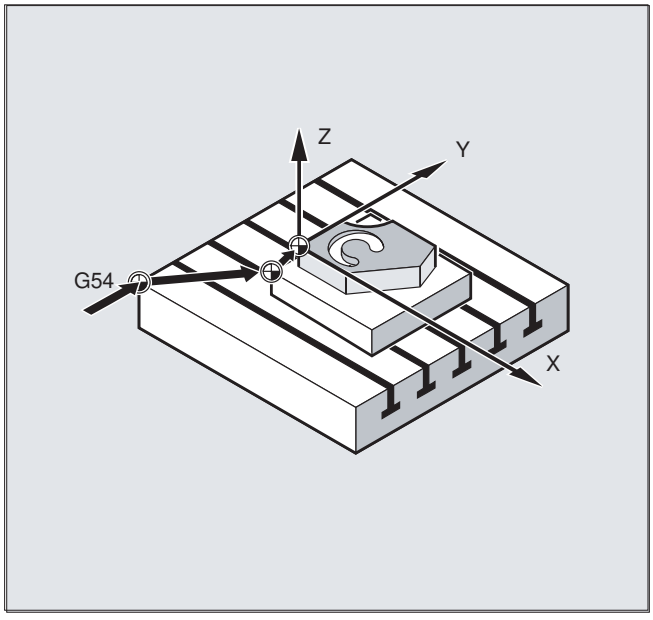
Вызов данных

В квадратных скобках стоят имя оси и компоненты фрейма значения, к которому выполняется обращение или которое должно быть изменено, к примеру, [X, RT] или [Z, MI].

4.8.2.3 Вычисление с фреймами

В программе ЧПУ возможно присвоение одного фрейма другому фрейму или соединение фреймов друг с другом.

Соединения фреймов подходят, к примеру, для описания нескольких деталей, расположенных на одной палете, которые должны быть обработаны в течение одного технологического процесса.



Для описания задач палет компоненты фрейма могут содержать, к примеру, только определенные частные значения, посредством соединения которых создаются различные нулевые точки детали.

Примеры

Присваивания

Программный код	Комментарий
`DEF FRAME EINSTELLUNG_1`	; Определение локальной фрейм-переменной
`EINSTELLUNG_1 = CTRANS(X,10)`	; Присваивание результата функции фрейм-переменной

Программный код	Комментарий
`$P_PFRAME = EINSTELLUNG_1`	; Присваивание фрейм-переменной актуальному фрейму
`DEF FRAME EINSTELLUNG_4`	; Определение локальной фрейм-переменной
`EINSTELLUNG_4 = $P_PFRAME`	; Буферизация актуального фрейма во фрейм-переменной
...	
`$P_PFRAME = EINSTELLUNG_4`	; Эхо-считывание актуального фрейма из фрейм-переменной

Соединения

Фреймы соединяются друг с другом в запрограммированной последовательности с помощью оператора : . Компоненты фрейма, например смещения, повороты и т. д., выполняются друг за другом аддитивно.

Программный код	Комментарий
`$P_IFRAME = $P_UIFR[15] : $P_UIFR[16]`	; Присваивание фрейма результата из соединения ; двух настраиваемых фреймов УД активному ; настраиваемому общему фрейму. ; Пример использования: ; `$P_UIFR[15]`: Смещение ; `$P_UIFR[16]`: Вращение
`$P_UIFR[3] = $P_UIFR[4] : $P_UIFR[5]`	; Присваивание фрейма результата из соединения ; двух настраиваемых фреймов УД какому-либо ; другому настраиваемому фрейму УД

4.8.2.4 Определение фрейм-переменных (DEF FRAME)

Наряду с предопределенными фрейм-переменными можно определять и собственные фрейм-переменные. Самостоятельно определяемые фрейм-переменные – это переменные пользователя типа FRAME. Имя фрейму можно давать произвольно в рамках правил работы с переменными пользователя.

С помощью функций CTRANS, CROT, CSCALE, CMIRROR самостоятельно определяемым фрейм-переменным можно присваивать значения.

Синтаксис

```
DEF FRAME <имя>
```

Значение

`DEF FRAME`:	Определить переменные пользователя типа FRAME.
<имя>:	Имя фрейм-переменной

Пример

Определение фрейм-переменной "PALETTE" и присваивание значений смещения и поворота:

Программный код	Комментарий
`DEF FRAME PALETTE`	; Определить фрейм-переменную `PALETTE`
`PALETTE = CTRANS(...) : CROT(...)`	; Присваивание фрейма результата соединения
	; смещения и поворота фрейм-переменной `PALETTE`

4.8.3 Грубое и точное смещение (CTRANS, CFINE)

Точное смещение

Точное смещение `CFINE(...)` можно применять к следующим фреймам:

- Настраиваемые фреймы: $P_UIFR или $P_IFRAME
- Базовые фреймы: $P_NCBFR[<n>], $P_CHBFR[<n>] или $P_CHBFRAMES[<n>] или $P_ACTBFRAME
- Программируемый фрейм: $P_PFRAME

Точное смещение фрейма программируется с командой `CFINE(...)`.

Грубое смещение

Грубое смещение `CTRANS(...)` можно применять ко всем фреймам.

Общее смещение

Общее смещение получается из сложения грубого и точного смещений.

Машинные данные

Разрешение точного смещения

Точное смещение разрешается с помощью машинных данных:

MD18600 $MN_MM_FRAME_FINE_TRANS = 1

Синтаксис

Точное смещение

- Общий фрейм
  - <фрейм> = CFINE(<K_1>,<значение>)
  - <фрейм> = CFINE(<K_1>,<значение>, <K_2>, <значение>)
  - <фрейм> = CFINE(<K_1>,<значение>, <K_2>, <значение>, <K_3>, <значение>)
- Фрейм-компонент
  - <фрейм>[<n>, <K_1>, FI] = <значение>

Грубое смещение

- Общий фрейм
  - <фрейм> = CTRANS(<K_1>,<значение>)
  - <фрейм> = CTRANS(<K_1>,<значение>, <K_2,<значение>)
  - <фрейм> = CTRANS(<K_1>,<значение>, <K_2,<значение>, <K_3,<значение>)
- Фрейм-компонент
  - <фрейм>[<n>,<K_1>,TR] = <значение>

Специально для программируемого фрейма $P_PFRAME:

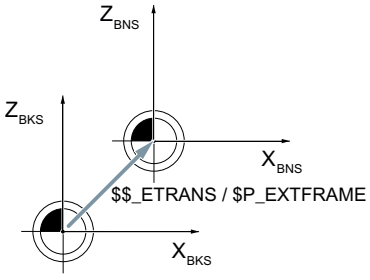
- TRANS <K_1> <значение>
- TRANS <K_1> <значение> <K_2> <значение>
- TRANS <K_1> <значение> <K_2> <значение> <K_3> <значение>

Значение

<фрейм>:	Фрейм, например настраиваемый фрейм системы УД $P_UIFR[<n>]
CFINE:	Точное смещение, аддитивное смещение.
CTRANS:	Грубое смещение, абсолютное смещение
TRANS:	Только программируемый фрейм: Грубое смещение, абсолютное смещение
<K_n>:	Оси координат X, Y, Z
<значение>:	Значение смещения

4.8.4 Внешнее смещение нулевой точки ($AA_ETRANS)

Внешнее смещение нулевой точки – это линейное смещение между базовой кинематической системой (BKS) и базовой системой нулевой точки (BNS).



Внешнее смещение нулевой точки с помощью $AA_ETRANS действует двумя способами в зависимости от параметрирования машинных данных:

1. Системная переменная $AA_ETRANS после активации интерфейсным сигналом ЧПУ/PLC действует непосредственно как значение смещения
2. Значение системной переменной $AA_ETRANS после активации интерфейсным сигналом ЧПУ/PLC принимается в активные системные фреймы $P:EXTFRAME и во фрейм УД $P_EXTFR. Затем активный общий фрейм $P_ACTFRAME пересчитывается заново.

Машинные данные

При использовании системной переменной $AA_ETRANS следует различать два способа действий, которые выбираются через следующие машинные данные:

MD28082 $MC_MM_SYSTEM_FRAME_MASK,бит1 = <значение>

<значение>	Объяснение
0	Функция: Прямая запись $AA_ETRANS[<ось>] через PLC, HMI или программу ЧПУ. Разрешение на вывод смещения нулевой точки из $AA_ETRANS[<ось>] в ближайшем кадре перемещения: DB31, ...DBX3.0
1	Функция: Активация активного системного фрейма $P:EXTFRAME и фрейма УД $P_EXTFR Разрешение на вывод смещения нулевой точки из $AA_ETRANS[<ось>] через: DB31, ... DBX3.0. После этого в канале происходит: • Остановка всех движений перемещения в канале (кроме командных осей/осей PLC) • Остановка предварительной обработки с последующей реорганизацией (STOPRE) • Грубое смещение активного фрейма $P_EXTFRAME[<ось>] = $AA_ETRANS[<ось>] • Грубое смещение фрейма УД $P_EXTFR[<ось>] = $AA_ETRANS[<ось>] • Пересчет активного общего фрейма $P_ACTFRAME • Вывод смещения в запрограммированных осях. • Продолжение прерванного движения перемещения или программы ЧПУ

Программирование

- Синтаксис
  `$AA_ETRANS[`<ось>`]` = <значение>
- Объяснение

`$AA_ETRANS`:	Системная переменная для буферизации внешнего смещения нулевой точки
<ось>:	Ось канала
<значение>:	Значение смещения

Интерфейсный сигнал ЧПУ/PLC

DB31, ... DBX3.0 = 0 → 1 ⇒ $P_EXTFRAME[<ось>] = $P_EXTFR[<ось>] = $AA_ETRANS[<ось>]

4.8.5 Установка фактического значения с потерей состояния реферирования (PRESETON)

Процедура PRESETON() устанавливает для одной или нескольких осей новое фактическое значение в системе координат станка (MCS). Это соответствует смещению нулевой точки MCS оси. При этом ось не перемещается.

С помощью PRESETON вызывается остановка предварительной обработки с синхронизацией. Фактическая позиция присваивается оси только в состоянии покоя.

Если ось при PRESETON не согласована с каналом, дальнейшие действия зависят от специфического для оси конфигурирования режима перехода оси:

MD30552 $MA_AUTO_GET_TYPE

Состояние реферирования

При установке нового фактического значения в системе координат станка состояние реферирования оси станка сбрасывается:

DB31, ... DBX60.4 / .5 = 0 (Реферирует / синхронизирует измерительную систему 1 / 2)

Рекомендуется использовать PRESETON только для осей без обязательной референтной точки.

Для восстановления первоначальной системы координат станка необходимо заново реферировать измерительную систему оси станка, например, через реферирование из программы обработки детали (G74).

⚠ ОСТОРОЖНО

Потеря состояния реферирования

При установке нового фактического значения в системе координат станка с помощью PRESETON состояние реферирования оси станка сбрасывается на "не референцировано / синхронизировано".

Программирование

Синтаксис
PRESETON(<ось_1>, <значение_1> [, <ось_2>, <значение_2>, ... <ось_8>, <значение_8>])

Объяснение

PRESETON:	Установка фактического значения с потерей состояния реферирования	
	Остановка предварительной обработки:	да
	Автономно в кадре:	да
<ось_x>:	Имя оси станка	
	Тип:	AXIS
	Диапазон значений:	определенные в канале имена осей станка
<значение_x>:	Новое фактическое значение оси станка в системе координат станка (MCS) Ввод осуществляется в актуальной действующей системе единиц (дюймовой / метрической) Учитывается активное программирование диаметра (DIAMON)	
	Тип:	REAL

Дополнительная информация

PRESETON в программе ЧПУ

Подробное описание PRESETON в программах ЧПУ см.: Описание функций – Основные функции.

PRESETON в синхронных действиях

Подробное описание PRESETON в синхронных действиях см.: Описание функций – Синхронные действия

4.8.6 Установка фактического значения без потери состояния реферирования (PRESETONS)

Процедура PRESETONS() устанавливает для одной или нескольких осей новое фактическое значение в системе координат станка (MCS). Это соответствует смещению нулевой точки MCS оси. При этом ось не перемещается.

С помощью PRESETONS вызывается остановка предварительной обработки с синхронизацией. Фактическая позиция присваивается оси только в состоянии покоя.

Если ось при PRESETONS не согласована с каналом, дальнейшие действия зависят от специфического для оси конфигурирования режима перехода оси:

MD30552 $MA_AUTO_GET_TYPE

Состояние реферирования

При установке нового фактического значения в системе координат станка (MKS) с помощью PRESETONS состояние реферирования оси станка **не** изменяется.

Условия

- **Тип датчика**
  PRESETONS можно использовать только при следующих типах датчика активной измерительной системы:
  - MD30240 $MA_ENC_TYPE[<измерительная система>] = 0 (Симулированный датчик)
  - MD30240 $MA_ENC_TYPE[<измерительная система>] = 1 (Датчик необработанного сигнала)
- **Режим реферирования**
  PRESETONS можно использовать только при следующих режимах реферирования активной измерительной системы:
  - MD34200 $MA_ENC_REFP_MODE[<измерительная система>] = 0 (Реферирование невозможно)
  - MD34200 $MA_ENC_REFP_MODE[<измерительная система>] = 1 (Реферирование инкрементных, ротационных или линейных измерительных систем: нулевой импульс в канале датчика)

Программирование

Синтаксис
`PRESETONS`(<ось_1>, <значение_1> [, <ось_2>, <значение_2>, ... <ось_8>, <значение_8>])

Объяснение

`PRESETONS`:	Установка фактического значения без потери состояния реферирования	
	Остановка предварительной обработки:	да
	Автономно в кадре:	да
<ось_x>:	Имя оси станка	
	Тип:	AXIS
	Диапазон значений:	определенные в канале имена осей станка
<значение_x>:	Новое актуальное фактическое значение оси станка в системе координат станка (MCS) Ввод осуществляется в активной системе единиц (дюймовой / метрической) Учитывается активное программирование диаметра (DIAMON)	
	Тип:	REAL

Дополнительная информация

PRESETONS в программе ЧПУ

Подробное описание PRESETONS в программах ЧПУ см.: Описание функций – Основные функции.

PRESETONS в синхронных действиях

Подробное описание PRESETONS в синхронных действиях см.: Описание функций – Синхронные действия

4.8.7 Вычисление фрейма из 3 точек измерения в пространстве (MEAFRAME)

Функция MEAFRAME используется для поддержки измерительных циклов. Она вычисляет фрейм из трех идеальных и соответствующих измеренных точек.

При позиционировании детали для обработки ее позиция относительно декартовой системы координат станка касательно ее идеальной позиции по умолчанию как смещена, так и повернута. Для точной обработки или измерения необходима либо дорогостоящая физическая юстировка, либо изменение движений в программе обработки детали.

Фрейм может быть определен через сканирование трех точек в пространстве, идеальные позиции которых известны. Сканирование выполняется посредством контактного или оптического сенсора, который касается точно зафиксированных на несущей пластине отверстий или измерительных сфер.

Синтаксис

`MEAFRAME` (<идеальные точки>,<точки измерения>,<качество>)

Объяснение

`MEAFRAME`:	Вызов функции
<идеальные точки>:	2-мерное REAL-поле, содержащее три координаты идеальных точек
<точки измерения>:	2-мерное REAL-поле, содержащее три координаты измеренных точек

<качество>:	Переменная, возвращающая информацию о качестве вычисления ФРЕЙМА		
	Тип:	VAR REAL	
	Значение:	-1	Идеальные точки располагаются практически на одной прямой: Не удалось вычислить фрейм. Возвращенная ФРЕЙМ-переменная содержит нейтральный фрейм.
		-2	Точки измерения располагаются практически на одной прямой: Не удалось вычислить фрейм. Возвращенная ФРЕЙМ-переменная содержит нейтральный фрейм.
		-4	Вычисление матрицы вращения не удалось по иной причине.
		≥ 0.0	Сумма искажений (расстояния между точками), необходимая для перевода измеренного треугольника в конгруэнтный идеальному треугольник.

Примечание

Качество измерения

Для того, чтобы измеренные координаты могли бы быть согласованы с идеальными координатами с комбинированным вращением/смещением, образованный точками измерения треугольник должен быть конгруэнтен идеальному треугольнику. Это осуществляется посредством алгоритма компенсации, минимизирующего сумму квадратов погрешностей, преобразующих измеренный треугольник в идеальный.

Эффективное необходимое искажение точек измерения может служить индикатором качества измерения и поэтому выводится как дополнительная переменная MEAFRAME.

Примечание

Созданный MEAFRAME фрейм через функцию ADDFRAME может быть преобразован в другой фрейм цепочки фреймов (см. пример "Соединение с ADDFRAME").

Примеры

Пример 1:

Программа обработки детали 1:

Программный код

```
...
DEF FRAME CORR_FRAME
```

Установка точек измерения:

Программный код	Комментарий
DEF REAL IDEAL_POINT[3,3]= SET(10.0,0.0,0.0,0.0,10.0,0.0,0.0,0.0,10.0)	
DEF REAL MEAS_POINT[3,3]= SET(10.1,0.2,-0.2,-0.2,10.2,0.1,-0.2,0.2,9.8)	; Для теста.
DEF REAL FIT_QUALITY=0	
DEF REAL ROT_FRAME_LIMIT=5	; Макс. разрешенный поворот позиции детали на 5 градусов.
DEF REAL FIT_QUALITY_LIMIT=3	; Разрешено макс. смещение в 3 мм между идеальным и измеренным треугольником.
DEF REAL SHOW_MCS_POS1[3]	
DEF REAL SHOW_MCS_POS2[3]	
DEF REAL SHOW_MCS_POS3[3]	

Программный код	Комментарий
N100 G01 G90 F5000	
N110 X0 Y0 Z0	
N200 CORR_FRAME=MEAFRAME(IDEAL_POINT,MEAS_POINT,FIT_QUALITY)	
N230 IF FIT_QUALITY < 0	
SETAL(65000)	
GOTOF NO_FRAME	
ENDIF	
N240 IF FIT_QUALITY > FIT_QUALITY_LIMIT	
SETAL(65010)	
GOTOF NO_FRAME	
ENDIF	
N250 IF CORR_FRAME[X,RT] > ROT_FRAME_LIMIT	; Ограничение 1-го угла RPY
SETAL(65020)	
GOTOF NO_FRAME	
ENDIF	
N260 IF CORR_FRAME[Y,RT] > ROT_FRAME_LIMIT	; Ограничение 2-го угла RPY.
SETAL(65021)	
GOTOF NO_FRAME	
ENDIF	
N270 IF CORR_FRAME[Z,RT] > ROT_FRAME_LIMIT	; Ограничение 3-го угла RPY
SETAL(65022)	
GOTOF NO_FRAME	
ENDIF	
N300 $P_IFRAME=CORR_FRAME	; Активировать фрейм выборки с устанавливаемым фреймом. ; Проверить фрейм через позиционирование гео-осей на идеальные точки.

Программный код	Комментарий
N400 X=IDEAL_POINT[0,0] Y=IDEAL_POINT[0,1] Z=IDEAL_POINT[0,2]	
N410 SHOW_MCS_POS1[0]=$AA_IM[X]	
N420 SHOW_MCS_POS1[1]=$AA_IM[Y]	
N430 SHOW_MCS_POS1[2]=$AA_IM[Z]	
N500 X=IDEAL_POINT[1,0] Y=IDEAL_POINT[1,1] Z=IDEAL_POINT[1,2]	
N510 SHOW_MCS_POS2[0]=$AA_IM[X]	
N520 SHOW_MCS_POS2[1]=$AA_IM[Y]	
N530 SHOW_MCS_POS2[2]=$AA_IM[Z]	
N600 X=IDEAL_POINT[2,0] Y=IDEAL_POINT[2,1] Z=IDEAL_POINT[2,2]	
N610 SHOW_MCS_POS3[0]=$AA_IM[X]	
N620 SHOW_MCS_POS3[1]=$AA_IM[Y]	
N630 SHOW_MCS_POS3[2]=$AA_IM[Z]	
N700 G500	; Деактивировать устанавливаемый фрейм, так как предустановка с нулевым фреймом (значение не внесено).
No_FRAME	; Деактивировать устанавливаемый фрейм, так как предустановка с нулевым фреймом (значение не внесено).
M0	
M30	

Пример 2: Соединение фреймов

Соединение MEAFRAME для коррекций

Функция MEAFRAME выводит фрейм коррекции. Если этот фрейм коррекции связывается с настраиваемым фреймом $P_UIFR[1], который был активен при вызове функции (например, G54), то получается настраиваемый фрейм для дальнейших пересчетов для перемещения или обработки.

Соединение с ADDFRAME

Если этот фрейм коррекции должен действовать в цепочке фреймов на другом месте или если перед настраиваемым фреймом есть другие активные фреймы, то функция ADDFRAME для включения в цепочку может использоваться в одном из базовых фреймов канала или системном фрейме.

При этом во фреймах не должны быть активны:

- Отражение с MIRROR
- Масштабирование с SCALE

Входные параметры для заданных и фактических значений стоят в координатах детали. В основной системе СЧПУ эти координаты всегда должны указываться метрическими или дюймовыми (G71/G70) и как относящийся к радиусу (DIAMOF) размер.

Дополнительная информация по ADDFRAME: Описание функций – Основные функции.

4.8.8 Глобальные фреймы

Глобальные фреймы существуют для каждого ЧПУ только в одном экземпляре для всех каналов. Глобальные фреймы могут записываться и считываться из всех каналов. Активация глобальных фреймов осуществляется в соответствующем канале.

Посредством глобальных фреймов возможно масштабирование и отражение **осей канала и осей станка** со смещениями.

Геометрические связи и цепочки фреймов

Для глобальных фреймов не существует геометрической связи между осями. Поэтому выполнение вращений и программирование идентификаторов гео-осей невозможно.

К глобальным фреймам не могут применяться вращения. Программирование вращения отклоняется с аварийным сообщением 18310 "Канал %1 Кадр %2: вращение недопустимо".

Соединение глобальных фреймов и спец. для канала фреймов возможно. Результирующий фрейм содержит все компоненты фрейма вкл. вращения для всех осей. Присваивание фрейма с компонентами вращения глобальному фрейму отклоняется с ошибкой "Фрейм: вращение недопустимо".

Глобальные фреймы

Глобальные базовые фреймы $P_NCBFR[n]

Можно сконфигурировать до 8 глобальных базовых фреймов:

Одновременно возможно наличие спец. для канала базовых фреймов.

Глобальные базовые фреймы могут записываться и считываться из всех каналов ЧПУ. При записи глобальных фреймов пользователь должен обеспечить координацию каналов. Это может быть реализовано, к примеру, с помощью меток ожидания (WAITMC).

Примечание

Изготовитель станка

Число глобальных базовых фреймов проектируется через машинные данные.

Дополнительная информация: Описание функций – Основные функции

Настраиваемые фреймы $P_UIFR[n]

Все настраиваемые фреймы G500, G54...G599 можно конфигурировать глобально или спец. для канала.

Примечание

Изготовитель станка

Все настраиваемые фреймы с помощью машинных данных MD18601 $MN_MM_NUM_GLOBAL_USER_FRAMES могут быть перепроектированы в глобальные фреймы.

В качестве идентификаторов осей для программных команд фреймов могут использоваться идентификаторы осей канала и идентификаторы осей станка. Программирование идентификаторов гео-осей отклоняется с ошибкой.

4.8.8.1 Специфические для канала фреймы ($P_CHBFR, $P_UBFR)

Настраиваемые фреймы или базовые фреймы могут записываться и считываться через программу обработки детали и через BTSS с панели оператора и с PLC.

Точное смещение возможно и для глобальных фреймов. Подавление глобальных фреймов, как и спец. для канала фреймов, осуществляется через G53, G153, SUPA и G500.

Изготовитель станка

Через машинные данные MD28081 $MC_MM_NUM_BASE_FRAMES можно спроектировать число базовых фреймов в канале. Стандартная конфигурация такова, что существует мин. один базовый фрейм на канал. Возможно макс. 8 базовых фреймов на канал. В дополнение к 8 базовым фреймам в канале может существовать еще 8 глобальных базовых фреймов ЧПУ.

Специфические для канала фреймы

$P_CHBFR[n]

Через системную переменную $P_CHBFR[n] возможно чтение и запись базовых фреймов. При записи базового фрейма связанный общий базовый фрейм не активируется, а активация происходит только при выполнении оператора G500, G54 ... G599. Переменная служит в первую очередь памятью для процессов записи в базовый фрейм с HMI или PLC. Эти фрейм-переменные сохраняются через архивацию данных.

Первый базовый фрейм в канале

Запись в предопределенную переменную $P_UBFR активирует базовый фрейм с индексом поля 0 не сразу же, а активация выполняется только при выполнении оператора G500, G54 ... G599. Запись и чтение переменной возможно и в программе.

$P_UBFR

$P_UBFR идентична $P_CHBFR[0]. Стандартно всегда имеется один базовый фрейм в канале, поэтому системная переменная является совместимой с более старыми версиями. Если спец. для канала базовый фрейм отсутствует, то при записи или чтении выводится аварийное сообщение "Фрейм: недопустимый оператор".

4.8.8.2 Действующие в канале фреймы

Действующие в канале фреймы вводятся из программы обработки детали через соответствующие системные переменные этих фреймов. К ним относятся и системные фреймы. Через эти системные переменные возможно чтение и запись актуального системного фрейма в программе обработки детали.

Актуальные действующие в канале фреймы

Обзор

Актуальные системные фреймы	для:
$P_PARTFRAME	TCARR и PAROT
$P_SETFRAME	Установка фактического значения и касание
$P_EXTFRAME	Внешнее смещение нулевой точки
$P_NCBFRAME[n]	Актуальные глобальные базовые фреймы
$P_CHBFRAME[n]	Актуальные базовые фреймы канала
$P_BFRAME	Актуальный 1-й базовый фрейм в канале
$P_ACTBFRAME	Общий базовый фрейм
$P_CHBFRMASK и $P_NCBFRMASK	Общий базовый фрейм
$P_IFRAME	Актуальный настраиваемый фрейм
Актуальные системные фреймы	для:
$P_TOOLFRAME	TOROT и TOFRAME
$P_WPFRAME	Исходные точки детали
$P_TRAFRAME	Трансформации
$P_PFRAME	Актуальный программируемый фрейм
Актуальный системный фрейм	для:
$P_CYCFRAME	Циклы
P_ACTFRAME	Актуальный общий фрейм
Цепочка фреймов	Актуальный фрейм состоит из общего базового фрейма

$P_NCBFRAME [n] Актуальные глобальные базовые фреймы

Через системную переменную $P_NCBFRAME[n] возможно чтение и запись элементов поля актуальных глобальных базовых фреймов. Результирующий общий базовый фрейм через процесс записи учитывается в канале.

Измененный фрейм активируется только в том канале, в котором он был запрограммирован. Если необходимо изменить фрейм для всех каналов ЧПУ, то необходима одновременная запись в $P_NCBFR[n] и $P_NCBFRAME[n]. После прочие каналы еще должны активировать фрейм, к примеру, с G54. При записи базового фрейма общий базовый фрейм вычисляется заново.

$P_CHBFRAME[n] Актуальные базовые фреймы канала

Через системную переменную $P_CHBFRAME[n] возможно чтение и запись элементов поля базовых фреймов канала. Результирующий общий базовый фрейм через процесс записи учитывается в канале. При записи базового фрейма общий базовый фрейм вычисляется заново.

$P_BFRAME Актуальный 1-й базовый фрейм в канале

Через предопределенную фрейм-переменную $P_BFRAME актуальный базовый фрейм с индексом поля 0, действующий в канале, может быть считан и записан в программе обработки детали. Записанный базовый фрейм сразу же учитывается.

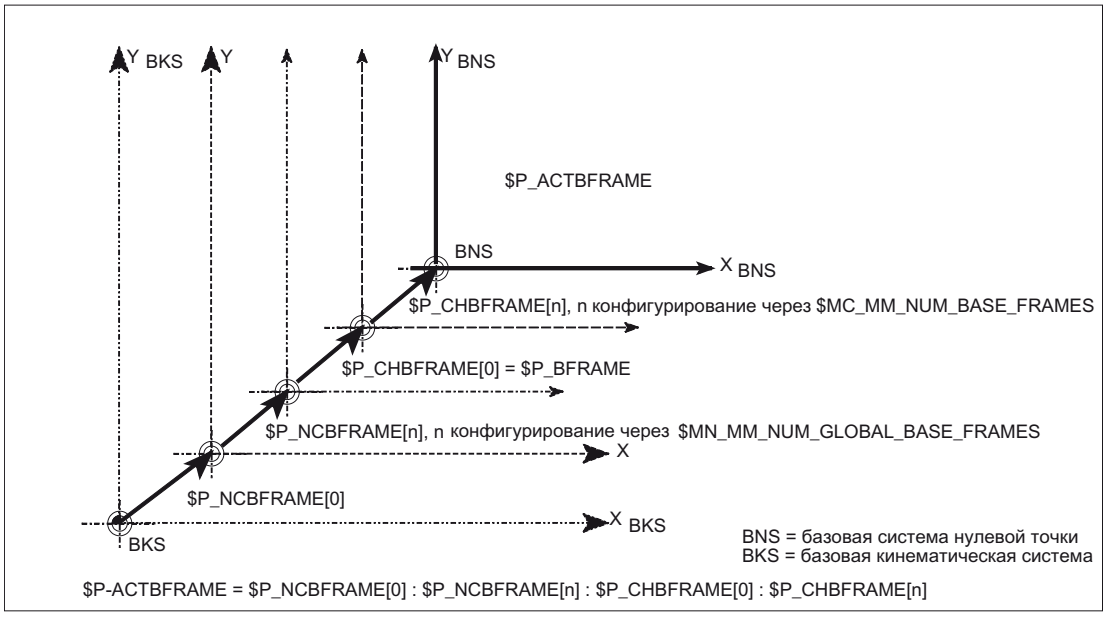
$P_BFRAME идентична $P_CHBFRAME[0]. Системная переменная стандартно всегда имеет действительное значение. Если спец. для канала базовый фрейм отсутствует, то при записи или чтении выводится аварийное сообщение "Фрейм: недопустимый оператор".

$P_ACTBFRAME Общий базовый фрейм

Переменная $P_ACTFRAME определяет связанный общий базовый фрейм. Переменная только для чтения.

$P_ACTFRAME соответствует:

$P_NCBFRAME[0] : ... : $P_NCBFRAME[n] : $P_CHBFRAME[0] : ... : $P_CHBFRAME[n].



$P_CHBFRMASK и $P_NCBFRMASK Общий базовый фрейм

Через системную переменную $P_CHBFRMASK и $P_NCBFRMASK пользователь может выбрать, какие базовые фреймы он хотел бы включить в вычисление "общего" базового фрейма. Переменные могут программироваться только в программе и считываться через BTSS. Значение переменных интерпретируется как битовая маска и указывает, какой элемент поля базового фрейма из $P_ACTFRAME включается в вычисление.

$P_CHBFRMASK позволяет установить, какие спец. для канала базовые фреймы учитываются; $P_NCBFRMASK позволяет установить, какие глобальные базовые фреймы учитываются.

При программировании переменных общий базовый фрейм и общий фрейм рассчитываются заново. После Reset и в первичной установке $P_CHBFRMASK и $P_NCBFRMASK имеют следующее значение:

$P_CHBFRMASK = $MC_CHBFRAME_RESET_MASK

$P_NCBFRMASK = $MC_CHBFRAME_RESET_MASK

Пример:

```
$P_NCBFRMASK = 'H81' ;$P_NCBFRAME[0] : $P_NCBFRAME[7]
$P_CHBFRMASK = 'H11' ;$P_CHBFRAME[0] : $P_CHBFRAME[4]
```

$P_IFRAME Актуальный настраиваемый фрейм

Через предопределенную фрейм-переменную $P_IFRAME актуальный настраиваемый фрейм, действующий в канале, может быть считан и записан в программе обработки детали. Записанный настраиваемый фрейм сразу же учитывается.

Для настраиваемых глобальных фреймов измененный фрейм действует только в канале, в котором фрейм был запрограммирован. Если необходимо изменить фрейм для всех каналов ЧПУ, то потребуется одновременная запись в $P_UIFR[n] и $P_IFRAME. После прочие каналы еще должны активировать соответствующий фрейм, к примеру, с G54.

$P_PFRAME Актуальный программируемый фрейм

$P_PFRAME это программируемый фрейм, получаемый из программирования TRANS/ATRANS, G58/G59, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR или из присваивания CTRANS, CROT, CMIRROR, CSCALE программируемому ФРЕЙМУ.

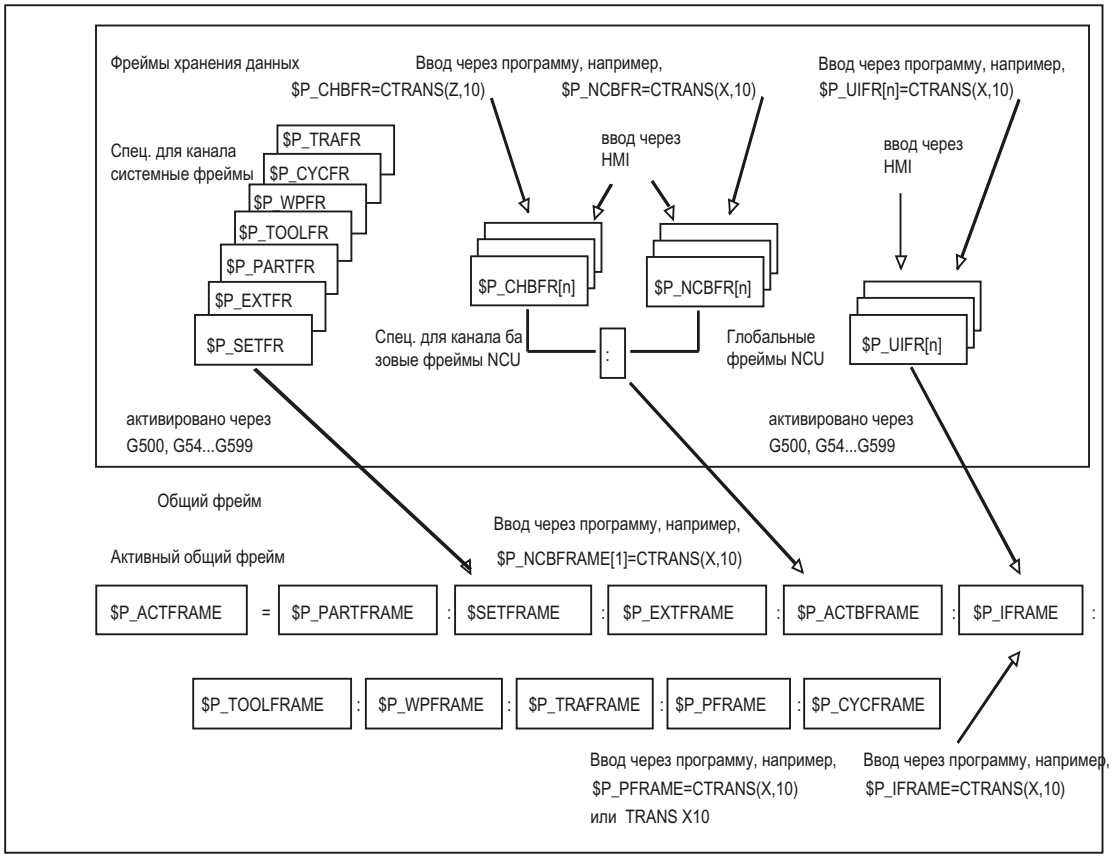
Актуальная, программируемая фрейм-переменная, устанавливающая отношение между настраиваемой системой нулевой точки (ENS) и системой координат детали (WCS).

P_ACTFRAME Актуальный общий фрейм

Актуальный результирующий общий фрейм $P_ACTFRAME получается как соединение всех базовых фреймов, актуального настраиваемого фрейма и программируемого фрейма. Актуальный фрейм актуализируется всегда, когда изменяется один из его компонентов.

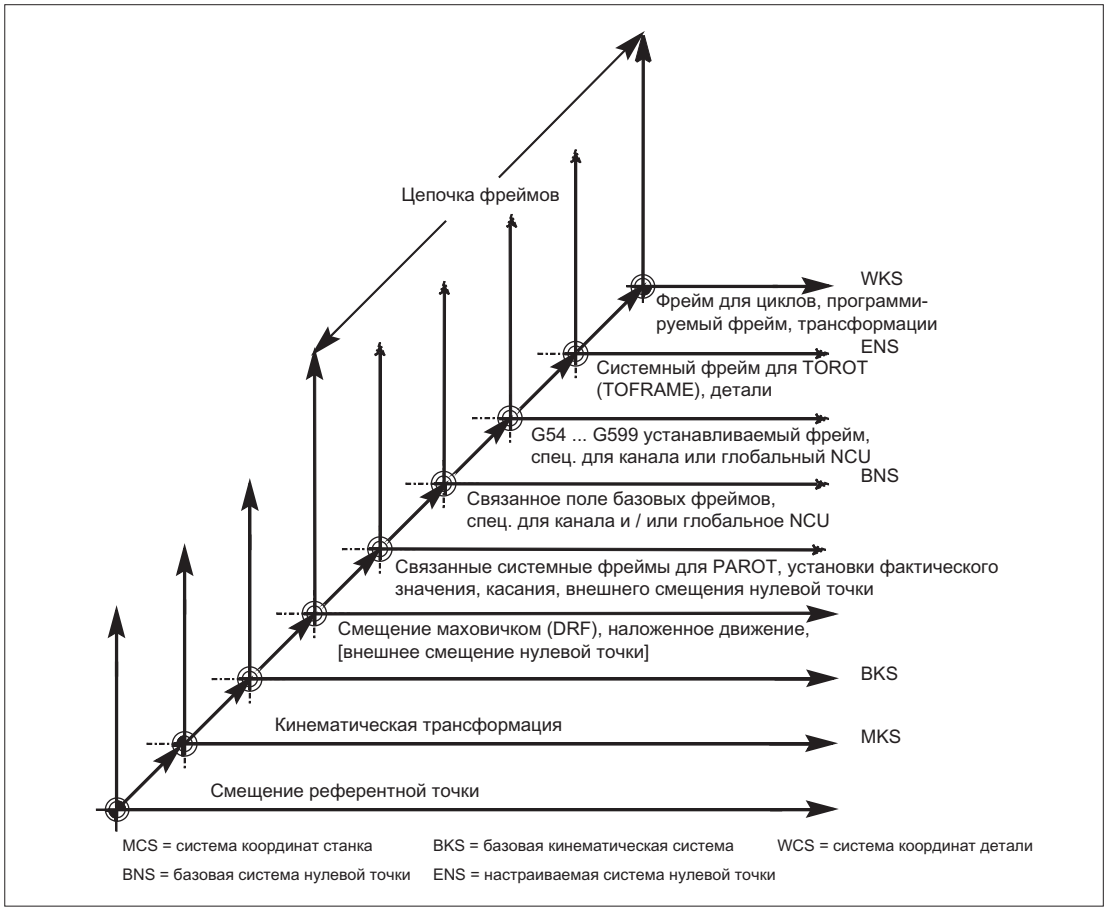
`$P_ACTFRAME` соответствует:

```
$P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME : $P_ACTBFRAME : $P_IFRAME :
$P_TOOLFRAME : $P_WPFRAME : $P_TRAFRAME : $P_PFRAME : $P_CYCFRAME
```


Фреймы хранения данных
Ввод через программу, например,
$P_CHBFR=CTRANS(Z,10)
$P_NCBFR=CTRANS(X,10)
Ввод через программу, например,
$P_UIFR[n]=CTRANS(X,10)
Спец. для канала
системные фреймы
$P_TRAFR
$P_CYCFR
$P_WPFR
$P_TOOLFR
$P_PARTFR
$P_EXTFR
$P_SETFR
ввод через
HMI
ввод через
HMI
$P_CHBFR[n]
$P_NCBFR[n]
$P_UIFR[n]
Спец. для канала ба
зовые фреймы NCU
:
Глобальные
фреймы NCU
активировано через
G500, G54...G599
активировано через
G500, G54...G599
Общий фрейм
Ввод через программу, например,
$P_NCBFRAME[1]=CTRANS(X,10)
Активный общий фрейм
$P_ACTFRAME
=
$P_PARTFRAME
:
$SETFRAME
:
$P_EXTFRAME
:
$P_ACTBFRAME
:
$P_IFRAME
:
$P_TOOLFRAME
:
$P_WPFRAME
:
$P_TRAFRAME
:
$P_PFRAME
:
$P_CYCFRAME
Ввод через программу, например,
$P_PFRAME=CTRANS(X,10)
или TRANS X10
Ввод через программу, например,
$P_IFRAME=CTRANS(X,10)

Цепочка фреймов

Актуальный фрейм состоит из общего базового фрейма, настраиваемого фрейма, системного фрейма и программируемого фрейма согласно указанному выше актуальному общему фрейму.



4.9 Трансформации

4.9.1 Общее программирование типов трансформаций

Трансформации корректируют СЧПУ в соответствии с кинематикой станков. Активация трансформации выполняется посредством вызова функции в программе ЧПУ. Через эти параметры можно задать и ориентацию инструмента в пространстве, и движения ориентации круговых осей.

В случае трансформаций ориентации (3-, 4-, 5-осевая трансформация) запрограммированные данные позиций всегда относятся к острию инструмента. Инструмент отслеживается ортогонально к находящейся в пространстве обрабатываемой поверхности. Декартовы координаты пересчитываются из базовой

кинематической системы в систему координат станка и относятся к гео-осям. Они описывают рабочую точку. Виртуальные круговые оси описывают ориентации инструмента в пространстве.

При кинематической трансформации (TRANSMIT, TRACYL, TRAANG) позиции могут программироваться в декартовой системе координат. СЧПУ преобразует запрограммированные движения перемещения декартовой системы координат в движения перемещения реальных осей станка.

Программирование

Активация трансформации ориентации

`TRAORI(`<номер трансформации>,<вектор ориентации>,<смещение круговой оси>)

Активация кинематической трансформации

`TRANSMIT(`<номер трансформации>)

`TRACYL(`<рабочий диаметр>,<номер трансформации>)

`TRAANG(`<угол наклонной оси>,<номер трансформации>)

Выключение активной трансформации

`TRAFOOF`

Трансформация ориентации

3-, 4-, 5-осевые трансформации (TRAORI)

Для оптимальной обработки оформленных в пространстве поверхностей в рабочей зоне станка, станку, кроме трех линейных осей X, Y и Z, необходимы дополнительные оси. Дополнительные оси описывают ориентацию в пространстве и далее будут называться "оси ориентации". Как оси вращения они доступны для четырех типов станков с различной кинематикой:

1. Двухосевая качающаяся головка
   Например, карданная инструментальная головка с круговой осью параллельно линейной оси при зафиксированном инструментальном столе
2. Двухосевой поворотный стол
   Например, зафиксированная качающаяся головка с инструментальным столом, поворачивающимся вокруг двух осей.
3. Одноосевая качающаяся головка и одноосевой поворотный стол
   Например, качающаяся головка с повернутым инструментом при инструментальном столе, поворачивающимся вокруг одной оси.
4. Двухосевая качающаяся головка и одноосевой поворотный стол
   Например, при инструментальном столе, поворачивающимся вокруг одной оси, и вращающаяся качающаяся головка с вращающимся вокруг самого себя инструментом

3- и 4-осевые трансформации это особые формы 5-осевой трансформации, которые программируются аналогично 5-осевым трансформациям.

Объем функций трансформации "**Базовая 3-/4-/5-/6-осевая трансформация**" включает расположенные под прямым углом круговые оси, а также трансформации для карданной фрезерной головки; как и любую другую трансформацию ориентации, ее для этих четырех типов станков может активировать TRAORI. При базовой 5/6-осевой трансформации ориентация инструмента имеет дополнительную третью степень свободы, при которой инструмент может вращаться вокруг собственной оси произвольно в пространстве в направлении инструмента.

Дополнительная информация: Описание функций – Трансформации

Независящая от кинематики первичная установка ориентации инструмента

ORIRESET

Если с TRAORI активна трансформация ориентации, то с помощью ORIRESET можно указать первичные установки макс. 3 осей ориентации с опционными параметрами A, B, C. Согласование запрограммированных параметров с круговыми осями выполняется согласно определенной через трансформацию последовательности осей ориентации. Следствием программирования ORIRESET(A, B, C) является линейное и синхронное движение осей ориентации от их актуальной позиции к указанной позиции первичной установки.

Кинематические трансформации

TRANSMIT и TRACYL

Для фрезерных обработок на токарных станках для согласованной трансформации можно активировать следующие режимы обработки:

- Торцовая обработка в токарном зажиме с TRANSMIT
- Обработка пазов произвольной формы на цилиндрических телах с TRACYL

TRAANG

Если, например, для технологии шлифования, необходима возможность подачи оси подачи под углом, то TRAANG позволяет запрограммировать параметрируемый угол для согласованной трансформации.

Движение “от точки к точке” в декартовой системе координат

К кинематической трансформации относится и "Движение "от точки к точке" в декартовой системе координат", при котором может быть запрограммировано до 8 положений шарнира (STAT=...). Позиции программируются в декартовой системе координат, при этом движение станка осуществляется в координатах станка.

Дополнительная информация: Описание функций – Трансформации

Последовательная связь трансформаций

Возможно последовательное включение двух трансформаций соответственно. Для связанной таким образом второй трансформации компоненты движения осей берутся из первой трансформации.

В качестве первой трансформации возможны:

- Трансформация ориентации TRAORI
- Полярная трансформация TRANSMIT
- Цилиндрическая трансформация TRACYL
- Трансформация наклонной оси TRAANG

Второй трансформацией должна быть наклонная ось TRAANG.

4.9.1.1 Движения ориентации при трансформациях

Движения перемещения и движения ориентации

Движения перемещения программируемых ориентаций зависят, в первую очередь, от типа станка. При 3-, 4- и 5-осевой трансформации с TRAORI круговые оси или поворотные линейные оси описывают движения ориентации инструмента.

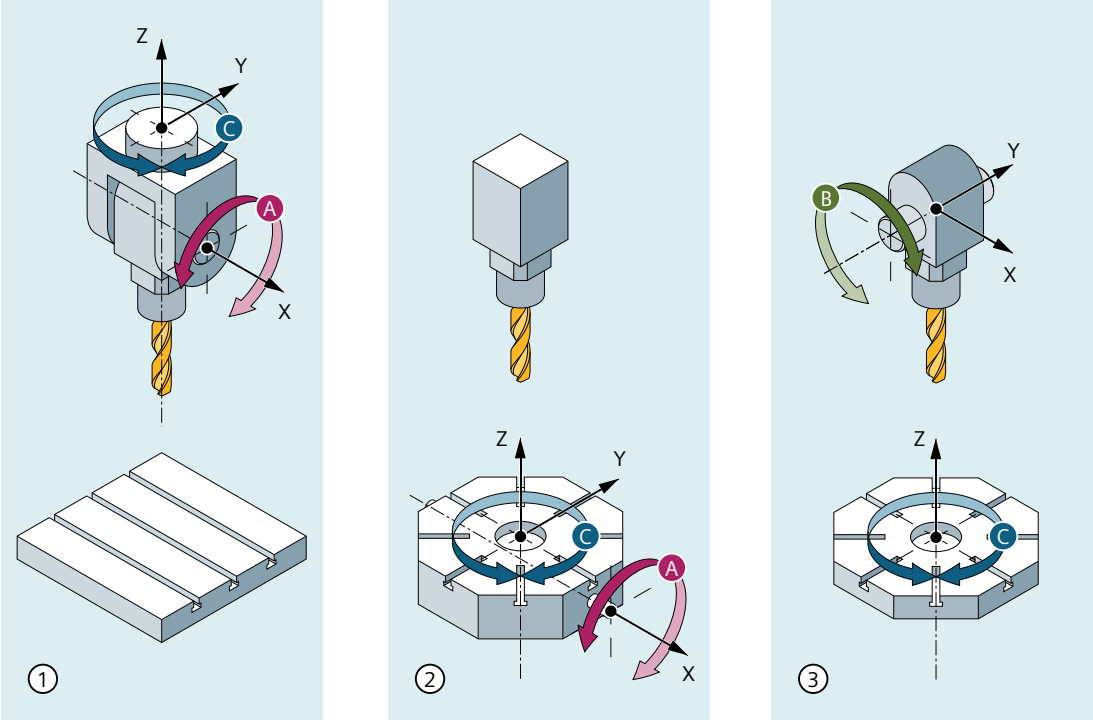
Изменения позиций участвующих в трансформации ориентации круговых осей приводят к движениям компенсации прочих осей станка. При этом позиция острия инструмента остается неизменной.

Движения ориентации инструмента могут программироваться через идентификаторы круговых осей A..., B..., C... виртуальных осей, в зависимости от использования, либо через указание эйлерова угла или угла RPY или векторов направления или векторов нормалей плоскостей, нормированных векторов для оси вращения конуса, либо для промежуточной ориентации на боковой поверхности конуса.

При кинематической трансформации с TRANSMIT, TRACYL и TRAANG СЧПУ преобразует запрограммированные движения перемещения декартовой системы координат в движения перемещения реальных осей станка.

Кинематика станка при 3-, 4- и 5-осевой трансформации TRAORI

Возможно вращение либо инструмента, либо инструментального стола с помощью макс. двух круговых осей. Комбинация одноосевой качающейся головки и поворотного стола также возможна.



① Тип станка 1 с последовательностью осей CA
- Инструментальный суппорт может вращаться вокруг оси X и оси Z
- Неподвижный стол изделия

② Тип станка 2 с последовательностью осей AC
- Неподвижный инструментальный суппорт
- Стол изделия может вращаться вокруг оси X и оси Z

③ Тип станка 3 с последовательностью осей BC
- Инструментальный суппорт может вращаться вокруг оси Y
- Стол изделия может вращаться вокруг оси Z

Тип станка	Программирование ориентации
3-осевая трансформация, типы станка 1 и 2	Программирование ориентации инструмента только в плоскости, расположенной вертикально к круговой оси. Существует **две** поступательные оси (линейные оси) и **одна** вращательная ось (круговая ось).
4-осевая трансформация, типы станка 1 и 2	Программирование ориентации инструмента только в плоскости, расположенной вертикально к круговой оси. Существует **три** поступательные оси (линейные оси) и **одна** вращательная ось (круговая ось).
5-осевая трансформация, типы станков 3 Одноосевая качающаяся головка и одноосевой поворотный стол	Программирование трансформации ориентации. Кинематика с **тремя** линейными осями и **двумя** ортогональными круговыми осями. Круговые оси расположены параллельно двум из трех линейных осей. Первая круговая ось приводится в движение двумя декартовыми линейными осями. Она вращает третью линейную ось с инструментом. Вторая круговая ось вращает деталь.

Базовые 5/6-осевые трансформации

Тип станка	Программирование трансформации ориентации
Базовая 5-/6-осевая трансформация, тип станка 4 Двухосевая качающаяся головка с вращающимся вокруг самого себя инструментом и одноосевой поворотный стол	Программирование трансформации ориентации. Кинематика с **тремя** линейными осями и **тремя** ортогональными круговыми осями. Круговые оси расположены параллельно двум из трех линейных осей. Первая круговая ось приводится в движение двумя декартовыми линейными осями. Она вращает третью линейную ось с инструментом. Вторая круговая ось вращает деталь. Базовая ориентация инструмента может быть запрограммирована посредством дополнительного вращения вокруг себя самой с углом поворота THETA.

При вызове "Базовой 3-, 4- и 5-/6-осевой трансформации" дополнительно может быть передана первичная ориентация инструмента. Ограничения касательно направлений круговых осей более не действуют. Если круговые оси расположены не точно вертикально друг на друге или если имеющиеся круговые оси расположены не точно параллельно линейным осям, то "Базовая 5-/6-осевая трансформация" может обеспечить лучшие результаты ориентации инструмента..

Кинематические трансформации TRANSMIT, TRACYL и TRAANG

Для фрезерных обработок на токарных станках или для подаваемой под наклоном оси при шлифовании, в зависимости от трансформации, в стандартной ситуации действуют следующие расположения осей:

TRANSMIT	Активация полярной трансформации
Торцовая обработка в токарном зажиме	Одна круговая ось Одна ось подачи вертикально к оси вращения Одна продольная ось параллельно оси вращения

TRACYL	Активация трансформации боковой поверхности цилиндра
Обработка пазов произвольной формы на цилиндрических телах	Одна круговая ось Одна ось подачи вертикально к оси вращения Одна продольная ось параллельно оси вращения

TRAANG	Активация трансформации наклонной оси
Обработка с наклонной осью подачи	Одна круговая ось Одна ось подачи с параметрируемым углом Одна продольная ось параллельно оси вращения

Движение "от точки к точке" в декартовой системе координат

Движение станка выполняется в координатах станка и программируется с:

TRAORI	Активация трансформации
PTP Движение "от точки к точке"	Подвод к позиции в декартовой системе координат (MCS)
CP	Движение по траектории декартовых осей в (BKS)
STAT	Положение шарниров зависит от трансформации
TU	На какой угол оси двигаются по кратчайшему пути

Движение от точки к точке при базовой 5/6-осевой трансформации

Движение станка выполняется в координатах станка и ориентация инструмента может быть запрограммирована как с позициями круговых осей, так и с независимыми от кинематики векторами эйлерова угла или угла PRY, или векторами направления.

При этом возможна интерполяция круговой оси, векторная интерполяция с большой круговой интерполяцией или интерполяция вектора ориентации на боковой поверхности конуса.

Пример 3- до 5-осевой трансформации для карданной фрезерной головки

Станок имеет минимум 5 осей, из которых

- Три линейные оси для прямолинейных движений, перемещающие рабочую точку в любую позицию в рабочей зоне.
- Две круговые оси качаний, расположенные под сконфигурированным углом (в большинстве случаев 45 градусов), позволяют инструменту принимать ориентации в пространстве, которые при расположении под углом 45 градусов ограничиваются полусферой.

4.9.1.2 Обзор трансформации ориентации TRAORI

Возможные типы программирования в комбинации с TRAORI

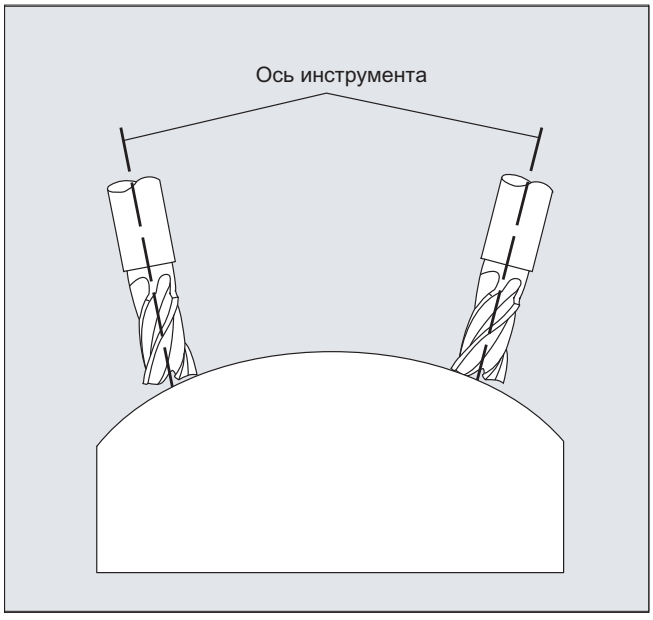
Тип станка	Программирование при активной трансформации TRAORI
Типы станков 1, 2 или 3 Двухосевая качающаяся головка или двухосевой поворотный стол или комбинация одноосевой качающейся головки и поворотного стола соответственно.	Последовательность осей ориентации и направление ориентации инструмента может проектироваться либо **относительно станка** через машинные данные в зависимости от кинематики станка, либо **относительно детали** с программируемой ориентацией независимо от кинематики станка Направления вращения осей ориентации в базовой системе программируются с: - ORIMKS базовая система = система координат станка - ORIWKS базовая система = система координат детали Первичной установкой является ORIWKS. Программирование осей ориентации с: A, B, C позиций осей станка напрямую A2, B2, C2 программирование угла виртуальных осей с - ORIEULER через эйлеров угол (стандарт) - ORIRPY через угол RPY - ORIVIRT1 через виртуальные оси ориентации 1-ого определения - ORIVIRT2 через виртуальные оси ориентации 2-ого определения с дифференциацией типа интерполяции: **линейная интерполяция** - ORIAXES осей ориентации или осей станка **большая круговая интерполяция** (интерполяция вектора ориентации) - ORIVECT осей ориентации Программирование осей ориентации через указание A3, B3, C3 компонентов вектора (направления/нормали плоскости) Программирование результирующей ориентации инструмента A4, B4, C4 вектора нормалей плоскостей в начале кадра A5, B5, C5 вектора нормалей плоскостей в конце кадра LEAD угол предварения для ориентации инструмента TILT боковой угол для ориентации инструмента

Тип станка	Программирование при активной трансформации TRAORI
	Интерполяция вектора ориентации на боковой поверхности конуса Изменения ориентации на расположенной **произвольно в пространстве** боковой поверхности конуса через интерполяцию: - ORIPLANE в плоскости (большая круговая интерполяция) - ORICONCW на боковой поверхности конуса по часовой стрелке - ORICONCCW на боковой поверхности конуса против часовой стрелки A6, B6, C6 вектора направления (ось вращения конуса) -OICONIO интерполяция на боковой поверхности конуса с: A7, B7, C7 промежуточные векторы (стартовая и конечная ориентация) или - ORICONTO на боковой поверхности конуса тангенциальный переход Изменения ориентации относительно **траектории** с - ORICURVE задача движения двух контактных точек через PO[XH]=(xe, x2, x3, x4, x5) Полиномы ориентации до 5-ого порядка PO[YH]=(ye, y2, y3, y4, y5) Полиномы ориентации до 5-ого порядка PO[ZH]=(ze, z2, z3, z4, z5) Полиномы ориентации до 5-ого порядка - ORIPATHS Сглаживание характеристики ориентации с A8, B8, C8 Этап переориентации инструмента соответствует: направление и длина хода инструмента при движении отвода
Типы станков 1 и 3 Для других типов станков с дополнительным вращением инструмента вокруг себя самого необходима третья круговая ось Трансформация ориентации, к примеру, базовая 6-осевая трансформация. Вращение вектора ориентации.	Программирование **вращений** ориентации инструмента с LEAD Угол предварения относительно вектора нормалей плоскостей PO[PHI] Программирование полинома макс. 5-ого порядка TILT Боковой угол Вращение вокруг касательной к траектории (направление Z) PO[PSI] Программирование полинома макс. 5-ого порядка THETA Угол поворота (вращение вокруг направления инструмента в Z) THETA= значение, достигаемое в конце кадра THETA=AC(...) Покадровое переключение на абсолютное указание размера THETA=IC(...) Покадровое переключение на указание составного размера THETA=Θ_e Интерполяция запрограммированного угла G90/G91 PO[THT]=(..) Программирование полинома макс. 5-ого порядка Программирование вектора вращения - ORIROTA абсолютное вращение - ORIROTR относительный вектор вращения - ORIROTT тангенциальный вектор вращения
Ориентация относительно траектории для изменений ориентации относительно траектории или вращения вектора вращения тангенциально к траектории	Изменения ориентации **относительно траектории** с - ORIPATH Ориентация инструмента относительно траектории - ORIPATHS дополнительно при изломе характеристики ориентации Программирование вектора вращения - ORIROTC тангенциальный вектор вращения, вращение к касательной к траектории

4.9.2 3-, 4- и 5-осевая трансформация (TRAORI)

4.9.2.1 Общие связи карданной инструментальной головки

Для достижения оптимальных условий резания при обработке изогнутых в пространстве поверхностей требуется возможность изменения угла установки инструмента.



С какой конструкцией станка это достигается, зафиксировано в данных осей.

5-осевая трансформация

Карданная инструментальная головка

Здесь три линейные оси (X, Y, Z) и две оси ориентации (C, A) определяют угол установки и рабочую точку инструмента. Одна из двух осей ориентации является наклонной осью, здесь в примере A' – во многих случаях располагается под углом 45°.

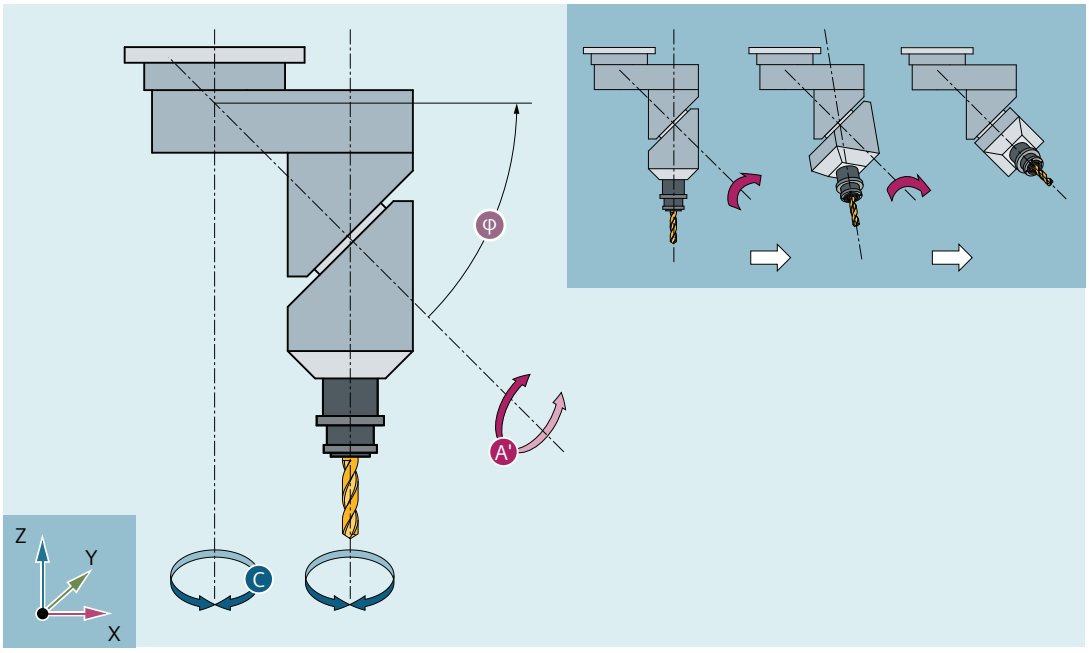
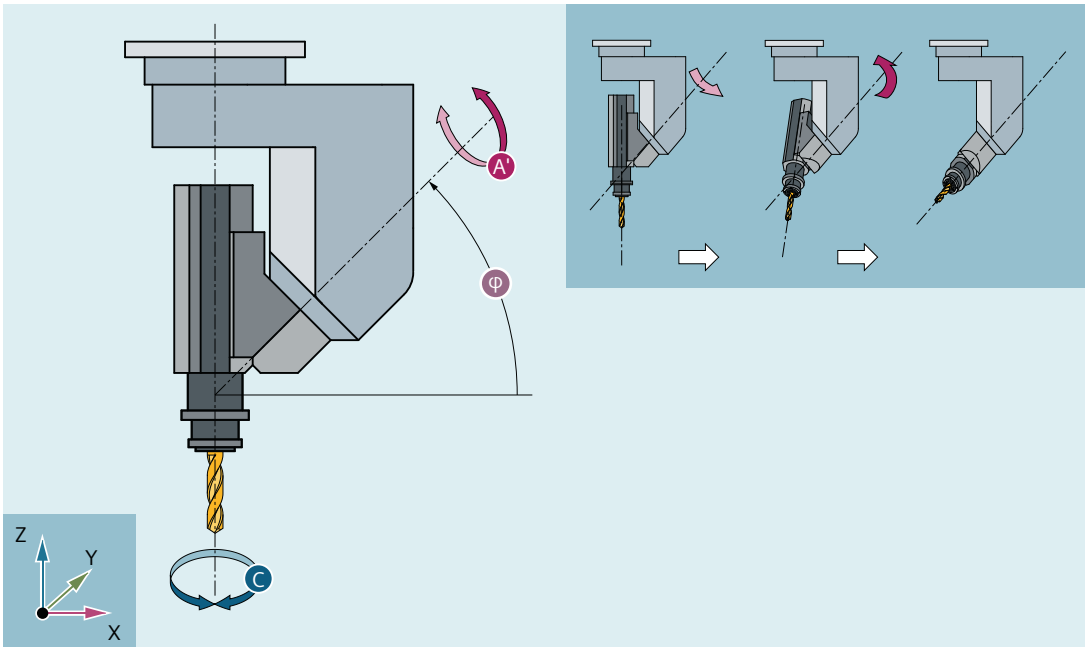


Рисунок 4-3 Карданная инструментальная головка, вариант 1

В приведенных здесь примерах показаны расположения на примере с карданной инструментальной головкой кинематики станка CA!

Последовательность круговых осей и направление ориентации инструмента могут устанавливаться в зависимости от кинематики станка через машинные данные (→ изготовитель станка).



В этом примере A' лежит под углом φ к оси X

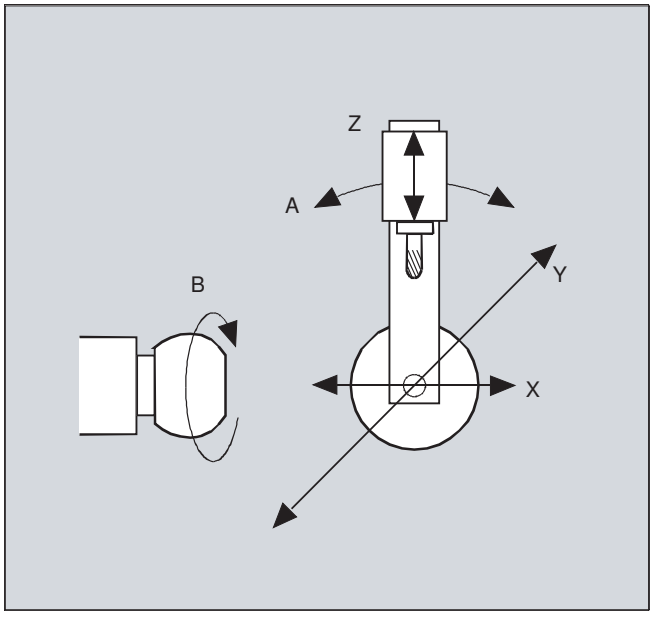
В общем и целом действуют следующие возможные связи:

A' лежит под углом φ к	оси X
B' лежит под углом φ к	оси Y
C' лежит под углом φ к	оси Z

Угол φ может быть сконфигурирован в диапазоне 0° до +89° через машинные данные.

С качающейся линейной осью

Здесь речь идет о компоновке с подвижной деталью и подвижным инструментом. Кинематика состоит из трех линейных осей (X, Y, Z) и двух расположенных под прямым углом осей вращения. Первая круговая ось приводится в движение, к примеру, через крестовый суппорт двумя линейными осями, инструмент располагается параллельно третьей линейной оси. Вторая ось вращения поворачивает деталь. Третья линейная ось (ось качания) лежит в плоскости крестового суппорта.



Последовательность круговых осей и направление ориентации инструмента могут устанавливаться в зависимости от кинематики станка через машинные данные.

Действуют следующие возможные связи:

Оси:	**Последовательности осей:**
1-я круговая ось	A A B B C C
2-я круговая ось	B C A C A B
Повернутая линейная ось	Z Y Z X Y X

Дополнительная информация по конфигурируемым последовательностям осей для направления ориентации инструмента см.: Описание функций — Трансформации

4.9.2.2 3-, 4- и 5-осевая трансформация (TRAORI)

Пользователь может сконфигурировать две или три линейные оси и две круговые оси. Трансформации исходят из того, что круговые оси расположены ортогонально на плоскости ориентации.

Ориентация инструмента возможна только в плоскости, расположенной вертикально к круговой оси. Трансформация поддерживает типы станков с подвижным инструментом и подвижной деталью.

Проектирование и программирование 3- и 4-осевых трансформаций аналогично таковым для 5-осевой трансформации.

Синтаксис

```
TRAORI(<n>)
TRAORI(<n>,<X>,<Y>,<Z>,<A>,<B>)
TRAFOOF
```

Объяснение

TRAORI:	Активирует первую согласованную трансформацию ориентации	
TRAORI(<n>):	Активирует согласованную с n трансформацию ориентации	
<n>:	номер трансформации	
	Значение:	1 или 2
	Пример: TRAORI(1) активирует трансформацию ориентации 1	
<X>,<Y>,<Z>:	Компонент вектора ориентации, на который указывает инструмент.	
<A>,<B>:	Программируемое смещение для круговых осей	
TRAFOOF:	Отключить трансформацию	

Ориентация инструмента

В зависимости от выбранного направления ориентации инструмента, в программе ЧПУ активная рабочая плоскость (G17, G18, G19) должна быть установлена таким образом, чтобы коррекция длин инструмента действовала в направлении ориентации инструмента.

Примечание

После включения трансформации данные позиций (X, Y, Z) всегда относятся к острию инструмента. Изменение позиций участвующих в трансформации круговых осей приводит к движениям компенсации прочих осей станка, из-за чего позиция острия инструмента остается неизменной.

Трансформация ориентации всегда направлена от острия к зажиму инструмента.

Смещение для осей ориентации

При активации трансформации ориентации можно напрямую запрограммировать дополнительное смещение для осей ориентации.

Можно пропустить параметры, если при программировании соблюдается правильная последовательность.

Пример:

`TRAORI(, , , ,A,B)` : если необходимо ввести только одно единственное смещение.

В качестве альтернативы прямому программированию, дополнительное смещение для осей ориентации может быть автоматически взято и из актуального активного смещения нулевой точки. Передача конфигурируется через машинные данные.

Примеры

```
TRAORI(1,0,0,1)          ; Первичная ориентация инструмента указывает в направлении Z
TRAORI(1,0,1,0)          ; Первичная ориентация инструмента указывает в направлении Z
TRAORI(1,0,1,1)          ; Первичная ориентация инструмента указывает в направлении Y/Z (соответствует позиции -45°)
```

4.9.2.3 Варианты программирования ориентации и первичная установка (ORIRESET)

Программирование ориентации инструмента при TRAORI

В комбинации с программируемой трансформацией ориентации TRAORI дополнительно к линейным осям X, Y, Z через идентификаторы круговых осей A.., B..., C... могут быть запрограммированы позиции осей или виртуальные оси с углами или компоненты вектора. Для осей ориентации и осей станка возможны различные типы интерполяции. Независимо от того, какие полиномы ориентации PO[угол] и полиномы осей PO[ось] активны на соответствующий момент, может быть запрограммировано несколько различных типов полиномов, например, G1, G2, G3, CIP или POLY.

Изменение ориентации инструмента может быть запрограммировано и через векторы ориентации. При этом конечная ориентация каждого кадра может быть выполнена либо через прямое программирование вектора, либо через программирование позиций круговых осей.

Варианты программирования ориентации при 3- до 5-осевой трансформации

Следующие варианты программирования ориентации являются взаимоисключающими.

`A, B, C`	Прямое указание позиции круговой оси
`A2, B2, C2`	Программирование угла виртуальных осей через угол Эйлера или угол RPY.
`A3, B3, C3`	Указание компонентов вектора
`LEAD, TILT`	Указание угла предварения и бокового угла относительно траектории и поверхности
`A4, B4, C4` `A5, B5, C5`	вектор нормали плоскости в начале кадра и в конце кадра

A6, B6, C6 A7, B7, C7	Интерполяция вектора ориентации на боковой поверхности конуса
A8, B8, C8	Переориентация инструмента, направления и длины хода инструмента при движении отвода

Подвод к первичной установке ориентации инструмента

Посредством ORIRESET(...) оси ориентации соответствующей кинематики станка линейно и синхронно перемещаются из своей текущей позиции на запрограммированную позиция первичной установки. Если позиция первичной установки для оси запрограммирована, то используется позиция из соответствующих машинных данных $MC_TRAFO5_ROT_AX_OFFSET_1/2.

Активные фреймы круговых осей не учитываются.

Примеры кинематики станка CA (имена осей канала C, A)

Команды	Описание
ORIRESET (90, 45)	Ось C: 90° Ось A: 45°
ORIRESET(, 30)	Ось C: $MC_TRAFO5_ROT_AX_OFFSET_1/2[0] Ось A: 30°
ORIRESET( )	Ось C: $MC_TRAFO5_ROT_AX_OFFSET_1/2[0] Ось A: $MC_TRAFO5_ROT_AX_OFFSET_1/2[1]

Примеры кинематики станка CAC (имена осей канала C, A, B)

Команды	Описание
ORIRESET(90, 45, 90)	Ось C: 90° Ось A: 45° Ось B: 90°
ORIRESET( )	Ось C: $MC_TRAFO5_ROT_AX_OFFSET_1/2[0] Ось A: $MC_TRAFO5_ROT_AX_OFFSET_1/2[1] Ось B: $MC_TRAFO5_ROT_AX_OFFSET_1/2[2]

Примечание

Подвод к исходному положению ориентации инструмента с помощью функции ORIRESET(...) разрешено выполнять, только если трансформация ориентации TRAORI(...) активна.

Программирование вращений LEAD, TILT и THETA

Угол предварения LEAD и боковой угол TILT

Вращения ориентации инструмента программируются при 3- до 5-осевой трансформации с углом предварения LEAD и боковым углом TILT.

Угол поворота THETA

При трансформации **с третьей круговой осью** можно запрограммировать вращение инструмента вокруг себя самого с углом поворота THETA для ориентирования как с компонентами вектора, так и с программированием углов LEAD, TILT.

4.9.2.4 Программирование ориентации инструмента (A..., B..., C..., LEAD, TILT)

Для программирования ориентации инструмента существуют следующие возможности:

1. Прямое программирование движения круговых осей. Изменение ориентации всегда выполняется в базовой кинематической или системе координат станка. Оси ориентации перемещаются как синхронные оси.
2. Программирование в углах Эйлера или RPY согласно определению угла через `A2, B2, C2`.
3. Программирование вектора направления через `A3, B3, C3`. Вектор направления указывает от острия в направлении зажима инструмента.
4. Программирование вектора нормалей плоскостей в начале кадра с `A4, B4, C4` и в конце кадра с `A5, B5, C5` (торцовое фрезерование).
5. Программирование через угол предварения `LEAD` и боковой угол `TILT`
6. Программирование оси вращения конуса как нормированного вектора через `A6, B6, C6` или промежуточной ориентации на боковой поверхности конуса через `A7, B7, C7`,
   см. главу "Программирование ориентации вдоль боковой поверхности конуса (ORIPLANE, ORICONxx)".
7. Программирование переориентации, направления и длины хода инструмента при движении отвода через `A8, B8, C8`,
   см. главу "Сглаживание характеристики ориентации (ORIPATHS A8=, B8=, C8=)"

Примечание

Во всех случаях программирование ориентации разрешено только тогда, когда включена трансформация ориентации.

Преимущество: Эти программы работают на любой кинематике станка.

Определение ориентации инструмента через команду G

Примечание

Изготовитель станка

Через машинные данные можно переключаться между углами Эйлера или RPY. При соответствующих настройках машинных данных возможно переключение, как зависимое, так и не зависимое от активной команды G группы 50. На выбор имеются следующие возможности настройки:

1. Если оба машинных параметра для определения осей ориентации и определения угла ориентации установлены G-командой на ноль: Запрограммированные с `A2, B2, C2` углы, **независимо от машинных данных** определения углов программирования ориентации, интерпретируются либо как углы Эйлера, либо как углы RPY.
2. Если машинный параметр определения осей ориентации через G-команду установлен на единицу, то переключение осуществляется **в зависимости** от активной G-команды группы 50:
   Запрограммированные с `A2, B2, C2` углы интерпретируются согласно одной из активных G-команд `ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2, ORIAXPOS` и `ORIPY2`. Запрограммированные с осями ориентации значения согласно активной G-команде группы 50 интерпретируются и как угол ориентации.
3. Если машинный параметр для определения угла ориентации через G-команду установлен на единицу, а машинный параметр для определения осей ориентации через G-команду установлен на ноль, то переключение осуществляется **независимо** с активной G-команды группы 50:
   Запрограммированные с `A2, B2, C2` углы интерпретируются согласно одной из активных G-команд `ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2 ORIAXPOS` и `ORIPY2` . Независимо от активной G-команды группы 50 запрограммированные с осями ориентации значения всегда интерпретируются как позиции круговых осей.

Синтаксис

Позиции круговых осей

```
G1 X<значение> Y<значение> Z<значение> A<значение> B<значение>
C<значение>
```

Угол Эйлера

```
G1 X<значение> Y<значение> Z<значение> A2=<значение> B2=<значение>
C2=<значение>
```

Вектор направления

```
G1 X<значение> Y<значение> Z<значение> A3=<значение> B3=<значение>
C3=<значение>
```

Вектор нормали к плоскости в начале кадра

```
G1 X<значение> Y<значение> Z<значение> A4=<значение> B4=<значение>
C4=<значение>
```

Вектор нормали к плоскости в конце кадра

G1 X<значение> Y<значение> Z<значение> A5=<значение> B5=<значение> C5=<значениеt>

Угол предварения

LEAD=<значение>

Боковой угол

TILT=<значение>

Объяснение

G1:	Линейная интерполяция
X, Y, Z:	Позиции линейных осей
A, B, C:	Позиции круговых осей
A2=, B2=, C2=:	Программирование угла (угол Эйлера или угол RPY)
A3=, B3=, C3=:	Вектор направления в координатах X, Y, Z в WCS
A4=, B4=, C4=:	Вектор нормали плоскости в начале кадра в координатах X, Y, Z в WCS
A5=, B5=, C5=:	Вектор нормали плоскости в конце кадра в координатах X, Y, Z в WCS
LEAD= :	Угол предварения [1)]
TILT= :	Боковой угол [1)]
1) Интерпретация указаний углов зависит от предустановки в MD21094 $MC_ORIPATH_MODE	

Дополнительная информация

Как правило, 5-осевые программы создаются системами CAD/CAM, а не вводятся на СЧПУ. Поэтому объяснения ниже прежде всего предназначены для программистов постпроцессоров.

Для программирования ориентации доступны следующие команды:

Команда	Объяснение
ORIEULER:	Угол Эйлера с последовательностью вращений ZX'Z"
ORIRPY:	Угол RPY с последовательностью вращений XY'Z"
ORIRPY2:	Угол RPY с последовательностью вращений ZY'X"
ORIVIRT1:	Виртуальные оси ориентации со свободно определяемой последовательностью вращений через: MD21120 $MC_ORIAX_TURN_TAB_1
ORIVIRT2:	Виртуальные оси ориентации со свободно определяемой последовательностью вращений через: MD21130 $MC_ORIAX_TURN_TAB_2
ORIAXPOS:	Виртуальные оси ориентации с позициями круговых осей

Примечание

Через машинные данные изготовителем станка могут быть определены различные варианты. Следовать указаниям изготовителя станка.

Программирование в углах Эйлера ORIEULER, последовательность вращений Z X' Z"

Значения, запрограммированные при программировании ориентации ORIEULER с A2, B2, C2, интерпретируются как угол Эйлера (в градусах).

Новый вектор ориентации получается посредством следующих трех вращений изначального вектора ориентации

1. с круговой осью A2 вокруг оси координат Z
2. с круговой осью B2 вокруг новой оси координат X'
3. с круговой осью C2 вокруг новой оси координат Z"

В этом случае значение C2 (поворот вокруг новой оси Z) не имеет значения, и его программирование не требуется.

Программирование в углах RPY ORIRPY ORIRPY, последовательность вращений X Y' Z"

Значения, запрограммированные при программировании ориентации ORIRPY посредством A2, B2, C2, интерпретируются как угол RPY (в градусах) с последовательностью вращений X Y' Z'.

Примечание

В отличие от программирования посредством ORIEULER, при ORIRPY все три значения влияют на вектор ориентации.

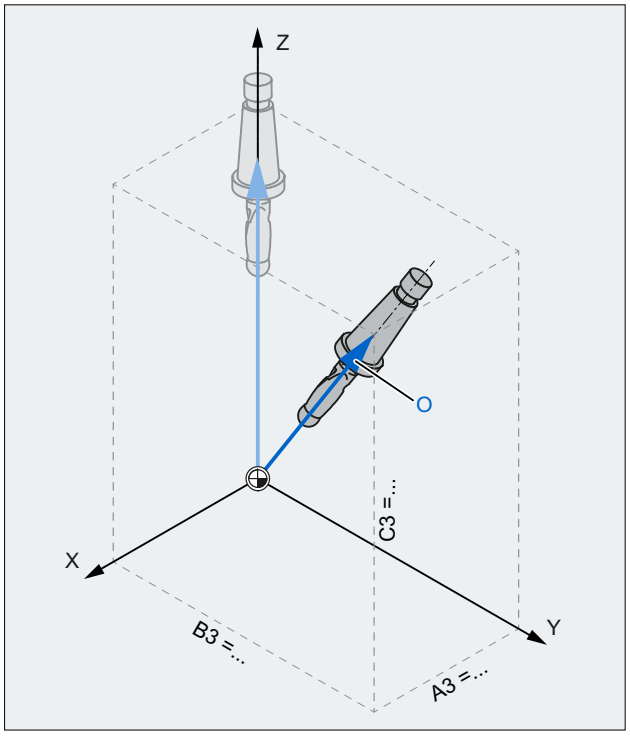
Новый вектор ориентации получается посредством следующих трех вращений изначального вектора ориентации

1. с круговой осью A2 вокруг оси координат X
2. с круговой осью B2 вокруг новой оси координат Y'
3. с круговой осью C2 вокруг новой оси координат Z"

Программирование вектора направления

Компоненты вектора направления программируются с A3, B3, C3 . Вектор указывает в направлении зажима инструмента; длина вектора при этом не имеет значения.

Не запрограммированные компоненты вектора устанавливаются на ноль.



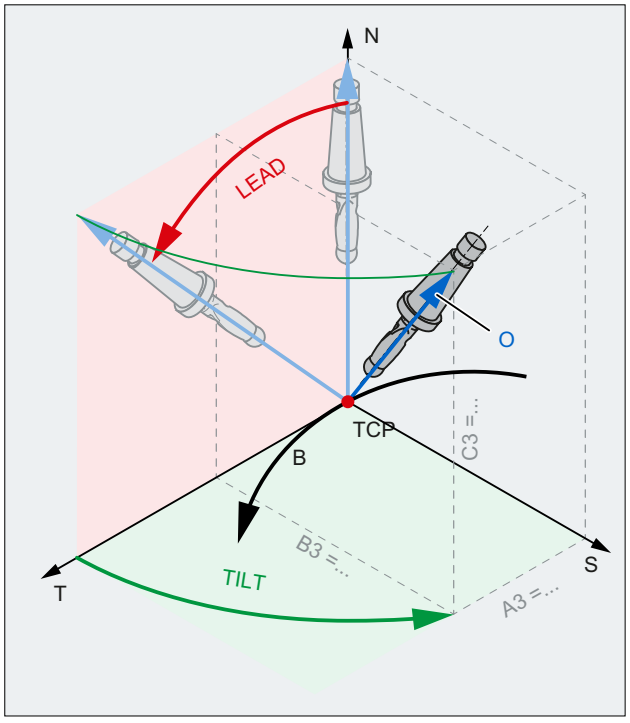
X, Y, Z	Оси координат WCS
A3, B3, C3	Компоненты вектора направления
O	вектор ориентации

Рисунок 4-4 Программирование вектора направления

Программирование ориентации инструмента посредством LEAD и TILT

Результирующая ориентация инструмента получается из:

- касательной к траектории
- вектора нормали к плоскости
  В начале кадра `A4, B4, C4` и в конце кадра `A5, B5, C5`
- Угол предварения `LEAD`
  Угол в плоскости, образованной касательной к траектории и вектором нормали к плоскости
- Боковой угол `TILT` в конце кадра
  Угол в плоскости, вертикально к касательной к траектории относительно вектора нормали к плоскости



T	Касательная к траектории
S	Перпендикуляр к касательной к траектории.
N	Нормаль к плоскости
B	Траектория
TCP	Tool Center Point (центр инструмента)
O	вектор ориентации

Рисунок 4-5 Программирование LEAD TILT

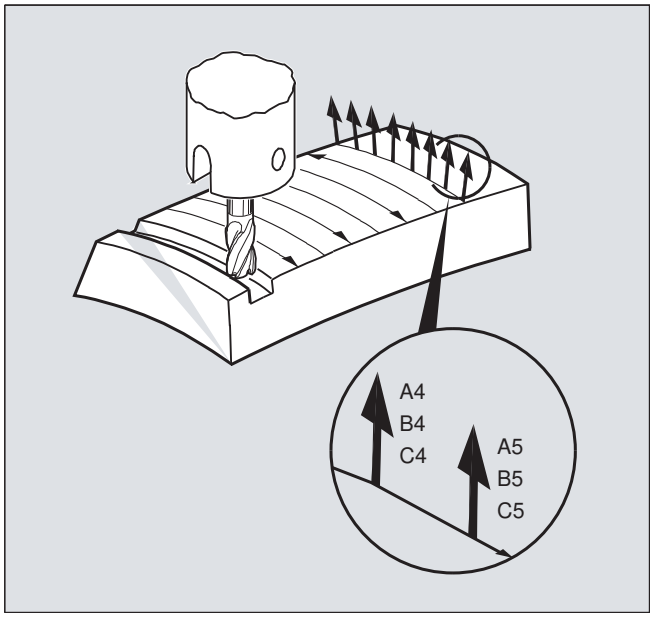
Примечание

Поведение на внутренних углах при 3D-коррекции инструмента (WZK)

Даже если кадр на внутреннем углу укорачивается, запрограммированная ориентация инструмента на конце кадра все равно устанавливается.

4.9.2.5 Торцовое фрезерование (A4, B4, C4, A5, B5, C5)

Торцовое фрезерование служит для обработки поверхностей с любым изгибом.



Для этого типа фрезерования 3D необходимо построчное описание траекторий 3D на поверхности детали.

Вычисления с учетом формы и размеров инструмента обычно выполняются в CAM. После полностью вычисленные кадры ЧПУ через постпроцессоры загружаются в СЧПУ.

Программирование искривления траектории

Описание поверхностей

Описание искривления траектории осуществляется через векторы нормалей плоскостей со следующими компонентами:

`A4, B4, C4` Стартовый вектор в начале кадра

`A5, B5, C5` Конечный вектор на конце кадра

Если в кадре стоит только стартовый вектор, то вектор нормали плоскости остается постоянным во всем кадре. Если в кадре стоит только конечный вектор, то от конечного значения предшествующего кадра через большую круговую интерполяцию выполняется интерполяция к запрограммированному конечному значению.

Если запрограммированы стартовый и конечный вектор, то между двумя направлениями также выполняется большая круговая интерполяция. Благодаря этому достигаются непрерывные ровные ходы траектории.

В первичной установке векторы нормалей плоскостей, независимо от активной плоскости G17 до G19, указывают в направлении Z.

Длина вектора не имеет значения.

Не запрограммированные компоненты вектора устанавливаются на ноль.

При активной ORIWKS (см. "Нулевая точка осей ориентации (ORIWKS, ORIMKS) (Страница 700)") векторы нормалей плоскостей относятся к активному фрейму и при вращении фрейма также поворачиваются.

Изготовитель станка

Вектор нормали плоскости должен располагаться в пределах настраиваемого через машинные данные граничного значения вертикально к касательной к траектории, иначе выводится ошибка.

4.9.2.6 Нулевая точка осей ориентации (ORIWKS, ORIMKS)

При программировании ориентации в системе координат детали через

- угол Эйлера или угол RPY или
- вектор ориентации

можно установить ход вращательного движения через ORIMKS/ORIWKS.

Примечание

Изготовитель станка

Тип интерполяции для ориентации устанавливается с помощью машинных данных:

MD21104 $MC_ORI_IPO_WITH_G_CODE

=FALSE: Базой являются G-команды ORIWKS и ORIMKS

=TRUE: Базой являются G-команды 51-й группы (ORIAXES, ORIVECT, ORIPLANE, ...)

Синтаксис

```
ORIMKS=...
ORIWKS=...
```

Объяснение

ORIMKS:	Вращение в системе координат станка
ORIWKS:	Вращение в системе координат детали

Примечание

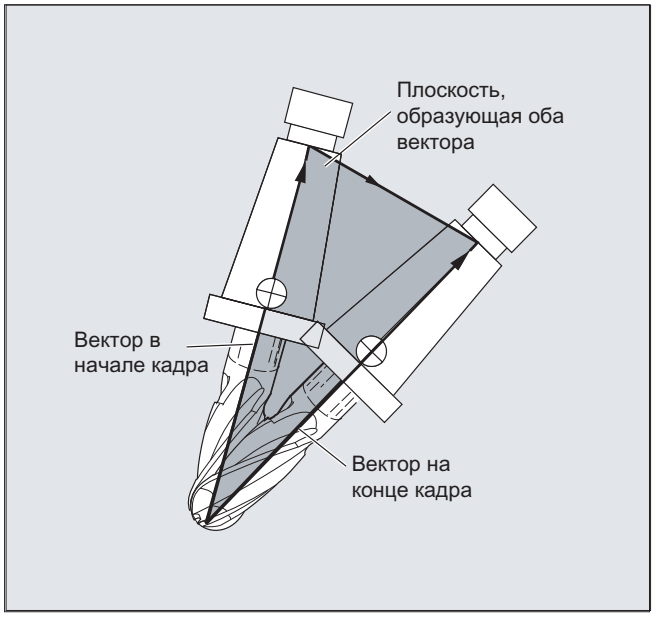
ORIWKS является первичной настройкой. Если в случае 5-осевой программы изначально не ясно, на каком станке она должна работать, то всегда выбирается ORIWKS . Какие движения в действительности выполняет станок, зависит от его кинематики.

С помощьюORIMKSможно запрограммировать фактические движения станка, например, во избежание столкновений с приспособлениями или и т. п.

Дополнительная информация

При ORIMKS выполняемое движение инструмента **зависит** от кинематики станка. При изменении ориентации с зафиксированным в пространстве острием инструмента выполняется линейная интерполяция между позициями круговых осей.

При ORIWKS движение инструмента **не зависит** от кинематики станка. При изменении ориентации с зафиксированным в пространстве острием инструмента, инструмент движется в образованной начальным и конечным вектором плоскости.



Сингулярные позиции

Примечание

ORIWKS

Для движений ориентации в области сингулярной позиции 5-осевого станка требуются большие движения осей станка. (Например, в случае поворотной токарной головки и C в качестве оси вращения и A в качестве оси качания все позиции с A=0 являются сингулярными).

Изготовитель станка

Во избежание перегрузки осей станка, управление скоростью значительно снижает скорость движения по траектории вблизи от сингулярных позиций.

С помощью машинных данных

$MC_TRAFO5_NON_POLE_LIMIT

$MC_TRAFO5_POLE_LIMIT

трансформацию можно спараметрировать так, что движения ориентации вблизи полюса прокладываются через полюс, и возможна быстрая обработка.

Сингулярные позиции обрабатываются только с MD $MC_TRAFO5_POLE_LIMIT .

Дополнительная информация: Описание функций – Трансформации

4.9.2.7 Программирование осей ориентации (ORIAXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2)

Функция "Оси ориентации" описывает ориентацию инструмента в пространстве и достигается через программирование смещений для круговых осей. Следующая третья степень свободы может быть достигнута через дополнительное вращение инструмента вокруг себя самого. Эта ориентация инструмента выполняется произвольно в пространстве через третью круговую ось и для нее необходима 6-осевая трансформация. Самовращение инструмента вокруг себя самого устанавливается в зависимости от типа интерполяции векторов вращения с углом поворота THETA (см. "Вращения ориентации инструмента (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Страница 711)".

Оси ориентации программируются через идентификаторы осей A2, B2, C2.

Синтаксис

```
N... ORIAXES/ORIVECT                ; линейная или большая круговая интерполяция
N... G1 X Y Z A B C

N... ORIPLANE                       ; интерполяция ориентации плоскости

N... ORIEULER/ORIRPY/ORIRPY2        ; угол ориентации Эйлера/RPY
N... G1 X Y Z A2 = B2 = C2 =        ; программирование угла виртуальных осей

N... ORIVIRT1/ORIVIRT2              ; виртуальные оси ориентации опр. 1/2
N... G1 X Y Z A3 = B3 = C3 =        ; программирование вектора направления
```

Примечание

Для изменений ориентации находящейся в пространстве боковой поверхности конуса могут быть запрограммированы и другие смещения круговой оси осей ориентации (см. "Программирование ориентации вдоль боковой поверхности конуса (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Страница 704)".

Значение

ORIAXES:	Линейная интерполяция осей станка или осей ориентации
ORIVECT:	Большая круговая интерполяция (идентично ORIPLANE)
ORIMKS: ORIWKS:	Вращение в системе координат станка Вращение в системе координат детали Описание см. "Нулевая точка осей ориентации (ORIWKS, ORIMKS) (Страница 700)".
A= B= C=:	Программирование позиции оси станка
ORIEULER:	Программирование ориентации через угол Эйлера

ORIRPY:	Программирование ориентации через угол RPY Последовательность вращения XYZ, при этом действует: • A2 это угол поворота вокруг X • B2 это угол поворота вокруг Y • C2 это угол поворота вокруг Z
ORIRPY2:	Программирование ориентации через угол RPY Последовательность вращения ZYX, при этом действует: • A2 это угол поворота вокруг Z • B2 это угол поворота вокруг Y • C2 это угол поворота вокруг X
A2= B2= C2=:	Программирование угла виртуальных осей
ORIVIRT1/ORIVIRT2:	Программирование ориентации через виртуальные оси ориентации Определение 1: установка по MD21120 $MC_ORIAX_TURN_TAB_1 Определение 2: установка по MD21130 $MC_ORIAX_TURN_TAB_2
A3= B3= C3=:	Программирование вектора направления оси направления

Дополнительная информация

Изготовитель станка

Посредством MD21102 $MC_ORI_DEF_WITH_G_CODE устанавливается, как определяются запрограммированные углы A2, B2, C2:

определение выполняется по MD21100 $MC_ORIENTATION_IS_EULER (стандарт) или определение выполняется по G-группе 50 (ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2).

Посредством MD21104 $MC_ORI_IPO_WITH_G_CODE устанавливается, какой тип интерполяции действует: ORIWKS/ORIMKS или ORIAXES/ORIVECT.

Режим работы JOG

В этом режиме работы всегда выполняется линейная интерполяция углов ориентации. При непрерывном и инкрементальном перемещении клавишами перемещения может перемещаться только одна ось ориентации. Маховичками оси ориентации можно перемещать одновременно.

Для перемещения осей ориентации вручную действует спец. для канала переключатель процентовки подачи или переключатель процентовки ускоренного хода при наложении ускоренного хода.

Следующие машинные данные обеспечивают отдельную задачу скорости:

MD21160 $MC_JOG_VELO_RAPID_GEO

MD21165 $MC_JOG_VELO_GEO

MD21150 $MC_JOG_VELO_RAPID_ORI

MD21155 $MC_JOG_VELO_ORI

Примечание

SINUMERIK 840D sl с "Пакетом трансформаций для манипуляторов"

С помощью функции "Декартово перемещение вручную" в режиме JOG возможно раздельное преобразование поступательного движения гео-осей в базовых системах MCS, WCS и TCS.

Дополнительная информация: Описание функций – Трансформации

4.9.2.8 Программирование ориентации вдоль боковой поверхности конуса (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO)

С помощью расширенной ориентации можно выполнять изменения ориентации вдоль находящейся в пространстве боковой поверхности конуса. Интерполяция вектора ориентации на боковой поверхности конуса осуществляется с помощью модальных команд ORICONxx. Для интерполяции в одной плоскости можно запрограммировать конечную ориентацию с ORIPLANE. В общем и целом стартовая ориентация определяется предшествующими кадрами.

Программирование

Конечная ориентация определяется либо через указание программирования угла в углах Эйлера или RPY с A2, B2, C2, либо через программирование позиций круговых осей с A, B, C. Для осей ориентации вдоль боковой поверхности конуса требуются дополнительные данные программирования:

- Ось вращения конуса как вектор с `A6, B6, C6`
- Апертурный угол `PSI` с идентификатором `NUT`
- Промежуточная ориентация на боковой поверхности конуса с `A7, B7, C7`

Примечание

Программирование вектора направления A6, B6, C6 для оси вращения конуса

Программирование конечной ориентации не является обязательным. Если конечная ориентация не указана, то выполняется интерполяция всей боковой поверхности конуса с 360 градусами.

Программирование апертурного угла конуса с NUT=угол

Обязательно указать конечную ориентацию.

Интерполяция всей боковой поверхности конуса с 360 градусами таким образом невозможна.

Программирование промежуточной ориентации A7, B7, C7 на боковой поверхности конуса

Обязательно указать конечную ориентацию. Изменение ориентации и направление вращения однозначно определяются через три вектора стартовой, конечной и промежуточной ориентации. При этом все три вектора должны отличаться друг от друга. Если запрограммированная промежуточная ориентация параллельна стартовой или конечной ориентации, то выполняется линейная большая круговая интерполяция в плоскости, образуемой стартовым и конечным вектором.

Расширенная интерполяция ориентации на боковой поверхности конуса

```
N... ORICONCW или ORICONCCW
N... A6= B6= C6= A3= B3= C3=
или
N... ORICONTO
N... G1 X Y Z A6= B6= C6=
или
N... ORICONIO
N... G1 X Y Z A7= B7= C7=
N... PO[PHI]=(a2, a3, a4, a5)
N... PO[PSI]=(b2, b3, b4, b5)
```

Интерполяция на боковой поверхности конуса с

вектором направления по/против часовой стрелки конуса и конечной ориентацией или

тангенциальным переходом и указанием конечной ориентации

или

указанием конечной ориентации с промежуточной ориентации на боковой поверхности конуса с

полиномом для угла поворота с полиномом для апертурного угла

Параметры

ORIPLANE:	Интерполяция в плоскости (большая круговая интерполяция)
ORICONCW:	Интерполяция на боковой поверхности конуса по часовой стрелке
ORICONCCW:	Интерполяция на боковой поверхности конуса против часовой стрелки
ORICONTO:	Интерполяция на боковой поверхности конуса, тангенциальный переход
A6= B6= C6=:	Программирование оси вращения конуса (нормированный вектор)
NUT=угол:	Апертурный угол конуса в градусах
NUT=+179:	Угол перемещения меньше или равен 180 градусам
NUT=-181:	Угол перемещения больше или равен 180 градусам
ORICONIO:	Интерполяция на боковой поверхности конуса
A7= B7= C7=:	Промежуточная ориентация (программирование как нормированный вектор)
PHI:	Угол поворота ориентации вокруг оси направления конуса
PSI:	Апертурный угол конуса
Возможные полиномы PO[PHI]=(a2, a3, a4, a5) PO[PSI]=(b2, b3, b4, b5)	Кроме соответствующих углов могут быть запрограммированы и полиномы макс. 5-го порядка

Пример: Различные изменения ориентации

Программный код	Комментарий
…	
N10 G1 X0 Y0 F5000	
N20 TRAORI(1)	; Трансформация ориентации вкл.
N30 ORIVECT	; Интерполяция ориентации инструмента как вектора.
…	; Ориентация инструмента в плоскости.
N40 ORIPLANE	; Выбрать большую круговую интерполяцию.
N50 A3=0 B3=0 C3=1	
N60 A3=0 B3=1 C3=1	; Ориентация в плоскости Y/Z повернута на 45 градусов, на конце кадра достигается ориентация (0,1/√2,1/√2).
…	
N70 ORICONCW	; Программирование ориентации на боковой поверхности конуса:
N80 A6=0 B6=0 C6=1 A3=0 B3=0 C3=1	; Выполняется интерполяция вектора ориентации на боковой поверхности конуса с направлением (0,0,1) до ориентации (1/√2,0,1/√2) по часовой стрелке, угол поворота при этом составляет 270 градусов.
N90 A6=0 B6=0 C6=1	; Ориентация выполняет полный оборот на той же боковой поверхности конуса.

Дополнительная информация

Если необходимо описать изменения ориентации на расположенной произвольно в пространстве боковой поверхности конуса, то должен быть известен вектор, вокруг которого должна вращаться ориентация инструмента. Кроме этого, должны быть заданы начальная и конечная ориентации. Стартовая ориентация следует из предшествующего кадра, а конечная ориентация должна быть либо запрограммирована, либо определена через иные условия.

Программирование в плоскости ORIPLANE соответствует ORIVECT

Программирование большой круговой интерполяции в комбинации с угловыми полиномами соответствует линейной и полиномиальной интерполяции контуров. Интерполяция ориентации инструмента выполняется в плоскости, образованной стартовой и конечной ориентацией. При программировании дополнительных полиномов вектор ориентации может быть опрокинут из плоскости.

Программирование окружностей в плоскости G2/G3, CIP и CT

Расширенная ориентация соответствует интерполяции окружностей в плоскости. Соответствующие возможности программирования окружностей с указанием центра или радиуса как G2/G3, окружности через промежуточную точку CIP и тангенциальных окружностей CT см. главу "Круговая интерполяция (Страница 197)".

Программирование ориентации

Интерполяция вектора ориентации на боковой поверхности конуса ORICONxx

Для интерполяции ориентаций на боковой поверхности конуса можно выбрать четыре различных типа интерполяции из группы команд G 51:

1. Интерполяция на боковую поверхность конуса по часовой стрелке ORICONCW с указанием конечной ориентации и направления конуса или апертурного угла. Вектор направления программируется с идентификаторами A6, B6, C6, а апертурный угол конуса – с идентификатором NUT= диапазон значений в интервале 0 градусов до 180 градусов.
2. Интерполяция на боковую поверхность против часовой стрелки ORICONCWW с указанием конечной ориентации и направления конуса или апертурного угла. Вектор направления программируется с идентификаторами A6, B6, C6, а апертурный угол конуса – с идентификатором NUT= диапазон значений в интервале 0 градусов до 180 градусов.
3. Интерполяция на боковую поверхность конуса ORICONIO с указанием конечной ориентации и промежуточной ориентации, которая программируется с идентификаторами A7, B7, C7.
4. Интерполяция на боковую поверхность конуса ORICONTO с тангенциальным переходом и указанием конечной ориентации. Вектор направления программируется с идентификаторами A6, B6, C6.

4.9.2.9 Задача ориентации двух контактных точек (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=)

Программирование изменения ориентации через вторую пространственную кривую ORICURVE

Следующей возможностью программирования изменений ориентации является программирование, кроме острия инструмента вдоль пространственной кривой, и движения второй контактной точки инструмента с ORICURVE. Таким образом возможно однозначное определение изменений ориентации инструмента, как и при программировании самого вектора инструмента.

Изготовитель станка

Следовать указаниям изготовителя станка касательно настраиваемых через машинные данные идентификаторов осей для программирования 2-й траектории ориентации инструмента.

Программирование

При этом типе интерполяции для обеих пространственных кривых могут быть запрограммированы точки с G1 или полиномы с POLY. Окружности и эвольвенты запрещены. Дополнительно можно активировать сплайн-интерполяцию с BSPLINE и функцию "Соединение коротких сплайн-кадров".

Дополнительная информация по функции "Объединение коротких сплайн-кадров": см. Описание функций – Основные функции.

Другие типы сплайнов ASPLINE и CSPLINE, а также активация компрессора с COMP... запрещены.

Движение двух контактных точек инструмента при программировании полиномов ориентации может задаваться для координат до макс. 5-го порядка.

Расширенная интерполяция ориентации с дополнительной пространственной кривой и полиномами для координат

```
N... ORICURVE                          ; Указание движения второй контактной точки инструмента и дополнительные полиномы соответствующих координат
N... PO[XH]=(xe, x2, x3, x4, x5)
N... PO[YH]=(ye, y2, y3, y4, y5)
N... PO[ZH]=(ze, z2, z3, z4, z5)
```

Параметры

`ORICURVE`	Интерполяция ориентации с задачей движения двух контактных точек инструмента.
`XH YH ZH`	Идентификатор координат второй контактной точки инструмента дополнительного контура как пространственная кривая

`PO[XH]=(xe, x2, x3, x4, x5)`
`PO[YH]=(ye, y2, y3, y4, y5)`
`PO[ZH]=(ze, z2, z3, z4, z5)`
	Возможные полиномы
	Кроме соответствующих конечных точек, пространственные кривые дополнительно могут программироваться с полиномами.

xe, ye, ze	Конечные точки пространственной кривой
xi, yi, zi	Коэффициенты полиномов макс. 5-го порядка

Примечание

Идентификаторы XH YH ZH для программирования 2-й траектории ориентации

Идентификаторы должны быть выбраны таким образом, чтобы не возникло конфликта с другими идентификаторами линейных осей

осями X Y Z

и круговыми осями, например,

A2 B2 C2 угол Эйлера или угол RPY

A3 B3 C3 векторы направления

A4 B4 C4 или A5 B5 C5 векторы нормалей плоскостей

A6 B6 C6 векторы вращения или A7 B7 C7 координаты промежуточной точки

или иными параметрами интерполяции.

4.9.3 Полиномы ориентации (PO[угол], PO[координата])

Независимо от того, какая полиномиальная интерполяция группы функций G 1 активна на соответствующий момент, два различных типа полиномов ориентации макс. 5-го порядка могут быть запрограммированы при 3-, 4-, 5-осевой трансформации.

1. Полиномы для **угла:** Угол предварения LEAD, боковой угол TILT относительно плоскости, образуемой стартовой и конечной ориентацией.
2. Полиномы для **координат:** XH, YH, ZH второй пространственной кривой для ориентации инструмента исходной точки на инструменте.

При 6-осевой трансформации дополнительно к ориентации инструмента можно запрограммировать вращение вектора вращения THT с полиномами до макс. 5-ого порядка для вращений самого инструмента.

Синтаксис

Полиномы ориентации типа 1 для **угла**

```
N... PO[PHI]=(a2, a3, a4, a5)
N... PO[PSI]=(b2, b3, b4, b5)
```

3- до 5-осевая трансформация

Полиномы ориентации типа 2 для **координат**

```
N… PO[XH]=(xe, x2, x3, x4, x5)
N… PO[YH]=(ye, y2, y3, y4, y5)
N… PO[ZH]=(ze, z2, z3, z4, z5)
```

Идентификаторы для координат второй траектории ориентации для ориентации инструмента

Дополнительно в обоих случаях можно запрограммировать полином для **вращения** при 6-осевой трансформации с

```
N… PO[THT]=(c2, c3, c4, c5)
```

или

```
N… PO[THT]=(d2, d3, d4, d5)
```

Интерполяция вращения относительно траектории

абсолютной, относительной и тангенциальной интерполяции для изменения ориентации

вектора ориентации. Это возможно тогда, когда трансформация поддерживает вектор вращения с программируемым через угол поворота THETA и интерполируемым смещением.

Объяснение

`PO[PHI]`	Угол в плоскости между стартовой и конечной ориентацией
`PO[PSI]`	Угол, описывающий опрокидывание ориентации из плоскости между стартовой и конечной ориентацией
`PO[THT]`	Угол поворота G-команд группы 54, запрограммированных через поворот вектора вращения с THETA
`PHI`	Угол предварения LEAD
`PSI`	Боковой угол TILT
`THETA`	Вращение вокруг направления инструмента в Z
`PO[XH]`	Координата X исходной точки на инструменте
`PO[YH]`	Координата Y исходной точки на инструменте
`PO[ZH]`	Координата Z исходной точки на инструменте

Дополнительная информация

Полиномы ориентации не могут быть запрограммированы

- Если активны сплайн-интерполяции ASPLINE, BSPLINE, CSPLINE.
  Полиномы типа 1 для угла ориентации возможны для любого типа интерполяции кроме сплайна, т. е. для линейной интерполяции с ускоренным ходом G00 или с подачей G01
  при полиномиальной интерполяции с POLY и
  при круговой или эвольвентной интерполяции с G02, G03, CIP, CT, INVCW и INCCCW
  .
  Полиномы типа 2 для координат ориентации, напротив, возможны только тогда, когда активна
  линейная интерполяция с ускоренным ходом G00 или с подачей G01 или полиномиальная интерполяция с POLY.
- Если выполняется интерполяция ориентации посредством осевой интерполяции ORIAXES. В этом случае напрямую могут быть запрограммированы полиномы с PO[A] и PO[B] для осей ориентации A и B.

Полиномы ориентации типа 1 с ORIVECT, ORIPLANE и ORICONxx

При большой круговой интерполяции и интерполяции боковой поверхности конуса с ORIVECT, ORIPLANE и ORICONxx возможны только полиномы ориентации типа 1.

Полиномы ориентации типа 2 с ORICURVE

Если активна интерполяция с дополнительной пространственной кривой ORICURVE, то выполняется интерполяция декартовых компонентов вектора ориентации и возможны полиномы ориентации типа 2.

4.9.4 Вращения ориентации инструмента (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA)

Если для типов станков с подвижным инструментом необходима возможность изменения ориентации инструмента, то каждый кадр программируется с конечной ориентацией. В зависимости от кинематики станка, можно запрограммировать либо направление ориентации, либо направление вращения вектора ориентации THETA. Для этих векторов вращения могут быть запрограммированы различные типы интерполяции:

- ORIROTA: угол поворота к заданному абсолютно направлению вращения.
- ORIROTR: угол поворота относительно плоскости между стартовой и конечной ориентаций.
- ORIROTT: угол поворота относительно изменения вектора ориентации.
- ORIROTC: тангенциальный угол поворота к касательной к траектории.

Синтаксис

Только при активном типе интерполяции ORIROTA можно запрограммировать угол поворота или вектор вращения на четыре возможных способа следующим образом:

1. Напрямую позиции круговых осей `A, B, C`
2. Угол Эйлера (в градусах) через `A2, B2, C2`
3. Угол RPY (в градусах) через `A2, B2, C2`
4. Вектор направления через `A3, B3, C3` (угол поворота с помощью `THETA=`<значение>)

Если ORIOTR или ORIOTT активны, то угол поворота может быть запрограммирован и напрямую с `THETA`.

Вращение может быть запрограммировано и одно в кадре, изменения ориентации при этом не происходит. При этом ORIROTR и ORIROTT не имеют значения. В этом случае угол поворота всегда интерпретируется относительно абсолютного направления (ORIROTA).

`N... ORIROTA`	Определить интерполяцию вектора вращения
`N... ORIROTR`	
`N... ORIROTT`	
`N... ORIROTC`	
`N... A3= B3= C3= THETA=`<значение>	Определить вращение вектора ориентации
`N... PO[THT]=(`d_2`, `d_3`, `d_4`, `d_5`)`	Интерполяция угла поворота с полиномом 5-го порядка

Значение

`ORIROTA`:	Угол поворота к заданному абсолютно направлению вращения
`ORIROTR`:	Угол поворота относительно плоскости между стартовой и конечной ориентацией
`ORIROTT`:	Угол поворота как тангенциальный вектор вращения к изменению ориентации
`ORIROTC`:	Угол поворота как тангенциальный вектор вращения к касательной к траектории
`THETA`:	Вращение вектора ориентации
`THETA=`<значение>:	Угол поворота в градусах, достигаемый в конце кадра
`THETA=`Θe:	Угол поворота с конечным углом Θ_e вектора вращения
`THETA=AC(...)`:	Покадровое переключение на абсолютное указание размера
`THETA=AC(...)`:	Покадровое переключение на указание составного размера
Θe:	Конечный угол вектора вращения как абсолютно с G90, так к относительно с G91 (указание составного размера) активен
`PO[THT]=(....)`:	Полином для угла поворота

Пример: Вращения ориентаций

Программный код	Комментарий
N10 TRAORI	; Активировать трансформацию ориентации
N20 G1 X0 Y0 Z0 F5000	; Ориентация инструмента
N30 A3=0 B3=0 C3=1 THETA=0	; в направлении Z с углом поворота 0
N40 A3=1 B3=0 C3=0 THETA=90	; в направлении X и поворот на 90 градусов
N50 A3=0 B3=1 C3=0 PO[THT]=(180,90)	; Ориентация
N60 A3=0 B3=1 C3=0 THETA=IC(-90)	; в направлении Y и поворот на 180 градусов
N70 ORIROTT	; остается постоянной и поворот на 90 градусов
N80 A3=1 B3=0 C3=0 THETA=30	; Угол поворота относительно изменения ориентации
	; Вектор вращения под углом в 30 градусов к плоскости X–Y

При интерполяции кадра N40 выполняется линейная интерполяция от начального значения 0 градусов до конечного значения 90 градусов. В кадре N50 угол поворота изменяется с 90 градусов до 180 градусов по параболе θ(u) = +90u2. В N60 также может быть выполнено вращение, изменения ориентации при этом не происходит.

На N80 ориентация инструмента поворачивается из направления Y в направление X. При этом изменение ориентации лежит в плоскости X–Y, а вектор вращения образует угол в 30 градусов к этой плоскости.

Дополнительная информация

ORIROTA

Угол поворота THETA интерпретируется относительно определенного абсолютно направления в пространстве. Базовое направление вращения выполняется через машинные данные.

ORIROTR

Угол поворота THETA интерпретируется относительно плоскости, образуемой начальной и конечной ориентацией.

ORIROTT

Угол поворота THETA интерпретируется относительно изменения ориентации. Для THETA=0 вектор вращения интерполируется тангенциально изменению ориентации и отличается от ORIROTR только в том случае, если для ориентации был запрограммирован как минимум один полином для "угла поворота PSI". Тем самым получается изменение ориентации, которое выполняется не в плоскости. Тогда через дополнительно запрограммированный угол поворота THETA возможна, к примеру, такая интерполяция вектора вращения, при которой он всегда образует определенное значение для изменения ориентации.

ORIROTC

Интерполяция вектора вращения осуществляется относительно касательной к траектории с программируемым через угол THETA смещением. При этом для угла смещения можно запрограммировать и полином PO[THT]=(c2, c3, c4, c5) макс. 5-го порядка.

4.9.5 Ориентации относительно траектории

4.9.5.1 Типы ориентаций относительно траектории

С помощью этой расширенной функции относительная ориентация достигается не только на конце кадра, а по всему ходу траектории. Достигнутая в предшествующем кадре ориентация через большую круговую интерполяцию переводится в запрограммированную конечную ориентацию. В принципе, существует две возможности программирования желаемой ориентации относительно траектории:

1. Выполняется интерполяция ориентации инструмента и вращения инструмента посредством ORIPATH, ORPATHTS относительно траектории.
2. Программирование и интерполяция вектора ориентации выполняются как и прежде. С ORIROTC устанавливается вращение вектора ориентации относительно касательной к траектории.

Синтаксис

Тип интерполяции ориентации и вращение инструмента программируются с:

`N... ORIPATH`	Ориентации относительно траектории
`N... ORIPATHS`	Ориентация относительно траектории со сглаживанием характеристики ориентации
`N... ORIROTC`	Интерполяция вектора вращения относительно траектории

Вызванный углом в ходе траектории излом ориентации может быть сглажен с помощью ORIPATHS. Направление и длина хода движения отвода программируется через вектор с компонентами A8 = X, B8 = Y, C8 = Z.

С помощью ORIPATH/ORIPATHS для всего хода траектории могут быть запрограммированы могут быть запрограммированы различные отношения к касательной к траектории через три угла

- LEAD = указание прямого угла относительно траектории и поверхности
- TILT = указание бокового угла относительно траектории и поверхности
- THETA = угол поворота

. Для угла поворота THETA с помощью PO[THT] = (...) дополнительно могут быть запрограммированы полиномы макс. 5-го порядка.

Примечание

Изготовитель станка

Следовать указаниям изготовителя станка. Через конфигурируемые машинные и установочные данные для типа ориентации относительно траектории могут быть осуществлены дополнительные настройки.

Дополнительная информация: Описание функций – Трансформации

Объяснение

Возможны различные настройки интерполяции углов LEAD и TILT через машинные данные:

- Запрограммированное с LEAD и TILT отношение ориентации инструмента действует во всем кадре.
- Прямой угол LEAD: вращение вокруг направления вертикально к касательной и вектору нормали TILT: вращение ориентации вокруг вектора нормали.
- Прямой угол LEAD: вращение вокруг направления вертикально к касательной и вектору нормали боковой угол TILT: вращение ориентации вокруг направления касательной к траектории.
- Угол поворота THETA: вращение инструмента вокруг себя самого с дополнительной третьей круговой осью в качестве оси ориентации при 6-осевой трансформации.

Примечание

Ориентация относительно траектории в комбинации с OSC, OSS, OSSE, OSD, OST запрещена.

Интерполяция ориентации относительно траектории ORIPATH или ORIPATHS и ORIOTC не может быть запрограммирована в сочетании со сглаживанием характеристики ориентации с одной из G-команд группы 34. Для этого должна быть активна OSOF.

4.9.5.2 Вращение ориентации инструмента относительно траектории (ORIPATH, ORIPATHS, угол поворота)

При 6-осевой трансформации для произвольной ориентации инструмента в пространстве с помощью третьей круговой оси возможно вращение инструмента вокруг себя самого. При вращении ориентации инструмента относительно траектории с ORIPATH или ORIPATHS можно запрограммировать дополнительное вращение через угол поворота THETA. В качестве альтернативы углы LEAD и TILT могут быть запрограммированы через вектор, лежащий в плоскости вертикально к направлению инструмента.

Изготовитель станка

Следовать указаниям изготовителя станка. Через машинные данные возможны различные установки интерполяции углов LEAD и TILT.

Синтаксис

Вращение ориентации инструмента и инструмента

Тип ориентации инструмента относительно траектории активируется с ORIPATH или ORIPATHS.

`N... ORIPATH`	Активировать тип ориентации относительно траектории
`N... ORIPATHS`	Активировать тип ориентации относительно траектории со сглаживанием характеристики ориентации

Активация трех возможных углов с поворотным действием:

`N... LEAD=`	Угол для запрограммированной ориентации относительно вектора нормали плоскости
`N... TILT=`	Угол для запрограммированной ориентации в плоскости вертикально к касательной к траектории относительно вектора нормали плоскости
`N... THETA=`	Угол поворота относительно изменения ориентации вокруг направления инструмента третьей круговой оси

Значения угла на конце кадра программируются с `LEAD`=значение, `TILT`=значение или `THETA`=значение. В дополнение к постоянным углам для всех трех углов могут быть запрограммированы полиномы макс. 5-го порядка.

`N... PO[PHI]=(a2, a3, a4, a5)`	Полином для угла предварения LEAD
`N... PO[PSI]=(b2, b3, b4, b5)`	Полином для бокового угла TILT
`N... PO[THT]=(d2, d3, d4, d5)`	Полином для угла поворота THETA

При программирование более высокие коэффициенты полиномов, равные нулю, могут быть опущены. Пример `PO[PHI]=a2` дает параболу для угла предварения `LEAD` .

Значение

Ориентация инструмента относительно траектории

`ORIPATH`:	Ориентация инструмента относительно траектории
`ORIPATHS`:	Ориентация инструмента относительно траектории, изгиб в характеристике ориентации сглаживается
`LEAD`:	Угол относительно вектора нормали плоскости, в образованной касательной к траектории и вектором нормали плоскости
`TILT`:	Вращение ориентации вокруг направления Z или вращение вокруг касательной к траектории
`THETA`:	Вращение вокруг направления инструмента в Z
`PO[PHI]`:	Полином ориентации для угла предварения LEAD
`PO[PSI]`:	Полином ориентации для бокового угла TILT
`PO[THT]`:	Полином ориентации для угла поворота THETA

Примечание

Угол поворота THETA

Для вращения инструмента с помощью третьей круговой оси в качестве оси ориентации вокруг себя самого необходима 6-осевая трансформация.

4.9.5.3 Интерполяция вращения инструмента относительно траектории (ORIROTC, THETA)

Интерполяция с векторами вращения

К запрограммированному с ORIROTC вращению инструмента относительно касательной к траектории возможна интерполяция вектора вращения и с программируемым через угол поворота THETA смещением. При этом для угла смещения с PO[THT] может быть запрограммирован полином макс. 5-го порядка.

Синтаксис

`N... ORIROTC`	Установка вращения инструмента относительно касательной к траектории
`N... A3= B3= C3= THETA=`значение	Определить вращение вектора ориентации
`N... A3= B3= C3= PO[THT]=(c2, c3, c4, c5)`	Интерполяция угла смещения с полиномом макс. 5-го порядка

Вращение может быть запрограммировано и одно в кадре, изменения ориентации при этом не происходит.

Значение

Интерполяция вращения инструмента относительно траектории при 6-осевой трансформации

`ORIROTC`:	Установить тангенциальный вектор вращения к касательной к траектории
`THETA=`значение:	Угол поворота в градусах, достигаемый в конце кадра
`THETA=`Θe:	Угол поворота с конечным углом Θ_e вектора вращения
`THETA=AC(...)`:	Покадровое переключение на абсолютное указание размера
`THETA=IC(...)`:	Покадровое переключение на указание составного размера
`PO[THT]=(c2, c3, c4, c5)`:	Интерполяция угла смещения с полиномом 5-го порядка

Примечание

Интерполяция вектора вращения ORIROTC

Если против направления ориентации инструмента должно быть установлено и вращение инструмента относительно касательной к траектории, то это возможно только при 6-осевой трансформации.

При активной ORIROTC

Вектор вращения ORIROTA не может быть запрограммирован. В случае программирования выводится ОШИБКА 14128 "Абсолютное программирование вращения инструмента при активной ORIROTC".

Направление ориентации инструмента при 3- до 5-осевой трансформации

Направление ориентации инструмента может быть запрограммировано как обычно при 3- до 5-осевой трансформации через угол Эйлера или угол RPY или векторы направления. Изменения ориентации инструмента в пространстве также возможны через программирование большой круговой интерполяции ORIVECT, линейной интерполяции осей ориентации ORIAXES, всех интерполяций на боковой поверхности конуса ORICONxx, а также интерполяции дополнительно к пространственной кривой с двумя контактными точками инструмента ORICURVE.

G....:	Указание типа движения круговых осей
X, Y, Z:	Указание линейных осей
ORIAXES:	Линейная интерполяция осей станка или осей ориентации
ORIVECT:	Большая круговая интерполяция (идентично ORIPLANE)
ORIMKS: ORIWKS:	Вращение в системе координат станка Вращение в системе координат детали Описание см. в гл. Вращения ориентации инструмента
A= B= C=:	Программирование позиции оси станка
ORIEULER:	Программирование ориентации через угол Эйлера
ORIRPY:	Программирование ориентации через угол RPY
A2= B2= C2=:	Программирование угла виртуальных осей
ORIVIRT1: ORIVIRT2:	Программирование ориентации через виртуальные оси ориентации (Определение 1), установка по MD $MC_ORIAX_TURN_TAB_1 (Определение 2), установка по MD $MC_ORIAX_TURN_TAB_2
A3= B3= C3=:	Программирование вектора направления оси направления
ORIPLANE:	Интерполяция в плоскости (большая круговая интерполяция)
ORICONCW:	Интерполяция на боковой поверхности конуса по часовой стрелке
ORICONCCW:	Интерполяция на боковой поверхности конуса против часовой стрелки
ORICONTO:	Интерполяция на боковой поверхности конуса, тангенциальный переход
A6= B6= C6=:	Программирование оси вращения конуса (нормированный вектор)
NUT=угол	Апертурный угол конуса в градусах
NUT=+179	Угол перемещения меньше или равен 180 градусам
NUT=-181	Угол перемещения больше или равен 180 градусам
ORICONIO:	Интерполяция на боковой поверхности конуса
A7= B7= C7=:	Промежуточная ориентация (программирование как нормированный вектор)
ORICURVE XH YH ZH, к примеру, с полиномом PO[XH]=(xe, x2, x3, x4, x5)	Интерполяция ориентации с задачей движения двух контактных точек инструмента. Кроме соответствующих конечных точек, возможно программирование дополнительных полиномов пространственных кривых.

Примечание

Если интерполяция ориентации инструмента с активной ORIAXES осуществляется через оси ориентации, то установка угла поворота относительно траектории выполняется только на конце кадра.

4.9.5.4 Сглаживание характеристики ориентации (ORIPATHS A8=, B8=, C8=)

При изменениях ориентации на контуре с постоянным ускорением, нежелательными являются прерывания движений по траектории, в особенности на углу контура. Возникающий из-за этого излом характеристики ориентации может быть сглажен через вставку собственного промежуточного кадра. Изменение ориентации выполняется с постоянным ускорением тогда, когда при переориентации активна и ORIPATHS. На этой фазе может быть выполнено движение отвода инструмента.

Изготовитель станка

Следовать указаниям изготовителя станка касательно возможно предопределенных машинных данных и установочных данных, с помощью которых активируется эта функция.

Через машинные данные может быть установлено, как интерпретируется вектор отвода:

1. В системе координат инструмента координата Z определяется через направление инструмента.
2. В системе координат детали координата Z определяется через активную плоскость.

Дополнительная информация по функции "Ориентация относительно траектории" см.: Описание функций – Трансформации.

Синтаксис

Для постоянных ориентаций инструмента относительно всей траектории на углу контура необходимы дополнительные данные программирования. Направление и длина хода этого движения программируется через вектор с компонентами A8 = X, B8 = Y, C8 = Z.

```
N... ORIPATHS A8=X B8=Y C8=Z
```

Объяснение

`ORIPATHS`:	Ориентация инструмента относительно траектории, излом в характеристике ориентации сглаживается.
`A8= B8= C8=`:	Компоненты вектора для направления и длины хода
`X, Y, Z`:	Движение отвода в направлении инструмента

Примечание

Программирование вектора направления A8, B8, C8

Если длина этого вектора равна нулю, то движение отвода не выполняется.

ORIPATHS

Относящаяся к траектории ориентация инструмента активируется с ORIPATHS. В ином случае ориентация посредством линейной большой круговой интерполяции переводится от стартовой к конечной ориентации.

4.9.6 Сжатие ориентации (COMPON, COMPCURV, COMPCAD, COMPSURF)

Программы ЧПУ, в которых активна трансформация ориентации (TRAORI) и запрограммированы ориентации инструмента (любого типа), могут быть сжаты при соблюдении заданных допусков.

Программирование

Ориентация инструмента

Если активна трансформация ориентации (TRAORI), программирование ориентации инструмента у 5-осевых станков возможно следующим образом (независимо от кинематики):

- Программирование**вектора** направления через:
  `A3=<...> B3=<...> C3=<...>`
- Программирование **угла** Эйлера или **угла** RPY через:
  `A2=<...> B2=<...> C2=<...>`

Вращения инструмента

Для **6-осевых** станков, дополнительно к ориентации инструмента, можно запрограммировать и вращение инструмента.

Программирование угла поворота осуществляется с:

`THETA=<...>`

См. "Вращения ориентации инструмента (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Страница 711)".

Примечание

Кадры УП, в которых дополнительно запрограммировано вращение, могут сжиматься только тогда, когда угол поворота изменяется **линейно**. Это означает, что для угла поворота не может быть запрограммирован полином с PO[THT] = (...).

Общая форма сжимаемого кадра УП

Поэтому общая форма сжимаемого кадра ЧПУ может выглядеть следующим образом:

```
N... X=<...> Y=<...> Z=<...> A3=<...> B3=<...> C3=<...> THETA=<...> F=<...>
```

или

```
N... X=<...> Y=<...> Z=<...> A2=<...> B2=<...> C2=<...> THETA=<...> F=<...>
```

Примечание

Значения позиция могут указываться напрямую (например, X90) или косвенно через присваивания параметров (например, X=R1*(R2+R3)).

Программирование ориентации инструмента через позиции круговых осей

Ориентация инструмента может быть указана и через позиции круговых осей, например, в форме:

```
N... X=<...> Y=<...> Z=<...> A=<...> B=<...> C=<...> THETA=<...> F=<...>
```

В этом случае сжатие выполняется двумя различными способами в зависимости от того, выполняется ли большая круговая интерполяция или нет. Если большая круговая интерполяция не осуществляется, то сжатое изменение ориентации отображается как обычно через осевые полиномы для круговых осей.

Точность контура

В зависимости от установленного режима сжатия (MD20482 $MC_COMPRESSOR_MODE), для гео-осей и осей ориентации при сжатии действует либо сконфигурированные, спец. для осей допуски (MD33100 $MA_COMPRESS_POS_TOL), либо следующие, настраиваемые через установочные данные спец. для канала допуски:

SD42475 $SC_COMPRESS_CONTUR_TOL (макс. погрешность контура)

SD42476 $SC_COMPRESS_ORI_TOL (макс. угловая погрешность для ориентации инструмента)

SD42477 $SC_COMPRESS_ORI_ROT_TOL (макс. угловая погрешность для угла поворота инструмента) (доступно только для 6-осевых станков)

Активация / деактивация

Функции компрессора включаются через модальные команды G COMPON, COMPCURV, COMPCAD или COMPSURF.

Завершение функции компрессора осуществляется с помощью COMPOF.

См. "Включение / выключение сжатия кадра ЧПУ (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF) (Страница 626)".

Примечание

Движение ориентации сжимается только при активной большой круговой интерполяции (т. е. изменение ориентации инструмента осуществляется в плоскости, образуемой стартовой и конечной ориентацией).

Большая круговая интерполяция выполняется при следующих условиях:

- MD21104 $MC_ORI_IPO_WITH_G_CODE = 0,
  ORIWKS активна и
  ориентация запрограммирована посредством векторов (с A3, B3, C3 или A2, B2, C2).
- MD21104 $MC_ORI_IPO_WITH_G_CODE = 1 и
  ORIVECT или ORIPLANE активна.
  Ориентация инструмента может быть запрограммирована либо как вектор направления, либо с позициями круговых осей. Если активна одна из команд ORICONxx или ORICURVE, или если запрограммированы полиномы для углов ориентации (PO[PHI] и PO[PSI]), то большая круговая интерполяция не выполняется.

Пример

В примере программы ниже сжимается окружность, приближенная через полигон. Ориентация инструмента при этом перемещается синхронно на боковой поверхности конуса. Хотя ход последовательных запрограммированных изменений ориентации является дискретным, функция компрессора создает ровную характеристику ориентации.

Программирование	Комментарий
`DEF INT ANZAHL=60`	
`DEF REAL RADIUS=20`	
`DEF INT COUNTER`	
`DEF REAL WINKEL`	
`N10 G1 X0 Y0 F5000 G64`	
`$SC_COMPRESS_CONTUR_TOL=0.05`	; Макс. погрешность контура = 0.05 мм
`$SC_COMPRESS_ORI_TOL=5`	; Макс. погрешность ориентации = 5 градусов
`TRAORI`	
`COMPCURV`	; Проходится круг, образуемый из полигонов. Ориентация при этом движется на конусе вокруг оси `Z` с апертурным углом в 45 градусов.
`N100 X0 Y0 A3=0 B3=-1 C3=1`	
`N110 FOR COUNTER=0 TO ANZAHL`	
`N120 WINKEL=360*COUNTER/ANZAHL`	
`N130 X=RADIUS*cos(WINKEL) Y=RADIUS*sin(WINKEL)`	
`A3=sin(WINKEL) B3=-cos(WINKEL) C3=1`	
`N140 ENDFOR`	

4.9.7 Включение/выключение сглаживания характеристики ориентации (ORISON, ORISOF)

"Сглаживание характеристики ориентации" включается/выключается в программе обработки детали G-командами группы 61. Команды действуют модально.

Условия

- Система с 5/6-осевой транскормацией.
- Функция компрессора COMPCAD активна.

Синтаксис

```
ORISON
...
ORISOF
```

Объяснение

ORISON:	Включение сглаживания характеристики ориентации
ORISOF:	Выключение сглаживания характеристики ориентации

Пример

Программный код	Комментарий
...	
TRAORI()	; Включение трансформации ориентации.
COMPCAD	; Включение функции компрессора COMPCAD
ORISON	; Включение сглаживания ориентации.
$SC_ORISON_TOL=1.0	; Макс. угловое отклонение ориентации инструмента = 1,0 градуса.
G91	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
...	
ORISOF	; Выключение сглаживания ориентации.
...	

Ориентация поворачивается на 90 градусов в плоскости XZ с -45 до +45 градусов. Из-за сглаживания характеристики ориентации, ориентация более не достигает макс. угловых значений в -45 или +45 градусов.

4.9.8 Кинематическая трансформация

4.9.8.1 Активация трансформации с торцевой стороны (TRANSMIT)

Трансформация с торцевой стороны (TRANSMIT) включается в программе обработки детали или синхронном действии через оператор `TRANSMIT`.

Синтаксис

```
TRANSMIT
TRANSMIT(<n>)
```

Объяснение

`TRANSMIT`:	Включить TRANSMIT с первым блоком данных TRANSMIT
`TRANSMIT(n)`:	Включить TRANSMIT с <n>-ным блоком данных TRANSMIT

Примечание

Активная в канале трансформация TRANSMIT выключается через:

- выключение трансформации: `TRAFOOF`
- включение другой трансформации: например, `TRACYL`, `TRAANG`, `TRAORI`

4.9.8.2 Включение трансформации боковой поверхности цилиндра (TRACYL)

Трансформация боковой поверхности цилиндра (TRACYL) включается в программе обработки детали или синхронном действии через оператор `TRACYL`.

Синтаксис

```
TRACYL(<d>)
TRACYL(<d>,<n>)
TRACYL(<d>,<n>,<k>)
```

Объяснение

TRACYL(<d>):	Включить TRACYL с первым блоком данных TRACYL и рабочим диаметром <d>	
TRACYL (<d>,<n>):	Включить TRACYL с <n>-ным блоком данных TRACYL и рабочим диаметром <d>	
<d>:	Базовый или рабочий диаметр Значение может быть больше 1.	
<n>:	TRACYL-номер блока данных (опция)	
	Диапазон значений:	1, 2
<k>:	Параметр <k> релевантен только для типа трансформации 514	
	k = 0:	без коррекции стенки паза
	k = 1:	с коррекцией стенки паза
	Если параметр не указывается, действует спараметрированная первичная установка: $MC_TRACYL_DEFAULT_MODE_<n> с <n> = TRACYL-номер блока данных	

Примечание

Активная в канале трансформация TRACYL выключается через:

- выключение трансформации: TRAFOOF
- включение другой трансформации: например, TRAANG, TRANSMIT, TRAORI

Пример

Программный код	Комментарий
...	
N40 TRACYL(40.)	; Включить TRACYL с первым блоком данных TRACYL и рабочим диаметром 40 мм.
...	

Дополнительная информация

Структура программы

Программа обработки детали для фрезерования паза с помощью трансформации TRACYL 513 (TRACYL с коррекцией стенки паза), как правило, состоит из следующих шагов:

1. Выбрать инструмент.
2. Выбрать TRACYL.
3. Выбрать подходящее смещение координат (FRAME).
4. Выполнить позиционирование.
5. Запрограммировать OFFN.

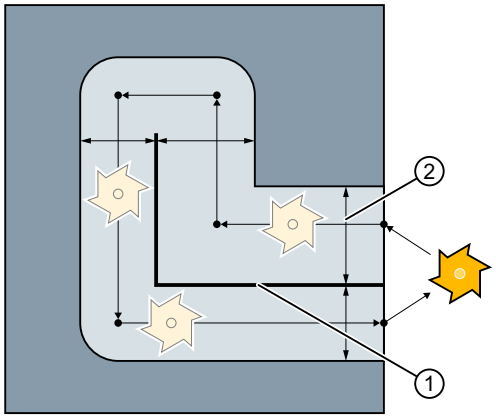
6. Выбрать КРИ.
7. Кадр подвода (ввод КРИ и подвод к стенке паза).
8. Контур линии центров паза.
9. Отключить КРИ.
10. Кадр отвода (вывод КРИ и отвод от стенки паза).
11. Выполнить позиционирование.
12. TRAFOOF.
13. Снова выбрать первоначальное смещение координат (FRAME).

Смещение контура (OFFN)

Для фрезерования пазов с трансформацией TRACYL 513, в программе обработки детали программируется линия центров паза и через адрес OFFN **половина ширины паза**.

Во избежание повреждения стенки паза, OFFN активируется только при выбранной коррекции на радиус инструмента.

Изменение OFFN внутри программы обработки детали возможно. Тем самым линия центра паза может быть смещена из центра:



① OFFN
② Запрограммированная траектория

Примечание

OFFN должно быть не меньше радиуса инструмента, чтобы исключить повреждение противоположной стенки паза.

Примечание

OFFN с TRACYL действует по иному, чем без TRACYL. Т.к. OFFN и без TRACYL учитывается при активной КРИ, то после TRAFOOF необходимо снова установить OFFN на ноль.

ВНИМАНИЕ
Действие OFFN зависит от типа трансформации При трансформации TRACYL 513 (TRACYL с коррекцией стенки паза) для OFFN программируется половина ширины паза. А при трансформации TRACYL 512 (TRACYL без коррекции стенки паза) значение из OFFN действует как припуск к КРИ.

Коррекция на радиус инструмента (КРИ)

При трансформации TRACYL 513 КРИ учитывается не относительно стенки паза, а к запрограммированному центру паза. Чтобы инструмент проходил слева от стенки паза, вместо G41 нужно запрограммировать оператор G42 или указать значение OFFN с отрицательным знаком.

Диаметр инструмента

С TRACYL и инструментом, диаметр которого меньше, чем ширина паза, создается геометрия стенки паза, отличная от таковой, создаваемой с помощью инструмента, диаметр которого равен ширине паза. Для улучшения точности рекомендуется выбирать диаметр инструмента лишь ненамного меньше, чем ширина паза.

Использование осей

Примечание

Следующие оси не могут использоваться в качестве позиционирующей или качающейся оси:

- Гео-ось в направлении обвода боковой поверхности цилиндра (ось Y)
- Дополнительная линейная ось для коррекции стенки паза (ось Z)

4.9.8.3 Включение косоугольной трансформации с программируемым углом (TRAANG)

Косоугольная трансформация с программируемым углом включается оператором `TRAANG` в программе обработки детали или синхронном действии.

Ограничение в случае кинематических цепей

При параметрировании на основе машинных данных угол α наклонной оси задан, в отличие от соответствующей оси координат, в прямоугольной системе координат через машинные данные. Угол α можно изменить при вызове через TRAANG(<α>). Изменение угла влечет за собой изменение кинематической модели, поэтому могут быть затронуты и другие приложения. Поэтому указание угла допустимо, только если угол совпадает с углом, который получается по кинематической цепи.

Примечание

Определение угла

Угол определен, только если наклонная ось получается в результате поворота вокруг одной из осей координат. Поэтому есть только одна гео-ось, не параллельная одной из осей координат и расположенная в главной плоскости.

Синтаксис

```
TRAANG
TRAANG()
TRAANG(,<n>)
TRAANG(<α>)
TRAANG(<α>,<n>)
```

Объяснение

TRAANG: TRAANG():	Включить TRAANG с первым блоком данных TRAANG и последним действительным углом <α>	
TRAANG(,<n>):	Включить TRAANG с <n>-ным блоком данных TRAANG и последним действительным углом <α>	
TRAANG(<α>):	Включить TRAANG с первым блоком данных TRAANG и углом <α>	
TRAANG(<α>,<n>):	Включить TRAANG с <n>-ным блоком данных TRAANG и углом <α>	
<α>:	Угол наклонной оси (опция)	
	Диапазон значений:	-90° < α < + 90°
	Если угол не указывается, действует первичная установка из машинных данных: MD2xxxx $MC_TRAANG_ANGLE_<n>	
<n>:	TRAANG-номер блока данных (опция)	
	Диапазон значений:	1, 2

Примечание

Активная в канале косоугольная трансформация TRAANG выключается через:

- выключение трансформации: TRAFOOF
- включение другой трансформации: например, TRACYL, TRANSMIT, TRAORI

Пример

Программный код	Комментарий
`N20 TRAANG(45)`	; Включить `TRAANG` с первым блоком данных `TRAANG` и углом 45°.

4.9.8.4 Наклонное врезание на шлифовальных станках (G5, G7)

G-команды G7 и G5 служат для упрощения программирования при наклонном врезном шлифовании на шлифовальных станках с трансформацией «Наклонная ось (TRAANG)», при этом при врезном шлифовании перемещается только наклонная ось.

При этом программируется только требуемая конечная позиция движения врезания в X и Z. Соответствующая стартовая позиция вычисляется ЧПУ для G7, исходя из актуальной позиции оси X, запрограммированной конечной позиции и угла α наклонной оси и выполняется подвод.

Стартовая позиция получается из точки пересечения обеих прямых:

- Прямая, параллельная ось Z, на расстоянии текущей позиции оси X
- Прямая параллельно наклонной оси через запрограммированную конечную позицию

При следующей G5 наклонная ось движется на запрограммированную конечную позицию.

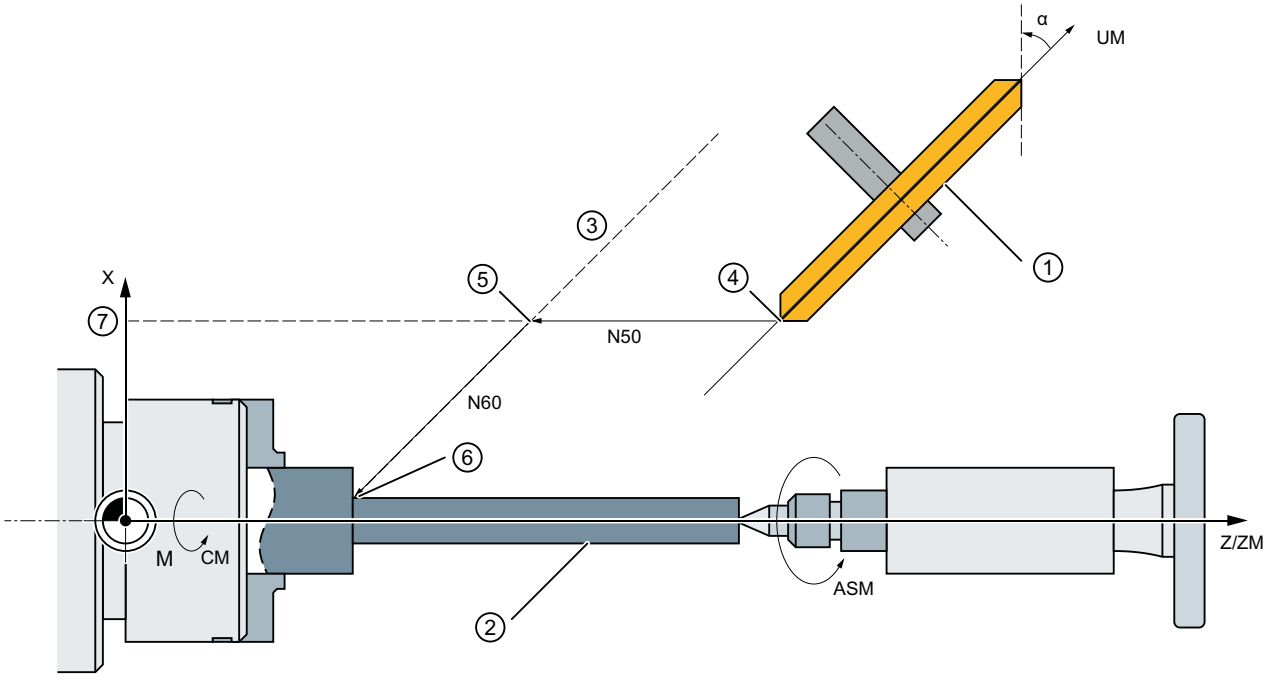
Синтаксис

```
G7 <Endpos_X> <Endpos_Z>
G5 <Endpos_X>
```

Объяснение

`G7`:	Вычисление стартовой позиции и подвод в нее для наклонного врезания
`G5`:	Перемещение наклонной оси на запрограммированную конечную позицию
`<Endpos_X>`:	Конечная позиция оси X
`<Endpos_Z>`:	Конечная позиция оси Z

Пример



① Шлифовальный круг
② Деталь
③ Параллельно наклонной оси через запрограммированную конечную позицию
④ Исходное положение
⑤ Врезание: Стартовая позиция
⑥ Врезание: Конечная позиция
⑦ Параллельно оси Z, на расстоянии текущей позиции оси X
X Гео-ось
Z Гео-ось
ZM Ось станка
UM Ось станка

Рисунок 4-6 Запрограммировать наклонную ось

Программный код	Комментарий
`N... G18`	; Выбор плоскости `XZ`.
`N40 TRAANG(45.0)`	; Включить трансформацию `TRAANG`, угол = 45°.
`N50 G7 X40 Z70 F4000`	; Вычисление стартовой позиции и подвод в нее.
`N60 G5 X40 F100`	; Перемещение наклонной оси на конечную позицию.
`N70 ...`	

4.9.9 Включить связанную трансформацию (TRACON)

Сконфигурированная последовательная связь трансформаций в программе обработки детали или синхронном действии включается операторами TRAFOON (Страница 754) или TRACON.

Возможность активации спараметрированной с кинематической цепью трансформации с TRACON необходимо спараметрировать в системной переменной $NT_TRACON_CHAIN.

Синтаксис

```
TRACON(<Trafo_Nr>,<Par_1>,...,<Par_n>,<Par_n+1>)
...
TRAFOOF
```

Объяснение

TRACON:	Включить связанную трансформацию Активированная ранее другая трансформация не явно выключается через TRACON().		
<трансф_№>:	Номер связанной трансформации		
	Тип:	INT	
	Диапазон значений:	0 ... 2	
	Значение:	0, 1	первая/единственная связанная трансформация
		2	вторая связанная трансформация
		не указано	равнозначно 0 или 1
		Примечание: Значения, отличные от 0, 1, 2, вызывают сигнализацию сбоя.	

<Par_1>, ..., <Par_n>, <Par_n+1>:	Параметры для связанных друг с другом трансформаций	
	<Par_1>, ..., <Par_n>	Параметры первой трансформации в цепочке Фактическое количество зависит от типа трансформации: • TRAORI: 2 параметра • TRACYL: 1–2 параметра
	<Par_n+1>	Параметры второй трансформации в цепочке (TRAANG) ≙ Угол наклонной оси
	При не установленных параметрах активируются предустановки или последние использовавшиеся параметры. Посредством запятой устанавливается ожидаемая последовательность обработки указанных параметров, если для предстоящих параметров должны действовать предустановки. В частности, при указании минимум одного параметра перед ним должна стоять запятая, даже если указание <трансф_№> (<Trafo_Nr>) не обязательно, например, TRACON(, 3.7).	
TRAFOOF:	Выключение последней включенной (связанной) трансформации	

Пример

Программный код	Комментарий
...	
N230 TRACON(1,45.)	; Включить первую связанную трансформацию.
	; Активная прежде трансформация отключается автоматически.
	; Угол для наклонной оси равен 45°.
...	
N330 TRACON(2,40.)	; Включить вторую связанную трансформацию.
	; Угол для наклонной оси равен 40°.
...	
N380 TRAFOOF	; Выключить вторую связанную трансформацию.
...	

4.9.10 Движение “от точки к точке” в декартовой системе координат

4.9.10.1 Включение / выключение движения “от точки к точке” в декартовой системе координат (PTP, PTPG0, PTPWOC, CP)

Движение “от точки к точке” в декартовой системе координат включается / выключается в программе ЧПУ командами группы G 49.

Команды действуют модально. По умолчанию задано перемещение по траектории в декартовой системе координат (CP).

В отличие от коммуникационного процессора при активном движении от точки к точке выполняется только трансформация конечной точки в декартовой системе координат, и оси станка перемещаются синхронно.

Чтобы конечную точку в декартовой системе координат можно было пересчитать в значения осей станка, дополнительно к указанию положения и угла необходима информация о маркировке,положений осей. Эти данные вводятся при помощи настраиваемых адресов STAT (Страница 734) и TU (Страница 738).

Условие

Трансформация TRAORI_DYN, TRAORI_STAT, TRANSMIT_K, RCTRA или ROBX активна (обычно TRAORI или TRANSMIT).

RCTRA или ROBX для кинематических цепей недоступны.

Синтаксис

```
PTP / PTPG0 / PTPWOC
...
CP
```

Объяснение

PTP:	Включение позиционного движения PTP Переход в запрограммированную позицию в декартовой системе координат в кадрах G0 и G1 осуществляется при помощи движения синхронной оси.
PTPG0:	Включение позиционного движения PTPG0 Переход в запрограммированную позицию в декартовой системе координат осуществляется при помощи движения синхронной оси только в кадрах G0. В кадрах G1 выполняется переключение на движение по траектории CP.
PTPWOC:	Включение позиционного движения ("от точки к точке") PTPWOC (возможно только при активной трансформации ориентации) Аналогично PTP, разумеется, без компенсирующих движений, обусловленных движением круговых осей и осей ориентации.
CP:	Выключение позиционного движения и включение движения по траектории CP С помощью CP выполняется движение по траектории в декартовой системе координат.

Примечание

PTPWOC

В сочетании с трансформацией RCTRA или ROBX использование PTPWOC не имеет смысла! RCTRA или ROBX для кинематических цепей недоступны.

Примеры

См.:

- Пример 1: Движение “от точки к точке" 6-осевого робота с трансформацией ROBX (Страница 740)
- Пример 2: Движение "от точки к точке" при базовой 5-осевой трансформации (Страница 742)
- Пример 3: PTPG0 и (Страница 742)

4.9.10.2 Задание положения шарниров (STAT)

Указания положения в декартовых координатах и ориентации инструмента недостаточно для однозначного определения положения станка, так как в одной и той же ориентации инструмента допускается несколько позиций сочленения. В зависимости от того, о какой кинематике идет речь, существует до 8 различных позиций сочленения. Эти различные позиции сочленения являются спец. для трансформаций.

Во избежание неточностей положение сочленений задается в адресе STAT.

Примечание

Управление учитывает запрограммированные значения STAT только при позиционном движении. При движении CP они игнорируются, так как при перемещении с активной трансформацией смена позиции, как правило, невозможна. При перемещении с активной командой CP позиция для конечной точки берется из начальной точки.

Синтаксис

`STAT=`<значение>

Объяснение

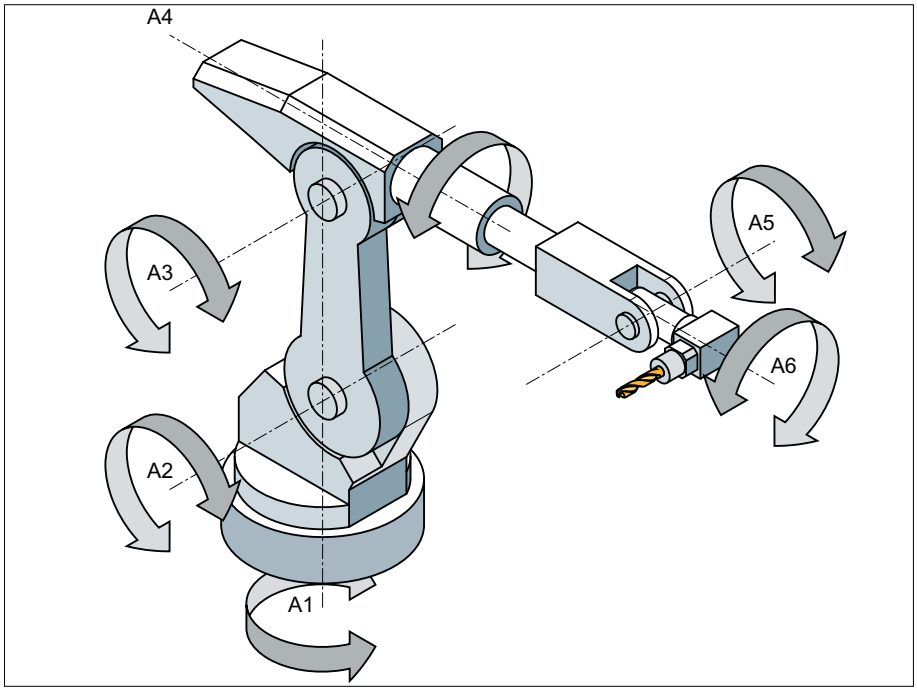
`STAT`:	Настраиваемый адрес для указания позиций сочленений
<значение>:	Двоичное или десятичное значение Содержит один бит для каждой из возможных позиций. Значение битов определяется соответствующей трансформацией.

Использование STAT необходимо проиллюстрировать на примере 6-осевого робота со сгибающейся рукой с фрезерным шпинделем. Кинематическая трансформация должна выполняться посредством трансформации ROBX (условие: Компилируемый цикл "Расширенная роботизированная трансформация RMCC/ROBX " загружен и активен).

Примечание

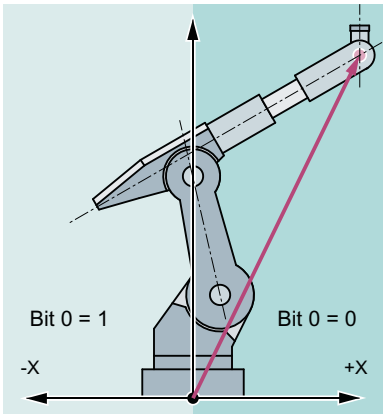
Пример с ROBX только с обычным параметрированием с машинными данными

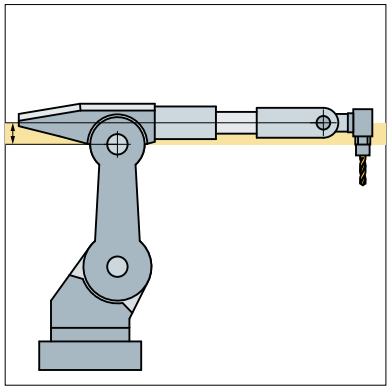
Пример возможен только при параметрировании с машинными данными, не с кинематическими цепями.



Оси A1, A2 и A3 — главные оси робота со сгибающейся рукой. Посредством главных осей оси A4, A5 и A6, обозначаемые "головные" или "ручные" оси, размещаются в рабочем пространстве. Благодаря дополнительным возможностям движения ручных осей фрезерный шпиндель можно ориентировать в пространстве так, как это необходимо для задач обработки. При этом при одновременной ориентации инструмента возможны различные положения шарнира.

Используемые для обработки положения шарнира программируются (бит 0 ... 2 настраиваемого адреса STAT:

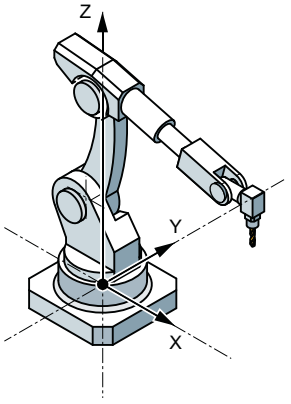
Бит 0	позиция точки пересечения ручных осей (A4, A5, A6)		
	=0	Базовая область (**Shoulder Right**) Робот находится в базовой области, когда значение X точки пересечения ручных осей положительно в системе координат A1.	 Пример: Точка пересечения "ручных" осей находится в базовой области
	=1	Верхняя область (**Shoulder Left**) Робот находится в верхней области, когда значение X точки пересечения ручных осей отрицательно в системе координат A1.	

Бит 1	позиция оси 3 Угол, при котором изменяется значение бита 1, зависит от типа робота. Для роботов, оси 3 и 4 которых пересекаются, справедливо:	
	=0	A3 < 0° (**Elbow Down**)
	=1	A3 ≥ 0° (**Elbow Up**)
	Примечание: Для роботов со смещением между осями 3 и 4 угол, при котором изменяется значение бита 1, зависит от величины этого смещения.	 Смещение между A3 и A4.
Бит 2	позиция оси 5	
	=0	A5 ≥ 0° (**No Handflip**)
	=1	A5 < 0° (**Handflip**)

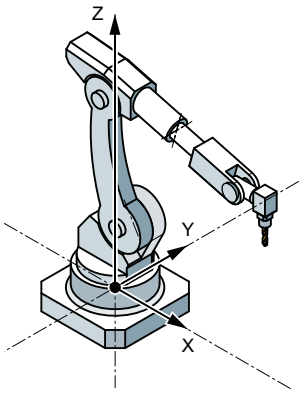
Пример программы:

Программный код	**Комментарий**
...	
`N14 T="T8MILLD20" D1`	`; $TC_DP3[1,1]=132.95`
`N16 ORIMKS`	
`N17 G1 PTP X1665.67 Y0 Z1377.405 A=0 B=0 C=0 STAT=... F2000`	; Значение `STAT` определяет положения шарнира (см. далее).
...	

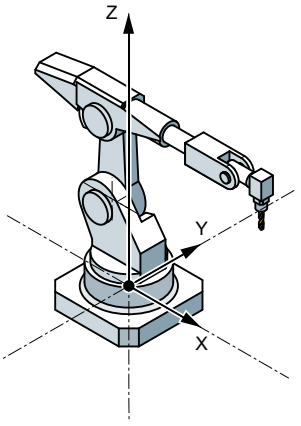
`STAT=1 ('B001')` → Shoulder Left
→ Elbow Down
→ No Handflip



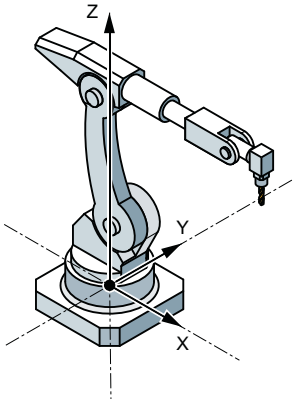
`STAT=2 ('B010')` → Shoulder Right
→ Elbow Up
→ No Handflip



`STAT=5 ('B101')` → Shoulder Left
→ Elbow Down
→ Handflip



`STAT=6 ('B110')` → Shoulder Right
→ Elbow Up
→ Handflip



TRANSMIT_K (обычно TRANSMIT)

В TRANSMIT_K используется адрес STAT, позволяющий исключить многозначность относительно полюса.

Если круговая ось должна вращаться на 180°, или контур при CP должен идти через полюс:

Бит 0	Относится только к $NT_POLE_SIDE_FIX[n] = 1 или 2 при параметрировании с машинными данными $MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 1 или 2:	
	=0	Круговая ось перемещается на +180° или поворачивается по часовой стрелке.
	=1	Круговая ось перемещается на -180° или поворачивается против часовой стрелки.
Бит 1	Относится только к $NT_POLE_SIDE_FIX[n] = 0 при параметрировании с машинными данными $MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 0:	
	=0	Выполняется перемещение через полюс. Круговая ось не поворачивается.
	=1	Выполняется поворот вокруг полюса. При этом имеет значение бит 0 в STAT.

4.9.10.3 Знак угла между осями (TU)

Чтобы оси вращения могли поворачиваться на угол между осями, величина которого больше +180° или меньше -180°, без применения специальной стратегии перемещения (например, промежуточная точка), необходимо задать в настраиваемом адресе TU знак угла между осями.

Примечание

Управление учитывает запрограммированные значения TU только при позиционном движении. При движении CP они игнорируются.

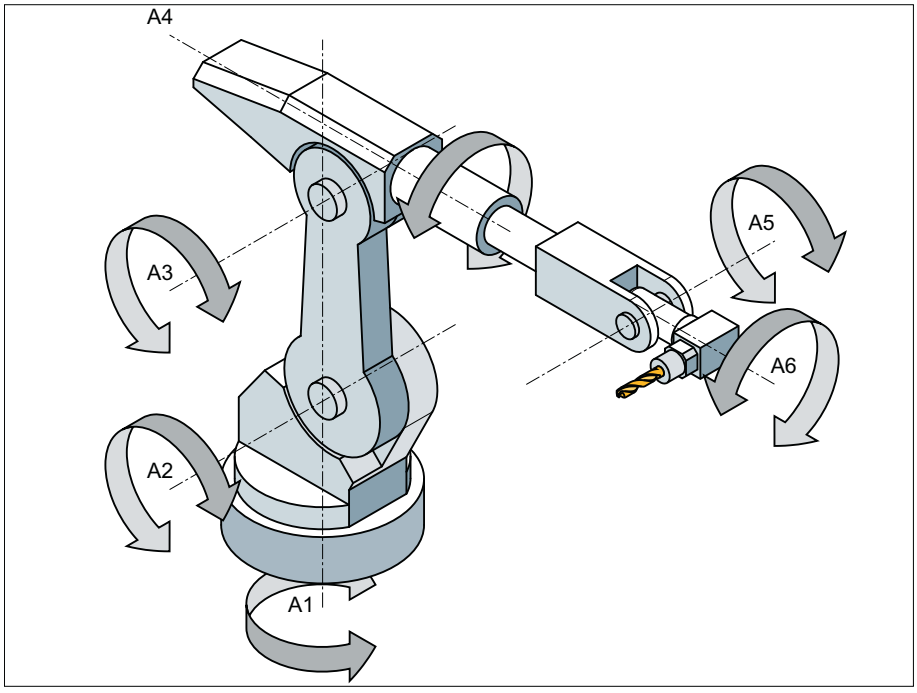
Синтаксис

`TU=<значение>`

Объяснение

`TU`:	Настраиваемый адрес для указания знака угла между осями			
<значение>:	Двоичное или десятичное значение Для каждой оси, участвующей в трансформации, имеется бит, указывающий знак угла между осями (θ) и, тем самым, направление перемещения.			
	Бит	=0	Знак перед осевым углом: +	Диапазон осевых углов: $0° \le \theta < 360°$
		=1	Знак перед осевым углом: -	Диапазон осевых углов: $-360° < \theta < 0°$

Пример: 6-осевой робот со сгибающейся рукой



Бит	Объяснение	Значение	Знак перед осевым углом	Осевой угол
Бит 0 [1]	Знак перед осевым углом A1	=0	+	≥ 0°
		=1	-	< 0°
Бит 1 [1]	Знак перед осевым углом A2	=0	+	≥ 0°
		=1	-	< 0°
Бит 2 [1]	Знак перед осевым углом A3	=0	+	≥ 0°
		=1	-	< 0°
Бит 3 [1]	Знак перед осевым углом A4	=0	+	≥ 0°
		=1	-	< 0°
Бит 4 [1]	Знак перед осевым углом A5	=0	+	≥ 0°
		=1	-	< 0°
Бит 5 [1]	Знак перед осевым углом A6	=0	+	≥ 0°
		=1	-	< 0°

[1] Фактические номера битов TU получаются из номеров осей канала осей робота! В примере оси робота (с A1 по A6) — это первые шесть осей в канале, следовательно, используются биты TU 0 ... 5. При другом соотнесении осей канала робота номера битов TU осей робота соответствующим образом изменятся (z. B.: оси робота — . это оси канала с 3 по 8, т. е. для осей робота используются биты TU 2 ... 7).

TU=19 (соответствует TU='B010011) будет также обозначать:

Бит	Значение		Осевой угол
0	=1	⇒	$\theta_{A1} < 0°$
1	=1	⇒	$\theta_{A2} < 0°$

2	=0	⇒	$\theta_{A3} \geq 0°$
3	=0	⇒	$\theta_{A4} \geq 0°$
4	=1	⇒	$\theta_{A5} < 0°$
5	=0	⇒	$\theta_{A6} \geq 0°$

Примечание

Для осей с диапазоном перемещения > ±360° перемещение всегда выполняется по кратчайшему пути, т.к. позиция оси не может быть однозначно определена через TU-информацию.

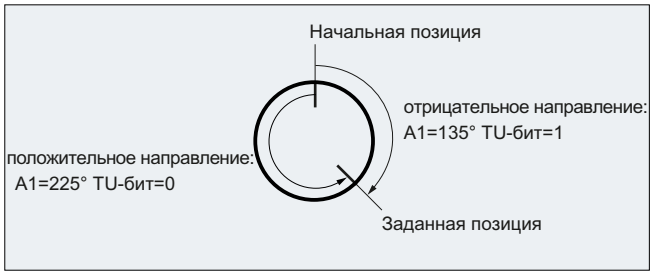
Если для позиции не программируется TU, то в зависимости от MD30455 $MA_MISC_FUNCTION_MASK проходится более короткий или более длинный путь.

TRANSMIT_K (обычно TRANSMIT)

При позиционном перемещении при активном TRANSMIT адрес TU не имеет значения!

Пример

Подвод к указанной на следующем рисунке позиции круговой оси возможен в отрицательном или положительном направлении. По адресу A1 программируется положение угла. Однозначное направление перемещения задается только параметром TU.



4.9.10.4 Пример 1: Движение “от точки к точке" 6-осевого робота с трансформацией ROBX

В следующем примере использования показываются движение "от точки к точке" в декартовой системе координат и связанные с ним команды ЧПУ.

Примечание

Только с обычным параметрированием с машинными данными

Пример не работает с кинематическими цепями.

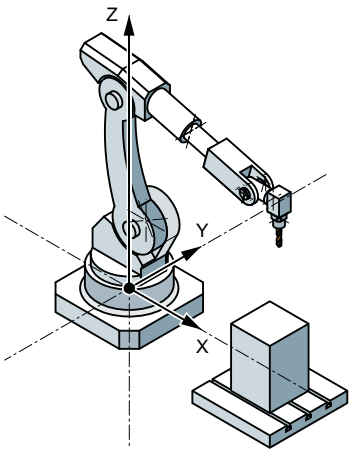


Рисунок 4-7 6-осевой робот со сгибающейся рукой и фрезерным шпинделем

```
N1 G90
N2 T="T8MILLD20" D1 M6
N3 TRAORI
;$P_UIFR[1]=CTRANS(X,1500,Y,0,Z,400):CROT(X,0,Y,0,Z,-90)
N4 G54
N5 M3 S20000
N6 ORIWKS
N7 ORIVIRT1
N8 CYCLE832(0.01,_FINISH,1)
;HOME
N9 TRAFOOF
N10 G0 RA1=0.0000 RA2=-90.0000 RA3=90.0000 A=0.0000 B=90.0000 C=0.0000
N11 TRAORI
N12 G54
N13 G0 PTP X1369.2426 Y956.7528 Z502.5517 A=135.5761 B=-33.2223
C=161.1435 STAT='B010' TU='B001011'
N14 G0 X1355.1242 Y1014.9394 Z424.9695 A=135.8491 B=-33.1439
C=160.9941 STAT='B010' TU='B001011'
N15 G1 CP X1354.8361 Y1016.1269 Z423.3862 A=136.0635 B=-33.0819 C=160.8770
F1000
N16 G1 X1336.4283 Y1016.1269 Z426.6311 A=136.0484 B=-32.2151 C=160.9643
F2000
N17 G1 X1317.9831 Y1016.1269 Z429.6730 A=136.0175 B=-31.3394 C=161.0655
;HOME
N18 TRAFOOF
N19 G0 RA1=0.0000 RA2=-90.0000 RA3=90.0000 A=0.0000 B=90.0000 C=0.0000
N20 M30
```

4.9.10.5 Пример 2: Движение "от точки к точке" при базовой 5-осевой трансформации

Допущение: основой является прямоугольная кинематика CA.

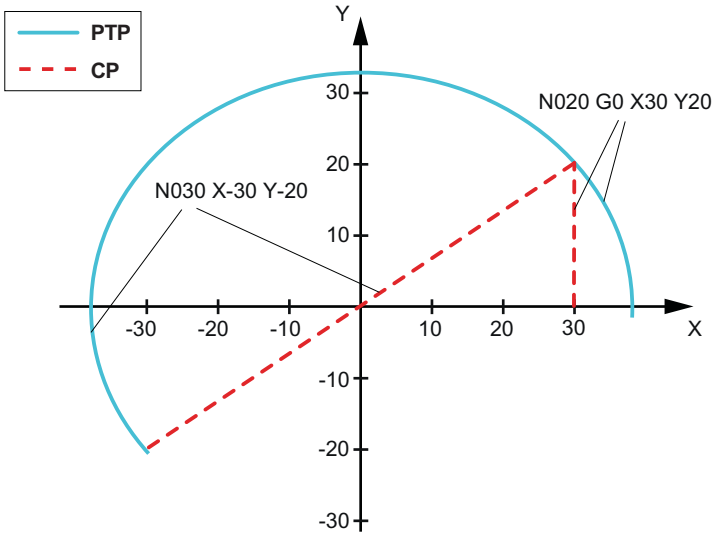
Программный код	Комментарий
TRAORI	; Трансформация кинематики CA вкл
PTP	; Включить движение "от точки к точке"
N10 A3=0 B3=0 C3=1	; позиция круговой оси C=0 A=0
N20 A3=1 B3=0 C3=1	; позиция круговой оси C=90 A=45
N30 A3=1 B3=0 C3=0	; позиция круговой оси C=90 A=90
N40 A3=1 B3=0 C3=1 **STAT=1**	; позиция круговой оси C=270 A=–45

Выбрать однозначное положение подвода к позиции круговых осей:

При этом в кадре N40 круговые оси через программирование **STAT=1** двигаются по более длинному пути от своей стартовой точки (C=90, A=90) к конечной точке (C=270, A=–45). При **STAT=0** круговые оси, напротив, двигаются к конечной точке (C=90, A=45) более коротким путем.

4.9.10.6 Пример 3: PTPG0 и

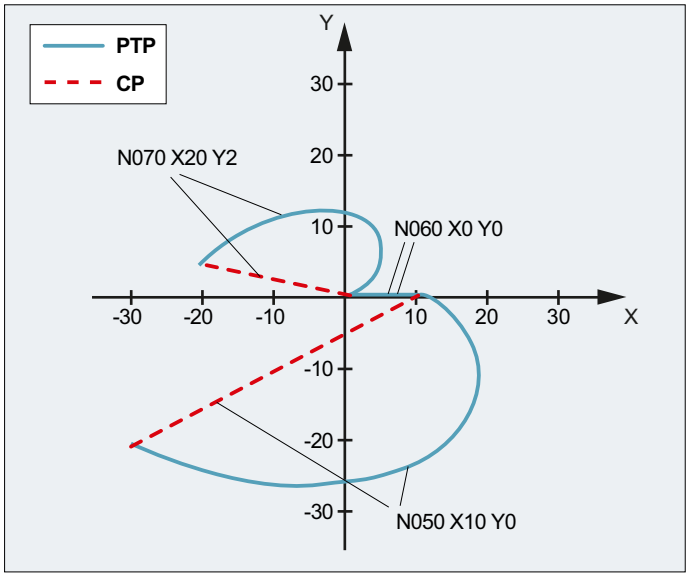
Обход полюса командами PTPG0 и



Программный код	Комментарий
N001 G0 X30 Z0 F10000 T1 D1 G90	; Исходная позиция, абсолютный размер
N002 SPOS=0	
N003 **TRANSMIT**	; трансформация TRANSMIT
N010 **PTPG0**	; к каждому G0-кадру автоматически PTP и затем снова CP.
N020 G0 X30 Y20	
N030 X-30 Y-20	

Программный код	Комментарий
N120 G1 X30 Y20	
N110 X30 Y0	
M30	

Выход из полюса с PTPG0 и TRANSMIT



Программирование	Комментарий
N001 G0 X90 Z0 F10000 T1 D1 G90	; исходное положение
N002 SPOS=0	
N003 **TRANSMIT**	; трансформация TRANSMIT
N010 **PTPG0**	; к каждому G0-кадру автоматически PTP и затем снова CP.
N020 G0 X90 Y60	
N030 X-90 Y-60	
N040 X-30 Y-20	
N050 X10 Y0	
N060 X0 Y0	
N070 X-20 Y2	
N170 G1 X0 Y0	
N160 X10 Y0	
N150 X-30 Y-20	
M30	

4.9.11 Граничные условия при выборе трансформации

Функция

Выбор трансформаций возможен через программу обработки детали или MDA. При этом учитывать:

- Промежуточный кадр движения не вставляется (фаски/радиусы).
- Последовательность кадров сплайна должна быть завершена; если нет, то появляется сообщение.
- Точная коррекция инструмента должна быть отключена (FTOCOF); если нет, то появляется сообщение.
- Коррекция радиуса инструмента должна быть отключена (G40); если нет, то появляется сообщение.
- Активированная коррекция длин инструмента передается СЧПУ в трансформацию.
- Действующий перед трансформацией текущий фрейм отключается СЧПУ.
- Активное ограничение рабочего поля отключается СЧПУ для задействованных трансформацией осей (соответствует WALIMOF).
- Контроль защищенных областей отключается.
- Режим управления траекторией и перешлифовка прерываются.
- Все указанные в машинных данных оси должны быть синхронизированы с привязкой к кадру.
- Выполняется обратный переход перешедших осей, если нет, то появляется сообщение.
- Для зависимых осей выводится сообщение.

Смена инструмента

Смена инструмента допускается только при отмененной коррекции на радиус инструмента.

Смена коррекции длин инструмента и включение/выключение коррекции радиуса инструмента не могут программироваться в одном кадре.

Смена фрейма

Все операторы, относящиеся только к базовой кинематической системе, разрешены (FRAME, коррекция радиуса инструмента). Но смена фрейма при G91 (составной размер) – в отличие от не активной трансформации – не обрабатывается отдельно. Исполняемый инкремент обрабатывается в системе координат детали нового фрейма – независимо от того, какой фрейм действовал в предшествующем кадре.

Исключения

Затронутые трансформацией оси не могут использоваться:

- как ось Preset (аварийное сообщение),
- для подвода к фиксированной точке (ошибка),
- для реферирования (аварийное сообщение).

4.9.12 Отключение трансформации (TRAFOOF)

С помощью предопределенной процедуры TRAFOOF все активные трансформации и фреймы отключаются.

Примечание

Для отмены трансформации действуют те же граничные условия (Страница 744) что и для выбора.

Необходимые в дальнейшем фреймы должны быть активированы посредством повторного программирования.

Синтаксис

```
...
TRAFOOF
```

Значение

`TRAFOOF`:	Отключение всех активных трансформаций / фреймов

4.10 Кинематические цепочки

4.10.1 Удаление компонентов (DELOBJ)

Функция `DELOBJ()` "удаляет" компоненты путем сброса назначенных системных переменных на значения по умолчанию:

- Элементы кинематических цепочек
- Защищенные области, элементы защищенных областей и соударяемые пары
- Данные трансформации

Синтаксис

```
[<RetVal>=] DELOBJ(<CompType>[,,,<NoAlarm>)])
```

```
[<RetVal>=] DELOBJ(<CompType>,<Index1>[,,<NoAlarm>])
[<RetVal>=] DELOBJ(<CompType>[,<Index1>][,<Index2>][,<NoAlarm>])
```

Значение

DELOBJ:	Удаление элементов кинематических цепочек, защищенных областей, элементов защищенных областей, соударяемых пар и данных трансформаций	
<CompType>:	Тип удаляемого компонента	
	Тип данных:	STRING
	Величина: "KIN_CHAIN_ELEM" **значение**: Системные переменные всех кинематических элементов: $NK_...	
	Величина: "KIN_CHAIN_SWITCH" **значение**: Системная переменная $NK_SWITCH[<i>]	
	Величина: "KIN_CHAIN_ALL" **Значение:** Все кинематические элементы и коммутаторы. Равнозначно последовательному вызову DELOBJ с KIN_CHAIN_ELEM и KIN_CHAIN_SWITCH	
	Величина: PROT_AREA **значение**: Системные переменные защищенных областей: • $NP_PROT_NAME • $NP_CHAIN_NAME • $NP_CHAIN_ELEM • $NP_1ST_PROT	
	Величина: PROT_AREA_ELEM **значение**: Системные переменные элементов защищенных областей станка и / или автоматических защищенных областей инструмента: • $NP_NAME • $NP_NEXT • $NP_NEXTP • $NP_COLOR • $NP_D_LEVEL • $NP_USAGE • $NP_TYPE • $NP_FILENAME • $NP_PARA • $NP_OFF • $NP_DIR • $NP_ANG	
	Величина: PROT_AREA_COLL_PAIRS **значение**: Системные переменные соударяемых пар: • $NP_COLL_PAIR • $NP_SAFETY_DIST	
	Величина: PROT_AREA_ALL **значение**: Все защищенные области, элементы защищенных областей и соударяемые пары (системные переменные $NP_...) Равнозначно при последовательном вызове DELOBJ с PROT_AREA, PROT_AREA_ELEM и PROT_AREA_COLL_PAIRS	
	Величина: TRAFO_DATA **значение**: Системные переменные всех трансформаций $NT_...	

<Index1>:	Индекс первого удаляемого компонента (**опция**)	
	Тип данных:	INT
	Значение по умолчанию:	-1
	Диапазон значений:	-1 ≤ x ≤ (макс. число сконфигурированных компонентов -1)
	Значение	**Значение**
	0, 1, 2, ...	Индекс удаляемого компонента
	-1	Все компоненты указанного типа удаляются. <Index2> не обрабатывается.
<Index2>:	Индекс последних удаляемых компонентов (**опция**) Если <Index2> не запрограммирован, то удаляются только системные переменные упомянутого в <Index1> компонента.	
	Тип данных:	INT
	Значение по умолчанию:	Удаляются только системные переменные упомянутого в <Index1> компонента.
	Диапазон значений:	<Index1> < x ≤ (макс. число сконфигурированных компонентов -1)
<NoAlarm>:	Подавление аварийных сообщений (**опция**)	
	Тип данных:	BOOL
	Значение по умолчанию:	false
	Значение	**Объяснение**
	false	В случае ошибки (<RetVal> < 0) выполнение программы останавливается, и сигнализируется аварийное сообщение..
	TRUE	В случае ошибки выполнение программы не останавливается и аварийное сообщение не отображается. Пример использования: Специализированная реакция согласно возвращаемому значению
<RetVal>:	Возвращаемое значение функции	
	Тип данных:	INT
	Диапазон значений:	0, -1, -2, ... -7
	Значение	**Значение**
	0	Нет ошибок
	-1	Вызов функции без параметров. Как минимум, должен быть указан параметр <CompType>.
	-2	<CompType> обозначает неизвестный компонент
	-3	<Index1> меньше -1
	-4	<Index1> больше сконфигурированного числа компонентов
	-5	<Index1> имеет при удалении **группы** компонентов значение, отличное от -1
	-6	<Index2> меньше <Index1>
	-7	<Index2> больше сконфигурированного числа компонентов
	-8	Прав для записи недостаточно

4.10.2 Определение индекса по имени (NAMETOINT)

В поля системных переменных типа STRING внесены пользовательские имена. На основании идентификатора системных переменных и имени функция `NAMETOINT()` определяет относящееся к имени значение индекса, под которым оно записано в поле системной переменной.

Синтаксис

```
<RetVal> = NAMETOINT(<SysVar>,<Name>[,<NoAlarm>])
```

Значение

`NAMETOINT`:	Определение индекса системной переменной	
`<SysVar>`:	Имя поля системной переменной типа STRING	
	Тип данных:	STRING
	Диапазон значений:	Имена всех полей системной переменной ЧПУ типа STRING
`<Name>`:	Цепочка символов или имя, к которому должен быть найден индекс системной переменной.	
	Тип данных:	STRING
`<NoAlarm>`:	Подавление аварийных сообщений (**опция**)	
	Тип данных:	BOOL
	Значение по умолчанию:	FALSE
	Значение	**Значение**
	TRUE	В случае ошибки выполнение программы не останавливается и аварийное сообщение не отображается. Пример использования: Специализированная реакция согласно возвращаемому значению
	FALSE	В случае ошибки (<RetVal> < 0) выполнение программы останавливается, и сигнализируется аварийное сообщение.
`<RetVal>`:	Индекс системной переменной или сообщение об ошибке	
	Тип данных:	INT
	Диапазон значений:	-1 ≤ x ≤ (макс. число сконфигурированных компонентов -1)
	Значение	**Значение**
	≥ 0	Искомое имя найдено под указанным индексом системной переменной.
	-1	Искомое имя не было найдено или возникла ошибка.

Пример

Программный код	**Комментарий**
`DEF INT INDEX`	
`$NP_PROT_NAME[27] = «кожух»`	

Программный код	Комментарий
...	
INDEX = NAMETOINT(«$NP_PROT_NAME»,«кожух»)	; INDEX == 27

4.11 Предотвращение столкновений с кинематическими цепочками

Примечание

Защищенные области

Названные в последующих главах защищенные области относятся к функции "Геометрическое моделирование станка"

Информация по этой функции: Описание функций — Контроль и компенсация.

4.11.1 Проверка на соударяемую пару (COLLPAIR)

4.11.2 Запрос повторного вычисления модели станка для предотвращения столкновений (PROTA)

Если системная переменная кинематической цепи $NK_..., геометрического моделирования станка или предотвращения столкновений $NP_... описываются в программе обработки детали, то затем необходимо вызвать процедуру PROTA, чтобы принять изменение модели станка для предотвращения столкновений внутри ЧПУ.

Синтаксис

```
PROTA[(<Par>)]
```

Объяснение

PROTA:	Запрос повторного вычисления модели станка для предотвращения столкновений		
	• Вызывает остановку предварительной обработки. • Должен стоять автономно в кадре.		
<Par>:	Параметры (**опционально**)		
	Тип данных:	STRING	
	Значение:	---	Без параметров. Выполняется повторный расчет модели станка. Состояния защищенных областей сохраняются.
		"R"	Выполняется повторный расчет модели станка. Защищенные области приводятся в состояние инициализации согласно $NP_INIT_STAT.

Граничные условия

Моделирование

Процедуру PROTA нельзя использовать в программах обработки детали в сочетании с моделированием (simNC).

Пример: Предотвращение вызова PROTA,, пока моделирование активно.

Программный код	Комментарий
...	
IF $P_SIM == FALSE	; IF моделирование не активно
PROTA	; THEN заново рассчитать модель столкновений
ENDIF	; ENDIF
...	

См. также

Установка состояния защищенной области (PROTS) (Страница 751)

4.11.3 Установка состояния защищенной области (PROTS)

Процедура PROTS(...) устанавливает указанное значение состояния защищенных областей.

Синтаксис

```
PROTS(<State>[,<Name_1>,...,<Name_n>])
```

Объяснение

PROTS:	Установка состояния защищенных областей		
	• Должна стоять автономно в кадре.		
<State>:	Состояние, на которое должны быть установлены указанные защищенные области		
	Тип данных:	CHAR	
	Значение:	"A" или "a"	Состояние: активна
		"I" или "i"	Состояние: не активна
		"P"или "p"	Состояние: предварительно активировано или управляется через PLC [1)]
		"R"или "r"	Состояние: Внутреннее для ЧПУ значение состояния инициализации [2)]

<Name_1> ... <Name_n>:	Имя одной или нескольких защищенных областей, которые должны быть установлены на указанное состояние (**опционально**) Если имя не указано, то указанное состояние устанавливается для всех определенных защищенных областей.	
	Тип данных:	STRING
	Диапазон значений:	Спараметрированные имена защищенных областей
	Примечание: Максимальное число защищенных областей, которые могут быть указаны как параметры, зависит только от максимально возможного числа символов на строку программы.	
[1] Активация / деактивация осуществляется через: DB10.DBX234.0 - DBX241.7 [2] Для состояния устанавливается внутреннее для ЧПУ значение состояния инициализации, т. е. значение, которое системная переменная $NP_INIT_STAT имела на момент последнего вызова PROTA(...) (Страница 750).		

4.11.4 Определение интервала между двумя защищенными областями (PROTD)

Функция `PROTD(...)` вычисляет отступ двух защищенных областей.

Свойства функции:

- Расчет интервала выполняется независимо от состояния защищенных областей (активированы, деактивированы, предактивированы).
- Для вычисления отступа двух защищенных областей используются только элементы защищенных областей, обозначенные $NP_USAGE = "C" или "A". Элементы защищенной области, обозначенные посредством $NP_USAGE = "V", не учитываются.
- Защищенные области, в которых элементы защищенной области обозначены посредством $NP_USAGE = "V", не могут быть использованы для вычисления отступа.
- Расчет интервала выполняется с действительными на конце предшествующего кадра позициями.
- Наложения, учитываемые при главном ходе (например, DRF-смещение или внешние смещения нулевой точки) включаются в расчет отступа с действующими на **момент интерполяции** функции значениями.

Примечание

Синхронизация

При использовании функции `PROTD(...)` ответственность за возможно необходимую синхронизацию главного хода и предварительной обработки с помощью остановки предварительной обработки `STOPRE` лежит только на пользователе.

Столкновение

Если между указанными защищенными областями имеет место столкновение, то функция возвращает отступ 0,0. Столкновение имеет место, если обе защищенные области соприкасаются или имеют наложение.

Безопасное расстояние для проверки столкновений (MD10622 $MN_COLLISION_SAFETY_DIST) при вычислении отступа не учитывается.

Синтаксис

```
[<RetVal> =] PROTD([<Name_1>],[<Name_2>],VAR <Vector>[,<System>])
```

Объяснение

PROTD:	Рассчитывает отступ между двумя указанными защищенными областями		
	• Должна стоять автономно в кадре.		
<RetVal>:	Возвращаемое значение функции: Абсолютное значение интервала между двумя защищенными областями или 0,0 при столкновении (см. выше: раздел Столкновение)		
	Тип данных:	REAL	
	Диапазон значений:	0,0 ≤ x ≤ +макс. REAL-значение	
<Name_1>, <Name_2>:	Имена обеих защищенных областей, отступ между которыми необходимо вычислить (**опционально**)		
	Тип данных:	STRING	
	Диапазон значений:	Спараметрированные имена защищенных областей	
	Значение по умолчанию:	"" (пустая строка) Если защищенные области не указаны, то функция вычисляет наименьшее текущее расстояние из всех в содержащихся в модели столкновений активированных и предактивированных защищенных областей.	
<Vector>:	Возвращаемое значение: 3-мерный вектор расстояния от защищенной области <Name_2> к защищенной области <Name_1> с: • <Vector>[0]: X-координата в мировой системе координат • <Vector>[1]: Y-координата в мировой системе координат • <Vector>[2]: Z-координата в мировой системе координат При столкновении: <Vector> == нулевой вектор		
	Тип данных:	VAR REAL [3]	
	Диапазон значений:	<Vector> [n]: 0,0 ≤ x ≤ ±макс. REAL-значение	
<System>:	Система единиц (дюймовая / метрическая) для интервала и вектора расстояния (**опция**)		
	Тип данных:	BOOL	
	Значение:	FALSE (по умолчанию)	Система единиц согласно текущей активной G-команде из G-группы 13 (G70, G71, G700, G710).
		TRUE	Система единиц согласно установленной основной системе: MD52806 $MN_ISO_SCALING_SYSTEM

4.12 Трансформации с кинематическими цепями

4.12.1 Активация трансформации (TRAFOON)

Трансформацию, определенную посредством кинематических цепей, активирует предварительно определенная процедура TRAFOON. В кадре должен находится только вызов.

Примечание

В качестве альтернативы трансформацию, определенную посредством кинематических цепей, можно активировать и традиционными языковыми командами, например, TRAORI или TRANSMIT. Для этого в системной переменной $NT_TRAFO_INDEX должно быть внесено соответствующее значение, не равное нулю.

Дополнительная информация по $NT_TRAFO_INDEX представлена в "Справочник таблиц — Системные переменные".

Синтаксис

```
TRAFOON(<Trafoname>,<Diameter>,<k>)
```

Объяснение

TRAFOON:	Процедура активации трансформации, определенной посредством кинематических цепей		
<Trafoname>:	имя блока данных трансформации		
	Тип данных:	STRING	
	Диапазон значений:	все определенные посредством $NT_NAME имена блоков данных трансформаций	
	Указание: Имя блока данных трансформации должно быть однозначным. Оно должно встречаться в $NT_NAME только один раз.		
<Diameter>:	базовый или рабочий диаметр (только TRACYL)		
	Тип данных:	REAL	
	Значение должно быть > 1.		
<k>:	определяет использование коррекции стенки паза (только TRACYL).		
	Тип данных:	BOOL	
	Значение:	FALSE	без коррекции стенки паза
		TRUE	с коррекцией стенки паза
	Соответствует типу трансформации 514 TRACYL (коррекция стенки паза программируемая). Если <k> не задано, действует параметризированная настройка бита 10 в $NT_CNTRL[<n>].		

Пример

Программный код	Комментарий
`TRAFOON["Trans_1"]`	Активирует трансформацию под именем `Trans_1`.

4.12.2 Модифицирование трансформации ориентации после замеров станка (CORRTRAFO)

В станках с трансформациями ориентации, которые были определены посредством кинематических цепей, пользователь может использовать предварительно определенную функцию CORRTRAFO для модифицирования векторов смещения или векторов направления осей ориентации в кинематической модели станка после замеров станка.

Примечание

Записанные при помощи функции CORRTRAFO значения коррекции не сразу вступают в силу в трансформации. Значения коррекции вступают в силу только после отмены транформации, применения NEWCONF и выбора трансформации.

Синтаксис

```
<Corr_Status> = CORRTRAFO(<Corr_Vect>, <Corr_Index>, <Corr_Mode>,
[ <No_Alarm>])
```

Объяснение

CORRTRAFO:	Вызов функции

<Corr_Status>:	Возвращаемое значение функции		
	Тип данных:	INT	
	Значения:	0	Функция была выполнена без ошибок.
		1	Ни одна из трансформаций не активна.
		2	Активная на данный момент трансформация не является трансформацией ориентации.
		3	Активная трансформация ориентации не была определена посредством кинематических цепей.
		10	Параметр вызова <Corr_Index> отрицательный.
		11	Параметр вызова <Corr_Mode> отрицательный.
		12	Недействительная ссылка на сегмент подцепочки (1-я позиция <Corr_Index>). Значение не должно быть больше значения количества осей ориентации в подцепочке.
		13	Недействительная ссылка на ось ориентации подцепочки (1-я позиция <Corr_Index>). Значение должно быть меньше значения количества осей ориентации в подцепочке.
		14	Недействительная ссылка на подцепочку (10-я позиция <Corr_Index>). Допустимы только значения 0 и 1 (ссылка на цепь детали или цепь инструмента). Этот номер ошибки отображается также, если подцепочка, на которую ссылается <Corr_Index>, не существует.
		15	На сегменте, на который выполняется ссылка параметром <Corr_Index>, элемент коррекции не определен ($NT_CORR_ELEM_P или $NT_CORR_ELEM_T).
		20	Недействительный режим коррекции (1-я позиция <Corr_Mode>). Допустимы только значения 0 и 1.
		21	Недействительный режим коррекции (10-я и/или 100-я позиция <Corr_Mode>). При записи осевого направления только 1-я позиция должна быть не равна нулю.
		30	100-я позиция <Corr_Mode> недействительна. Допустимы только значения 0 и 1.
		31	1000-я позиция <Corr_Mode> недействительна. Допустимы только значения 0 и 1.
		40	Вектор направления, который необходимо принять как осевое направление, является нулевым вектором. Эта ошибка может возникнуть, только если 1000-я позиция <Corr_Mode> равна 0. Если 1000-я позиция этого параметра равна 1 (контроль максимальной коррекции деактивирован), может быть записан только нулевой вектор.
		41	При коррекции вектора смещения отклонение по сравнению с текущим значением, по меньшей мере, в одной координате больше максимального значения, заданного в установочных данных SD41610 $SN_CORR_TRAFO_LIN_MAX. Параметр <Corr_Vect> перезаписывается через вектор ошибки. То же действительно, и если обработку прервало аварийное сообщение (см. параметр <No_Alarm>). Вектор ошибки содержит в компонентах, значение коррекции которых превысило допустимый предел, разницу (с

			учётом знака) установленного значения коррекции и предела. Содержание компонентов, не превысивших свой предел, равно нулю.
		42	При коррекции вектора направления угловая погрешность по сравнению с текущим направлением больше максимального значения, заданного в установочных данных SD41611 $SN_CORR_TRAFO_DIR_MAX.
		43	Попытка записи системной переменной была отклонена из-за отсутствия прав записи.
<Corr_Vect>:	Вектор коррекции Содержание вектора коррекции определяется последующими параметрами <Corr_Index> и <Corr_Mode>. Если <Corr_Status> = 41, содержание вектора перезаписывается (см. выше).		
	Тип данных:	REAL	
<Corr_Index>:	Сегмент, элемент коррекции которого необходимо модифицировать / Индекс оси ориентации, вектор направления которой необходимо модифицировать		
	Тип данных:	INT	
	Параметр <Corr_Index> записан в десятичном коде (с 1-й позиции по 10-ю позицию).		
	1-я позиция:	Содержит индекс сегмента или оси ориентации в подцепочке.	
	10-я позиция:	Ссылается на подцепочку.	
		0x	Цепь детали
		1x	Цепь инструмента

<Corr_Mode>:	Режим коррекции		
	Тип данных:	INT	
	Параметр <Corr_Mode> записан в десятичном коде (с 1-й позиции по 1000-ю позицию).		
	1-я позиция:	Определяет, который элемент необходимо корректировать.	
		xxx**0**	Коррекция линейного вектора смещения
		xxx**1**	Коррекция вектора направления оси ориентации
	10-я позиция:	Определяет, каким образом необходимо модифицировать элемент коррекции, на который ссылается содержание <Corr_Index>.	
		xx**0**x	Вектор коррекции записывается непосредственно в элемент коррекции. Этот вариант можно использовать для непосредственного описания элемента коррекции, причем не требуется знать индекс <n> соответствующих системных данных ($NK_OFF_DIR[<n>, ...]).
		xx**1**x	Как 0, но с тем отличием, что переданное значение коррекции интерпретируется в мировой системе координат. Разница между вариантами 0 и 1 всегда возникает в случае, если кинематическая цепь содержит в исходном положении (позиции всех осей ориентации равны 0) дополнительные вращения.
		xx**2**x	Как 1, но с тем отличием, что значение коррекции относится ко всему сегменту, т. е. в элемент коррекции вносится такое значение, что весь сегмент достигает длины, определенной посредством значения коррекции.
		Указание: Для описания вектора направления оси ориентации значения 1 и 2 не разрешены.	
	100-я позиция:	Определяет, как следует интерпретировать содержание параметра <Corr_Vect>.	
		x**0**xx	Переданный вектор коррекции <Corr_Vect> содержит всю новую длину элемента коррекции или сегмента, на который ссылается <Corr_Index> в сочетании с 10-й позицией <Corr_Mode> (абсолютная коррекция).
		x**1**xx	Переданный вектор коррекции <Corr_Vect> содержит только разницу по отношению к длине элемента коррекции или сегмента, на который ссылается <Corr_Index> в сочетании с 10-й позицией <Corr_Mode> (инкрементальная коррекция).
		Указание: При коррекции вектора направления оси ориентации содержание 100-й позиции должно быть 0.	
	1000-я позиция:	Определяет, необходимо ли ограничивать коррекцию посредством следующего максимального значения: • SD41610 $SN_CORR_TRAFO_LIN_MAX или • SD41611 $SN_CORR_TRAFO_DIR_MAX	
		0xxx	Контроль максимальной коррекции активен.

		1xxx	Контроль максимальной коррекции не активен.
<No_Alarm>:	Поведение в случае аварийного сообщения (возвращаемое значение > 0) (**опционально**)		
	Тип данных:	BOOL	
	Значение:	FALSE (по умолчанию)	В случае ошибки выполнение программы приостанавливается, и отображается аварийное сообщение 14103.
		TRUE	В случае ошибки выполнение программы не останавливается и аварийное сообщение не отображается. Пример использования: Специализированная реакция согласно возвращаемому значению

Примечание

Если при вызове функции появляется ошибка, то либо выводится аварийное сообщение, либо возвращается номер ошибки (см. Параметр <No_Alarm>), так что пользователь может самостоятельно надлежащим образом отреагировать на неполадку. Причина ошибки подробнее обозначена в параметре аварийного сообщения. Возвращаемый вместо аварийного сообщения номер ошибки идентичен параметру аварийного сообщения.

Дополнительная информация по CORRTRAFO

Кинематическая структура станка с трансформацией ориентации описывается на основе одной или двух кинематических цепей (подцепей), исходящих из нулевой точки мировой системы координат. Одна из этих цепей — **цепь инструмента** — заканчивается в исходной точке инструмента, другая — **цепь детали** — в нулевой точке базовой кинематической системы (BKS).

Функция CORRTRAFO записывает длины плеча рычага и осевые направления в станках с трансформацией ориентации в специальные элементы коррекции. Кинематическая цепь описывается, кроме прочего, посредством элементов типа OFFSET, определяемых посредством $NK_TYPE.

CORRTRAFO работает с сегментами.

Обе подцепочки могут подразделяться не более чем на четыре сегмента:

- Сегмент 1 начинается в стартовой точке цепочки и заканчивается на первой оси ориентации.
- Сегмент 2 – сегмент между осью ориентации 1 и осью ориентации 2.
- Сегмент 3 – сегмент между осью ориентации 2 и осью ориентации 3.
- Сегмент 4 – сегмент между осью ориентации 3 и концом цепи инструмента или детали.

Сегменты могут содержать соответственно постоянные элементы цепочки типов OFFSET или ROT_CONST.

На следующем рисунке показана трансформация ориентации с 2 осями ориентации.

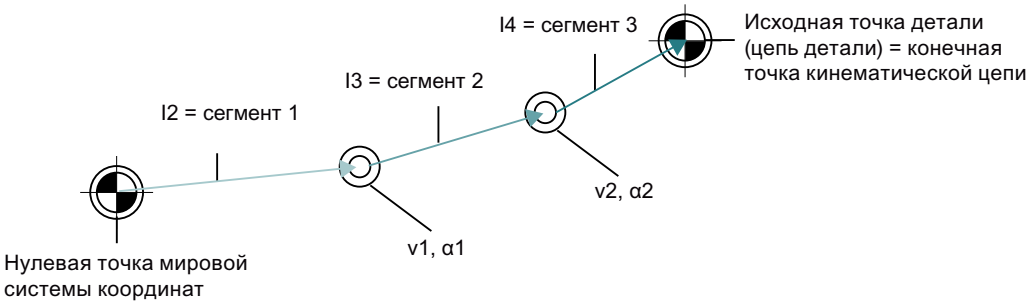


Рисунок 4-8 Пример CORRTRAFO

Сегменты имеют однозначное определение: При следовании по кинематической подцепочке от ее стартовой точки до конечной точки получается, что у первого сегмента — индекс 0, у следующего сегмента — индекс 1 и т. п. В таком случае индекс последнего сегмента всегда равен количеству осей ориентации.

Элементы коррекции

На каждый постоянный элемент кинематической цепи (элемент цепи типа $NK_TYPE[<n>] = "OFFSET") в каждом из этих сегментов можно ссылаться посредством системной переменной $NT_CORR_ELEM_T[<n>, 0 ... 3] или $NT_CORR_ELEM_P[<n>, 0 ... 3]. При помощи функции CORRTRAFO в обозначенные таким образом элементы записываются значения коррекции, установленные при замерах станка.

Пример с индексом трансформации = 1:

- $NT_CORR_ELEM_T[1,0] = "C_AXIS_OFFSET"; смещение оси C (ось ориентации 1) в сегменте 1 определено как элемент коррекции.
- $NT_CORR_ELEM_T[1,1] = "B_AXIS_OFFSET"; смещение оси B (ось ориентации 2) в сегменте 2 определено как элемент коррекции.
- $NT_CORR_ELEM_T[1,2] = "BASE_TOOL_OFFSET"; смещение оси B к исходной точке инструмента в сегменте 3 определено как элемент коррекции.

Последовательность ссылок в $NT_CORR_ELEM_T/P[<n>, 0 ... 3] должна соответствовать описанным выше сегментам, т. е., в $NT_CORR_ELEM_T/P [<n>, 0] может находиться только элемент цепи, расположенный перед первой осью ориентации, и т. п.

Функция CORRTRAFO записывает в определенные таким образом элементы коррекции значения, которые были определены посредством замеров станка. Модифицирование значений коррекции определяется в CORRTRAFO посредством параметра <Corr_Mode>.

Закрытие цепи

Если в системной переменной $NT_CNTRL[<n>] установлены бит 7 или бит 8, то в конце цепи детали (бит 7) или перед стартовой точкой цепи инструмента (бит 8) внутри автоматически добавляются дополнительные постоянные элементы цепи, создающие связь от конечной точки цепи к нулевой точке станка ("Закрытие цепи").

Извне эти автоматически добавленные элементы можно только читать, запись в них невозможна (см. системные переменные $AC_TRAFO_CORR_ELEM_P/T).

Точка закрытия цепи инструментов

Если системная переменная $NT_CLOSE_CHAIN_T не пустая, цепь инструмента закрывается не в конце цепи, а в конечной точке обозначенного элемента цепи. И тогда другие элементы цепи, находящиеся позади этой точки, приводят при активации трансформации к соответствующему смещению нулевой точки.

Индекс оси ориентации

Кроме постоянного смещения между осями ориентации, функция CORRTRAFO может описывать и векторы направления осей ориентации. Под индексом оси ориентации в данном случае понимают индекс, результирующий, если пройти кинематическую подцепочку с начала до её конца, при этом счёт начинается с нуля. Поэтому индекс оси ориентации всегда равен индексу расположенного перед ней сегмента.

Индекс оси ориентации можно определить и посредством системной переменной $AC_TRAFO_ORIAX_LOC.

Максимально допустимое изменение элемента цепи

Максимально допустимое изменение элемента цепи можно ограничить обоими установочными данными SD41610 $SN_CORR_TRAFO_LIN_MAX для векторов смещения и SD41611 $SN_CORR_TRAFO_DIR_MAX для векторов направления осей ориентации. SD41610 $SN_CORR_TRAFO_LIN_MAX задает, на какую максимальную величину разрешено изменять каждый отдельный компонент вектора по сравнению с его исходным значением. SD41611 $SN_CORR_TRAFO_DIR_MAX задает, на какой максимальный угол разрешено изменять направление вектора оси по сравнению с его исходным значением. При этом исходное значение — это значение, действующее в трансформации, активной при вызове CORRTRAFO. Это означает, что, возможно, измененное при активации трансформации содержание данных кинематики в системе УД не влияет на действие функции CORRTRAFO.

4.13 Коррекции инструмента

4.13.1 Память коррекций

Структура памяти коррекций

Любое поле данных может быть вызвано с помощью номера T и D и содержит наряду с геометрическими данными инструмента и другие элементы, например, тип инструмента.

Данные резцов пользователя

Через машинные данные могут быть сконфигурированы данные резцов пользователя. Следовать указаниям изготовителя станка.

Параметры инструмента

Примечание

Отдельные значения в памяти коррекций

Отдельные значения памяти коррекций P1 до P25 через системные переменные могут считываться и записываться из программы. Все прочие параметры зарезервированы.

Параметры инструмента $TC_DP6 до $TC_DP8, $TC_DP10 и $TC_DP11, а также $TC_DP15 до $TC_DP17, $TC_DP19 и $TC_DP20, в зависимости от типа инструмента, имеют иное значение.

Номер параметра инструмента (DP)	Значение системных переменных	Примечание
$TC_DP1	Тип инструмента	Обзор см. Список
$TC_DP2	Положение резцов	Только для токарных инструментов
Геометрия	**Коррекция длин**	
$TC_DP3	Длина 1	Вычисление по
$TC_DP4	Длина 2	типу и плоскости
$TC_DP5	Длина 3	
Геометрия	**Радиус**	
$TC_DP6 [1)] $TC_DP6 [2)]	Радиус 1 / длина 1 Диаметр d	Фрезерный/токарный/шлифовальный инструмент Наградка
$TC_DP7 [1)] $TC_DP7 [2)]	Длина 2 / угловой радиус конической фрезы Ширина паза b Угловой радиус	Фрезерные инструменты Наградка
$TC_DP8 [1)] $TC_DP8 [2)]	Радиус закругления 1 для фрезерных инструментов Выступ k	Фрезерные инструменты Наградка
$TC_DP9 [1) 3)]	Радиус закругления 2	зарезервировано
$TC_DP10 [1)]	Угол 1 торцовая сторона инструмента	Конические фрезерные инструменты
$TC_DP11 [1)]	Угол 2 продольная ось инструмента	Конические фрезерные инструменты
Износ	**Коррекция длин и радиуса**	
$TC_DP12	Длина 1	
$TC_DP13	Длина 2	
$TC_DP14	Длина 3	
$TC_DP15 [1)] $TC_DP15 [2)]	Радиус 1 / длина 1 Диаметр d	Фрезерный/токарный/шлифовальный инструмент Наградка
$TC_DP16 [1)] $TC_DP16 [3)]	Длина 2 / угловой радиус конической фрезы Ширина паза b Угловой радиус	Фрезерные инструменты Наградка
$TC_DP17 [1)] $TC_DP17 [2)]	Радиус закругления 1 для фрезерных инструментов Выступ k	Фрезерование / 3D торцовое фрезерование Наградка
$TC_DP18 [1) 3)]	Радиус закругления 2	зарезервировано

Номер параметра инструмента (DP)	Значение системных переменных	Примечание
$TC_DP19[1)]	Угол 1 торцовая сторона инструмента	Конические фрезерные инструменты
$TC_DP20[1)]	Угол 2 продольная ось инструмента	Конические фрезерные инструменты
Базовый размер/адаптер	**Коррекции длин**	
$TC_DP21	Длина 1	
$TC_DP22	Длина 2	
$TC_DP23	Длина 3	
Технология		
$TC_DP24	Задний угол	Только для токарных инструментов
$TC_DP25		зарезервировано

1) Действительно и для фрезерных инструментов для 3D-торцового фрезерования

2) Для типа инструмента "наградка"

3) зарезервировано

Примечания

Для геометрических величин (например, длина 1 или радиус) существует несколько компонентов записи. Они аддитивно вычисляются в результирующую величину (например, общая длина 1, общий радиус), которая после активируется.

Не нужным коррекциям присваивается значение ноль.

Параметры инструмента $TC-DP1 до $TC-DP23 с контурными инструментами

Примечание

Параметры инструментов, не включенные в таблицу, например, $TC_DP7, не обрабатываются, т. е. их содержание не имеет значения.

Номер параметра инструмента (DP)	Объяснение	Резцы Dn		Примечание
$TC_DP1	Тип инструмента			400 до 599
$TC_DP2	Положение резцов			
Геометрия	**Коррекция длин**			
$TC_DP3	Длина 1			
$TC_DP4	Длина 2			
$TC_DP5	Длина 3			
Геометрия	**Радиус**			
$TC_DP6	Радиус			
Геометрия	**Предельный угол**			
$TC_DP10	Мин. предельный угол			
$TC_DP11	Макс. предельный угол			

Номер параметра инструмента (DP)	Объяснение	Резцы Dn		Примечание
Износ	**Коррекция длин и радиуса**			
$TC_DP12	Износ длина 1			
$TC_DP13	Износ длина 2			
$TC_DP14	Износ длина 3			
$TC_DP15	Износ радиус			
Износ	**Предельный угол**			
$TC_DP19	Износ мин. предельный угол			
$TC_DP20	Износ макс. предельный угол			
Базовый размер/адаптер	**Коррекции длин**			
$TC_DP21	Длина 1			
$TC_DP22	Длина 2			
$TC_DP23	Длина 3			

Основное значение и значение износа

Результирующие величины получаются как сумма основного значения и значения износа соответственно (например, $TC_DP6 + $TC_DP15 для радиуса). К длине инструмента первого резца кроме этого прибавляется и базовый размер ($TC_DP21 – $TC_DP23). Дополнительно на эту длину инструмента действуют все другие величины, которые и у обычного инструмента могут влиять на эффективную длину инструмента (адаптер, ориентируемый инструментальный суппорт, установочные данные).

Предельный угол 1 и 2

Предельные углы 1 или 2 относятся к вектору от центра резца к исходной точке резца соответственно и подсчитываются против часовой стрелки.

4.13.2 Аддитивные коррекции

4.13.2.1 Выбор аддитивных коррекций (DL)

Аддитивные коррекции могут рассматриваться как программируемые при обработке коррекции процесса. Они относятся к геометрическим данным резца, являясь тем самым составной частью данных резцов инструмента.

Обращение к данным аддитивной коррекции осуществляется через номер DL (DL: Location dependent; коррекции относительно соответствующего места использования) и они вводятся через интерфейс.

Использование

Через аддитивные коррекции возможно исправление обусловленных местом использования погрешностей размера.

Синтаксис

```
DL=<номер>
```

Значение

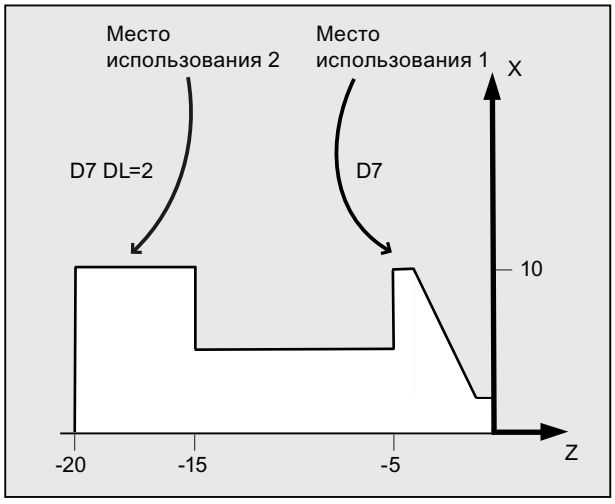
DL:	Команда для активации аддитивной коррекции
<номер>:	Через параметр <номер> указывается активируемый аддитивный блок данных коррекции инструмента

Примечание

Определение числа и активация аддитивных коррекций осуществляется через машинные данные (→ Следовать указаниям изготовителя станка!).

Пример

Один и тот же резец используется для 2 гнезд подшипника:



Программный код	Комментарий
N110 T7 D7	; Револьвер позиционируется на место 7. D7 и DL=1 активируются и выводятся в следующем кадре.
N120 G0 X10 Z1	
N130 G1 Z-6	
N140 G0 DL=2 Z-14	; Аддитивно к D7 активируется DL=2 и выводится в следующем кадре.
N150 G1 Z-21	
N160 G0 X200 Z200	; Подвод к точке смены инструмента.
...	

4.13.2.2 Определение значений износа и установочных значений ($TC_SCPxy[t,d], $TC_ECPxy[t,d])

4.13.2.3 Удаление аддитивных коррекций (DELDL)

С помощью команды DELDL аддитивные коррекции для резца инструмента удаляются (освобождение памяти). При этом удаляются как определенные значения износа, так и установочные значения.

Синтаксис

```
DELDL[<t>,<d>]
DELDL[<t>]
DELDL
<состояние>=DELDL[<t>,<d>]
```

Значение

DELDL:	Команда для удаления аддитивных коррекций	
<t>:	номер T инструмента	
<d>:	номер D резца инструмента	
DELDL[<t>,<d>]:	Удаляются все аддитивные коррекции резца <d> инструмента <t>.	
DELDL[<t>]:	Удаляются все аддитивные коррекции всех резцов инструмента <t>.	
DELDL:	Удаляются все аддитивные коррекции всех резцов всех инструментов блока TO (для канала, в котором программируется команда).	
<состояние>:	Состояние удаления	
	Вели-чина:	Значение:
	0	Удаление было выполнено успешно.
	-	Удаление не было выполнено (если параметрирование обозначает точно один резец), или удаление было выполнено не полностью (если параметрирование обозначает несколько резцов).

Примечание

Значения износа и установочные значения активных инструментов не могут быть удалены (поведение аналогично удалению D или данных инструмента).

4.13.3 Коррекция инструмента - специальная обработка

С помощью установочных данных SD42900 bis SD42960 можно управлять нормированием знаков для длины инструмента и износа.

Это же относится к поведению компонентов износа при отражении гео-осей или при смене плоскости обработки, а также к температурной компенсации в направлении инструмента.

Значения износа

Если ниже имеются ссылки на значения износа, то под ними понимается сумма самих значений износа ($TC_DP12 до $TC_DP20) и суммарных коррекций со значениями износа ($SCPX3 до $SCPX11) и установочными значениями ($ECPX3 до $ECPX11).

Информация по суммарным коррекциям: см. Описание функций – Управление инструментом

Установочные данные

SD42900 $SC_MIRROR_TOOL_LENGTH	Отражение компонентов длин инструмента и компонентов базового размера.
SD42910 $SC_MIRROR_TOOL_WEAR	Отражение значений износа компонентов длин инструмента.
SD42920 $SC_WEAR_SIGN_CUTPOS	Нормирование знака компонентов износа в зависимости от положения резца.
SD42930 $SC_WEAR_SIGN	Инвертирует знаки размеров износа.
SD42935 $SC_WEAR_TRANSFORM	Преобразование значений износа.
SD42940 $SC_TOOL_LENGTH_CONST	Согласование компонентов длин инструмента с гео-осями.
SD42950 $SC_TOOL_LENGTH_TYPE	Согласование компонентов длин инструмента независимо от типа инструмента.
SD42960 $SC_TOOL_TEMP_COMP	Значение температурной компенсации в направлении инструмента. Действует и при наличии ориентации инструмента.

Дополнительная информация

Активация измененных установочных данных

Новое нормирование компонентов инструмента при изменении описываемых установочных данных активируется только при следующем выборе резца инструмента. Если инструмент уже активен и необходимо активировать измененное нормирование данных этого инструмента, то требуется заново выбрать этот инструмент.

Это же относится и к случаю, когда изменяется результирующая длина инструмента, т.к. было изменено состояние отражения оси. Для активации измененных компонентов длин инструмента необходимо заново выбрать инструмент после команды отражения.

Ориентируемые инструментальные суппорты и новые установочные данные

Установочные данные SD42900 до SD42940 не действуют на компоненты возможно активного ориентируемого инструментального суппорта. Однако инструмент со всей его полученной длиной (длина инструмента + износ + базовый размер) всегда включается в вычисление с ориентируемым инструментальным суппортом. При расчете результирующей общей длины учитываются все изменения, вызванные установочными

данными; т. е. векторы ориентируемого инструментального суппорта не зависят от плоскости обработки.

Примечание

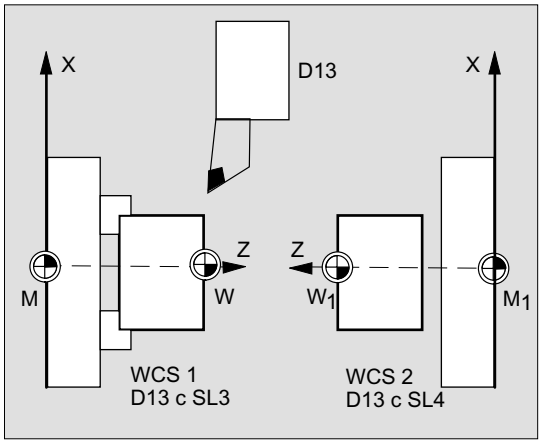
Часто при использовании ориентируемого инструментального суппорта имеет смысл определить все инструменты для не отраженной основной системы, включая те, которые используются только при зеркальной обработке. Тогда при обработке с отраженными осями инструментальный суппорт поворачивается таким образом, что фактическое положение инструмента описывается правильно. В этом случае все компоненты длин инструмента автоматически действуют в правильном направлении, поэтому управление нормированием отдельных компонентов через установочные данные в зависимости от состояния отражения отдельных осей становится ненужным.

Другие возможности использования

Использование функциональности ориентируемого инструментального суппорта имеет смысл и тогда, когда на станке не предусмотрено физической возможности вращения инструментов, но жестко установлены инструменты с различной ориентацией. В этом случае измерение инструментов может осуществляться унифицировано в первичной ориентации, а релевантные для обработки размеры получаются через вращения виртуального инструментального суппорта.

4.13.3.1 Отражение длин инструмента

При установочных данных SD42900 $SC_MIRROR_TOOL_LENGTH и SD42910 $SC_MIRROR_TOOL_WEAR, отличных от нуля, возможно отражение компонентов длин инструмента и компонентов базовых размеров со значениями износа их соответствующих осей.



SD42900 $SC_MIRROR_TOOL_LENGTH

Установочные данные **отличны** от нуля:

Компоненты длин инструмента ($TC_DP3, $TC_DP4 и $TC_DP5) и компоненты базовых размеров ($TC_DP21, $TC_DP22 и $TC_DP23), соответствующие оси которых отражены, также отражаются - через инверсию знака.

Значения износа при этом **не** отражаются. Если они также должны быть отражены, то должны быть установлены установочные данные SD42910 $SC_MIRROR_TOOL_WEAR.

SD42910 $SC_MIRROR_TOOL_WEAR

Установочные данные **отличны** от нуля:

Значения износа компонентов длин инструмента, соответствующие оси которых отражены, также отражаются - через инверсию знака.

4.13.3.2 Нормирование знака износа

При установочных данных SD42920 $SC_WEAR_SIGN_CUTPOS и SD42930 $SC_WEAR_SIGN, отличных от нуля, возможна инверсия нормирования знака компонентов износа.

SD42920 $SC_WEAR_SIGN_CUTPOS

Установочные данные **отличны** от нуля:

У инструментов с релевантным положением резцов (токарные и шлифовальные инструменты, типы инструмента 400) нормирование знака компонентов износа в плоскости обработки зависит от положения резцов. Для типов инструментов без релевантного положения резцов эти установочные данные не имеют значения.

В таблице ниже через X обозначены размеры, знак которых инвертируется через SD42920 (отличны от 0):

Положение резца	**длина 1**	**длина 2**
1		
2		X
3	X	X
4	X	
5		
6		
7		X
8	X	
9		

Примечание

Нормирования знака через SD42920 и SD42910 не зависят друг от друга. Если, к примеру, знак указания размера изменяется через оба установочных данных, то результирующий знак остается без изменений.

SD42930 $SC_WEAR_SIGN

Установочные данные **отличны** от нуля:

Знак всех размеров износа инвертируется. Это действует как на длину инструмента, так и на прочие величины, как то радиус инструмента, радиус закругления и т.д.

Если вводится положительный размер износа, то тем самым инструмент становится "короче" и "тоньше", см. главу "Коррекция инструмента, специальная обработка", активация измененных установочных данных".

4.13.3.3 Система координат активной обработки (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)

В зависимости от кинематики станка или от наличия ориентируемого инструментального суппорта измеренные в одной из этих систем координат значения износа переводятся или преобразуются в подходящую систему координат.

Системы координат активной обработки

Из следующих систем координат могут происходить смещения длины инструмента, которые включают компонент длины инструмента "износ" через соответствующую G-команду группы 56 в вычисление активного инструмента.

- Система координат станка (MCS)
- Базовая кинематическая система (BKS)
- Система координат детали (WCS)
- Система координат инструмента (TCS)
- Система координат инструмента кинематической трансформации (KCS)

Синтаксис

```
TOWSTD
TOWMCS
TOWWCS
TOWBCS
TOWTCS
TOWKCS
```

Объяснение

TOWSTD:	Значение по умолчанию для коррекций в длине инструмента, значение износа
TOWMCS:	Коррекции в длине инструмента в MCS
TOWWCS:	Коррекции в длине инструмента в WCS
TOWBCS:	Коррекции в длине инструмента в BKS
TOWTCS:	Коррекции длины инструмента на исходной точке инструментального суппорта (ориентируемый инструментальный суппорт)
TOWKCS:	Коррекции длины инструмента инструментальной головки (кинематической трансформации)

Дополнительная информация

Отличительные особенности

В следующей таблице представлены важнейшие отличительные особенности:

G-команда	Значение износа	Активный ориентируемый инструментальный суппорт
TOWSTD	Установка по умолчанию, длина инструмента	Значения износа подвержены вращению.
TOWMCS	Значение износа в MCS. TOWMCS идентична TOWSTD, если нет активного ориентируемого инструментального суппорта.	Вращается только вектор результирующей длины инструмента без учета износа.
TOWWCS	Значение износа пересчитывается из WCS в MCS.	Вектор инструмента вычисляется без учета износа, как при TOWMCS.
TOWBCS	Значение износа пересчитывается из BKS в MCS.	Вектор инструмента вычисляется без учета износа, как при TOWMCS.
TOWTCS	Значение износа пересчитывается из системы координат инструмента в MCS.	Вектор инструмента вычисляется без учета износа, как при TOWMCS.

TOWWCS, TOWBCS, TOWTCS: вектор износа прибавляется к вектору инструмента.

Линейная трансформация

Длину инструмента можно рационально определить в MCS, только если MCS следует из BKS через линейную трансформацию.

Не линейная трансформация

Если, например, с TRANSMIT активна не линейная трансформация, то при указании MCS в качестве желаемой системы координат автоматически используется BKS.

Кинематическая трансформация отсутствует, ориентируемый инструментальный суппорт отсутствует

При отсутствии активной кинематической трансформации и ориентируемого инструментального суппорта все системы координат, кроме WCS, совпадают. Тем самым только WCS отличается от других систем координат. Поскольку нормируются только длины инструмента, смещения между системами координат не имеют значения.

Учет значений износа

Установочные данные **SD42935 $SC_WEAR_TRANSFORM** определяют, какой из трех компонентов износа:

- Износ
- Суммарные коррекции точные
- Суммарные коррекции грубые

необходимо подвергнуть вращению через трансформацию адаптера или ориентируемый инструментальный суппорт, если активна одна из следующих G-команд:

- TOWSTD
  Установка по умолчанию. Для коррекций в длине инструмента.
- TOWMCS
  Значения износа в системе координат станка (MCS).
- TOWWCS
  Значения износа в системе координат детали (WCS).
- TOWBCS
  Значения износа в базовой кинематической системе (BKS).
- TOWTCS
  Значения износа в системе координат инструмента на зажиме инструмента (T исходная точка инструментального суппорта).
- TOWKCS
  Значения износа в системе координат инструментальной головки при кинематической трансформации.

Примечание

Нормирование отдельных компонентов износа (согласование с гео-осями, нормирование знака) управляется следующими факторами:

- Активная плоскость
- Трансформация адаптера
- Установочные данные:
  - SD42910 $SC_MIRROR_TOOL_WEAR
  - SD42920 $SC_WEAR_SIGN_CUTPOS
  - SD42930 $SC_WEAR_SIGN
  - SD42940 $SC_TOOL_LENGTH_CONST
  - SD42950 $SC_TOOL_LENGTH_TYPE

4.13.3.4 Длина инструмента и смена плоскостей

При установочных данных SD42940 $SC_TOOL_LENGTH_CONST, отличных от нуля, можно согласовать такие компоненты длин инструмента, как длина, износ и базовый размер, с геометрическими осями для токарных и шлифовальных инструментов при смене плоскости.

SD42940 $SC_TOOL_LENGTH_CONST

Установочные данные **отличны** от нуля:

Согласование компонентов длин инструмента (длина, износ и базовый размер) с гео-осями при смене плоскости обработки (`G17 - G19`) не изменяется.

Таблица ниже показывает согласование компонентов длин инструмента с гео-осями для токарных и шлифовальных инструментов (тип инструмента 400 до 599):

Содержание	**длина 1**	**длина 2**	**длина 3**
17	Y	X	Z
*)	X	Z	Y
19	Z	Y	X
-17	X	Y	Z
-18	Z	X	Y
-19	Y	Z	X

*) Любое значение, отличное от 0, которое не равно одному из шести приведенных значений, нормируется как значение 18.

Таблица ниже показывает согласование компонентов длин инструмента с гео-осями для всех других инструментов (тип инструмента < 400 или > 599):

Плоскость обработки	**длина 1**	**длина 2**	**длина 3**
*)	Z	Y	X
18	Y	X	Z
19	X	Z	Y
-17	Z	X	Y
-18	Y	Z	X
-19	X	Y	Z

*) Любое значение, отличное от 0, которое не равно одному из шести приведенных значений, нормируется как значение 17.

Примечание

При представлении в таблицах предполагается, что гео-оси до 3 обозначаются как X, Y, Z. Для согласования коррекции с осью определяющим является не идентификатор оси, а последовательность осей.

4.13.4 Коррекция инструмента Online

4.13.4.1 Определение функции полинома (FCTDEF)

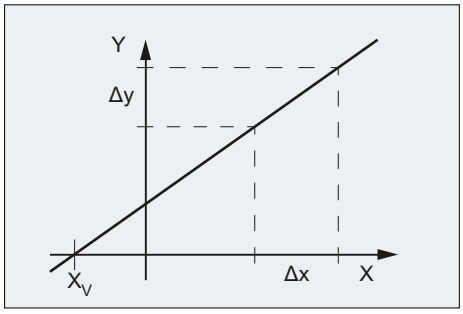
Определенные стратегии правки (к примеру, шарошка) характеризуются тем, что радиус шлифовального круга непрерывно (линейно) уменьшается с подачей шарошки. Для этого необходима линейная функция между подачей шарошки и записью значения износа соответствующей длины. Линейная функция определяется через предопределенную процедуру FCTDEF(...) для функций полинома макс. до третьей степени.

Уравнение прямой

$y = f(x) = a_0 + a_1 * x_1$

a_1: Подъем прямой, где $a_1 = \Delta x / \Delta y$

a_0: Смещение прямой вдоль оси X, где $a_0 = -a1 * X_V$



Синтаксис

```
FCTDEF(<Func>,<LLimit>,<ULimit>,<a0>,<a1>,<a2>,<a3>)
```

Значение

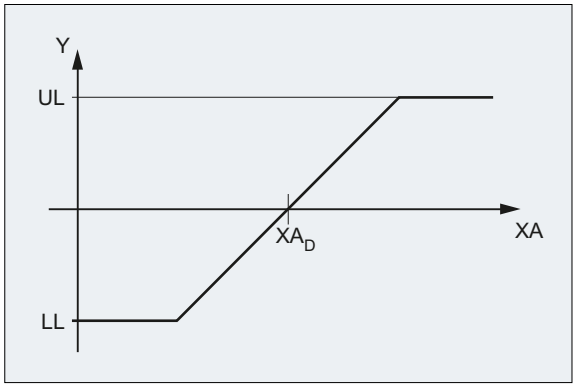
`FCTDEF(...)`:	Определение функции полинома для PUTFTOCF(...): $y = f(x) = a_0 + a_1*x + a_2*x^2 + a_3*x^3$	
`<Func>`:	Номер функции	
	Тип данных:	INT
	Диапазон значений:	1, 2, 3
`<LLimit>`:	Нижняя величина ограничения	
	Тип данных:	REAL
`<ULimit>`:	Верхняя величина ограничения	
	Тип данных:	REAL
`<a0>,<a1>,<a2>,<a3>`:	Коэффициенты полиномиальной функции	
	Тип данных:	REAL

Пример

Определения

- Номер функции: 1
- Нижняя и верхняя величина ограничения: -100, 100
- Подъем характеристики: $a_1 = 1$
- Рабочая точка должна находиться в середине характеристики. Для этого характеристика должна быть смещена в отрицательном Y-направлении в соответствии с номинальной позицией оси XA в СКД к моменту определения функции в программе ЧПУ: $a_0 = -a_1 * XA_D = -1 * \$AA_IW$
- $a_2 = a_3 = 0$

Характеристика



UL Верхняя величина ограничения

LL Нижняя величина ограничения

XA_D Заданное значение оси XA к моменту определения функции в программе ЧПУ

Программирование

Программный код	Комментарий
`FCTDEF(1,-100,100,-$AA_IW[XA],1)`	; определение функций

4.13.4.2 Онлайн-запись коррекции инструмента, непрерывная (PUTFTOCF)

Предварительно определенная процедура PUTFTOCF(...) осуществляет онлайн-коррекцию инструмента при помощи предварительно определенной через FCTDEF(...) (Страница 774) функции полинома.

Примечание

Онлайн-коррекция инструмента может быть также осуществлена через синхронное действие.

Дополнительную информацию см. Справочник по функциям, синхронное действие.

Синтаксис

```
PUTFTOCF(<Func>,<RefVal>,<ToolPar>,<Chan>,<Sp>)
```

Значение

`PUTFTOCF(...)`:	Запись онлайн-коррекции инструмента, непрерывно покадрово на основании функции полинома, определенной через FCTDEF(...)	
`<Func>`:	Номер функции, заданный при определении функции через FCTDEF(...)	
	Тип данных:	INT
	Диапазон значений:	1, 2, 3

<RefVal>:	Исходное значение, из которого должна выводиться коррекция (например, . заданное значение оси)	
	Тип данных:	VAR REAL
<ToolPar>:	Номер параметра износа (длина 1, 2 или 3), в котором должна быть учтена поправка.	
	Тип данных:	INT
<Chan>:	Номер канала, в котором должна действовать онлайн-коррекция инструмента **Примечание**: Требуется только в том случае, если коррекция должна действовать не в активном канале.	
	Тип данных:	INT
<Sp>:	Номер шпинделя, для которого должна действовать онлайн-коррекция инструмента **Примечание:** Требуется только в том случае, если вместо активного используемого инструмента должен быть исправлен неактивный шлифовальный круг.	
	Тип данных:	INT

4.13.4.3 Дискретная запись онлайн-коррекции инструмента (PUTFTOC)

Функция

С помощью предварительно определенной процедуры PUTFTOC(...) онлайн-коррекция инструмента осуществляется с фиксированной поправкой.

Синтаксис

```
PUTFTOC(<CorrVal>,<ToolPar>,<Chan>,<Sp>)
```

Значение

PUTFTOC(...):	Запись онлайн-коррекции инструмента	
<CorrVal>:	Поправка, добавляемая к параметру износа	
	Тип данных:	REAL
<ToolPar>:	Номер параметра износа (длина 1, 2 или 3), в котором должна быть учтена поправка.	
	Тип данных:	INT
<Chan>:	Номер канала, в котором должна действовать онлайн-коррекция инструмента **Примечание:** Требуется только в том случае, если коррекция должна действовать не в активном канале.	
	Тип данных:	INT

<Sp>:	Номер шпинделя, для которого должна действовать онлайн-коррекция инструмента **Примечание:** Требуется только в том случае, если вместо активного используемого инструмента должен быть исправлен неактивный шлифовальный круг.	
	Тип данных:	INT

4.13.4.4 Включение/выключение онлайн-коррекции инструмента (FTOCON/FTOCOF)

Команды G FTOCON и FTOCOF включают / выключают онлайн-коррекцию инструмента.

Синтаксис

```
FTOCON
...
FTOCOF
```

Значение

FTOCON:	Включение онлайн-коррекции инструмента Команда должна быть запрограммирована в канале, в котором должна быть включена онлайн-коррекция инструмента.
FTOCOF:	Выключение онлайн-коррекции инструмента Команда должна быть запрограммирована в канале, в котором должна быть выключена онлайн-коррекция инструмента. **Примечание:** FTOCOF более не выводит коррекцию инструмента. В специфичных для резца данных коррекции сохраняется значение, рассчитанное в PUTFTOC/PUTFTOCF. Для окончательной деактивации онлайн-коррекции инструмента, после FTOCOF необходимо выполнить и включение/выключение инструмента (T...).

4.13.5 Коррекция радиуса инструмента 3D

4.13.5.1 Коррекция на радиус инструмента 3D для периферийного фрезерования 3D (CUT3DC, CUT3DCD, ISD)

Коррекция на радиус инструмента 3D (3D-КРИ) для периферийного фрезерования 3D без учёта ограничивающих поверхностей выбирается посредством действующей модально G-команды CUT3DC или CUT3DCD.

Сама активация выполняется посредством G41 или G42. Коррекция на радиус инструмента выключается посредством G40.

Синтаксис

```
G41/G42 ORIC/ORID ISD=... CUT3DC/CUT3DCD CDOF2 X... Y... Z...
...
G40 X... Y... Z...
```

Объяснение

CUT3DC:	3D-КРИ для периферийного фрезерования (только при активной 5-осевой трансформации)	
CUT3DCD:	3D-КРИ относительно дифференциального инструмента для периферийного фрезерования (только при активной 5-осевой трансформации) Разница радиусов устанавливается посредством параметра инструмента $TC_DP15.	
G41/G42 X... Y... Z...:	Включение коррекции на радиус инструмента	
	G41:	Коррекция на радиус инструмента слева от контура
	G42:	Коррекция на радиус инструмента справа от контура
	Указание: Включение должно происходить в линейном кадре (G0/G1).	
CDOF2:	Выключение контроля столкновений для периферийного фрезерования 3D	
ORIC/ORID:	G-команды ORIC и ORID устанавливают поведение при изменениях ориентации на наружных углах.	
	ORIC:	Изменения ориентации на наружных углах накладываются на вставляемый круговой кадр.
	ORID:	Изменения ориентации на наружных углах выполняются перед вставляемым круговым кадром.
ISD=<Value>:	Адрес ISD позволяет менять глубину врезания инструмента при периферийном фрезеровании и **активной** коррекции на радиус инструмента 3D.	
	<Value>:	Длина глубины врезания
G40 X... Y... Z...:	Выключение коррекции на радиус инструмента **Указание:** Выключение должно происходить в линейном кадре (G0/G1) с движениями гео-оси.	

Примечание

G-команды для выбора 3D-КРИ обрабатываются в кадре подвода, т. е. обычно в кадре, содержащем G41 или G42.

G41 или G42 могут быть запрограммированы и в кадрах без перемещения в значимых для коррекции гео-осях. В этом случае кадром подвода является первый кадр перемещения, следующий за таким кадром.

Смена варианта 3D-КРИ при активной коррекции на радиус инструмента игнорируется без аварийного сообщения.

Пример

Программный код	Комментарий
	; определение инструмента D1:
$TC_DP1[1,1]=120	; тип (концевая фреза)
$TC_DP3[1,1]=20	; вектор коррекции на длину
$TC_DP6[1,1]=8	; радиус
N10 X0 Y0 Z0 T1 D1 F12000	; выбор инструмента.
N20 TRAORI(1)	; включение трансформации.
N30 G42 ORIC ISD=10 **CUT3DC** G64 X30	; активация периферийного фрезерования 3D,
	; изменения ориентации на наружных углах (непрерывно),
	; глубина врезания: 10 мм
N40 ORIWKS A30 B15	; изменение ориентации на углу посредством указания позиций осей.
N50 Y20 A3=1 C3=1	; кадр перемещения с изменением ориентации,
	; указание ориентации с вектором направления.
N60 X50 Y30	; кадр перемещения с постоянной ориентацией.
N70 Y50 A3=0.5 B3=1 C3=5	; кадр перемещения с изменением ориентации.
N80 M63	; кадр без информации перемещения.
N90 X0 ISD=20	; кадр перемещения с изменением глубины врезания.
N100 G40 Y0	; деактивация коррекции на радиус инструмента.
N110 M30	

Дополнительная информация

Траектория и ориентация

Используемый здесь вариант периферийного фрезерования реализован посредством задания траектории (направляющей линии) и соответствующей ориентации. При таком типе обработки форма инструмента на траектории не имеет значения. Значение имеет только радиус на полюсе зацепления инструмента.

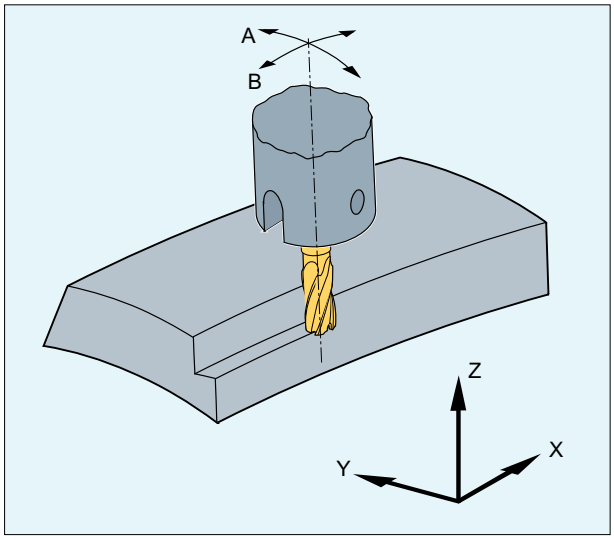


Рисунок 4-9 Периферийное фрезерование

Режим подвода

Режимом подвода в случае 3D-вариантов коррекции на радиус инструмента всегда является NORM.

Поведение на наружных углах

На наружных углах при периферийном фрезеровании с 3D-КРИ по аналогии с условиями при 2½D-КРИ обрабатываются G-команды группы 18 (поведение на углах коррекции на инструмент):

- G450: Переходная окружность (инструмент обходит углы детали по круговой траектории)
  В отличие от решения при 2½D-КРИ добавленный элемент контура на наружном угле – это всегда окружность с радиусом 0, на которую коррекция на радиус инструмента воздействует так же, как и на любую другую запрограммированную траекторию. Вставка конических сечений вместо окружностей невозможна. Поэтому в этом случае адрес DISC не имеет значения и не обрабатывается.
- G451: Точка пересечения эквидистант (инструмент выполняет свободное резание в углу детали)
  Для определения точки пересечения кривые смещения обоих задействованных кадров удлиняются, и определяется их точка пересечения в плоскости перпендикулярно к ориентации инструмента на углу.

Перемещение точки пересечения (G451) не используется, если между задействованными кадрами перемещения вставлен, по меньшей мере, один кадр, содержащий изменение ориентации инструмента. В таком случае на углу всегда вставляется окружность.

Поведение при изменениях ориентации на наружных углах

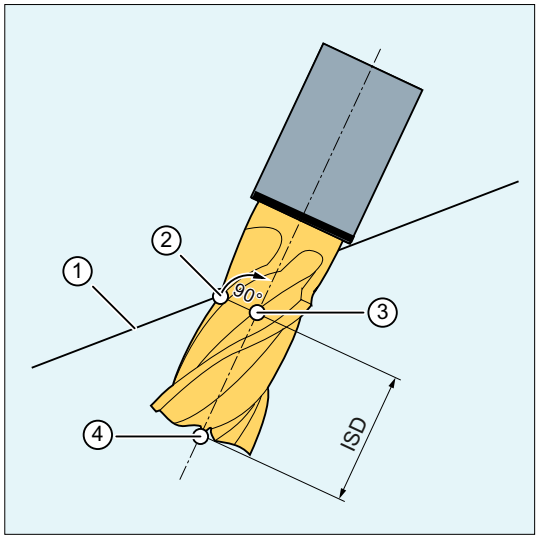
С помощью G-команд ORIC и ORID можно установить, будут ли изменения ориентации, запрограммированные между обоими образующими угол кадрами перемещения, выполняться перед началом вставленного кругового кадра (ORID) или одновременно с ним (ORIC).

Глубина врезания

Глубина врезания фрезы — это отступ вспомогательной точки фрезы от острия инструмента.

Вспомогательная точка фрезы — это вертикальная проекция точки обработки фрезы на запрограммированной траектории на продольную ось инструмента.

Так глубина врезания позволяет настраивать расположение точки обработки на боковой поверхности инструмента.



① Запрограммированная траектория
② Точка обработки фрезы
③ Вспомогательная точка фрезы
④ Острие фрезы
ISD Глубина врезания (InSertion Depth)

Рисунок 4-10 Глубина врезания

Коррекция радиуса инструмента относительно дифференциального инструмента

3D-КРИ относительно дифференциального инструмента для периферийного фрезерования активируется командой CUT3DCD. Ее следует применять, если контур запрограммирован относительно траектории центра стандартного инструмента, а обработка осуществляется другим инструментом. В этом случае при расчете коррекции на радиус инструмента 3D учитывают только значение износа радиуса активного инструмента ($TC_DP15) и, при наличии, запрограммированные смещения коррекции на инструмент OFFN и TOFFR/TOFFLR. Базовый радиус ($TC_DP6) активного инструмента **не** учитывается.

Фрезерование кармана с наклонными боковыми стенками для периферийного фрезерования посредством CUT3DC

При этой 3D-коррекции на радиус инструмента погрешность радиуса фрезы компенсируется посредством подачи в направлении нормалей к обрабатываемой поверхности. При этом плоскость, в которой лежит торцевая сторона фрезы, остается неизменной, если глубина врезания ISD не изменилась. Фреза с, например, меньшим радиусом по сравнению со стандартным инструментом, не достигла бы основания

кармана, являющегося и ограничивающей поверхностью. Для автоматической подачи инструмента СЧПУ эта ограничивающая поверхность должна быть известна: см. главу "Периферийное фрезерование 3D с учетом ограничивающей поверхности (CUT3DCC, CUT3DCCD) (Страница 789)".

Advanced Surface / Top Surface

Примечание

При использовании коррекции на радиус инструмента CUT3DCD в сочетании с лицензируемой функцией "Advanced Surface" или "Top Surface" необходимо соблюдать рекомендации по настройке "Advanced Surface" / "Top Surface"!

Для проверки настроенных данных на портале SIOS есть специальная программа проверки:

→ Программа проверки для Advanced Surface / Top Surface (https://support.industry.siemens.com/cs/ww/en/view/109738423)

4.13.5.2 Коррекция на радиус инструмента 3D для торцового фрезерования 3D (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD)

3D-коррекция на радиус инструмента (3D-КРИ) для торцового фрезерования 3D выбирается посредством G-команды (действует модально) CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD.

Сама активация выполняется посредством G41 или G42.

Для вычисления коррекции на радиус инструмента при торцовом фрезеровании 3D необходимо определение нормали к обрабатываемой плоскости. Она должна выполняться в кадре посредством G41 или G42 через адреса A4, B4, C4 и A5, B5, C5.

Коррекция на радиус инструмента выключается посредством G40.

Синтаксис

```
G41/G42 ORIC/ORID CUT3DF/CUT3DFS/CUT3DFF/CUT3DFD X... Y... Z... A4=... B4=...
C4=... A5=... B5=... C5=...
...
G40 X... Y... Z...
```

Объяснение

CUT3DFS:	3D-КРИ для торцового фрезерования с постоянной ориентацией. Ориентация инструмента установлена через G17 – G19 и не подвергается воздействию фреймов.
CUT3DFF:	3D-КРИ для торцового фрезерования с постоянной ориентацией. Ориентация инструмента – это направление, установленное через G17 – G19 и, при необходимости, повернутое фреймом.

CUT3DF:	3D-КРИ для торцового фрезерования с изменением ориентации (только при активной 5-осевой трансформации)	
CUT3DFD:	3D-КРИ относительно дифференциального инструмента для торцового фрезерования с изменением ориентации (только при активной 5-осевой трансформации) Разница радиусов устанавливается посредством параметра инструмента $TC_DP15. **Указание**: CUT3DFD возможна только в сочетании с "Сглаживание нормалей к плоскости при торцовом фрезеровании 3D". Её активирует вызов лицензируемой функции "Top Surface" посредством CYCLE832(...).	
G41/G42 X... Y... Z...:	Включение коррекции на инструмент Поведение при G41 и при G42 при торцовом фрезеровании 3D идентично. **Указание:** Включение должно происходить в линейном кадре (G0/G1).	
A4/5=... B4/5=... C4/5=...:	Определение нормали к обрабатываемой плоскости	
	A4=... B4=... C4=...:	Определение в начале кадра
	A5=... B5=... C5=...:	Определение в конце кадра
ORIC/ORID:	G-команды ORIC и ORID устанавливают поведение при изменениях ориентации на наружных углах.	
	ORIC:	Изменения ориентации на наружных углах накладываются на вставляемый круговой кадр.
	ORID:	Изменения ориентации выполняются перед круговым кадром.
G40 X... Y... Z...:	Выключение коррекции на радиус инструмента **Указание:** Выключение должно происходить в линейном кадре (G0/G1) с движениями гео-оси.	

Примечание

G41 или G42 могут быть запрограммированы и в кадрах без перемещения в значимых для коррекции гео-осях. В этом случае кадром подвода является первый кадр перемещения, следующий за таким кадром.

Смена варианта 3D-КРИ при активной коррекции на радиус инструмента игнорируется без аварийного сообщения.

Примеры

Пример 1: Торцовое фрезерование 3D посредством CUT3DF

Программный код	Комментарий
N10	; определение инструмента D1:

Программный код	Комментарий
N20 $TC_DP1[1,1]=121	; тип инструмента (тороидальная фреза)
N30 $TC_DP3[1,1]=20	; коррекция на длину
N40 $TC_DP6[1,1]=5	; радиус
N50 $TC_DP7[1,1]=3	; радиус закругления
N60	
N70	
N80 X0 Y0 Z0 A0 B0 C0 G17 T1 D1 F12000	; выбор инструмента.
N90 TRAORI(1)	; выбор трансформации ориентации.
N100 B4=-1 C4=1	; определение плоскости.
N110 **G41** ORID **CUT3DF** G64 X10 Y0 Z0	; включение коррекции на инструмент.
N120 X30	
N130 Y20 A4=1 C4=1	; наружный угол, переопределение плоскости.
N140 B3=1 C3=5	; изменение ориентации посредством ORID.
N150 B3=1 C3=1	; изменение ориентации посредством ORID.
N160 X-10 A5=1 C5=2 ORIC	
N170 A3=-2 C3=1	; изменение ориентации посредством ORIC
N180 A3=-1 C3=1	; изменение ориентации посредством ORIC
N190 Y-10 A4=-1 C4=3	; переопределение плоскости.
N200 X-20 Y-20 Z10	; внутренний угол с предшествующим кадром.
N210 X-30 Y10 A4=1 C4=1	; внутренний угол, переопределение плоскости.
N220 A3=1 B3=0.5 C3=1.7	; изменение ориентации с ORIC.
N230 X-20 Y30 A4=1 B4=-2 C4=3 ORID	
N240 A3 = 0.5 B3=-0.5 C3=1	; изменение ориентации.
N250 X0 Y30 C4=1	; движение по траектории, новая плоскость,
	; ориентация с относительным программированием.
N260 BSPLINE X20 Z15	; начало сплайна, относительное программирование ориентации
N270 X30 Y25 Z18	; остается активным при сплайне.
N280 X40 Y20 Z13	
N290 X45 Y0 PW=2 Z8	
N300 Y-20	
N310 G2 ORIMKS A30 B45 I-20 X25 Y-40 Z0	; винтовая линия, ориентация с программированием осей.
N320 G1 X0 A3=-0.123 B3=0.456 C3=2.789 B4=-1 C4=5 B5=-1 C5=2	; движение по траектории, ориентация, не постоянная плоскость.
N330 X-20 G40	; отключение коррекции радиуса инструмента.
N340 M30	

Пример 2: Созданная из CAD-системы программа ЧПУ (фрагмент) с CUT3DFD

Программный код	Комментарий
N01 G710	
N03 T="12"	
N06 S5305 M03	
N07 G642	
; подвод к стартовой позиции в MCS с учётом длины инструмента.	

Программный код	Комментарий
G00 G90 X-250.62787 Y-38.37944 A=DC(253.12719) B-12.49543	
G00 G90 Z251.80052	
; конец позиционирования в MCS.	
;	
TRAORI(1)	; выбор трансформации ориентации.
G500	
D1	
CYCLE832(0.01, _TOP_SURFACE_SMOOTH_ON + _ORI_FINISH, 1)	; вызов CYCLE832 через: ; допуск контура = 0,01 мм, ; режим обработки: Top Surface со сглаживанием, ; чистовая обработка с вводом допуска ориентации, ; допуск ориентации = 1 градус
CUT3DFD	
N08 G90 G94	
N09 G00 X-269.21195 Y128.32027 Z1.18577 A3=-.216361688 B3=.934284397 C3=-.283373051	; Кадры с N09 по N10 – быстрая доля движения подвода к детали с постоянной ориентацией.
N10 G00 X-251.90301 Y53.57752 Z23.85561	
N11 G01 X-247.57578 Y34.89183 Z29.52308 F50000.00000	; В кадрах с N11 по N21 реализуется медленная доля движения подвода. Инструмент уже находится рядом с деталью, кроме того, теперь активны настройки изготовления пресс-форм (например, COMPSURF) из CYCLE832 (из-за активной G01). Путь траектории этой так называемой фазы вхождения для поведения изготовления пресс-форм должен составлять приблизительно в 1000 раз больше допуска контура (т. е., в этом примере: 10 мм).
N12 X-247.69126 Y33.82182 Z24.78219 F1061.00000	
N13 X-247.76560 Y33.13299 Z21.73022	
N14 X-247.82755 Y32.55897 Z19.18691	
N15 X-247.87918 Y32.08062 Z17.06748	
N16 X-247.92220 Y31.68200 Z15.30129	
N17 X-247.95805 Y31.34981 Z13.82947 A3=-.216361686 B3=.934284391 C3=-.283373071	
N18 X-247.98792 Y31.07299 Z12.60295 A3=-.216360662 B3=.934280801 C3=-.283385691	
N19 X-248.01282 Y30.84230 Z11.58085 A3=-.216336015 B3=.934194446 C3=-.283689030	
N20 X-248.03357 Y30.65006 Z10.72910 A3=-.216233089 B3=.933833626 C3=-.284952647	
N21 X-248.05086 Y30.48986 Z10.01931 A5=-.060687572 B5=.974940255 C5=-.214029243 A3=-.215712821 B3=.932005189 C3=-.291263295	
N22 **G41** X-248.06237 Y30.32400 Z9.36695 A5=-.060431854 B5=.973045457 C5=-.222554556 A3=-.214974689 B3=.929398552 C3=-.300007025 F1061.03295	; Начиная с кадра N22 нормаль к плоскости впервые полностью определена на протяжении всего кадра (т. е., нормаль к плоскости есть в конце предшествующего кадра N21 и, тем самым, в начале кадра N22 и в конце кадра N22). Таким образом выполнено условие включения коррекции на инструмент посредством G41/G42.

Программный код	Комментарий

```
N23 X-248.07130 Y30.15119 Z8.71082
    A5=-.060165696 B5=.971048883 C5=-.231179920
    A3=-.214177198 B3=.926684940 C3=-.308841625
N24 X-248.07829 Y29.97126 Z8.05094
    A5=-.059884286 B5=.968941717 C5=-.239928784
    A3=-.213318480 B3=.923853466 C3=-.317789237
N25 X-248.08317 Y29.78487 Z7.38844
    A5=-.059584206 B5=.966718449 C5=-.248807482
    A3=-.212397895 B3=.920898045 C3=-.326854594
N26 X-248.08578 Y29.59254 Z6.72679
    A5=-.059263963 B5=.964380907 C5=-.257793037
    A3=-.211418355 B3=.917822366 C3=-.336012474
...
```

Примечание

При торцовом фрезеровании 3D посредством CUT3DFD для включения коррекции на инструмент при помощи G41/G42 необходимо определение нормалей к плоскости. Программирование G41/G42 без определения нормалей к плоскости приводит к выведению аварийного сообщения.

См. также

CYCLE832 - High Speed Settings (Страница 1156)

Дополнительная информация

Торцовое фрезерование 3D

Для этого типа фрезерования 3D необходимо построчное описание траекторий 3D на поверхности детали. Обычно вычисления выполняются в CAM с учетом формы и размеров инструмента. Постпроцессор записывает в программу обработки детали — наряду с кадрами ЧПУ — ориентации инструмента (при активной 5-осевой трансформации) и G-команду желаемой 3D-коррекции на инструмент. Благодаря этому оператор станка может использовать немного меньшие инструменты, отличающиеся от инструмента, используемого для вычисления траекторий ЧПУ.

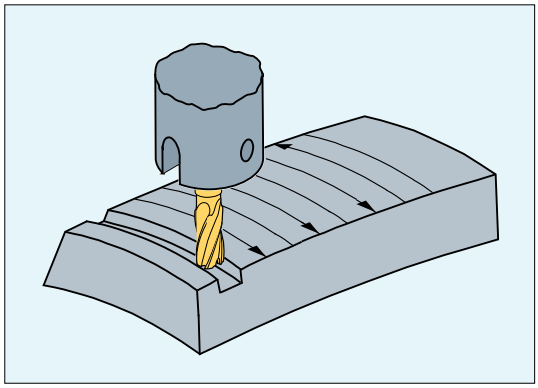


Рисунок 4-11 Торцовое фрезерование

Режим подвода

Режимом подвода в случае 3D-вариантов коррекции на радиус инструмента всегда является NORM.

Поведение на наружных углах

При торцовом фрезеровании наружные углы обрабатываются как окружности с радиусом 0, при этом плоскость окружности образуется конечной касательной первого и стартовой касательной второго кадра. Тем самым возможно изменение ориентации на переходе кадра. Поэтому на наружном угле окружность всегда вставляется как элемент контура. Перемещение точки пересечения при торцовом фрезеровании не доступно.

Поведение при изменениях ориентации на наружных углах

С помощью G-команд ORIC и ORID можно установить, будут ли изменения ориентации, запрограммированные между обоими образующими угол кадрами перемещения, выполняться перед началом вставленного кругового кадра (ORID) или одновременно с ним (ORIC).

Коррекция радиуса инструмента относительно дифференциального инструмента

3D-коррекция на радиус инструмента относительно дифференциального инструмента выбирается командой CUT3DFD. Ее следует применять, если контур запрограммирован относительно траектории центра стандартного инструмента, а обработка осуществляется другим инструментом. В этом случае при расчете коррекции на радиус инструмента 3D учитывают только значение износа радиуса активного инструмента ($TC_DP15) и, при наличии, запрограммированные смещения коррекции на инструмент OFFN и TOFFR/TOFFLR. Базовый радиус ($TC_DP6) активного инструмента **не** учитывается.

Торцовое фрезерование 3D посредством CUT3DFD возможно только в сочетании с "Сглаживание нормалей к плоскости при торцовом фрезеровании 3D". Её активирует вызов лицензируемой функции "Top Surface" посредством CYCLE832(...). Активацию необходимо выполнить **до** включения коррекции на инструмент посредством G41/G42, причём не непосредственно перед зацеплением инструмента, а заранее, за одну длину пути траектории, приблизительно равной 1000-кратному допуску контура (например, 1000 x 0,01 мм = 10 мм). Дективация должна выполняться в обратной последовательности: сначала выключение коррекции инструмента посредством G40, затем, после прохождения одной длины пути траектории, соответствующей приблизительно 1000-кратному допуску контура, деактивация посредством, например, CUT2D (или подобной).

Чтобы иметь возможность применять "Сглаживание нормалей к плоскости при торцовом фрезеровании 3D", дополнительно необходимо разрешить функцию "Интерполяция нормалей к плоскости через полиномы":

MD28291 $MC_MM_SMOOTH_SURFACE_NORMALS = TRUE

Примечание

Для торцового фрезерования 3D посредством CUT3DFD в сочетании с функцией "Top Surface" необходимо соблюдать рекомендации по настройке в отношении "Top Surface"!

Для проверки настроенных данных на портале SIOS есть специальная программа проверки.

→ Программа проверки Top Surface (https://support.industry.siemens.com/cs/ww/en/view/109738423)

См. также

CYCLE832 - High Speed Settings (Страница 1156)

4.13.5.3 Периферийное фрезерование 3D с учетом ограничивающей поверхности (CUT3DCC, CUT3DCCD)

При периферийном фрезеровании 3D с непрерывным или постоянным изменением ориентации инструмента часто программируется траектория центра инструмента для определенного стандартного инструмента. Поскольку на практике часто нет подходящих стандартных инструментов, можно использовать инструмент, не сильно отличающийся от стандартного (≤ 5 %).

CUT3DCCD позволяет учесть для реального дифференциального инструмента такую ограничивающую поверхность, которую описал бы запрограммированный стандартный инструмент. Программа ЧПУ описывает траекторию центра стандартного инструмента.

CUT3DCC учитывает при использовании цилиндрических инструментов такую ограничивающую поверхность, которой достиг бы запрограммированный стандартный инструмент. Программа ЧПУ описывает контур на поверхности обработки.

Вектор нормали к ограничивающей плоскости задается, как и при торцовом фрезеровании 3D, через A4, B4, C4 и A5, B5, C5.

Синтаксис

```
G41/G42 CUT3DCCD/CUT3DCC CDOF2 X... Y... Z... A4=... B4=... C4=... A5=... B5=...
C5=...
...
G40 X... Y... Z...
```

Объяснение

CUT3DCCD:	3D-КРИ для периферийного фрезерования с учетом ограничивающей поверхности с дифференциальным инструментом на траектории центра инструмента: Подача к ограничивающей поверхности	
CUT3DCC:	3D-КРИ для периферийного фрезерования с учетом ограничивающей поверхности с 3D-коррекцией радиуса: контур на поверхности обработки	
G41/G42 X... Y... Z...:	Включение коррекции на радиус инструмента	
	G41:	Коррекция на радиус инструмента слева от контура
	G42:	Коррекция на радиус инструмента справа от контура
	Указание: Включение должно происходить в линейном кадре (G0/G1).	
CDOF2:	Выключение контроля столкновений для периферийного фрезерования 3D	
A4/5=... B4/5=... C4/5=...:	Определение нормали к ограничивающей плоскости	
	A4=... B4=... C4=...:	Определение в начале кадра
	A5=... B5=... C5=...:	Определение в конце кадра
G40 X... Y... Z...:	Выключение коррекции на радиус инструмента **Указание:** Выключение должно происходить в линейном кадре (G0/G1) с движениями гео-оси.	

Примечание

G-команды для выбора 3D-КРИ обрабатываются в кадре подвода, т. е. обычно в кадре, содержащем G41 или G42.

G41 или G42 могут быть запрограммированы и в кадрах без перемещения в значимых для коррекции гео-осях. В этом случае кадром подвода является первый кадр перемещения, следующий за таким кадром.

Смена варианта 3D-КРИ при активной коррекции на радиус инструмента игнорируется, аварийное сообщение не создается.

Пример

Программный код	Комментарий
N10 $TC_DP1[1,1]=120	; цилиндрическая фреза
N20 $TC_DP6[1,1]=10	
N30 $TC_DP15[1,1]=-3	
...	
; обработка цилиндрической фрезой и CUT3DCCD	
N110 TRAORI	; включение трансформации.
N120 A4=0 B4=0 C4=1	; определение нормали к ограничивающей плоскости в начале кадра.
N130 X0 Y0 Z0 A0 C0 T1 D1 F20000	

Программный код	Комментарий
`N140 X10 Y0 Z0 G41 CUT3DCCD CDOF2 G64`	; активация периферийного фрезерования 3D с учетом ограничивающей поверхности + выключение контроля столкновений.
`N150 X20`	
`N160 X30 A45`	; тупой угол ==> подача отсутствует
`N170 X40 A-45`	; острый угол ==> подача
`N180 X55`	
`N190 Y10 Z10`	; движение в направлении инструмента.
`N200 Y20`	
`N210 C45`	; чистое изменение ориентации.
`N220 Y30 C90`	
`N230 A5=-1 B5=0 C5=2 Y40`	; изменение поверхности.
`N240 Y50 G40`	; деактивация коррекции на радиус инструмента.
...	

Дополнительная информация

Тип инструмента

Обрабатывается тип инструмента (параметр инструмента $TC_DP1). Разрешены только фрезы с цилиндрическим хвостовиком (цилиндрическая или концевая фреза, тороидальная фреза и в граничном случае цилиндрическая фреза для изготовления штампов). Это соответствует типам инструментов с номерами 1 – 399, за исключением номеров 111, 155, 156, 157.

Стандартные инструменты с закруглением углов

Закругление углов стандартного инструмента описывается через параметр инструмента $TC_DP7. По параметру инструмента $TC_DP16 получается отклонение закругления углов реального инструмента от закругления углов стандартного инструмента.

Пример: Тороидальная фреза с радиусом меньше радиуса стандартного инструмента

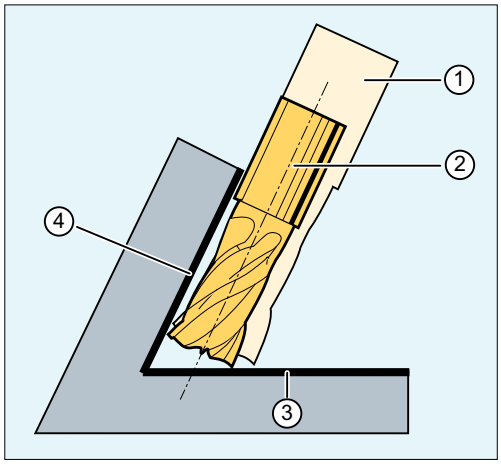
Тип инструмента	Радиус хвостовика (R)	Угловой радиус (r)
Стандартный инструмент с закруглением углов	R = $TC_DP6	r = $TC_DP7
Реальный инструмент с закруглением углов Типы инструмента 121 и 131 — тороидальная фреза (концевая фреза с закруглением углов)	R' = $TC_DP6 + $TC_DP15 + OFFN	r' = $TC_DP7 + $TC_DP16

В этом примере отрицательными являются и $TC_DP15 + OFFN, и $TC_DP16.

3D-КРИ через CUT3DCCD: Траектория центра инструмента с подачей до ограничивающей поверхности

Если используется инструмент, имеющий меньший радиус по сравнению с подходящим стандартным инструментом, то подаваемая в продольном направлении фреза продолжает движение до тех пор, пока она снова не коснется основания кармана. Тем

самым угол, образуемый поверхностью обработки и ограничивающей поверхностью, выбирается настолько, насколько это позволяет инструмент. При этом речь идет о смешанном принципе фрезерования (периферийного и торцового фрезерования). По аналогии с инструментом с уменьшенным радиусом в случае инструмента с увеличенным радиусом соответствующая подача выполняется в противоположном направлении.



① Стандартный инструмент
② Инструмент с меньшим радиусом, с подачей до ограничивающей поверхности
③ Ограничивающая поверхность
④ Поверхность обработки

В отличие от всех других коррекций на инструмент G-группы 22, заданный для CUT3DCCD параметр инструмента $TC_DP6 не имеет значения для радиуса инструмента и не влияет на результирующую коррекцию. Смещение коррекции результирует из суммы значения износа радиуса инструмента (параметр инструмента $TC_DP15) и смещения инструмента OFFN, запрограммированного для вычисления вертикального смещения к ограничивающей поверхности.

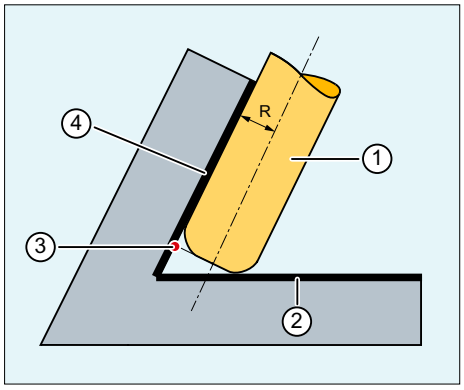
По созданной программы обработки детали не видно, лежит ли обрабатываемая поверхность слева или справа от траектории. Поэтому предполагается положительный радиус и отрицательное значение износа оригинального инструмента. Отрицательное значение износа всегда описывает инструмент с уменьшенным диаметром.

Использование цилиндрических инструментов

При использовании цилиндрических инструментов подача необходима, только если поверхность обработки и ограничивающая поверхность образуют острый угол (меньше 90 градусов). Если используется тороидальная фреза (концевая фреза с закруглением углов), то и для острых углов, и для тупых углов требуется подача в продольном направлении инструмента.

3D-КРИ через CUT3DCC: контур на поверхности обработки

Если CUT3DCC активна с тороидальной фрезой, то запрограммированная траектория относится к условной цилиндрической фрезе того же диаметра. Полученная таким образом исходная точка траектории при использовании тороидальной фрезы представлена на следующем рисунке.



① Тороидальная фреза
② Ограничивающая поверхность
③ Исходная точка траектории
④ Поверхность обработки
R Радиус хвостовика (радиус инструмента)

Разрешен переход угла между поверхностью обработки и ограничивающей поверхностью из острого в тупой угол или наоборот и внутри одного кадра.

По сравнению со стандартным инструментом, используемый реальный инструмент может быть как больше, так и меньше. При этом результирующий угловой радиус не может стать отрицательным, и знак результирующего радиуса инструмента должен быть сохранен.

В случае CUT3DCC программа ЧПУ для обработки детали относится к контуру на обрабатываемой поверхности. При этом, как и при обычной коррекции на радиус инструмента, используется общий радиус, состоящий из следующих компонентов:

- Радиус инструмента (параметр инструмента $TC_DP6),
- Значение износа (параметр инструмента $TC_DP15),
- Смещение инструмента OFFN, запрограммированное для вычисления вертикального смещения к ограничивающей поверхности.

Положение ограничивающей поверхности определяется по следующей разнице:

Размеры стандартного инструмента – радиус инструмента (параметр инструмента $TC_DP6)

Advanced Surface / Top Surface

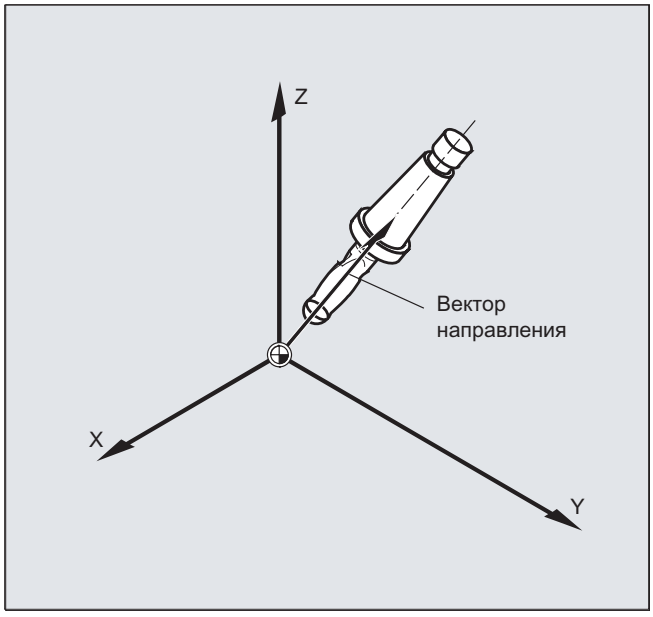
Примечание

При использовании коррекции на радиус инструмента CUT3DCC / CUT3DCCD в сочетании с лицензируемой функцией "Advanced Surface" или "Top Surface" необходимо соблюдать рекомендации по настройке "Advanced Surface" / "Top Surface"! Для проверки настроенных данных на портале SIOS есть специальная программа проверки.

→ Программа проверки для Advanced Surface / Top Surface (https://support.industry.siemens.com/cs/ww/en/view/109738423)

4.13.6 Ориентация инструмента (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST)

Под ориентацией инструмента понимается геометрическая точная установка инструмента в пространстве. Для 5-осевого станка ориентация инструмента может быть настроена через программные команды.



Активированные с `OSD` и `OST` движения перешлифовки ориентации, в зависимости от типа интерполяции, образуются для ориентации инструмента различно.

При активной векторной интерполяции сглаженная характеристика ориентации интерполируется и посредством векторной интерполяции. Напротив, при активной интерполяции круговой оси, ориентация сглаживается напрямую посредством движений круговых осей.

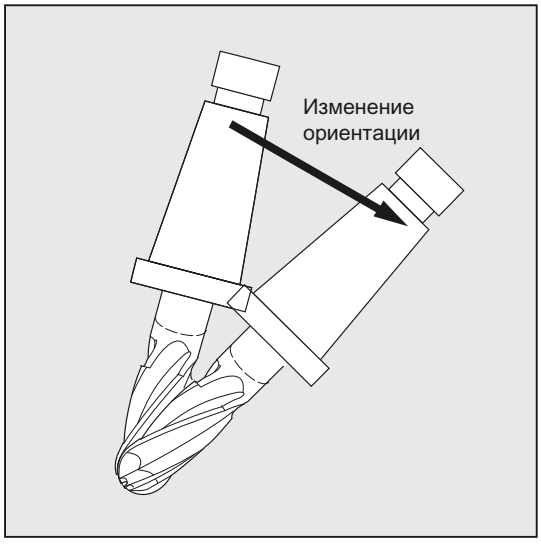
Программирование

Программирование изменения ориентации:

Изменение ориентации инструмента может быть запрограммировано через:

- Прямое программирование круговых осей `A`, `B`, `C` (интерполяция круговой оси)
- Угол Эйлера или угол RPY
- Вектор направления (векторная интерполяция через указание `A3` или `B3` или `C3`)
- `LEAD/TILT` (торцовое фрезерование)

Исходной системой координат является либо система координат станка (`ORIMKS`), либо текущая система координат детали (`ORIWKS`).



Программирование ориентации инструмента:

ORIC:	Ориентация и движение по траектории параллельно
ORID:	Ориентация и движение по траектории последовательно
OSOF:	Нет сглаживания ориентации
OSC:	Постоянная ориентация
OSS:	Сглаживание ориентации только в начале кадра
OSSE:	Сглаживание ориентации в начале и конце кадра
ORIS:	Скорость изменения ориентации при включенном сглаживании ориентации в градусах на мм (действует для OSS и OSSE)
OSD:	Перешлифовка ориентации через задачу длины перешлифовки с установочными данными: SD42674 $SC_ORI_SMOOTH_DIST
OST:	Перешлифовка ориентации через задачу углового допуска в градусах при векторной интерполяции с установочными данными: SD42676 $SC_ORI_SMOOTH_TOL При интерполяции круговой оси заданный допуск принимается как макс. погрешность осей ориентации.

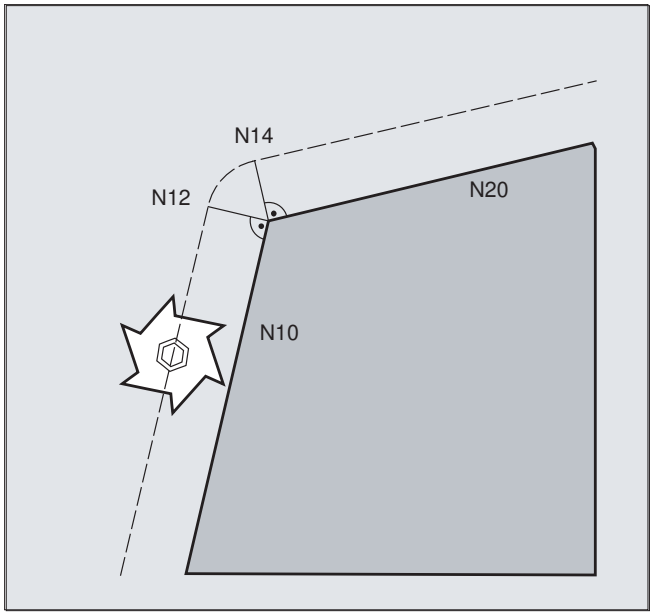
Примечание

Все команды по перешлифовке ориентации инструмента (OSOF, OSC, OSS, OSSE, OSD, OST) собраны в группе G-функций 34. Они действуют модально, т. е. единовременно может действовать только одна из этих команд.

Примеры

Пример 1: ORIC

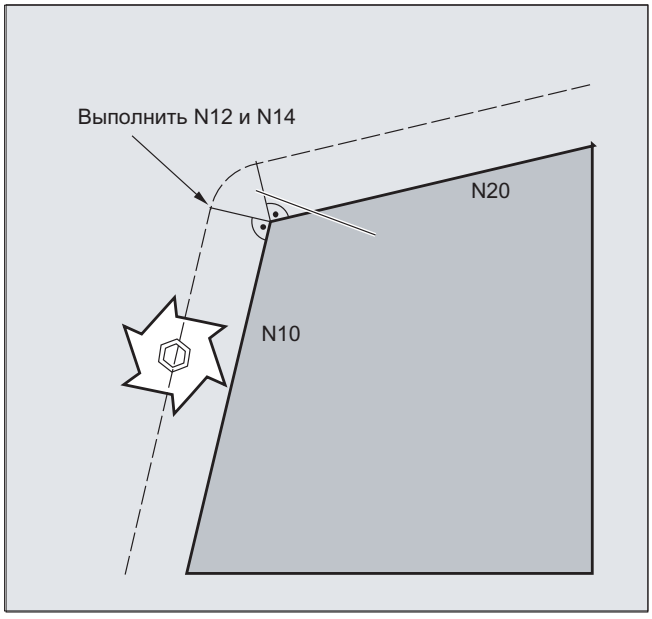
Если между кадрами перемещения N10 и N20 запрограммировано два или более кадров с изменениями ориентации (например, A2=... B2 =... C2=...) и ORIC активна, то вставленный круговой кадр согласно величине угловых изменений распределяется на эти промежуточные кадры.



Программный код	Комментарий
ORIC	
N8 A2=… B2=… C2=…	
N10 X… Y… Z…	
N12 C2=… B2=…	; Круговой кадр, вставляемый на наружном углу, распределяется на N12 и N14, согласно изменению ориентации. Круговое движение и изменение ориентации при этом выполняются параллельно.
N14 C2=… B2=…	
N20 X =…Y=… Z=… G1 F200	

Пример 2: ORID

Если активна ORID, то все кадры между обоими кадрами перемещения выполняются в конце первого кадра перемещения. Круговой кадр с постоянной ориентацией выполняется непосредственно перед вторым кадром перемещения.



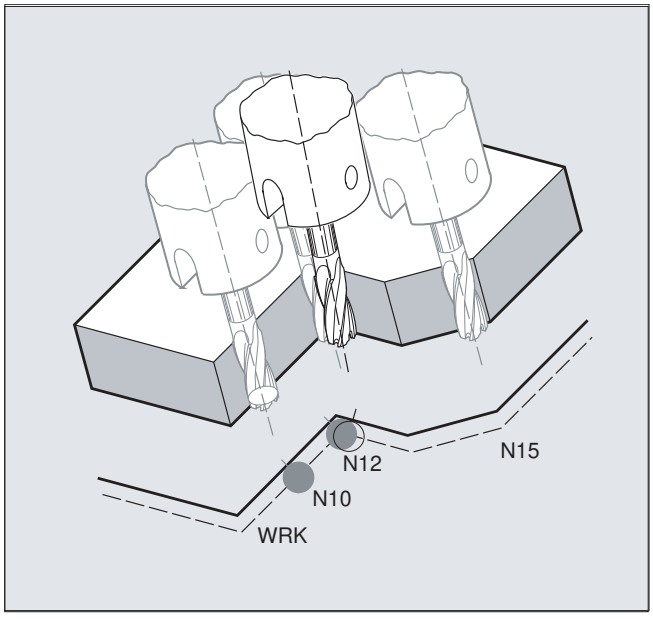
Программный код	Комментарий
`ORID`	
`N8 A2=… B2=… C2=…`	
`N10 X… Y… Z…`	
`N12 A2=… B2=… C2=…`	; Кадр `N12` и `N14` выполняется в конце `N10`. После выводится круговой кадр с текущей ориентацией.
`N14 M20`	; Вспомогательные функции и т. п.
`N20 X… Y… Z…`	

Примечание

Для типа изменения ориентации на наружном углу главной является программная команда, активная в первом кадре перемещения наружного угла.

Без изменения ориентации: Если ориентация на границе кадра не изменяется, то поперечным сечением инструмента является окружность, касающаяся обеих контуров.

Пример 3: Изменение ориентации на внутреннем угле



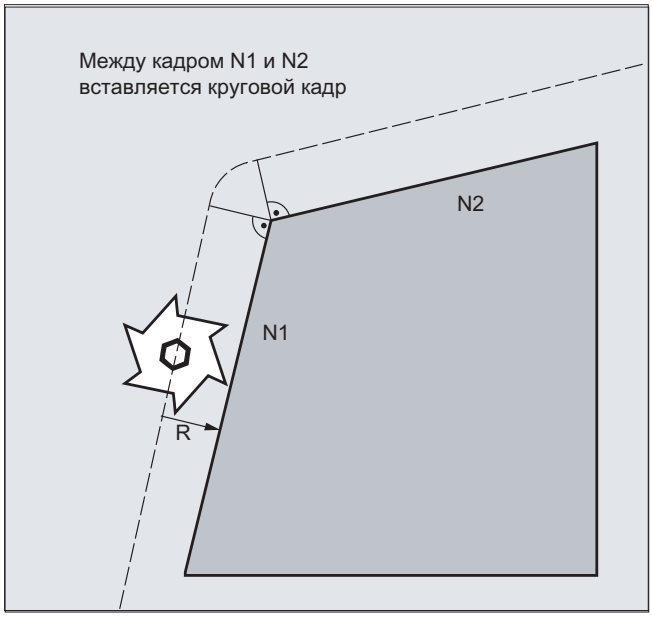
Программный код
ORIC
N10 X …Y… Z… G1 F500
N12 X …Y… Z… A2=… B2=… C2=…
N15 X …Y… Z… A2=… B2=… C2=…

Дополнительная информация

Поведение на наружных углах

На наружном углу всегда вставляется круговой кадр с радиусом фрезы.

С помощью программных команд `ORIC` или `ORID` можно установить, будут ли изменения ориентации, запрограммированные между кадром `N1` и `N2` , выполняться до начала вставленного кругового кадра или одновременно с ним.



Если на наружных углах необходимо изменение ориентации, то оно возможно по выбору параллельно с интерполяцией, или отдельно с движением по траектории.

При ORID сначала выполняются вставленные кадры без движения по траектории. Круговой кадр вставляется непосредственно перед вторым из двух кадров перемещения, образующих угол.

Если на наружном углу вставлено несколько кадров ориентации и ORIC выбрана, то круговое движение согласно значениям отдельных вставленных кадров распределяется на эти кадры.

Перешлифовка ориентации с OSD или OST

При перешлифовке с G642 макс. погрешности для контурных осей и осей ориентации не должны слишком сильно отличаться. Меньший из двух допуск определяет форму движения перешлифовки или угловой допуск, относительно сильное сглаживание характеристики ориентации, не принимая при этом больших погрешностей контура.

Посредством активации OSD или OST с заданной длиной перешлифовки или угловым допуском можно "щедро" сглаживать очень маленькие погрешности характеристики траектории без существенных погрешностей контура.

Примечание

В отличие от перешлифовки контура (и характеристики ориентации) с G642, при перешлифовке ориентации с OSD или OST отдельный кадр не образуется, а движение перешлифовки вставляется напрямую в запрограммированные оригинальные кадры.

С помощью OSD или OST невозможна перешлифовка переходов кадра, на которых выполняется смена типа интерполяции для ориентации инструмента (вектор –> круговая ось, круговая ось –> вектор). При необходимости перешлифовка этих переходов кадра возможна посредством обычных функций перешлифовки G641, G642 или G643.

4.13.7 Присвоение свободных номеров D, номеров резцов

4.13.7.1 Присвоение свободных номеров D, номеров резцов (адрес CE)

номер D

Номера D могут использоваться в качестве номеров коррекции. Дополнительно через адрес CE возможна адресация номера резца. Через системную переменную $TC_DPCE возможно описание номера резца.

Предустановка: Номер коррекции == номер резца

Через машинные данные устанавливается макс. число номеров D (номеров резцов) и макс. число резцов на инструмент (→ изготовитель станка). Следующие команды имеют смысл, только если макс. номер резца (MD18105) был установлен большим, чем число резцов на инструмент (MD18106). Следовать указаниям изготовителя станка.

Дополнительная информация

Справочник по функциям — Инструменты

4.13.7.2 Присвоение свободных номеров D: Проверка номеров D (CHKDNO)

С помощью команды `CKKDNO` проверяется, присвоены ли имеющиеся номера D однозначно. Номера D всех определенных внутри блока TO инструментов могут встречаться только один раз. Запасные инструменты при этом не учитываются.

Синтаксис

```
state=CHKDNO(Tno1,Tno2,Dno)
```

Значение

`state`:	=TRUE:	Номера D были присвоены для проверенного диапазона однозначно.
	=FALSE:	Произошел конфликт номеров D или недействительное параметрирование. Через Tno1, Tno2 и Dno передаются параметры, приведшие к конфликту. Эти данные могут быть проанализированы в программе обработки детали.
`CHKDNO(Tno1,Tno2)`:	Проверяются все номера D названных инструментов.	
`CHKDNO(Tno1)`:	Проверяются все номера D из Tno1 по сравнению со всеми остальными инструментами.	
`CHKDNO`:	Проверяются все номера D всех инструментов по сравнению со всеми остальными инструментами.	

4.13.7.3 Присвоение свободных номеров D: Переименование номеров D (GETDNO, SETDNO)

Номера D должны присваиваться однозначно. Два различных резца одного инструмента не могут иметь один и тот же номер D.

GETDNO

Эта команда выводит номер D определенного резца (ce) инструмента с номером T t. Если номер D для введенных параметров не существует, то устанавливается d=0. Если номер D является недействительным, то возвращается значение больше 32000.

SETDNO

С помощью этой команды присваивается значение d номера D резца ce инструмента t. Через state результат этого оператора возвращается (TRUE или FALSE). Если блок данных для введенных параметров не существует, то возвращается FALSE. Синтаксические ошибки приводят к сигнализации ошибки. Номер D не может быть явно установлен на 0.

Синтаксис

```
d = GETDNO (t,ce)
state = SETDNO (t,ce,d)
```

Значение

d:	Номер D резца инструмента
T:	Номер T инструмента
ce:	Номер резца (номер CE) инструмента
state:	Указывает, была ли команда выполнена без ошибок (TRUE или FALSE).

Пример переименования номера D

Программирование	Комментарий
$TC_DP2[1,2]=120	
$TC_DP3[1,2]=5.5	
$TC_DPCE[1,2] = 3	;номер резца CE
...	
N10 def int DNrAlt, DNrNeu = 17	
N20 DNrAlt = GETDNO(1,3)	
N30 SETDNO(1,3,DNrNeu)	

Таким образом, резцу CE=3 присваивается новое значение D 17. Теперь обращение к данным этого резца выполняется через номер D 17; как через системные переменные, так и при программировании с адресом ЧПУ.

4.13.7.4 Присвоение свободных номеров D: Определение номера T для заданного номера D (GETACTTD)

Предварительно определенная функция GETACTTD вычисляет к абсолютному номеру D соответствующий номер T. Проверка однозначности не выполняется. Если существует несколько одинаковых номеров D внутри одного блока TO, то возвращается номер T первого найденного инструмента.

Синтаксис

<состояние>=GETACTTD(<TNr>,<DNr>)

Объяснение

GETACTTD():	Вызов функции		
<DNr>:	Номер D, для которого должен быть найден номер T.		
	Тип данных:	INT	
<TNr>:	Найденный номер T		
	Тип данных:	VAR INT	
<состояние>:	Результат		
	Тип данных:	INT	
	Величина:	0	Номер T обнаружен. <TNr> содержит значение номера T.
		-1	Номера T для указанного номера D не существует; <TNr>=0.
		-2	Номер D не абсолютный. <TNr> получает значение первого найденного инструмента, содержащего номер D со значением <DNr>.
		-5	Функция не была выполнена по иной причине.

4.13.7.5 Присвоение свободных номеров D: Установка номеров D как недействительных (DZERO)

4.13.8 Кинематика инструментального суппорта

Условия

Инструментальный суппорт может выполнять ориентацию инструмента во всех возможных пространственных направлениях только в том случае, если

- Имеется две оси вращения V_1 и V_2.
- Оси вращения располагаются вертикально друг на друге.
- Продольная ось инструмента располагается вертикально на второй оси вращения V_2.

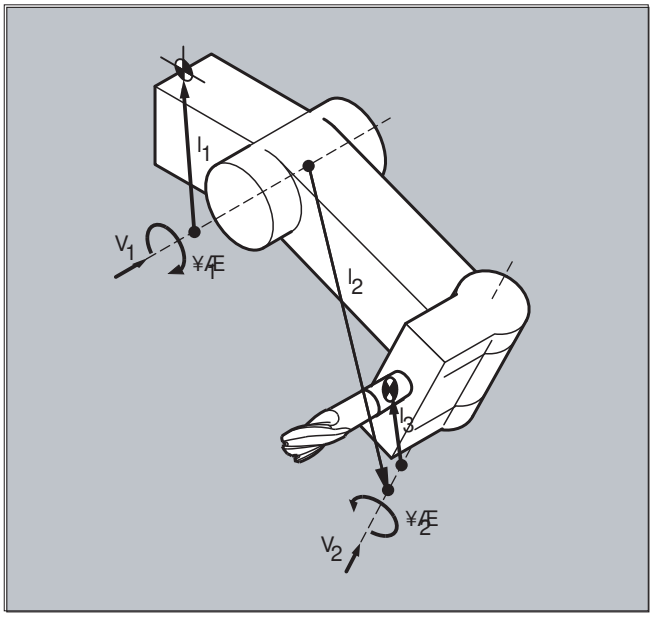
Дополнительно для станков, для которых должны быть возможны любые ориентации, существует следующее требование:

- Ориентация инструмента должна стоять вертикально на первой оси вращения V_1.

Функция

Кинематика инструментального суппорта макс. с двумя осями вращения v_1 или v_2 описывается 17 системными переменными $TC_CARR1[m] до $TC_CARR17[m]. Описание инструментального суппорта состоит из:

- векторного расстояния от первой оси вращения до исходной точки инструментального суппорта I_1, векторного расстояния от первой до второй оси вращения I_2, векторного расстояния от второй оси вращения до исходной точки инструмента I_3.
- векторов направления обеих осей вращения V_1, V_2.
- углов поворота α1, α2 вокруг обеих осей. Углы поворота подсчитываются положительно при направлении взгляда в сторону векторов осей вращения по часовой стрелке.



Для станков с **подвижной кинематикой** (могут вращаться как инструмент, так и деталь) к системным переменным были добавлены элементы $TC_CARR18[m] до $TC_CARR23[m].

Параметры

Функция системных переменных для ориентируемых инструментальных суппортов			
Обозначение	**Компонент x**	**Компонент y**	**Компонент z**
I_1 вектор смещения	$TC_CARR1[m]	$TC_CARR2[m]	$TC_CARR3[m]

Функция системных переменных для ориентируемых инструментальных суппортов			
l_2 вектор смещения	$TC_CARR4[m]	$TC_CARR5[m]	$TC_CARR6[m]
v_1 ось вращения	$TC_CARR7[m]	$TC_CARR8[m]	$TC_CARR9[m]
v_2 ось вращения	$TC_CARR10[m]	$TC_CARR11[m]	$TC_CARR12[m]
α_1 угол поворота α_2 угол поворота	$TC_CARR13[m] $TC_CARR14[m]		
l_3 вектор смещения	$TC_CARR15[m]	$TC_CARR16[m]	$TC_CARR17[m]

Расширения системных переменных для ориентируемых инструментальных суппортов			
Обозначение	**Компонент x**	**Компонент y**	**Компонент z**
l_4 вектор смещения	$TC_CARR18[m]	$TC_CARR19[m]	$TC_CARR20[m]
Идентификатор оси ось вращения v_1 ось вращения v_2	Идентификаторы осей вращения v_1и v_2 (предустановка ноль) $TC_CARR21[m] $TC_CARR22[m]		
Тип кинематики **T**ool **P**art **M**ixed mode	$TC_CARR23[m]		
	Тип кинематики-T ->	Тип кинематики-P ->	Тип кинематики-M
	Вращается только инструмент (предустановка).	Вращается только деталь	Вращаются деталь & инструмент
Смещение оси вращения v_1 оси вращения v_2	Угол в градусах осей вращения v_1и v_2 при принятии первичной установки $TC_CARR24[m] $TC_CARR25[m]		
Угловое смещение оси вращения v_1 оси вращения v_2	Смещение торцового зубчатого зацепления в градусах осей вращения v_1и v_2 $TC_CARR26[m] $TC_CARR27[m]		
Угловая разрешающая способность v_1 ось вращения v_2 ось вращения	Инкремент торцового зубчатого зацепления в градусах осей вращения v_1и v_2 $TC_CARR28[m] $TC_CARR29[m]		
Мин. позиция ось вращения v_1 ось вращения v_2	Программный лимит для мин. позиции осей вращения v_1и v_2 $TC_CARR30[m] $TC_CARR31[m]		
Макс. позиция ось вращения v_1 ось вращения v_2	Программный лимит для макс. позиции осей вращения v_1и v_2 $TC_CARR32[m] $TC_CARR33[m]		
Имя инструментального суппорта	Вместо цифры инструментальный суппорт может получить имя. $TC_CARR34[m]		
Пользователь: имя оси 1 имя оси 2 идентификатор	Предполагаемое использование в измерительных циклах пользователя. $TC_CARR35[m] $TC_CARR36[m] $TC_CARR37[m]		
Позиция	$TC_CARR38[m]	$TC_CARR39[m]	$TC_CARR40[m]

Расширения системных переменных для ориентируемых инструментальных суппортов			
Точное смещение	Параметры, которые могут быть **прибавлены к значениям в базовых параметрах**.		
l_1 вектор смещения	$TC_CARR41[m]	$TC_CARR42[m]	$TC_CARR43[m]
l_2 вектор смещения	$TC_CARR44[m]	$TC_CARR45[m]	$TC_CARR46[m]
l_3 вектор смещения	$TC_CARR55[m]	$TC_CARR56[m]	$TC_CARR57[m]
l_4 вектор смещения	$TC_CARR58[m]	$TC_CARR59[m]	$TC_CARR60[m]
v_1 ось вращения	$TC_CARR64[m]		
v_2 ось вращения	$TC_CARR65[m]		

Примечание

Пояснения к параметрам

С "m" указывается соответствующий номер описываемого инструментального суппорта.

$TC_CARR47 до $TC_CARR54, а также $TC_CARR61 до $TC_CARR63, не определены и приводят к аварийному сообщению при попытке обращения по чтению или записи.

Начальные или конечные точки векторов расстояния на осях могут выбираться свободно. Углы поворота α_1, α_2 вокруг обеих осей в исходном состоянии инструментального суппорта определяются с 0°. Таким образом, для описания кинематики инструментального суппорта имеется бесконечное число возможностей.

Инструментальные суппорта с только одной или без осей вращения могут быть описаны через установку на ноль векторов направления одной или обеих осей вращения.
Для инструментального суппорта без оси вращения векторы расстояния действуют как дополнительные коррекции инструмента, компонент которых не затрагиваются переключением плоскостей обработки (G17 до G19)°.

Расширения параметров

Параметры осей вращения

Системные переменные были дополнены строками $TC_CARR24[m] до $TC_CARR33[m] и описаны следующим образом:

Смещение осей вращения v_1, v_2	Изменение позиции оси вращения v_1 или v_2 при первичной установке ориентируемого инструментального суппорта.
Угловое смещение/ угловая разрешающая способность осей вращения v_1, v_2	Смещение или угловая разрешающая способность торцового зубчатого зацепления осей вращения v_1 и v_2. Запрограммированный или вычисленный угол округляется до ближайшего значения, получаемого при целом n из phi = s + n * d.
Мин. и макс. позиция оси вращения v_1, v_2	Мин. позиция/макс. позиция оси вращения, предельный угол (программный лимит) оси вращения v1 и v2.

Параметры для пользователя

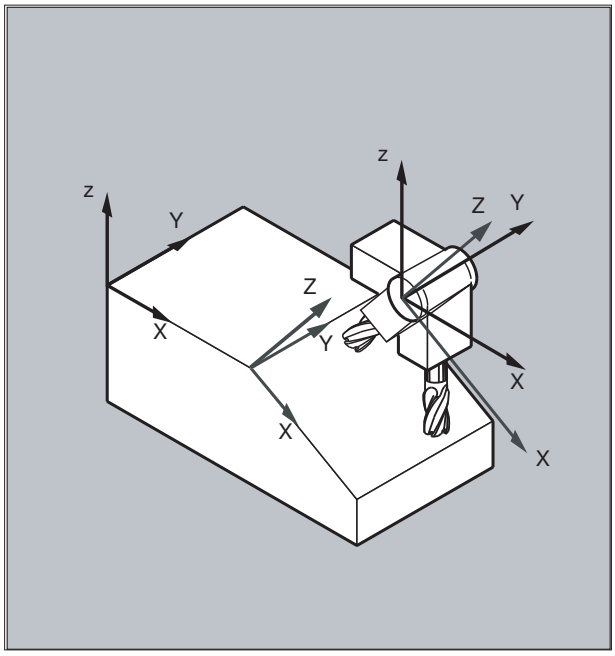
$TC_CARR34 до $TC_CARR40 содержат параметры, свободно доступные пользователю, которые до ПО 6.4 стандартно более не обрабатываются в NCK или не имеют значения.

Параметры точного смещения

$TC_CARR41 до $TC_CARR65 содержат параметры точного смещения, которые могут прибавляться к значениям в базовых параметрах. Согласованное с базовым параметром значение точного смещения получается через прибавление значения 40 к номеру параметра.

Пример

Используемый в следующим примере инструментальный суппорт может быть полностью описан через вращение вокруг оси Y.



Программный код	Комментарий
`N10 $TC_CARR8[1]=1`	; Определение компонента `Y` первой оси вращения инструментального суппорта 1.
`N20 $TC_DP1[1,1]=120`	; Определение концевой фрезы.
`N30 $TC_DP3[1,1]=20`	; Определение концевой фрезы с длиной 20 мм.
`N40 $TC_DP6[1,1]=5`	; Определение концевой фрезы с радиусом 5 мм.
`N50 ROT Y37`	; Определение фрейма с вращением в 37° вокруг оси `Y`.
`N60 X0 Y0 Z0 F10000`	; Подвод в исходную позицию.

Программный код	Комментарий
`N70 G42 CUT2DF TCOFR TCARR=1 T1 D1 X10`	; Установить коррекцию радиуса, коррекцию длин инструмента в повернутом фрейме, выбрать инструментальный суппорт 1, инструмент 1.
`N80 X40`	; Выполнить обработку при повороте на 37°.
`N90 Y40`	
`N100 X0`	
`N110 Y0`	
`N120 M30`	

Дополнительная информация

Подвижная кинематика

Для станков с подвижной кинематикой (вращается как инструмент, так и деталь) к системным переменным были добавлены строки `$TC_CARR18[m]` до `$TC_CARR23[m]` и они были описаны следующим образом:

Поворотный инструментальный стол, состоящий из:

- векторного расстояния от второй оси вращения V_2 до исходной точки поворотного инструментального стола I_4 третьей оси вращения.

Круговые оси, состоящие из:

- обеих идентификаторов канала для исходной точки осей вращения V_1 и V_2, к позиции которых при необходимости выполняется обращение при определении ориентации ориентируемого инструментального суппорта.

Тип кинематики с одним из значений T, P или M:

- Тип кинематики T: вращается только инструмент.
- Тип кинематики P: вращается только деталь.
- Тип кинематики M: вращаются инструмент и деталь.

Удаление данных инструментального суппорта

С помощью `$TC_CARR1[0]=0` могут быть удалены данные всех блоков данных инструментального суппорта.

Типу кинематики `$TC_CARR23[T]=T` должна быть присвоена одна из трех допустимых прописных или строчных букв (T,P,M) и по этой причине он не должен быть удален.

Изменение данных инструментального суппорта

Любое из описанных значений может быть изменено посредством присваивания нового значения в программе обработки детали. Любой другой символ, кроме T, P или M, приводит при попытке активации ориентируемого инструментального суппорта к ошибке.

Чтение данных инструментального суппорта

Любое из описанных значений может быть считано посредством присваивания переменной в программе обработки детали.

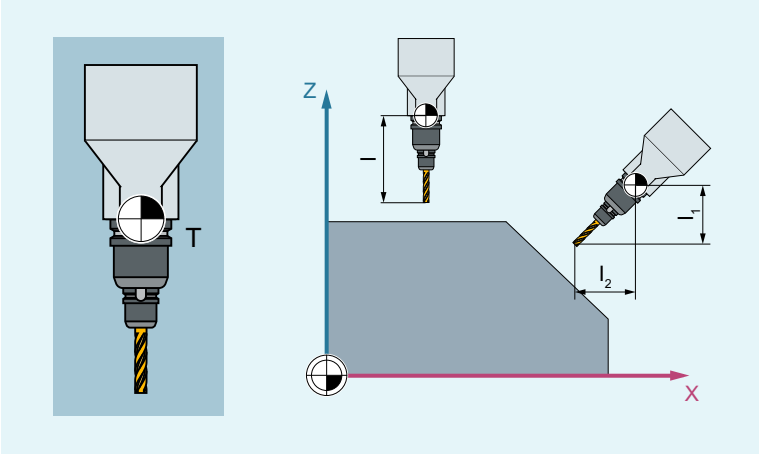
Точные смещения

Недопустимое значение точного смещения определяется только тогда, когда активируется ориентируемый инструментальный суппорт, который содержит такое смещение и одновременно установочные данные SD42974
$SC_TOCARR_FINE_CORRECTION = TRUE

Величина допустимого точного смещения ограничивается через машинные данные до макс. допустимого значения.

4.13.9 Коррекция длин инструмента для ориентируемого инструментального суппорта (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)

При изменении ориентации инструментального суппорта и тем самым инструмента в пространстве изменяются и его компоненты длин:



T	Исходная точка инструментального суппорта
l_1, l_2	Компоненты длин инструмента

Поэтому после переоснащения, к примеру, посредством ручной установки или замены инструментального суппорта с фиксированной точной установкой в пространстве, необходимо заново определить его компоненты длин инструмента. Для этого есть команды группы G 42 "Инструментальный суппорт".

Синтаксис

```
TCARR=<m>
TCOABS
TCOFR
TCOFRZ/TCOFRY/TCOFRX
```

Объяснение

Элемент	Тип	Значение	
TCARR=<m>	Адрес	Запросить инструментальный суппорт	
		<m>	Номер инструментального суппорта
TCOABS	G-команда	Определить компоненты длин инструмента из актуальной ориентации инструментального суппорта	
TCOFR	G-команда	Определить компоненты длин инструмента из ориентации активного фрейма	
TCOFRZ/TCOFRY/TCOFRX	G-команда	Команда TCOFRX/TCOFRY/TCOFRZ исходит от ориентированного в соответствующем направлении (X/Y/Z) инструмента и вычисляет углы установки ориентируемого инструментального суппорта, так что посредством активного фрейма выполняется ориентация инструмента в том же направлении.	
		TCOFRZ	Ориентированный в направлении Z инструмент направлен так, что в активном фрейме он также ориентирован в направлении Z.
		TCOFRY	Ориентированный в направлении Y инструмент направлен так, что в активном фрейме он также ориентирован в направлении Y.
		TCOFRX	Ориентированный в направлении X инструмент направлен так, что в активном фрейме он также ориентирован в направлении X.

Дополнительная информация

Подбор инструментального суппорта (TCARR)

С инструментальным суппортом связан блок данных инструментального суппорта, описывающий его геометрию. Если в момент выбора инструментального суппорта инструмент активен, инструмент присваивается новому выбранному инструментальному суппорту, и данные геометрии инструментального суппорта немедленно активируются. Смена или новое программирование активного инструмента не требуется. Если активируется новый инструмент, то он обрабатывается так, как если бы он был установлен на активный инструментальный суппорт.

TCARR=0 отключает активный инструментальный суппорт.

Примечание

Актуальные данные геометрии инструментального суппорта можно определить и в программе ЧПУ через соответствующие системные переменные (см. раздел "Доступ к блокам данных инструментального суппорта").

Примечание

С точки зрения СЧПУ номера инструментальных суппортов <m> и номера инструментов <n> можно комбинировать свободно. В реальном приложении комбинации могут исключаться по причинам, связанным с обработкой или механикой. СЧПУ не проверяет на наличие комбинаций, имеющих смысл.

Доступ к блокам данных инструментального суппорта

Доступ к актуальным данным геометрии инструментального суппорта из программы ЧПУ следующий:

- Запись:
  $TC_CARR<n>[<m>]=<значение>
- Чтение:
  <значение>=$TC_CARR<n>[<m>]
  (<значение> должно быть переменной типа данных REAL)

Ели указанный инструментальный суппорт не определен, отображается аварийное сообщение.

Обнуление блоков данных инструментального суппорта

Из программы ЧПУ все значения всех блоков данных инструментального суппорта можно удалить следующей командой:

```
$TC_CARR1[0] = 0
```

Отдельные блоки данных инструментального суппорта можно удалять выборочно посредством предварительно определенной процедуры DELTC.

Действие выбора инструментального суппорта

Инструментальный суппорт начинает действовать, если активированы и инструментальный суппорт, и инструмент. Выбор лишь одного инструментального суппорта не имеет действия.

Действие выбора инструментального суппорта зависит от команды G-группы 42 (инструментальный суппорт), запрограммированной вместе с TCARR.

Смена G-команды из этой группы вызывает повторное вычисление компонентов длин инструмента, если инструментальный суппорт активен.

Коррекция длин инструмента из ориентации инструментального суппорта (TCOABS)

TCOABS вычисляет коррекцию длин инструмента из актуальных углов ориентации инструментального суппорта; зафиксированы в системных переменных $TC_CARR13 и $TC_CARR14. Учитываемая ориентация не зависит от ориентации активного фрейма.

Определение кинематики инструментального суппорта с системными переменными представлено в главе "Кинематика инструментального суппорта (Страница 802)" Руководства по программированию — Программирование ЧПУ".

Определение компонентов длин инструмента по ориентации активного фрейма (TCOFR)

При TCOFR углы ориентации инструментального суппорта определяются по ориентации активного фрейма Но зафиксированные в данных инструментального суппорта значения сохраняются неизменными. Поэтому они также используются, чтобы снять многозначность, получаемую при вычислении углов поворота по фрейму (см. следующий абзац).

Многозначности

Определенную заданную посредством фрейма ориентацию инструмента можно установить с помощью двух осей, как правило, через **две** различные пары углов

поворота. Из этих двух возможных позиций СЧПУ выбирает такую, при которой углы поворота расположены как можно ближе к запрограммированным углам поворота.

Примечание

Во всех случаях возникновения многозначности, как правило, необходимо сохранить ожидаемые из фрейма углы в данных инструментального суппорта.

Ориентация инструмента в активном фрейме (TCOFRZ/TCOFRY/TCOFRX)

Независимо от активной ориентации команда TCOFRX/TCOFRY/TCOFRZ исходит при вычислении углов установки инструментального суппорта из ориентации в направлении X/Y/Z. Выбор плоскости обработки (G17/G18/G19) не влияет на ориентацию в активном фрейме. Но он влияет на вычисление коррекции длин инструмента.

Смена фрейма

После выбора инструмента пользователь может сменить фрейм. Это не влияет на компоненты коррекции на длину инструмента.

Определенные через фреймы углы поворота не влияют на углы поворота, сохраненные в данных инструментального суппорта. При переключении с TCOFR на TCOABS исходные (запрограммированные) углы поворота в данных инструментального суппорта снова активируются.

Повторное вычисление коррекции длин инструмента

Для нового вычисления коррекции длин инструмента после смены фрейма необходимо заново выбрать инструмент. При вычислении коррекции длин инструмента на промежуточном этапе вычисляется и угол поворота инструментального суппорта. Поскольку у инструментальных суппортов с двумя осями вращения, как правило, существует две пары углов поворота, с помощью которых ориентация инструмента может быть согласована с активным фреймом, то зафиксированные в данных инструментального суппорта значения углов поворота хотя бы приблизительно должны соответствовать установленным механически углам поворота.

Примечание

Практически во всех случаях, при которых может возникнуть многозначность, необходимо сохранить ожидаемые из фрейма углы в данных инструментального суппорта.

Примечание

Ориентация инструмента должна быть вручную согласована с активным фреймом.

Примечание

СЧПУ не может проверить вычисленный через ориентацию фрейма угол поворота на предмет возможности его установки на станке. Если конструкция станка такова, что оси вращения инструментального суппорта не могут достигнуть вычисленной через ориентацию фрейма ориентации инструмента, то выводится ошибка.

Примечание

Комбинация точной коррекции инструмента и функциональностей для коррекции длин инструмента при подвижных инструментальных суппортах запрещена. При попытке одновременного вызова обеих функций, следует сообщение об ошибке.

Примечание

TOFRAME (Страница 342) позволяет определить фрейм на основе направления ориентации выбранного инструментального суппорта.

Примечание

При активной трансформации ориентации (3-, 4-, 5-осевая трансформация) можно выбрать инструментальный суппорт с отличной от нулевого положения ориентацией, ошибка при этом не выводится.

Примечание

С активным фреймом без вращения ориентация посредством TCOFRX/TCOFRY/TCOFRZ ведет к тривиальному решению, поскольку ориентация инструмента уже указывает в направлении X/Y/Z.

Ориентация в направлении Z (TOFRAME)

Команда TOFRAME G-группы 53 (Вращение фрейма относительно инструмента) позволяет задать фрейм, направление Z которого параллельно направлению ориентации выбранного инструментального суппорта.

Если нет активного инструментального суппорта или активен инструментальный суппорт без изменения ориентации, то направление Z в новом фрейме следующее:

- При G17 идентично старому направлению Z
- При G18 идентично старому направлению Y
- При G19 идентично старому направлению X

Передаваемые параметры стандартных и измерительных циклов

Для передаваемых параметров стандартных и измерительных циклов действуют определенные диапазоны значений.

В значениях углов заданы следующие диапазоны значений:

Вращение вокруг	Диапазон значений
1-я гео-ось	-180 ° ≤ угол ≤ +180 °
2-я гео-ось	-90 ° ≤ угол ≤ +90 °
3-я гео-ось	-180 ° ≤ угол ≤ +180 °

Примечание

При передаче угловых значений в стандартный или измерительный цикл учитывать:

Значения меньше дискретности вычисления ЧПУ округлять до нуля!

Дискретность вычисления ЧПУ для угловых позиций установлена в машинных данных:

MD10210 $MN_INT_INCR_PER_DEG

4.13.10 Модифицирование ориентируемых инструментальных суппортов после замеров станка (CORRTC)

Функция `CORRTC` позволяет записывать измеренные элементы кинематической цепи инструментального магазина в специальные элементы коррекции.

Примечание

Записанные при помощи функции CORRTC значения коррекции не сразу вступают в силу в инструментальном магазине. Значения коррекции вступают в силу только после отмены инструментального магазина, применения NEWCONF и выбора инструментального магазина.

Синтаксис

```
<_Corr_Status> = CORRTC(<_Corr_Vect>, <_Corr_Index>, <_Corr_Mode>,
[ <_No_Alarm>])
```

Объяснение

CORRTC:	Вызов функции		
<_Corr_Status>:	Возвращаемое значение функции		
	Тип данных:	INT	
	Значения:	0	Функция была выполнена без ошибок.
		1	Нет активного инструментального магазина.
		2	Активный инструментальный магазин не был определен посредством кинематических цепей.
		10	Параметр вызова <_Corr_Index> отрицательный.
		11	Параметр вызова <_Corr_Mode> отрицательный.
		12	Недействительная ссылка на сегмент подцепочки (_CORR_INDEX).
		13	На сегменте, на который выполняется ссылка параметром _CORR_INDEX, элемент коррекции не определен ($TC_CARR_CORR_ELEM).
		20	100-я позиция _CORR_MODE недействительна. Допустимы только значения 0 и 1.
		21	1000-я позиция _CORR_MODE недействительна. Допустимы только значения 0 и 1.
		30	При коррекции вектора смещения отклонение по сравнению с текущим значением, по меньшей мере, в одной координате больше максимального значения, заданного в установочных данных SD41612 $SN_CORR_TOCARR_LIN_MAX.
		31	Попытка записи системной переменной была отклонена из-за отсутствия прав записи.
<_Corr_Vect>:	Вектор коррекции Содержание вектора коррекции определяется последующими параметрами <Corr_Index> и <_Corr_Mode>. Если <_Corr_Status> = 30, содержание вектора перезаписывается (см. выше).		
	Тип данных:	REAL	
<_Corr_Index>:	Означает сегмент, для которого необходимо скорректировать вектор направления элемента коррекции.		
	Тип данных:	INT	

<_Corr_Mode>:	Режим коррекции		
	Тип данных:	INT	
	Параметр <Corr_Mode> записан в десятичном коде (с 1-й позиции по 1000-ю позицию).		
	1-я позиция:	Зарезервировано	
	10-я позиция:	Определяет, как необходимо модифицировать элемент коррекции, на который ссылается содержание <_Corr_Index>.	
		xx**0**x	Вектор коррекции записывается непосредственно в элемент коррекции. Этот вариант можно использовать для непосредственного описания элемента коррекции, причем не требуется знать индекс <n> соответствующих системных данных ($NK_OFF_DIR[<n>, ...]).
		xx**1**x	Как 0, но с тем отличием, что переданное значение коррекции интерпретируется в мировой системе координат. Разница между вариантами 0 и 1 всегда возникает в случае, если кинематическая цепь содержит в исходном положении (позиции всех круговых осей равны 0) дополнительные вращения.
		xx**2**x	Как 1, но с тем отличием, что значение коррекции относится ко всему сегменту, т. е. в элемент коррекции вносится такое значение, что весь сегмент достигает длины, определенной посредством значения коррекции.
	100-я позиция:	Определяет, как следует интерпретировать содержание параметра <_Corr_Vect>.	
		x**0**xx	Переданный вектор коррекции <_Corr_Vect> содержит всю новую длину элемента коррекции или сегмента, на который ссылается <_Corr_Index> в сочетании с 10-й позицией <_Corr_Mode> (абсолютная коррекция).
		x**1**xx	Переданный вектор коррекции <_Corr_Vect> содержит только разницу по отношению к текущей длине элемента коррекции или сегмента, на который ссылается <_Corr_Index> в сочетании с 10-й позицией <_Corr_Mode> (инкрементальная коррекция).
	1000-я позиция:	Определяет, необходимо ли ограничить максимально допустимую коррекцию при помощи установочных данных $SN_ CORR_TOCARR_LIN_MAX.	
		0xxx	Контроль предельных значений активен.
		1xxx	Контроль предельных значений подавлен.
<_No_Alarm>:	Поведение в случае аварийного сообщения (возвращаемое значение > 0) (**опционально**)		
	Тип данных:	BOOL	
	Значение:	FALSE (по умолчанию)	В случае ошибки выполнение программы останавливается, и выводится аварийное сообщение.
		TRUE	В случае ошибки выполнение программы не останавливается и аварийное сообщение не отображается. Пример использования: Специализированная реакция согласно возвращаемому значению

Дополнительная информация по CORRTC

Кинематическая структура инструментального магазина описывается посредством одной (тип T и тип P) или двух (тип M) кинематических цепей (подцепочек), исходящих из соответствующей исходной точки (исходной точки машины или исходной точки инструментального суппорта). Одна из этих цепей — цепь инструмента — заканчивается в исходной точке инструмента, другая — цепь детали — в нулевой точке базовой кинематической системы (BKS).

Функция CORRTC записывает осевые направления в станках с инструментальными магазинами в специальные элементы коррекции. Кинематическая цепь описывается, кроме прочего, посредством элементов типа OFFSET, определяемых посредством $NK_TYPE.

CORRTC работает с сегментами.

Обе подцепочки могут подразделяться не более чем на четыре сегмента:

- Сегмент 1 начинается в стартовой точке цепочки и заканчивается на первой круговой оси.
- Сегмент 2 – сегмент между круговой осью 1 и круговой осью 2.
- Сегмент 3 – сегмент между круговой осью 2 и концом цепочки.

На следующем рисунке показан ориентируемый инструментальный суппорт с 2 круговыми осями.

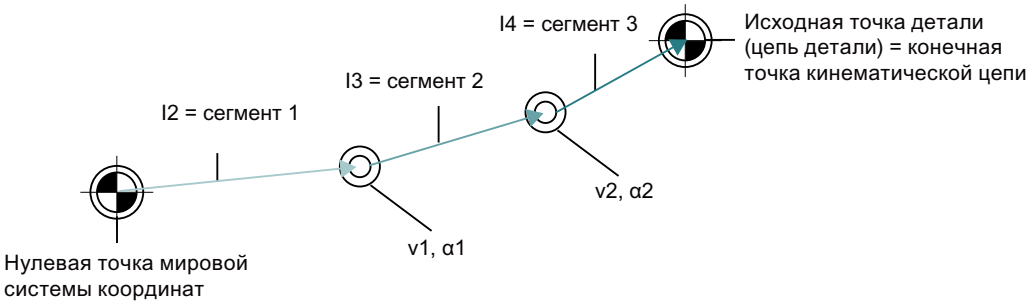


Рисунок 4-12 Пример CORRTC

Сегменты имеют однозначное определение: При следовании по кинематической подцепочке от ее стартовой точки до конечной точки получается, что у первого сегмента — индекс 1, у следующего сегмента — индекс 2 и т. д.

Элементы коррекции

На каждый постоянный элемент кинематической цепи (элемент цепи типа $NK_TYPE[<n>] = "OFFSET") в каждом из этих сегментов можно ссылаться посредством системной переменной $TC_CARR_CORR_ELEM [, 0 ... 3]. При помощи функции CORRTC в обозначенные таким образом элементы записываются значения коррекции, установленные при замерах станка.

Последовательность ссылок в $TC_CARR_CORR_ELEM[m, 0..3] должна соответствовать описанным выше сегментам, т. е., в $TC_CARR_CORR_ELEM[m, 0] может находиться только элемент цепи, относящийся к вектору смещения **I**1, и т. д.

При этом исходное значение — это всегда значение, действующее в инструментальном магазине, активном при вызове CORRTC. Измененное после выбора инструментального магазина содержание данных кинематики в сохранении данных не влияет на действие функции CORRTC.

4.13.11 Коррекция длин инструмента Online (TOFFON, TOFFOF)

Через системную переменную $AA_TOFF[<n>] возможно трехмерное наложение эффективных длин инструмента согласно трем направлениям инструмента в реальном времени.

В качестве индекса <n> используются три идентификатора гео-осей. Тем самым, число активных направлений коррекции определено через активные в это же время гео-оси.

Все коррекции могут быть активны одновременно.

Функция коррекции длин инструмента Online может использоваться для:

- Трансформация ориентации TRAORI
- Ориентируемого инструментального суппорта TCARR

Примечание

Коррекция длин инструмента Online эта **опция**, требующая разрешения. Эта функция имеет смысл только в комбинации с активной трансформацией ориентации или активным ориентируемым инструментальным суппортом.

Синтаксис

```
TRAORI
TOFFON(<направление коррекции>[,<значение смещения>])
WHEN TRUE DO $AA_TOFF[<направление коррекции>]                    ; В синхронных действиях.
...
TOFFOF(<направление коррекции>)
```

Объяснение

TOFFON:	**Активировать** коррекцию длин инструмента Online	
	<направление коррекции>:	Направление инструмента (X, Y, Z), в котором должна действовать коррекция длин инструменте Online.
	<значение смещения>:	При активации для соответствующего направления коррекции может быть указано значение смещения, которое сразу же выводится.
TOFFOF:	**Сбросить** коррекцию длин инструмента Online Значения коррекции в указанном направлении коррекции сбрасываются и запускается остановка предварительной обработки.	

Примеры

Пример 1: Выбор коррекции длин инструмента

Программный код	Комментарий
MD21190 $MC_TOFF_MODE =1	; Подвод к абсолютным значениям.
MD21194 $MC_TOFF_VELO[0] =1000	
MD21196 $MC_TOFF_VELO[1] =1000	
MD21194 $MC_TOFF_VELO[2] =1000	
MD21196 $MC_TOFF_ACCEL[0] =1	
MD21196 $MC_TOFF_ACCEL[1] =1	
MD21196 $MC_TOFF_ACCEL[2] =1	
N5 DEF REAL XOFFSET	
N10 TRAORI(1)	; Трансформация вкл.
N20 TOFFON(Z)	; Активация коррекции длин инструмента `Online` для направления инструмента `Z`.
N30 WHEN TRUE DO $AA_TOFF[Z]=10 G4 F5	; Для направления инструмента `Z` выполняется интерполяция коррекции длин инструмента 10.
...	
N100 XOFFSET=$AA_TOFF_VAL[X]	; Присвоить актуальную коррекцию в направлении X.
N120 TOFFON(X,-XOFFSET) G4 F5	; Для направления инструмента `X` коррекция длин инструмента снова возвращается на 0.

Пример 2: Отмена коррекции длин инструмента

Программный код	Комментарий
N10 TRAORI(1)	; Трансформация вкл.
N20 TOFFON(X)	; Активация коррекции длин инструмента `Online` для направления инструмента X.
N30 WHEN TRUE DO $AA_TOFF[X] = 10 G4 F5	; Для направления инструмента X выполняется интерполяция коррекции длин инструмента 10.
...	
N80 TOFFOF(X)	; Смещение позиции направления инструмента `X` удаляется: `...$AA_TOFF[X]=0` Оси не перемещаются. К актуальной позиции в `WCS` добавляется смещение позиции согласно актуальной ориентации.

Дополнительная информация

Подготовка кадра

При подготовке кадра на предварительной обработке действующее в главном ходе актуальное смещение длин инструмента также учитывается. Для возможности

практически полного использования макс. допустимых осевых скоростей необходимо остановить подготовку кадра с остановкой предварительной обработки STOPRE при установлении смещения инструмента.

Смещение инструмента всегда известно на момент предварительной обработки и тогда, когда коррекции длин инструмента после старта программы более не изменяются, или когда после изменения коррекций длин инструмента было обработано больше кадров, чем буфер IPO может принять между предварительной обработкой и главным ходом.

Переменная $AA_TOFF_PREP_DIFF

Мера для разницы между актуальной действующей в интерполяторе коррекцией и коррекцией, которая действовала на момент подготовки кадра, может быть запрошена в переменной $AA_TOFF_PREP_DIFF[<n>] .

Установка машинных и установочных данных

Для коррекции длин инструмента Online имеются следующие системные данные:

- MD20610 $MC_ADD_MOVE_ACCEL_RESERVE (резерв ускорения для наложенного движения)
- MD21190 $MC_TOFF_MODE
  Содержание системной переменной $AA_TOFF[<n>] выводится или интегрируется как абсолютное значение.
- MD21194 $MC_TOFF_VELO (скорость коррекции длин инструмента Online)
- MD21196 $MC_TOFF_ACCEL (ускорение коррекции длин инструмента Online)
- Установочные данные для задачи предельных значений:
  SD42970 $SC_TOFF_LIMIT (верхняя граница значения коррекции длин инструмента)

4.13.12 Изменение данных коррекции для вращающихся инструментов

4.13.12.1 Вычисление ориентаций (ORISOLH)

Предварительно определенная функция ORISOLH помогает пользователю настраивать позиции круговых осей станка так, чтобы токарный инструмент приводился в определенное, не зависящее от кинематики положение относительно детали. В данном случае исходят из того, что активна 6-осевая трансформация, параметры которой заданы посредством кинематических цепей.

Доступны две основных функции:

- Точная установка инструмента
  Задаются оба угла β и γ. Функция вычисляет необходимые для этого углы трёх осей ориентации.
- Точная установка инструмента (прямая)
  Задаются углы второй и третьей осей ориентации. Функция вычисляет по ним соответствующие углы β и γ, а также угол недостающей первой оси ориентации.

Примечание

Последовательность осей ориентации

Если проходить кинематическую цепь, описывающую строение станка, от детали до инструмента, то для последовательности трёх осей ориентации 6-осевой трансформации действует следующее:

- Ближайшая к **детали** ось ориентации — **первая** ось ориентации.
- Ближайшая к **инструменту** ось ориентации — **третья** ось ориентации.

Первая ось ориентации, будет, как правило, шпинделем, и поэтому в таких случаях соответствующее вращение выполняется посредством вращающего фрейма.

Синтаксис

```
<RetVal> = ORISOLH(<Cntrl>,<W1>,<W2>)
```

Объяснение

ORISOLH:	Вызов функции		
<RetVal>:	Возвращаемое значение функции		
	Тип данных:	INT	
	Диапазон значений:	0, -2, -3, ..., -17	
	Значения:	0	Функция была завершена без ошибок.
		-2	Ни одна из действительных трансформаций (6-осевая трансформация ориентации) не активна.
		-3	Первый параметр (<Cntrl>) отрицательный.
		-4	1-я позиция первого параметра (<Cntrl>) недействительна. Допустимы только значения 0 и 1.
		-5	10-я позиция первого параметра (<Cntrl>) недействительна. Допустимы только значения от 0 до 3.
		-6	100-я позиция первого параметра (<Cntrl>) недействительна. Допустимы только значения 0 и 1.
		-7	1000-я позиция первого параметра (<Cntrl>) недействительна. Допустимы только значения от 0 до 3.
		-8	В случае функции "Точная установка инструмента (прямая)" угол γ слишком большой.
		-9	В случае функции "Точная установка инструмента (прямая)", по меньшей мере, одна из заданных позиций осей нарушает границу оси.
		-10	Ни один из инструментов не активен.
		-11	Требуемая ориентация не может быть настроена.
		-12	Подгонка свободных осевых углов при хиртовом зубчатом зацеплении для первого или единственного решения невозможна.
		-13	Подгонка свободных осевых углов при хиртовом зубчатом зацеплении для второго решения невозможна.
		-14	Подгонка свободных осевых углов при хиртовом зубчатом зацеплении не возможна ни для одного из двух решений.
		-15	Первая ось ориентации параметризирована как хиртовая ось.
		-16	И вторая, и третья круговая ось параметризированы как хиртовые оси. Только одна из обеих осей может быть хиртовой.
		-17	В случае функции "Поворот прямо", по меньшей мере, одна из заданных позиций осей не совместима с соответствующим хиртовым зубчатым зацеплением.

<Cntrl>:	управляет поведением функции		
	Тип данных:	INT	
	Параметр <Cntrl> записан в десятичном коде (с 1-й позиции по 1000-ю позицию).		
	1-я позиция:	1-я позиция управляет поведением при ошибках:	
		xxx**0**	В случае ошибки (возвращаемое значение < 0) обработка программы прерывается, и выводится аварийное сообщение 14106. **Указание:** Независимо от значения 1-й позиции аварийное сообщение выводится и в том случае, если параметр <Cntrl> отрицательный.
		xxx**1**	В случае ошибки (возвращаемое значение < 0) аварийное сообщение **не** выводится. Пользователь может самостоятельно надлежащим образом отреагировать на неполадку.
	10-я позиция:	управляет поведением при наличии оси ориентации с хиртовым зубчатым зацеплением: **Указание:** Этот параметр обрабатывается только в случае функции "Точная установка инструмента" (т. е., если 100-я позиция имеет значение "0").	
		xx**0**x	Позиция оси округляется до ближайшей позиции.
		xx**1**x	Позиции осей округляются так, что отклонение угла β от его запрограммированного значения становится минимальным.
		xx**2**x	Позиции осей округляются так, что значение угла β равно максимально возможному значению, которое меньше запрограммированного значения (β округляется).
		xx**3**x	Позиции осей округляются так, что значение угла β равно минимально возможному значению, которое больше запрограммированного значения (β округляется).
	100-я позиция:	задает, какая функция должна быть выполнена, или какое значение имеют оба последующих параметра <W1> и <W2>.	
		x**0**xx	Функция "Точная установка инструмента" Параметры <W1> и <W2> имеют следующее значение: • <W1> = β • <W2> = γ Вычисляются соответствующие углы осей ориентации.
		x**1**xx	Функция "Точная установка инструмента (прямая)" <W1> — заданное значение позиции второй оси ориентации, <W2> — заданное значение позиции третьей оси ориентации 6-осевой трансформации. Определяются позиция первой оси ориентации и углы β и γ, совместимые с обоими заданными значениями позиций. Если ошибка не появляется, в системной переменной $P_ORI_POS[<n>, <m>] всегда выводятся два решения. При этом первый индекс <n> (0 или 1) ссылается на решение, а второй индекс <m> (0 ... 2) — на ось ориентации: • $P_ORI_POS[0/1, **0**]: Позиция первой оси ориентации • $P_ORI_POS[0/1, **1**]: Угол β

			• $P_ORI_POS[0/1, **2**]: Угол γ Проверяется, совместимы ли заданные значения позиций <W1> и <W2> с активными хиртовыми зубчатыми зацеплениями (если имеются) или активными программными лимитами. Если нет, возвращается соответствующий номер ошибки (см. параметр <RetVal>). При любом выборе обоих углов <W1> и <W2> резец инструмента, как правило, находится не в плоскости обработки. Угол γ, на который поворачивается резец из плоскости обработки, не должен быть больше предельного значения, определенного посредством установочных данных SD42999 $SC_ORISOLH_INCLINE_TOL.
	1000-я позиция:	Задает, как при необходимости модифицируются позиции решений, если 100-я позиция имеет значение "0", т. е. в случае функции "Точная установка инструмента".	
		0xxx	Вычисленные позиции осей должны быть расположены максимально близко к актуальным полициям осей станка.
		1xxx	Вычисленные позиции осей должны быть в случае осей по модулю расположены максимально близко к центру диапазона по модулю, в случае других осей — максимально близко к 0. Для осей не по модулю это означает, что позиции осей ограничены диапазоном -180° ... +180°.
		2xxx	Вычисленные позиции осей должны быть независимо от типа оси ограничены диапазоном -180° ... +180°.
<W1>:	первый угол Значение результирует из 100-й позиции параметра <Cntrl>.		
	Тип данных:	REAL	
<W1>:	Второй угол Значение результирует из 100-й позиции параметра <Cntrl>.		
	Тип данных:	REAL	

Примечание

Незапрограммированные параметры имеют стандартное значение "0".

Дополнительная информация

Количество обнаруженных решений, а также другая информация о состоянии при выполнении функции ORISOLH можно считывать следующими системными переменными:

Системная переменная	**Объяснение**		
$P_ORI_POS [<n>, <m>]	Возвращает углы осей ориентации, получающиеся при программировании ориентации.		
	<n>:	Индекс решения	
		Диапазон значений:	0, 1
	<m>:	Индекс оси ориентации	
		Диапазон значений:	0 ... 2
			Последовательность осей ориентации (1 ... 3) относится к определению осей в $NT_ROT_AX_NAME.
	При вызове функции ORISOLH в режиме "Точная установка инструмента (прямая)" переменные $P_ORI_POS[0/1, 1] и P_ORI_POS[0/1, 2] содержат относящиеся к обоим решениям значения обоих углов β и γ. Первое внесенное в $P_ORI_POS[<n>, <m>] решение, т. е. с индексом <n> = 0, всегда то, которое выбирает система СЧПУ, если выполняется прямой подвод к требуемой ориентации. Второй индекс <m> относится при этом к оси ориентации, т. е. к $NT_ROT_AX_NAME. Внесенные в $P_ORI_POS[<n>, <m>] позиции осей учитывают смещения, внесенные в $NK_OFF и $NK_OFF_FINE, т. е., эти осевые углы могут использоваться в следующих кадрах без дальнейшей модификации для настройки желаемой ориентации. Если круговая ось является хиртовой осью, то позиции решения округляются до ближайшего следующего фиксированного положения хиртового зубчатого зацепления. Для круговых осей с хиртовым зубчатым зацеплением в системной переменной $P_ORI_DIFF могут считываться разницы между позициями осей для точных решений и скорректированными по хиртовой развертке решениями.		
$P_ORI_DIFF [<n>, <m>]	Возвращает разницу между точными и предоставляемыми в $P_ORI_POS позициями осей ориентации, получающимися при программировании ориентации.		
	<n>:	Индекс решения	
		Диапазон значений:	0, 1
	<m>:	Индекс оси ориентации	
		Диапазон значений:	0 ... 2
			Последовательность осей ориентации (1 ... 3) относится к определению осей в $NT_ROT_AX_NAME.
	Содержание может быть не равно нулю, только если позиции растрируются (хиртовое зубчатое зацепление), т. е., если системные данные $NT_HIRTH_INCR соответствующей оси не равны нулю, и если эта ось является ручной круговой осью.		

Системная переменная	Объяснение		
$P_ORI_SOL	Если при трансформации ориентации с более чем одной осью ориентации вычисляются осевые углы, которые должны вести к заданной ориентации, то существует более одного решения. В системной переменной $P_ORI_SOL содержится количество действительных решений, а также дополнительная информация о состоянии. Содержание $P_ORI_SOL кодировано следующим образом:		
	Значения < 0	Общие состояния ошибки	
		-1	Для активной трансформации решения еще не были вычислены (отсутствие вызова ORISOLH).
		-2	Ни одна из трансформаций не активна, или активная трансформация не является трансформацией ориентации (6-осевая трансформация), которая может предоставить позиции для заданного программирования ориентации.
		-4	С имеющейся кинематикой желаемую ориентацию настроить невозможно.
		-5	При вызове функции ORISOLH в режиме "Точная установка инструмента (прямая)" решений обнаружено не было.
		-6	При вызове функции ORISOLH в режиме "Точная установка инструмента (прямая)" угол γ слишком большой.
		-7	При вызове функции ORISOLH в режиме "Точная установка инструмента (прямая)" был задан угол, который из-за хиртового зубчатого зацепления настроить невозможно.
		-8	Первая ось ориентации (фреймовая ось) не должна иметь параметры хиртовой оси.
		-9	И вторая, и третья круговая ось параметризированы как хиртовые оси. Только одна из обеих осей может быть хиртовой.
		-10	Подгонки решения (решений) по хиртовому зубчатому зацеплению обнаружено не было.
	Значения > 0	Количество математически возможных решений без учёта границ оси и возможных условий ошибок.	
	1-я позиция (разряд единиц)	0	Решения не существует, т. е., требуемую ориентацию настроить невозможно. У такого случая может быть три различных причины: • Требуемая ориентация принципиально недостижима из-за кинематики станка (не перпендикулярно размещенные оси ориентации) и при любом диапазоне перемещения осей ориентации. В этом случае 10-я и 100-я позиции $P_ORI_SOL равняются нулю, а присвоенные осям ориентации переменные состояния $P_ORI_STAT получают значение "-4". • Вычисленные решения недостижимы, поскольку они нарушили бы ограничения осей. Позиции осей ориентации, которые получились бы без ограничений осей, можно считывать в $P_ORI_POS. • При вызове функции ORISOLH в режиме "Точная установка инструмента (прямая)" позиции осей были заданы так, что либо вектор ориентации, либо вектор

Системная переменная	Объяснение		
			ориентации нормали инструмента установлены параллельно первой оси ориентации, позицию которой необходимо вычислить. В этих случаях позиция этой оси не определена.
		1	Существует одно решение. У такого случая может быть три различных причины: • На основе заданной ориентации, кинематики станка и без учёта границ оси есть только одно решение (с математической точки зрения речь идёт о двух совпадающих решениях). Этот случай возникает в случае непрямоугольных кинематик на краю диапазона ориентации. $P_ORI_POS содержит оба (идентичных) решения. • Есть только одно решение, поскольку из-за нарушения границ оси второе решение недействительно. В этом случае действительное решение — это всегда первое решение в $P_ORI_POS. Второе решение, которое получилось бы без учёта границ оси, может быть также считано в $P_ORI_POS. • При вызове функции ORISOLH в режиме "Точная установка инструмента (прямая)" это является обычной ситуацией. По заданным позициям двух осей ориентации существует только одна действительная позиция подлежащей вычислениям отсутствующей оси ориентации.
		2	Существует два решения.
		8	Существует бесконечное количество решений, т. е., позиция оси ориентации (полярной оси) — любая. Однако из обоих возможных положений другой оси (других осей) одно исключено из-за нарушения границ оси.
		9	Существует бесконечное количество решений, т. е., позиция оси ориентации (полярной оси) — неопределенная. Неопределенную ось можно определить по 100-й позиции или по системной переменной $P_ORI_STAT.
	Значения > 0 10-я позиция (разряд десятков)	Индикация (в двоичном коде) нарушенных границ оси. Точную причину ошибки можно определить по системной переменной $P_ORI_STAT.	
		Бит 0 (значение 10):	По меньшей мере, для одного решения, нарушена, по меньшей мере, одна граница 1-й оси ориентации.
		Бит 1 (значение 20):	По меньшей мере, для одного решения, нарушена, по меньшей мере, одна граница 2-й оси ориентации.
		Бит 2 (значение 40):	По меньшей мере, для одного решения, нарушена, по меньшей мере, одна граница 3-й оси ориентации.
	Значения > 0	Индикация (в двоичном коде) не определенных позиций осей (может возникать, только если есть бесконечное количество решений, т. е., если 1-я позиция равна "9").	

Системная переменная	**Объяснение**		
	100-я позиция (разряд сотен)	Бит 0 (значение 100):	Позиция 1-й оси ориентации не определена.
		Бит 1 (значение 200):	Позиция 2-й оси ориентации не определена.
		Бит 2 (значение 400):	Позиция 3-й оси ориентации не определена.
	Обозначения 1-й, 2-й и 3-й оси ориентации относятся к определению осей в $NT_ROT_AX_NAME.		

Системная переменная	**Объяснение**			
$P_ORI_STAT [<n>]	Возвращает для каждой из максимально 3 осей ориентации состояние после вызова ORISOLH.			
	<n>:	Индекс оси ориентации (соответствует индексу соответствующей оси ориентации в $NT_ROT_AX_NAME)		
		Диапазон значений:	0 ... 2	
			Последовательность осей ориентации (1 ... 3) относится к определению осей в $NT_ROT_AX_NAME.	
	Содержание $P_ORI_STAT кодировано следующим образом:			
	Значения < 0	Общие состояния ошибки		
		-1	Состояние не определено (отсутствие вызова ORISOLH).	
		-2	Ни одна из трансформаций не активна, или активная трансформация не является трансформацией ориентации (6-осевая трансформация), которая может предоставить позиции для заданного программирования ориентации.	
		-3	В активной трансформации ось не содержится.	
		-4	Позицию оси вычислить невозможно, поскольку при имеющейся кинематике требуемая ориентация недостижима независимо от принятого диапазона перемещения оси.	
		-5	При вызове функции ORISOLH в режиме "Точная установка инструмента (прямая)" позиции осей были заданы так, что либо вектор ориентации, либо вектор ориентации нормали инструмента установлены параллельно первой оси ориентации, позицию которой необходимо вычислить. В этих случаях позиция этой оси не определена.	
		-6	При вызове функции ORISOLH в режиме "Точная установка инструмента (прямая)" угол γ слишком большой.	
		-7	При вызове функции ORISOLH в режиме "Точная установка инструмента (прямая)" был задан угол, который из-за хиртового зубчатого зацепления настроить невозможно.	
		-8	Первая ось ориентации (фреймовая ось) не должна иметь параметры хиртовой оси.	
		-9	И вторая, и третья круговая ось параметризированы как хиртовые оси. Только одна из обеих осей может быть хиртовой.	
		-10	Подгонки решения (решений) по хиртовому зубчатому зацеплению обнаружено не было.	
	Значения > 0	Индикация (в двоичном коде) нарушенных границ оси первого решения.		
	1-я позиция (разряд единиц)	Бит 0 (значение 1):	Первое решение нарушает нижнюю границу оси.	
		Бит 1 (значение 2):	Первое решение нарушает верхнюю границу оси.	
	Значения > 0	Индикация (в двоичном коде) нарушенных границ оси второго решения.		
		Бит 0 (значение 10):	Второе решение нарушает нижнюю границу оси.	

Системная переменная	Объяснение		
	10-я позиция (разряд десятков)	Бит 1 (значение 20):	Второе решение нарушает верхнюю границу оси.
	Значения > 0 100-я позиция (разряд сотен)	Индикация не определенной позиции оси.	
		Бит 0 (значение 100):	Позиция оси ориентации не определена, т. е., требуемая ориентация достигается при любой настройке круговой оси (полюсное положение). Эта информация содержится также в системной переменной $P_ORI_SOL.
	Одновременно может отображаться несколько номеров ошибок, указывающих на нарушение границ оси. Поскольку при нарушении границы оси предпринимается попытка достижения позиции в пределах разрешенных границ оси посредством сложения или вычитания кратного 360°, то (если это невозможно) нет неоднозначного определения того, которая граница оси была нарушена — верхняя или нижняя. Если для требуемой ориентации решения нет ($P_ORI_SOL = 0), то состояние содержащихся в трансформации осей ориентации равно "0".		

Примечание

$NT_ROT_AX_NAME

Эта системная переменная ссылается не более чем на 3 оси, служащие для настройки ориентации. Она содержит имена элементов цепи ($NK_NAME), определяющих те оси станка (круговые оси), которые должны выполнять движения ориентации, результирующих из кинематической трансформации. Последовательность, в которой содержится не более трех круговых осей в этой системной переменной, не имеет значения для кинематики станка, поскольку она выводится из структуры кинематических цепей. Поскольку она определяет последовательность, в которой другие переменные обращаются к круговым осям, последовательность осей ориентации в $NT_ROT_AX_NAME должна совпадать с описанием кинематики.

Примечание

Информация о состоянии

Информация о состояниях, которая отражает, например, что ориентация недостижима или достижима только с нарушением значимых границ оси, не вызывает аварийного сообщения ЧПУ. В обязанности пользователя входит должное реагирование на указанные условия.

4.13.12.2 Активация модификации данных коррекции для вращающихся инструментов (CUTMOD, CUTMODK)

Модификация данных коррекции для вращающихся инструментов активируется в программе ЧПУ языковой командой CUTMOD (в сочетании с ориентируемыми инструментальными суппортами) или CUTMODK (для трансформаций ориентации, определяемых посредством кинематических цепей).

Примечание

Поскольку ориентируемые инструментальные суппорты и трансформации ориентации, определенные посредством кинематических цепей, не могут быть активны одновременно, конфликтов между обоими вариантами не возникает.

Синтаксис

```
CUTMOD = <Value>
```

или

```
CUTMODK = <Command>
```

Объяснение

CUTMOD:	Вызов функции в сочетании с ориентируемыми инструментальными суппортами		
<Value>:	присвоенное значение		
	Тип данных:	INT	
	Значение:	0	Функция деактивирована. Возвращаемые системной переменной $P_AD... значения идентичны соответствующим параметрам инструмента.
		> 0	Функция активируется, если активен ориентируемый инструментальный суппорт с указанным номером, т. е. активация привязана к определенному ориентируемому инструментальному суппорту. Возвращаемые системной переменной $P_AD... значения возможно изменены по сравнению с соответствующими параметрами инструмента в зависимости от активного вращения. Деактивация обозначенного ориентируемого инструментального суппорта деактивирует функцию временно, активация другого ориентируемого инструментального суппорта деактивирует ее постоянно. Поэтому в первом случае функция снова активируется при повторном выборе того же ориентируемого инструментального суппорта, во втором случае необходим повторный выбор, причем и в том случае, когда позднее ориентируемый инструментальный суппорт с указанным номером активируется заново. Reset не влияет на функцию.
		-1	Функция активируется всегда при активном ориентируемом инструментальном суппорте. При смене инструментального суппорта или при его отключении и повторном выборе позднее, повторной установки CUTMOD не требуется.
		-2	Функция активируется всегда при активном ориентируемом инструментальном суппорте, номер которого идентичен таковому актуального активного ориентируемого инструментального суппорта. Если нет активного ориентируемого инструментального суппорта, то это равнозначно CUTMOD=0. Если ориентируемый инструментальный суппорт активен, то это равнозначно прямому указанию актуального инструментального суппорта.
		< -2	Значения меньше 2 игнорируются, т. е. этот случай обрабатывается так, как если бы CUTMOD не была запрограммирована. **Указание:** Этот диапазон значений не должен использоваться, т.к. он зарезервирован для возможных будущих расширений.
CUTMODK:	Вызов функции в сочетании с трансформациями ориентации, определяемыми посредством кинематических цепей		

<Command>:	присвоенная команда		
	Тип данных:	STRING	
	Значение:	"NEW"	Сохраняются значимые для функции "Модификация данных коррекции" состояния активной трансформации, определенной посредством кинематических цепей , имя трансформации и актуальный фрейм контура. **Указание:** Эта команда допустима, только если подходящая трансформация (TRAORI_DYN, TRAORI_STAT или TRAANG_K) активна.
		"OFF"	Выключает активную функцию "Модификация данных коррекции". Сохраненные до этого командой "NEW" данные сохраняются. **Указание:** Эта команда допустима также, если CUTMODK не активна. В этом случае она не оказывает влияния. Блок данных для "Модификация данных коррекции" (при наличии) сохраняется.
		"ON"	Эта команда повторно активирует "Модификацию данных коррекции" с блоком данных, сохраненным ранее командой "NEW". Если при выполнении этой команды активна трансформация с именем сохраненного набора данных, "Модификация данных коррекции" немедленно становится действительной. В противном случае активация откладывается до тех пор, пока не будет активирована текущая трансформация.
		"CLEAR"	Как и команда "OFF", выключает функцию "Модификация данных коррекции" и в дополнение к этому удаляет сохраненный блок данных. **Указание:** Эта команда допустима также, если CUTMODK не активна.

Примечание

SD42984 $SC_CUTDIRMOD

Активируемая командой CUTMOD или CUTMODK функция заменяет активируемую через установочные данные SD42984 $SC_CUTDIRMOD функцию. Но эта функция продолжает оставаться доступной без изменений. Но поскольку параллельное использование обеих функций нерационально, она может быть активирована, только если CUTMOD равна нулю, а CUTMODK равна нулевой строке.

Дополнительная информация

Считывание модифицированных данных коррекции

Модифицированные данные коррекции предоставляются в следующих системных переменных и переменных BTSS:

Объяснение	Системная переменная	BTSS-переменная
Положение резца	$P_AD[2]	cuttEdgeParam2
Угол державки	$P_AD[10]	cuttEdgeParam10
Направление резания	$P_AD[11]	cuttEdgeParam11
Задний угол	$P_AD[24]	cuttEdgeParam24

Данные всегда модифицированы по сравнению с соответствующими параметрами инструмента ($TC_DP2[..., ...], ...), если функция "Модификация данных коррекции для вращающихся инструментов" была активирована командой `CUTMOD` или `CUTMODK`, и инструмент вращался посредством ориентируемого инструментального суппорта или подходящей трансформации ориентации.

Другие значимые для функций системные переменные

Системная переменная	Объяснение
$P_CUTMOD_ANG / $AC_CUTMOD_ANG	Возвращает угол, на который был повернут инструмент в активной плоскости обработки, и который лежит в основе определения модифицированных данных резцов функций CUTMOD или CUTMODK.
$P_CUTMOD / $AC_CUTMOD	Считывает текущее действительное значение, запрограммированное последним с помощью команды CUTMOD (номер инструментального суппорта, для которого необходимо активировать модификацию данных коррекции). Если последнее запрограммированное значение `CUTMOD=-2` (активация с текущим активным ориентируемым инструментальным суппортом), то в системной переменной возвращается не значение "-2", а номер активного на момент программирования ориентируемого инструментального суппорта.
$P_CUTMODK / $AC_CUTMODK	Считывает имя трансформации, под которым был сохранён действительный на данный момент блок данных для "Модификация данных коррекции".

Системная переменная	Объяснение
$P_CUT_INV / $AC_CUT_INV	Возвращает значение TRUE, если инструмент повернут таким образом, что необходима инверсия направления вращения шпинделя. Для этого в кадре, к которому относится соответствующая операция чтения, должны быть выполнены следующие четыре условия: 1. Активен токарный или шлифовальный инструмент (типы инструмента 400 до 599 и/ или SD42950 $SC_TOOL_LENGTH_TYPE = 2). 2. Модификация данных коррекции была активирована командой CUTMOD или CUTMODK. 3. Активны ориентируемый инструментальный суппорт или определенная посредством кинематических цепей трансформация ориентации, который/которая были выбраны командой CUTMOD или CUTMODK. 4. Ориентируемый инструментальный суппорт или кинематическая трансформация ориентации вращают инструмент таким образом, что результирующая нормаль резца инструмента повернута более чем на 90° по отношению к исходному положению (обычно 180°). Если, по меньшей мере, одно из названных условий не выполнено, то переменная возвращает значение FALSE. Для инструментов, положение резцов которых не определено, значение переменной всегда FALSE.

Системная переменная	Объяснение	
$P_CUTMOD_ERR	Ошибка после последнего вызова функции CUTMOD Функция CUTMOD может быть вызвана и не явно при смене инструмента. При сбросе (Reset) переменная сбрасывается на нуль. При каждой смене инструмента она сначала сбрасывается на нуль и при необходимости описывается заново. Переменная кодирована в битах. Биты имеют следующие значения:	
	Бит 0:	Для активного инструмента действительное направление резания не определено.
	Бит 1:	Оба угла резца (задний угол и угол державки) активного инструмента равны нулю.
	Бит 2:	Задний угол активного инструмента имеет недопустимое значение (< 0° или > 180°).
	Бит 3:	Угол державки активного инструмента имеет недопустимое значение (< 0° или > 90°).
	Бит 4:	Угол режущей пластины активного инструмента имеет недопустимое значение (< 0° или > 90°).
	Бит 5:	Комбинация "положение резца – угол державки" активного инструмента недопустима (при положении резцов от 1 до 4 угол державки должен быть ≤ 90°, при положении резцов от 5 до 8 он должен быть ≥ 90°).
	Бит 6:	Недопустимое вращение активного инструмента. Инструмент был вывернут на ± 90° (с погрешностью приблизительно в 1°) из активной плоскости обработки. Из-за этого положение резца в плоскости обработки более не определено.
	Бит 7:	Режущая пластина расположена не в плоскости обработки, и угол между режущей пластиной и плоскостью обработки превышает верхнюю границу, заданную в установочных данных SD42998 $SC_CUTMOD_PLANE_TOL.
	Бит 8:	Режущая пластина расположена не в плоскости обработки. Величина угла α больше 1°. Угол α — угол поворота вокруг оси координат, которая перпендикулярна как оси вращения угла β, так и оси вращения угла γ (для G18 — ось X).

$P_...: Переменные предварительной обработки

$AC_...: Переменные главного хода

Все переменные главного хода могут считываться в синхронных действиях. Обращение по чтению из предварительной обработки создает останов предварительной обработки.

Смена плоскостей

Для определения измененного положения резца, направления резания и угла державки или заднего угла определяющим является рассмотрение резца в соответствующей активной плоскости (G17 – G19).

Если же установочные данные SD42940 $SC_TOOL_LENGTH_CONST (смена компонентов длин инструмента при смене плоскостей) содержат действительное значение, отличное от нуля (плюс или минус 17, 18 или 19), то их содержание определяет плоскость, в которой рассматриваются релевантные величины.

Это правило преимущества установочных данных по отношению к G-кодам можно отменить, установив бит 18 машинных данных $MC_TOOL_PARAMETER_DEF_MASK. Это

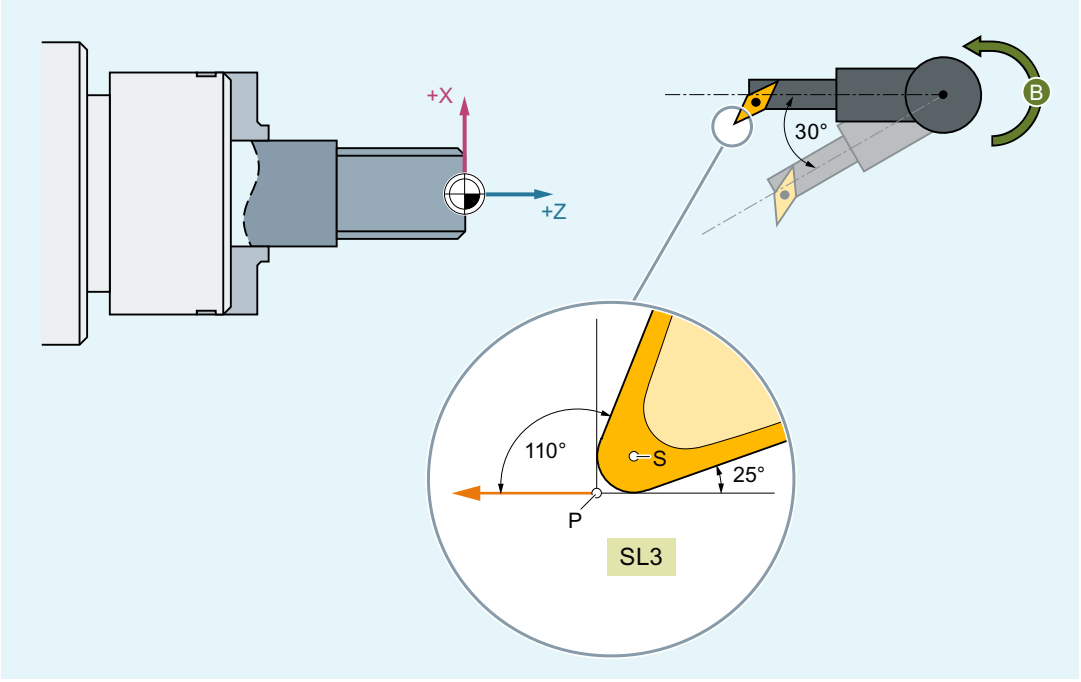
означает, что если этот бит установлен, то всегда действует плоскость, определенная посредством G-команды группы 6.

Активность измененных данных резцов

Измененное положение резцов и измененная опорная точка резцов сразу же активируются при программировании и для уже активного инструмента. Нового выбора инструмента для этого не требуется.

Пример

У инструмента с положением резца 3 и ориентируемым инструментальным суппортом, который может вращать инструмент вокруг оси B, необходимо модифицировать при помощи команды CUTMOD положение резца после вращения инструмента.



S: Центр резца
P: Исходная точка резца
SL: Положение резца

Программный код	Комментарий
N10 $TC_DP1[1,1]=500	
N20 $TC_DP2[1,1]=3	; Положение резцов
N30 $TC_DP3[1,1]=12	
N40 $TC_DP4[1,1]=1	
N50 $TC_DP6[1,1]=6	
N60 $TC_DP10[1,1]=110	; Угол державки
N70 $TC_DP11[1,1]=3	; Направление резания
N80 $TC_DP24[1,1]=25	; Задний угол

Программный код	Комментарий
N90 $TC_CARR7[2]=0 $TC_CARR8[2]=1 $TC_CARR9[2]=0	; Ось B
N100 $TC_CARR10[2]=0 $TC_CARR11[2]=0 $TC_CARR12[2]=1	; Ось C
N110 $TC_CARR13[2]=0	
N120 $TC_CARR14[2]=0	
N130 $TC_CARR21[2]=X	
N140 $TC_CARR22[2]=X	
N150 $TC_CARR23[2]="M"	
N160 TCOABS CUTMOD=0	
N170 G18 T1 D1 TCARR=2	; X Y Z
N180 X0 Y0 Z0 F10000	; 12.000 0.000 1.000
N190 $TC_CARR13[2]=30	
N200 TCARR=2	
N210 X0 Y0 Z0	; 10.892 0.000 -5.134
N220 G42 Z-10	; 8.696 0.000 -17.330
N230 Z-20	; 8.696 0.000 -21.330
N240 X10	; 12.696 0.000 -21.330
N250 G40 X20 Z0	; 30.892 0.000 -5.134
N260 CUTMOD=2 X0 Y0 Z0	; 8.696 0.000 -7.330
N270 G42 Z-10	; 8.696 0.000 -17.330
N280 Z-20	; 8.696 0.000 -21.330
N290 X10	; 12.696 0.000 -21.330
N300 G40 X20 Z0	; 28.696 0.000 -7.330
N310 M30	

Числовые значения в комментариях указывают соответствующие позиции конца кадра в координатах станка (MCS) в последовательности X → Y → Z.

Пояснения

В кадре N180 сначала выбирается инструмент при CUTMOD=0 и не повернутом ориентируемом инструментальном суппорте. Т.к. все векторы смещения ориентируемого инструментального суппорта 0, выполняется подвод к позиции, соответствующей указанным в $TC_DP3[1,1] и $TC_DP4[1,1] длинами инструмента.

В кадре N200 ориентируемый инструментальный суппорт активируется с вращением в 30° вокруг оси B. Т.к. положение резцов из-за CUTMOD=0 не изменяется, определяющей как и прежде является старая опорная точка резцов. Поэтому в кадре N210 выполняется подвод к позиции, содержащей старую опорную точку резцов в нулевой точке (т.е. вектор (1, 12) поворачивается в плоскости Z/X на 30°).

В кадре N260 действует в отличие от кадра N200 CUTMOD=2. Из-за вращения ориентируемого инструментального суппорта получается измененное положение резца 8. Следствием этого являются и отклонения в позициях осей.

В кадрах N220 или N270 активируется коррекция на радиус инструмента (КРИ). Различное положение резцов в обеих отрезках программы не влияет на конечные позиции кадров, в которых активна КРИ, поэтому соответствующие позиции идентичны. Только в кадрах сброса N260 или N300 различные положения резцов снова сказываются.

4.13.13 Работа с инструментальным окружением

Обзор функций

- Сохранение инструментального окружения (TOOLENV) (Страница 838)
- Удаление инструментального окружения (DELTOOLENV) (Страница 841)
- Чтение номеров T, D и DL (GETTENV) (Страница 842)
- Считывание компонентов длины инструмента или компонентов длины (GETTCOR) (Страница 844)
- Изменение компонентов инструмента (SETTCOR) (Страница 850)

Обзор системных переменных

- Считывание данных сохраняемого инструментального окружения ($P_TOOLENVN, ($P_TOOLENV) (Страница 843)

4.13.13.1 Сохранение инструментального окружения (TOOLENV)

Функция TOOLENV служит для сохранения всех текущих состояний, имеющих значение для нормирования находящихся в памяти данных инструмента.

По отдельности это следующие данные:

- Активная G-команда группы:
  - 6 (G17, G18, G19)
  - 56 (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)
- Активная поперечная ось
- Машинные данные:
  - MD18112 $MN_MM_KIND_OF_SUMCORR (свойства суммарных коррекций в области TO)
  - MD20360 $MC_TOOL_PARAMETER_DEF_MASK (определение параметров инструмента)

- Установочные данные:
  - SD42900 $SC_MIRROR_TOOL_LENGTH (смена знака длины инструмента при отражении)
  - SD42910 $SC_MIRROR_TOOL_WEAR (смена знака износа инструмента при отражении)
  - SD42920 $SC_WEAR_SIGN_CUTPOS (знак износа у инструментов с системами резцов)
  - SD42930 $SC_WEAR_SIGN (знак износа)
  - SD42935 $SC_WEAR_TRANSFORM (трансформации для компонентов инструмента)
  - SD42940 $SC_LENGTH_CONST (смена компонентов длин инструмента при смене плоскости)
  - SD42942 $SC_TOOL_LENGTH_CONST_T (смена компонентов длин инструмента для токарных инструментов при смене плоскости)
  - SD42950 $SC_TOOL_LENGTH_TYPE (соотнесение компонентов длин инструмента независимо от типа инструмента)
  - SD42954 $SC_TOOL_ORI_CONST_M (смена компонентов ориентации инструмента для фрезерных инструментов при смене плоскости)
  - SD42956 $SC_TOOL_ORI_CONST_T (смена компонентов ориентации инструмента для токарных инструментов при смене плоскости)
- Компонент ориентации текущего общего фрейма (вращения и отражения, не смещения нулевой точки или отражения)
- Компонент ориентации и результирующая длина активного ориентируемого инструментального суппорта
- Компонент ориентации и результирующая длина активной трансформации

Кроме названных данных, описывающих окружение инструмента, сохраняются и номера T, номера D и номера DL активного инструмента, чтобы после можно было обратиться к этому инструменту в том же окружении, что и при вызове TOOLENV без повторного обозначения инструмента.

Синтаксис

<состояние> = TOOLENV(<имя>)

Объяснение

TOOLENV(...):	Предопределенная функция сохранения инструментального окружения	
	Автономно в кадре:	да

<состояние>:	Возвращаемое значение функции. Отрицательные значения показывают состояния ошибки.		
	Тип данных:		INT
	Значение:	0	Функция OK
		-1	не зарезервировано места в памяти для инструментального окружения: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Т. е. функциональность "Инструментальные окружения" отсутствует.
		-2	более нет свободного места для инструментальных окружений в памяти.
		-3	нулевая строка как имя инструментального окружения не допускается.
		-4	Параметр (<имя>) не указан.
Параметры			
1 <имя>:	Имя, под которым необходимо сохранять текущий блок данных. Если блок данных с тем же именем уже имеется, то он заменяется. В этом случае состояние 0.		
	Тип данных:		STRING

Дополнительная информация

Базовый размер / размер адаптера — коррекция на длину инструмента

Входит ли длина адаптера или базовый размер (спец. для резца параметры $TC_DP21, $TC_DP22, $TC_DP23) в вычисление длины инструмента, определяется при активном управлении магазином инструментов (доступно только с опцией "Магазин инструментов"!) по значению в машинных данных:

MD18104 $MN_MM_NUM_TOOL_ADAPTER (адаптер инструмента в области ТО)

Поскольку изменение этих машинных данных может начать действовать только запуске СЧПУ, оно в инструментальном окружении не сохраняется.

Результирующая длина ориентируемых инструментальных суппортов и трансформаций

Примечание

Как при ориентируемых инструментальных суппортах, так и при трансформациях, существуют системные переменные или машинные данные, которые действуют как дополнительные компоненты длин инструмента, и которые полностью или частично подвержены этим исполняемым вращениям. Дополнительные компоненты длин инструмента, получаемые вследствие этого, должны также быть сохранены при вызове TOOLENV, т. к. они являются частью окружения, в котором используется инструмент.

Трансформация адаптера

Трансформация адаптера это свойство адаптера инструмента и тем самым всего инструмента. Поэтому она не является составной частью инструментального окружения, которая может быть применена к другому инструменту.

Посредством сохранения всех необходимых для определения общей длины инструмента данных позднее можно вычислить эффективную длину инструмента, даже если на этот момент он более не активен или если условия окружения (например, G-команды или установочные данные) изменились. Также можно вычислить эффективную длину другого инструмента с допущением, что он использовался в тех же условиях, что и инструмент, для которого было сохранено состояние.

Макс. число наборов данных для инструментальных окружений

Машинный параметр MD18116 $MN_MM_NUM_TOOL_ENV определяет, какое максимальное количество блоков данных можно сохранить для описания инструментальных окружений. Данные находятся в диапазоне TOA. Они сохраняются и после отключения СЧПУ.

Сохранение данных невозможно. Это означает, что эти данные невозможно передать на другую систему управления.

4.13.13.2 Удаление инструментального окружения (DELTOOLENV)

С помощью функции DELTOOLENV блоки данных описания инструментальных окружений можно удалить. Удаление означает, что обращение к сохраненному под определенным именем блоку данных более невозможно (попытка обращения приводит к аварийному сообщению).

Примечание

Блоки данных можно удалять только функцией DELTOOLENV, через загрузку INITIAL.INI или через холодный пуск (запуск ЧПУ со стандартными машинными данными). Других автоматических процессов удаления нет.

Синтаксис

```
<состояние> = DELTOOLENV(<имя>)
<состояние> = DELTOOLENV()
```

Объяснение

`DELTOOLENV(...)`:	Предопределенная функция удаления инструментального окружения	
	Автономно в кадре:	да

<состояние>:	Возвращаемое значение функции. Отрицательные значения показывают состояния ошибки.		
	Тип данных:		INT
	Значение:	0	Функция OK
		-1	не зарезервировано места в памяти для инструментального окружения: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Т. е. функциональность "Инструментальные окружения" отсутствует.
		-2	инструментальное окружение с указанными именем не существует.
Параметры			
1 <имя>:	Имя удаляемого блока данных		
	Тип данных:		STRING
DELTOOLENV():	DELTOOLENV() без указания имени удаляет все блоки данных описания инструментального окружения		

4.13.13.3 Чтение номеров T, D и DL (GETTENV)

Функция GETTENV служит для чтения сохраненных в инструментальном окружении номеров T, D и DL.

Синтаксис

```
<состояние> = GETTENV(<Name>, <TDDL>)
```

Объяснение

GETTENV(...):	Предопределенная функция чтения номеров T, D, DL в блоке данных описания инструментального окружения		
	Автономно в кадре:		да
<состояние>:	Возвращаемое значение функции. Отрицательные значения показывают состояния ошибки.		
	Тип данных:		INT
	Значение:	0	Функция OK
		-1	не зарезервировано места в памяти для инструментального окружения: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Т. е. функциональность "Инструментальные окружения" отсутствует.
		-2	инструментальное окружение с указанными именем не существует.
Параметры			
1 <имя>:	Имя блока данных, из которого необходимо считывать номера T, D, DL.		
	Тип данных:		STRING

2	<TDDL>:	В поле этого параметра результата содержатся номера T, D, DL инструмента, инструментальное окружение которого сохранено в указанном блоке данных: • <TDDL> [0]: номер T • <TDDL> [1]: номер D • <TDDL> [2]: номер DL	
		Тип данных:	INT[3]
GETTENV(,<TDDL>), GETTENV("",<TDDL>):		Допускается пропустить при вызове функции GETTENV первый параметр или передать в качестве первого параметра нулевую строку. В обоих этих специальных случаях в <TDDL> возвращаются номера T, D, DL **активного** инструмента.	

4.13.13.4 Считывание данных сохраняемого инструментального окружения ($P_TOOLENVN, ($P_TOOLENV)

Данные сохраняемого инструментального окружения считываются по следующим системным переменным:

$P_TOOLENVN:	Выводит количество определенных с помощью TOOLENV (и еще не удаленных) блоков данных для описания инструментальных окружений.			
	Синтаксис:	<n> = $P_TOOLENVN		
	Объяснение:	<n>:	Количество определенных блоков данных	
			Тип данных:	INT
			Диапазон значений:	0 ... MD18116 $MN_MM_NUM_TOOL_ENV
	Обращение к этой системной переменной возможен и тогда, когда инструментальные окружения невозможны (MD18116 = 0). В этом случае возвращаемое значение — "0".			
$P_TOOLENV:	Выводит имя i-го блока данных описания инструментального окружения.			
	Синтаксис:	<Name> = $P_TOOLENV[<i>]		
	Объяснение:	<имя>:	Имя блока данных с номером <i>.	
			Тип данных:	STRING
		<i>:	Номер блока данных	
			Тип данных:	INT
			Диапазон значений:	1 ... $P_TOOLENVN
	Согласование номеров с блоками данных не является фиксированным, а может изменяться через удаление и повторное создание блоков данных. Нумерация блоков данных является внутренней. Если <i> ссылается на не определенный блок данных, возвращается нулевая строка. Если индекс <i> недействителен, т. е. <i> меньше 1 или больше значения максимального количества блоков данных для инструментальных окружений (MD18116 $MN_MM_NUM_TOOLENV), выводится следующее аварийное сообщение: Аварийное сообщение "17020 (неразрешенный индекс массива1)"			

4.13.13.5 Считывание компонентов длины инструмента или компонентов длины (GETTCOR)

Функция GETTCOR служит для считывания длин инструмента или компонентов длин инструмента.

При этом посредством параметрирования можно указать, какие компоненты необходимо учесть, и при каких условиях использования необходимо рассматривать инструмент.

Синтаксис

```
<состояние> = GETTCOR(<Len>[, <Comp>, <Stat>, <T>, <D>, <DL>])
```

Объяснение

GETTCOR(...):	Предопределенная функция чтения длин инструмента или компонентов длин инструмента		
	Автономно в кадре:		да
<состояние>:	Возвращаемое значение функции. Отрицательные значения показывают состояния ошибки.		
	Тип данных:		INT
	Значение:	0	Функция OK
		-1	не зарезервировано места в памяти для инструментального окружения: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Т. е. функциональность "Инструментальные окружения" отсутствует.
		-2	Инструментальное окружение с указанным в <Stat> именем не существует.
		-3	Недействительная строка в параметре <Comp>. Причиной этой ошибки могут быть недействительные или запрограммированные дважды символы.
		-4	недействительный номер T
		-5	недействительный номер D
		-6	недействительный номер DL
		-7	Попытка обращения к отсутствующему модулю памяти
		-8	Попытка обращения к отсутствующей опции (например, программируемая ориентация инструмента, управление инструментом)
		-9	Строка <Comp> содержит двоеточие (идентификатор спецификаций системы координат), но за ним не следует действительный символ обозначения требуемой системы координат.
Параметры			

1	<Len>:	Результирующий вектор	
		Тип данных:	REAL[11]
		Компоненты вектора расположены в следующей последовательности: • <Len> [0]: Тип инструмента • <Len> [1]: Положение резцов • <Len> [2]: Абсцисса • <Len> [3]: Ордината • <Len> [4]: Аппликата • <Len> [5]: Радиус инструмента Исходной системой координат для компонентов длин является определенная в <Comp> и <Stat> система координат. Если в <Comp> система координат не определена, то длины инструмента отображаются в системе координат станка. Соотнесение абсциссы, ординаты и аппликаты с гео-осями зависит от активной плоскости в используемом инструментальном окружении. Т. е. при G17 абсцисса параллельна X, при G18 — Z и т. д. Компоненты с <Len>[6] по <Len>[10] содержат дополнительные параметры, которые могут быть указаны для описания геометрии инструмента (например, с $TC_DP7 по $TC_DP11 для геометрии или соответствующие компоненты для износа или суммарной и отладочной коррекции). Эти 5 дополнительных элементов и радиус инструмента определены только для компонентов E, G, S и W. Их нормирование не зависит от <Stat>. Соответствующие значения в компонентах с <Len>[6] по <Len>[10] могут отличаться от нуля, только если в вычислении длин инструмента участвует, по меньшей мере, один из четырех названных компонентов. Прочие компоненты на результат не влияют. Данные размеров относятся к основной системе СЧПУ (дюймовая или метрическая).	

2	<Comp>:	Компоненты длин инструмента (опционально)			
		Тип данных:	STRING		
		Эта цепочка символов состоит из двух частей, разделенных двоеточием. Общая форма: "<SubStr_1> [: <SubStr_2]"			
		<SubStr_1>:	**Первая часть строки** обозначает компоненты длин инструмента, которые необходимо учитывать при вычислении длин инструмента. Последовательность символов в части строки и их написание (прописное или строчное) — произвольные. Между символами можно вставлять любое число пробелов или символов табулятора (white spaces). **Указание:** **Двойное** программирование символов в части строки не допускается.		
			Символ:	-	знак минус (допустим только как первый символ!) Вычисляется вся длина инструмента, за исключением компонентов, указанных в следующей строке.
				C	размер адаптера или базовый размер (один из двух существующих как альтернатива компонентов, который активен для используемого инструмента)
				E	отладочные коррекции
				G	геометрия
				K	Кинематическая трансформация (обрабатывается только при базовой 3-, 4- и 5-осевой трансформации)
				S	Суммарные коррекции
				T	Ориентируемый инструментальный суппорт
				W	Износ
			Если первая часть строки (за исключением white spaces) пустая, то это означает, что всю длину инструмента необходимо вычислять с учетом всех компонентов. То же действительно, и если параметр <Comp> не указывается.		
		<Substr_2>:	Опциональная **вторая часть строки** обозначает систему координат, в которой должна выводиться длина инструмента. Вторая часть строки состоит только из одного релевантного символа.		
			Символ:	A	Настраиваемая система координат (ACS)
				B	Базовая кинематическая система (BKS)
				K	Система координат инструмента кинематической трансформации (KCS)
				M	Система координат станка (MCS)
				T	Система координат инструмента (TCS)
				W	Система координат детали (WCS)

			Если система координат не указывается, то нормирование выполняется в системе координат станка (MCS). Учитываемые при необходимости вращения устанавливаются через определенное в <Stat> инструментальное окружение.
3	<_Stat>:	Имя блока данных для описания инструментального окружения (опционально)	
		Тип данных:	STRING
		Если значение этого параметра нулевая строка ("") или оно не указывается, то используется текущее состояние. Если инструмент не задан, используется текущий инструмент.	
4	<T>:	Внутренний номер T инструмента (опционально)	
		Тип данных:	INT
		Если этот параметр не указан или если его значение 0, то используется сохраненный в <Stat> инструмент. Если значение этого параметра "-1", то используется номер T активного инструмента. Допускается явное указание номера активного инструмента. **Указание:** Если <Stat> не указана, то в качестве инструментального окружения используется текущее состояние. Поскольку с <T> = 0 выполняется ссылка на сохраненный в инструментальном окружении номер T, в нем используется активный инструмент, т. е. данные <T> = 0 и <T> = -1 в этом особом случае равнозначны.	
5	<D>:	Номер резца инструмента (опционально)	
		Тип данных:	INT
		Если этот параметр не указан или его значение 0, то используемый номер D зависит от источника номера T. Если используется номер T из инструментального окружения, то считывается и номер D инструментального окружения, в ином случае таковой текущего активного инструмента.	
6	<DL>:	Номер зависящей от места коррекции (опционально)	
		Тип данных:	INT
		Если этот параметр не указан, то используемый номер DL зависит от источника номера T. Если используется номер T из инструментального окружения, то считывается и номер D инструментального окружения, в ином случае таковой текущего активного инструмента.	

Примеры

`GETTCOR(_LEN)`	Вычисляет длину текущего активного инструмента в системе координат станка с учетом всех компонентов.
`GETTCOR(_LEN,"CGW:W")`	Вычисляется длина инструмента, состоящая из размера адаптера или базового размера, геометрия и износ для активного инструмента. Прочие компоненты, например, ориентируемый инструментальный суппорт или кинематические трансформации, не учитываются. Вывод выполняется в системе координат детали.

`GETTCOR (_LEN,"-K:B")`	Вычисляется общая длина активного инструмента без учета компонентов длин возможно активной кинематической трансформации. Вывод в базовой кинематической системе.
`GETTCOR (_LEN,":M","Testenv1",,3)`	Вычисляется общая длина инструмента для сохраненного в инструментальном окружении под именем "Testenv1" инструмента в системе координат станка. Но вычисление выполняется независимо от сохраненного номера резца для номера резца D3.

Дополнительная информация

Трансформация адаптера / ориентируемые инструментальные суппорты / кинематическая трансформация

Вращения и смены компонентов, которые в некоторых случаях выполняются трансформацией адаптера, ориентируемым инструментальным суппортом и кинематической трансформацией, являются составной частью инструментального окружения. Поэтому они выполняются всегда, и тогда, когда соответствующий компонент длины не должен учитываться. Если это нежелательно, то необходимо определить инструментальные окружения, в которых соответствующие трансформации не активны. Во многих случаях (а именно всегда, когда на станке не используются трансформации или ориентируемые инструментальные суппорты), сохраненные блоки данных для инструментального окружения выполняют эти условия автоматически, поэтому отдельного рассмотрения их пользователем не требуется.

Токарные и шлифовальные инструменты: Вычисление длины инструмента в зависимости от MD20360 $MC_TOOL_PARAMETER_DEF_MASK

В машинных данных определяется, как у токарного и шлифовального инструмента следует нормировать износ или длину инструмента в, возможно, имеющейся оси диаметра.

MD20360 $MC_TOOL_PARAMETER_DEF_MASK (определение параметров инструмента)

Бит	Значение	
0	В случае токарного и шлифовального инструмента **параметр износа** поперечной оси учитывается при вычислении как значение диаметра:	
	= 0 (по умолчанию)	нет
	=1	да
1	В случае токарного и шлифовального инструмента **компонент длины инструмента** поперечной оси учитывается при вычислении как значение диаметра:	
	= 0 (по умолчанию)	нет
	=1	да

Если соответствующие биты установлены, то соответствующая строка нормируется с коэффициентом 0.5. Такое нормирование отражается и в возвращенной GETTCOR длине инструмента.

Пример:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK = 3

MD20100 $MC_DIAMETER_AX_DEF (гео-ось с функцией поперечной оси) = "X"

X — это ось диаметра (стандартная конфигурация токарных станков)

Программный код	Комментарий
N30 $TC_DP1[1,1]=500	
N40 $TC_DP2[1,1]=2	
N50 $TC_DP3[1,1]=3.0	; геометрия L1
N60 $TC_DP4[1,1]=4.0	
N70 $TC_DP5[1,1]=5.0	
N80 $TC_DP12[1,1]=12.0	; износ L1
N90 $TC_DP13[1,1]=13.0	
N100 $TC_DP14[1,1]=14.0	
N110 T1 D1 G18	
N120 R1=GETTCOR(_LEN,"GW")	
N130 R3=_LEN[2]	; 17.0 (= 4.0 + 13.0)
N140 R4=_LEN[3]	; 7.5 (= 0.5 * 3.0 + 0.5 * 12.0)
N150 R5=_LEN[4]	; 19.0 (= 5.0 + 14.0)
N160 M30	

Компоненты длины кинематической трансформации и ориентируемого инструментального суппорта

Если при вычислении длин инструмента учитывается **ориентируемый инструментальный суппорт**, то следующие векторы включаются в вычисление длин инструмента:

Тип	Векторы
M	l1 и l2
T	l1, l2 и l3
P	Ориентируемый инструментальный суппорт не влияет на длину инструмента.

При базовой **5-осевой трансформации** при типах трансформации 24 и 56 следующие машинные данные включаются в вычисление длин инструмента:

Тип трансформации	Машинные данные
24	MD24550/24650 $MC_TRAFO5_BASE_TOOL_1/2 MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 MD24558/24658 $MC_TRAFO5_PART_OFFSET_1/2
56	MD24550/24650 $MC_TRAFO5_BASE_TOOL_1/2 MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2

Тип трансформации 56 (подвижный инструмент и подвижная деталь) соответствует типу M для ориентируемого инструментального суппорта.

При этой 5-осевой трансформации в прежних версиях ПО вектор MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 (вектор кинематического смещения 1-ой/2-ой 5-осевой трансформации в канале) соответствует сумме обоих векторов l_1 и l_3 у ориентируемого инструментального суппорта типа M:

В обоих случаях для трансформации важна только сумма. Тип структуры из двух отдельных компонентов не имеет значения. Но при вычислении длины инструмента имеет значение, какой компонент согласован с инструментом, а какой – с инструментальным столом. Поэтому были введены машинные данные MD24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 (вектор кинематического смещения в столе). Он соответствует вектору l3. Теперь машинные данные MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 больше не соответствуют сумме l1 и l3, а соответствуют только вектору l1. Если машинные данные MD24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 равны нулю, то поведение идентично прежнему.

Совместимость

Функция GETTCOR в комбинации с функциями TOOLENV и SETTCOR служит, среди прочего, для замены частей функциональности, прежде реализованных отдельно в измерительных циклах.

В измерительных циклах обрабатывалась только часть параметров, которые в конечном итоге определяют эффективную длину инструмента. Названные функции можно спараметрировать так, что появляется возможность воспроизвести поведение измерительных циклов в отношении вычисления длин инструмента.

4.13.13.6 Изменение компонентов инструмента (SETTCOR)

Функция SETTCOR изменяет компоненты инструмента с учетом всех граничных условий, входящих в нормирование отдельных компонентов.

Примечание

Терминология: Если далее в связи с длиной инструмента речь идет о компонентах инструмента, то подразумеваются компоненты, рассматриваемые в векторной форме, из которых состоит полная длина инструмента (например, геометрия или износ). Такой компонент состоит из трех отдельных значений (L1, L2, L3), обозначаемых далее "значения координат".

Компонент инструмента "геометрия" состоит, например, из трех значений координат от $TC_DP3 до $TC_DP5.

Синтаксис

```
<состояние> = SETTCOR(<CorVal>, <Comp>, [<CorComp>, <CorMode>,
<GeoAx>, <Stat>, <T>, <D>, <DL>])
```

Объяснение

`SETTCOR(...)`:	Предопределенная функция изменения компонентов инструмента	
	Автономно в кадре:	да

<состояние>:	Возвращаемое значение функции. Отрицательные значения показывают состояния ошибки.		
	Тип данных:		INT
	Значение:	0	Функция OK
		-1	Не зарезервировано места в памяти для инструментального окружения: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Т. е. функциональность "Инструментальные окружения" отсутствует.
		-2	Инструментальное окружение с указанным в <Stat> именем не существует.
		-3	Недействительная строка в параметре <Comp>. Причиной этой ошибки могут быть недействительные или запрограммированные дважды символы.
		-4	Недействительный номер T
		-5	Недействительный номер D
		-6	Недействительный номер DL
		-7	Попытка обращения к отсутствующему модулю памяти
		-8	Попытка обращения к отсутствующей опции (например, программируемая ориентация инструмента, управление инструментом)
		-9	Недопустимое числовое значение параметра <CorComp**>**.
		-10	Недопустимое числовое значение параметра <CorMode>.
		-11	Содержание параметров <Comp> и <CorComp> противоречиво.
		-12	Содержание параметров <Comp> и <CorMode> противоречиво.
		-13	Содержание параметра <GeoAx> не обозначает гео-ось.
		-14	Попытка записи на отсутствующую отладочную коррекцию
Параметры			
1	`<CorVal>`:	Вектор коррекции В определенной в <Stat> системе координат детали (WCS) действует следующее соотнесение: • <CorVal> [0]: Абсцисса • <CorVal> [1]: Ордината • <CorVal> [2]: Аппликата Ели необходимо исправить только один компонент инструмента (например, невекторная коррекция, см. параметр <CorMode>), то значение коррекции всегда находится в <CorVal>[0] независимо от того, для какой оси но действует. При этом содержание двух остальных компонентов не обрабатывается. Если <CorVal> или компонент <CorVal> относится к поперечной оси, то данные нормируются как **размер радиуса**. Это означает, что инструмент удлиняется, например, на заданный размер, и это ведет, соответственно, к двойному увеличению диаметра детали. Данные размеров относятся к **основной системе** СЧПУ (дюймовая или метрическая).		
		Тип данных:	REAL[3]

2	<Comp>:	Компонент(ы) инструмента			
		Тип данных:	STRING		
		Эта цепочка символов состоит из двух частей, разделенных двоеточием. Общая форма: "<SubStr_1> [: <SubStr_2]"			
		<SubStr_1>:	**Первая часть строки** всегда должна быть в наличии и может состоять из одного или из двух символов. При этом первый или единственный символ обозначает 1-й компонент (Val_1), а второй символ — 2-й компонент (Val_2), обрабатываемые в соответствии со следующими параметрами <CorComp> и <CorMode>.		
			Символ:	C	размер адаптера или базовый размер (один из двух существующих как альтернатива компонентов, который активен для используемого инструмента)
				E	отладочные коррекции
				G	геометрия
				S	суммарные коррекции
				W	износ
		<Substr_2>:	**Вторая часть строки** опциональна. Она может состоять на выбор из (единственной) буквы "W" или "T".		
			Символ:	W	Если вторая часть строки пустая или содержит букву "W", то значения коррекции вычисляются так, как если бы их замеряли в **системе координат детали** (WCS).
				T	Если вторая часть строки содержит букву "T", то то значения коррекции вычисляются так, как если бы их замеряли в **системе координат инструмента** (**T**ool Coordinate System, TCS).
		Написание символов в строке (прописное или строчное) является произвольным. Можно вставлять любое число пробелов или символов табулятора (white spaces).			

3	<CorComp>:	Задает компонент(ы) блока данных инструмента, которые необходимо описать (опционально)		
		Тип данных:		INT
		Значение:	0	Значение коррекции <CorVal>[0] относится к гео-оси, определенной в параметре <GeoAx>, в системе координат детали или в системе координат инструмента (см. описание параметра <Comp>). Это означает, что значение коррекции должно быть так учтено в обозначенных компонентах инструмента, что с учетом всех параметров, способных повлиять на вычисление длин инструмента, результатом оказывается изменение всей длины инструмента в указанном осевом направлении на заданное значение. Такое изменение можно получить коррекцией указанных в <Comp> компонентов и указанного в <CorMode> правила вычисления символов (см. следующий параметр). Поэтому получившаяся коррекция может сказываться на всех трех компонентах оси.
			1	Как "0", но векторн. Содержание вектора <CorVal> относится к абсциссе, ординате и аппликате в системе координат детали или в системе координат инструмента (см. описание параметра <Comp>). Следующий параметр <GeoAx> не обрабатывается.
			2	Векторная коррекция, т.е., L1, L2, L3 можно изменять одновременно. Однако в отличие от вариантов "0" и "1" содержащиеся в <CorVal> значения коррекции относятся к координатам компонента Val_1 (см. следующий параметр <CorMode>) инструмента. Возможное наклонное положение инструмента относительно системы координат детали на коррекцию не влияет.
			3 – 5	Коррекция значений длины инструмента с L1 по L3 (с $TC_DP3 по $TC_DP5) или соответствующих значений для коррекции на износ, отладочной коррекции или суммарной коррекции. Значение коррекции содержится в <CorVal>[0]. Оно измерено в координатах компонента Val_1 (см. следующий параметр <CorMode>). Возможное наклонное положение инструмента относительно системы координат детали на коррекцию не влияет.
			6	Коррекция значения радиуса инструмента ($TC_DP6) или соответствующих значений для коррекции на износ, отладочной коррекции или суммарной коррекции. Учитываются биты 10 и 11 (нормирование указания диаметра или износа диаметра на выбор как указание радиуса или диаметра) в машинных данных MD20360 $MC_TOOL_PARAMETER_DEF_MASK.
			7 – 11	Коррекция с $TC_DP7 по $TC_DP11 или соответствующих значений для коррекции на износ, отладочной коррекции или суммарной коррекции. Эти параметры обрабатываются как радиус инструмента.
		Если этот параметр не указан, его значение — "0".		

<table>
<tr><td>4</td><td><CorMode>:</td><td colspan="3">Задает тип выполняемой операции записи (опционально)</td></tr>
<tr><td></td><td></td><td colspan="2">Тип данных:</td><td>INT</td></tr>
<tr><td></td><td></td><td rowspan="4">Значе-ние:</td><td>0</td><td>Val_{1neu} = <CorVal></td></tr>
<tr><td></td><td></td><td>1</td><td>Val_{1neu} = Val_{1alt} + <CorVal></td></tr>
<tr><td></td><td></td><td>2</td><td>Val_{1neu} = <CorVal>
Val_{2neu} = 0</td></tr>
<tr><td></td><td></td><td>3</td><td>Val_{1neu} = Val_{1alt} + Val_{2alt} + <CorVal>
Val_{2neu} = 0</td></tr>
<tr><td></td><td></td><td colspan="3">Форму записи Val_{1alt} + Val_{2alt} следует понимать как символическую. Если оба компонента (на основании состояния <_Stat>) нормированы различно, т. е. если между обоими компонентами действует вращение, то перед сложением Val_{2alt} преобразуется таким образом, что получающаяся длина инструмента после удаления Val_{2neu} и перед сложением <CorVal> остается неизменной.
<CorVal> всегда относится к Val_1. <CorVal> — это значение, измеряемое в зависимости от второй части параметра <Comp> в системе координат детали (WCS) или в системе координат инструмента (TCS). Поэтому по отношению к компоненту инструмента, для которого его необходимо учесть, оно в соответствующих случаях уже преобразовано. Поэтому его невозможно вычислить с сохраненным значением: перед сложением необходимо предпринять обратное преобразование в значение Val_1 или Val_2. Это может привести к тому, что коррекция сказывается на другой оси, а не оси, определенной посредством <CorComp>, или на нескольких осях.
Если <CorComp> = 0, т. е., если <CorVal> содержит не вектор, а лишь отдельное значение, описанные операции выполняются в координатах замера <CorVal> (WCS / TCS). В частности, это действительно для установления на нуль Val_{2neu} в вариантах 2 и 3. Затем этот результат преобразуется обратно в координаты инструмента. Это может привести к тому, что ни одно из установленных на нуль значений координат (L1, L2, L3) не равно нулю, или к тому, что значения координат, которые ранее были равны нулю, теперь отличны от нуля. Однако если соответствующие операции выполняются последовательно для всех трех гео-осей, есть гарантия того, что все три значения координат удаляемого компонента будут равны нулю. Если инструмент не повернут по отношению к системе координат детали или повернут так, что все компоненты инструмента остаются параллельными к осям координат (перестановки осей), есть гарантия того, что изменяется соответственно только одна координата инструмента.
Последовательное выполнение одинаковой операции (<CorMode>) при <CorComp> = 0 для всех трех осей координат в произвольной последовательности идентично однократному выполнению одинаковой операции при <CorComp>=1.
Для значений параметров "0" и "1" параметр <Comp> должен содержать один символ, для значений параметров "2" и "3" — два символа.
Пример:
<Comp> содержит строку "ES", <CorMode> содержит значение "2"
⇒ отладочная коррекция$_{нов}$ = <CorVal>, суммарная коррекция$_{нов}$ = 0
Если параметр <CorMode> не указан, его значение — "0".</td></tr>
</table>

5	<GeoAx>:	Указывает индекс гео-оси, в который измерялось значение коррекции <CorVal>[0] (опционально)	
		Тип данных:	INT
		Диапазон значений:	0 ... 2
		Индексы от 0 до 2 относятся к абсциссе, ординате и аппликате в действующей плоскости (G17/G18/G19) текущего инструментального окружения. Содержание этого параметра обрабатывается, только если параметр <CorComp> имеет значение "0".	
6	<Stat>:	Имя блока данных для описания инструментального окружения (опционально)	
		Тип данных:	STRING
		Если значение этого параметра нулевая строка ("") или оно не указывается, то используется текущее состояние. Если инструмент не задан, используется текущий инструмент.	
7	<T>:	Внутренний номер T инструмента (опционально)	
		Тип данных:	INT
		Если этот параметр не указан, или если его значение 0, то используется сохраненный в <Stat> инструмент. Если значение этого параметра "-1", то используется номер T активного инструмента. Допускается явное указание номера активного инструмента. **Указание:** Если <Stat> не указана, то в качестве инструментального окружения используется текущее состояние. Поскольку с <T> = 0 выполняется ссылка на сохраненный в инструментальном окружении номер T, в нем используется активный инструмент, т. е. данные <T> = 0 и <T> = -1 в этом особом случае равнозначны.	
8	<D>:	Номер резца инструмента (опционально)	
		Тип данных:	INT
		Если этот параметр не указан или его значение 0, то используемый номер D зависит от источника номера T. Если используется номер T из инструментального окружения, то считывается и номер D инструментального окружения, в ином случае таковой текущего активного инструмента.	
9	<TL>:	Номер зависящей от места коррекции (опционально)	
		Тип данных:	INT
		Если этот параметр не указан, то используемый номер DL зависит от источника номера T. Если используется номер T из инструментального окружения, то считывается и номер D инструментального окружения, в ином случае таковой текущего активного инструмента. Если T, D, DL задают инструмент без зависящих от местоположения коррекций, то в параметре <Comp> не должны задаваться суммарные или отладочные коррекции (код ошибки в <Status>).	

Примечание

Не все возможные сочетания трех параметров <Comp>, <CorComp> и <CorMode> имеют смысл. Например, правило вычисления 3 требует в <CorComp> указания двух знаков в <Comp>. Если указывается недопустимое сочетание параметров, в <Status> возвращается соответствующий код ошибки.

Примеры

Пример 1

Программный код	Комментарий
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; фрезерный инструмент
N30 $TC_DP3[1,1]=10.0	; геометрия L1
N40 $TC_DP12[1,1]=1.0	; износ L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"G",0,0,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> позиция MCS X0.000 Y0.000 **Z1.333**
N90 M30	

<CorComp> = "0", поэтому действующее в направлении Z значение координаты геометрического компонента заменяется на значение коррекции 0,333.

Таким образом, результирующая общая длина инструмента: L1 = 0,333 + 1,000 = 1,333

Пример 2

Программный код	Комментарий
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; фрезерный инструмент
N30 $TC_DP3[1,1]=10.0	; геометрия L1
N40 $TC_DP12[1,1]=1.0	; износ L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"W",0,1,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> позиция MCS X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> = "1", поэтому действующее в направлении Z значение коррекции 0,333 прибавляется к значению износа 1,0.

Таким образом, результирующая общая длина инструмента: L1 = 10,0 + 1,333 = 11,333

Пример 3

Программный код	Комментарий
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; фрезерный инструмент
N30 $TC_DP3[1,1]=10.0	; геометрия L1
N40 $TC_DP12[1,1]=1.0	; износ L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"GW",0,2,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> позиция MCS X0.000 Y0.000 Z0.333
N90 M30	

<CorComp> = 2, поэтому действующая в направлении Z коррекция вносится в геометрический компонент (старое значение заменяется), и значение износа удаляется.

Таким образом, результирующая общая длина инструмента: L1 = 0,333 + 0,0 = 0,333

Пример 4

Программный код	Комментарий
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; фрезерный инструмент
N30 $TC_DP3[1,1]=10.0	; геометрия L1
N40 $TC_DP12[1,1]=1.0	; износ L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"GW",0,3,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> позиция MCS X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> = 3, поэтому значение износа и значение коррекции прибавляется к геометрическому компоненту, и компонент износа удаляется.

Таким образом, результирующая общая длина инструмента: L1 = 11,333 + 0,0 = 11,333.

Пример 5

Программный код	Комментарий
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; фрезерный инструмент
N30 $TC_DP3[1,1]=10.0	; геометрия L1
N40 $TC_DP12[1,1]=1.0	; износ L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"GW",0,3,0)**	
N80 T1 D1 X0 Y0 Z0	; ==> позиция MCS X0.333 Y0.000 Z11.000
N90 M30	

<CorComp> = 3, как в предыдущем примере, но теперь коррекция действует на гео-ось с индексом "0" (ось X), с которой из-за G17 у фрезерного инструмента соотнесен компонент инструмента L3. Поэтому вызов SETTCOR не влияет на параметры инструмента $TC_DP3 и $TC_DP12. Вместо этого значение коррекции вносится в $TC_DP5.

Пример 6

Программный код	Комментарий
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; токарный инструмент
N30 $TC_DP3[1,1]=10.0	; геометрия L1
N40 $TC_DP4[1,1]=15.0	; геометрия L2
N50 $TC_DP12[1,1]=10.0	; износ L1
N60 $TC_DP13[1,1]=0.0	; износ L2
N70 _CORVAL[0]=5.0	

Программный код	Комментарий
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=**SETTCOR(_CORVAL,"GW",0,3,1)**	
N110 T1 D1 X0 Y0 Z0	; ==> позиция MCS X24.330 Y0.000 Z17.500
N120 M30	

Инструмент — это токарный инструмент. В кадре N80 активируется вращение фрейма, поэтому базовая кинематическая система (BCS) повернута по отношению к системе координат детали (WCS). Значение коррекции (N70) воздействует в WCS на гео-ось с индексом "1", т. е., поскольку G18 активна, на ось X. Поскольку <CorMode> = 3, износ инструмента в направлении оси X WCS после выполнения N100 должен стать равным нулю.

Поэтому содержание релевантных параметров инструмента в конце программы:

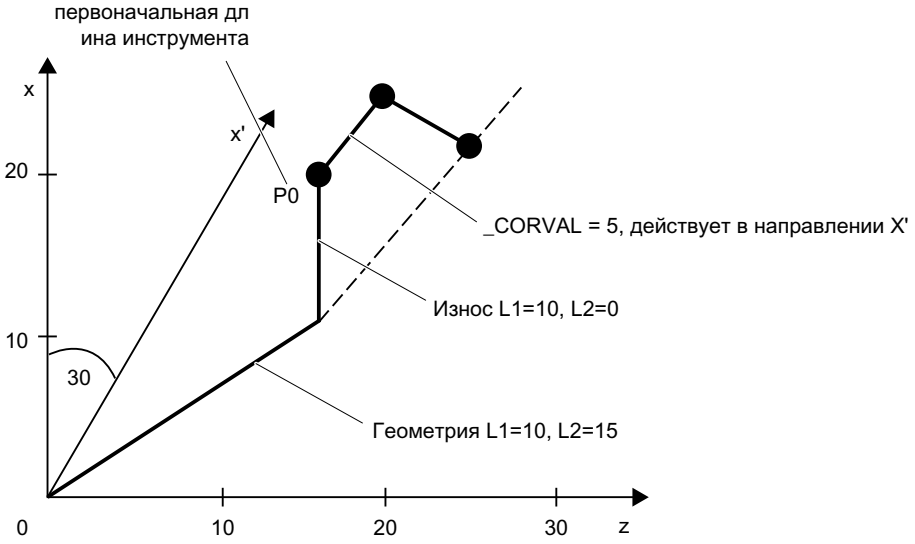
$TC_DP3[1,1] : 21.830 ; геометрия L1

$TC_DP4[1,1] : 21.830 ; геометрия L2

$TC_DP12[1,1] : 2.500 ; износ L1

$TC_DP13[1,1] : -4.330 ; износ L2

Геометрические соотношения представлены на следующем изображении. Весь износ, включая _CORVAL, проецируется на направление X' в WCS. Это дает точку P2. Координаты этой точки (измерены в координатах X-Y) вносятся в геометрический компонент инструмента. В износе остается дифф. вектор $P_2 – P_1$. Тем самым у износа больше нет компонента в направлении _CORVAL.



Если иллюстративная программа продолжается после N110 со следующими операторами, то оставшийся износ полностью передается в геометрию, поэтому коррекция теперь действует в оси Z' (параметр <GeoAx> = 0).

```
N120 _CORVAL[0]=0.0
N130 R1=SETTCOR(_CORVAL,"GW",0,3,0)
```

```
N140 T1 D1 X0 Y0 Z0 ; ==> позиция MCS X24.330 Y0.000 Z17.500
```

Поскольку новое значение коррекции равно 0, изменение общей длины инструмента и тем самым позиции подвода в N140 запрещено. Если бы _CORVAL в N120 была отлична от 0, то получилась бы новая общая длина инструмента, и тем самым и измененная позиция в N140, но компонент износа длины инструмента был бы в любом случае равен нулю, т. е. общая длина инструмента в результате в любом случае содержится в геометрическом компоненте инструмента.

Результат, идентичный двукратному вызову функции SETTCOR с параметром <CorComp> = 0, достигается и посредством однократного вызова с <CorComp> = 1 (векторная коррекция):

Программный код	Комментарий
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; токарный инструмент
N30 $TC_DP3[1,1]=10.0	; геометрия L1
N40 $TC_DP4[1,1]=15.0	; геометрия L2
N50 $TC_DP12[1,1]=10.0	; износ L1
N60 $TC_DP13[1,1]=0.0	; износ L2
N70 _CORVAL[0]=0.0	
N71 _CORVAL[1]=5.0	
N72 _CORVAL[2]=0.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=**SETTCOR(_CORVAL,"GW",1,3,1)**	
N110 T1 D1 X0 Y0 Z0	; ==> позиция MCS X24.330 Y0.000 Z17.500
N120 M30	

В этом случае все компоненты износа инструмента сразу же после первого вызова SETTCOR в N100 равны нулю.

Пример 7

Программный код	Комментарий
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; токарный инструмент
N30 $TC_DP3[1,1]=10.0	; геометрия L1
N40 $TC_DP4[1,1]=15.0	; геометрия L2
N50 $TC_DP12[1,1]=10.0	; износ L1
N60 $TC_DP13[1,1]=0.0	; износ L2
N70 _CORVAL[0]=5.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=**SETTCOR(_CORVAL,"GW",3,3)**	
N110 T1 D1 X0 Y0 Z0	; ==> позиция MCS X25.000 Y0.000 Z15.000
N120 M30	

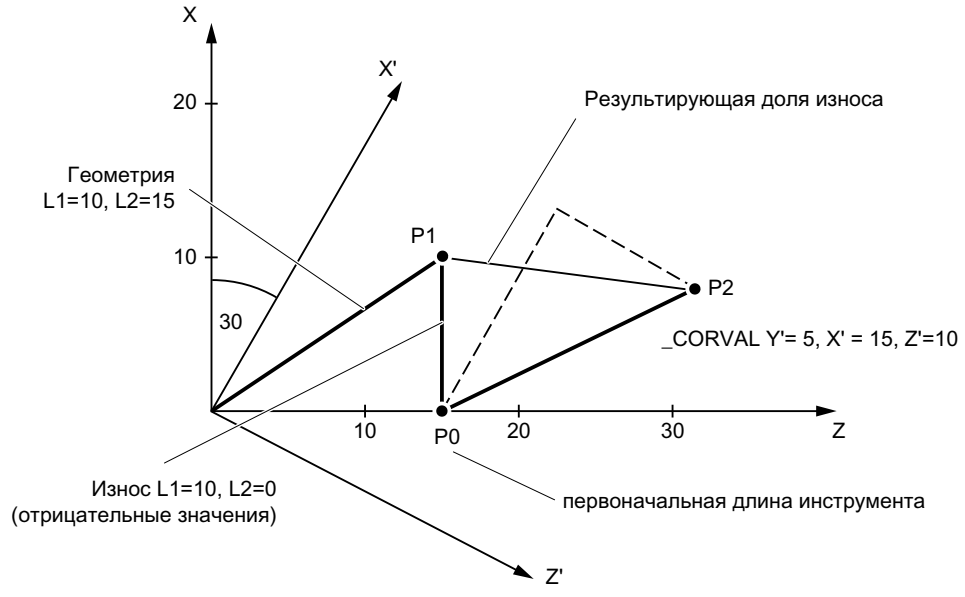
По сравнению с примером 6 здесь параметр <CorComp> = 3, поэтому можно не указывать параметр <GeoAx>. Теперь содержащееся в _CORVAL[0] значение действует

непосредственно на компонент инструмента L1, вращение в N80 не влияет на результат, компоненты износа в $TC_DP12 передаются вместе с _CORVAL[0] в геометрический компонент, поэтому из-за $TC_DP13 вся длина инструмента уже после первого вызова SETTCOR в N100 стоит в геометрическом компоненте инструмента.

Пример 8

Программный код	Комментарий
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; токарный инструмент
N30 $TC_DP3[1,1]=10.0	; геометрия L1
N40 $TC_DP4[1,1]=15.0	; геометрия L2
N50 $TC_DP5[1,1]=20.0	; геометрия L3
N60 $TC_DP12[1,1]=10.0	; износ L1
N70 $TC_DP13[1,1]=0.0	; износ L2
N80 $TC_DP14[1,1]=0.0	; износ L3
N90 $SC_WEAR_SIGN=TRUE	
N100 _CORVAL[0]=10.0	
N110 _CORVAL[1]=15.0	
N120 _CORVAL[2]=5.0	
N130 ROT Y-30	
N140 T1 D1 G18 G0	
N150 R1=**SETTCOR(_CORVAL,"W",1,1)**	
N160 T1 D1 X0 Y0 Z0	; ==> позиция MCS X7.990 Y25.000 Z31.160
N170 M30	

В N90 устанавливается SD42930 $SC_WEAR_SIGN, т. е. износ должен быть нормирован отрицательным знаком. Коррекция является векторной (<CorComp> = 1), и вектор коррекции необходимо прибавить к износу (<CorMode> = 1). Геометрические отношения в плоскости Z-X представлены на следующем изображении:



Из-за <CorMode> = 1 геометрический компонент инструмента остается неизменным. Определенный в WCS (вращение вокруг оси y) вектор коррекции должен быть передан в компонент износа так, чтобы общая длина инструмента на рис. 3 ссылалась на точку P_2. Поэтому результирующий компонент износа инструмента задан через расстояние между точками P_1 и P_2.
Но поскольку износ из-за установочных данных SD42930 $SC_WEAR_SIGN нормируется отрицательно, полученная таким образом коррекция должна быть внесена в память коррекций с отрицательным знаком. Поэтому содержание релевантных параметров инструмента в конце программы:

$TC_DP3[1,1] : 10.000 ; геометрия L1 (без изменений)

$TC_DP4[1,1] : 15.000 ; геометрия L2 (без изменений)

$TC_DP5[1,1] : 10.000 ; геометрия L3 (без изменений)

$TC_DP12[1,1] : 2.010 ; износ L1 (= 10 - 15 * cos(30) + 10 * sin(30))

$TC_DP13[1,1] : -16.160 ; износ L2 (= -15 * sin(30) - 10 * cos(30))

$TC_DP14[1,1] : -5.000 ; износ L3

По компоненту L3 в направлении Y можно определить действие установочных данных SD42930 $SC_WEAR_SIGN без дополнительного усложнения вращением фрейма.

Дополнительная информация

Токарный и шлифовальный инструмент Вычисление длины инструмента в зависимости от MD20360 $MC_TOOL_PARAMETER_DEF_MASK

В машинных данных определяется, как у токарного или шлифовального инструмента следует нормировать износ или длину инструмента в оси диаметра (если имеется).

MD20360 $MC_TOOL_PARAMETER_DEF_MASK.<Bit> = <значение>

<Bit>	<значение>	Объяснение
0	0	В случае токарного и шлифовального инструмента **параметр износа** поперечной оси учитывается при вычислении **значения радиуса**:
	1	В случае токарного и шлифовального инструмента **параметр износа** поперечной оси учитывается при вычислении как **значение диаметра**:
1	0	В случае токарного и шлифовального инструмента **компонент длины инструмента** поперечной оси учитывается при вычислении как **значение радиуса**:
	1	В случае токарного и шлифовального инструмента **компонент длины инструмента** поперечной оси учитывается при вычислении как **значение диаметра**:

Если соответствующие биты установлены, то соответствующая строка нормируется с коэффициентом 0.5. Коррекция посредством SETTCOR выполняется таким образом, что все эффективное изменение длины инструмента равно указанному в <CorVal> значению. Если при расчете длины длина нормируется на основании машинных данных MD20360 $MC_TOOL_PARAMETER_DEF_MASK с коэффициентом 0,5, то коррекция этого компонента должна выполняться с двойным указанным значением.

Пример

MD20360 $MC_TOOL_PARAMETER_DEF_MASK = 2 (длина инструмента должна быть нормирована в оси диаметра с коэффициентом 0.5)

Ось X – ось диаметра.

Программный код	Комментарий
N10 DEF REAL _LEN[11]	
N20 DEF REAL _CORVAL[3]	
N30 $TC_DP1[1,1]=500	; Тип инструмента
N40 $TC_DP2[1,1]=2	; Положение резцов
N50 $TC_DP3[1,1]=3.	; Геометрия – длина 1
N60 $TC_DP4[1,1]=4.	; Геометрия – длина 2
N70 $TC_DP5[1,1]=5.	; Геометрия – длина 3
N80 _CORVAL[0]=1.	
N90 _CORVAL[1]=1.	
N100 _CORVAL[2]=1.	
N110 T1 D1 G18 G0 X0 Y0 Z0	; ==> позиция MCS X1.5 Y5 Z4
N120 R1=SETTCOR(_CORVAL,"G",1,1)	
N130 T1 D1 X0 Y0 Z0	; ==> позиция MCS X2.5 Y6 Z5
N140 R3=$TC_DP3[1,1]	; = 5. = (3.000 + 2.*1.000)
N150 R4=$TC_DP4[1,1]	; = 5. = (4.000 + 1.000)
N160 R5=$TC_DP5[1,1]	; = 6. = (5.000 + 1.000)
N170 M30	

Коррекция длины инструмента должна составлять 1 мм в каждой оси (с N80 по N100). Поэтому в длинах L2 и L3 к исходной длине прибавляется 1 мм. И наоборот, к исходной длине инструмента в L1 прибавляется двойное значение коррекции (2 мм), чтобы общая длина инструмента изменилась, как и требуется, на 1 мм. При сравнении с позициями подвода в кадрах N110 и N130 видно, что каждая позиция оси изменилась на 1 мм.

4.13.14 Считывание соотнесения длин инструмента L1, L2, L3 с осями координат (LENTOAX)

Функция LENTOAX выводит информацию по соотнесению длин инструмента L1, L2 и L3 **активного** инструмента с абсциссой, ординатой и аппликатой. Управление соотнесением абсциссы, ординаты и аппликаты с гео-осями осуществляется через фреймы и активную плоскость (G17 – G19).

При этом рассматривается только геометрический компонент инструмента (от $TC_DP3[<t>,<d>] до $TC_DP5[<t>,<d>]), т. е. возможное отличное от данного соотнесение осей других компонентов (например, износ) на результат не влияет.

Синтаксис

<состояние> = LENTOAX(<AxInd>, <Matrix>[, <Coord>])

Принцип

LENTOAX(...):	Предопределенная функция выведения информации по соотнесению длин инструмента L1, L2 и L3 активного инструмента с осями координат		
	Автономно в кадре:		да
<состояние>:	Возвращаемое значение функции. Отрицательные значения показывают состояния ошибки.		
	Тип данных:		INT
	Значение:	0	Функция OK Информации в <AxInd> для описания достаточно (все компоненты длин инструмента параллельны гео-осям).
		1	Функция OK, но для правильного описания необходимо обработать содержание <Matrix> (компоненты длин инструмента не параллельны гео-осям).
		-1	Недействительная строка в параметре <Coord>.
		-2	нет активного инструмента.
Параметры			
1 <AxInd>:	Если компоненты длин инструмента параллельны гео-осям, то индексы осей, соотнесенных с компонентами длин с L1 по L3, возвращаются в поле **<**AxInd>. • <AxInd> [0]: Абсцисса • <AxInd> [1]: Ордината • <AxInd> [2]: Аппликата		
	Тип данных:		INT[3]
	Значение:	0	Соотнесение отсутствует (ось не существует)
		1 ... 3 или -1 ... -3	номер длины, действующей в соответствующей оси координат. Знак является отрицательным, если компонент длины инструмента указывает в отрицательном направлении координат.
	Если не все компоненты длин параллельны или встречно-параллельны гео-осям, то в <AxInd> возвращается индекс оси, содержащий самую большую долю компонента длин инструмента. В этом случае (если только функция не выводит ошибку по иной причине) возвращаемое значение <Status> = 1. Тогда отображение компонентов длин инструмента с L1 по L3 на гео-осях с 1-ой по 3-ю полностью описывается содержанием 2-го параметра <Matrix>.		
2 <Matrix>:	Матрица, отображающая вектор длин инструмента (L1=1, L2=1, L3=1) в вектор осей координат (абсцисса, ордината, аппликата), т. е. столбцам присвоены компоненты длин инструмента в последовательности L1, L2, L3, строкам — оси в последовательности: абсцисса, ордината, аппликата.		
	Тип данных:		REAL
	В матрице всегда действительны все элементы, в т. ч., если относящаяся к оси координат гео-ось отсутствует, т. е. если соответствующий элемент в <AxInd> = 0.		

3	<Coord>:	Система координат, для которой действует соотнесение (опционально)		
		Тип данных:		STRING
		Символ:	MCS M	Отображение длины инструмента в систему координат станка
			BCS B	Отображение длины инструмента в базовую кинематическую систему
			WCS W	Отображение длины инструмента в систему координат инструмента (по умолчанию)
			KCS K	Отображение длины инструмента в систему координат инструмента кинематической трансформации
			TCS T	Отображение длины инструмента в систему координат инструмента
		Написание символов в строке (прописное или строчное) является произвольным. Если параметр <Coord> не указывается, используется WCS (по умолчанию).		

Примечание

В TCS всегда все компоненты длин инструмента параллельны или встречно-параллельны осям.

Встречно-параллельными компоненты могут быть только тогда, когда активно отражение и установлены следующие установочные данные:

SD42900 $SC_MIRROR_TOOL_LENGTH (смена знака длины инструмента при отражении)

Пример

Стандартный случай: фрезерный инструмент при G17.

L1 действует в Z (аппликата), L2 действует в Y (ордината), L3 действует в X (абсцисса).

Вызов функции в форме:
<состояние>=LENTOAX(<AxInd>,<Matrix>,"WCS")

В этом случае параметр результата <AxInd> содержит значения:

<AxInd>[0] = 3

<AxInd>[1] = 2

<AxInd>[2] = 1

или кратко: (3, 2, 1)

Соответствующая матрица (<Matrix>) в этом случае:

$$\text{<Matrix>} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

Переключение с G17 на G18 или G19 ничего не меняет в результате, т. к. соотнесение компонентов длин с гео-осями изменяется также, как и соотнесение абсциссы, ординаты и аппликаты.

Теперь при активной G17 программируется вращение фрейма вокруг Z на 60 градусов, например,
```
ROT Z60
```

Направление аппликаты (направление Z) остается неизменным, основной компонент L2 теперь лежит в направлении новой оси X, основной компонент L1 в отрицательном направлении оси Y. Поэтому возвращаемое значение <Status>) = 1, <AxInd> содержит значения (2, -3, 1).

Соответствующая матрица (<Matrix>) в этом случае:

$$\text{<Matrix>} = \begin{pmatrix} 0 & \sin 60° & \cos 60° \\ 0 & \cos 60° & -\sin 60° \\ 1 & 0 & 0 \end{pmatrix}$$

4.14 Параметры траектории

4.14.1 Характеристика подачи (FNORM, FLIN, FCUB, FPO)

Для более гибкой задачи характеристики подачи программирование подачи по DIN 66025 дополнено линейными и кубическими характеристиками.

Кубические характеристики могут программироваться напрямую или как интерполирующие сплайны. Благодаря этому возможно, в зависимости от изгиба обрабатываемой детали, непрерывное программирование ровных характеристик скоростей.

Эти характеристики скорости обеспечивают изменения ускорения без рывков и тем самым изготовление равномерных поверхностей детали.

Синтаксис

```
F... FNORM
F... FLIN
F... FCUB
F=FPO(...,...,...)
```

Значение

FNORM	**Обычная** подача по DIN66025 (первичная установка) Значение подачи задается через ход траектории кадра и после действует как модальное значение.
FLIN	**Линейный** профиль скорости движения по траектории Значение подачи линейно выводится от актуального значения в начале кадра до конца кадра через ход траектории и после действует как модальное значение. Характеристика подачи FLIN и FCUB действует с G93, G94 и G95, **не** действует с G96/G961 и G97/G971. При G95 и изменяемой через процентовку или синхронное действие скорости шпинделя возможна некоторая задержка синхронизации.
FCUB	**Кубический** профиль скорости движения по траектории Запрограммированные покадрово значения F соединяются относительно конечной точки кадра через сплайн. Сплайн начинается и заканчивается по касательной к предшествующему и последующему заданному значению подачи. Если адрес F отсутствует в кадре, то для этого используется последнее запрограммированное значение F. Характеристика подачи FCUB действует с G93 и G94, **не** действует с G95, G96/G961 и G97/G971.
F=FPO(...,...,...)	Профиль скорости движения по траектории **через полином** Адрес F обозначает характеристику подачи через полином от актуального значения до конца кадра. После конечное значение считается модальным значением.

Независимо от запрограммированной характеристики подачи действуют функции для программирования параметров движения по траектории.

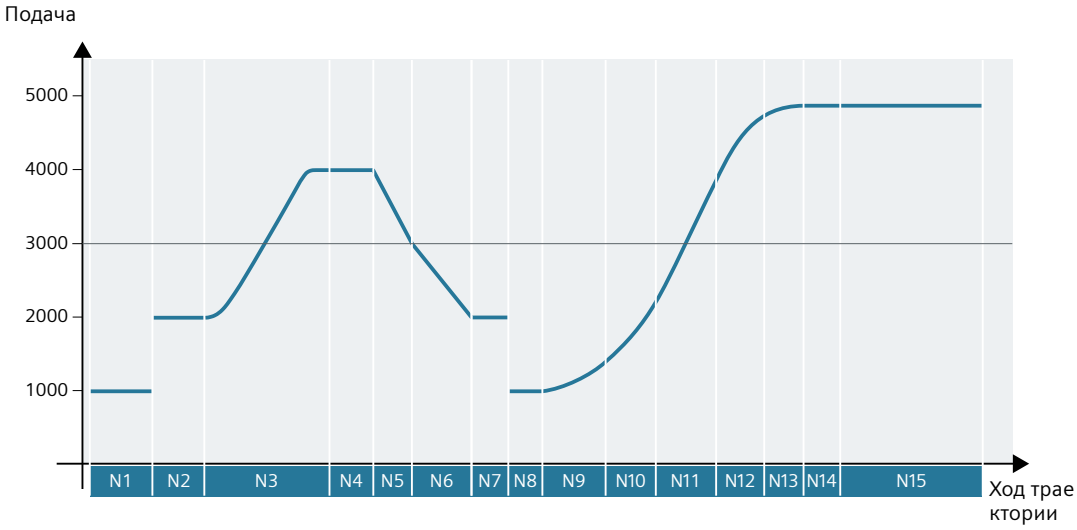
Запрограммированная характеристика подачи всегда действует абсолютно, независимо от G90 или G91.

Примечание

Оптимизация подачи на изогнутых участках траектории

Полином подачи F=FPO() и сплайн подачи FCUB всегда должны проходиться с постоянной подачей по контуру CFC . Благодаря этому возможно создание постоянного по ускорению профиля заданной подачи.

Пример: Различные профили подачи

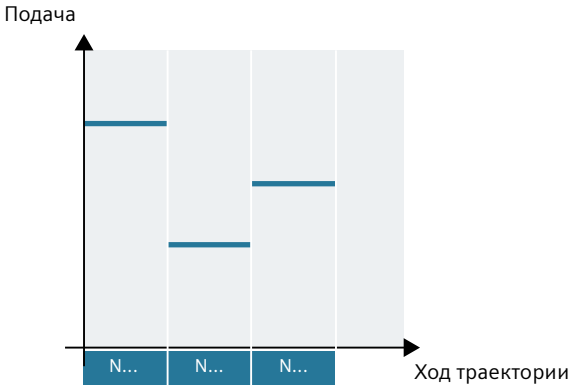


Программный код	Комментарий
`N1 F1000 FNORM G1 X8 G91 G64`	; Постоянный профиль подачи, указание составного размера
`N2 F2000 X7`	; Скачкообразное изменение заданной скорости
`N3 F=FPO(4000, 6000, -4000)`	; Профиль подачи через полином с подачей 4000 на конце кадра.
`N4 X6`	; Полиномиальная подача 4000 действует как модальное значение.
`N5 F3000 FLIN X5`	; Линейный профиль подачи
`N6 F2000 X8`	; Линейный профиль подачи
`N7 X5`	; Линейная подача действует как модальное значение
`N8 F1000 FNORM X5`	; Постоянный профиль подачи со скачкообразным изменением ускорения.
`N9 F1400 FCUB X8`	; Все последующие покадрово запрограммированные значения `F` соединяются сплайнами.
`N10 F2200 X6`	
`N11 F3900 X7`	
`N12 F4600 X7`	
`N13 F4900 X5`	; Отключить профиль сплайна.
`N14 FNORM X5`	
`N15 X20`	

Дополнительная информация

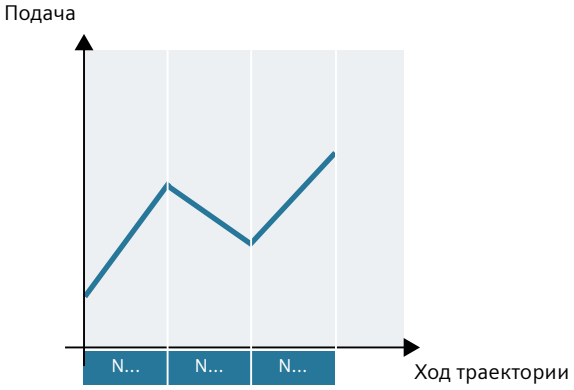
FNORM

Адрес подачи F обозначает подачу по траектории как постоянное значение по DIN 66025.



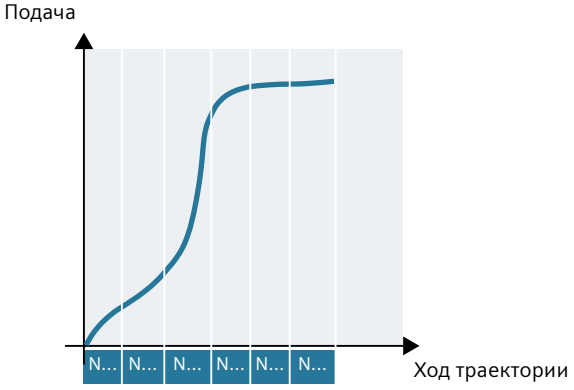
FLIN

Характеристика подачи выводится линейно от актуального значения подачи к запрограммированному значению F до конца кадра.



FCUB

Подача выводится от актуального значения подачи к запрограммированному значению F до конца кадра по кубической характеристике. СЧПУ соединяет все запрограммированные покадрово с активной FCUB значения подачи через сплайны. Значения подачи служат здесь как опорные точки для вычисления сплайн-интерполяции.



F=FPO(... , ... , ...)

Характеристика подачи программируется напрямую через полином. Указание коэффициентов полинома выполняется аналогично полиномиальной интерполяции.

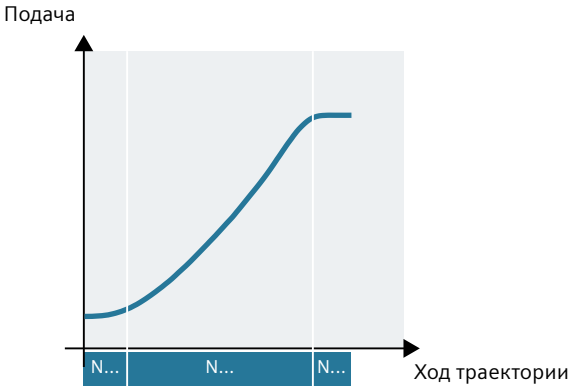
Пример:

```
F=FPO(endfeed,quadf,cubf)
```

endfeed, quadf и cubf - это заранее определенные переменные.

endfeed	подача на конце кадра
quadf	квадратный коэффициент полинома
cubf	кубический коэффициент полинома

При активной FCUB сплайн в начале и в конце кадра тангенциально примыкает к установленной через FPO характеристике.



Поведение при активной функции компрессора

При активной функции компрессора (COMP...) и объединении нескольких кадров в сегмент сплайна действительно следующее:

FNORM:	Для сегмента сплайна действует слово F последнего соответствующего кадра.
FLIN:	Для сегмента сплайна действует слово F последнего соответствующего кадра. Запрограммированное значение F действует к концу сегмента и после выполняется линейный подвод.
FCUB:	Созданный сплайн подачи отклоняется макс. на определенное в машинных данных MD20172 $MC_COMPRESS_VELO_TOL значение от запрограммированных конечных точек.
F=FPO(..., ..., ...):	Эти кадры не сжимаются.

4.14.2 Динамические характеристики

4.14.2.1 Режим ускорения (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA)

Для программирования режима ускорения имеются следующие команды программы обработки детали:

- "BRISK, BRISKA"
  Отдельные оси или траекторные оси движутся с макс. ускорением до достижения запрограммированной скорости подачи (**Ускорение без ограничения рывка**).
- "SOFT, SOFTA"
  Отдельные оси или траекторные оси движутся с постоянным ускорением до достижения запрограммированной скорости подачи (**Ускорение с ограничением рывка**).
- "DRIVE, DRIVEA"
  Отдельные оси или траекторные оси движутся с макс. ускорением до сконфигурированной границы скорости (установка MD!). После этого осуществляется уменьшение ускорения (установка MD!) до достижения запрограммированной скорости подачи.

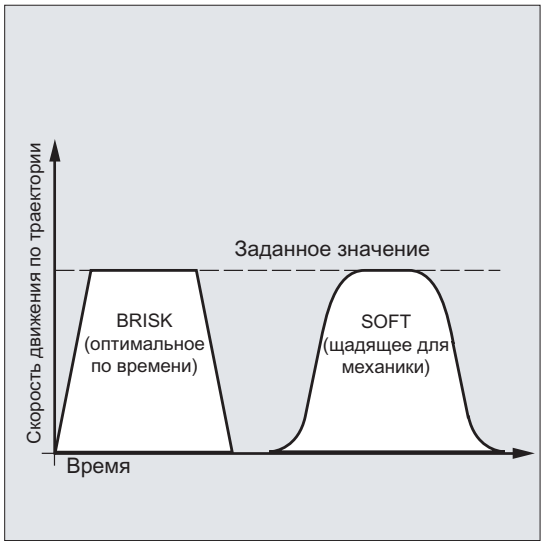


Рисунок 4-13 Характеристика скорости движения по траектории при `BRISK` и `SOFT`

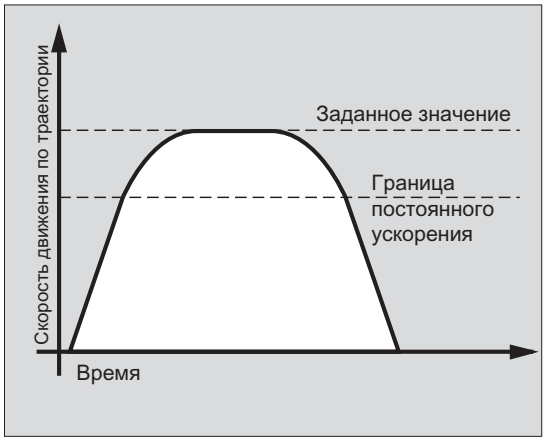


Рисунок 4-14 Характеристика скорости движения по траектории при DRIVE

Синтаксис

```
BRISK
BRISKA(<ось1>,<ось2>,...)
SOFT
SOFTA(<ось1>,<ось2>,...)
DRIVE
DRIVEA(<ось1>,<ось2>,...)
```

Объяснение

BRISK:	Команда для включения "Ускорения без ограничения рывка" для траекторных осей.
BRISKA:	Команда для включения "Ускорения без ограничения рывка" для движений отдельных осей (JOG, JOG/INC, позиционирующая ось, качающаяся ось, и т.д.).
SOFT:	Команда для включения "Ускорения с ограничением рывка" для траекторных осей.
SOFTA:	Команда для включения "Ускорения с ограничением рывка" для движений отдельных осей (JOG, JOG/INC, позиционирующая ось, качающаяся ось, и т.д.).
DRIVE:	Команда для включения уменьшенного ускорения выше сконфигурированной границы скорости (MD35220 $MA_ACCEL_REDUCTION_SPEED_POINT) для траекторных осей.
DRIVEA:	Команда для включения уменьшенного ускорения выше сконфигурированной границы скорости (MD35220 $MA_ACCEL_REDUCTION_SPEED_POINT) для движений отдельных осей (JOG, JOG/INC, позиционирующая ось, качающаяся ось, и т.д.).
(<ось1>,<ось2>,...):	Отдельные оси, для которых должен действовать вызванный режим ускорения.

Граничные условия

Смена режима ускорения при обработке

Если в программе обработки детали режим ускорения изменяется при обработке (BRISK ↔ SOFT), то и в режиме управления траекторией на переходе выполняется смена кадра с точным остановом на конце кадра.

Примеры

Пример 1: SOFT и BRISKA

Программный код
N10 G1 X… Y… F900 SOFT
N20 BRISKA(AX5,AX6)
…

Пример 2: DRIVE и DRIVEA

Программный код
N05 DRIVE
N10 G1 X… Y… F1000
N20 DRIVEA (AX4, AX6)
…

4.14.2.2 Управление ускорением для ведомых осей (VELOLIMA, ACCLIMA, JERKLIMA)

В соединениях осей (Страница 932) (тангенциальное слежение, буксировка, соединение по главному значению, электронный редуктор) ведомые оси/шпиндели перемещаются в зависимости от одной или нескольких ведущих осей/шпинделей.

Управление динамическими ограничениями ведомых осей/шпинделей возможно с помощью функций VELOLIMA, ACCLIMA и JERKLIMA из программы обработки детали или из синхронных действий, и при уже активированном соединении осей.

Примечание

Функция JERKLIMA доступна не для всех типов соединений.

- ACCLIMA и VELOLIMA воздействуют в шпиндельном режиме на общее движение шпинделя, в том числе в случае без сопряжения. В переменных $AC_SMAXACC_INFO или $AC_SMAXVELO_INFO ограничение отображается посредством идентификатора 22.
- В шпиндельном режиме значение VELOLIMA ограничено до 100 процентов.
- В отношении разрешенной скорости для соединений шпинделя действует следующее: Если движение соединения уже исчерпывает максимальное число оборотов, и тем самым запрограммированное основное движение останавливается, то отображается аварийное сообщение 22015 "Ведомый шпиндель не имеет динамики для дополнительного движения".

Синтаксис

```
VELOLIMA(<ось>)=<значение>
ACCLIMA(<ось>)=<значение>
JERKLIMA(<ось>)=<значение>
```

Объяснение

VELOLIMA:	Команда для коррекции спараметрированной максимальной **скорости**
ACCLIMA:	Команда для коррекции спараметрированного максимального **ускорения**
JERKLIMA:	Команда для коррекции спараметрированного максимального **рывка**
<ось>:	Ведомая ось, динамические ограничения которой должны быть исправлены
<значение>:	Процентная поправка

Примеры

Пример 1: Коррекция динамических ограничений для ведомой оси (AX4)

Программный код	Комментарий
...	
VELOLIMA[AX4]=75	; Коррекция ограничения до 75 % зафиксированной в машинных данных макс. осевой скорости.
ACCLIMA[AX4]=50	; Коррекция ограничения до 50 % зафиксированного в машинных данных макс. осевого ускорения.
JERKLIMA[AX4]=50	; Коррекция ограничения до 50 % зафиксированного в машинных данных макс. осевого рывка при движении по траектории.
...	

Пример 2: Электронный редуктор

Ось 4 через соединение "Электронный редуктор" соединена с осью X. Приемистость ведомой оси ограничивается до 70 % макс. ускорения. Макс. допустимая скорость ограничивается до 50 % макс. скорости. После успешного включения соединения макс. допустимая скорость снова устанавливается на 100 %.

Программный код	Комментарий
...	
N120 ACCLIMA[AX4]=70	; Уменьшенное макс. ускорение.
N130 VELOLIMA[AX4]=50	; Уменьшенная макс. скорость.
...	
N150 EGON(AX4,"FINE",X,1,2)	; Включение соединения электронного редуктора.
...	
N200 VELOLIMA[AX4]=100	; Полная макс. скорость.
...	

Пример 3: Управления соединением по главному значению через статическое синхронное действие

Ось 4 через соединение по главному значению соединяется с осью X. Динамическая характеристика через статическое синхронное действие 2 от позиции 100 ограничивается до 80 %.

Программный код	Комментарий
...	
N120 IDS=2 WHENEVER $AA_IM[AX4] > 100 DO ACCLIMA[AX4]=80	; Синхронное действие
N130 LEADON(AX4, X, 2)	; Соединение по главному значению вкл
...	

4.14.2.3 Активация спец. для технологии динамических значений (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH, DYNPREC)

С помощью команд G-группы 59 "Динамический режим для интерполяции траектории" для различных технологических этапов обработки можно активировать подходящую динамику.

Динамические значения и G-команды могут конфигурироваться и поэтому зависят от установок машинных данных.

Дополнительная информация: Описание функций – Основные функции

Синтаксис

Активация динамический значений:
DYNNORM/DYNPOS/DYNROUGH/DYNSEMIFIN/DYNFINISH/DYNPREC

Примечание

Динамические значения активируются уже в том кадре, в котором программируется соответствующая G-команда. Обработка не останавливается.

Запись или чтение определенного элемента поля:
R<m>=$MA...[n,X]
$MA...[n,X]=<значение>

Объяснение

DYNNORM:	Активировать **обычную динамику**
DYNPOS:	Активировать динамику для **режима позиционирования, нарезания внутренней резьбы**
DYNROUGH:	Активировать динамику для **черновой обработки**
DYNSEMIFIN:	Активировать динамику для **предварительной чистовой обработки**
DYNFINISH:	Активировать динамику для **чистовой обработки**
DYNPREC:	Активировать динамику для **точной чистовой обработки**

R<m>:	R-параметр с номером <m>	
$MA...[n,X]:	Машинные данные с определяющим динамику элементом поля	
<n>:	Индекс поля	
	Диапазон значений:	0 ... 5
	0	Обычная динамика (DYNNORM)
	1	Динамика для режима позиционирования (DYNPOS)
	2	Динамика для черновой обработки (DYNROUGH)
	3	Динамика для предварительной чистовой обработки (DYNSEMIFIN)
	4	Динамика для чистовой обработки (DYNFINISH)
	5	Динамика для точной чистовой обработки (DYNPREC)
<X>:	Адрес оси	
<значение>:	Динамическое значение	

Примеры

Пример 1: Активация динамических значений

Программный код	Комментарий
DYNNORM G1 X10	; Первичная установка
DYNPOS G1 X10 Y20 Z30 F…	; Режим позиционирования, нарезание внутренней резьбы
DYNROUGH G1 X10 Y20 Z30 F10000	; Черновая обработка
DYNSEMIFIN G1 X10 Y20 Z30 F2000	; Предварительная чистовая обработка
DYNFINISH G1 X10 Y20 Z30 F1000	; Чистовая обработка
DYNPREC G1 X10 Y20 Z30 F600	; Точная чистовая обработка

Пример 2: Запись или чтение определенного элемента поля

Макс. ускорение для черновой обработки, ось X.

Программный код	Комментарий
R1=$MA_MAX_AX_ACCEL[2,X]	; Чтение
$MA_MAX_AX_ACCEL[2,X]=5	; Запись

4.14.3 Движение с предуправлением (FFWON, FFWOF)

Благодаря предуправлению зависящий от скорости путь выбега уменьшается практически до нуля. Движение с предуправлением способствует более высокой точности контура и тем самым лучшим производственным результатам.

Синтаксис

```
FFWON
FFWOF
```

Значение

FFWON:	Команда для **вкл**ючения предуправления
FFWOF:	Команда для **выкл**ючения предуправления

Примечание

Через машинные данные устанавливается тип предуправления и то, какие траекторные оси должны перемещаться через предуправление.

Стандарт: зависящее от скорости предуправление

Опция: зависящее от ускорения предуправление

Пример

Программный код

```
N10 FFWON
N20 G1 X… Y… F900 SOFT
```

4.14.4 Программирование допуска контура/ориентации (CTOL, OTOL, ATOL)

Адресами CTOL, OTOL, ATOL можно подбирать определенные в машинных и установочных данных допуски на обработку для функций компрессора, перешлифовку и сглаживание ориентации в программе обработки детали.

Запрограммированные значения допуска действуют до их повторного программирования или удаления через присваивание отрицательного значения. Кроме этого, они удаляются при завершении программы или сбросе (Reset). После удаления снова действуют установленные в параметрах значения допусков.

Синтаксис

```
CTOL=<Value>
OTOL=<Value>
ATOL[<Axis>]=<Value>
```

Объяснение

CTOL:	Адрес программирования **допуска контура**			
	Сфера применения:	• всех функций компрессора • всех типов перешлифовки кроме G641 и G644		
	Остановка предварительной обработки:	нет		
	Активность:	модально		
	<Value>:	Значение для допуска контура это линейные данные.		
		Тип:	REAL	
		Единица:	дюйм/мм (в зависимости от текущей настройки указания размеров)	
		Диапазон значений:	≥ 0:	Значение допуска
			< 0:	Удаление запрограммированного значения допуска ⇒ Снова действует значение допуска, установленное в параметрах машинных или установочных данных.
OTOL:	Адрес программирования **допуска ориентации**			
	Сфера применения:	• всех функций компрессора • сглаживания ориентации ORISON • всех типов перешлифовки, кроме G641, G644, OSD		
	Остановка предварительной обработки:	нет		
	Активность:	модально		
	<Value>:	Значение для допуска ориентации это угловые данные.		
		Тип:	REAL	
		Единица:	Градус	
		Диапазон значений:	≥ 0:	Значение допуска
			< 0:	Удаление запрограммированного значения допуска ⇒ Снова действует значение допуска, установленное в параметрах машинных или установочных данных.

ATOL:	Адрес программирования **спец. для оси допуска**			
	Сфера применения:	• всех функций компрессора • сглаживания ориентации ORISON • всех типов перешлифовки, кроме G641, G644, OSD		
	Остановка предварительной обработки:	нет		
	Активность:	модально		
	<Axis>:	Имя оси канала, для которой должен действовать запрограммированный допуск		
	<Value>:	Значение для осевого допуска, в зависимости от типа оси (линейная или круговая ось) это линейные или угловые данные.		
		Тип:	REAL	
		Единица:	для линейных осей:	дюйм/мм (в зависимости от текущей настройки указания размеров)
			для круговых осей:	Градус
		Диапазон значений:	≥ 0:	Значение допуска
			< 0:	Удаление запрограммированного значения допуска ⇒ Снова действует значение допуска, установленное в параметрах машинных или установочных данных.

Примечание

Приоритет запрограммированных посредством CTOL и OTOL спец. для канала значений допуска выше, чем у запрограммированных посредством ATOL спец. для оси значений допуска.

Примечание

Масштабирующие фреймы

Масштабирующие фреймы действуют на запрограммированные допуски так же, как и на позиции осей, т. е. относительный допуск остается тем же.

Пример

Программный код	Комментарий
COMPCAD G645 G1 F10000	; Активировать функцию компрессора COMPCAD.
X... Y... Z...	; Здесь действуют машинные и установочные данные.
X... Y... Z...	
X... Y... Z...	
CTOL=0.02	; С этого места действует допуск контура в 0,02 мм.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

Программный код	Комментарий
ASCALE X0.25 Y0.25 Z0.25	; С этого места действует допуск контура в 0,005 мм.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
CTOL=-1	; С этого места снова действуют машинные и установочные данные.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

Дополнительная информация

Системные переменные

Через следующие системные переменные могут считываться действующие на соответствующий момент допуски:

- Чтение с остановкой предварительной обработки (в программе обработки детали и синхронном действии)
  - $AC_CTOL
    Спец. для канала допуск контура, который действовал при подготовке текущего кадра главного хода
    Если действующего допуска контура нет, то $AC_CTOL выводит корень из суммы квадратов допусков гео-осей.
  - $AC_OTOL
    Спец. для канала допуск ориентации, который действовал при подготовке текущего кадра главного хода.
    Если действующего допуска ориентации нет, то $AC_OTOL выводит во время текущей трансформации ориентации корень от суммы квадратов допусков осей ориентации, в иных случаях — значение "-1".
  - $AA_ATOL[<ось>]
    Спец. для оси допуск контура, который действовал при подготовке текущего кадра главного хода.
    Если допуск контура активен, то $AA_ATOL[<гео-ось>] выводит допуск контура, деленный на корень из количества гео-осей.
    Если допуск ориентации и трансформация ориентации активны, то $AA_ATOL[<ось ориентации>] выводит допуск ориентации, деленный на корень из количества осей ориентации.

Примечание

Если значения допуска не были запрограммированы, то переменные $A не являются достаточно дифференцированными для различения допусков отдельных функций.

Такие случаи возможны, если машинные и установочные данные устанавливают различные допуски для функций компрессора, перешлифовки и сглаживания ориентации. В этом случае системные переменные выводят наибольшее значение, встречающееся в активных на настоящий момент функциях. Если, например, активны функция компрессора с допуском ориентации 0,1° и сглаживание ориентации ORISON с 1°, то $AC_OTOL выводит значение "1". Если сглаживание ориентации отключается, то $AC_OTOL выводит значение "0,1"

- Чтение без остановки предварительной обработки (только в программе обработки детали)
  - $P_CTOL
    Действующий на соответствующий момент спец. для канала допуск контура
  - $P_OTOL
    Действующий на соответствующий момент спец. для канала допуск ориентации
  - $PA_ATOL
    Действующий на соответствующий момент спец. для оси допуск контура

Граничные условия

Запрограммированные посредством CTOL, OTOL, ATOL допуски действуют и для функций, косвенно зависящих от этих допусков:

- Ограничение ошибки хорды при расчете заданного значения
- Основные функции режима поверхностей произвольной формы

Программирование CTOL, OTOL, ATOL **не** затрагивает следующие функции перешлифовки:

- перешлифовка ориентации с OSD
  OSD использует не допуск, а расстояние до перехода кадра.
- перешлифовка с G644
  G644 служит не для обработки, а для оптимизации смен инструмента и других движений без обработки.
- перешлифовка с G645
  G645 практически всегда ведет себя как G642 и тем самым использует запрограммированные допуски. Только на переходах (с постоянным касанием) между кадрами с перепадом кривизны, например, на тангенциальном переходе окружность — прямая, используется значение допуска из машинных данных MD33120 $MA_PATH_TRANS_POS_TOL. Путь перешлифовки в таких местах может проходить и по внешней стороне запрограммированного контура, что не в полной мере поддерживается многими приложениями. Кроме этого, как правило, для компенсации изменений кривизны достаточно небольшого постоянного допуска, который не является проблемой для программиста ЧПУ.

4.14.5 Включение/выключение программируемой точности контура (CPRECON, CPRECOF)

Функция "Программируемая точность контура" сокращает погрешности траектории на изогнутых контурах за счет автоматической адаптации скорости.

Она включается и выключается модально действующими командами G-группы 39 (программируемая точность контура).

Синтаксис

```
CPRECON
...
CPRECOF
```

Значение

CPRECON:	**вкл**ючить функцию "Программируемая точность контура"
CPRECOF:	**выкл**ючить функцию "Программируемая точность контура"

Пример

Программный код	Комментарий
`N10 G0 X0 Y0`	
`N20 CPRECON`	; Включение "Программируемой точности контура".
`N30 G1 G64 X100 F10000`	; Обработка с 10 м/мин в режиме управления траекторией.
`N40 G3 Y20 J10`	; Автоматическое ограничение подачи в круговом кадре.
`N50 G1 X0`	; Подача снова без ограничения (10 м/мин).
...	
`N100 CPRECOF`	; Выключение "Программируемой точности контура".
`N110 G0 ...`	

См. также

Программирование допуска контура/ориентации (CTOL, OTOL, ATOL) (Страница 876)

Дополнительная информация

Точность контура

Соблюдаемая точность контура задается в зависимости от конфигурации станка (MD20470 $MC_MC_CPREC_WITH_FFW; см. данные изготовителя станка) либо через установочные данные SD42450$SC_CONTPREC, либо через запрограммированный допуск контура CTOL. Чем меньше значение и чем меньше коэффициент K_V гео-осей, тем сильнее уменьшается подача по траектории на изогнутых контурах.

Точность контура при движениях ускоренного хода

При обработке деталей с изогнутым контуром и активной функцией "Программируемая точность контура" с целью поддержания заданной точности контура скорость движения по траектории ограничивается и при движениях инструмента ускоренным ходом, напр., на углах и в кадрах перехода при обходе детали. Для минимизации снижения скорости движения по траектории при движениях ускоренного хода, для функции "Программируемая точность контура" может быть установлена отличная от обработки детали точность контура ускоренного хода:

SD42451 $SC_CONTPREC_G00_ABS (точность контура при ускоренном ходе)

При SD42451 = 0 для движений ускоренного хода действует установленная в $SC_CONTPREC[DYNNORM] точность контура.

Мин. подача по траектории

Через следующие установочные данные пользователь может задать мин. подачу по траектории для функции "Программируемая точность контура":

SD42460 $SC_MINFEED (мин. подача по траектории при CPRECON)

Подача не ограничивается этим значением, разве что было запрограммировано более низкое F-значение или динамические ограничения осей вызывают более низкую скорость движения по траектории.

Отсутствие влияния на позиционирующие оси.

Функция "Программируемая точность контура" рассматривает только гео-оси контура. Она не влияет на скорости позиционирующих осей.

Поведение при запуске программы обработки детали и после сброса канала или сброса конца программы

При запуске программы обработки детали и после сброса канала или сброса конца программы включается определенная для группы G-функций 39 первичная установка СЧПУ (см. информацию изготовителя станка).

4.14.6 Включение/выключение автоматического переключения фильтров (AFISON, AFISOF)

С помощью функции "Автоматическое переключение фильтров" пользователь может отмечать области в программе ЧПУ, в которых все разрешенные для этой функции оси при движениях G0 автоматически будут переключаться на 2-ю цепочку фильтров. Если параметры 2-й цепочки фильтров настроены таким образом, что происходит усиленное гашение вибраций, то рывок для движений G0 может быть увеличен. Благодаря этому скорость движения по траектории для движений G0, напр., на углах, снижается не так сильно, что способствует сокращению времени выполнения программы.

Синтаксис

Для языковых команд включения и выключения функции требуется отдельный кадр.

```
AFISON
...
AFISOF
```

Значение

AFISON	**Вкл**ючить функцию "Автоматическое переключение фильтров"
AFISOF	**Выкл**ючить функцию "Автоматическое переключение фильтров"

Пример

Программный код	Комментарий
...	
N390 G1 X1100	; цепочка фильтров 1 активна
	; останов на X=1100
N400 **AFISON**	; автоматическое переключение на цепочку фильтров 2
N410 G0 X1150	; останов на X=1150
N420 G1 X1200	; автоматическое переключение на цепочку фильтров 1
N430 **AFISOF**	
N440 G0 X1300	; цепочка фильтров 1 активна и в кадре G0

Программный код	Комментарий
N450 G1 X1400	
N460 **AFISON**	
N470 G1 X1450	
N480 G1 X1500	
N490 **AFISOF**	
N500 G1 X1600	
N510 G0 X1700	; цепочка фильтров 1 активна и в кадре G0
N520 **AFISON**	
N530 G1 X1750	; останов на X=1750
	; автоматическое переключение на цепочку фильтров 2
N540 G0 X1800	; останов на X=1800
	; автоматическое переключение на цепочку фильтров 1
N550 **AFISOF**	
N560 G1 X1900	
N570 G0 X2000	; цепочка фильтров 1 активна и в кадре G0
	; останов на X=2100
N580 **AFISON**	; автоматическое переключение на цепочку фильтров 2
N590 G0 X2050	
N600 G0 X2100	; останов на X=2100
	; автоматическое переключение на цепочку фильтров 1
N610 **AFISOF**	
N620 G0 X2200	; цепочка фильтров 1 активна и в кадре G0
...	

Дополнительная информация

Условия

Для использования функции "Автоматическое переключение фильтров" должны быть выполнены следующие условия:

- Должна быть установлена подлежащая лицензированию опция "Адаптация рывка":
  MD19321 $ON_TECHNO_FUNCTION_MASK_1, бит 22 = 1
- Функция должна быть разрешена для канала:
  MD20630 $MC_AFIS_MODE = 1
- Функция должна быть разрешена для каждой предусмотренной для автоматического переключения фильтров оси:
  MD32332 $MD_AFIS_ENABLE = 1
- Во всех разрешенных для функции осях:
  - должно быть активным ограничение рывка:
    MD32400 $MA_AX_JERK_ENABLE = 1
  - должны быть выбраны и установлены два типа фильтров рывка (→ MD32402 $MA_AX_JERK_MODE).

Если условия не выполнены, то появляется аварийное сообщение 14782 или 26380.

Останов при включении/выключении

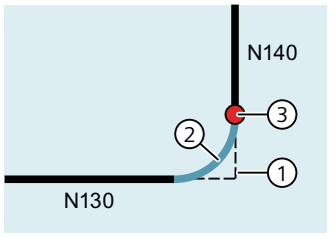
Останов движения по траектории при включении и выключении функции выполняется только тогда, когда требуется переключение цепочки фильтров заданного значения положения.

Перешлифовка

Если при активной перешлифовке (G64x) при смене кадра происходит останов и автоматическое переключение цепочки фильтров, то это не влияет на контур перешлифовки.

Пример:

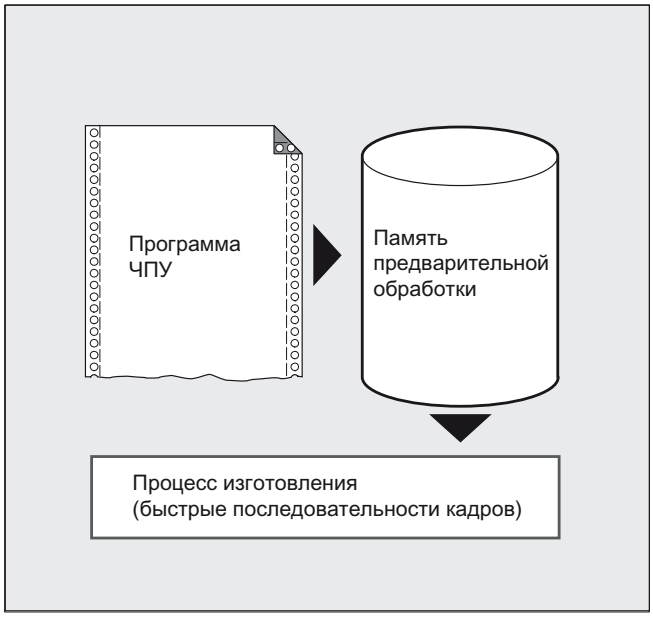
```
N100 G642 F1000 CTOL=10 CTOLG0=10
N110 G0 X0 Y0
N120 AFISON
N130 G1 X100 Y0
N140 G0 X100 Y100
```



① Запрограммированная траектория
② Контур перешлифовки
③ Останов и переключение на 2-ю цепочку фильтров

4.14.7 Выполнение программы с памятью предварительной обработки (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE)

В зависимости от конфигурации, СЧПУ имеет определенное число памятей предварительной обработки, которые сохраняют подготовленные кадры перед выполнением и выводят их в процессе производства как быстрые последовательности кадров. Благодаря этому возможно прохождение коротких путей с высокими скоростями. Если это допускает остаточное время СЧПУ, память предварительной обработки всегда заполняется.



Обозначить участок обработки

Участок обработки, который должен быть буферизирован в память предварительной обработки, обозначается в программе обработки детали в начале с "STOPFIFO" и на конце с "STARTFIFO". Выполнение подготовленных и буферизированных кадров начинается только после команды "STARTFIFO" или заполнения памяти предварительной обработки.

Автоматическое управление памятью предварительной обработки

Автоматическое управление памятью предварительной обработки вызывается командой "FIFOCTRL". "FIFOCTRL" сначала действует идентично "STOPFIFO". При каждом программировании ожидается заполнение памяти предварительной обработки и после начинается выполнение. Разница состоит в поведении при опустошении памяти предварительной: С "FIFOCTRL" начиная с уровня заполнения в 2/3 скорость движения по траектории значительно уменьшается, чтобы не допустить полного опустошения и торможения до состояния покоя.

Остановка предварительной обработки

Подготовка и буферизация кадров останавливается, если в кадре запрограммирована команда "STOPRE". Следующий кадр выполняется только тогда, когда все подготовленные

и сохраненные прежде кадры полностью выполнены. Предыдущий кадр останавливается с точным остановом (как G9).

ВНИМАНИЕ
Отмена программы При включенной коррекция инструмента и при сплайн-интерполяциях нельзя программировать "STOPRE", т. к. в этом случае будут прерваны связанные последовательности кадров.

Синтаксис

Таблица 4-3 Обозначить участок обработки:

```
STOPFIFO
...
STARTFIFO
```

Таблица 4-4 Автоматическое управление памятью предварительной обработки:

```
...
FIFOCTRL
...
```

Таблица 4-5 Остановка предварительной обработки:

```
...
stopre
...
```

Примечание

Команды "STOPFIFO", "STARTFIFO", "FIFOCTRL" и "STOPRE" должны программироваться в отдельном кадре.

Значение

STOPFIFO:	"STOPFIFO" обозначает начало участка обработки, который должен быть помещен в память предварительной обработки. С "STOPFIFO" обработка останавливается и память предварительной обработки заполняется: • до определения "STARTFIFO" или "STOPRE" или • до заполнения памяти предварительной обработки или • до достижения конца программы.
STARTFIFO:	С "STARTFIFO" запускается быстрое выполнение участка обработки, параллельно с этим осуществляется заполнение памяти предварительной обработки
FIFOCTRL:	Включение автоматического управления памятью предварительной обработки
STOPRE:	Остановить предварительную обработку

Примечание

Заполнение памяти предварительной обработки не выполняется или прерывается, если участок обработки содержит команды, вызывающие режим без буферизации (реферирование, измерительные функции, ...).

Примечание

При обращении к данным состояния станка ($SA...), СЧПУ создает внутреннюю остановку предварительной обработки.

Пример: Остановить предварительную обработку

Программный код	Комментарий
...	
N30 MEAW=1 G1 F1000 X100 Y100 Z50	; Кадр измерения со щупом первого измерительного входа и линейная интерполяция.
N40 STOPRE	; Остановка предварительной обработки.
...	

4.14.8 Определение областей задержки останова (DELAYFSTON, DELAYFSTOF)

Для определения условно прерываемой области задержки останова в программе обработки детали служат предварительно определенные процедуры DELAYFSTON и DELAYFSTOF.

Примечание

В синхронных действиях использование DELAYFSTON и DELAYFSTOF **не** допускается!

Синтаксис

```
DELAYFSTON
...
DELAYFSTOF
```

Объяснение

DELAYFSTON:	Определение начала области задержки останова	
	Автономно в кадре:	да
DELAYFSTOF:	Определить конец области задержки останова	
	Автономно в кадре:	да

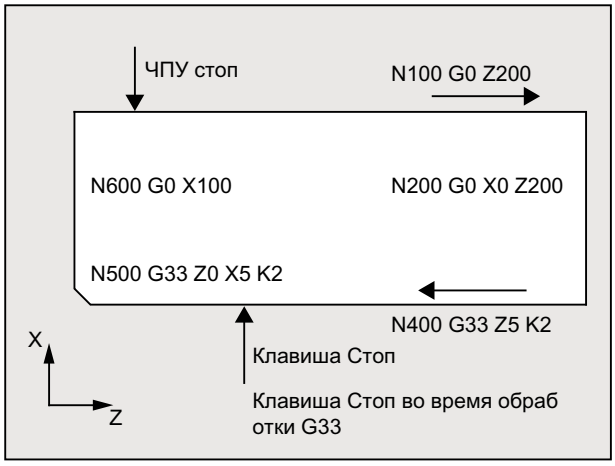
Пример программирования

Циклично повторяется следующий программный блок:

Программный код

```
...
N99 MY_LOOP:
N100 G0 Z200
N200 G0 X0 Z200
N300 DELAYFSTON
N400 G33 Z5 K2 M3 S1000
N500 G33 Z0 X5 K3
N600 G0 X100
N700 DELAYFSTOF
N800 GOTOB MY_LOOP
...
```

На следующем изображении видно, что пользователь нажимает на Стоп в области задержки останова, а ЧПУ начинает процесс торможения вне области задержки останова, т. е. в кадре N100. Тем самым ЧПУ останавливается в передней области N100.



Дополнительная информация

Конец подпрограммы

При завершении подпрограммы, в которой была вызвана DELAYFSTON, не явно активируется DELAYFSTOF.

Вложенность

Если подпрограмм 1 в области задержки останова вызывает подпрограмму 2, то вся подпрограмма 2 это область задержки останова. В частности, DELAYFSTOF не действует в подпрограмме 2.

Пример:

Программный код	Комментарий
N10010 **DELAYFSTON**	; Кадры с N10xxx программный уровень 1.
N10020 R1 = R1 + 1	
N10030 G4 F1	; Область задержки останова начинается.
...	
N10040 подпрограмма2	
...	
...	; Интерпретация подпрограммы 2.
N20010 **DELAYFSTON**	; Не действует, повторное начало, 2-й уровень.
...	
N20020 **DELAYFSTOF**	; Не действует, завершение на другом уровне.
N20030 RET	
N10050 **DELAYFSTOF**	; Конец области задержки останова в той же плоскости.
...	
N10060 R2 = R2 + 2	
N10070 G4 F1	; Область задержки останова завершается. Остановы действуют с этого места сразу же.

Системные переменные

О нахождении этапа программы обработки детали в данный момент времени в области задержки останова можно судить по следующим системным переменным:

- в программе обработки детали - $P_DELAYFST
- в синхронных действиях с $AC_DELAYFST

Значение	Объяснение
0	Область задержки останова не активна
1	Область задержки останова активна

4.14.9 Пропуск места в программе для SERUPRO (IPTRLOCK, IPTRUNLOCK)

Для определенных сложных механических ситуаций на станке необходимо недопущение поиска кадра SERUPRO.

С помощью программируемого указателя прерываний существует возможность вмешательства для установа при "Поиске на месте прерывания" перед непригодным для поиска местом.

Могут быть также определены непригодные для поиска области в областях программы обработки детали, повторный вход ЧПУ в которые невозможен. При отмене программы ЧПУ отмечает последний обработанный кадр, поиск которого возможен через интерфейс пользователя HMI.

Синтаксис

```
IPTRLOCK
IPTRUNLOCK
```

Команды стоят отдельно в одной строке программы обработки детали и обеспечивают возможность программируемого указателя прерываний.

Объяснение

IPTRLOCK:	Начало непригодного для поиска сегмента программы
IPTRUNLOCK:	Конец непригодного для поиска сегмента программы

Обе команды допускаются только в программах обработки детали, но **не** в синхронных действиях.

Пример

Вложение непригодных для поиска сегментов программы на двух программных уровнях с не явной "IPTRUNLOCK". Не явная "IPTRUNLOCK" в подпрограмме 1 завершает непригодную для поиска область.

Программный код	Комментарий
N10010 IPTRLOCK()	

Программный код	Комментарий
`N10020 R1 = R1 + 1`	
`N10030 G4 F1`	; Кадр останова, начинающий непригодный для поиска сегмент программы.
...	
`N10040 подпрограмма2`	
...	; Интерпретация подпрограммы 2.
`N20010 IPTRLOCK ()`	; Не действует, повторное начало.
...	
`N20020 IPTRUNLOCK ()`	; Не действует, завершение на другом уровне.
`N20030 RET`	
...	
`N10060 R2 = R2 + 2`	
`N10070 RET`	; Конец непригодного для поиска сегмента программы.
`N100 G4 F2`	; Главная программа продолжается.

После прерывание на 100 снова выводит указатель прерываний.

Дополнительная информация

Регистрация и поиск непригодных для поиска областей

Непригодные для поиска сегменты программы обозначаются с помощью языковых команд "IPTRLOCK" и "IPTRUNLOCK".

Команда "IPTRLOCK" фиксирует указатель прерываний на исполняемом в главном ходе отдельном кадре (SB1). Этот кадр в дальнейшем обозначается как кадр останова. Если после "IPTRLOCK" возникает отмена программы, то на интерфейсе пользователя HMI можно искать этот так называемый кадр останова.

Повторный установ на актуальный кадр

Указатель прерываний с помощью "IPTRUNLOCK" для следующего сегмента программы будет установлен на актуальный на момент прерывания кадр.

После найденной цели поиска с тем же кадром останова можно повторить новую цель поиска.

Отредактированный пользователем указатель прерываний снова должен быть удален через HMI.

Правила вложенности

Следующие моменты регулируют взаимодействие языковых команд "IPTRLOCK" и "IPTRUNLOCK" со вложенностями и концом программы обработки детали:

1. При завершении подпрограммы, в которой была вызвана "IPTRLOCK", , не явно активируется "IPTRUNLOCK".
2. "IPTRLOCK" в непригодной для поиска области не действует.
3. Если подпрограмма 1 вызывает в непригодной для поиска области подпрограмму 2, то вся подпрограмма 2 остается непригодной для поиска. В частности, "IPTRUNLOCK" не действует в подпрограмме 2.

Дополнительная информация: Описание функций – Основные функции

Системная переменная

Непригодная для поиска область может быть распознана в программе обработки детали с помощью "$P_IPTRLOCK".

Автоматический указатель прерываний

Функция автоматического указателя прерываний автоматически устанавливается определенный ранее тип связи как непригодный для поиска. Посредством машинных данных автоматический указатель прерываний активируется для

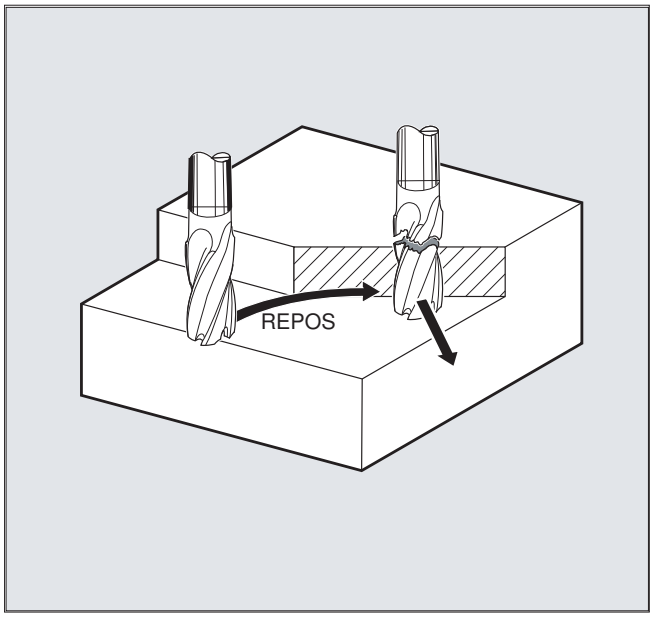
- электронного редуктора при "EGON",
- осевого соединения по главному значению при "LEADON".

При пересечении запрограммированного и активируемого через машинные данные автоматического указателя прерываний, создается макс. возможная непригодная для поиска область.

4.14.10 Повторный подвод к контуру (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)

Если при обработке текущая программа прерывается и выполняется отвод инструмента – к примеру, из-за поломки инструмента или для измерения детали – то затем можно выполнить повторный подвод к контуру в выбранной точке, используя программное управление.

Команда `REPOS` действует в ASUP, как возврат из подпрограммы (например, M17). Последующие кадры больше не исполняются. По прерыванию выполнения программы см. также "Обработчик прерываний (ASUP) (Страница 559)".



Синтаксис

```
REPOSA RMIBL DISPR=…
REPOSA RMBBL
REPOSA RMEBL
REPOSA RMNBL
REPOSL RMIBL DISPR=…
REPOSL RMBBL
REPOSL RMEBL
REPOSL RMNBL
REPOSQ RMIBL DISPR=… DISR=…
REPOSQ RMBBL DISR=…
REPOSQ RMEBL DISR=…
REPOSQA DISR=…
REPOSH RMIBL DISPR=… DISR=…
REPOSH RMBBL DISR=…
REPOSH RMEBL DISR=…
REPOSHA DISR=…
```

Значение

Выбор пути подвода

REPOSA:	Повторный подвод к контуру с геометрическими осями на одной прямой. Все прочие оси канала также повторно позиционируются.
REPOSL:	Повторный подвод к контуру с геометрическими осями на одной прямой. Все прочие оси канала должны быть явно запрограммированы.
REPOSQ DISR=… :	Повторный подвод к контуру с геометрическими осями на четверти окружности с радиусом DISR. Все прочие оси канала должны быть явно запрограммированы.
REPOSQA DISR=… :	Повторный подвод к контуру с геометрическими осями на четверти окружности с радиусом DISR. Все прочие оси канала также повторно позиционируются.
REPOSH DISR=… :	Повторный подвод к контуру с геометрическими осями на полуокружности с диаметром DISR. Все прочие оси канала должны быть явно запрограммированы.
REPOSHA DISR=… :	Повторный подвод к контуру с геометрическими осями на полуокружности с радиусом DISR. Все прочие оси канала также повторно позиционируются.

Выбор точки повторного подвода

RMIBL:	Подвод к точке прерывания
RMIBL DISPR=…:	Точка входа на расстоянии DISPR в мм/дюймах перед точкой прерывания
RMBBL:	Подвод к начальной точке кадра
RMEBL:	Подвод к конечной точке кадра

RMEBL DISPR=… :	Подвод к конечной точке кадра на расстоянии DISPR перед конечной точкой
RMNBL:	Подвод к ближайшей точке траектории
A0 B0 C0 :	Оси, в которых должен выполняться подвод

Примечание

Совместимость

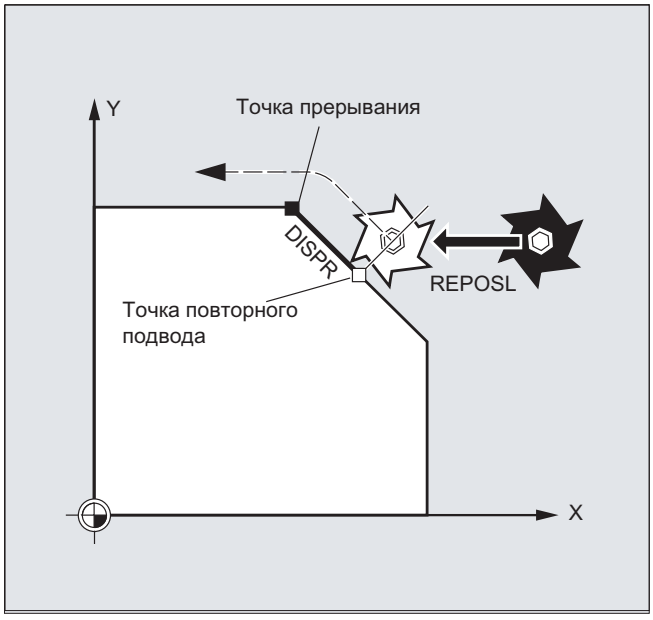
Для обеспечения совместимости с прежними версиями ПО, можно продолжать программировать режим подвода REPOS через модальные команды RMI, RMB, RME и RMN. При использовании внутри ASUP эту подпрограмму нужно снабдить атрибутом SAVE в операторе PROC. В противном случае используемый в ASUP модальный режим подвода REPOS, если он отклоняется от предустановки RMI, действует и для последующих процессов REPOS.

Повторный подвод к контуру на прямой, REPOSA, REPOSL

Инструмент движется к точке повторного подвода по прямой.

Пример

```
REPOSL RMIBL DISPR=6 F400
```

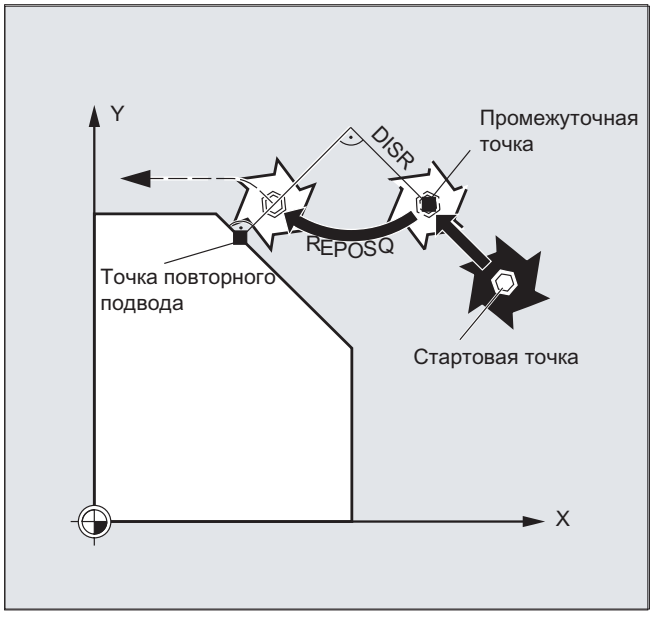


Повторный подвод к контуру по четверти окружности, REPOSQ, REPOSQA

Инструмент движется к точке повторного подвода по четверти круга с радиусом DISR=. . . . Необходимую промежуточная точку между стартовой точкой и точкой повторного подвода СЧПУ вычисляет автоматически.

Пример

```
REPOSQ RMIBL DISR=10 F400
```

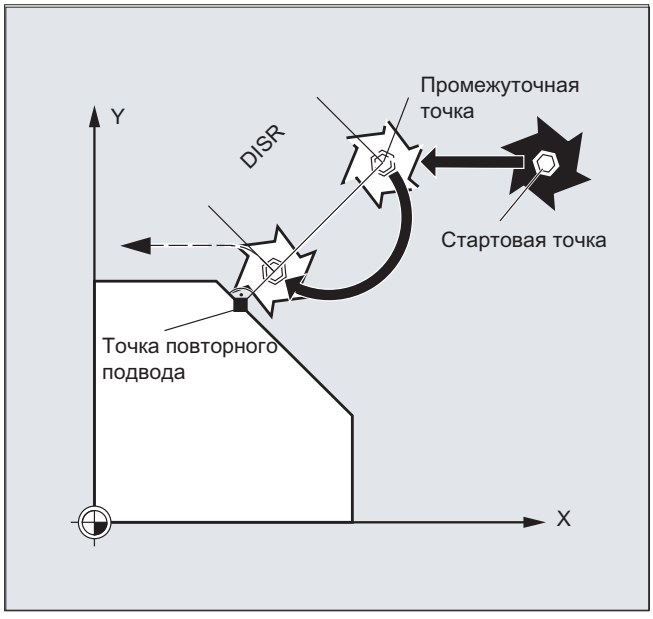


Повторный подвод к контуру по полукругу, REPOSH, REPOSHA

Инструмент движется к точке повторного подвода по полукругу с диаметром `DISR=...`. Необходимую промежуточная точку между стартовой точкой и точкой повторного подвода СЧПУ вычисляет автоматически.

Пример

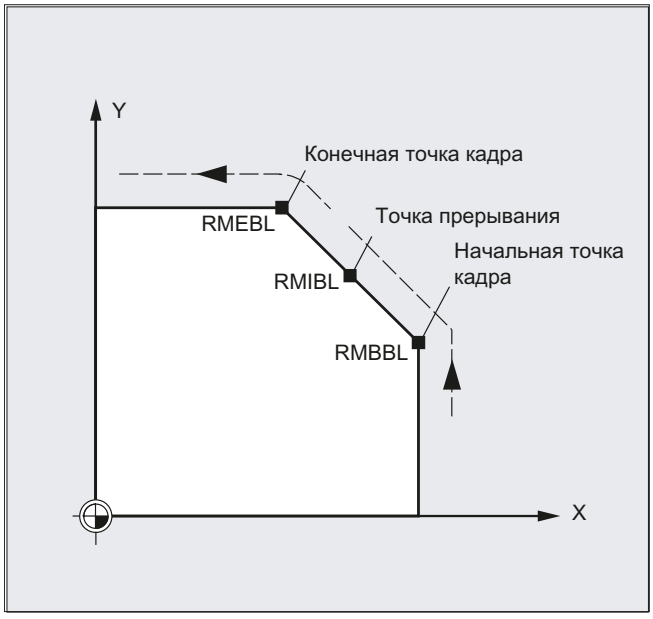
```
REPOSH RMIBL DISR=20 F400
```



Определение точки повторного подвода (не для подвода SERUPRO с RMNBL)

Относительно кадра ЧПУ, в котором выполнение программы было прервано, можно выбирать между тремя точками повторного подвода:

- RMIBL, точка прерывания
- RMBBL, начальная точка кадра или последняя конечная точка
- RMEBL, конечная точка кадра



С RMIBL DISPR=... или с RMEBL DISPR=... можно определить точку повторного подвода, лежащую перед точкой прерывания или перед конечной точкой кадра.

С DISPR=... описывается путь контура в мм/дюймах, на который точка повторного подвода удалена от точки прерывания или конечной точки. Эта точка может и для больших по величине значений лежать макс. в начальной точке кадра.

Если DISPR=... не программируется, то действует DISPR=0 и тем самым точка прерывания (для RMIBL) или конечная точка кадра (для RMEBL).

Знак DISPR

Знак DISPR обрабатывается. При положительном знаке сохраняется прежнее поведение.

При отрицательном знаке продолжение за точкой прерывания или при RMBBL за стартовой точкой.

Расстояние м/у точкой прерывания и точкой возобновления получается из величины DISPR. Эта точка может и для больших по величине значений лежать макс. в конечной точке кадра.

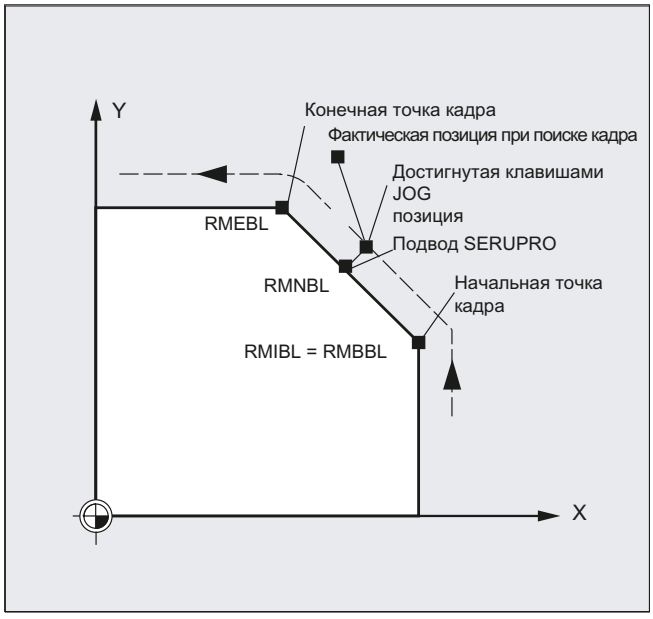
Пример использования:

Датчик определяет приближение к прихвату. Запускается ASUP, с которой выполняется обход прихвата.

После с отрицательной DISPR выполняется репозиционирование на точку за прихватом и программа продолжается.

Подвод SERUPRO с RMNBL

Если при обработке в любом месте инициируется отмена, то при подводе SERUPRO с RMNBL выполняется переход кратчайшего пути от места отмены, чтобы после лишь обработать остаточный путь. Для этого пользователь запускает процесс SERUPRO на кадр прерывания и выполняет позиционирование с помощью клавиш JOG на позицию перед поврежденным местом конечного кадра.



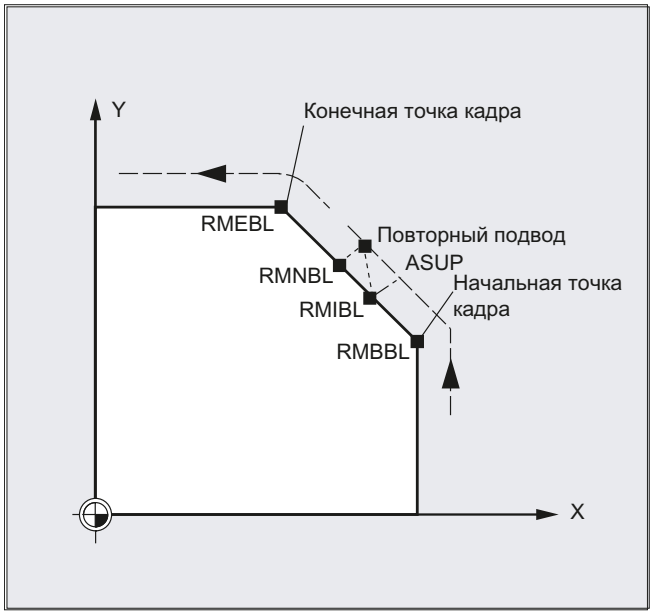
Примечание

SERUPRO

Для SERUPRO RMIBL и RMBBL идентичны. RMNBL ограничивается не только SERUPRO, но имеет общее действие.

Подвод к ближайшей точке траектории RMNBL

На момент интерполяции REPOSA после прерывания кадр повторного подвода с RMNBL не начинается заново, а лишь обрабатывается остаточный путь. Выполняется подвод к ближайшей точке траектории прерванного кадра.



Состояние для действительного режима REPOS

Действительный режим REPOS прерванного кадра может быть считан через синхронные действия с переменной $AC_REPOS_PATH_MODE:

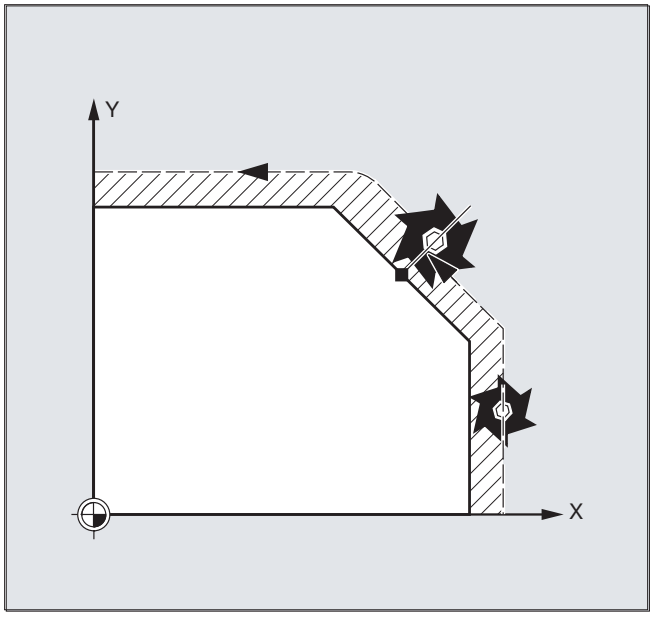
0 подвод не определен
1 RMBBL: подвод к началу
2 RMIBL: подвод к точке прерывания
3 RMEBL: подвод к конечной точке кадра
4 RMNBL: подвод к ближайшей точке траектории прерванного кадра

Подвод с новым инструментом

Если выполнение программы было остановлено из-за поломки инструмента:

При программировании нового номера D программа продолжается с точки повторного подвода с измененными значениями коррекции инструмента.

При изменении значений коррекции инструмента может случиться, что подвод к точке прерывания будет более невозможен. В этом случае выполняется подвод к ближайшей от точки прерывания точке на новом контуре (возможно измененной на DISPR).



Подвод к контуру

Движение, с которым инструмент снова подводится к контуру, может быть запрограммировано. Адреса перемещаемых осей указываются со значением ноль.

С помощью команд REPOSA, REPOSQA и REPOSHA выполняется автоматическое репозиционирование всех осей. Указание оси не требуется.

При программировании REPOSL, REPOSQ и REPOSH все гео-оси выполняют подвод автоматически, т. е. и без указания в команде. Все другие оси должны быть указаны в команде.

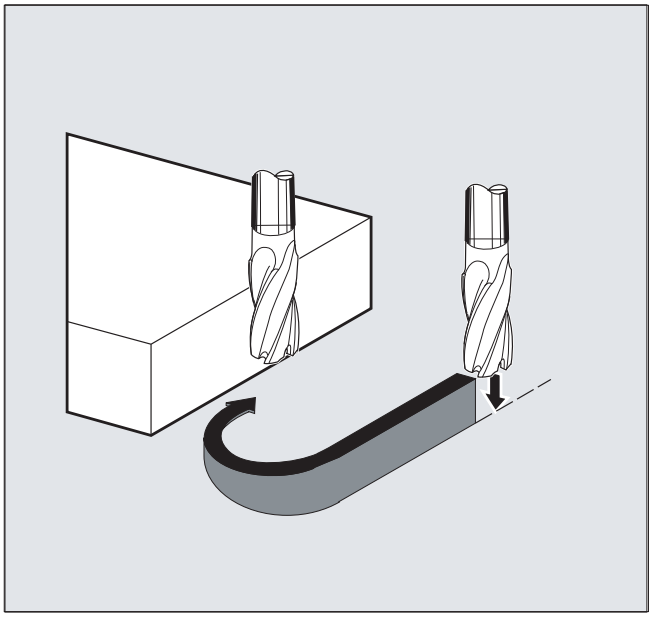
Для круговых движений REPOSH и REPOSQ действует:

Окружность проходится в указанной рабочей плоскости G17 до G19.

Если в кадре подвода указывается третья гео-ось (направление подачи), то подвод к точке повторного подвода в том случае, когда позиция инструмента и запрограммированная позиция в направлении подачи не совпадают, выполняется по спирали.

В следующих случаях происходит автоматическое переключение на линейный подвод REPOSL:

- Значение для DISR не указано.
- Определенного направления подвода не существует (прерывание программы в кадре без информации перемещения).
- При направлении подвода вертикально к актуальной рабочей плоскости.



4.14.11 Воздействие на управление движением

4.14.11.1 Коррекция максимальной скорости оси или частоты вращения шпинделя (VELOLIM)

С помощью команды VELOLIM в программе обработки детали можно уменьшить установленную через машинные данные макс. возможную скорость оси или макс. возможную, зависящую от ступени редуктора частоту вращения шпинделя.

Активность

VELOLIM действует:

- в автоматических режимах работы.
- на траекторные и позиционирующие оси.
- на шпиндели в шпиндельном/осевом режиме

Синтаксис

```
VELOLIM[<Ax>]=<Value>
```

Объяснение

VELOLIM:	Коррекция значения скорости или частоты вращения
<Ax>:	Ось или шпиндель, предельное значение скорости или частоты вращения которых необходимо настроить. Через MD30455 $MA_MISC_FUNCTION_MASK, бит 6 можно установить, действует ли VELOLIM независимо от текущего использования как шпиндель или ось (бит 6 = 1) или должна программироваться отдельно для каждого режима работы (бит 6 = 0). Если сконфигурировано раздельное действие, то выбор осуществляется через идентификатор при программировании: • Идентификатор шпинделя для режимов работы шпинделя: S<n> • Идентификатор оси для осевого режима, например, C
<Value>:	Процентная поправка В случае осей или шпинделей в осевом режиме с настройкой MD30455 бит 6 = 0 поправка относится к сконфигурированной максимальной скорости оси (MD32000 $MA_MAX_AX_VELO). В случае шпинделей в шпиндельном или осевом режиме с настройкой MD30455 бит 6 = 1 поправка относится к максимальной частоте вращения активной ступени редуктора (MD35130 $MA_GEAR_STEP_MAX_VELO_LIMIT[<n>]).
	Диапазон значений: 1 ... 100
	Значение 100 не вызывает управления скоростью или частотой вращения.

Примечание

Поведение при завершении программы обработки детали и сбросе канала

Поведение VELOLIM при завершении программы обработки детали и сбросе канала зависит от настройки бита 0 в машинных данных MD32320 $MA_DYN_LIMIT_RESET_MASK.

Примечание

Ограничение частоты вращения в шпиндельном режиме

Ограничение частоты вращения через VELOLIM (меньше 100 %) в шпиндельном режиме можно узнать через следующую системную переменную:

- $AC_SMAXVELO (макс. возможная частота вращения шпинделя)
- $AC_SMAXVELO_INFO (идентификатор для причины ограничения частоты вращения)

Примеры

Пример 1: Ограничение скорости оси станка

Программный код	Комментарий
...	
N70 VELOLIM[X]=80	; Осевые салазки в направлении X должны перемещаться только макс. с 80 % от допустимой для оси скорости.
...	

Пример 2: Ограничение частоты вращения, шпиндель 1 (AX5)

Конфигурация:

- MD35130 $MA_GEAR_STEP_MAX_VELO_LIMIT[1, AX5] = 1000
  (Максимальная частота вращения ступени редуктора 1 = 1000 об/мин)
- MD30455 $MA_MISC_FUNCTION_MASK[AX5], бит 6 = 1
  (Программирование VELOLIM действует совместно для шпиндельного и осевого режима независимо от запрограммированного идентификатора).

Программирование:

Программный код	Комментарий
`N05 VELOLIM[S1]=90`	; Ограничение максимальной частоты вращения шпинделя 1 до 90 % от 1000 об/мин.
...	
`N50 VELOLIM[C]=45`	; Ограничение максимальной частоты вращения шпинделя 1 до 45 % от 1000 об/мин, `C` означает идентификатор оси `S1`.
...	

4.14.11.2 Коррекция максимального рывка оси (JERKLIM)

С помощью команды JERKLIM установленный через машинные данные макс. возможный рывок оси при движении по траектории в критических сегментах программы может быть уменьшен или увеличен в программе обработки детали.

Условие

Режим ускорения SOFT должен быть активен.

Активность

JERKLIM действует:

- в автоматических режимах работы.
- только на траекторные оси.

Синтаксис

```
JERKLIM[<Ax>]=<Value>
```

Объяснение

`JERKLIM`:	Настроить предельное значение рывка
`<Ax>`:	Ось станка, предельное значение рывка которой необходимо настроить.

<значение>:	Процентная поправка Поправка относится к сконфигурированному макс. рывку оси при движении по траектории (MD32431 $MA_MAX_AX_JERK).	
	Диапазон значений:	1 ... 200
	Значение 100 не вызывает управления рывком.	

Примечание

Поведение при завершении программы обработки детали и сбросе канала

Поведение JERKLIM при завершении программы обработки детали и сбросе канала зависит от настройки бита 0 в машинных данных MD32320 $MA_DYN_LIMIT_RESET_MASK.

Пример

Программный код	Комментарий
`N1000 G0 X0 Y0 F10000 SOFT G64`	
`N1100 G1 X20 RNDM=5 ACC[X]=20 ACC[Y]=30`	
`N1200 G1 Y20 JERKLIM[Y]=200`	; Осевые салазки в направлении `Y` должны ускоряться/замедляться макс. с 200 % от допустимого для оси рывка.
`N1300 G1 X0 JERKLIM[X]=2`	; Осевые салазки в направлении X должны ускоряться/замедляться макс. с 2 % от допустимого для оси рывка.
`N1400 G1 Y0`	
`M30`	

4.14.11.3 Коррекция максимальной скорости движения по траектории (FLIM)

Функция "Коррекция максимальной скорости движения по траектории" позволяет ограничить или снизить на критических участках программы обработки детали скорость движения по траектории, результирующую из значений ограничения оси. Значение скорости, до которого необходимо ограничить максимальную скорость движения по траектории, программируется через адрес FLIM. Запрограммированное значение действует всегда только до ближайшего сброса ЧПУ или конца программы обработки детали.

Процентовка действует и далее на подачу по траектории, ограниченную посредством FLIM Поэтому при помощи переключателя процентовки можно умеренно превысить FLIM.

Активность

Функция действует:

- в автоматических режимах работы.
- только на траекторные оси.

- только для G94
- не на ускоренном ходу

Примечание

Функция "Коррекция максимальной скорости движения по траектории" действует только в сочетании с линейной подачей G94. В сочетании с подачей других типов (G93, G931, G95, G96, G97, G971, G972) эта функция неприменима.

Синтаксис

```
...
FLIM=<значение>
...
FLIM=-1
...
```

Объяснение

FLIM	Адрес для коррекции максимальной скорости движения по траектории	
<значение>	Значение скорости, на которое необходимо ограничить максимальную скорость движения по траектории	
	Тип данных:	REAL
	Диапазон значений:	$1{,}0 * 10^{-6}$ ... $1{,}0 * 10^{38}$
	Единица:	мм/мин или дюймов/мин (в зависимости от активной системы единиц измерения)
FLIM=-1	Снова снимает ограничение, запрограммированное посредством FLIM=<значение>.	

Пример

Программный код	Комментарий
...	
N1000 G0 X0 Y0 F10000 G64 G710	
N1100 G1 X20 RNDM=5	
N1200 G1 Y20 **FLIM=5000**	; Салазки оси в направлении Y: скорость движения макс. 5 м/мин.
N1300 G1 X0 Y40	; Движение салазок оси по траектории в направлении X и направлении Y: скорость движения макс. 5 м/мин.
N1400 G1 Y0 **FLIM=-1**	; Салазки оси в направлении Y: скорость движения 10 м/мин.
M30	

4.14.11.4 Коррекция максимального ускорения по траектории (PACCLIM)

Функция "Коррекция максимального ускорения по траектории" позволяет снизить на критических участках программы обработки детали ускорение движения по траектории, результирующее из значений ограничения оси. Значение ускорения, до которого необходимо снизить максимальное ускорение движения по траектории, программируется через адрес PACCLIM. Запрограммированное значение действует всегда только до ближайшего сброса ЧПУ или конца программы обработки детали.

Условия

Лицензирование

Функция является лицензируемой опцией ("Ограничение ускорения движения по траектории"; № артикула: 6FC5800-0xP26-0YB0); № артикула необходимо присвоить аппаратному обеспечению через менеджер лицензий.

Активность

Функция действует:

- в автоматических режимах работы.
- только на траекторные оси.

Синтаксис

```
...
PACCLIM=<значение>
...
PACCLIM=-1
...
```

Объяснение

PACCLIM	Адрес для коррекции максимального ускорения движения по траектории	
<значение>	Значение ускорения, до которого необходимо снизить максимальное ускорение движения по траектории	
	Тип данных:	REAL
	Диапазон значений:	$1{,}0 * 10^{-6}$ ... $1{,}0 * 10^{38}$
	Единица:	м/сек^2 или дюймов/сек^2 (в зависимости от активной системы единиц измерения)
PACCLIM=-1	Снова снимает ограничение, запрограммированное посредством PACCLIM=<значение>.	

Примечание

По своему действию PACCLIM схожа с установочными данными SD42500 $SC_SD_MAX_PATH_ACCEL (максимальное ускорение движения по траектории). Однако в отличие от SD42500 PACCLIM работает синхронно в кадре.

Примечание

При вычислении значения ограничения значение SD42500 $SC_SD_MAX_PATH_ACCEL учитывается, только если SD42502 $SC_IS_SD_MAX_PATH_ACCEL установлено на TRUE.

Если активны и PACCLIM, и SD42500 $SC_SD_MAX_PATH_ACCEL, то действует меньшее из двух значений ограничения.

Пример

Программный код	Комментарий
...	
N1000 G0 X0 Y0 F10000 G64 G710	
N1100 G1 X20 RNDM=5	
N1200 G1 Y20 **PACCLIM=0.5**	; Салазки оси в направлении Y: ускорение/замедление движения макс. 0,5 м/сек.
N1300 G1 X0 Y40	; Движение салазок оси по траектории в направлении X и направлении Y: ускорение/замедление движения макс. 0,5 м/сек.
N1400 G1 Y0	
M30	

4.14.12 Параметры смены кадров при активном соединении (CPBC)

С помощью команды CPBC задается критерий смены кадров, который должен быть выполнен, чтобы в программе обработки детали при активном соединении была выполнена смена кадров.

Синтаксис

CPBC[<ведомая ось>] = <критерий>

Объяснение

CPBC:	Критерий смены кадров при активном соединении
<ведомая ось>:	идентификатор ведомой оси

<критерий>:	критерий смены кадра	
	Тип:	STRING
	Значение	**Значение:** смена кадра выполняется
	"NOC"	независимо от состояния соединения
	"IPOSTOP"	при синхронном ходе со стороны заданного значения
	"COARSE"	при синхронном ходе со стороны фактического значения "грубом"
	"FINE"	при синхронном ходе со стороны фактического значения "точном"

Пример

Программный код
; смена кадра выполняется при:
; - соединение с ведомой осью `X2 ==` активно
; - синхронный ход со стороны заданного значения == активен
`CPBC[X2]="IPOSTOP"`

4.15 Функции осей

4.15.1 Переход оси, переход шпинделя (RELEASE, GET, GETD)

Интерполяция одной или нескольких осей или шпинделей возможна только в одном канале. Если необходима попеременная работа одной оси в двух различных каналах (к примеру, устройство автоматической смены палет), то она сначала должна быть разрешена в актуальном канале и после принята в другой канал. Ось переходит между каналами.

Расширения перехода оси

Возможен переход оси/шпинделя с остановкой предварительной обработки и синхронизацией между предварительной обработкой и главным ходом или, в качестве альтернативы, и без остановки предварительной обработки. Кроме этого, переход оси возможен и через:

- Вращение осевого контейнера AXCTSWE или AXCTWED посредством не явной `GET/GETD`.
- Фрейм с вращением, если посредством этого эта ось соединена с другими осями.
- Синхронные действия, см. Синхронные действия движения, "Переход оси `RELEASE`, `GET`".

Изготовитель станка

Следовать указаниям изготовителя станка. Через конфигурируемые машинные данные одна ось должна быть однозначно определена для перехода оси во всех каналах и изменение поведения перехода оси должно быть возможным через машинные данные.

Синтаксис

RELEASE(имя оси, имя оси, ...) или RELEASE(S1)

GET(имя оси, имя оси, ...) или GET(S2)

GETD(имя оси, имя оси, ...) или GETD(S3)

С помощью GETD (GET Directly) ось напрямую забирается из другого канала. Это означает, что для этой GETD не требуется программирования подходящей RELEASE в другом канале. Но это означает и установку иной коммуникации каналов (к примеру, метки ожидания).

Объяснение

RELEASE (имя оси, имя оси, ...):	Разрешение оси(ей)
GET (имя оси, имя оси, ...):	Получение оси(ей)
GETD (имя оси, имя оси, ...):	Прямое получение оси(ей)
Имя оси:	Согласование оси в системе: AX1, AX2, ... или указание имен осей станка
RELEASE(S1) :	Разрешение шпинделей S1, S2, ...
GET(S2) :	Получение шпинделей S1, S2, ...
GETD(S3) :	Прямое получение шпинделей S1, S2, ...

Требование GET без остановки предварительной обработки

Если после требования GET **без** остановки предварительной обработки ось снова разрешается с RELEASE(ось) или WAITP(ось), то последующая GET приводит к GET**с** остановкой предварительной обработки.

⚠ ОСТОРОЖНО

Согласование осей изменено

Полученная с GET ось или шпиндель остается согласованной с этим каналом и после клавишного или программного RESET.

При новом старте программы необходимо программно-техническое согласование переданных осей или шпинделей, если ось необходима в своем базовом канале.

При POWER ON она согласуется с зафиксированным в машинных данных каналом.

Примеры

Пример 1: Переход оси между двумя каналами

Из 6 осей в канале 1 для обработки используются: 1-я, 2-я, 3-я и 4-я ось.
5-я и 6-я ось используются в канале 2 для смены детали.

Необходим переход оси 2 между двумя каналами и ее согласование после POWER ON с каналом 1.

Программа "MAIN" в канале 1:

Программный код	Комментарий
`INIT (2,"TAUSCH2")`	; Выбрать программу `TAUSH2` в канале 2.
`N… START (2)`	; Запустить программу в канале 2.
`N… GET (AX2)`	; Принять ось `AX2`.
...	
`N… RELEASE(AX2)`	; Разрешить ось `AX2`.
`N… WAITM (1,1,2)`	; Ожидание метки `WAIT` в канале 1 и в канале 2 для синхронизации в обоих каналах.
...	; Продолжение процесса после перехода оси.
`N… M30`	

Программа "TAUSCH2" в канале 2:

Программирование	Комментарий
`N… RELEASE(AX2)`	
`N160 WAITM(1,1,2)`	; Ожидание метки `WAIT` в канале 1 и в канале 2 для синхронизации в обоих каналах.
`N150 GET(AX2)`	; Принять ось `AX2`.
...	; Продолжение процесса после перехода оси.
`N… M30`	

Пример 2: Переход оси без синхронизации

Если синхронизация оси не требуется, то GET не создает остановки предварительной обработки.

Программирование	Комментарий
`N01 G0 X0`	
`N02 RELEASE(AX5)`	
`N03 G64 X10`	
`N04 X20`	
`N05 GET(AX5)`	; Если синхронизация не нужна, то это не исполняемый кадр.
`N06 G01 F5000`	; Не исполняемый кадр.
`N07 X20`	; Не исполняемый кадр, так как позиция X как в `N04`.
`N08 X30`	; Первый исполняемый кадр после `N05`.
...	

Пример 3: Активация перехода оси без остановки предварительной обработки

Условие: Переход оси без остановки предварительной обработки должен быть сконфигурирован через машинные данные.

Программирование	Комментарий
`N010 M4 S100`	

Программирование	Комментарий
N011 G4 F2	
N020 M5	
N021 SPOS=0	
N022 POS[B]=1	
N023 WAITP(B)	; Ось B становится нейтральной осью.
N030 X1 F10	
N031 X100 F500	
N032 X200	
N040 M3 S500	; Ось не инициирует остановку предварительной обработки/REORG.
N041 G4 F2	
N050 M5	
N099 M30	

Если шпиндель или ось B непосредственно после кадра N023 перемещается как **ось PLC**, к примеру, на 180 градусов и назад на 1 градус, то эта ось снова становится нейтральной осью и не вызывает остановки предварительной обработки в кадре N40.

Дополнительная информация

Условия для перехода оси

- Ось через машинные данные должна быть определена во всех каналах, которые хотят ее использовать.
- Через спец. для **оси** машинные данные необходимо установить, с каким каналом ось должна быть согласована после POWER ON.

Описание

Разрешить ось: RELEASE

При разрешении оси учитывать:

1. Ось не должна участвовать в трансформации.
2. В случае соединений осей (тангенциальное управление), все оси структуры должны быть разрешены.
3. Переход конкурирующей позиционирующей оси в этом состоянии невозможен.
4. В случае мастер-оси Gantry переходят и все ведомые оси.
5. В соединениях осей (буксировка, соединение по главному значению, электронный редуктор) возможно разрешение только ведущей оси соединения.

Передача оси: GET

С помощью этой команды запускается сам процесс перехода оси. Ответственность за ось полностью лежит на канале, в котором была запрограммирована команда.

Действие GET:

Переход оси с синхронизацией:

Синхронизаций оси всегда необходима в тех случаях, когда она была промежуточно согласована в другом канале или с PLC, и перед GET не была выполнена синхронизация через "WAITP", G74 или стирание остатка пути.

- Выполняется остановка предварительной обработки (как при STOPRE).
- Обработка прерывается до тех пор, пока переход не будет выполнен полностью.

Автоматическая "GET"

Если ось принципиально имеется в канале, но не доступна в настоящий момент как "Ось канала", то автоматически выполняется GET. Если ось(и) уже синхронизированы, то остановка предварительной обработки не создается.

Возможность изменения поведения перехода оси

Момент передачи осей может быть установлен через машинные данные следующим образом:

- Автоматический переход оси выполняется между двумя каналами и тогда, когда ось через WAITP была переведена в нейтральное состояние (прежнее поведение)
- При требовании вращения осевого контейнера все согласованные с исполняемым каналом оси осевого контейнера посредством не явной GET или GETD передаются в канал. Последующий переход оси снова разрешается только после завершения вращения осевого контейнера.
- После вставленного промежуточного кадра в главном ходе проверяется, необходима ли реорганизация или нет. Только если состояния осей этого кадра **не** совпадают с актуальными состояниями осей, то необходима реорганизация.
- Вместо кадра GET с остановкой предварительной обработки и синхронизацией между предварительной обработкой и главным ходом, возможет и переход оси и без остановки предварительной обработки. В этом случае создается только один промежуточный кадр с требованием GET. В главном ходе при выполнении этого кадра проверяется, совпадают ли состояния оси в кадре с актуальными состояниями оси.

4.15.2 Передача оси в другой канал (AXTOCHAN)

С помощью языковой команды `AXTOCHAN` ось может быть запрошена для передачи ее в другой канал. Ось может быть переведена в соответствующий канал как из программы обработки детали ЧПУ, так и из синхронного действия.

Синтаксис

`AXTOCHAN(`имя оси,номер канала[,имя оси,номер канала[,...]])

Объяснение

Элемент	Описание
AXTOCHAN:	Запросить ось для определенного канала
Имя оси:	Согласование оси в системе: X, Y, O или указание участвующих имен осей станка. Исполняемый канал не должен быть собственным каналом и это также не должен быть канал, который в настоящий момент имеет право интерполяции для оси
Номер канала:	Номер канала, с которым должна быть согласована ось

Примечание

Конкурирующая позиционирующая ось и контролируемая исключительно PLC ось

Ось PLC как конкурирующая позиционирующая ось не может менять канал. Контролируемая исключительно с PLC ось не может быть согласована с программой ЧПУ.

Дополнительная информация: Описание функций – Оси и шпиндели

Пример

AXTOCHAN в программе ЧПУ

Оси X и Y известны в 1-м канале и во 2-м канале. В настоящий момент канал 1 имеет право интерполяции и в канале 1 запускается следующая программа:

Программный код	Комментарий
`N110 AXTOCHAN(Y,2)`	; Переместить ось `Y` во 2-й канал.
`N111 M0`	
`N120 AXTOCHAN(Y,1)`	; Снова получить ось `Y` (нейтральная).
`N121 M0`	
`N130 AXTOCHAN(Y,2,X,2)`	; Переместить ось `Y` и ось X во 2-й канал (нейтральные оси).
`N131 M0`	
`N140 AXTOCHAN(Y,2)`	; Переместить ось `Y` во 2-й канал (программа ЧПУ).
`N141 M0`	

Дополнительная информация

AXTOCHAN в программе ЧПУ

При этом только при запросе оси для программы ЧПУ в собственном канале выполняется GET и тем самым происходит ожидание фактического изменения состояния. Если ось запрашивается для другого канала и если она должна стать нейтральной осью в собственном канале, только тогда запрос соответственно устанавливается.

AXTOCHAN из синхронного действия

Если ось запрашивается для собственного канала, то AXTOCHAN из синхронного действия отображается на GET из синхронного действия. В этом случае ось при первом запросе

для собственного канала становится нейтральной осью. При втором запросе ось согласуется с программой ЧПУ аналогично запросу GET в программе ЧПУ. Касательно запроса GET из синхронного действия см. главу "Синхронные действия движения".

4.15.3 Осевые функции (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)

"AXNAME" используется, к примеру, при создании общих циклов, когда имена осей неизвестны.

"AX" используется для косвенного программирования геометрических и синхронных осей. При этом идентификатор оси фиксируется в переменной типа AXIS или выводится командой, к примеру, "AXNAME" или "SPI".

"SPI" используется, когда осевые функции программируются для шпинделя, к примеру, синхронного шпинделя.

"AXTOSPI" используется для преобразования идентификатора оси в индекс шпинделя (обратная функция к "SPI").

"AXSTRING" используется для преобразования идентификатора оси (тип данных AXIS) в строку (обратная функция к "AXNAME").

"ISAXIS" используется в общих циклах, чтобы убедиться, что имеется определенная гео-ось и тем самым последующий вызов $P_AXNX не будет отменен с ошибкой.

"MODAXVAL" для определения позиции модуло для круговых осей модуло.

Синтаксис

```
AXNAME("строка")
AX[AXNAME("строка")]
SPI(n)

AXTOSPI(A) или AXTOSPI(B) или AXTOSPI(C)
AXSTRING(SPI(n))
ISAXIS(<номер гео-оси>)
<позиция модуло>=MODAXVAL(<ось>,<позиция оси>)
```

Объяснение

AXNAME:	Преобразует входную строку в идентификатор оси; входная строка должна содержать действительное имя оси.
AX:	Переменный идентификатор оси
SPI:	Преобразует номер шпинделя в идентификатор оси; передаваемый параметр должен содержать действительный номер шпинделя.
n:	Номер шпинделя
AXTOSPI:	Преобразует идентификатор оси в индекс шпинделя типа Integer. "AXTOSPI" соответствует обратной функции к "SPI".
X, Y, Z:	Идентификатор оси типа AXIS как переменная или постоянная
AXSTRING:	Выводится строка с согласованным номером шпинделя.

ISAXIS:	Проверяет наличие указанной гео-оси.
MODAXVAL:	Определяет для круговых осей модуло позицию модуло; она соответствует остатку модуло относительно спараметрированного диапазона модуло (составляет в стандартной установке 0 до 360 градусов; через MD30340 MODULO_RANGE_START и MD30330 $MA_MODULO_RANGE можно изменять начало и размер диапазона модуло).

Примечание

Расширения SPI

Осевая функция SPI(n) может использоваться и для чтения и записи компонентов фрейма. С ее помощью могут записываться фреймы, к примеру, с синтаксисом, `$P_PFRAME[SPI(1),TR]=2.22`.

Посредством дополнительного программирования позиции оси через адрес `AX[SPI(1)] =` <позиция оси> ось может перемещаться. Условием этого является нахождение шпинделя в режиме позиционирования или осевом режиме.

Примеры

Пример 1: AXNAME, AX, ISAXIS

Программный код	Комментарий
`OVRA[AXNAME("поперечная ось")]=10`	; Процентовка для поперечной оси
`AX[AXNAME("поперечная ось")]=50.2`	; Конечная позиция для поперечной оси
`OVRA[SPI(1)]=70`	; Процентовка для шпинделя 1
`AX[SPI(1)]=180`	; Конечная позиция для шпинделя 1
`IF ISAXIS(1)==FALSE GOTOF WEITER`	; Имеется ли абсцисса?
`AX[$P_AXN1]=100`	; Перемещение по абсциссе
`WEITER:`	

Пример 2: AXSTRING

При программировании с AXSTRING[SPI(n)] более в качестве номера шпинделя не выводится индекс оси, с которой согласован шпиндель, а выводится строка "Sn".

Программный код	Комментарий
`AXSTRING[SPI(2)]`	; Выводится строка `"S2"`.

Пример 3: MODAXVAL

Необходимо определить позицию модуло круговой оси модуло A.

Исходным значением для вычисления является позиция оси 372.55.

Спараметрированный диапазон модуло составляет 0 до 360 градусов:

MD30340 MODULO_RANGE_START = 0

MD30330 $MA_MODULO_RANGE = 360

Программный код	Комментарий
`R10=MODAXVAL(A,372.55)`	; Вычисленная позиция модуло `R10 = 12.55.`

Пример 4: MODAXVAL

Если запрограммированный идентификатор оси относится не к круговой оси модуло, то преобразуемое значение (<позиция оси>) возвращается без изменений.

Программный код	Комментарий
`R11=MODAXVAL(X,372.55)`	`; X` это линейная ось; `R11 = 372.55.`

4.15.4 Переключаемые геометрические оси (GEOAX)

С помощью функции "Переключаемые гео-оси" можно заменять геометрические оси, сконфигурированные через машинные данные, на оси других каналов.

Синтаксис

`GEOAX(`<n>`,`<ось канала>`,`<n>`,`<ось канала>`,`<n>`,`<ось канала>`)`
`GEOAX( )`

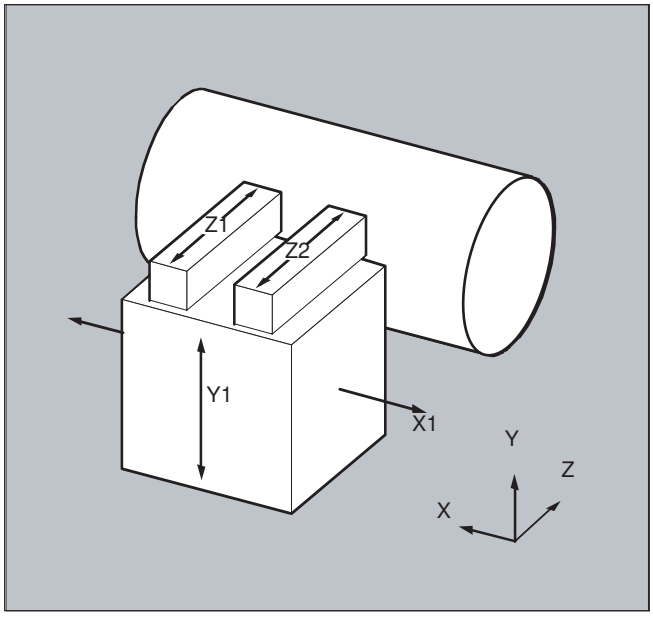
Объяснение

`GEOAX(...)`	Функция для переключения геометрических осей. **Примечание:** `GEOAX()` без указания параметра повторно активирует базовую конфигурацию гео-осей, спараметрированную в машинных данных.
<n>	Номер геометрической оси, которая заменяется на указанную ось канала. Диапазон значений: 0, 1, 2, 3 **Примечание:** 0: Указанная ось канала без замены удаляется из соединения геометрических осей 1: 1-я геометрическая ось ≙ ось координат X (абсцисса) системы WCS 2: 2-я геометрическая ось ≙ ось координат Y (ордината) системы WCS 3: 3-я геометрическая ось ≙ ось координат Z (аппликата) системы WCS
<ось канала>	Имя оси канала, которая должна быть включена в соединение гео-осей.

Примеры

Пример 1: Попеременное включение двух осей как гео-оси

Инструментальная каретка может перемещаться через оси канала X1, Y1, Z1, Z2:



Гео-оси сконфигурированы таким образом, что после включения сначала Z1 действует как 3-ья гео-ось под именем гео-оси "Z" и образует вместе с X1 и Y1 соединение гео-осей.

В программе обработки детали оси Z1 и Z2 теперь должны попеременно работать в качестве гео-оси Z:

Программный код	Комментарий
...	
N100 GEOAX(3,Z2)	; Как 3-я гео-ось (Z) используется ось канала Z2.
N110 G1 ...	
N120 GEOAX(3,Z1)	; Как 3-я гео-ось (Z) используется ось канала Z1.
...	

Пример 2: Переключение гео-осей при 6 осях канала

Станок имеет 6 осей канала с именами XX, YY, ZZ, U, V, W.

Первичная установка конфигурации гео-осей через машинные данные это:

Ось канала XX = 1-я гео-ось (ось X)

Ось канала YY = 2-я гео-ось (ось Y)

Ось канала ZZ = 3-я гео-ось (ось Z)

Программный код	Комментарий
N10 GEOAX()	; Базовая конфигурация гео-осей активна.
N20 G0 X0 Y0 Z0 U0 V0 W0	; Все оси ускоренным ходом на позицию 0.
N30 GEOAX(1,U,2,V,3,W)	; Ось канала U становится первой (X), V – второй (Y)
	; и W – третьей гео-осью (Z).
N40 GEOAX(1,XX,3,ZZ)	; Ось канала XX становится первой (X), ZZ – третьей
	; гео-осью (Z). Ось канала V остается второй
	; гео-осью (Y).

Программный код	Комментарий
N50 G17 G2 X20 I10 F1000	; Полный круг в плоскости X/Y. Двигаются
	; оси канала XX и V.
N60 GEOAX(2,W)	; Ось канала W становится второй гео-осью (Y).
N80 G17 G2 X20 I10 F1000	; Полный круг в плоскости X/Y. Двигаются
	; оси канала XX и W.
N90 GEOAX()	; Сброс на исходное состояние.
N100 GEOAX(1,U,2,V,3,W)	; Ось канала U становится первой (X), V – второй
	; (Y) и W – третьей гео-осью (Z).
N110 G1 X10 Y10 Z10 XX=25	; Оси канала U, V, W движутся на
	; позицию 10. XX как дополнительная ось движется на позицию 25.
N120 GEOAX(0,V)	; V изымается из соединения гео-осей.
	; U и W остаются первой (X) и третьей
	; гео-осями (Z).
	; Вторая гео-ось (Y) остается свободной.
N130 GEOAX(1,U,2,V,3,W)	; Ось канала U остается первой (X), V становится
	; второй (Y), W остается третьей гео-осью (Z).
N140 GEOAX(3,V)	; V становится третьей гео-осью (Z), при этом W
	; переписывается и тем самым изымается из
	; соединения гео-осей. Вторая гео-ось (Y),
	; как и прежде, свободна.

Машинные данные

Конфигурация осей

Согласование геометрических осей, дополнительных осей и осей станка с осями каналов:

- MD10000 $MN_AXCONF_MACHAX_NAME_TAB
- MD20050 $MC_AXCONF_GEOAX_ASIGN_TAB
- MD20060 $MC_AXCONF_GEOAX_NAME_TAB
- MD20070 $MC_AXCONF_MACHAX_USED
- MD20080 $MC_AXCONF_CHANAX_NAME_TAB
- MD35000 $MA_SPIND_ASSIGN_TO_MACHAX

Поведение при Reset

Поведение измененных согласований геометрических осей при Reset:

- MD20110 $MC_RESET_MODE_MASK, бит 12
- MD20118 $MC_GEOAX_CHANGE_RESET

Поведение при запуске ЧПУ

- MD20112 $MC_START_MODE_MASK, бит 12

Сообщение для программы электроавтоматики

Возможность параметрирования M-команды, которая при переключении геометрических осей выдается на интерфейс ЧПУ/PLC:

- MD22532 $MC_GEOAX_CHANGE_M_CODE

Граничные условия

Нет переключения геометрических осей

- Если активна одна из следующих функций, переключение геометрических осей невозможно:
  - Трансформация
  - Сплайн-интерполяция
  - Коррекция на радиус инструмента
  - Точная коррекция на инструмент
- Геометрическая ось и дополнительная ось канала имеют одно и то же имя.
- Одна из осей, участвующих в переключении геометрических осей, участвует в действии, которое продолжается за границы кадра. Например, выходящая за границы кадра позиционирующая или ведомая ось соединения осей.

Круговые оси

Круговые оси нельзя сделать геометрическими осями.

Состояние оси после замены

Замененная через переключение в соединении гео-осей ось после процесса переключения через своё имя оси канала может быть запрограммирована как дополнительная ось.

Фреймы, защищенные области, ограничения рабочего поля

При переключении гео-осей все фреймы, защищенные области и ограничения рабочего поля удаляются.

Полярные координаты

Смена геометрических осей с помощью `GEOAX` аналогично смене плоскостей через `G17-G19` устанавливает модальные полярные координаты на значение 0.

DRF, NPV

Возможное смещение маховичком (DRF) или внешнее смещение нулевой точки (WO) продолжает действовать после переключения.

Базовая конфигурация гео-осей

Команда `GEOAX()` вызывает базовую конфигурацию соединения геометрических осей.

После POWER ON и при переключении в режим работы "Реферирование" выполняется автоматический возврат к базовой конфигурации.

Коррекция длин инструмента

Активная коррекция длин инструмента действует и после процесса переключения. Но для новых принятых или изменивших позиции гео-осей она считается еще не выведенной. При первой команде движения для этих гео-осей результирующий путь перемещения состоит соответственно из суммы коррекции длин инструмента и запрограммированного пути перемещения.

Гео-оси, сохранившие при переключении свою позицию в структуре осей, сохраняют и свой статус касательно коррекции длин инструмента.

Конфигурация гео-осей при активной трансформации

- Конфигурацию гео-осей, спараметрированную для активной трансформации через машинные данные трансформации, изменить с помощью функции "Переключаемые гео-оси" невозможно.
- Различная конфигурация гео-осей для трансформации, в машинных данных трансформации нужно спараметрировать разные блоки данных.
- Измененная через `GEOAX` конфигурация гео-осей при активации трансформации удаляется.
- В отношении гео-осей варианты их параметрирования для конкретных трансформаций при активных трансформациях имеют приоритет перед вариантами параметрирования, связанными с переключением гео-осей.
  Пример: Одна из трансформаций активна. Согласно машинным данным эта трансформация при Reset канала должна сохраниться. Но одновременно при Reset канала должна восстановиться базовая конфигурация гео-осей. Сохраняется конфигурация гео-осей, которая задана для трансформации.
- С выключением трансформации спараметрированная первичная установка конфигурации гео-осей становится снова активной.

Режим работы JOG, функция станка REF

При переключении на режим работы JOG, функция станка REF (реферирование), спараметрированная в машинных данных конфигурация гео-осей становится активной.

4.15.5 Осевой контейнер (AXCTSWE, AXCTSWED, AXCTSWEC)

Через команды "AXCTSWE" или "AXCTSWED" разрешается вращение указанного осевого контейнера.

С помощью команды "AXCTSWEC" уже данное разрешение на вращение осевого контейнера может быть снова отменено.

Синтаксис

```
AXCTSWE(<ID>)
AXCTSWED(<ID>)
AXCTSWEC(<ID>)
```

Объяснение

AXCTSWE:	Разрешение на вращение осевого контейнера Обработка программы через "AXCTSWE" не останавливается. Вращение выполняется после того, как все участвующие в осевом контейнере каналы дали разрешение.	
AXCTSWED:	Разрешение на вращение осевого контейнера без учета других участвующих в контейнере каналов **Указание** • Вариант команды специально для упрощения ввода в эксплуатацию программы обработки детали или синхронного действия. • Поведение касательно других участвующих в осевом контейнере каналов может быть задано через: MD12760 $MN_ AXCT_FUNCTION_MASK, Bit 0	
AXCTSWEC:	Отмена разрешения на вращение осевого контейнера **Указание** Разрешение вращения осевого контейнера может быть отменено, только если вращение еще не началось: $AN_AXCTSWA[<осевой контейнер>] == 0 По системным переменным см. "Осевой контейнер (AXCTSWE, AXCTSWED, AXCTSWEC) (Страница 920)"	
<ID>:	Идентификатор осевого контейнера или оси контейнера:	
	CT<номер>:	Идентификатор по умолчанию осевого контейнера: MD12750 $MN_AXCT_NAME_TAB Пример: "CT1"
	<контейнер>:	Спец. для пользователя идентификатор осевого контейнера: MD12750 $MN_AXCT_NAME_TAB Пример: "CONTAINER_1"
	<ось>:	Идентификатор известной в канале оси контейнера

Примечание

Размер шага

Размер шага вращения осевого контейнера задается через установочные данные:

SD41700 $SN_AXCT_SWWIDTH

Дополнительная информация

Диагностика

Актуальное состояние осевого контейнера может быть считано через следующие системные переменные:

Системная переменная	Тип	Описание
$AC_AXCTSWA[<имя>]	BOOL	Спец. для канала состояние осевого контейнера
$AN_AXCTSWA[<осевой контейнер>]	BOOL	Спец. для NCU состояние осевого контейнера

Системная переменная	Тип	Описание
$AN_AXCTSWE[<осевой контейнер>]	INT	Спец. для слота состояние вращения осевого контейнера Системная переменная выводит **по битам** состояние слотов осевого контейнера. Каждый бит соответствует одному слоту.
$AN_AXCTAS[<осевой контейнер>]	INT	Число мест (слотов), на которое выполнено текущее продвижение осевого контейнера.

Вращение осевого контейнера с не явной GET / GETD

Через следующие машинные данные можно установить, что при команде "AXCTSWE" все оси контейнера канала посредством не явной "GET / GETD" будут переданы в канал. Переход оси снова возможен только после завершения вращения контейнера.

MD10722 $MN_AXCHANGE_MASK, бит 1 = 1

Примечание

Вращение осевого контейнера с не явной "GET / GETD" **не** выполняется для оси в состоянии "Ось главного хода" (к примеру, PLC-ось), т.к. ось для вращения осевого контейнера должна выйти из этого состояния.

4.15.6 Ожидание действительной позиции оси (WAITENC)

С помощью языковой команды "WAITENC" в программе ЧПУ можно подождать, пока для сконфигурированных с MD34800 $MA_WAIT_ENC_VALID = 1 осей не будут доступны синхронизированные или восстановленные позиции осей.

В состоянии ожидания возможно прерывание, к примеру, через запуск ASUP или через смену режимов работы на JOG. При продолжении программы состояние ожидания при необходимости возобновляется.

Примечание

Состояние ожидания отображается на интерфейсе через режим приостановки "Ожидание измерительной системы".

Синтаксис

"WAITENC" может быть запрограммирована в программном блоке любой программы ЧПУ.

Программирование должно выполняться в отдельном кадре:

```
...
WAITENC
...
```

Пример

"WAITENC" используется, к примеру, в управляемой событием программе пользователя .../_N_CMA_DIR/_N_PROG_EVENT_SPF, как показывает следующий пример применения.

Пример использования: Отвод инструмента после POWER OFF с трансформацией ориентации

Обработка с ориентацией инструмента была отменена через прерывание подачи питания.
При последующем запуске вызывается управляемая событиями программа пользователя.../_N_CMA_DIR/_N_PROG_EVENT_SPF.

В управляемой событиями программе пользователя с "WAITENC" выполняется ожидание синхронизированных или восстановленных позиций осей, чтобы после можно было бы вычислить фрейм, который точно устанавливает WCS в направлении инструмента.

Программный код	Комментарий
...	
IF $P_PROG_EVENT == 4	; Запуск.
IF $P_TRAFO <> 0	; Трансформация была выбрана.
WAITENC	; Ожидать действительных осевых позиций осей ориентации.
TOROTZ	; Повернуть Z-ось WCS в направлении оси инструмента.
ENDIF	
M17	
ENDIF	
...	

После этого возможен свободный ход инструмента в режиме работы JOG через движение отвода в направлении оси инструмента.

4.15.7 Программируемое переключение блоков параметров (SCPARA)

С помощью команды SCPARA можно запросить для оси переключение на определенный блок параметров.

Примечание

Без переключения блоков параметров при обработке резьбы

При резьбонарезании G33 и нарезании внутренней резьбы G331 / G332 блок параметров выбирается СЧПУ и не может быть изменен.

Заблокированное переключение блоков параметров

Переключение блоков параметров может быть запрошено и через интерфейс ЧПУ/PLC. Для недопущения конфликтов переключения, через интерфейс ЧПУ/PLC можно заблокировать переключение блоков параметров ЧПУ (SCPARA):

DB31, ... DBX9.3 (установка блока параметров через ЧПУ заблокирована)

Примечание

Если переключение блоков параметров запрашивается через SCPARA, в то время как переключение блоков параметров через интерфейс ЧПУ/PLC заблокировано, то переключение отклоняется без сообщения об ошибке.

Синтаксис

SCPARA[<ось>]=<значение>

Объяснение

SCPARA:	Команда: Переключить блок параметров	
<ось>:	Идентификатор оси (ось канала)	
	Тип:	AXIS
<значение>:	Номер блока параметров: 1, 2, 3, ... макс. номер блока параметров	

Пример

Программный код	Комментарий
...	
N110 SCPARA[X] = 3	; Выбор: ось X, 3-й блок параметров
...	

Дополнительная информация

Разрешение переключения блоков параметров

Переключение блоков параметров оси должно быть разрешено явно:

MD35590 $MA_PARAMSET_CHANGE_ENABLE[<ось>]

Чтение номера блока параметров

Номер выбранного блока параметров (заданный блок параметров) может быть считан через системную переменную $AA_SCPAR.

4.15.8 Включение/выключение адаптации (CADAPTON, CADAPTOF)

Предварительно определенные процедуры CADAPTON() и CADAPTOF() позволяют активировать, обновлять и деактивировать предварительно определенные адаптации для коррекции параметров динамики или регулирования из программы обработки детали. Они используются и CYCLE782 (Страница 1137).

Синтаксис

```
...
CADAPTON(<Result>,<Axis>,<InVar>[,<InVal>])
...
CADAPTOF(<Result>,<Axis>,<InVar>)
...
```

Значение

CADAPTON():	Активировать отношение адаптации		
CADAPTOF():	Деактивировать отношение адаптации **Указание:** Игнорировать постоянно действующие отношения адаптации (MD16501 = 1) CADAPTOF(), входное значение, при необходимости запрограммированное ранее, (<InVal>) сохраняет действительность.		
<Result>:	Переменная результата: Возвращаемое значение для состояния (параметр "Call by reference")		
	Тип данных:	INT	
	Значение:	0	нет ошибок
		1	действительная таблица адаптаций не спараметрирована
		2	Параметр <ось> недействительный
		3	Параметр <InVar> недействительный
		4	зарезервировано
		5	Параметр <InVal> недействительный
<Axis>:	Имя входной оси станка (отношение адаптации)		
	Тип данных:	AXIS	
	Диапазон значений:	Определенные в канале имена осей станка	
	Примечание: Этот параметр выполняет адрессацию адаптаций, которые в MD16504 $MN_CADAPT_INPUT_AX внесли соответствующее параметру <ось> значение. Другие параметры <InVar> и <InVal> согласованы с этой осью.		
<InVar>:	Входная величина отношения адаптации		
	Тип данных:	INT	
	Значение:	1	Инерция оси
		2	Позиция оси
		3	Скорость оси

<InVal>:	Входное значение отношения адаптации Опциональный параметр для интеллектуальной коррекции нагрузки (<InVar>=1).		
	Тип данных:	REAL	
	Значение:	> 0	Текущая инерция
		=0	Нагрузка/загрузка не определена или неизвестна. В этом случае согласованные адаптации возвращают в качестве исходной величины заменяющее значение 1.0.
		< 0	Вызов завершается с <состояние> = 5 (параметр <InVal> недействительный). Последнее активное входное значение остается активным.
	Примечание: Без программирования входного значения действительным становится последнее запрограммированное значение или (после запуска СЧПУ) стандартное значение (= 0).		

Пример

Программный код	Комментарий
`DEF INT RESULT`	; Определение переменной результата
...	
`CADAPTON(RESULT,MX1,2)`	; Активация адаптации
`IF RESULT <> 0`	; Обработка переменных результата (требуется после каждого оператора `CADAPTON/CADAPTOF`)
`  MSG("CADAPT-RESULT=" << RESULT)`	
`  STOPRE`	
`  SETAL(61000)`	
`ENDIF`	
...	
`CADAPTOF(RESULT,MX1,2)`	; Деактивация адаптации
...	; Обработка переменной результата
...	

Дополнительная информация

Link-оси

Эта функция не может применяться к Link-осям (функция NCU-Link). В этом случае переменная результата возвращала бы возвращаемое значение 2 ("Параметр <ось> недействительный").

Поиск кадра

- **Поиск кадра без вычисления**
  Операторы CADAPTON/CADAPTOF, запрограммированные между началом программы и конечным кадром, игнорируются. Пользователь должен так выбирать цель поиска, чтобы выбирались значимые для этапа обработки адаптации.
- **Поиск кадра с вычислением на контуре или на конечной точке кадра**
  Команды активации или деактивации, а также возможные запрограммированные входные значения собираются, и при помощи кадров действий поиска выводится текущее состояние в конце поиска.
- **Поиск с вычислением в режиме "Тестирование программы (SERUPRO)**
  Выключение SERUPRO переводит состояние адаптаций, достигнутое посредством операторов CADAPTON/CADAPTOF, в реальную эксплуатацию.

4.15.9 Коррекция FIR-фильтра для динамического режима (CALCFIR)

Чтобы после смены динамического режима достигнуть для всех осей в канале, для которых активен FIR-фильтр нижних частот, идентичного глушения и точности контура, можно языковой командой CALCFIR ЧПУ вычислить и активировать настройки FIR-фильтра, зависящие от динамики.

Использование

CALCFIR используется в цикле изготовителя CUST_832.

Стандартное выполнение

В начале программы ЧПУ для обработки поверхности произвольной формы программируется CYCLE832. При выполнении программы CYCLE832 вызывает цикл изготовителя CUST_832, и выполняется языковая команда CALCFIR ЧПУ, добавленная изготовителем станка в CUST_832.

Активность

Перезаписанные при помощи CALCFIR настройки фильтра действуют до следующей перезаписи посредством вызова CALCFIR. Это необходимо всегда, когда при помощи команды G-группы 59 (динамический режим для интерполяции траектории) запрограммировано изменение текущего динамического режима.

Поведение при запуске СЧПУ / сбросе канала / сбросе конца программы

При запуске СЧПУ, сбросе канала, сбросе конца программы автоматически активируются настройки FIR-фильтра, подходящие для положение удаления G-группы 59.

Условия

- Опция "Top Speed" установлена.

  Примечание

  Для получения оптимальных результатов при опции "Top Speed" рекомендуется дополнительно использовать опцию "Top Surface".

- Для осей в структуре интерполяции должны быть выполнены следующие условия:
  - FIR-фильтр рывка нижних частот активен.
  - Зависящие от динамики настройки FIR-фильтра сконфигурированы.
  - Перезапись настроек фильтра посредством CALCFIR разрешена.
- Ограничение рывка SOFT/SOFTA активно.

Синтаксис

CALCFIR запрограммирован в отдельном кадре. Вызов в цикле изготовителя CUST_832 выполняется непосредственно после программирования динамического режима:

```
...
DYN...
CALCFIR
...
```

Объяснение

DYN...:	Команда из G-группы 59 для выбора динамического режима	
	DYNNORM:	Активировать **обычную динамику**
	DYNPOS:	Активировать динамику для **режима позиционирования, нарезания внутренней резьбы**
	DYNROUGH:	Активировать динамику для **черновой обработки**
	DYNSEMIFIN:	Активировать динамику для **предварительной чистовой обработки**
	DYNFINISH:	Активировать динамику для **чистовой обработки**
	DYNPREC:	Активировать динамику для **точной чистовой обработки**
CALCFIR:	Предопределенная процедура коррекции динамики FIR-фильтров рывка низких частот	

Примечание

CALCFIR выполняет неявную NEWCONFIG для ввода в действие результата расчета FIR-фильтра и в спец. для оси машинных данных для параметрирования FIR-фильтров рывка низких частот, и для CPRECON (Страница 881). Неявная NEWCONFIG ведет к тому, что другие действительные для NEWCONFIG машинные данные могут быть активированы.

Граничные условия

Использование в синхронных действиях

CALCFIR **не может** быть запрограммирована в синхронных действиях.

4.15.10 Чтение/запись параметров привода в программе обработки детали (DRVPRD, DRVPWR)

Языковые команды DRVPRD и DRVPWR позволяют изготовителю станка считывать и записывать параметры привода по осям на уровне программы обработки детали.

Так как в случае DRVPRD и DRVPWR речь идет о готовых процедурах, для которых требуется отдельный кадр УП, в один момент времени в одном канале возможны чтение **или** запись только одного параметра привода.

DRVPRD и DRVPWR используются в созданном изготовителем станка цикле.

Условия

Действуют следующие условия:

- Ввод в эксплуатацию управляемого ЧПУ привода завершен.
- Циклы в директории CST (Siemens) или CMA (изготовитель) всегда имеют необходимые права на чтение и запись.

Синтаксис

```
DRVPRD(<Result>,<Axis>,<DrvParNo>,<DrvParIdx>,<Value>)
DRVPWR(<Result>,<Axis>,<DrvParNo>,<DrvParIdx>,<Value>)
```

Значение

`DRVPRD()`:	Предопределенная процедура для **чтения** параметров привода		
`DRVPWR()`:	Предопределенная процедура для **записи** параметров привода		
`<Result>`:	Переменная результата: Возвращаемое значение для состояния		
	Тип данных:	INT	
	Значение:	=0	Функция была выполнена без ошибок
		> 0	Функция завершена с ошибкой Подробности см. "Дополнительная информация" > "Диагностика через возвращаемые значения".
`<Axis>`:	Имя оси станка		
	Тип данных:	AXIS	
	Диапазон значений:	Определенные в канале имена осей станка	

<DrvParNo>:	Номер параметра привода	
	Тип данных:	INT
<DrvParIdx>:	Индекс параметра привода	
	Тип данных:	INT
<Value>:	Значение параметра привода DRVPRD: считанное значение DRVPWR: записываемое значение	
	Тип данных:	REAL

Пример

Чтение параметра p1460[0] "Регулятор скорости П-усиление скорость адаптации" оси X:

Программный код	Комментарий
`N100 DEF INT _Status`	; определение переменных для возвращаемого значения языковой команды
`N110 DEF REAL _Result`	; определение переменных для считанного значения параметра
`N120 DRVPRD(_Status,AX1,1460,0,_Result)`	; чтение параметра p1460

Дополнительная информация

Диагностика через возвращаемые значения

Обращение к параметру привода считается неудавшимся, если возвращаемое значение языковой команды отлично от нуля.

Для возвращаемого значения зарезервировано два диапазона номеров:

Возвращаемые значения при вызове языковой команды:

0 Нет ошибок доступа.
1 Недопустимое значение.
2 Ось отсутствует.
3 Привод отсутствует.
4 Нет прав доступа.
5 Невозможно при поиске кадра/тестировании программы.
6 Доступ к приводу невозможен.

Возвращаемые значения при записи/чтении параметра привода:

1000 Недопустимый номер параметра.
1001 Значение параметра не может быть изменено.
1002 Выход за нижнее или верхнее предельное значение.
1003 Ошибочный субиндекс.

1004 Нет массива и субиндекса.

...

1107 Обращение по записи при разблокированном регуляторе запрещено.

1110 Обращение по записи разрешается только в состоянии ввода в эксплуатацию: Двигатель (p0010 = 3).

...

1204 Доступ по записи запрещен.

Для того, чтобы отличать возвращаемые значения языковой команды от возвращаемых значений привода, прибавляется значение 1000.

Дополнительная информация: Справочник по диагностике SINUMERIK, аварийное сообщение 201042 "Ошибки параметров при загрузке проекта"

Поведение при программной симуляции

Доступные параметры привода зависят от варианта и объема используемой симуляции привода. При отсутствии симуляции привода языковая команда возвращает значение 3.

С помощью переменной $P_SIM изготовитель станка может предусмотреть специальную обработку для программной симуляции.

Взаимодействие с другими функциями

- Параллельные обращения по записи/чтению
  Параллельные обращения по записи/чтению со стороны HMI или других клиентов могут влиять на время обработки языковой команды.
- REPOS
  При прерывании обработки языковой команды событием REORG команда повторяется.
- Поиск кадра / тестирование программы
  При поиске кадра и тестировании программы (также SERUPRO) языковая команда для чтения и записи параметров привода не поддерживается. Переменная результата возвращает ошибку доступа 5.
  Опрос с помощью системных переменных $P_SEARCH и $P_ISTEST в программе обработки детали на предмет активности поиска кадра или тестирования программы, позволяет пропустить DRVPRD и DRVPWR для реализации запасной стратегии.

Граничные условия

При программировании учитывать следующие условия:

- Использование в синхронных действиях
  Языковая команда не может быть запрограммирована в синхронных действиях, так как синхронное действие не ожидает выполнения нециклического обращения.
  При использовании языковой команды в синхронном действии появляется аварийное сообщение 12571.
- Параметры привода для чтения/записи
  Возможно чтение или запись только осевых параметров привода SERVO-DO или HLA-DO, а также параметров стандартных приводов PROFIDRIVE, с которым согласованы оси ЧПУ.
  Чтение/запись параметров следующих DO невозможна: CU_I, CU_NX, CU3x, TM, HUB, INFEED.

4.16 Соединения осей

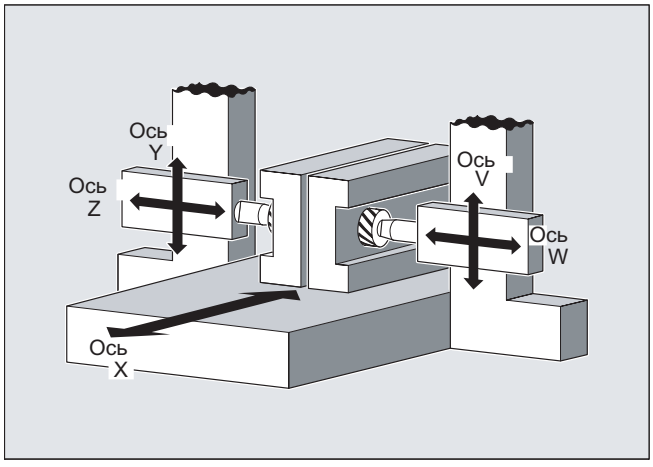
4.16.1 Буксировка (TRAILON, TRAILOF)

При движении определенной ведущей оси согласованные с ней буксируемые оси (= ведомые оси) проходят, с учетом коэффициента связи, произведенные ведущей осью пути перемещения.

Ведущая ось и ведомые оси вместе образуют структуру буксировок.

Области применения

- Перемещение оси через симулированную ось. Ведущая ось это симулированная ось, а буксируемая ось этот реальная ось. Тем самым реальная ось может перемещаться с учетом коэффициента связи.
- Двухсторонняя обработка с 2 структурами буксировок:
  1. Ведущая ось Y, буксируемая ось V
  2. Ведущая ось Z, буксируемая ось W



Синтаксис

```
TRAILON(<ведомая ось>,<ведущая ось>,<коэффициент связи>)
TRAILOF(<ведомая ось>,<ведущая ось>,<ведущая ось 2>)
TRAILOF(<ведомая ось>)
```

Значение

TRAILON:	Команда для включения и определения структуры буксировок	
	Активность:	модально
<ведомая ось>:	Параметр 1: Обозначение буксируемой оси **Примечание:** Буксируемая ось также может быть ведущей осью для других буксируемых осей. Таким образом могут создаваться различные структуры буксировок.	
<ведущая ось>:	Параметр 2: Обозначение ведущей оси	

<коэффициент связи>:	Параметр 3: коэффициент связи Коэффициент связи указывает желаемое соотношение ходов буксируемой оси и ведущей оси: <коэффициент связи> = ход буксируемой оси / ход ведущей оси	
	Тип:	REAL
	Предустановка:	1
	Ввод отрицательного значения вызывает движение перемещения ведущей и буксируемой оси в противоположном направлении. Если коэффициент связи при программировании не указывается, то автоматически действует коэффициент связи 1.	
TRAILOF:	Команда отключения структуры буксировок	
	Активность:	модально
	TRAILOF с 2 параметрами отключает только связь с указанной ведущей осью: TRAILOF(<ведомая ось>,<ведущая ось>) Если буксируемая ось имеет 2 ведущие оси, то для отключения обоих связей можно вызвать TRAILOF с 3 параметрами: TRAILOF(<ведомая ось>,<ведущая ось>,<ведущая ось 2>) Идентичный результат дает программирование TRAILOF без указания ведущей оси: TRAILOF(<ведомая ось>)	

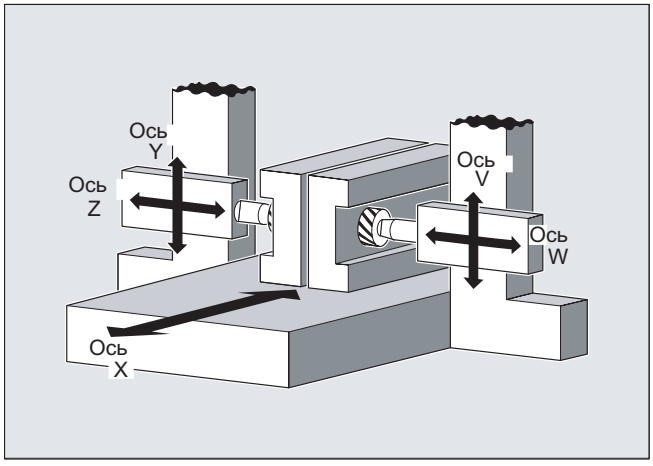
Примечание

Буксировка всегда выполняется в базовой кинематической системе (BKS).

Число активируемых одновременно структур буксировок ограничивается только возможностями комбинирования имеющихся на станке осей.

Пример

Деталь должна быть обработана с двух сторон с помощью представленной конфигурации осей. Для этого создается 2 структуры буксировок.



Программный код	Комментарий
...	
`N100 TRAILON(V,Y)`	; Включение 1-й структуры буксировок
`N110 TRAILON(W,Z,-1)`	; Включение 2-й структуры буксировок. Отрицательный коэффициент связи: Буксируемая ось движется в направлении, противоположном ведущей оси.
`N120 G0 Z10`	; Подача оси `Z` и `W` в противоположном осевом направлении.
`N130 G0 Y20`	; Подача оси `Y` и оси `V` в одном осевом направлении.
...	
`N200 G1 Y22 V25 F200`	; Наложение зависимого и независимого движения буксируемой оси `V`.
...	
`TRAILOF(V,Y)`	; Отключение 1-й структуры буксировок.
`TRAILOF(W,Z)`	; Отключение 2-й структуры буксировок.

Дополнительная информация

Типы осей

Структура буксировок может состоять из любых комбинаций линейных и круговых осей. В качестве ведущей оси при этом может быть определена и симулированная ось.

Буксируемые оси

С одной буксируемой осью одновременно может быть согласовано макс. 2 ведущие оси. Согласование осуществляется в различных структурах буксировок.

Буксируемая ось может быть запрограммирована со всеми доступными командами движения (`G0`, `G1`, `G2`, `G3`, ...). В дополнение к независимо определенным ходам буксируемая ось перемещается по ходам, полученным с помощью коэффициента связи из ее ведущих осей.

Динамическое ограничение

Динамическое ограничение зависит от типа активации структуры буксировок:

- Активация в программе обработки детали
  Если активация выполняется в программе обработки детали и все ведущие оси находятся в активирующем канале как программные оси, то при перемещении ведущих осей динамика буксируемых осей учитывается таким образом, что не возникает перегрузок буксируемых осей.
  Если активация выполняется в программе обработки детали с ведущими осями, которые не активны в активирующем канале как программные оси ($AA_TYP ≠ 1), то при перемещении ведущих осей динамика буксируемой оси не учитывается. Из-за этого у буксируемых осей с меньшей, чем требуемая для связи, динамикой, возможна перегрузка.
- Активация в синхронном действии
  Если активация выполняется в синхронном действии, то при перемещении ведущих осей динамика буксируемых осей не учитывается. Из-за этого у буксируемых осей с меньшей, чем требуемая для связи, динамикой, возможна перегрузка.

⚠ ОСТОРОЖНО

Перегрузка оси

Если структура буксировок активируется

- в синхронных действиях
- в программе обработки детали с ведущими осями, которые не являются программными осями в канале буксируемой оси,

то пользователь/изготовитель станка должен предусмотреть подходящие меры по недопущению перегрузки буксируемых осей из-за движений перемещения ведущей оси.

Состояние связи

Состояние связи оси может быть запрошено в программе обработки детали с помощью системной переменной:

$AA_COUP_ACT[<ось>]

Вели-чина	Значение
0	нет активного соединения
8	буксировка активна

Индикация остаточного пути буксируемой оси для круговых осей модуло

Если ведущая и буксируемая ось это круговые оси модуло, то движения перемещения в ведущей оси суммируются n * 360° где n = 1, 2, 3... , на индикации остаточного пути буксируемой оси для отключения соединения.

Пример: Сегмент программы с TRAILON и ведущей осью B и ведомой осью C

Программный код	Комментарий
`TRAILON(C,B,1)`	; включить соединение
`G0 B0`	; Исходное положение

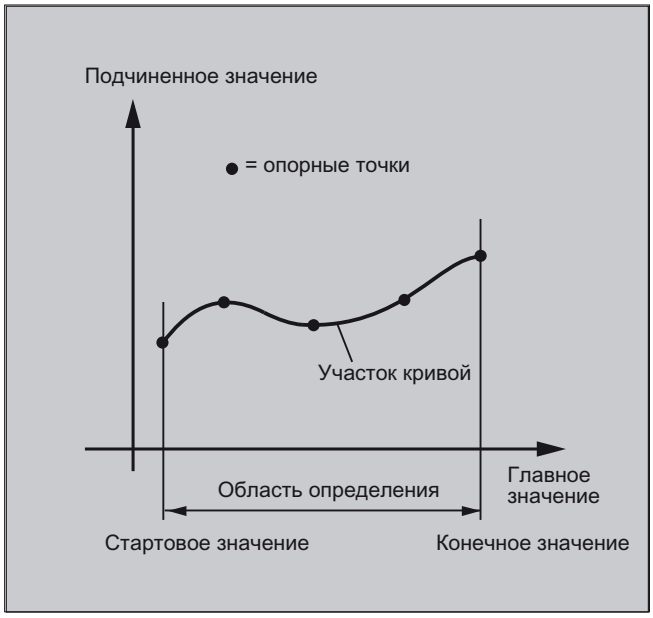
Программный код	Комментарий
	; Индикация остаточного пути на начале кадра:
G91 B360	; B=360, C=360
G91 B720	; B=720, C=1080
G91 B360	; B=360, C=1440

4.16.2 Таблицы кривых (CTAB)

4.16.2.1 Определение таблиц кривых (CTABDEF, CATBEND)

Таблица кривых представляет собой программу обработки детали или сегмент программы обработки детали, обозначенный препозицией CTABDEF и завершающей командой CTABEND .

Внутри этого сегмента программы обработки детали через операторы движения отдельным позициям ведущей оси присваиваются однозначные позиции ведомой оси, служащие в качестве опорных точек для вычисления линии кривой в форме полинома макс. 5-го порядка.



Условие

Для определения таблиц кривых через соответствующие конфигурирование MD должно быть зарезервировано место в памяти (→ Изготовитель станка!).

Синтаксис

CTABDEF(<ведомая ось>,<ведущая ось>,<n>,<периодичность>[,<место сохранения>])
...

```
CTABEND
```

Значение

CTABDEF( ):	Начало определения таблиц кривых	
CTABEND:	Конец определения таблиц кривых	
<ведомая ось>:	Ось, движение которой должно быть рассчитано через таблицу кривых	
<ведущая ось>:	Ось, предоставляющая главные значения для расчета движения ведомой оси	
<n>:	Номер (**ID**) таблицы кривых Номер таблицы кривых является однозначным и не зависит от места сохранения. Таблицы с одинаковым номером не могут находиться в статической и динамической памяти.	
<периодичность>:	Периодичность таблицы	
	0	Таблица не периодическая (обрабатывается только один раз, и для круговых осей)
	1	Таблица периодическая относительно ведущей оси
	2	Таблица периодическая относительно ведущей оси и ведомой оси
<место сохранения>:	Указание места сохранения (опция)	
	"SRAM"	Таблица кривых создается в **статической** памяти ЧПУ.
	"DRAM"	Таблица кривых создается в **динамической** памяти ЧПУ.
	Примечание: Если значение для этого параметра не программируется, то используется установленное с MD20905 $MC_CTAB_DEFAULT_MEMORY_TYPE стандартное место сохранения.	

Примечание

Перезапись

Таблица кривых перезаписывается, как только при повторном определении таблиц используется ее номер (<n>) (исключение: Таблица кривых активна в соединении осей или заблокирована с CTABLOCK). **При замене соответствующее предупреждение не выводится!**

Примеры

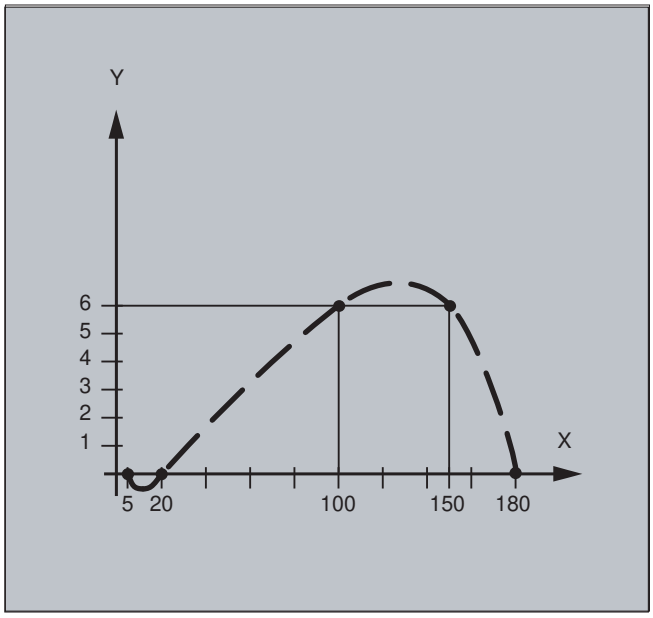
Пример 1: Сегмент программы как определение таблиц кривых

Сегмент программы должен быть использован без изменений для определения таблицы кривых. Встречающаяся там команда для остановки предварительной обработки STOPRE может быть остановлена и снова сразу же активируется, как только сегмент программы более не будет использоваться для определения таблиц и CTABDEF и CTABEND были удалены

Программный код	Комментарий
...	
CTABDEF(Y,X,1,1)	; Определение таблицы кривых.

Программный код	Комментарий
…	
IF NOT ($P_CTABDEF)	
stopre	
ENDIF	
…	
CTABEND	

Пример 2: Определение не периодической таблицы кривых



Программный код	Комментарий
N100 CTABDEF(Y,X,3,0)	; Начало определения не периодической таблицы кривых с номером 3.
N110 X0 Y0	; 1.Оператор движения, определяет стартовые значения и 1-ю опорную точку: главное значение: 0, подчиненное значение: 0
N120 X20 Y0	; 2-я опорная точка: главное значение: 0...20, подчиненное значение: Стартовое значение... 0
N130 X100 Y6	; 3-я опорная точка: главное значение: 20...100, подчиненное значение: 0...6
N140 X150 Y6	; 4-я опорная точка: главное значение: 100...150, подчиненное значение: 6...6
N150 X180 Y0	; 5-я опорная точка: главное значение: 150...180, подчиненное значение: 6...0
N200 CTABEND	; Конец определения. Таблица кривых в своем внутреннем представлении создается как полином макс. 5-го порядка. Расчет участка кривой с указанными опорными точками зависит от выбранного модально типа интерполяции (круговая, линейная, сплайн-интерполяция). Состояние программы обработки детали перед началом определения восстанавливается.

Пример 3: Определение периодической таблицы кривых

Определение периодической таблицы кривых с номером 2, диапазон главного значения от 0 до 360, движение ведомой оси от 0 до 45 градусов и обратно на 0:

Программный код	Комментарий
N10 DEF REAL DEPPOS	
N20 DEF REAL GRADIENT	
N30 CTABDEF(Y,X,2,1)	; Начало определения.
N40 G1 X=0 Y=0	
N50 POLY	
N60 PO[X]=(45.0)	
N70 PO[X]=(90.0) PO[Y]=(45.0,135.0,-90)	
N80 PO[X]=(270.0)	
N90 PO[X]=(315.0) PO[Y]=(0.0,-135.0,90)	
N100 PO[X]=(360.0)	
N110 CTABEND	; Конец определения.
;Тест кривой через связь Y с X:	
N120 G1 F1000 X0	
N130 LEADON(Y,X,2)	
N140 X360	
N150 X0	
N160 LEADOF(Y,X)	
N170 DEPPOS=CTAB(75.0,2,GRADIENT)	; Чтение табличной функции для главного значения 75.0
N180 G0 X75 Y=DEPPOS	; Позиционирование ведущей и ведомой оси.
;После включения связи синхронизация ведомой оси не требуется.	
N190 LEADON(Y,X,2)	
N200 G1 X110 F1000	
N210 LEADOF(Y,X)	
N220 M30	

Дополнительная информация

Стартовое и конечное значение таблицы кривых

Стартовым значением для начала области определения таблицы кривых считается первое указание связанных позиций осей (первый оператор движения) внутри определения таблицы кривых. Конечное значение области определения таблицы кривых определяется через последнюю команду перемещения соответственно.

Доступная мощность языка

Внутри определения таблиц кривых доступна вся мощность языка ЧПУ.

Примечание

Следующие данные недопустимы в определениях таблиц кривых:

- Остановка предварительной обработки
- Скачки в движении ведущих осей (к примеру, при смене трансформаций)
- Оператор движения только для ведомой оси
- Реверсирование ведущей оси, т. е. позиция ведущей оси всегда должна быть однозначной
- Операторы CTABDEF и CTABEND на различных программных уровнях.

Активность модальных операторов

Все действующие модально операторы, встречающиеся внутри определения таблицы кривых, по завершении определения таблицы становятся недействительными. Поэтому программа обработки детали, в которой выполняется определение таблицы, до и после определения таблицы находится в идентичном состоянии.

Присваивания R-параметрам

Присваивания R-параметрам внутри определения таблиц после CTABEND отменяются.

Пример:

Программный код	Комментарий
...	
R10=5 R11=20	; R10=5
...	
CTABDEF	
G1 X=10 Y=20 F1000	
R10=R11+5	; R10=25
X=R10	
CTABEND	
...	; R10=5

Активация ASPLINE, BSPLINE, CSPLINE

Если внутри определения таблицы кривых CTABDEF ... CTABEND активируется ASPLINE, BSPLINE или CSPLINE, то перед этой активацией сплайна нужно запрограммировать как минимум одну стартовую точку. Немедленной активации после CTABDEF следует избегать, поскольку иначе сплайн зависит от актуальной позиции оси перед определением таблицы кривых.

Пример:

Программный код
...
CTABDEF(Y,X,1,0)

Программный код

```
X0 Y0
ASPLINE
X=5 Y=10
X10 Y40
...
CTABEND
```

Повторное использование таблиц кривых

Вычисленная через таблицу кривых функциональная связь ведущей и ведомой оси сохраняется под выбранным номером таблицы и после завершения программы обработки детали и после POWER OFF, если таблица находится в статической памяти ЧПУ (SRAM).

Таблица, созданная в динамической памяти (DRAM), удаляется при POWER OFF и при необходимости должна быть создана заново.

Созданная однажды таблица кривых может применяться к любым комбинациям ведущей и ведомой оси и не зависит от того, какие оси используются для создания таблицы кривых.

Замена таблиц кривых

Таблица кривых заменяется, как только при повторном определении таблиц используется ее номер.

Исключение: Таблица кривых активна в соединении осей или заблокирована с CTABLOCK.

Примечание

При замене таблиц кривых соответствующее предупреждение не выводится!

Определение таблиц кривых активно?

С помощью системной переменной `$P_CTABDEF` из программы обработки детали можно запросить, активно ли определение таблиц кривых.

Отмена определения таблиц кривых

Сегмент программы обработки детали после вычленения операторов для определения таблиц кривых снова может использоваться как реальная программа обработки детали.

Загрузка таблиц кривых через "Выполнение с внешнего устройства"

При внешней обработке таблиц кривых размер буфера догрузки (DRAM) должен быть выбран через MD18360 $MN_MM_EXT_PROG_BUFFER_SIZE таким образом, чтобы все определение таблиц кривых одновременно могло бы быть помещено в буфер догрузки. В ином случае выполнение программы обработки детали отменяется с ошибкой.

Скачки ведомой оси

В зависимости от установки машинных данных:
MD20900 $MC_CTAB_ENABLE_NO_LEADMOTION
можно установить допуск для скачков ведомой оси при отсутствующем движении ведущей оси.

4.16.2.2 Проверка наличия таблицы кривых (CTABEXISTS)

С помощью команды CTABEXISTS можно проверить, имеется ли определенный номер таблицы кривых в памяти ЧПУ.

Синтаксис

```
CTABEXISTS(<n>)
```

Значение

CTABEXISTS:	Проверяет, имеется ли таблица кривых с номером <n> в статической или динамической памяти ЧПУ	
	0	Таблица не существует
	1	Таблица существует
<n>:	Номер (**ID**) таблицы кривых	

4.16.2.3 Удаление таблиц кривых (CTABDEL)

С помощью CTABDEL можно удалять таблицы кривых.

Примечание

Таблиц кривых, активные в соединении осей, не могут быть удалены.

Синтаксис

```
CTABDEL(<n>)
CTABDEL(<n>,<m>)
CTABDEL(<n>,<m>,<место сохранения>)
CTABDEL()
CTABDEL(,,<место сохранения>)
```

Значение

CTABDEL:	Команда удаления таблиц кривых
<n>:	Номер (**ID**) удаляемой таблицы кривых При удалении области таблиц кривых CTABDEL(<n>,<m>) с <n> указывается номер первой таблицы кривых области.
<m>:	При удалении области таблиц кривых CTABDEL(<n>,<m>) с <m> указывается номер последней таблицы кривых области. <m> должна быть больше <n>!

<место сохранения>:	Указание места сохранения (опция) При удалении **без** указания места сохранения, указанные таблицы кривых удаляются в статической и динамической памяти ЧПУ. При удалении **с** указанием места сохранения, из указанных таблиц кривых удаляются только те, которые находятся в указанной памяти. Остальные сохраняются.	
	`"SRAM"`	Удаление в **статической** памяти ЧПУ
	`"DRAM"`	Удаление в **динамической** памяти ЧПУ

Если `CTABDEL` программируется без указания удаляемой таблицы кривых, то удаляются **все** таблицы кривых или все таблицы кривых в указанной памяти:

`CTABDEL():`	Удаляет все таблицы кривых в статической и динамической памяти ЧПУ
`CTABDEL(,,"SRAM"):`	Удаляет все таблицы кривых в статической памяти ЧПУ
`CTABDEL(,,"DRAM"):`	Удаляет все таблицы кривых в динамической памяти ЧПУ

Примечание

Если при многократном удалении `CTABDEL(<n>,<m>)` или `CTABDEL()` минимум одна из удаляемых таблиц кривых активна в соединении, то команда удаления не выполняется, т. е. **ни одна** из адресованных таблиц кривых не удаляется.

4.16.2.4 Блокировка таблиц кривых от удаления и перезаписи (CTABLOCK, CTABUNLOCK)

Таблицы кривых через установку блокировок могут быть защищены от непреднамеренного удаления и перезаписи. Установленная блокировка снова может быть снята в любое время.

Синтаксис

Установить блокировку:

```
CTABLOCK(<n>)
CTABLOCK(<n>,<m>)
CTABLOCK(<n>,<m>,<место сохранения>)
CTABLOCK()
CTABLOCK(,,<место сохранения>)
```

Снять блокировку:

```
CTABUNLOCK(<n>)
CTABUNLOCK(<n>,<m>)
CTABUNLOCK(<n>,<m>,<место сохранения>)
CTABUNLOCK()
CTABUNLOCK(,,<место сохранения>)
```

Значение

CTABLOCK:	Команда для **установки** блокировки от удаления/перезаписи	
CTABUNLOCK:	Команда для **снятия** блокировки от удаления/перезаписи CTABUNLOCK снова разрешает заблокированные с CTABLOCK таблицы кривых. Таблицы кривых, действующие в активном соединении, остаются заблокированными и не могут быть удалены. Блокировка с CTABLOCK отменена, как только блокирование через активное соединение отменяется с деактивацией соединения. Таким образом, эта таблица может быть удалена. Повторного вызова CTABUNLOCK не требуется.	
<n>:	Номер (**ID**) блокируемой/разблокируемой таблицы кривых При блокировке/разблокировке области таблиц кривых CTABLOCK(<n>,<m>)/CTABUNLOCK(<n>,<m>) с <n> указывается номер первой таблицы кривых области.	
<m>:	При блокировке/разблокировке области таблиц кривых CTABLOCK(<n>,<m>)/CTABUNLOCK(<n>,<m>) с <m> указывается номер последней таблицы кривых области. <m> должна быть больше <n>!	
<место сохранения>:	Указание места сохранения (опция) При установке/снятии блокировки **без** указания места сохранения, указанные таблицы кривых блокируются/разблокируются в статической и динамической памяти ЧПУ. При установке/снятии блокировки **с** указанием места сохранения, из указанных таблиц кривых блокируются/разблокируются только те, которые находятся в указанной памяти. Прочие не блокируются/разблокируются.	
	"SRAM"	Установить/снять блокировку в **статической** памяти ЧПУ
	"DRAM"	Установить/снять блокировку в **динамической** памяти ЧПУ

Если CTABLOCK/CTABUNLOCK программируется без указания блокируемой/разблокируемой таблицы кривых, то блокируются/разблокируются **все** таблицы кривых или все таблицы кривых в указанной памяти:

CTABLOCK():	Блокирует все таблицы кривых в статической и динамической памяти ЧПУ
CTABLOCK(,,"SRAM"):	Блокирует все таблицы кривых в статической памяти ЧПУ
CTABLOCK(,,"DRAM"):	Блокирует все таблицы кривых в динамической памяти ЧПУ
CTABUNLOCK():	Деблокирует все таблицы кривых в статической и динамической памяти ЧПУ
CTABUNLOCK(,,"SRAM"):	Деблокирует все таблицы кривых в статической памяти ЧПУ
CTABUNLOCK(,,"DRAM"):	Деблокирует все таблицы кривых в динамической памяти ЧПУ

4.16.2.5 Таблицы кривых: Определение свойств таблиц (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD)

С помощью этих команд можно запросить важные свойства таблицы кривых (номер таблицы, состояние блокировки, место сохранения, периодичность).

Синтаксис

```
CTABID(<p>)
CTABID(<p>,<место сохранения>)
CTABISLOCK(<n>)
CTABMEMTYP(<n>)
TABPERIOD(<n>)
```

Значение

CTABID:	Выводит **номер таблицы**, внесенной в указанную память в качестве <p>-ной таблицы кривых. Пример: CTABID(1,"SRAM") возвращает номер первой таблицы кривых в статической памяти ЧПУ. Первая таблица кривых при этом соответствует таблице кривых со старшим номером таблицы. **Примечание:** Если между следующими друг за другом вызовами CTABID последовательность таблиц кривых в памяти изменяется, к примеру, из-за удаления таблиц кривых с CTABDEL, то CTABID(<p>,...) может вернуть с тем же номером <p> отличную от прежней таблицу кривых.	
CTABISLOCK:	Возвращает **состояние блокировки** таблицы кривых с номером <n>:	
	0	Таблица не заблокирована
	1	Таблица заблокирована через CTABLOCK
	2	Таблица заблокирована через активное соединение
	3	Таблица заблокирована через CTABLOCK и активное соединение
	-1	Таблица не существует
CTABMEMTYP:	Возвращает **место сохранения** таблицы кривых с номером <n>:	
	0	Таблица в статической памяти ЧПУ
	1	Таблица в динамической памяти ЧПУ
	-1	Таблица не существует
CTABPERIOD:	Возвращает **периодичность** таблицы кривых с номером <n>:	
	0	Таблица не периодическая
	1	Таблица периодическая в ведущей оси
	2	Таблица периодическая в ведущей и ведомой оси
	-1	Таблица не существует
<p>:	Номер элемента в памяти	
<n>:	Номер (**ID**) таблицы кривых	
<место сохранения>:	Указание места сохранения (опция)	
	"SRAM"	**Статическая** память ЧПУ
	"DRAM"	**Динамическая** память ЧПУ
	Примечание: Если значение для этого параметра не программируется, то используется установленное с MD20905 $MC_CTAB_DEFAULT_MEMORY_TYPE стандартное место сохранения.	

4.16.2.6 Чтение значений таблиц кривых (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX)

Следующие значения таблиц кривых могут считываться в программе обработки детали:

- Значения ведомой и ведущей оси в начале и конце таблицы кривых
- Значения ведомой оси в начале и конце сегмента кривой
- Значение ведомой оси к значению ведущей оси
- Значение ведущей оси к значению ведомой оси
- Мин. и макс. значение ведомой оси
  - во всей области определений таблицы кривых
    или
  - на определенном интервале таблицы кривых

Синтаксис

```
CTABTSV(<n>,<градиент>[,<ведомая ось>])
CTABTEV(<n>,<градиент>[,<ведомая ось>])
CTABTSP(<n>,<градиент>[,<ведущая ось>])
CTABTEP(<n>,<градиент>[,<ведущая ось>])
CTABSSV(<главное значение>,<n>,<градиент>[,<ведомая ось>])
CTABSEV(<главное значение>,<n>,<градиент>[,<ведомая ось>])
CTAB(<главное значение>,<n>,<градиент>[,<ведомая ось>,<ведущая ось>]
CTABINV(<подчиненное значение>,<приближенное
значение>,<n>,<градиент>[,<ведомая ось>,<ведущая ось>]
CTABTMIN(<n>[,<ведомая ось>])
CTABTMAX(<n>[,<ведомая ось>])
CTABTMIN(<n>,<a>,<b>[,<ведомая ось>,<ведущая ось>])
CTABTMAX(<n>,<a>,<b>[,<ведомая ось>,<ведущая ось>])
```

Значение

CTABTSV:	Читать значение ведомой оси в **начале** таблицы кривых Nr. <n>
CTABTEV:	Читать значение ведомой оси в **конце** таблицы кривых Nr. <n>
CTABTSP:	Читать значение ведущей оси в **начале** таблицы кривых Nr. <n>
CTABTEP:	Читать значение ведущей оси в **конце** таблицы кривых Nr. <n>
CTABSSV:	Читать значение ведомой оси в **начале** относящегося к указанному значению ведущей оси (<главное значение>) сегмента кривой
CTABSEV:	Читать значение ведомой оси в **конце** относящегося к указанному значению ведущей оси (<главное значение>) сегмента кривой
CTAB:	Читать значение ведомой оси к указанному значению ведущей оси (<главное значение>)
CTABINV:	Читать значение ведущей оси к указанному значению ведомой оси (<подчиненное значение>)

CTABTMIN:	Определить **мин. значение** ведомой оси: • во всей области определений таблицы кривых или • на определенном интервале <a> ... <b>
CTABTMAX:	Определить **макс. значение** ведомой оси: • во всей области определений таблицы кривых или • на определенном интервале <a> ... <b>
<n>:	Номер (**ID**) таблицы кривых
<градиент>:	В параметре <градиент> возвращается **градиент** функции таблицы кривых на полученной позиции
<ведомая ось>:	Ось, движение которой должно быть рассчитано через таблицу кривых (опция)
<ведущая ось>:	Ось, предоставляющая главные значения для расчета движения ведомой оси (опция)
<подчиненное значение>:	Значение ведомой оси для чтения соответствующего значения ведущей оси при CTABINV
<главное значение>:	Значение ведущей оси: • для чтения соответствующего значения ведомой оси при CTAB или • для выбора сегмента кривой при CTABSSV/CTABSEV
<приближенное значение>:	Согласование значения ведущей оси со значением ведомой оси при CTABINV не всегда должно быть однозначным. Поэтому для CTABINV в качестве параметра необходимо приближенное значение для ожидаемого значения ведущей оси.
<a>:	**Нижняя** граница интервала главного значения при CTABTMIN/CTABTMAX
<b>:	**Верхняя** граница интервала главного значения при CTABTMIN/CTABTMAX
	Примечание: Интервал главного значения <a> ... <b> не должен выходить за область определения таблицы кривых.

Примеры

Пример 1:

Определить значения ведомой и ведущей оси в начале и конце таблицы кривых, а также мин. и макс. значение ведомой оси во всей области определения таблицы кривых.

Программный код	Комментарий
N10 DEF REAL STARTPOS	
N20 DEF REAL ENDPOS	
N30 DEF REAL STARTPARA	
N40 DEF REAL ENDPARA	
N50 DEF REAL MINVAL	
N60 DEF REAL MAXVAL	
N70 DEF REAL GRADIENT	

Программный код	Комментарий
...	
N100 CTABDEF(Y,X,1,0)	; начало определения таблицы
N110 X0 Y10	; стартовая позиция, 1-й сегмент таблицы
N120 X30 Y40	; конечная позиция 1-го сегмента таблицы = стартовая позиция 2-ого сегмента таблицы
N130 X60 Y5	; конечная позиция 2-й сегмент таблицы = ...
N140 X70 Y30	
N150 X80 Y20	
N160 CTABEND	; Конец определения таблицы.
...	
N200 STARTPOS=CTABTSV(1,GRADIENT)	; Значение ведомой оси на начале таблицы кривых = 10
N210 ENDPOS=CTABTEV(1,GRADIENT)	; Значение ведомой оси на конце таблицы кривых = 20
N220 STARTPARA=CTABTSP(1,GRADIENT)	; Значение ведущей оси на начале таблицы кривых = 0
N230 ENDPARA=CTABTEP(1,GRADIENT)	; Значение ведущей оси на конце таблицы кривых = 80
N240 MINVAL=CTABTMIN(1)	; Мин. значение ведомой оси при Y=5
N250 MAXVAL=CTABTMAX(1)	; Макс. значение ведомой оси при Y=40

Пример 2:

Определение значений ведомой оси в начале и конце относящегося к значению ведущей оси X=30 сегмента кривой.

Программный код	Комментарий
N10 DEF REAL STARTPOS	
N20 DEF REAL ENDPOS	
N30 DEF REAL GRADIENT	
...	
N100 CTABDEF(Y,X,1,0)	; Начало определения таблицы.
N110 X0 Y0	; стартовая позиция, 1-й сегмент таблицы
N120 X20 Y10	; конечная позиция 1-го сегмента таблицы = стартовая позиция 2-ого сегмента таблицы
N130 X40 Y40	; конечная позиция 2-й сегмент таблицы = ...
N140 X60 Y10	
N150 X80 Y0	
N160 CTABEND	; Конец определения таблицы.
...	
N200 STARTPOS=CTABSSV(30.0,1,GRADIENT)	; стартовая позиция Y во 2-м сегменте = 10
N210 ENDPOS=CTABSEV(30.0,1,GRADIENT)	; конечная позиция Y во 2-м сегменте = 40

Дополнительная информация

Использование в синхронных действиях

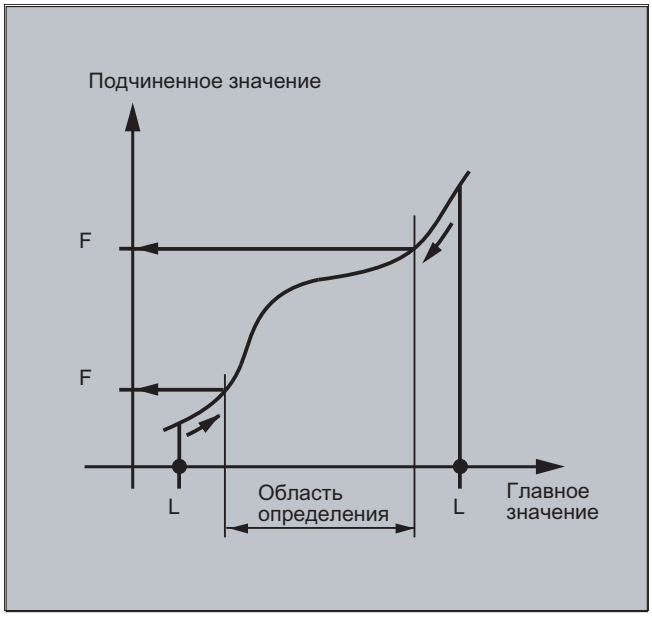
Все команды по чтению значений таблиц кривых могут использоваться и в синхронных действиях (см. также главу "Синхронные действия движения").

При использовании команд CTABINV, CTABTMIN и CTABTMAX следить за тем, чтобы:

- на момент выполнения было бы достаточно мощности ЧПУ
  или
- перед вызовом запросить число сегментов таблицы кривых, чтобы при необходимости можно было бы разделить данную таблицу.

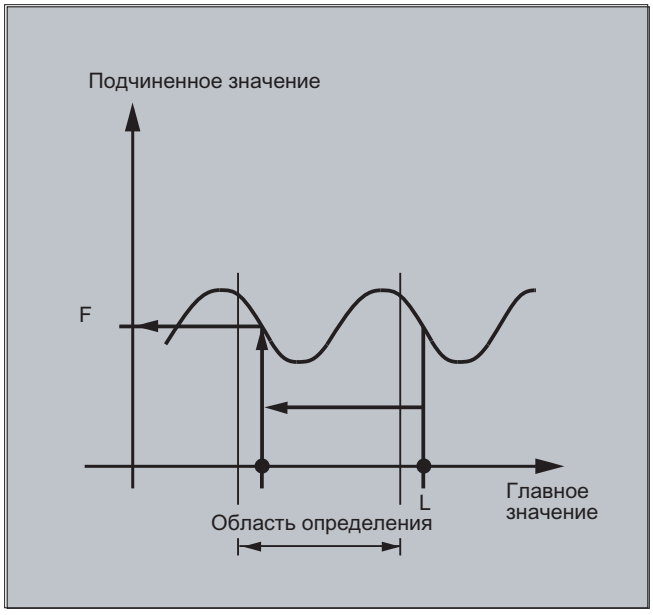
CTAB для не периодических таблиц кривых

Если указанное <главное значение> лежит за пределами области определения, то в качестве подчиненного значения выводится верхняя или нижняя граница:



CTAB для периодических таблиц кривых

Если <главное значение> лежит вне области определения, то главное значение модуло области определения нормируется и соответствующее подчиненное значение выводится.



Приближенное значение для CTABINV

Для команды CTABINV необходимо приближенное значение для ожидаемого главного значения. CTABINV возвращает главное значение, наиболее близкое к приближенному значению. Приближенным значением может быть, к примеру, главное значение из предшествующего такта интерполятора.

Градиент функции таблицы кривых

Вывод градиента (<градиент>) позволяет рассчитать скорость ведущей или ведомой оси на соответствующей позиции.

Указание ведущей или ведомой оси

Опциональное указание ведущей и/или ведомой оси важно, т.к. ведущая и ведомая оси сконфигурированы в различных единицах длины.

CTABSSV, CTABSEV

Команды CTABSSV и CTABSEV в следующих случаях **не** подходят для опроса запрограммированных сегментов:

- Запрограммированы окружности или эвольвенты.
- Фаска или закругление активны с CHF/RND.
- Перешлифовка с G643 активна.
- Активно сжатие кадра ЧПУ (COMP...).

4.16.2.7 Таблицы кривых: Проверка использования ресурсов (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL)

С помощью этих команд программист может получить актуальную информацию об использовании ресурсов для таблиц кривых, сегментов таблиц и полиномов.

Синтаксис

```
CTABNO
CTABNOMEM(<место сохранения>)
CTABFNO(<место сохранения>)
CTABSEGID(<n>,<место сохранения>)
CTABSEG(<место сохранения>,<тип сегмента>)
CTABFSEG(<место сохранения>,<тип сегмента>)
CTABMSEG(<место сохранения>,<тип сегмента>)
CTABPOLID(<n>)
CTABPOL(<место сохранения>)
CTABFPOL(<место сохранения>)
CTABMPOL(<место сохранения>)
```

Значение

CTABNO:	Определить общее число **определенных** таблиц кривых (в статической и динамической памяти ЧПУ)	
CTABNOMEM:	Определить число **определенных** таблиц кривых в указанном <месте сохранения>	
CTABFNO:	Определить число **еще возможных** таблиц кривых в указанном <месте сохранения>	
CTABSEGID:	Определить число сегментов кривой указанного <типа сегмента>, которые используются таблицей кривых с номером <n>	
CTABSEG:	Определить число **использованных** сегментов кривой указанного <типа сегмента> в указанном <месте сохранения>	
CTABFSEG:	Определить число **еще возможных** сегментов кривой указанного <типа сегмента> в указанном <месте сохранения>	
CTABMSEG:	Определить число **макс. возможных** сегментов кривой указанного <типа сегмента> в указанном <месте сохранения>	
CTABPOLID:	Определить число криволинейных полиномов, которые используются таблицей кривых с номером <n>	
CTABPOL:	Определить число **использованных** криволинейных полиномов в указанном <месте сохранения>	
CTABFPOL:	Определить число **еще возможных** криволинейных полиномов в указанном <месте сохранения>	
CTABMPOL:	Определить число **макс. возможных** криволинейных полиномов в указанном <месте сохранения>	
<n>:	Номер (**ID**) таблицы кривых	
<место сохранения>:	Указание места сохранения (опция)	
	"SRAM"	**Статическая** память ЧПУ
	"DRAM"	**Динамическая** память ЧПУ
	Примечание: Если значение для этого параметра не программируется, то используется установленное с MD20905 $MC_CTAB_DEFAULT_MEMORY_TYPE стандартное место сохранения.	

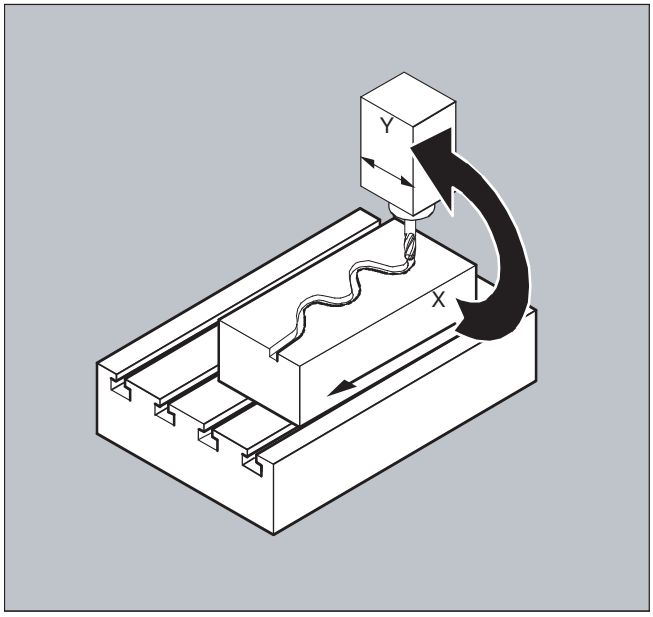
<тип сегмента>:	Указание типа сегмента (опция)	
	"L"	Линейные сегменты
	"P"	Полиномиальные сегменты
	Примечание: Если значение для этого параметра не программируется, то выводится сумма из линейных и полиномиальных сегментов.	

4.16.3 Осевое соединение по главному значению (LEADON, LEADOF)

Примечание

Эта функция недоступна для SINUMERIK 828D!

При осевом соединении по главному значению одна ведущая и одна ведомая ось перемещаются синхронно. При этом соответствующая позиция ведомой оси через таблицу кривых или вычисленный из нее полином однозначно согласована с – возможно симулированной – позицией ведущей оси.



Ведущей осью называется та ось, которая выводит входные значения для таблицы кривых. **Ведомой осью** называется ось, занимающая вычисленные через таблицу кривых позиции.

Соединение по фактическому и заданному значению

В качестве главных значений, т. е. выходных значений для определения позиции ведомой оси, могут использоваться:

- Фактические значения позиции ведущей оси: соединение по фактическому значению
- Заданные значения позиции ведущей оси: соединение по заданному значению

Соединение по главному значению всегда действует в базовой кинематической системе.

По созданию таблиц кривых см. главу "Таблицы кривых".

Синтаксис

LEADON(<ведомая ось>,<ведущая ось>,<n>)
LEADOF(<ведомая ось>,<ведущая ось>)

или отключение без указания ведущей оси:
LEADOF(<ведомая ось>)

Включение и выключение соединения по главному значению возможно как из программы обработки детали, так и при движении из синхронных действий.

Значение

LEADON:	Включить соединение по главному значению
LEADOF:	Выключить соединение по главному значению
<ведомая ось>:	Ведомая ось
<ведущая ось>:	Ведущая ось
<n>:	Номер таблиц кривых
$SA_LEAD_TYPE:	Переключение между соединением по заданному и фактическому значению

Отключение соединения по главному значению, LEADOF

При отключении соединения по главному значению ведомая ось снова становится обычной командной осью!

Осевое соединение по главному значению и различные рабочие состояния, RESET

В зависимости от установки в машинных данных, соединения по главному значению при RESET отключаются.

Пример соединения по главному значению из синхронного действия

На прессовальной установке обычное механическое соединение между ведущей осью (вал пуансона) и осями системы передачи из осей передачи и вспомогательных осей должно быть заменено на электронную соединительную систему.

Пример показывает, как на прессовальной установке механическая система передачи заменяется на электронную систему передачи. Процессы соединения и разъединения реализованы как **статические синхронные действия**.

От ведущей оси LW (вал пуансона) выполняется определенное управление осями передачи и вспомогательными осями как ведомыми осями через таблицы кривых.

Ведомые оси

X ось подачи или продольная ось
YL замыкающая или поперечная ось
ZL Hubachse
U подача вальцов, вспомогательная ось

V корректирующая головка, вспомогательная ось
W смазка, вспомогательная ось

Действия

В качестве операций в синхронных действиях, к примеру, встречаются:

- Соединение, LEADON(<ведомая ось>,<ведущая ось>,<номер таблицы кривых>)
- Разъединение,LEADOF(<ведомая ось>,<ведущая ось>)
- Установка фактического значения, PRESETON(<ось>,<значение>)
- Установка маркера, $AC_MARKER[i]=<значение>
- Тип соединения: реальное/виртуальное главное значение
- Подвод к позициям осей, POS[<ось>]=<значение>

Условия

В качестве условий обрабатываются цифровые быстрые входы, переменные реального времени $AC_MARKER и сравнения позиций, связанные логическим оператором И.

Примечание

В примере ниже переход строк, вставки и **жирный** шрифт используются только для того, чтобы улучшить читабельность программирования. Для СЧПУ все, стоящее под одним номером строки, является однострочным.

Комментарий

Программный код	Комментарий
	; Определяет все статические синхронные действия.
	; ****Сбросить маркер
N2 $AC_MARKER[0]=0 $AC_MARKER[1]=0 $AC_MARKER[2]=0 $AC_MARKER[3]=0 $AC_MARKER[4]=0 $AC_MARKER[5]=0 $AC_MARKER[6]=0 $AC_MARKER[7]=0	
	; **** E1 0=>1 соединение передача ВКЛ
N10 IDS=1 EVERY ($A_IN[1]==1) AND ($A_IN[16]==1) AND ($AC_MARKER[0]==0) DO LEADON(X,LW,1) LEADON(YL,LW,2) LEADON(ZL,LW,3) $AC_MARKER[0]=1	
	; **** E1 0=>1 соединение подача вальцов ВКЛ
N20 IDS=11 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_MARKER[5]==0) DO LEADON(U,LW,4) PRESETON(U,0) $AC_MARKER[5]=1	
	; **** E1 0->1 соединение корректирующая головка ВКЛ
N21 IDS=12 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_MARKER[6]==0) DO LEADON(V,LW,4) PRESETON(V,0) $AC_MARKER[6]=1	
	; **** E1 0->1 соединение смазка ВКЛ
N22 IDS=13 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_MARKER[7]==0) DO LEADON(W,LW,4) PRESETON(W,0) $AC_MARKER[7]=1	
	; **** E2 0=>1 соединение ВЫКЛ
N30 IDS=3 EVERY ($A_IN[2]==1) DO LEADOF(X,LW) LEADOF(YL,LW) LEADOF(ZL,LW) LEADOF(U,LW) LEADOF(V,LW) LEADOF(W,LW) $AC_MARKER[0]=0 $AC_MARKER[1]=0 $AC_MARKER[3]=0 $AC_MARKER[4]=0 $AC_MARKER[5]=0 $AC_MARKER[6]=0 $AC_MARKER[7]=0	
....	

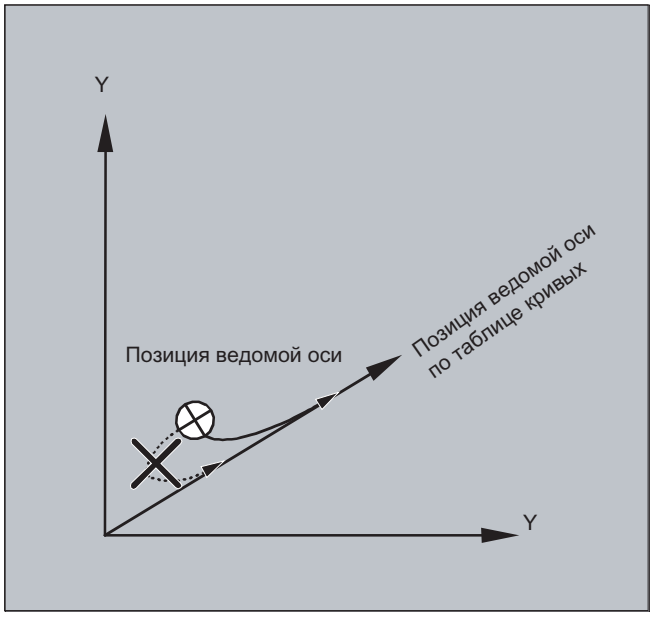
Программный код	Комментарий
N110 G04 F01	
N120 M30	

Описание

Для соединения по главному значению необходима синхронизация ведущей и ведомой оси. Эта синхронизация может быть достигнута только тогда, когда ведомая ось при включении соединения по главному значению находится в пределах диапазона допуска вычисленного из таблицы кривых участка кривой.

Диапазон допуска для положения ведомой оси определен через машинные данные `MD 37200: COUPLE_POS_POL_COARSE A_LEAD_TYPE` .

Если ведомая ось при включении соединения по главному значению еще не находится на соответствующей позиции, то синхронный ход восстанавливается автоматически, как только вычисленное заданное значение позиции для ведомой оси приближается к фактической позиции ведомой оси. При этом ведомая ось в процессе синхронизации перемещается в направлении, которое определено через заданную скорость ведомой оси (вычислена из скорости ведущей оси и по таблице кривых CTAB).

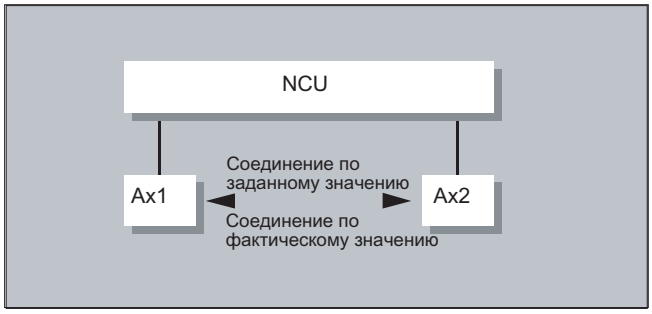


Нет синхронного хода

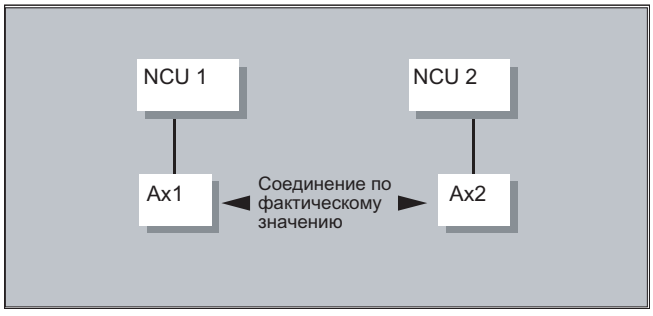
Если вычисленная заданная позиция ведомой оси при включении соединения по главному значению удаляется от актуальной позиции ведомой оси, то синхронный ход не устанавливается.

Соединение по фактическому и заданному значению

Соединение по заданному значению обеспечивает по сравнению с соединением по фактическому значению лучший синхронный ход между ведущей и ведомой осью и поэтому является стандартной предустановкой.



Соединение по заданному значению возможно только тогда, когда интерполяция ведущей и ведомой оси выполняется с одного NCU В случае внешней ведущей оси ведомая ось может быть соединена с ведущей осью только через фактические значения.



Переключение возможно через установочные данные $SA_LEAD_TYPE.

Переключение между соединением по фактическому и заданному значению всегда должно выполняться при остановленной ведомой оси. Т.к. только в состоянии покоя после переключения выполняется повторная синхронизация.

Пример использования

Безошибочное чтение фактических значений при сильных вибрациях станка невозможно. Поэтому при использовании соединения по главному значению в прессовом транспорте в рабочих операциях с сильными вибрациями может возникнуть необходимость переключения с соединения по фактическому значению на соединение по заданному значению.

Симуляция главного значения при соединении по заданному значению

Через машинные данные можно разделить интерполятор для ведущей оси и Servo. Таким образом, при соединении по заданному значению, заданные значения могут создаваться и без фактического движения ведущей оси.

Созданные через соединение по заданному значению главные значения для использования, к примеру, в синхронных действиях, могут считываться из следующих переменных:

- $AA_LEAD_P	Главное значение, позиция
- $AA_LEAD_V	Главное значение, скорость

Создание главных значений

Главные значения по выбору могут создаваться посредством иных самостоятельно запрограммированных методов. Созданные таким образом главные значения записываются и считываются из переменных

- $AA_LEAD_SP	Главное значение, позиция
- $AA_LEAD_SV	Главное значение, скорость

. Для использования этих переменных необходимы установочные данные $SA_LEAD_TYPE = 2.

Состояние соединения

В программе ЧПУ состояние соединения может быть запрошено посредством следующей системной переменной:

`$AA_COUP_ACT[`ось]
0: нет активного соединения
16: активно соединение по главному значению

Управление состоянием в синхронных действиях

Процессы переключения и соединения управляются через переменную реального времени:

```
$AC_MARKER[i] = n
```
с помощью:
i номер меркера
n значение состояния

4.16.4 Электронный редуктор (EG)

С помощью функции "Электронный редуктор" можно управлять движением **ведомой оси** по линейному кадру движения в зависимости от макс. пяти **ведущих осей**. Связи между ведущими осями и ведомой осью определены через коэффициент связи для каждой ведущей оси.

Вычисленный компонент движения ведомой оси образуется посредством сложения из компонентов движения отдельных ведущих осей, умноженных на соответствующие коэффициенты связи. При активизации структуры осей электронного редуктора может быть запущена синхронизация ведомой оси на определенную позицию. Из программы обработки детали структура редуктора может:

- определяться,
- включаться,
- отключаться,
- удаляться

.

Движение ведомой оси может следовать по выбору из

- заданных значений ведущих осей, а также
- фактических значений ведущих осей.

В качестве расширения возможна реализация и нелинейных связей между ведущими осями и ведомой осью через **таблицы кривых** (см. главу "Параметры траектории"). Возможно каскадирование электронного редуктора, т. е. ведомая ось одного электронного редуктора может быть ведущей осью для следующего электронного редуктора.

4.16.4.1 Определение электронного редуктора (EGDEF)

Структура осей электронного редуктора определяется через указание ведомой оси и минимум одной и макс. пяти ведущих осей с соответствующим типом соединения.

Условие

Условие для определения структуры осей электронного редуктора:

Для ведомой оси еще не должно быть определено соединение осей (при необходимости сначала удалить существующее посредством EGDEL).

Синтаксис

`EGDEF(`ведомая ось,ведущая ось1,тип связи1,ведущая ось2,тип связи2,...)

Значение

<table>
<tr><td>EGDEF:</td><td colspan="2">Определение электронного редуктора</td></tr>
<tr><td>Ведомая ось:</td><td colspan="2">Ось, управляемая ведущими осями</td></tr>
<tr><td>Ведущая ось1
,...,
Ведущая ось5</td><td colspan="2">Оси, управляющие ведомой осью</td></tr>
<tr><td rowspan="4">Тип связи1
,...,
Тип связи5</td><td colspan="2">Тип связи
Тип связи не должен быть одинаковым для всех ведущих осей и поэтому указывается отдельно для каждой ведущей оси.</td></tr>
<tr><td>Величина:</td><td>Значение:</td></tr>
<tr><td>0</td><td>Ведомая ось управляется фактическим значением соответствующей ведущей оси.</td></tr>
<tr><td>1</td><td>Ведомая ось управляется заданным значением соответствующей ведущей оси.</td></tr>
</table>

Примечание

Коэффициенты связи при определении структуры соединения электронного редуктора предустанавливаются на ноль.

Примечание

EGDEF вызывает остановку предварительной обработки. Определение редуктора с EGDEF должно использоваться без изменений и тогда, когда в системах одна или несколько ведущих осей через **таблицу кривых** воздействуют на ведомую ось.

Пример

Программный код	Комментарий
`EGDEF(C,B,1,Z,1,Y,1)`	; Определение структуры осей электронного редуктора. Ведущие оси `B, Z, Y` управляют ведомой осью C через заданное значение.

4.16.4.2 Включение электронного редуктора (EGON, EGONSYN, EGONSYNE)

Для включения структуры осей электронного редуктора существует 3 варианта.

Синтаксис

Вариант 1:

Включение структуры осей электронного редуктора по выбору без синхронизации с помощью:
`EGON(FA,"Режим смены кадра",LA1,Z1,N1,LA2,Z2,N2,...,LA5,Z5,N5)`

Вариант 2:

Включение структуры осей электронного редуктора по выбору с синхронизацией с помощью:
`EGONSYN(FA,"Режим смены кадра",SynPosFA,[,LAi,SynPosLAi,Zi,Ni])`

Вариант 3:

Включение структуры осей электронного редуктора по выбору с синхронизацией и задачей режима подвода с помощью:
`EGONSYNE(FA,"Режим смены кадра",SynPosFA,`режим подвода`[,LAi,SynPosLAi,Zi,Ni])`

Значение

Вариант 1:

FA	Ведомая ось	
Режим смены кадра:	Возможно использование следующих режимов:	
	`"NOC"`	Смена кадра выполняется сразу же
	`"FINE"`	Смена кадра при "Синхронном ходе точном"
	`"COARSE"`	Смена кадра при "Синхронном ходе грубом"
	`"IPOSTOP"`	Смена кадра при синхронном ходе со стороны заданного значения
LA1, ... LA5	Ведущие оси	

Z1, ... Z5	Числитель для коэффициента связи i
N1, ... N5	Знаменатель для коэффициента связи i Коэффициент связи i = числитель i/знаменатель i

Могут быть запрограммированы только ведущие оси, которые до этого были специфицированы с EGDEF. Должна быть запрограммирована минимум одна ведущая ось.

Вариант 2:

FA	Ведомая ось	
Режим смены кадра:	Возможно использование следующих режимов:	
	"NOC"	Смена кадра выполняется сразу же
	"FINE"	Смена кадра при "Синхронном ходе точном"
	"COARSE"	Смена кадра при "Синхронном ходе грубом"
	"IPOSTOP"	Смена кадра при синхронном ходе со стороны заданного значения
[,LAi,SynPosLAi,Zi,Ni]	(не записывать квадратные скобки) Мин. 1, макс. 5 последовательностей:	
LA1, ... LA5	Ведущие оси	
SynPosLAi	Синхронная позиция для ведущей оси i.	
Z1, ... Z5	Числитель для коэффициента связи i	
N1, ... N5	Знаменатель для коэффициента связи i Коэффициент связи i = числитель i/знаменатель i	

Могут быть запрограммированы только ведущие оси, которые до этого были специфицированы с EGDEF. Через запрограммированные "синхронные позиции" для ведомой оси (SynPosFA) и для ведущих осей (SynPosLA), определяются позиции, в которых структура соединения считается *синхронной*. Если при включении электронный редуктор находится не в синхронном состоянии, то ведомая ось перемещается на свою определенную синхронную позицию.

Вариант 3:

Параметры соответствуют таковым варианта 2, плюс:

Режим подвода:	Возможно использование следующих режимов:	
	"NTGT"	Оптимизированный по времени подвод к следующей впадине между зубьями
	"NTGP"	Оптимизированный по ходу подвод к следующей впадине между зубьями
	"ACN"	Абсолютное перемещение круговой оси в отрицательном направлении вращения
	"ACP"	Абсолютное перемещение круговой оси в положительном направлении вращения
	"DCT"	С оптимизацией по времени к запрограммированной синхронной позиции
	"DCP"	С оптимизацией по ходу к запрограммированной синхронной позиции

Вариант 3 влияет только на ведомые оси модуло, связанные с ведущими осями модуло. Оптимизация по времени учитывает границы скорости ведомой оси.

Дополнительная информация

Описание вариантов включения

Вариант 1:

Позиции ведущих осей, а также ведомой оси, на момент включения сохраняются как "Синхронные позиции". "Синхронные позиции" могут быть считаны с помощью системной переменной `$AA_EG_SYN`.

Вариант 2:

Если оси по модулю находятся в структуре соединения, то их значения позиций уменьшаются по модулю. Тем самым обеспечивается подвод к следующей возможной синхронной позиции (т.н. *относительная синхронизация*: например, следующая впадина между зубьями). Если для ведомой оси не дано "Разрешение наложения ведомой оси", сигнал интерфейсов DB(30 +номер оси), DBX 26 бит 4, то движение на синхронную позицию не выполняется. Вместо этого программа останавливается на кадре EGONSYN и самоудаляющаяся ошибка 16771 сигнализируется до тех пор, пока в.у. сигнал не будет установлен.

Вариант 3:

Расстояние между зубьями (градус) получается из: 360 * Zi/Ni. В том случае, если ведомая на момент вызова остановлена, то поведение оптимизации по ходу идентично таковому оптимизации по времени.

Если ведомая ось уже перемещается, с помощью NTGP синхронизация со следующей впадиной между зубьями выполняется независимо от текущей скорости ведомой оси. Если ведомая ось уже перемещается, с помощью NTGT синхронизация со следующей впадиной между зубьями выполняется в зависимости от текущей скорости ведомой оси. Для этого при необходимости выполняется и торможение оси.

Таблицы кривых

Если для одной из ведущих осей используется **таблица кривых**, то:

Ni	Знаменатель коэффициента связи линейных соединений должен быть установлен на 0. (знаменатель 0 был бы недопустим для линейных соединений). Знаменатель ноль указывает СЧПУ на то, что
Zi	должен быть интерпретирован как номер используемой таблицы кривых. Таблица кривых с указанным номером на момент включения должна быть уже определена.
LAi	Указание ведущей оси соответствует указанию ведущей оси при соединении через коэффициент связи (линейное соединение).

Дополнительные указания по использованию таблиц кривых и каскадированию электронных редукторов и их синхронизации см.:
Другие функции:
Описание функций – Оси и шпиндели

Поведение электронного редуктора при Power On, RESET, смене режимов работы, поиске кадра

- После Power On **нет** активного соединения.
- Активные соединения сохраняются после RESET и смене режимов работы.
- При поиске кадра команды для включения, удаления, определения электронного редуктора не выполняются и не собираются, а пропускаются.

Системные переменные электронного редуктора

С помощью системных переменных электронного редуктора программа обработки детали может определять текущее состояния структуры осей электронного редуктора и при необходимости реагировать на них.

Системные переменные электронного редуктора обозначены следующим образом:

$AA_EG_ ...

или

$VA_EG_ ...

Дополнительная информация
Справочник по функциям – Системные переменные

4.16.4.3 Выключение электронного редуктора (EGOFS, EGOFC)

Для выключения активной структуры осей электронного редуктора существует 3 варианта.

Программирование

Вариант 1:

Синтаксис	Значение
EGOFS (ведомая ось)	Электронный редуктор отключается. Ведомая ось затормаживается до состояния покоя. Вызов запускает остановку предварительной обработки.

Вариант 2:

Синтаксис	Значение
EGOFS (ведомая ось,ведущая ось1, ..., ведущая ось5)	Такое параметрирование команды позволяет **выборочно** устранять влияние отдельных ведущих осей на движение ведомой оси.

Должна быть указана минимум одна ведущая ось. Воздействие указанных ведущих осей на ведомую ось целенаправленно отключается. Вызов запускает остановку предварительной обработки. Если еще остаются активные ведущие оси, то ведомая ось продолжает свое движение под их влиянием. Если все влияния ведущих осей отключаются таким способом, то ведомая ось затормаживается до состояния покоя.

Вариант 3:

Синтаксис	Значение
EGOFC(ведомый шпиндель)	Электронный редуктор отключается. Ведомый шпиндель продолжает движение с актуальным на момент времени отключения числом оборотов/скоростью. Вызов запускает остановку предварительной обработки.

Примечание

Этот вариант разрешен только для шпинделей.

4.16.4.4 Удаление определения электронного редуктора (EGDEL)

Необходимо отключить структуру осей электронного редуктора перед удалением его определения.

Программирование

Синтаксис	Значение
EGDEL(ведомая ось)	Определение соединения структуры осей удаляется. Снова появляется возможность повторного определения других структур осей с EGDEF до достижения макс. числа активированных одновременно структур осей. Вызов запускает остановку предварительной обработки.

4.16.4.5 Окружная подача (G95) / электронный редуктор (FPR)

С помощью команды FPR и ведомая ось электронного редуктора может быть указана в качестве определяющей подачу оси окружной подачи. На этот случай имеется следующее поведение:

- Подача зависит от заданной скорости ведомой оси электронного редуктора.
- Заданная скорость вычисляется из скоростей ходовых винтов и ведущих осей модуло (которые не являются траекторными осями) и их соответствующих коэффициентов связи.
- Компоненты скорости линейных ведущих осей или не модуло ведущих осей и наложенные движения ведомой оси не учитываются.

4.16.5 Синхронный шпиндель

В синхронном режиме существует один ходовой винт (LS) и один ведомый шпиндель (FS), т.н. **синхронная шпиндельная пара**. Ведомый шпиндель следует при активном соединении (синхронный режим) за движениями ходового винта согласно установленной функциональной связи.

Синхронные шпиндельные пары могут быть фиксировано сконфигурированы для любого станка посредством спец. для канала машинных данных, или определены спец. для приложения через программу обработки детали ЧПУ.. Для каждого канала ЧПУ возможно одновременное использование до 2 синхронных шпиндельных пар.

Соединение из программы обработки детали может быть

- определено или изменено
- включено
- отключено
- удалено

.

Кроме этого, в зависимости от версии ПО, можно

- ожидать условия синхронного хода
- изменить поведение смены кадра
- в качестве типа соединения выбрать либо соединение по заданному значению, либо соединение по фактическому значению или задать угловое смещение между ходовым винтом и ведомым шпинделем
- при включении соединения применить предшествующее программирование ведомого шпинделя
- исправить либо измеренную, либо уже известную погрешность синхронного хода

.

4.16.5.1 Синхронный шпиндель: Программирование (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC)

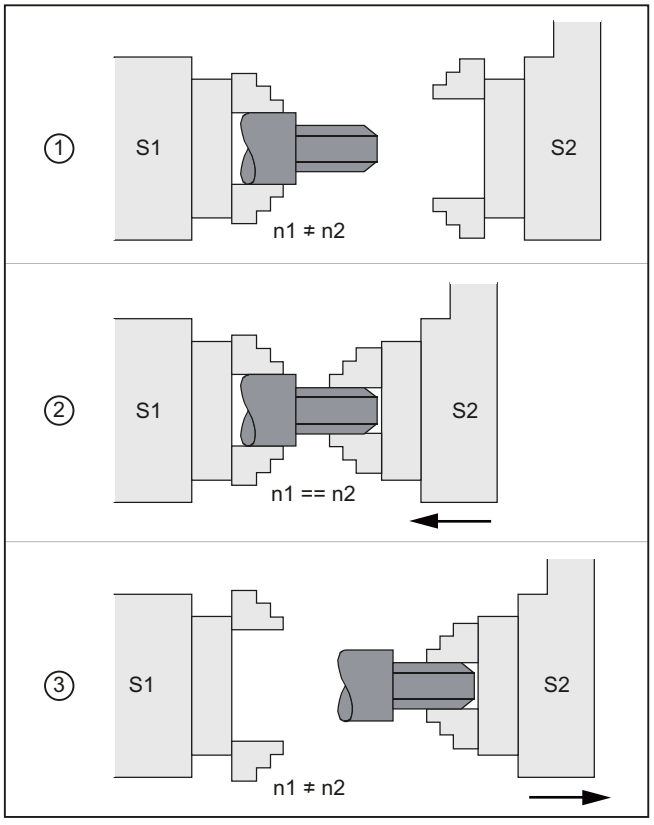
Функция "Синхронный шпиндель" позволяет выполнять синхронное по частоте вращения перемещение ведомого (FS) и ведущего шпинделя (LS) с программируемым передаточным отношением.

Функция предлагает следующие режимы:

- Синхронность по частоте вращения ($n_{FS} = n_{LS}$)
- Синхронность по положению ($\phi_{FS} = \phi_{LS}$)
- Синхронность по положению с угловым смещением ($\phi_{FS} = \phi_{LS} + \Delta\phi$)

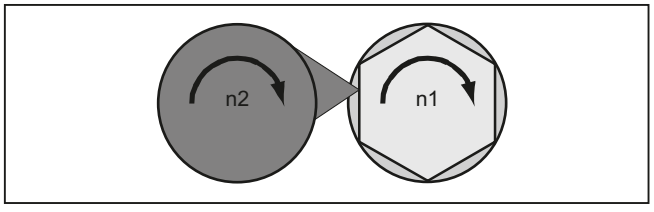
Примеры использования:

- Передача детали на лету, например, обработка задней стороны, передаточное отношение: 1:1



① Синхронизация частоты вращения
② Передача детали
③ Обработка "задней стороны"

- Многогранная обработка (полигональная токарная обработка), синхронная частота вращения, передаточное отношение: n_1:n_2



Синтаксис

```
COUPDEF(<FS>,<LS>,<ZFS>,<NLS>,<смена кадра>,<тип соединения>)
COUPON(<FS>,<LS>,<POSFS>)
COUPONC(<FS>,<LS>)
COUPOF(<FS>,<LS>,<POSFS>,<POSLS>)
COUPOFS(<FS>,<LS>)
COUPOFS(<FS>,<LS>,<POSFS>)
COUPRES(<FS>,<LS>)
```

```
COUPDEL(<FS>,<LS>)
WAITC(<FS>,<смена кадров>,<LS>,<смена кадров>)
```

Примечание

Сокращенная форма записи

Для операторов COUPOF, COUPOFS, COUPRES и COUPDEL возможна сокращенная форма записи без указания ведущего шпинделя.

Объяснение

COUPDEF:	Спец. для пользователя определение/изменение соединения
COUPON:	Включить соединение. Исходя из текущей частоты вращения, ведомый шпиндель синхронизируется с ведущим шпинделем
COUPONC:	При включении применить соединение с прежним программированием M3 S... или M4 S.... Дифф. частота вращения ведомого шпинделя применяется сразу же.
COUPOF:	Выключить соединение. • с мгновенной сменой кадров: COUPOF(<S2>,<S1>) • Смена кадров только после перехода через позицию(и) отключения <POSFS> или <POSLS>: COUPOF(<S2>,<S1>,<POSFS>) COUPOF(<S2>,<S1>,<POSFS>,<POSLS>)
COUPOFS:	Отключение соединения с остановкой ведомого шпинделя. Макс. быстрая смена кадра с мгновенной сменой кадра: COUPOFS(<S2>,<S1>) Смена кадра только после перехода позиции отключения: COUPOFS(<S2>,<S1>,<POSFS>)
COUPRES:	Сбросить параметры соединения на сконфигурированные MD и SD
COUPDEL:	Удалить определенное пользователем соединение
WAITC:	Ожидать условия синхронного хода (NOC на IPO при смене кадра отменяются)
<FS>:	Обозначение ведомого шпинделя
Опциональные параметры:	
<LS>:	Обозначение ведущего шпинделя Указание с номером шпинделя: например, S2, S1
<ZFS>, <NLS>:	Передаточное отношение между FS и LS. <ZFS> / <NLS> = числитель / знаменатель Предустановка: <ZFS> / <NLS> = 1.0 ; указание знаменателя не обязательно

<смена кадров>:	Параметры смены кадра Смена кадра выполняется:	
	"NOC"	сразу же
	"FINE"	при достижении "Синхронного хода точного"
	"COARSE"	при достижении "Синхронного хода грубого"
	"IPOSTOP"	при достижении IPOSTOP, т. е. после синхронного хода со стороны заданного значения (предустановка)
	Параметры смены кадра действуют модально	
<тип соединения>:	Тип соединения: Соединение между FS и LS	
	"DV"	Соединение по заданному значению (предустановка)
	"AV"	Соединение по фактическому значению
	"VV"	Соединение по скорости
	Тип соединения действует модально.	
<POSFS>:	Угловое смещение между ведущим и ведомым шпинделем	
	Диапазон значений:	0°... 359,999°
<POSFS>,<POSLS>:	Позиции отключения ведомого и ведущего шпинделя "Смена кадра разрешается после перехода через POS_{FS}, POS_{LS} "	
	Диапазон значений:	0°... 359,999°

Примеры

Работа с ведущим и ведомым шпинделем

Программный код	Комментарий
	Ведущий шпиндель = мастер-шпиндель = шпиндель 1
	Ведомый шпиндель = шпиндель 2
N05 M3 S3000 M2=4 S2=500	Ведущий шпиндель вращается с 3000 об/мин, ведомый шпиндель с 500 об/мин.
N10 COUPDEF(S2,S1,1,1,"NOC","Dv")	Определение соединения (возможно и конфигурирование).
...	
N70 SPCON	Перевести ведущий шпиндель винт в управление по положению (соединение по заданному значению)
N75 SPCON(2)	Перевести ведомый шпиндель в управление по положению.
N80 COUPON(S2,S1,45)	Соединение на позицию смещения = 45 градусов "на лету".
...	
N200 FA[S2]=100	Скорость позиционирования = 100 градусов/мин
N205 SPOS[2]=IC(-90)	Движение с наложением 90 градусов в отрицательном направлении.
N210 WAITC(S2,"Fine")	Ожидание синхронного хода "точного".
N212 G1 X... Y... F...	Обработка
...	
N215 SPOS[2]=IC(180)	Движение с наложением 180 градусов в положительном направлении.
N220 G4 S50	Время ожидания = 50 оборотов мастер-шпинделя

Программный код	Комментарий
N225 FA[S2]=0	Активировать сконфигурированную скорость (MD) .
N230 SPOS[2]=IC(-7200)	20 оборотов. Двигаться со сконфигурированной скоростью в отрицательном направлении.
...	
N350 COUPOF(S2,S1)	Разъединение "на лету", S=S2=3000
N355 SPOSA[2]=0	Остановить FS на нуле градусов.
N360 G0 X0 Y0	
N365 WAITS(2)	Ожидать шпиндель 2.
N370 M5	Остановить FS.
N375 M30	

Программирование дифф. скорости

Программный код	Комментарий
	Ведущий шпиндель = мастер-шпиндель = шпиндель 1
	Ведомый шпиндель = шпиндель 2
N01 M3 S500	Ведущий шпиндель вращается с 500 об/мин.
N02 M2=3 S2=300	Ведомый шпиндель вращается с 300 об/мин.
...	
N10 G4 F1	Время ожидания мастер-шпинделя.
N15 COUPDEF (S2,S1,-1)	Коэффициент связи с передаточным отношением -1:1
N20 COUPON(S2,S1)	Активировать связь. Частота вращения ведомого шпинделя получается из частоты вращения ведущего шпинделя и коэффициента связи.
...	
N26 M2=3 S2=100	Программирование дифф. частоты вращения

Примеры приема движения по дифф. скорости

1. Включить соединение с COUPON при предшествующем программировании ведомого шпинделя

Программный код	Комментарий
	Ведущий шпиндель = мастер-шпиндель = шпиндель 1
	Ведомый шпиндель = шпиндель 2
N05 M3 S100 M2=3 S2=200	Ведущий шпиндель вращается с 100 об/мин, ведомый шпиндель с 200 об/мин.
N10 G4 F5	Время ожидания = 5 секунд мастер-шпинделя
N15 COUPDEF(S2,S1,1)	Передаточное отношение FS к LS равно 1,0 (предустановка).
N20 COUPON(S2,S1)	Соединение с ведущим шпинделем "на лету".
N10 G4 F5	Ведомый шпиндель вращается с 100 об/мин.

2. Включить соединение с COUPONC при предшествующем программировании ведомого шпинделя

Программный код	Комментарий
	Ведущий шпиндель = мастер-шпиндель = шпиндель 1

Программный код	Комментарий
	Ведомый шпиндель = шпиндель 2
`N05 M3 S100 M2=3 S2=200`	Ведущий шпиндель вращается с 100 об/мин, ведомый шпиндель с 200 об/мин.
`N10 G4 F5`	Время ожидания = 5 секунд мастер-шпинделя
`N15 COUPDEF(S2,S1,1)`	Передаточное отношение `FS` к `LS` равно 1,0 (предустановка).
`N20 COUPONC(S2,S1)`	Соединение с ведущим шпинделем "на лету" и применить прежнюю частоту вращения к `S2`.
`N10 G4 F5`	`S2` вращается с 100об/мин + 200об/мин = 300об/мин

3. Включить соединение с `COUPON` при стоящем ведомом шпинделе

Программный код	Комментарий
	Ведущий шпиндель = мастер-шпиндель = шпиндель 1
	Ведомый шпиндель = шпиндель 2
`N05 SPOS=10 SPOS[2]=20`	Ведомый шпиндель `S2` в режиме позиционирования.
`N15 COUPDEF(S2,S1,1)`	Передаточное отношение `FS` к `LS` равно 1,0 (предустановка).
`N20 COUPON(S2,S1)`	Соединение с ведущим шпинделем "на лету".
`N10 G4 F1`	Соединение закрывается, `S2` останавливается на 20 градусах.

4. Включить соединение с `COUPONC` при стоящем ведомом шпинделе

Примечание

Режим позиционирования или осевой режим

Если ведомый шпиндель перед соединением находится в режиме позиционирования или осевом режиме, то поведение ведомого шпинделя при `COUPON(<FS>,<LS>)` и `COUPONC(<FS>,<LS>)` идентично.

Примечание

Ведущий шпиндель в осевом режиме

Если ведущий шпиндель перед определением соединения находится в осевом режиме, то и после включения соединения действует предельное значение скорости из машинных данных:

MD32000 $MA_MAX_AX_VELO (макс. скорость оси)

Во избежание такого поведения, ось перед определением соединения должна быть переведена в шпиндельный режим (`M3 S...` или `M4 S...`).

Дополнительная информация

Сконфигурированное соединение

В случае сконфигурированного соединения LS и FS определяются через машинные данные. Сконфигурированные шпиндели не могут быть изменены в программе

обработки детали. Параметрирование соединения возможно в программе обработки детали с COUPDEF (условие: защита записи не установлена).

Определенное пользователем соединение

С помощью COUPDEF можно заново определить или изменить соединение в программе обработки детали. Если соединение уже активно, то перед определением нового соединения оно должно быть сначала удалено с COUPDEL.

Соединение полностью определяется через:

COUPDEF(<FS>,<LS>,<ÜFS>,<ÜLS>, параметры смены кадра, тип соединения)

Ведомый шпиндель (FS) и ведущий шпиндель (LS)

С именами осей для FS и LS соединение определяется однозначно. Имена осей должны программироваться с каждым оператором COUPDEF. Другие параметры соединения действуют модально и должно повторно программироваться только после их изменения.

Пример:

```
COUPDEF(S2,S1)
```

Передаточное отношение

Передаточное отношение указывается как отношение по скорости между FS и LS:

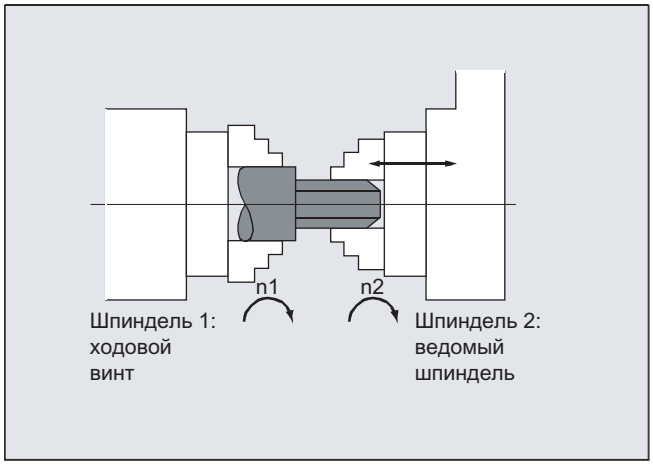
Ведомый шпиндель / ведущий шпиндель = числитель / знаменатель

Числитель должен быть запрограммирован. Программирования знаменателя не требуется. Тогда для знаменателя устанавливается значение по умолчанию 1.0.

Пример:

Ведомый шпиндель S2 и ведущий шпиндель S1, передаточное отношение = 1 / 1

```
COUPDEF(S2, S1, 1.0)
```



Примечание

Передаточное отношение может изменяться и при включенном соединении и вращающихся шпинделях.

Параметры смены кадра NOC, FINE, COARSE, IPOSTOP

При программировании параметров смены кадров возможна следующая сокращенная форма записи:

- "NO": сразу же (предустановка)
- "FI": при достижении "Синхронного хода точного"
- "CO": при достижении "Синхронного хода грубого"
- "IP": при достижении IPOSTOP, т. е. после синхронного хода со стороны заданного значения

Тип соединения

Примечание

Тип соединения может изменяться только при отключенном соединении!

Включение синхронного режима COUPON, <POSFS>

- Включение соединения с любым угловым смещением между FS и LS:
  - COUPON(S2,S1)
  - COUPON(S2)
- Включение соединения с угловым смещением <POSFS>
  <POSFS> относится к позиции 0° ведущего шпинделя в положительном направлении вращения
  Диапазон значений <POSFS>: 0°... 359,999°
  - COUPON(S2,S1,30)

Примечание

Угловое смещение может изменяться и при активном соединении.

Позиционирование ведомого шпинделя

И при включенном соединении синхронных шпинделей возможно позиционирование FS независимо от LS в диапазоне ±180°.

- Позиционирование шпинделя FS с SPOS
  Пример: SPOS[2]=IC(-90)
  Дополнительная информация по SPOS представлена в
  Дополнительная информация
  Позиционирование шпинделей (SPOS, SPOSA, M19, M70, WAITS) (Страница 123)

Дифф. скорость

Дифф. скорость возникает в режиме управления по скорости и активном соединении синхронных шпинделей через наложение со знаком скорости FS на основе движения LS и скорости FS на основе программирования шпинделя:

- Соединение синхронных шпинделей с `COUPONC`
- `S<FS>=`<скорость> [`M<FS>=`<направление вращения>]

Примечание

Граничные условия

- При направлении вращения `M3` / `M4` необходимо повторное программирование и скорости `S....`
- Наложения скорости шпинделя (`M`<направление вращения> `S<FS>`) через движение LS в соединении синхронных шпинделей `COUPONC` активируется только при разрешенном наложении.
- Динамика ведущего шпинделя должна быть ограничена настолько, чтобы при наложении ведомого шпинделя не были бы превышена его предельные значения динамики.

Дополнительная информация по дифф. скорости вращения представлена в
Дополнительная информация
Описание функций – Оси и шпиндели

Скорость, ускорение: FA, ACC, OVRA, VELOLIMA

Осевая скорость и ускорение ведомого шпинделя могут программироваться с:

- `FA[SPI(S<n>)]` или `FA[S<n>]` (осевая скорость)
- `ACC[SPI(S<n>)]` или `ACC[S<n>]` (осевое ускорение)
- `OVRA[SPI(S<n>)]` или `OVRA[S<n>]` (осевая процентовка)
- `VELOLIMA[SPI(S<n>)]` или `VELOLIMA[S<n>]` (осевое превышение или снижение скорости)

С `<n>` = 1, 2, 3, ... (номер ведомого шпинделя)

Дополнительная информация
Регулирование подачи (Страница 110)

Примечание

Понижение превышения макс. осевого рывка для шпинделей не действует.

Дополнительная информация по осевой динамике представлена в
Дополнительная информация
Описание функций – Оси и шпиндели

Программируемые параметры смена кадра WAITC

С помощью `WAITC` параметры смены кадра, к примеру, после изменения параметров соединения или процессов позиционирования, могут задаваться с различными условиями синхронного хода (грубый, точный, `IPOSTOP`). Если условия синхронного

хода не указаны, то действуют указанные при определении COUPDEF параметры смены кадра.

Примеры

- Ожидание достижения условия синхронного хода FINE для ведомого шпинделя S2 и COARSE для ведомого шпинделя S4: WAITC(S2,"FINE",S4,"COARSE")
- Ожидание достижения условия синхронного хода согласно COUPDEF: WAITC( )

Отключить соединение COUPOF

С помощью COUPOF могут быть заданы параметры отключения соединения:

- Отключение соединения с мгновенной сменой кадра:
  - COUPOF(S2,S1) (с указанием ведущего шпинделя)
  - COUPOF(S2) (с указанием ведущего шпинделя)
- Отключение соединения после перехода позиций отключения. Смена кадров происходит после перехода позиций отключения.
  - COUPOF(S2,S1,150) (позиция отключения FS: 150°)
  - COUPOF(S2,S1,150,30) (позиция отключения FS: 150°, LS: 30°)

Отключение соединения с остановкой ведомого шпинделя COUPOFS

С помощью COUPOFS могут быть заданы параметры отключения соединения с остановкой ведомого шпинделя:

- Отключение соединения с остановкой ведомого шпинделя и мгновенной сменой кадра:
  - COUPOFS(S2,S1) (с указанием ведущего шпинделя)
  - COUPOFS(S2) (с указанием ведущего шпинделя)
- Отключение соединения после перехода позиций отключения с остановкой ведомого шпинделя. Смена кадров происходит после перехода позиций отключения.
  - COUPOFS(S2,S1,150) (позиция отключения FS: 150°)

Удаление соединений COUPDEL

С COUPDEL соединение удаляется:

- COUPDEL(S2,S1) (с указанием ведущего шпинделя)
- COUPDEL(S2) (с указанием ведущего шпинделя)

Сброс параметров соединения COUPRES

С COUPRES активируются спараметрированные в машинных и установочных данных значения соединения:

- COUPRES(S2,S1) (с указанием ведущего шпинделя)
- COUPRES(S2) (с указанием ведущего шпинделя)

Системные переменные

- Актуальное состояние соединения ведомого шпинделя
  Текущее состояние соединения ведомого шпинделя может быть считано в битовой кодировке через:
  <значение> = $AA_COUP_ACT[<FS>]

Бит	<значение>	Объяснение
-	0	нет активного соединения
2	4	активно соединение синхронных шпинделей
Указание • Все другие значения относятся к осевому режиму • Если шпиндель является ведомым шпинделем из нескольких соединений, в качестве значения состояния соединения всех соединений возвращается суммарное соединение.		

- Актуальное угловое смещение
  Текущее угловое смещение ведомого шпинделя к ведущему шпинделю может быть считано через:
  - $AA_COUP_OFFS[<FS>] (угловое смещение со стороны заданного значения)
  - $VA_COUP_OFFS[<FS>] (угловое смещение со стороны фактического значения)

  Пример использования
  Коррекция разницы угловых смещений в программе ЧПУ после отмены режима слежения:
  Разность угловых смещений = запрограммированное угловое смещение - системная переменная

Дополнительная информация

Подробную информацию по системным переменным см.:

Справочник по параметрированию "Системные переменные"

4.16.6 Базовое соединение осей (CP...)

"Базовое соединение осей" это общая функция соединения, объединяющая в себе все свойства существующих типов соединений (совместное движение, соединение по главному значению, электронный редуктор и синхронный шпиндель).

Функция позволяет выполнять гибкое программирование:

- Пользователь может свободно выбирать необходимые для его приложения свойства соединения (модульный принцип).
- Каждое свойство соединения может быть запрограммировано по отдельности.
- Свойства определенного соединения (например, коэффициент связи) могут изменяться.
- Последующее использование других свойств соединения возможно.

- Базовая система ведомой оси (базовая кинематическая система или система координат станка) может программироваться.
- Определенные свойства соединения могут программироваться и в синхронных действиях.
  Дополнительная информация: Описание функций "Синхронные действия"

Примечание

Прежние вызовы соединений для совместного движения (TRAIL*), соединения по главному значению (LEAD*), электронного редуктора (EG*) и синхронного шпинделя (COUP*) продолжают поддерживаться через адаптивные циклы.

Обзор всех кодовых слов и свойств соединений

Следующая таблица предлагает обзор всех кодовых слов базового соединения осей и программируемых с их помощью свойств соединения:

Кодовое слово	Свойство соединения / значение	Синтаксис
CPDEF	Создание модуля сопряжения	CPDEF=(<FAx>)
CPDEL	Удаление модуля сопряжения	CPDEL=(<FAx>)
CPLA	Определение ведущей оси	CPLA[<FAx>]=(<LAx>)
CPLDEF	Определение ведущей оси и создание модуля сопряжения (также возможно с CPDEF + CPLA)	CPLDEF[<FAx>]=(<LAx>) или CPDEF=(<FAx>) CPLA[<FAx>]=(<LAx>)
CPLDEL	Удаление ведущей оси модуля сопряжения (также возможно с CPDEL + CPLA)	CPLDEL[<FAx>]=(<LAx>) или CPDEL=(<FAx>) CPLA[<FAx>]=(<LAx>)
CPON	Включение модуля сопряжения	CPON=(<FAx>)
CPOF	Выключение модуля сопряжения	CPOF=(<FAx>)
CPLON	Включение ведущей оси модуля сопряжения	CPLON[<FAx>]=<LAx>
CPLOF	Выключение ведущей оси модуля сопряжения	CPLOF[<FAx>]=<LAx>
CPLNUM	Числитель коэффициента связи	CPLNUM[FAx,LAx]=<значение>
CPLDEN	Знаменатель коэффициента связи	CPLDEN[FAx,LAx]=<значение>
CPLCTID	Номер таблицы кривых	CPLCTID[FAx,LAx]=<значение>

Кодовое слово	**Свойство соединения / значение**	**Синтаксис**		
CPLSETVAL	Источник соединения	CPLSETVAL[FAx,LAx]="<источник соединения>"		
		"<Источник соединения>":	"CMDPOS"	Соединение по заданному значению
			"CMDVEL"	Соединение по скорости
			"ACTPOS"	Соединение по фактическому значению
CPFRS	Исходная система координат	CPFRS[FAx]="<источник координат>"		
		"<Источник координат>":	"BCS"	Базовая кинематическая система
			"MCS"	Система координат станка
CPBC	Критерий смены кадров	CPBC[FAx]="<критерий смены кадра>"		
		"<Критерий смены кадра>":	"NOC"	Смена кадра выполняется независимо от состояния соединения.
			"IPOSTOP"	Смена кадра при синхронном ходе со стороны заданного значения.
			"COARSE"	Смена кадра при "Синхронном ходе грубом" со стороны фактического значения.
			"FINE"	Смена кадра при "Синхронном ходе точном" со стороны фактического значения.
CPFPOS + CPON	Синхронная позиция ведомой оси при включении	CPON=FAx CPFPOS[FAx]=<значение>		
CPLPOS + CPON	Синхронная позиция ведущей оси при включении	CPLPOS[FAx,LAx]=<значение>		

Кодовое слово	Свойство соединения / значение	Синтаксис		
CPFMSON	Режим синхронизации	CPFMSON[FAx]="<режим синхронизации>"		
		"<Режим синхронизации>":	"CFAST"	Соединение замыкается оптимально по времени.
			"CCOARSE"	Соединение включается только в том случае, если требуемая по правилу связи позиция ведомой оси находится в области текущей позиции ведомой оси.
			"NTGT"	Оптимизированный по времени подвод к следующей впадине между зубьями.
			"NTGP"	Оптимизированный по пути подвод к следующей впадине между зубьями.
			"NRGT"	Оптимизированный по времени подвод к следующему сегменту согласно отношению из числа ступеней к числу зубьев.
			"NRGP"	Оптимизированный по пути подвод к следующему сегменту согласно отношению из числа ступеней к числу зубьев.
			"ACN"	Только для круговых осей! Круговая ось подводится к синхронной позиции в отрицательном осевом направлении. Синхронизация происходит сразу же.
			"ACP"	Только для круговых осей! Круговая ось подводится к синхронной позиции в положительном осевом направлении. Синхронизация происходит сразу же.
			"DCT"	Только для круговых осей! Круговая ось подводится к запрограммированной синхронной позиции с оптимизацией по времени. Синхронизация происходит сразу же.
			"DCP"	Только для круговых осей!

Кодовое слово	Свойство соединения / значение	Синтаксис		
				Круговая ось подводится к запрограммированной синхронной позиции с оптимизацией по пути. Синхронизация происходит сразу же.
CPFMON	Поведение ведомой оси при включении	CPFMON[FAx]="<поведение при включении>"		
		"<Поведение при включении>":	"STOP"	Только для шпинделей! Активное движение ведомого шпинделя останавливается перед включением.
			"CONT"	Только для шпинделей и осей главного хода! Текущее движение ведомой оси/шпинделя берется в соединение как движение запуска.
			"ADD"	Только для шпинделей! Компоненты движения соединения действуют дополнительно к текущему наложенному движению, т. е. текущее движение ведомой оси/шпинделя сохраняется как наложенное движение.
CPFMOF	Поведение ведомой оси при полном выключении	CPFMOF[FAx]="<поведение при выключении>"		
		"<Поведение при выключении>":	"STOP"	Останов ведомой оси/шпинделя. Активное наложенное движение затормаживается до состояния покоя. После этого соединение размыкается.
			"CONT"	Только для шпинделей и осей главного хода! Ведомый шпиндель продолжает движение с актуальной на момент времени отключения частотой вращения/скоростью.
CPFPOS + CPOF	Позиция отключения ведомой оси при выключении	CPOF=(FAx) CPFPOS[FAx]=<значение>		

Кодовое слово	**Свойство соединения / значение**	**Синтаксис**		
CPMRESET	Поведение соединения при RESET	CPMRESET[FAx]="<реакция на Reset>"		
		"<Реакция на Reset>":	"NONE"	Текущее состояние соединения сохраняется.
			"ON"	Если соответствующий модуль сопряжения создан, то соединение включается. Активируются все определенные отношения ведущей оси. Это происходит и тогда, когда все или часть этих отношений ведущей оси уже активны, т. е. при полностью активированном соединении выполняется повторная синхронизация.
			"OF"	Активное наложенное движение затормаживается до состояния покоя. После этого соединение выключается. Если соответствующий модуль сопряжения был создан без явного определения (CPDEF), то модуль сопряжения удаляется. В ином случае он сохраняется, т. е. может использоваться и дальше.
			"OFC"	Возможно только для шпинделей! Ведомый шпиндель продолжает движение с актуальной на момент времени отключения частотой вращения/скоростью. Соединение выключается. Если соответствующий модуль сопряжения был создан без явного определения (CPDEF), то модуль сопряжения удаляется. В ином случае он сохраняется, т. е. может использоваться и дальше.
			"DEL"	Активное наложенное движение затормаживается до состояния покоя. После этого соединение деактивируется и удаляется.

Кодовое слово	Свойство соединения / значение	Синтаксис		
			"DELC"	Возможно только для шпинделей! Ведомый шпиндель продолжает движение с актуальной на момент времени отключения частотой вращения/скоростью. Соединение деактивируется и после удаляется.
CPMSTART	Поведение соединения при запуске программы обработки детали	CPMSTART[FAx]="<поведение при запуске>"		
		"<Поведение при запуске>":	"NONE"	Текущее состояние соединения сохраняется.
			"ON"	Соединение включается. Активируются все определенные отношения ведущей оси. Это происходит и тогда, когда все или часть этих отношений ведущей оси уже активны, т. е. при полностью активированном соединении выполняется повторная синхронизация.
			"OF"	Соединение выключается. Если соответствующий модуль сопряжения был создан без явного определения (CPDEF), то модуль сопряжения удаляется. В ином случае он сохраняется, т. е. может использоваться и дальше.
			"DEL"	Соединение деактивируется и после удаляется.
CPMPRT	Поведение соединения при запуске программы обработки детали под поиск кадра через тестирование программы	CPMPRT[FAx]="<поведение при запуске>"		
		"<Поведение при запуске>":	см. CPMSTART	
CPLINTR	Значение смещения для входного значения ведущей оси	CPLINTR[FAx,LAx]=<значение>		
CPLINSC	Коэффициент масштабирования для входного значения ведущей оси	CPLINSC[FAx,LAx]=<значение>		
CPLOUTTR	Значение смещения для выходного значения соединения	CPLOUTTR[FAx,LAx]=<значение>		
CPLOUTSC	Коэффициент масштабирования для выходного значения соединения	CPLOUTSC[FAx,LAx]=<значение>		

Кодовое слово	Свойство соединения / значение	Синтаксис		
CPSYNCOP	Пороговое значение для позиционного синхронного хода "грубого"	CPSYNCOP[FAx]= <значение>		
CPSYNFIP	Пороговое значение для позиционного синхронного хода "точного"	CPSYNFIP[FAx]=<значение>		
CPSYNCOP2	Второе пороговое значение для позиционного синхронного хода "грубого"	CPSYNCOP2[FAx]=<значение>		
CPSYNFIP2	Второе пороговое значение для позиционного синхронного хода "точного"	CPSYNFIP2[FAx]=<значение>		
CPSYNCOV	Пороговое значение для скоростного синхронного хода "грубого"	CPSYNCOV[FAx]=<значение>		
CPSYNFIV	Пороговое значение для скоростного синхронного хода "точного"	CPSYNFIV[FAx]=<значение>		
CPMBRAKE	Поведение ведомой оси при определенных сигналах и командах останова	CPMBRAKE[FAx]=<биткодированное значение>		
CPMVDI	Реакция ведомой оси на определенные интерфейсные сигналы ЧПУ/PLC	CPMVDI[FAx]=<биткодированное значение>		
CPMALARM	Блокировка вывода спец. относящихся к соединению аварийных сообщений	CPMALARM[FAx]=<биткодированное значение>		
CPSETTYPE	Тип соединения	CPSETTYPE[FAx]="<тип соединения>"		
		"<тип соединения>":	"CP"	Свободная программируемость
			"TRAIL"	Тип соединения "совместное движение"
			"LEAD"	Тип соединения "соединение по главному значению"
			"EG"	Тип соединения "электронный редуктор"
			"COUP"	Тип соединения "синхронный шпиндель"

FAx: ведомая ось/ведомый шпиндель

LAx: ведущая ось/ведущий шпиндель

Примечание

Свойства соединения, которые не программируются явно (в программе обработки детали или в синхронных действиях), активируются со своими стандратным настройкам.

В зависимости от настройки кодового слова `CPSETTYPE` вместо стандартных настроек (`CPSETTYPE="CP"`) могут быть активированы и предустановленные свойства соединения.

Дополнительная информация

Подробную информацию по базовым соединениям осей можно найти в:

- Описание функций – Оси и шпиндели

4.16.7 Тангенциальное управление

4.16.7.1 Определение соединения (TANG)

С помощью предварительно определенной процедуры TANG(...) определяется тангенциальное соединение между круговой осью в качестве ведомой оси и двумя геометрическими осями в качестве ведущих осей. При этом ведомая ось непрерывно выравнивается относительно касательной к траектории ведущей оси.

Примечание

коэффициент связи

Явного программирования коэффициента связи 1 не требуется.
Коэффициент связи -1 поворачивает направление тангенциальной оси.

Синтаксис

`TANG(`<ведомая ось>, <ведущая ось_1>, <ведущая ось_2>, <коэффициент связи>, <система координат>, <оптимизация>)

Объяснение

`TANG(...)`:	Определение тангенциального соединения	
<ведомая ось>:	Имя ведомой оси (круговая ось)	
	Тип данных:	AXIS
	Диапазон значений:	Имя оси канала

<ведущая ось_1> <ведущая ось_2>	Имена ведущих осей (геометрических осей) [1]		
	Тип данных:	AXIS	
	Диапазон значений:	Имена геометрических осей канала	
<коэффициент связи>:	Коэффициент n изменения угла ведомой оси относительно изменения касательной к траектории ведущих осей: Изменение угла$_{\text{ведомая ось}}$ = изменение угла$_{\text{касательная к траектории}}$ * n		
	Тип данных:	REAL	
	Значение по умолчанию:	1,0	
<Система координат>:	Действующая система координат [2]		
	Тип данных:	CHAR	
	Значение:	"B":	Базовая система координат (значение по умолчанию)
		"W":	Система координат детали (недоступно)
<Оптимизация>:	Вид оптимизации		
	Тип данных:	CHAR	
	Значение:	"S":	Стандарт (значение по умолчанию) Динамика круговой оси не оказывает обратного действия на ведущие оси. Если динамика круговой оси превышает значение, необходимое для слежения, это перемещение отличается достаточной точностью. Если динамика круговой оси недостаточно высока для отслеживания изменения касательной к траектории, центровка круговой оси вдоль неопределенного пути перешлифовки будет отличаться от заданной центровки.
		"P":	Динамика круговой оси учитывается при планировании траектории ведущих осей. Для этого при включении тангенциального соединения командой TANGON() необходимо указать два дополнительных параметра: • Путь перешлифовки • Угловой допуск См. главу "Включение соединения (TANGON) (Страница 985)" **Примечание** В связи с кинематическими трансформациями рекомендуется использовать вид оптимизации "P".
Указание Значения по умолчанию не требуют явного программирования.			

[1] **Примечание**

В качестве ведущих осей тангенциального соединения следует использовать геометрические оси, которые относительно исходного положения станка проходят запрограммированную траекторию в системе координат станка (MCS). Если, например, на фрезеровальном станке с качающейся головкой применяется поворотный цикл CYCLE800, то, в зависимости от конфигурации цикла, происходит интерполяция в WKS, например, по геометрическим осям X и Y. Тангенциальное соединение должно быть, напротив, определено с геометрическими осями в качестве ведущих осей, которые проходят в WCS запрограммированную траекторию. Для этого в качестве ведущих осей следует использовать геометрические оси в **неповернутом** состоянии станка.

[2] **Примечание**

Базовая система координат (BKS), в отличие от MCS, не может быть повернута. Если, например ,.через команду ROT или цикл поворота CYCLE800 БСК поворачивается, то тангенциальное управление перестает работать корректно.

4.16.7.2 Включение создания промежуточного кадра (TLIFT)

Если изменение касательной ведомой оси в какой-либо точке запрограммированной траектории ведущих осей превышает предельное значение, заданное в машинных данных MD37400 $MA_EPS_TLIFT_TANG_STEP, то дальнейшее планирование траектории будет зависеть от настроенных параметров угла. Без использования предварительно определенной процедуры TLIFT(...) перемещение будет осуществляться в соответствии с методом перешлифовки, заданным в TANG(...) (Страница 982) и TANGON(...) (Страница 985).

Включение создания промежуточного кадра

При программировании TLIFT(...) в связи с TANG(...) в ходе предварительной обработки при распознавании угла в этом месте траектории вводится промежуточный кадр, автоматически созданный системой управления.

При выполнении программы по достижении промежуточного кадра ведущие оси приостанавливаются. В рамках промежуточного кадра ведомая ось вращается с максимальной динамикой в направлении касательной к траектории следующего кадра. После этого ведущие оси перемещаются далее по запрограммированной траектории.

Отключение создания промежуточного кадра

Для выключения создания промежуточного кадра необходимо переопределить тангенциальное соединение посредством TANG(...), но без последующего включения создания промежуточного кадра при помощи TLIFT(...).

Синтаксис

`TLIFT(`<ведомая ось>`)`

Значение

TLIFT(...):	Включение распознавания углов с расчетом промежуточного кадра	
<ведомая ось>:	Имя ведомой оси (круговая ось)	
	Тип данных:	AXIS
	Диапазон значений:	Имя оси канала

Скорость вращения ведомой оси

Траекторная ось

Если ведомая ось перед активацией тангенциального соединения уже перемещалась в качестве траекторной оси, вращение в промежуточном кадре происходит в качестве траекторной оси.

При задании исходного радиуса при помощи FGREF[<ось>]=0.001 вращение осуществляется с параметрированной максимальной осевой скоростью:

MD32000 $MA_MAX_AX_VELO[<ведомая ось>]

Позиционирующая ось

Если ведомая ось перед активацией тангенциального соединения еще не перемещалась в качестве траекторной оси, вращение в промежуточном кадре происходит в качестве позиционирующей оси.

При этом вращение осуществляется с параметрированной скоростью позиционирующей оси:

MD32060 $MA_POS_AX_VELO[<ведомая ось>]

4.16.7.3 Включение соединения (TANGON)

Предварительно определенная процедура TANGON(...) включает определенное ранее в TANG(...) (Страница 982) тангенциальное соединение. При этом ведомая ось непрерывно выравнивается относительно касательной к траектории ведущей оси при последующем перемещении ведущих осей.

Угол ведомой оси

Угол, занимаемый ведомой осью относительно касательной к траектории, зависит от заданного в TANG(...) коэффициента трансформации, заданного в машинных данных MD37402 $MA_TANG_OFFSET угла смещения и действующего дополнительного к нему угла смещения, заданного через TANGON(...).

Оптимизация «P»

Если при определении тангенциального соединения (TANG(...)) в качестве параметра оптимизации будет указано значение «P», то при включении соединения потребуется указать параметр «Путь перешлифовки» и, возможно, параметр «Угловой допуск».

Если в качестве углового допуска будет задано значение 0, действует только параметр «Путь перешлифовки».

Если в качестве углового допуска будет задано значение выше 0, то эффективный путь перешлифовки будет складываться из минимального параметрированного пути перешлифовки и пути перешлифовки на основании параметрированного углового допуска.

Если динамика ведомой оси недостаточна для отслеживания параметрированных условий, то скорость движения ведущих осей по траектории будет соответственно снижена.

Синтаксис

TANGON(<ведомая ось>, <угол смещения>, <путь перешлифовки>, <угловой допуск>)

Значение

TANGON(...):	Включение тангенциального соединения	
<ведомая ось>:	Имя ведомой оси (круговая ось)	
	Тип данных:	AXIS
	Диапазон значений:	Имя оси канала
<Угол смещения>:	Угол смещения ведомой оси относительно касательной к траектории Исходной точкой служит нулевая точка круговой оси.	
	Тип данных:	REAL
<Путь перешлифовки>:	Макс. допустимый путь перешлифовки Если путь перешлифовки вследствие динамических условий окажется больше, то скорость движения ведущих осей по траектории будет снижена.	
	Тип данных:	REAL
<угловой допуск>:	Макс. допустимый допуск относительно заданного угла между нулевым положением ведомой оси и касательной к траектории	
	Тип данных:	REAL

4.16.7.4 Выключение соединения (TANGOF)

Предварительно определенная процедура TANGOF(...) выключает тангенциальное соединение, определенное в TANG(...) (Страница 982) и включенное в TANGON(...) (Страница 985). После этого ведомая ось прекращает выравниваться по касательной к траектории ведущей оси. Соединение ведомой оси с ведущими осям сохраняется даже после выключения, что препятствует использованию, например, следующих функций:

- Смена плоскостей
- Переключение геометрических осей
- Определение нового тангенциального соединения для ведомой оси

Полный разрыв соединения ведомой и ведущих осей происходит только после удаления соединения посредством TANGDEL(...) (Страница 987).

Программирование

TANGOF(<ведомая ось>)

Значение

TANGOF(...):	Выключение тангенциального соединения	
<ведомая ось>:	Имя ведомой оси (круговая ось)	
	Тип данных:	AXIS
	Диапазон значений:	Имя оси канала

4.16.7.5 Удаление соединения (TANGDEL)

Тангенциальное соединение, определенное в TANG(...) (Страница 982), сохраняется даже после отключения тангенциального соединения в TANGOF(...) (Страница 986). Существующее тангенциальное соединение препятствует выполнению, например, следующих пунктов:

- Смена плоскостей
- Переключение геометрических осей
- Определение нового тангенциального соединения для ведомой оси

Предварительно определенная процедура TANGDEL(...) после выключения тангенциального соединения в TANGOF(...) удаляет имеющее тангенциальное соединение.

Синтаксис

TANGDEL(<ведомая ось>)

Значение

TANGDEL(...):	Удаление тангенциального соединения, определенного в TANG()	
	Активность:	Покадрово
<ведомая ось>:	Имя ведомой оси, тангенциальное соединение которой нужно удалить	
	Тип данных:	AXIS
	Диапазон значений:	Имя оси канала

Примеры

Смена ведущей оси

Перед тем, как определить для ведомой оси новое тангенциальное соединение с другой ведущей осью, потребуется предварительно удалить имеющееся тангенциальное соединение.

Программный код	Комментарий
N10 **TANG**(A, X, Y, 1)	; определение тангенциального соединения для ведомой оси A: A к X и Y
N20 TANGON(A)	; включение тангенциального соединения для ведомой оси A
N30 X10 Y20	
...	
N80 TANGOF(A)	; выключение тангенциального соединения для ведомой оси A
N90 **TANGDEL**(A)	; удаление тангенциального соединения для ведомой оси A
...	
N120 **TANG**(A, X, Z)	; определение нового тангенциального соединения для ведомой оси A
N130 TANGON(A)	; включение нового тангенциального соединения для ведомой оси A
...	

Переключение геометрических осей

Перед переключением геометрической оси для имеющегося соединения необходимо предварительно удалить соединение.

Программный код	Комментарий
N10 GEOAX(2, Y1)	; 2. Геометрическая ось = ось станка Y1
N20 TANG(A, X, Y)	; определение тангенциального соединения для ведомой оси A
N30 TANGON(A, 90)	; включение тангенциального соединения для ведомой оси A
N40 G2 F8000 X0 Y0 I0 J50	; кадр движения
N50 TANGOF(A)	; выключение тангенциального соединения для ведомой оси A
N60 **TANGDEL**(A)	; удаление тангенциального соединения для ведомой оси A
N70 **GEOAX**(2, Y2)	; 2. Геометрическая ось = ось станка Y2
N80 TANG(A, X, Y)	; определение нового тангенциального соединения для ведомой оси A
N90 TANGON(A, 90)	; включение нового тангенциального соединения для ведомой оси A
...	

4.16.8 Соединение Master/Slave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS)

"Соединение Master/Slave" обеспечивает:

- Соединение Slave-осей с Master-осью в состоянии покоя участвующих осей.
- Соединение и разъединение **вращающихся**, управляемых по скорости шпинделей.
- Динамическое конфигурирование.

Примечание

Режим позиционирования

У осей и шпинделей в режиме позиционирования соединение замыкается и разъединяется только в состоянии покоя.

Синтаксис

```
MASLON(<Slave_1>,<Slave_2>,...)
MASLOF(<Slave_1>,<Slave_2>,...)
MASLOFS(<Slave_1>,<Slave_2>,...)
```

Динамическое конфигурирование:

```
MASLDEF(<Slave_1>,<Slave_2>,...,<Master>)
MASLDEL(<Slave_1>,<Slave_2>,...)
```

Объяснение

MASLON:	Включение временного соединения Master/Slave	
	<Slave_x>,...:	Slave-ось 1 ... n
MASLOF:	Выключение активного соединения Master/Slave	
	<Slave_1>,...:	Slave-ось 1 ... n
MASLOFS:	Разрыв соединения Master/Slave и автоматическое затормаживание Slave-шпинделей (см. указание "Поведение шпинделей при соединении в режиме управления по скорости"!)	
	<Slave_1>,...:	Slave-ось 1 ... n
MASLDEF:	Создание/изменение структуры Master/Slave из программы обработки детали	
	<Slave_1>,...:	Slave-ось 1 ... n
	<Master>:	Master-ось
MASLDEL:	Разорвать соединение Master/Slave и удалить определение структуры	
	<Slave_1>,...:	Slave-ось 1 ... n
	Примечание: Сконфигурированные в машинных данных определения Master/Slave сохраняются.	

Примечание

Поведение шпинделей при соединении в режиме управления по скорости

Для шпинделей в режиме управления по скорости поведение при соединении MASLON, MASLOF, MASLOFS и MASLDEL определяется явно через следующие машинные данные:

MD37263 $MA_MS_SPIND_COUPLING_MODE

В стандартной настройке с MD 37263 = 0 соединение и разъединение осей Slave осуществляется исключительно в состоянии покоя участвующих осей. MASLOFS соответствует MASLOF.

При MD37263 = 1 оператор соединения выполняется сразу же, т. е. и при движении. Соединение при MASLON сразу же замыкается, а при MASLOFS или MASLOF сразу же размыкается. Вращающиеся при этом Slave-шпиндели сохраняют при MASLOF свои скорости до повторного программирования скорости. При MASLOFS, напротив, они автоматически затормаживаются.

Примечание

При MASLOF/MASLOFS не явная остановка предварительной обработки отсутствует. Из-за отсутствия остановки предварительной обработки системные переменные $P до момента повторного программирования не выводят для осей Slave обновленных значений.

Примечание

Для Slave-осей фактическое значение через PRESETON может быть синхронизировано с тем же значением Master-оси. Для этого необходимо на короткое время отключить постоянное соединение Master/Slave, чтобы установить фактическое значение не реферированной Slave-оси при POWER ON на значение Master-оси. После этого снова восстанавливается постоянное соединение.

Постоянное соединение Master/Slave активируется следующей настройкой MD:
MD37262 $MA_MS_COUPLING_ALWAYS_ACTIVE = 1
Она не влияет на языковые команды временного соединения.

Примеры

Пример 1: Установка фактического значения для Slave-оси соединения Master/Slave

При постоянном соединении Master/Slave с помощью PRESETON фактическое значение Slave-оси устанавливается на значение Master-оси.

Программный код	Комментарий
$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=0	; Выключение постоянного соединения Slave-оси
NEWCONF	; Активация изменения машинных данных
STOPRE	
MASLOF(Y1)	; Выключение временного соединения

Программный код	Комментарий
`PRESETON(AX2,$VA_IM(M_AX))`	; Фактическое значение `Slave`-оси = фактическое значение `Master`-оси
`$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=1`	; Включение постоянного соединения `Slave`-оси
`NEWCONF`	; Активация изменения машинных данных

Пример 2: Динамическое конфигурирование соединения Master/Slave

Для того, чтобы соединение после вращения осевого контейнера, могло бы быть замкнуто с другим шпинделем, прежде необходимо разъединить старое соединение, удалить конфигурацию и сконфигурировать новое соединение.

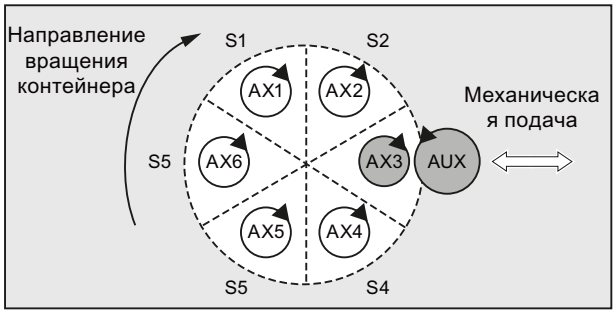


Рисунок 4-15 Перед поворотом осевого контейнера

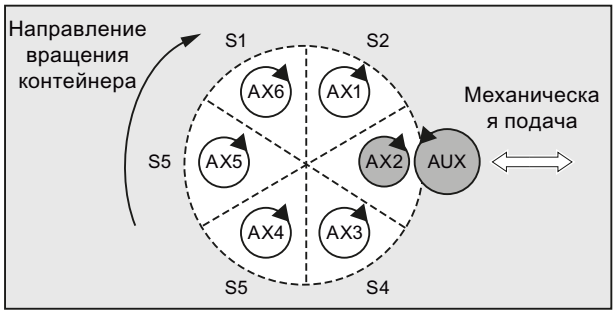


Рисунок 4-16 После поворота осевого контейнера на один слот

Программный код	Комментарий
`MASLDEF(AUX,S3)`	`; AUX: Slave, S3: Master = AX3`
`MASLON(AUX)`	; Соединение вкл
`M3=3 S3=4000`	; Поворот `Master`
`MASLDEL(AUX)`	; Разрыв и удаление соединения
`AXCTSWE(CT1)`	; Разрешение поворота осевого контейнера
`MASLDEF(AUX,S3)`	`; AUX: Slave, S3: Master = AX2`

Дополнительная информация

- Описание функций – Технологии
- Описание функций – Основные функции

4.17 Синхронные действия

4.17.1 Определение синхронного действия

Синхронное действие определяется в кадре программы обработки детали. В этом кадре не должно быть запрограммировано других команд, не являющихся составной частью синхронного действия.

Синхронное действие состоит из следующих компонентов:

Действительность, идент. номер. (опционально)	**Условная часть (опционально)**			**Операционная часть при выполненном условии**			**Операционная часть при невыполненном условии (опционально)**		
	Частота	**G-команда (опционально)**	**Условие**	**Кодовое слово**	**G-команда (опционально)**	**Операции**	**Кодовое слово**	**G-команда (опционально)**	**Операции**
--- [1] ID=<Nr> IDS=<Nr>	--- [1] WHENEVER FROM WHEN EVERY	G...	Логическое выражение	DO	G...	Операция 1 ... Операция n	ELSE	G...	Операция 1 ... Операция n

[1] не запрограммировано

Синтаксис

```
DO <операция 1> ... <операция n>
<частота> [<G-функция>] <условие> DO <операция 1> ... <операция n>
ID=<Nr> <частота> [<G-функция>] <условие> DO <операция 1> ... <операция n>
IDS=<Nr> <частота> [<G-функция>] <условие> DO <операция 1> ... <операция n>
IDS=<№> <частота> [<G-функция>] <условие> DO <операция 1...n> ELSE
<операция 1...n>
```

Дополнительная информация

Описание функций "Синхронные действия"

4.18 Качание

4.18.1 Асинхронное качание (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB)

Качающаяся ось выполняет поступательно-возвратные движения между двумя точками возврата 1 и 2 с заданной подачей до отключения маятникового движения.

При маятниковом движении возможна любая интерполяция других осей. Через движение по траектории или с помощью позиционирующей оси можно достичь непрерывной подачи. Но при этом **связь** между маятниковым движением и движением подачи отсутствует.

Свойства асинхронного качания

- Асинхронное качание действует спец. для оси за границы кадра.
- Через программу обработки детали обеспечивается синхронное с кадром включение маятникового движения.
- Общая интерполяция нескольких осей и наложение участков качания невозможны.

Программирование

Через следующие команды возможно соответствующее обработке программы ЧПУ включение и управление асинхронным качанием из программы обработки детали.

Запрограммированные значения синхронно с кадром на главном ходе вносятся в соответствующие установочные данные и действуют до следующего изменения.

Синтаксис

```
OSP1[<ось>]=<значение> OSP2[<ось>]=<значение>
OST1[<ось>]=<значение> OST2[<ось>]=<значение>
FA[<ось>]=<значение>
OSCTRL[<ось>]=(<опция установки>,<опция сброса>)
OSNSC[<ось>]=<значение>
OSE[<ось>]=<значение>
OSB[<ось>]=<значение>
OS[<ось>]=1
OS[<ось>]=0
```

Объяснение

<ось>:	Имя качающейся оси		
OS:	Включение / выключение качания		
	Значение:	1	**Вкл**ючить качание
		0	**Выкл**ючить качание
OSP1:	Определить позицию точки возврата 1		

OSP2:	Определить позицию точки возврата 2 **Примечание:** Если активно инкрементальное перемещение, то позиция вычисляется инкрементально к последней запрограммированной в программе ЧПУ соответствующей позиции возврата.		
OST1:	Определить время задержки в точке возврата 1 в [сек]		
OST2:	Определить время задержки в точке возврата 2 в [сек]		
	<значение>:	-2	Интерполяция продолжается без ожидания точного останова
		-1	Ожидание точного останова грубого
		0	Ожидание точного останова точного
		>0	Ожидание точного останова точного с последующим соблюдением указанного времени задержки **Примечание:** Единица для времени задержки идентична запрограммированному через G4 времени задержки.
FA:	Определить скорость подачи В качестве скорости подачи действует определенная скорость подачи позиционирующей оси. Если скорость подачи не определена, то действует зафиксированное в машинных данных значение.		

OSCTRL:	Указать опции установки и сброса Опциональные значения 0 - 3 кодируют поведение в точках возврата при отключении. Можно выбрать один из вариантов 0 - 3. Прочие установки при необходимости могут комбинироваться с выбранным вариантом. Несколько опций соединяются символом плюс (+).		
	<значение>:	0	При отключении маятникового движения остановиться в следующей точке возврата (предустановка) **Примечание:** Возможно только через сброс значений 1 и 2.
		1	При отключении маятникового движения остановка в точке возврата 1
		2	При отключении маятникового движения остановка в точке возврата 2
		3	При отключении маятникового движения не выполнять подвод к точке возврата, если ходы выхаживания не запрограммированы
		4	После выхаживания подвод к конечной позиции
		8	Если маятниковое движение отменяется через стирание остатка пути, то после должны быть выполнены ходы выхаживания и при необходимости переход к конечной позиции.
		16	Если маятниковое движение отменяется через стирание остатка пути, то, как при отключении, должен быть выполнен переход к соответствующей позиции возврата.
		32	Измененная подача активна только со следующей точки возврата
		64	FA равна 0, FA = 0: наложение хода активно FA отлична от 0, FA <> 0: наложение скорости активно
		128	Для круговой оси DC (кратчайший путь)
		256	Ход выхаживания выполняется как двойной ход (стандарт) 1=Ход выхаживания выполняется как одиночный ход.
		512	Сначала подвод в стартовую позицию
OSNSC:	Определить число ходов выхаживания		
OSE:	Определить конечную позицию (в WCS), к которой должен быть выполнен подвод после отключения качания **Примечание:** При программировании "OSE" для "OSCTRL" не явно начинает действовать опция 4.		
OSB:	Определить стартовую позицию (в WCS), в которую должен быть выполнен подвод перед включением качания Подвод в стартовую позицию выполняется до точки возврата 1. Если стартовая позиция совпадает с позицией возврата 1, то следующим шагом выполняется подвод в позицию возврата 2. При достижении стартовой позиции время задержки не действует, даже если стартовая позиция совпадает с позицией возврата 1, вместо этого выполняется ожидание точного останова точного. Установленное условие точного останова соблюдается. **Примечание:** Для подвода в стартовую позицию в установочных данных SD43770 $SA_OSCILL_CTRL_MASK должен быть установлен бит 9.		

Примеры

Пример 1: Качающаяся ось должна качаться между двумя точками возврата

Качающаяся ось Z должна качаться между позицией 10 и 100. Подвод к точке возврата 1 с точным остановом точным, к точке возврата 2 с точным остановом грубым. Подача для качающейся оси должна составлять 250. В конце обработки должны быть выполнены 3 хода выхаживания и осуществлен подход к конечной позиции 200 с качающейся осью. Подача для оси подачи должна составлять 1, конец подачи в направлении X должен быть достигнут на позиции 15.

Программный код	Комментарий
`WAITP(X,Y,Z)`	; исходное положение.
`G0 X100 Y100 Z100`	; переключение в режим позиционирующей оси.
`WAITP(X,Z)`	
`OSP1[Z]=10 OSP2[Z]=100`	; точка возврата 1, точка возврата 2.
`OSE[Z]=200`	; конечная позиция.
`OST1[Z]=0 OST2[Z]=-1`	; время ожидания в `U1:` точный останов точный
	; время ожидания в `U2:` точный останов грубый
`FA[Z]=250 FA[X]=1`	; подача качающейся оси, подача оси подачи.
`OSCTRL[Z]=(4,0)`	; установочные опции.
`OSNSC[Z]=3`	; 3 хода выхаживания.
`OS[Z]=1`	; запустить качание.
`WHEN $A_IN[3]==TRUE DO DELDTG(X)`	; стирание остатка пути.
`POS[X]=15`	; исходное положение оси X
`POS[X]=50`	; конечное положение оси `X`.
`OS[Z]=0`	; остановить качание.
`M30`	

Примечание

Последовательность команд от "OSP1[Z]=..." до "OSNCS[Z]=..." может быть запрограммирована и в одном кадре.

Пример 2: Качание с изменением позиции возврата Online

Необходимые для асинхронного качания установочные данные могут быть установлены в программе обработки детали.

Если запись в установочные данные осуществляется в программе обработки детали напрямую, то изменение активируется уже на момент предварительной обработки. Синхронное поведение может быть достигнуто через остановку предварительной обработки (`STOPRE`).

Программный код	Комментарий
`$SA_OSCILL_REVERSE_POS1[ Z ] = -10`	
`$SA_OSCILL_REVERSE_POS2[ Z ] = 10`	
`G0 X0 Z0`	
`WAITP(Z)`	

Программный код	Комментарий
`ID=1 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS1[Z] DO $AA_OVR[X]=0`	; Если фактическое значение качающейся оси превысило точку возврата, то ось подачи останавливается.
`ID=2 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[X]=0`	
`OS[Z]=1 FA[X]=1000 POS[X]=40`	; включить качание.
`OS[Z]=0`	; отключить качание.
`M30`	

Дополнительная информация

Качающаяся ось

Для качающейся оси действует:

- Любая ось может использоваться как качающаяся ось.
- Одновременно может быть активно несколько качающихся осей (макс.: число позиционирующих осей).
- Для качающейся оси всегда активна линейная интерполяция `G1` , независимо от актуальной действующей в программе команды G.

Качающаяся ось может:

- Быть входной осью для динамической трансформации
- Быть ведущей осью для осей Gantry и буксируемых осей
- перемещаться:
  - без ограничения рывка "BRISK"
    или
  - с ограничением рывка "SOFT"
    или
  - с ломаной характеристикой ускорения (как позиционирующие оси)

Точки возврата качания

При определении позиций качания учитывать актуальные смещения:

- Абсолютное указание
  "OSP1[Z]=<значение>"
  Позиция точки возврата = сумма смещений + запрограммированное значение
- Относительное указание
  "OSP1[Z]=IC(<значение>)"
  Позиция точки возврата = точка возврата 1 + запрограммированное значение

Пример:

Программный код
`N10 OSP1[Z] = 100 OSP2[Z] = 110`
`...`
`N40 OSP1[Z] = IC(3)`

WAITP

Если необходимо качание с гео-осью, то она должна быть разрешена с "WAITP" для качания.

После завершенного качания с помощью "WAITP" качающаяся ось снова вносится как позиционирующая ось и может снова использоваться обычно.

Качание с синхронными действиями движения и временем задержки

По истечении установленного времени задержки при качании выполняется внутренняя смена кадра (видна по новым остаточным путям осей). При смене кадра проверяется функция отключения. При этом функция отключения определяется согласно выполненной установке СЧПУ для процесса движения (`OSCTRL`). *Этой характеристикой в функции времени можно управлять через процентовку подачи.*

При определенных обстоятельствах, после выполняется еще один ход качания, прежде чем будут запущены ходы выхаживания или выполнен подвод к конечной позиции. *При этом создается впечатление, что изменяется характеристика отключения. Но это не так.*

4.18.2 Управляемое через синхронные действия качание (OSCILL)

При этом типе качания движение подачи разрешено только на точках возврата или внутри определенных областей возврата.

В зависимости от требований, при подаче маятниковое движение может быть

- продолжено или
- остановлено до полного выполнения подачи.

Синтаксис

1. **Определение параметров для качания**
2. **Определение синхронных действий движения**
3. **Согласование осей, определение подачи**

Объяснение

`OSP1[`<качающаяся ось>`]=`	Позиция точки возврата 1
`OSP2[`<качающаяся ось>`]=`	Позиция точки возврата 2
`OST1[`<качающаяся ось>`]=`	Время задержки в точке возврата 1 в секундах
`OST2[`<качающаяся ось>`]=`	Время задержки в точке возврата 2 в секундах
`FA[`<качающаяся ось>`]=`	Подача качающейся оси
`OSCTRL[`<качающаяся ось>`]=`	Опции установки или сброса
`OSNSC[`<качающаяся ось>`]=`	Число ходов выхаживания
`OSE[`<качающаяся ось>`]=`	Конечная позиция
`WAITP(`<качающаяся ось>`)`	Разрешить ось для качания

Согласование осей, подача
OSCILL[<качающаяся ось>]=(<ось подачи 1>,<ось подачи 2>,<ось подачи 3>)
POSP[<ось подачи>]=(<конечная позиция>,<длина фрагмента>,<режим>)

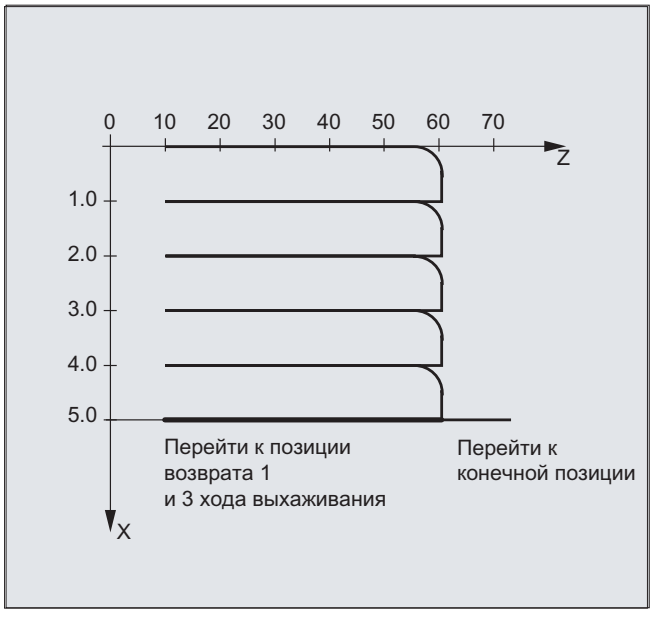
OSCILL:	Согласовать ось(и) подачи с качающейся осью
POSP:	Определить общую и частичные подачи (см. главу "Управление файлами и программами")
Конечная позиция:	Конечная позиция для оси подачи после прохождения всех частичных подач.
Длина фрагмента:	Величина частичной подачи в точке возврата/области возврата
Режим:	Разделение общей подачи на частичные подачи = два заключительных шага одинакового размера (предустановка); = все частичные подачи имеют одинаковый размер

Синхронные действия движения

WHEN… … DO	если..., тогда...
WHENEVER … DO	всегда когда..., тогда...

Пример

В точке возврата 1 подача не должна выполняться. Для точки возврата 2 подача должна быть выполнена уже на расстоянии ii2 перед точкой возврата 2 и качающаяся ось в точке возврата не должна ожидать завершения частичной подачи. Ось Z это качающаяся ось, а ось X – это ось подачи.



1. Параметры качания

Программный код	Комментарий
`DEF INT ii2`	; определить переменную для области возврата 2
`OSP1[Z]=10 OSP2[Z]=60`	; определить точку возврата 1 и 2
`OST1[Z]=0 OST2[Z]=0`	; точка возврата 1: точный останов точный
	Точка возврата 2: точный останов точный
`FA[Z]=150 FA[X]=0.5`	; подача качающейся оси `Z`, подача оси подачи `X`
`OSCTRL[Z]=(2+8+16,1)`	; отключить маятниковое движение в точке возврата 2; после `RWL` выхаживание и подвод к конечной позиции; после `RWL` подвод к соответствующей позиции возврата
`OSNC[Z]=3`	; ходы выхаживания
`OSE[Z]=70`	; конечная позиция = 70
`ii2=2`	; настроить область возврата
`WAITP(Z)`	; разрешить качание для оси `Z`

2. Синхронное действие движения

Программный код	Комментарий
`WHENEVER $AA_IM[Z]<$SA_OSCILL_REVERSE_POS2[Z] DO -> $AA_OVR[X]=0 $AC_MARKER[0]=0`	; Всегда, когда текущая позиция качающейся оси `Z` в `MCS` меньше, чем начало области возврата 2, тогда установить осевую процентовку оси подачи `X` на 0% и маркер с индексом 0 на значение 0.
`WHENEVER $AA_IM[Z]>=$SA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[Z]=0`	; Всегда, когда текущая позиция качающейся оси `Z` в `MCS` больше и равно позиции возврата 2, тогда установить осевую процентовку качающейся оси `Z 0%`.
`WHENEVER $AA_DTEPW[X]==0 DO $AC_MARKER[0]=1`	; Всегда, когда остаточный путь частичной подачи равен 0, тогда установить меркер с индексом 0 на значение 1.
`WHENEVER $AC_MARKER[0]==1 DO $AA_OVR[X]=0 $AA_OVR[Z]=100`	; Всегда, когда меркер с индексом 0 равен 1, тогда установить осевую процентовку оси подачи X на 0%. Тем самым не допускается слишком ранняя подача (качающаяся ось `Z` еще не вышла из области возврата 2, но ось подачи `X` готова для новой подачи). Установить осевую процентовку качающейся оси `Z` с 0 % (операция 2-го синхронного действия) для перемещения снова на 100 %.

-> должно быть запрограммировано в одном кадре

3. Запустить качание

Программный код	Комментарий
`OSCILL[Z]=(X) POSP[X]=(5,1,1)`	; запуск осей
	Качающейся оси `Z` в качестве оси подачи присваивается ось X.
	Ось X должна двигаться до конечной позиции 5 с шагом в 1.
`M30`	; конец программы

Дополнительная информация

1. **Определение параметров качания**
   Перед кадром движения, содержащим согласование оси подачи и качающейся оси, а также определение подачи, необходимо определить параметры для качания (см. "Асинхронное качание").
2. **Определение синхронных действий движения**
   Через синхронные условия осуществляется:
   Подавление подачи, пока качающаяся ось находится внутри области возврата (ii1, ii2) или на точке возврата (U1, U2).
   Остановить маятниковое движение при подаче в точке возврата.
   Снова **запустить маятниковое движение** после завершения частичной подачи.
   Определить **старт следующей частичной подачи**.
3. **Согласовать качающуюся ось и ось подачи**, а также определить **общую и частичную подачу**.

Определение параметров качания

Согласование качающейся оси и оси подачи: OSCILL

`OSCILL[`<качающаяся ось>]=(<ось подачи1>,<ось подачи2>,<ось подачи3>)

С помощью команды "OSCILL" выполняются согласования осей и старт маятникового движения.

Одной качающейся оси может быть присвоено макс. 3 оси подачи.

Примечание

Перед стартом качания необходимо определить синхронные условия для поведения осей.

Определение подач: POSP

`POSP[`<ось подачи>]=(<конечная позиция>,<длина фрагмента>,<режим>)

С помощью команды "POSP" СЧПУ сообщается:

- Общая подача (через конечную позицию)
- Размер соответствующей частичной подачи на точке возврата или в области возврата
- Поведение частичной подачи при достижении конечной позиции (через режим)

Режим = 0	Для обоих последних частичных подач выполняется разделение оставшегося пути до заданной точки на 2 заключительных шага равного размера (предустановка).
Режим = 1	Все частичные подачи имеют одинаковый размер. Они вычисляются из общей подачи.

Определение синхронных действий движения

Выполняемые в дальнейшем синхронные действия движения находят общее применение для качания.

Они находят примеры для решения отдельных задач, служащих элементами для создания прикладных пользовательских маятниковых движений.

Примечание

В отдельных случаях синхронные условия могут программироваться и по другому.

кодовых слов

`WHEN … DO …`	если..., тогда...
`WHENEVER … DO`	всегда когда..., тогда...

Функции

С помощью подробно описанных ниже языковых средств могут быть реализованы следующие функции:

1. Подача в точке возврата.
2. Подача в области возврата.
3. Подача в обоих точках возврата.
4. Остановка маятникового движения в точке возврата.
5. Перезапуск маятникового движения.
6. Не запускать частичную подачу слишком рано.

Для всех представленных здесь в качестве примера синхронных действий имеются допущения:

- точка возврата 1 < точки возврата 2
- Z = качающаяся ось
- X = ось подачи

Примечание

Подробные пояснения см. главу "Синхронные действия движения".

Согласовать качающуюся ось и ось подачи, а также определить общую и частичную подачу

Подача в области возврата

Движение подачи должно начинаться в пределах области возврата до достижения точки возврата.

Эти синхронные действия препятствуют движению подачи до тех пор, пока качающаяся ось находится в области возврата.

С учетом приведенных выше допущений получаются следующие операторы:

Область возврата 1:

```
WHENEVER $AA_IM[Z]>$SA_OSCILL_RESERVE_POS1[Z]+ii1 DO $AA_OVR[X] = 0
```

Всегда, когда актуальная позиция качающейся оси в MCS больше, чем начало области возврата 1, тогда установить осевую процентовку оси подачи на 0%.

Область возврата 2:

```
WHENEVER $AA_IM[Z]<$SA_OSCILL_RESERVE_POS2[Z]+ii2 DO $AA_OVR[X] = 0
```

Всегда, когда актуальная позиция качающейся оси в KS меньше, чем начало области возврата 2, тогда установить осевую процентовку оси подачи на 0%.

Подача в точке возврата

Пока качающаяся ось не достигла точки возврата, движение оси подачи не выполняется.

С учетом приведенных выше допущений получаются следующие операторы:

Область возврата 1:

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Всегда, когда актуальная позиция качающейся оси Z в MCS больше или меньше, чем позиция точки возврата 1, тогда установить осевую процентовку оси подачи X на 0% и осевую процентовку качающейся оси Z на 100%.

Область возврата 2:

Для точки возврата 2:

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Всегда, когда актуальная позиция качающейся оси Z в MCS больше или меньше, чем позиция точки возврата 2, тогда установить осевую процентовку оси подачи X на 0% и осевую процентовку качающейся оси Z на 100%.

Остановка маятникового движения в точке возврата

Качающаяся ось останавливается на точке возврата, одновременно начинается движение подачи. Маятниковое движение продолжается после полного завершения движения подачи.

Одновременно это синхронное действие может использоваться для того, чтобы запустить движение подачи, если оно было остановлено предшествующим синхронным действием, которое еще активно.

С учетом приведенных выше допущений получаются следующие операторы:

Область возврата 1:

```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Всегда, когда актуальная позиция качающейся оси в MCS равна позиции возврата 1, тогда установить осевую процентовку качающейся оси на 0% и осевую процентовку оси подачи на 100%.

Область возврата 2:

```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Всегда, когда актуальная позиция качающейся оси Z в MCS равна позиции возврата 2, тогда установить осевую процентовку качающейся оси X на 0% и осевую процентовку оси подачи на 100%.

Обработка точки возврата в режиме Online

Если на правой стороне сравнения стоит обозначенная с $$ переменная главного хода, то выполняется текущая обработка обеих переменных в такте IPO и их сравнение друг с другом.

Примечание

Подробности см. в главе "Синхронные действия движения".

Перезапуск маятникового движения

Это синхронное действие используется для того, чтобы продолжить движение качающейся оси, если движение частичной подачи завершено.

С учетом приведенных выше допущений получаются следующие операторы:

```
WHENEVER $AA_DTEPW[X]==0 DO $AA_OVR[Z] = 100
```

Всегда, когда остаточный путь для частичной подачи оси подачи X в WCS равен нулю, тогда установить осевую процентовку качающейся оси на 100%.

Следующая частичная подача

После выполненной подачи необходимо не допустить преждевременного запуска следующей частичной подачи.

Для этого используется спец. для канала маркер (`$AC_MARKER[`индекс]), который устанавливается в конце частичной подачи (частичный остаточный путь ≡ 0) и удаляется при выходе из области возврата. После синхронное действие не допускает следующего движения подачи.

С учетом приведенных выше допущений, к примеру, для точки возврата 1 получаются следующие операторы:

1. Установить маркер:

```
WHENEVER $AA_DTEPW[X]==0 DO $AC_MARKER[1]=1
```

Всегда, когда остаточный путь для частичной подачи оси подачи X в WCS равен нулю, тогда установить маркер с индексом 1 на 1.

2. Удалить маркер

```
WHENEVER $AA_IM[Z]<> $SA_OSCILL_RESERVE_POS1[Z] DO $AC_MARKER[1] = 0
```

Всегда, когда актуальная позиция качающейся оси Z в MCS больше или меньше, чем позиция точки возврата 1, тогда установить маркер 1 на 0.

3. Недопущение подачи

```
WHENEVER $AC_MARKER[1]==1 DO $AA_OVR[X]=0
```

Всегда, когда маркер 1 равен, тогда установить осевую процентовку оси подачи X на 0%.

4.19 Штамповка и вырубка

4.19.1 Активация/деактивация

4.19.1.1 Включение или выключение штамповки и вырубки (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC)

Активация/деактивация штамповки или вырубки

С PON и SON активируется функция штамповки или вырубки. SPOF завершает все спец. для штамповки и вырубки функции. Действующие модально команды PON и SON являются взаимоисключающими, т. е. PON деактивирует SON и наоборот.

Штамповка/вырубка с предварительным натяжением

Функции SONS и PONS также включают функции штамповки или вырубки.

В отличие от действующего при SON/PON управления ходом на уровне интерполяции в случае этих функций осуществляется сигнально-техническое управление запуском хода на сервоуровне. Благодаря этому работа может выполняться с увеличенной частотой хода и тем самым с увеличенной производительностью штамповки.

При обработке сигнала на предварительном натяжении все функции, ведущие к изменению позиции осей вырубки или штамповки (например, перемещение маховичком вручную, изменения фреймов через PLC, функции измерения) заблокированы.

Штамповка с задержкой

PDELAYON вызывает вывод хода штамповки с задержкой. Действующая модально команда имеет подготовленную функцию и тем самым как правило стоит перед PON. После PDELAYOF продолжается обычная штамповка.

Примечание

Время задержки устанавливается в установочных данных SD42400 $SC_PUNCH_DWELLTIME.

Зависимое от хода ускорение

С PUNCHACC может быть определена характеристика ускорения, которая, в зависимости от расстояния между отверстиями, определяет различные ускорения.

Второй интерфейс штамповки

Станки, которые попеременно должны использовать второй интерфейс штамповки (второй перфоратор или схожее устройство), могут быть переключены на вторую пару быстрых цифровых входов и выходов СЧПУ (пара I/O). Для обоих интерфейсов штамповки доступна вся функциональность штамповки/вырубки. Переключение между первым и вторым интерфейсом штамповки осуществляется через команды `SPIF1` и `SPIF2`.

Примечание

Условие: Через машинные данные вторая пара I/O должна быть определена для функциональности штамповки (→ См. указания изготовителя станка!).

Синтаксис

```
PON G... X... Y... Z...
SON G... X... Y... Z...
SONS G... X... Y... Z...
PONS G... X... Y... Z...
PDELAYON
PDELAYOF
PUNCHACC(<Sмин>,<Aмин>,<Sмакс>,<Aмакс>)
SPIF1/SPIF2
SPOF
```

Объяснение

`PON`:	Активировать штамповку	
`SON`:	Активировать вырубку	
`PONS`:	Активировать штамповку с предварительным натяжением	
`SONS`:	Активировать вырубку с предварительным натяжением	
`SPOF`:	Деактивировать штамповку/вырубку	
`PDELAYON`:	Активировать штамповку с задержкой	
`PDELAYOF`:	Деактивировать штамповку с задержкой	
`PUNCHACC`:	Активировать зависимое от хода ускорение Параметры:	
	<Sмин>	Мин. расстояние между отверстиями
	<Aмин>	Начальное ускорение <Aмин> может быть больше, чем <Aмакс> .
	<Sмакс>	Макс. расстояние между отверстиями
	<Aмакс>	Конечное ускорение <Aмакс> может быть больше, чем <Aмин> .
`SPIF1`:	Активировать **первый** интерфейс штамповки Управление ходом выполняется через первую пару быстрых I/O.	

SPIF2:	Активировать **второй** интерфейс штамповки Управление ходом выполняется через вторую пару быстрых I/O.
	Примечание: После RESET или запуска СЧПУ всегда активен первый интерфейс штамповки. Если используется только один интерфейс штамповки, то его программирование не требуется.

Примеры

Пример 1: Активировать вырубку

Программный код	Комментарий
...	
N70 X50 SPOF	; Позиционирование без запуска штамповки.
N80 X100 SON	; Активировать вырубку, запуск хода перед движением (X=50) и в конце запрограммированного движения (X=100).
...	

Пример 2: штамповка с задержкой

Программный код	Комментарий
...	
N170 PDELAYON X100 SPOF	; Позиционирование без запуска штамповки, активация запуска штамповки с задержкой.
N180 X800 PON	; Активировать штамповку. После достижения конечной позиции ход штамповки выводится с задержкой.
N190 PDELAYOF X700	; Деактивировать штамповку с задержкой, обычный запуск высечки в конце запрограммированного движения.
...	

Пример 3: Штамповка с двумя интерфейсами штамповки

Программный код	Комментарий
...	
N170 SPIF1 X100 PON	; На конце кадра выполняется запуск хода на первом быстром выходе. Сигнал "Ход активен" контролируется на первом входе.
N180 X800 SPIF2	; Второй запуск хода выполняется на втором быстром выходе. Сигнал "Ход активен" контролируется на втором входе.
N190 SPIF1 X700	; Управление ходом для всех дальнейших ходов выполняется с первым интерфейсом.
...	

Дополнительная информация

Штамповка и вырубка с предварительным натяжением (PONS/SONS)

Штамповка и вырубка с предварительным натяжением невозможна в нескольких каналах одновременно. PONS или SONS может быть активирована только в одном канале соответственно.

Зависимое от хода ускорение (PUNCHACC)

Пример:
`PUNCHACC(2,50,10,100)`

Расстояния между отверстиями меньше 2мм:

Движение выполняется с ускорением в 50% от макс. ускорения.

Расстояния между отверстиями от 2мм до 10мм:

Ускорение увеличивается пропорционально расстоянию до 100%.

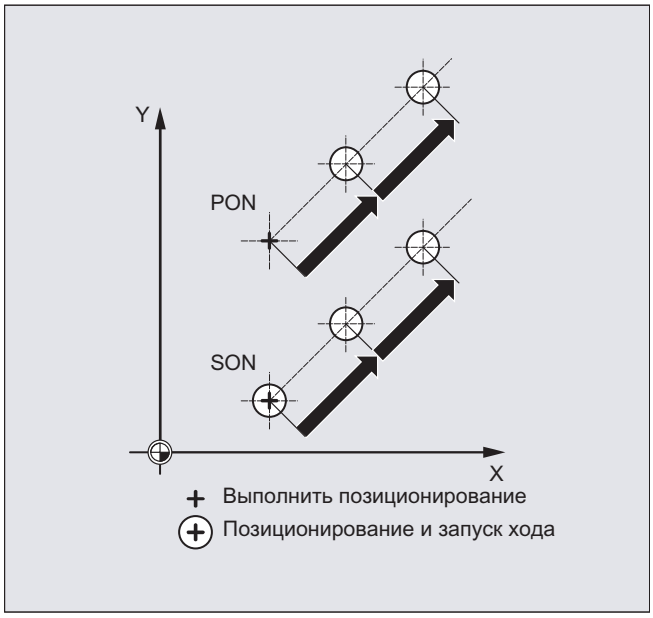
Расстояния между отверстиями больше, чем 10мм:

Движение с ускорением в 100%.

Запуск первого хода

Запуск первого хода после активизации функции осуществляется для вырубки и штамповки различно по времени:

- `PON/PONS`:
  - все ходы – включая ход первого кадра после активизации – осуществляются в конце кадра.
- `SON/SONS`:
  - первый ход после активизации вырубки осуществляется уже в начале кадра.
  - все последующие ходы соответственно запускаются в конце кадра.



Штамповка и вырубка на месте

Запуск хода осуществляется только тогда, когда кадр содержит информацию перемещения для осей штамповки или вырубки (оси активной плоскости).

Для того, чтобы все же запустить ход на том же месте, необходимо запрограммировать одну из осей штамповки/вырубки с путем перемещения 0.

Работа с вращающимися инструментами

Примечание

Для тангенциального подвода вращающихся инструментов к запрограммированной траектории необходимо использовать тангенциальное управление.

Использование команд M

С помощью техники макросов, как и прежде, можно использовать специальные функции M вместо языковых команд (совместимость). При этом действуют следующие соответствия с более старыми системами:

```
M20, M23   ≙  SPOF
M22        ≙  SON
M25        ≙  PON
M26        ≙  PDELAYON
```

Пример для файла макрокоманд:

Программный код	Комментарий
`DEFINE M25 AS PON`	; Штамповка вкл
`DEFINE M125 AS PONS`	; Штамповка с предварительным натяжением вкл
`DEFINE M22 AS SON`	; Вырубка вкл
`DEFINE M122 AS SONS`	; Вырубка с предварительным натяжением вкл
`DEFINE M26 AS PDELAYON`	; Штамповка с задержкой вкл
`DEFINE M20 AS SPOF`	; Штамповка, вырубка выкл
`DEFINE M23 AS SPOF`	; Штамповка, вырубка выкл

Пример программирования:

Программный код	Комментарий
...	
`N100 X100 M20`	; Позиционирование без запуска штамповки.
`N110 X120 M22`	; Активировать вырубку, запуск хода до и после движения.
`N120 X150 Y150 M25`	; Активировать штамповку, запуск хода в конце движения.
...	

4.19.2 Автоматическое разделение пути

Разделение на участки пути

При активной штамповке или вырубке как SPP, так и SPN приводит к разбивке всего запрограммированного для траекторных осей участка перемещения на определенное

число участков пути одинаковой длины (эквидистантное разделение пути). Внутренне каждый участок перемещения соответствует одному кадру.

Число ходов

При штамповке первый ход осуществляется в конечной точке первого участка пути, при вырубке, напротив, в стартовой точке первого участка пути. Поэтому по всему участку перемещения получаются следующие цифры:

Штамповка: число ходов = число участков пути

Вырубка: число ходов = число участков пути + 1

Вспомогательные функции

Вспомогательные функции выполняются в первом из созданных кадров.

Синтаксис

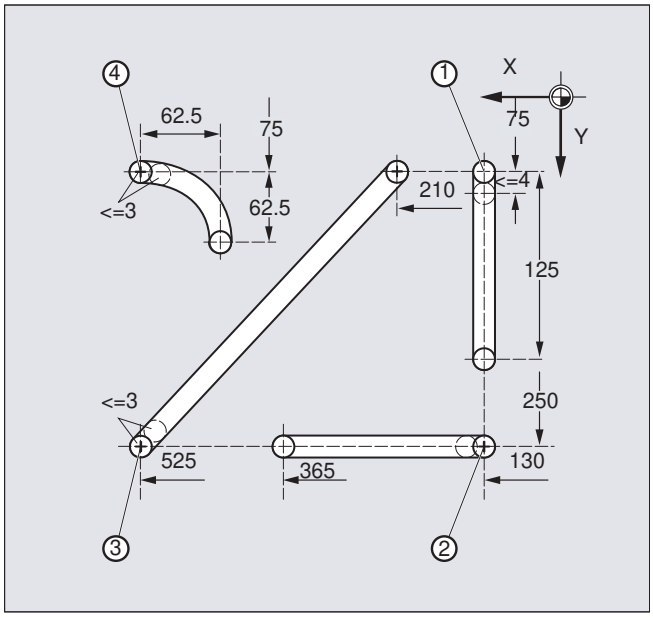
```
SPP=
SPN=
```

Объяснение

SPP:	Размер участка пути (макс. интервал хода); действует модально
SPN:	Число участков пути на кадр; действует покадрово

Пример 1

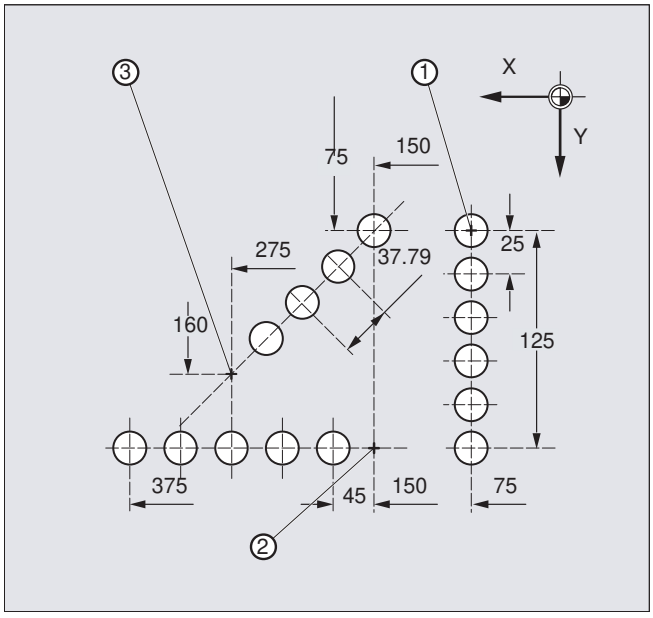
Запрограммированные участки вырубки должны быть автоматически разбиты на участки пути одинакового размера.



Программный код	Комментарий
N100 G90 X130 Y75 F60 SPOF	; Позиционирование на стартовую точку 1
N110 G91 Y125 SPP=4 SON	; Вырубка вкл; макс. длина участка пути для автоматического разделения пути: 4 мм
N120 G90 Y250 SPOF	; Вырубка выкл; позиционирование на стартовую точку 2
N130 X365 SON	; Вырубка вкл; макс. длина участка пути для автоматического разделения пути: 4 мм
N140 X525 SPOF	; Вырубка выкл; позиционирование на стартовую точку 3
N150 X210 Y75 SPP=3 SON	; Вырубка вкл; макс. длина участка пути для автоматического разделения пути: 3 мм
N160 X525 SPOF	; Вырубка выкл; позиционирование на стартовую точку 4
N170 G02 X-62.5 Y62.5 I J62.5 SPP=3 SON	; Вырубка вкл; макс. длина участка пути для автоматического разделения пути: 3 мм
N180 G00 G90 Y300 SPOF	; Вырубка выкл

Пример 2

Для отдельных рядов отверстий должно быть выполнено автоматическое разделение пути. Для разделения указывается макс. длина участка пути (значение SPP) соответственно.



Программный код	Комментарий
N100 G90 X75 Y75 F60 PON	; Позиционирование на стартовую точку 1; штамповка, штамповать отдельное отверстие
N110 G91 Y125 SPP=25	; Макс. длина участков пути для автоматического разделения пути: 25 мм

Программный код	Комментарий
`N120 G90 X150 SPOF`	; Вырубка выкл; позиционирование на стартовую точку 2
`N130 X375 SPP=45 PON`	; Вырубка вкл; макс. длина участка пути для автоматического разделения пути: 45 мм
`N140 X275 Y160 SPOF`	; Вырубка выкл; позиционирование на стартовую точку 3
`N150 X150 Y75 SPP=40 PON`	; Штамповка вкл; вместо запрограммированной длины участков пути в 40 мм используется ;вычисленная длина участков пути в 37,79 мм.
`N160 G00 Y300 SPOF`	; Штамповка выкл; позиционирование

4.19.2.1 Разделение пути для траекторных осей

Длина участка пути SPP

С помощью `SPP` задается макс. интервал хода и тем самым макс. длина участков перемещения, на которые должен быть разделен весь участок перемещения. Отключение команды осуществляется со `SPOF` или `SPP=0`.

Пример:

```
N10 SON X0 Y0
N20 SPP=2 X10
```

Весь участок перемещения в 10 мм делится на 5 участков перемещения по 2 мм каждый (`SPP=2`).

Примечание

Разделение пути с `SPP` всегда выполняется эквидистантно: все участки пути имеют одинаковую длину. Это означает, что запрограммированное значение участков пути (значение `SPP`) действительно только тогда, когда коэффициент из всего участка перемещения и значения SPP является целочисленным. Если это не так, то размер участков пути уменьшается таким образом, чтобы получился целочисленный коэффициент.

X2/Y2 Запрограммированный путь перемещения (кадр вырубки и штамповки)
E1 Запрограммированная длина участка пути
E1' Автоматически округленная длина участка пути

Пример:

```
N10 G1 G91 SON X10 Y10
N20 SPP=3.5 X15 Y15
```

При всем участке перемещения в 15 мм и длине участков пути в 3,5 мм получается не целочисленный коэффициент (4.28). Поэтому выполняется уменьшение значения SPP до следующего возможного целочисленного коэффициента. В этом случае получается длина участков пути в 3 мм.

Число участков пути SPN

С помощью `SPN` определяется число участков пути, которые должны быть созданы из всего пути перемещения. Длина участков пути вычисляется автоматически. Так как `SPN` действует покадрово, то сначала необходимо активировать штамповку или вырубку с `PON` или `SON` .

4.19.2.2 Разделение пути для отдельных осей

Если наряду с траекторными осями и отдельные оси определены как ось штамповки-вырубки, то и они могут быть подвергнуты автоматическому разделению пути.

Поведение отдельной оси при SPP

Запрограммированная длина участка пути (`SPP`) всегда относится к траекторным осям. Поэтому в кадре, в котором наряду с движением отдельной оси и значением SPP траекторная ось не запрограммирована, значение SPP игнорируется.

Если в кадре запрограммирована и отдельная, и траекторная ось, то поведение отдельной оси зависит от настройки соответствующих машинных данных.

1. Стандартная настройка
   Путь отдельной оси равномерно распределяется на созданные через `SPP` промежуточные кадры.

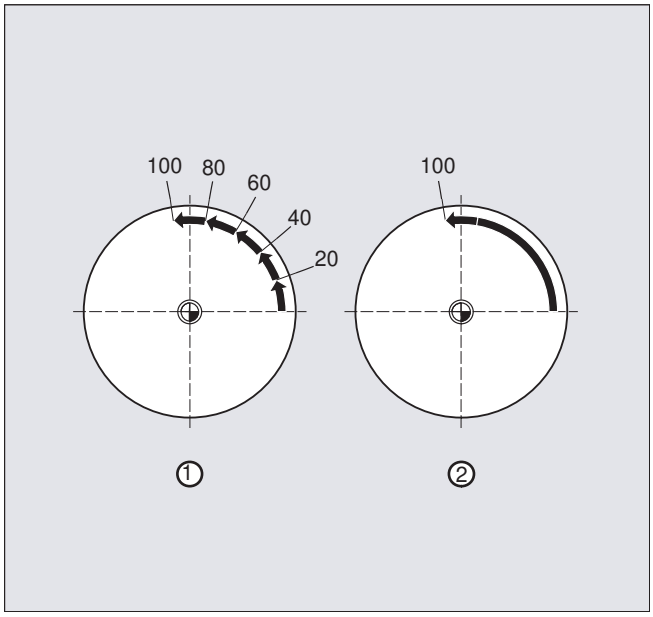
Пример:

```
N10 G1 SON X10 A0

N20 SPP=3 X25 A100
```

Посредством участка хода в 3 мм при всем участке перемещения оси X (траекторная ось) в 15 мм создается 5 кадров.

При этом ось A в каждом кадре поворачивается на 20 °.



1. Отдельная ось без разделения пути
   Отдельная ось проходит весь свой путь в первом из созданных кадров.
2. Различное разделение пути
   Поведение отдельной оси зависит от интерполяции траекторных осей:

- Круговая интерполяция: Разделение пути
- линейная интерполяция: нет разделения пути

Поведение при SPN

Запрограммированное число участков пути действует и в случае, если траекторная ось одновременно не запрограммирована.

Условие: Отдельная ось определена как ось штамповки/вырубки.

4.20 Шлифование

4.20.1 Включение / выключение специфического для шлифования контроля инструмента (TMON, TMOF)

Предварительно определенные процедуры TMON(...) и TMOF(...) включают / выключают специфичный для шлифования контроль инструмента (контроль геометрии и скорости).

Условие

Должны быть установлены специфичные для инструмента параметры инструмента $TC_TPG1 - $TC_TPG9.

Синтаксис

```
TMON(<TNr>)
...
TMOF(<TNr>)
```

Объяснение

TMON(...):	Включение специфического для шлифования контроля инструмента Команда должна быть запрограммирована в канале, в котором должен быть включен специфический для шлифования контроль инструмента.
TMOF(...):	Выключение специфического для шлифования контроля инструмента Команда должна быть запрограммирована в канале, в котором должен быть выключен специфический для шлифования контроль инструмента.
<TNr>:	номер T **Примечание:** требуется только в том случае, если контроль вместо активного, находящегося в эксплуатации инструмента должен быть включен / выключен для неактивного шлифовального круга.
TMOF(0):	Отключить контроль для всех инструментов

4.21 Расширенный останов и отвод (ESR)

Функция "Расширенный останов и отвод", в дальнейшем просто ESR, предлагает возможность гибкого реагирования на ошибки в зависимости от процесса:

- **Расширенный останов**
  Если это позволяет специфическая неполадка, все разрешенные для расширенного останова оси останавливаются упорядоченно.
- **Отвод**
  Находящийся в контакте инструмент отводится от детали с макс. возможной скоростью.
- **Генераторный режим (функция привода SINAMICS "Регулирование Vdc")**
  При падении напряжения промежуточного контура ниже параметрируемого значения, к примеру, при отказе напряжения сети, необходимая для отвода электроэнергия вырабатывается за счет рекуперации тормозной энергии предусмотренного для этого привода (генераторный режим).

Источники запуска

Общие источники (внешн. ЧПУ/глоб. или спец. для ГРР/канала)

- Цифровые входы (к примеру,на модуле ЧПУ) или внутренний, обратно считываемый образ цифровых выходов СЧПУ ($A_IN, $A_OUT)
- Состояние канала ($AC_STAT)
- VDI-сигналы ($A_DBB)
- Сводные сообщения определенного числа аварийных сообщений ($AC_ALARM_STAT)

Осевые источники

- Порог аварийного отвода ведомой оси (синхронный ход электронного соединения, $VC_EG_SYNCDIFF[<ведомая ось>])
- Привод: порог предупреждения промежуточного контура (угроза падения напряжения), $AA_ESR_STAT[<ось>]
- Привод: порог мин. скорости генератора (более нет рекуперируемой энергии вращения), $AA_ESR_STAT[<ось>].

Комбинационная логика статических синхронных действий: связь источников/реакций

Гибкая сочетаемость статических синхронных действий используется для запуска определенных относительно близко по времени реакций на основе источников.

Связь всех релевантных источников с помощью статических синхронных действий находится в руках пользователя. Он может обработать системные переменные-источники как единое целое или селективно с помощью битовых масок и привязать к ним свои желаемые реакции. Статические синхронные действия возможны во всех режимах работы.

Активация

Разрешение функции

Функции - генераторный режим, останов, отвод - разрешаются через установку соответствующего сигнала управления $AA_ESR_ENABLE. Этот управляющий сигнал может изменяться из синхронных действий.

Запуск функции

ESR запускается совместно для всех разрешенных осей через установку системной переменной $AC_ESR_TRIGGER.

Генераторный режим активируется в приводе "автоматически" при распознавании угрожающего падения напряжения промежуточного контура.

Автономный для привода останов и/или отвод активируются при распознавании отказа коммуникации (между ЧПУ и приводом), а также при распознавании пониженного напряжения промежуточного контура в приводе (при условии наличия конфигурирования и разрешения).

Автономный для привода останов и/или отвод также могут быть запущены со стороны ЧПУ через установку соответствующего сигнала управления $AN_ESR_TRIGGER (команда ретрансляции на все приводы).

Дополнительная информация

Описание функций – Оси и шпиндели

4.21.1 Управляемый ЧПУ ESR

4.21.1.1 Управляемый ЧПУ отвод (POLF, POLFA, POLFMASK, POLFMLIN)

Для управляемого ЧПУ отвода должны быть выполнены определенные исходные условия (см. "Управляемый ЧПУ отвод (POLF, POLFA, POLFMASK, POLFMLIN) (Страница 1017)"). Если эти условия выполнены, то для сконфигурированной(ых) в канале оси(ей) отвода через установку системной переменной $AC_ESR_TRIGGER (или $AA_ESR_TRIGGER для отдельных осей) активируется быстрый отвод (LIFTFAST).

Синтаксис

```
POLF[<ось>]=<позиция>
POLFA(<ось>,<тип>,<позиция>)
POLFMASK(<ось_1>,<ось_2>,...)
POLFMLIN(<ось_1>,<ось_2>,...)
```

Для POLFA разрешены следующие краткие формы:
POLFA(<ось>,<тип>) ; краткая форма для отвода отдельной оси
POLFA(ось,0/1/2) ; быстрая деактивация или активация
POLFA(ось,0,$AA_POLFA[ось]) ; вызывает остановку предварительной обработки
POLFA(ось,0) ; не вызывает остановки предварительной обработки

Объяснение

POLF:	адрес для указания заданного конечного положения оси отвода POLF действует модально.			
	<ось>:	имя отводимой гео-оси или оси канала/станка		
	<позиция>:	Позиция отвода		
		Тип:	REAL	
		Для гео-оси действует WCS, в иных случаях MCS. В случае идентичных идентификаторов для гео-оси или оси канала/станка отвод выполняется в WCS.		
POLFA:	предопределенный вызов подпрограммы для указания позиции отвода отдельных осей			
	<ось>:	Идентификатор оси канала		
	<тип>:	режим указания позиции		
		Тип:	INT	
		Значение:	0:	пометить значение позиции как недействительное
			1:	значение позиции является абсолютным
			2:	значение позиции является инкрементальным (дистанция)
		Примечание: Если ось не является отдельной осью или если тип отсутствует/тип=0, то выводится соответствующее аварийное сообщение.		
	<позиция>:	позиция отвода (см. выше)		
		Примечание: Значение позиции принимается и с типом=0. Только после это значение помечено как недействительное и должно быть запрограммировано для отвода заново.		
POLFMASK:	Предопределенный вызов подпрограммы для выбора осей, которые после запуска быстрого отвода должны отводиться **независимо друг от друга**.			
	<ось_1>,...:	Имена осей, которые при быстром отводе должны двигаться на свои определенные с POLF позиции. Все указанные оси должны находиться в одной системе координат.		
	POLFMASK() без указания оси деактивирует быстрый отвод для всех осей, которые были отведены независимо друг от друга.			
POLFMLIN:	Предопределенный вызов подпрограммы для выбора осей, которые после запуска быстрого отвода должны отводиться **в линейной связи**.			
	<ось_1>,...:	см. выше		
	POLFMLIN() без указания оси деактивирует быстрый отвод для всех осей, которые были отведены в линейной связи.			

Примечание

Перед разрешением быстрого отвода на фиксированную позицию через POLFMASKилиPOLFMLIN, для выбранных осей должна быть запрограммирована позиция с POLF.

Примечание

Если оси последовательно разрешаются с POLFMASK, POLFMLIN или POLFMLIN, POLFMASK, то для соответствующей оси всегда действует последняя установка.

Примечание

Запрограммированные с POLF позиции и активация через POLFMASK или POLFMLIN удаляются при старте программы обработки детали. Это означает, что пользователь в каждой программе обработки детали должен заново запрограммировать значения для POLF и выбранных осей в POLFMASK или POLFMLIN.

Примечание

При использовании сокращенного написания POLFA и изменении только типа необходимо обеспечить присваивание позиции отвода или пути отвода имеющего смысл значения! В частности, заново установить позицию отвода и путь отвода после POWER ON.

Пример

Отвод отдельной оси:

Программный код	Комментарий
`MD37500 $MA_ESR_REACTION[AX1]=21`	; Управляемый ЧПУ отвод.
...	
`$AA_ESR_ENABLE[AX1]=1`	
`POLFA(AX1,1,20.0)`	`; AX1` присваивается осевая позиция отвода 20.0 (абсолютно).
`$AA_ESR_TRIGGER[AX1]=1`	; С этого места начинается отвод.

Дополнительная информация

Условия для управляемого ЧПУ отвода

- В канале сконфигурирована ось отвода для управляемого ЧПУ отвода:
  MD37500 $MA_ESR_REACTION = 21
- ESR должен быть разрешен для этой оси:
  $AA_ESR_ENABLE = 1
- Время задержки определено:
  MD21380 $MC_ESR_DELAY_TIME1
  MD21381 $MC_ESR_DELAY_TIME2
- В программе обработки детали спец. для оси позиции отвода запрограммированы с POLF.
- Оси с POLFMASK/POLFMLIN выбраны для управляемого ЧПУ отвода.
- Для движения отвода должны быть установлены и оставаться установленными сигналы разрешения.

Разрешение и запуск управляемого ЧПУ отвода

Если устанавливается системная переменная $AC_ESR_TRIGGER = 1, и если в этом канале сконфигурирована ось отвода (т.е. MD37500 $MA_ESR_REACTION = 21) и для этой оси установлено $AA_ESR_ENABLE = 1, то в этом канале активируется быстрый отвод (LIFTFAST).

Сконфигурированное с `POLF` (или `LFPOS`) движение отвода выбранных с `POLFMASK` или `POLFMLIN` осей заменяет для этих осей определенное в программе обработки детали движение по траектории.

Для отвода предлагается макс. сумма времен MD21380 $MC_ESR_DELAY_TIME1 и MD21381 $MC_ESR_DELAY_TIME2. По истечении этого интервала времени и для оси отвода запускается быстрое торможение с последующим слежением.

Примечание

Расширенное движение отвода (т.е. запущенный через $AC_ESR_TRIGGER LIFTFAST/ LFPOS) **не может быть прервано** и может быть преждевременно завершено только через аварийный останов.

Примечание

Запущенный через $AC_ESR_TRIGGER отвод заблокирован от многократного отвода.

Отвод отдельной оси

При отводе отдельной оси с `POLFA` должна быть запрограммирована позиция отвода отдельной оси и следующие условия должны быть соблюдены:

- $AA_ESR_ENABLE = 1
- <ось> на момент запуска ($A**A**_ESR_TRIGGER = 1) должна быть отдельной осью.
- <тип> должен быть 1 или 2.

Направление отвода при быстром отводе

На момент активации быстрого отвода учитывается действительный фрейм.

Примечание

Фреймы с вращением через `POLF` влияют и на направление отвода.

Переход оси

Оси отвода всегда согласованы с одним каналом ЧПУ и их переход между каналами невозможен. При попытке передачи оси отвода в другой канал сигнализируется аварийное сообщение. Только после деактивации этой оси с $AA_ESR_ENABLE[AX] = 0 она может быть передана в новый канал. После успешного перехода оси снова могут быть нагружены с $AA_ESR_ENABLE[AX] = 1.

Нейтральные оси

Нейтральные оси не могут выполнять управляемый ЧПУ ESR.

4.21.1.2 Управляемый ЧПУ останов

Для сконфигурированной(ых) в канале оси(ей) останова через установку системной переменной $AC_ESR_TRIGGER (или $AA_ESR_TRIGGER для отдельных осей) можно активировать управляемый ЧПУ останов.

Условия

- В канале сконфигурирована ось останова для управляемого ЧПУ останова:
  MD37500 $MA_ESR_REACTION = 22
- ESR должен быть разрешен для этой оси:
  $AA_ESR_ENABLE = 1
- Время задержки определено:
  MD21380 $MC_ESR_DELAY_TIME1 (время задержки ESR-осей)
  MD21381 $MC_ESR_DELAY_TIME2 (ESR-время для интерполяционного торможения)

Процесс

В течение интервала времени в MD 21380 ось продолжает нормальную интерполяцию согласно программированию. По истечении интервала времени в MD 21380 запускается управляемое интерполяционно торможение (останов по рампе). Для управляемого интерполяционно торможения доступен макс. интервал времени в MD21381. По истечении этого интервала времени запускается быстрое торможение с последующим слежением.

Пример

Останов отдельной оси:

Программный код	Комментарий
`MD37500 $MC_ESR_REACTION[AX1]=22`	; Управляемый ЧПУ останов.
`MD21380 $MC_ESR_DELAY_TIME1[AX1]=0.3`	
`MD21381 $MC_ESR_DELAY_TIME2[AX1]=0.06`	
...	
`$AA_ESR_ENABLE[AX1]=1`	
`$AA_ESR_TRIGGER[AX1]=1`	; С этого места начинается останов.

4.21.2 Автономный для привода ESR

4.21.2.1 Конфигурирование автономного останова привода (ESRS)

С помощью функции `ESRS(...)` конфигурируются параметры привода для "остановки" автономной для привода функции ESR.

Синтаксис

```
ESRS(<ось_1>,<время останова_1>[,...,<ось_n>,<время останова_n>])
```

Объяснение

`ESRS(...)`:	Функция для записи параметров привода для функции ESR "Останов". Функция: • должна стоять одна в кадре • вызывает остановку предварительной обработки. • не может использоваться в синхронных действиях.	
<ось_1>, ..., <ось_n>:	Ось, для которой должен быть сконфигурирован автономный для привода останов. В приводе для этой оси записывается параметр привода p0888 (конфигурация): p0888 = 1	
	Тип:	AXIS
	Диапазон значений:	Идентификатор оси канала
<время останова_1>, ..., <время останова_n>:	Период времени, в течение которого привод после возникновения ошибки продолжает постоянное движение с текущим заданным значением скорости. В приводе для этой оси записывается параметр привода p0892 (ступенчатая выдержка времени): p0892 = <время останова>	
	Единица:	c
	Тип:	REAL
	Диапазон значений:	0.00 - 20.00
В одном вызове функции может быть запрограммировано макс. 5 осей; n = 5		

4.21.2.2 Конфигурирование автономного отвода привода (ESRS)

С помощью функции `ESRR(...)` конфигурируются параметры привода для "отвода" автономной функции ESR привода.

Синтаксис

```
ESRR(<ось_1>,<путь отвода_1>,<скорость отвода_1>[,...,<ось_n>,<путь отвода_n>,<скорость отвода_n>])
```

Значение

ESRR(...):	Функция для записи параметров привода для функции ESR "Отвод". Функция: • должна стоять одна в кадре • вызывает остановку предварительной обработки. • не может использоваться в синхронных действиях.	
<ось_1>, ..., <ось_n>:	Ось, для которой должен быть сконфигурирован автономный для привода отвод. В приводе для этой оси записывается параметр привода p0888 (конфигурация): p0888 = 2	
	Тип:	AXIS
	Диапазон значений:	Идентификатор оси канала
<путь отвода_1>, ..., <путь отвода_n>:	Путь отвода для привода пересчитывается в скорость отвода. Значение записывается для указанной оси в параметр привода p0893 (скорость): p0893 = (<путь отвода _n> пересчитанный в скорость отвода)	
	Единица:	мм/мин, дюймов/мин, градусов/мин (в зависимости от единицы оси)
	Тип:	REAL
	Диапазон значений:	MIN - MAX
<скорость отвода_1>, ..., <скорость отвода_n>:	Скорость отвода для привода пересчитывается в промежуток времени. Значение записывается для указанной оси в параметр привода p0892 (ступенчатая выдержка времени) [сек]: p0892 = <путь отвода_n> / <скорость отвода _n>	
	Единица:	мм/мин, дюймов/мин, градусов/мин (в зависимости от единицы оси)
	Тип:	REAL
	Диапазон значений:	0.00 - MAX
В одном вызове функции может быть запрограммировано макс. 5 осей; n = 5		

4.22 Время выполнения программы / счетчик деталей

Для поддержки оператора станка предоставляется информация по времени выполнения программы и числу деталей.

Эту информацию может обрабатывать как системные переменные в программе ЧПУ и/или PLC. Одновременно эта информация доступна для индикации на интерфейсе пользователя.

4.22.1 Время выполнения программы

Функция "Время выполнения программы" предоставляет внутренние таймеры ЧПУ для контроля технологических процессов, которые могут считываться через спец. для ЧПУ и канала системные переменные в программе обработки детали и в синхронных действиях.

Триггер для измерения времени выполнения ($AC_PROG_NET_TIME_TRIGGER) это единственная записываемая системная переменная функции и служит для выборочного измерения сегментов программы. Т.е. через запись в триггер в программе ЧПУ измерение времени может включаться и снова выключаться.

Системная переменная	Объяснение	Активность
Спец. для ЧПУ		
$AN_SETUP_TIME	Время от последнего запуска СЧПУ со стандартными значениями ("холодный пуск") в минутах При каждом запуске СЧПУ со стандартными значениями автоматически сбрасывается на "0".	• Всегда активен
$AN_POWERON_TIME	Время от последнего обычного запуска СЧПУ ("горячий пуск" в минутах При каждом обычном запуске СЧПУ со автоматически сбрасывается на "0".	
Спец. для канала		
$AC_OPERATING_TIME	Общее время выполнения программ ЧПУ в режиме работы АВТО в секундах Значение при каждом запуске СЧПУ со автоматически сбрасывается на "0".	• Активация через MD27860 • Только режим работы АВТО
$AC_CYCLE_TIME	Время выполнения выбранной программы ЧПУ в секундах Значение при старте новой программы ЧПУ автоматически сбрасывается на "0".	
$AC_CUTTING_TIME	Время обработки в секундах Измеряется время работы траекторных осей (минимум одна активна) без активного ускоренного хода во всех программах ЧПУ между NC-Start и концом программы / NC-Reset. Измерение дополнительно прерывается при активном времени ожидания. Значение при каждом запуске СЧПУ со стандартными значениями автоматически сбрасывается на "0".	

Системная переменная	Объяснение		Активность
$AC_ACT_PROG_NET_TIME	Актуальное время выполнения нетто текущей программы ЧПУ в секундах При старте программы ЧПУ автоматически сбрасывается на "0".		• Всегда активен • Только режим работы АВТО
$AC_OLD_PROG_NET_TIME	Время выполнения нетто только что правильно завершенной с M30 программы в секундах		
$AC_OLD_PROG_NET_TIME_COUNT	Изменения на $AC_OLD_PROG_NET_TIME После POWER ON $AC_OLD_PROG_NET_TIME_COUNT стоят на "0". $AC_OLD_PROG_NET_TIME_COUNT всегда увеличивается, когда СЧПУ заново записывает $AC_OLD_PROG_NET_TIME.		
$AC_PROG_NET_TIME_TRIGGER	Триггер для измерения времени выполнения:		• только режим работы АВТО
	0	Нейтральное состояние Триггер не активен.	
	1	Завершение Завершает измерение и копирует значение из $AC_ACT_PROG_NET_TIME в $AC_OLD_PROG_NET_TIME. $AC_ACT_PROG_NET_TIME устанавливается на "0" и затем продолжает выполняться.	
	2	Старт Запускает измерение и при этом устанавливает $AC_ACT_PROG_NET_TIME на "0". $AC_OLD_PROG_NET_TIME не изменяется.	
	3	Стоп Останавливает измерение. Не изменяет $AC_OLD_PROG_NET_TIME и поддерживает $AC_ACT_PROG_NET_TIME постоянной до продолжения.	
	4	Продолжение Продолжение измерения, т. е. остановленное ранее измерение снова возобновляется. $AC_ACT_PROG_NET_TIME продолжает выполняться. $AC_OLD_PROG_NET_TIME не изменяется.	
Через POWER ON все системные переменные сбрасываются на "0"!			

Примечание

Изготовитель станка

Включение активируемых таймеров осуществляется через машинные данные MD27860 $MC_PROCESSTIMER_MODE.

Поведение активных измерений времени при определенных функциях (например, GOTOS, процентовка = 0%, активна подача пробного хода, программный тест, ASUP, PROG_EVENT, ...) конфигурируется через машинные данные MD27850 $MC_PROG_NET_TIMER_MODE и MD27860 $MC_PROCESSTIMER_MODE.

Дополнительная информация: Описание функций – Основные функции

Примечание

Оставшееся время для детали

Если последовательно изготовляются идентичные детали, по следующим значениям времени можно определить оставшееся время для детали.

- Время обработки для последней изготовленной детали (см. $AC_OLD_PROG_NET_TIME)
- Актуальное время обработки (см. $AC_ACT_PROG_NET_TIME)

Оставшееся время дополнительно к текущему времени обработки отображается на интерфейсе.

Примечание

Использование STOPRE

Системные переменные $AC_OLD_PROG_NET_TIME и $AC_OLD_PROG_NET_TIME_COUNT не создают не явной остановки предварительной обработки. При использовании в программе обработки детали это некритично, если значение системных переменных происходит из предшествующего хода программы. Но если триггер для измерения времени выполнения ($AC_PROG_NET_TIME_TRIGGER) записывается высокочастотно и из-за этого $AC_OLD_PROG_NET_TIME изменяется очень часто, то в программе обработки детали следует использовать явную STOPRE.

Граничные условия

- **Поиск кадра**
  При поиске кадра время выполнения программы не определяется.
- **REPOS**
  Продолжительность процесса REPOS зачисляется в текущее время обработки ($AC_ACT_PROG_NET_TIME).

Примеры

Пример 1: Измерение длительности "mySubProgrammA"

Программный код
...
N50 DO $AC_PROG_NET_TIME_TRIGGER=2
N60 FOR ii= 0 TO 300
N70 mySubProgrammA
N80 DO $AC_PROG_NET_TIME_TRIGGER=1
N95 ENDFOR
N97 mySubProgrammB
N98 M30

После обработки программой строки N80, в $AC_OLD_PROG_NET_TIME стоит время выполнения нетто "mySubProgrammA".

Значение $AC_OLD_PROG_NET_TIME:

- сохраняется после M30.
- обновляется после каждого выполнения в цикле.

Пример 2: Измерение длительности "mySubProgrammA" и "mySubProgrammC"

Программный код
...
N10 DO $AC_PROG_NET_TIME_TRIGGER=2
N20 mySubProgrammA
N30 DO $AC_PROG_NET_TIME_TRIGGER=3
N40 mySubProgrammB
N50 DO $AC_PROG_NET_TIME_TRIGGER=4
N60 mySubProgrammC
N70 DO $AC_PROG_NET_TIME_TRIGGER=1
N80 mySubProgrammD
N90 M30

4.22.2 Счетчики деталей

Функция "Счетчики деталей" предлагает различные счетчики, которые прежде всего могут использоваться для подсчета деталей в СЧПУ.

Счетчики существуют как специфические для канала системные переменные с доступом по записи и чтению в диапазоне значений от 0 до 999 999 999.

Системная переменная	Объяснение
$AC_REQUIRED_PARTS	Число изготовляемых деталей (заданное число деталей) В этом счетчике может быть определено количество деталей, при достижении которого фактическое число деталей ($AC_ACTUAL_PARTS) сбрасывается на "0".
$AC_TOTAL_PARTS	Общее число изготовленных деталей (общее фактическое количество) Этот счетчик указывает число всех изготовленных с момента запуска деталей. Только при запуске СЧПУ со стандартными значениями, значение автоматически сбрасывается на "0".
$AC_ACTUAL_PARTS	Число изготовленных деталей (фактическое число деталей) В этом счетчике регистрируется число всех изготовленных с момента запуска деталей. При достижении заданного числа деталей ($AC_REQUIRED_PARTS) счетчик автоматически сбрасывается на "0" (при условии $AC_REQUIRED_PARTS > 0).
$AC_SPECIAL_PARTS	Число подсчитанных пользователем деталей Этот счетчик позволяет пользователю подсчитывать детали по собственному определению. Может быть определен вывод ошибки при достижении заданного числа деталей ($AC_REQUIRED_PARTS). Обнуление счетчика должно выполняться самим пользователем.

Примечание

Все счетчики деталей при запуске СЧПУ со стандартными значениями устанавливаются на "0" и могут считываться/записываться независимо от их активации.

Примечание

Через спец. для канала машинные данные можно управлять активацией счетчиков, моментом обнуления и алгоритмом подсчета.

Примечание

Подсчет деталей с определенной пользователем командой M

Через машинные данные можно установить, что счетные импульсы для различных счетчиков деталей вместо запуска через конец программы `M2`/`M30`, могут быть запущены через определенные пользователем команды M.

4.23 Симуляция программы

4.23.1 Функция

При симуляции программы осуществляется полное вычисление текущей программы обработки детали и представление результата в графическом виде в интерфейсе пользователя. Так результат программирования контролируется без перемещения осей станка. Неправильно запрограммированные этапы обработки своевременно распознаются; неправильная обработка детали не допускается.

ЧПУ моделирования

Моделирование использует собственный экземпляр ЧПУ (ЧПУ моделирования). Поэтому перед стартом симуляции выполняется калибровка реального ЧПУ по ЧПУ моделирования. При такой калибровке выполняется, кроме прочего, чтение из ЧПУ всех активных машинных данных и их загрузка в ЧПУ моделирования. Среди активных машинных данных находятся машинные данные ЧПУ и циклов.

4.23.2 Симуляция программы в сочетании с компилируемыми циклами

Некоторые компилируемые циклы (CC) поддерживаются при симуляции программы. Машинные данные поддерживаемых компилируемых циклов калибруются один раз после запуска СЧПУ. Калибровка посредством "Старт симуляции" не выполняется!

Примечание

В программах обработки детали спец. для компилируемых циклов языковые команды и машинные данные неподдерживаемых компилируемых циклов **не** используются (см. также абзац "Команды компилируемых циклов в программе обработки детали").

В некоторых случаях специальные движения поддерживаемых компилируемых циклов (трансформации OEM) отображаются неправильно.

Команды компилируемых циклов в программе обработки детали

Языковые команды в программе обработки детали из не поддерживаемых компилируемых циклов ((`OMA1` ... `OMA5`, `OEMIPO1/2`, `G810` ... `G829`, собственные процессы и функции) приводят без индивидуальной обработки с сообщениям об ошибках и отмене симуляции.

Решение:

В индивидуальном порядке обработать отсутствующие спец. для компилируемых циклов языковые элементы в программе обработки детали (запрос $P_SIM).

Пример:

Программный код	Комментарий
`N1 G01 X200 F500`	
`IF (1==$P_SIM)`	
`N5 X300`	; при симуляции компилируемых циклов не активен.
`ELSE`	
`N5 X300 OMA1=10`	
`ENDIF`	

4.24 Другие функции

4.24.1 Активация машинных данных (NEWCONF)

Команда NEWCONF активирует все машинные данные. Функция может быть активирована и на интерфейсе управления HMI посредством нажатия программной клавиши "Активировать MD".

При выполнении функции "NEWCONF" происходит не явная остановка предварительной обработки, т. е. движение по траектории прерывается.

Синтаксис

```
NEWCONF
```

Объяснение

NEWCONF:	Команда активации всех машинных данных с условием активации "NEW_CONFIG"

Выполнение NEWCONF из программы обработки детали с выходом за границы канала

Если осевые машинные данные изменяются из программы обработки детали и после активируются с NEWCON, то NEWCONF активирует только машинные данные, вызывающие изменения для канала программы обработки детали.

Примечание

Для надежной активации всех изменений, команда NEWCONF должен быть выполнен в каждом канале, где сейчас вычисляются и затронутые измененными машинными данными оси или функции.

При NEWCONF осевые машинные данные не активируются.

Для контролируемых с PLC осей необходимо выполнить осевой RESET.

Пример

Фрезерная обработка: Обработка позиции сверления с помощью различных технологий

Программный код	Комментарий
`N10 $MA_CONTOUR_TOL[AX]=1.0`	; Изменить машинные данные.
`N20 NEWCONF`	; Активировать машинные данные.
...	

4.24.2 Проверка имеющейся языковой среды ЧПУ (STRINGIS)

С помощью функции "STRINGIS(...)" можно проверить, доступна ли указанная строка как элемент языка программирования ЧПУ в текущей языковой среде.

Определение

`INT STRINGIS(STRING <имя>)`

Синтаксис

`STRINGIS(<имя>)`

Объяснение

`STRINGIS`:	Функция с возвращаемым значением
<имя>:	Имя проверяемого элемента языка программирования ЧПУ
Возвращаемое значение:	Формат возвращаемого значения yxx (дес.).

Элементы языка программирования ЧПУ

Могут быть проверены следующие элементы языка программирования ЧПУ:

- G-команды всех существующих групп G-функций, например, "G0", "INVCW", "POLY", "ROT", "KONT", "SOFT", "CUT2D", "CDON", "RMBBL", "SPATH"
- Адреса DIN или ЧПУ, например, "ADIS", "RNDM", "SPN", "SR", "MEAS"
- Функции, например, "TANG(...)" или "GETMDACT"
- Процедуры, например, "SBLOF".
- Кодовые слова, например, "ACN", "DEFINE" или "SETMS"
- Системные данные например, машинные данные $M... , установочные данные $S... или опциональные данные $O...
- Системные переменные $A... , $V... , $P...
- R-параметры R...
- Имена активированных циклов
- Переменные GUD и LUD

- Имена макрокоманд
- Имена меток

Возвращаемое значение

Возвращаемое значение релевантно только в первых 3 десятичных разрядах. Формат возвращаемого значения yxx, где y = базовая информация и xx = подробная информация.

Возвращаемое значение	**Объяснение**	
000	Строка 'name' неизвестна в данной системе [1)]	
100	Строка 'name' это элемент языка программирования ЧПУ, но сейчас не может быть запрограммирована (опция/функция не активна)	
2xx	Строка 'name' это программируемый элемент языка программирования ЧПУ (опция/функция активна). Подробная информация xx содержит дополнительную информацию по типу элемента:	
	xx	**Значение**
	01	Адрес DIN или адрес ЧПУ [2)]
	02	G-команда (например, G04, INVCW)
	03	Функция с возвращаемым значением
	04	Функция без возвращаемого значения
	05	Кодовое слово (например, DEFINE)
	06	Машинные ($M...), установочные ($S...) или опциональные данные ($O...)
	07	Системные параметры, например, системная переменная ($...) или R-параметр (R...)
	08	Цикл (цикл должен быть загружен в ЧПУ, программы циклов должны быть активны [3)])
	09	GUD–переменная (GUD-переменная должна быть определена в файле определений GUD и активирована)
	10	Имя макрокоманды (макрокоманда должна быть определена в файлах определения макрокоманд и активирована) [4)]
	11	LUD–переменная текущей программы обработки детали
	12	ISO G-команда (языковой режим ISO должен быть активен)
400	Строка 'name' это адрес ЧПУ, который не был определен как xx == 01 или xx == 10 и не является G или R [2)]	
y00	Спец. согласование невозможно	

1) В зависимости от СЧПУ при определенных обстоятельствах известно только подмножество языковых команд ЧПУ Siemens, например, SINUMERIK 802D sl. На этих СЧПУ для строк, которые в принципе являются языковыми командами ЧПУ Siemens, возвращается значение 0. Это поведение можно изменить через MD10711 $MN_NC_LANGUAGE_CONFIGURATION. В этом случае при MD10711 = 1 для языковых команд ЧПУ Siemens всегда возвращается значение 100.

2) Адресами ЧПУ являются следующие буквы: A, B, C, E, I, J, K, Q, U, V, W, X, Y, Z. Эти адреса ЧПУ могут быть запрограммированы и с расширением адреса. Расширение адреса может быть указано при проверке со STRINGIS. Пример: 201 == STRINGIS("A1").
Буквы: D, F, H, L, M, N, O, P, S, T это адреса ЧПУ или вспомогательные функции, которые определяются пользователем. Для них всегда возвращается значение 400. Пример: 400 == STRINGIS("D"). Эти адреса ЧПУ при проверке со STRINGIS не могут указываться с расширением адреса.
Пример: 000 == STRINGIS("M02"), но 400 == STRINGIS("M").

3) Имена параметров циклов невозможно проверить посредством STRINGIS.
4) Определенный как макрос адрес, например, G, H, M, L, идентифицируется как макрос.

Примеры

В следующих примерах допускается, что указанный в качестве строки языковой элемент ЧПУ, если не указано иначе, в принципе может быть запрограммирован в СЧПУ.

1. Строка "T" определена как вспомогательная функция:
   ```
   400 == STRINGIS("T")
   000 == STRINGIS ("T3")
   ```
2. Строка "X" определена как ось:
   ```
   201 == STRINGIS("X")
   201 == STRINGIS("X1")
   ```
3. Строка "A2" определена как адрес ЧПУ с расширением:
   ```
   201 == STRINGIS("A")
   201 == STRINGIS("A2")
   ```
4. Строка "INVCW" определена как упомянутая G-команда:
   ```
   202 == STRINGIS("INVCW")
   ```
5. Строка "$MC_GCODE_RESET_VALUES" определена как машинные данные:
   ```
   206 == STRINGIS("$MC_GCODE_RESET_VALUES")
   ```
6. Строка "GETMDACT" это языковая функция ЧПУ:
   ```
   203 == STRINGIS("GETMDACT ")
   ```
7. Строка "DEFINE" это кодовое слово:
   ```
   205 == STRINGIS("DEFINE")
   ```
8. Строка "$TC_DP3" это системный параметр (компонент длин инструмента):
   ```
   207 == STRINGIS("$TC_DP3")
   ```
9. Строка "$TC_TP4" это системный параметр (размер инструмента):
   ```
   207 == STRINGIS("$TC_TP4")
   ```
10. Строка "$TC_MPP4" это системный параметр (состояние места в магазине):
    - Управление магазином инструментов активно: `207 == STRINGIS("$TC_MPP4") ;`
    - Управление магазином инструментов не активно: `000 == STRINGIS("$TC_MPP4")`

    См. также ниже абзац: управление магазином инструментов.
11. Строка "MACHINERY_NAME" определена как GUD-переменная:
    ```
    209 == STRINGIS("MACHINERY_NAME")
    ```
12. Строка "LONGMACRO" определена как макрос:
    ```
    210 == STRINGIS("LONGMACRO")
    ```
13. Строка "MYVAR" определена как LUD-переменная:
    ```
    211 == STRINGIS("MYVAR")
    ```
14. Строка "XYZ" — неизвестная в ЧПУ команда, GUD-переменная, имя макрокоманды или цикла:
    ```
    000 == STRINGIS("XYZ")
    ```

Управление магазином инструментов

Если функция управления магазином инструментов не активна, то STRINGIS возвращает для системных параметров управления магазином инструментов, независимо от машинных данных

- MD10711 $MN_NC_LANGUAGE_CONFIGURATION

всегда значение 000.

Режим ISO

Если функция "Режим ISO" активна:

- MD18800 $MN_MM_EXTERN_LANGUAGE (активация внешних языков ЧПУ)
- MD10880 $MN_ MM_EXTERN_CNC_SYSTEM (адаптируемая система управления)

STRINGIS сначала проверяет указанную строку как G-команду SINUMERIK. Если строка не является G-командой SINUMERIK, то затем она проверяется как G-команда ISO.

Запрограммированные переключения (`G290` (режим SINUMERIK), `G291` (режим ISO)) не влияют на STRINGIS.

Пример

Релевантные для функции STRINGIS(...) машинные данные имеют следующие значения:

- MD10711 $MN_NC_LANGUAGE_CONFIGURATION = 2 (как известные рассматриваются только такие языковые команды ЧПУ, опции которых установлены)
- MD19410 $ON_TRAFO_TYPE_MASK = 'H0' (опция: трансформации)
- MD10700 $MN_PREPROCESSING_LEVEL='H43' (предварительная обработка для циклов активна)

Следующая иллюстративная программа выполняется без сообщения об ошибке:

Программный код	Комментарий
`N1 R1=STRINGIS("TRACYL")`	`; R1 == 0,` т.к. `TRACYL` из-за отсутствующей опции трансформации определяется как "не известная"
`N2 IF STRINGIS("TRACYL") == 204`	
`N3  TRACYL(1,2,3)`	`; N3` пропускается
`N4 ELSE`	
`N5  G00`	; и вместо этого выполняется `N5`
`N6 ENDIF`	
`N7 M30`	

4.24.3 Интерактивный вызов окон из программы обработки детали (MMC)

Посредством предопределенной подпрограммы MMC(...) из программы ЧПУ можно отобразить пользовательские диалоги на интерфейс пользователя.

Возможно конфигурирование диалогов следующих типов:

- Run MyScreens
- Easy XML
- XML пользователя

Дополнительная информация:

- Руководство по программированию Run MyScreens
- Руководство по программированию Easy XML

Синтаксис

```
MMC("<ADDRESS>,<COMMAND>,<FILE>,<DIALOG>","<QUIT>")
```

Объяснение

MMC(...):	идентификатор подпрограммы Параметры кодируются по месту и разделяются запятой в пределах двух строк — командной строки и строки квитирования.	
Параметры внутри командной строки:		
<ADDRESS>:	Область управления, в которой выполняются сконфигурированные диалоги с пользователем.	
	Функция	**Области управления**
	Диалог с пользователем "Run MyScreens"	CYCLES
	Диалог с пользователем "Easy XML"	CYCLES
	XML пользователя	XML
	Всплывающее окно "Run MyScreens"	POPUPDLG
	Всплывающее окно "Easy XML"	POPUPDLG
<COMMAND>:	Выполняемая команда	
	Функция	**Команды**
	Диалог с пользователем "Run MyScreens"	PICTURE_ON, PICTURE_OFF
	Диалог с пользователем "Easy XML"	PICTURE_ON, PICTURE_OFF
	XML пользователя	XML_ON, XML_OFF
	Всплывающее окно "Run MyScreens"	PICTURE_ON, PICTURE_OFF
	Всплывающее окно "Easy XML"	PICTURE_ON, PICTURE_OFF
<FILE>:	Имя файла, в котором запрограммирован отображаемый диалог	
	Функция	**Файлы**
	Диалог с пользователем "Run MyScreens"	<name>.com
	Диалог с пользователем "Easy XML"	<name>.xml
	XML пользователя	<name>.xml
	Всплывающее окно "Run MyScreens"	<name>.com
	Всплывающее окно "Easy XML"	<name>.xml
	Всплывающее окно "Easy XML" с конфигурированием непосредственно в программе ЧПУ (см. пример 2)	xmldial_emb.xml

<DIALOG>:	Имя отображаемого диалога		
	Функция		**Имя диалога**
	Все функции, кроме всплывающего окна "Easy XML" с конфигурированием непосредственно в программе ЧПУ		Имя диалога, сконфигурированное в файле <FILE>
	Всплывающее окно "Easy XML" с конфигурированием непосредственно в программе ЧПУ (см. пример 3)		main
Параметры внутри строки квитирования:			
<QUIT>:	Тип квитирования		
	N:	Нет квитирования Обработка программы продолжается после отправки команды. Квитирование не выполняется, если команду не получилось выполнить успешно. **Указание** Тип квитирования "N" используется, если в программе ЧПУ программируется время воспроизведения изображения (время ожидания) (см. ниже Пример 2)	
	A:	Асинхронное квитирование Обработка программы продолжается после подачи команды. Возвращаемое значение сохраняется в спец. переменной квитирования пользователя, определенной в рамках конфигурирования диалога (GUD-переменной) и может считываться в программе ЧПУ.	

Граничные условия

- Файлы определения *.com диалогов необходимо сохранять в папке "proj".
- Файлы определения Easy XML *.xml диалогов необходимо сохранять в папке "appl". Если файлы определений сохранены в другой директории, указание пути должно быть непрямым, исходя из директории "appl".
- Спец. диалоги пользователя невозможно отображать одновременно из различных каналов.
- При симуляции функционал MMC не поддерживается.

Примеры

Пример 1

Индикация диалога и реакция на пользовательское управление в программе ЧПУ.

Программный код	Комментарий
; Переменная квитирования QUIT уже была создана как глобальная переменная пользователя (GUD)	
; типом STRING при конфигурировании диалога:	
; DEF NCK STRING[20] QUIT	
QUIT = "XXX"	; Инициализация переменной квитирования
G4 F5	

Программный код	Комментарий
MMC("CYCLES,PICTURE_ON,test.com,test1","A")	; Показать диалог
	; – Область управления: CYCLES
	; – Статус изображения: PICTURE_ON (отобразить)
	; – Файл изображения диалога: test.com
	; – Изображение диалога: test1
INPUT:	; Ожидание пользовательского управления
STOPRE	; Остановка предварительной обработки
IF MATCH (QUIT,"RUN") >= 0 GOTOF WORK	; Программная клавиша "RUN"
IF MATCH (QUIT,"CHK") >= 0 GOTOF CHECK	; Программная клавиша "CHK"
GOTOB INPUT	; => Ожидание
WORK:	; Программная клавиша "RUN" нажата
MSG("Продолжить обработку -> Пуск ЧПУ")	; Вывести сообщение
MMC("CYCLES,PICTURE_OFF","N")	; Закрыть диалог
M0	; Ожидание пуска ЧПУ
GOTOF END	; => конец программы
CHECK:	; Программная клавиша "CHK" нажата
MSG("Подвести к позиции -> Пуск ЧПУ")	; Вывести сообщение
MMC("CYCLES,PICTURE_OFF","N")	; Закрыть диалог
M0	; Ожидание пуска ЧПУ
GOTOF END	; => конец программы
END:	
...	

Пример 2

Время отображения диалога в программе ЧПУ устанавливается, например, через время ожидания.

Программный код	Комментарий
F1000 G94	
...	
MMC("POPUPDLG,PICTURE_**ON**,xmldial_emb.xml,main","N")	; Показать диалог
X200	
Z40	
MMC("POPUPDLG,PICTURE_**OFF**","N")	; Закрыть диалог

Пример 3

Введение скрипта всплывающего окна в программу ЧПУ и его использование.

Программный код

```
PROC POPUP_TEST
; ---------------------------- Скрипт ----------------------------
```

Программный код

```
; <main_dialog entry="rpara_main">
;    <let name="xpos" />
;    <let name="ypos" />
;    <let name="field_name" type="string" />
;    <let name="num" />
;    <menu name="rpara_main">
;       <open_form name="rpara_form"/>
;       <softkey_back>
;          <close_form />
;       </softkey_back>
;    </menu>
;    <form name="rpara_form">
;       <init>
;          <caption>mask from NC part program</caption>
;          <let name="count" >0</let>
;          <op>
;             xpos = 120;
;             ypos = 34;
;             "nck/Channel/Parameter/R[10]" = 10;
;          </op>
;          <!-- load the number of controls -->
;          <op>
;             num = "nck/Channel/Parameter/R[10]";
;          </op>
;          <while>
;             <condition> count < num</condition>
;             <print name="field_name" text="edit%d">count</print>
;             <op>
;                ypos = ypos + 24;
;                count = count + 1;
;             </op>
;          </while>
;       </init>
;       <paint>
;          <op>
;             xpos =  8;
;             ypos = 36;
;             count = 0;
;          </op>
;          <while>
;             <condition>count < num</condition>
;             <print name="field_name" text="R-Parameter%d">count</print>
;             <text xpos = "$xpos" ypos = "$ypos" >$$$field_name</text>
;             <op>
```

Программный код

```
;                 ypos = ypos + 24;
;                 count = count + 1;
;              </op>
;           </while>
;        </paint>
;     </form>
; </main_dialog>
;========================= Блок программы ============================
...
G94 F100
MMC("POPUPDLG,PICTURE_ON,xmldial_emb.xml,main","N")
G4 F4
X200
MMC("POPUPDLG,PICTURE_OFF","N")
G4 F2
X0
...
```

4.24.4 Process DataShare - Вывод на внешнее устройство/файл (EXTOPEN, WRITE, EXTCLOSE)

Запись данных из программы обработки детали на внешнее устройство / файл осуществляется в три этапа:

1. Открыть внешнее устройство/файл
   С помощью команды EXTOPEN внешнее устройство/файл открывается для канала для записи.
2. Запись данных
   Выводимые данные могут быть подготовлены с помощью строковых функций языка ЧПУ, например, SPRINT. Сама запись выполняется через команду WRITE.
3. Закрыть внешнее устройство/файл
   С помощью команды EXTCLOSE или при достижении конца программы (M30), а также при сбросе канала, занятые в канале внешнее устройство/файл снова разрешаются.

Синтаксис

```
DEF INT <Result>
DEF STRING[<n>] <Output>
…
EXTOPEN(<Result>,<ExtDev>,<SyncMode>,<AccessMode>,<WriteMode>)
…
<Output>=«вывод данных»
WRITE(<Result>,<ExtDev>,<Output>)
```

```
...
EXTCLOSE(<Result>,<ExtDev>)
```

Объяснение

EXTOPEN:	Предварительно определенная процедура для открытия внешнего устройства/файла		
<Result>:	**Параметр 1:** Переменная результата На основе значения переменной результата в программе можно обработать удачный исход операции и выбрать соответствующее продолжение.		
	Тип:	INT	
	Значения:	0	нет ошибок
		1	внешнее устройство не может быть открыто
		2	внешнее устройство не сконфигурировано
		3	внешнее устройство сконфигурировано с недействительным путем
		4	нет прав доступа для внешнего устройства
		5	Режим использования: внешнее устройство уже занято "эксклюзивно"
		6	Режим использования: внешнее устройство уже занято "совместно"
		7	длина файла больше LOCAL_DRIVE_MAX_FILESIZE
		8	макс. число внешних устройств превышено
		9	опция для LOCAL_DRIVE не установлена
		11	зарезервировано
		12	режим записи: данные противоречат extdev.ini
		16	запрограммирован недействительный внешний путь
		22	внешнее устройство не смонтировано

<ExtDev>:	**Параметр 2:** Символический идентификатор для открываемого внешнего устройства/файла	
	Тип:	STRING
	Символический идентификатор состоит из: 1. логического имени устройства 2. возможно с последующим путем к файлу (добавляется с "/"). Определены следующие **логические имена устройств**:	
	"LOCAL_DRIVE":	локальная карта CompactFlash (предопределено)
	"CYC_DRIVE":	указание зарезервированного диска для использования в циклах SIEMENS (предопределено)
	"/dev/ext/1",... "/dev/ext/9":	доступные сетевые диски **Примечание:** Требуется конфигурация в файле extdev.ini!
	"/dev/cyc/1", "/dev/cyc/2":	указание зарезервированного диска для использования в циклах SIEMENS **Примечание:** Требуется конфигурация в файле extdev.ini!
	Путь доступа к файлу: • Для "LOCAL_DRIVE" и "CYC_DRIVE" должен быть указан путь доступа к файлу, к примеру: "LOCAL_DRIVE/my_dir/my_file.txt" • Логические имена устройств "/dev/ext/1...9" и "/dev/cyc/1...2" могут через конфигурацию: – уже ссылаться на файл, тогда может быть указано только логическое имя устройства, к примеру: "/dev/ext/4" – или на директорию, тогда должен быть указан путь к файлу, к примеру: "/dev/ext/5/my_dir/my_file.txt"	
	Примечание: Прописное/строчное написание для логических имен устройств «/dev/ext/1...9», «/dev/v24» и «/dev/cyc/1...2» игнорируется, при указании пути к файлу прописное/строчное написание значимо. Для "LOCAL_DRIVE" и "CYC_DRIVE" разрешается только прописное написание.	

<SyncMode>:	**Параметр 3:** Режим обработки для команд WRITE для этого устройства/файла		
	Тип:	STRING	
	Значения:	"SYN":	Синхронная запись Выполнение программы останавливается до завершения процесса записи. Успешное завершение синхронной записи может быть проверено через обработку переменных ошибок команды WRITE.
		"ASYN":	Асинхронная запись Выполнение программы не прерывается командой WRITE. **Примечание:** Переменная результата команды WRITE в этом режиме бессодержательна и всегда имеет значение 0 (ошибки отсутствуют). В этом режиме отсутствует уверенность, что команда WRITE была успешной.
<AccessMode>:	**Параметр 4:** Режим использования для этого устройства/файла		
	Тип:	STRING	
	Значения:	"SHARED":	Устройство/файл запрошен в "совместном" режиме. Другие каналы также могут использовать этой устройство, т. е. также открыть в этом режиме.
		"EXCL":	Устройство/файл используется эксклюзивно в канале, никакой другой канал не может использовать устройства.
<WriteMode>:	**Параметр 5:** Режим записи для команд WRITE для этого устройства/файла (опция)		
	Тип:	STRING	
	Значения:	"APP":	Присоединение Файл сохраняется в своем содержании, вызовы записи присоединяются в конце.
		"OVR":	Перезапись Содержание файла удаляется и заново создается через последующие вызовы записи.
	Примечание: С этим параметром сконфигурированный в файле extdev.ini режим записи не может быть переписан. При конфликте вызов EXTOPEN квитируется с ошибкой.		

WRITE:	Предварительно определенная процедура для записи выводимых данных

EXTCLOSE:	Предварительно определенная процедура для закрытия открытого внешнего устройства/файла		
<Result>:	**Параметр 1:** Переменная результата		
	Тип:	INT	
	Значения:	0	нет ошибок
		16	запрограммирован недействительный внешний путь
		21	ошибка при закрытии внешнего устройства
<ExtDev>:	**Параметр 2:** Символический идентификатор для закрываемого внешнего устройства/файла. Описание см. EXTOPEN! **Примечание:** Идентификатор должен быть идентичен указанному в команде EXTOPEN идентификатору!		

Пример

Программный код

```
N10       DEF INT RESULT
N20       DEF BOOL EXTDEVICE
N30       DEF STRING[80] OUTPUT
N40       DEF INT PHASE
N50       EXTOPEN(RESULT,"LOCAL_DRIVE/my_file.txt","SYN","SHARED")
N60       IF RESULT > 0
N70          MSG("Ошибка в EXTOPEN:" << RESULT)
N80       ELSE
N90          EXTDEVICE=TRUE
N100      ENDIF
…
N200      PHASE=4
N210      IF EXTDEVICE
N220         OUTPUT=SPRINT("конец фазы: %D",PHASE)
N230         WRITE(RESULT,"LOCAL_DRIVE/my_file.txt",OUTPUT)
N240      ENDIF
…
```

См. также

Операции со строкой (Страница 461)

Записать файл (WRITE) (Страница 585)

4.24.5 Ошибки (SETAL)

В программе ЧПУ могут устанавливаться аварийные сообщения. Они отображаются на интерфейсе пользователя в особом поле. С аварийным сообщением связана реакция СЧПУ согласно категории аварийного сообщения.

Синтаксис

```
SETAL(<Nr>[,<String>])
```

Объяснение

SETAL(...):	Предопределенная процедура для установки аварийного сообщения SETAL должна быть запрограммирована в собственном кадре ЧПУ.		
<Nr>:	Номер ошибки		
	Тип данных:	INT	
	Диапазон значений:	60000 ... 64999 (зарезервировано)	Аварийные сообщения циклов SIEMENS
		65000 ... 69999	Аварийные сообщения циклов пользователя
<строка>:	Строка символов (опционально)		
	Тип данных:	STRING	
	При программировании аварийных сообщений циклов пользователя может быть указана строка символов с макс. 4 параметрами. В этих параметрах могут быть определены переменные тексты пользователя. Доступны также предопределенные параметры:		
	%1	Номер канала	
	%2	Номер кадра, метка	
	%3	Индекс текста для аварийных сообщений циклов	
	%4	Дополнительный параметр аварийного сообщения	

Примечание

Тексты аварийных сообщений должны быть сконфигурированы на интерфейсе пользователя.

Примечание

Если аварийное сообщение должно быть выведено на активном на интерфейсе пользователя языке, то пользователю нужна информация о текущем установленном на HMI языке. Эта информация запрашивается в программе обработки детали и в синхронных действиях через системную переменную $AN_LANGUAGE_ON_HMI (Страница 1364).

Пример

Программный код	Комментарий
...	
N100 SETAL(65000)	; установить аварийное сообщение № 65000
...	

Дополнительная информация

Реакция на аварийное сообщение и квитирование

Аварийные сообщения циклов пользователя соотнесены с диапазонами номеров, которые различаются по реакциям на аварийное сообщение и квитированию:

Диапазон номеров	Реакция на ошибку	Квитирование аварийного сообщения
65000 - 65499	Индикация, блокировка NC-Start	Сброс
65500 - 65999	Индикация, блокировка NC-Start (не для ASUP при установленной MD20194)	Сброс
66000 - 66999	Индикация, блокировка NC-Start, остановка движения после обработки предварительно декодированных кадров	Сброс
67000 - 67999	Индикация	Cancel
68000 - 68999	Индикация, блокировка NC-Start, немедленный останов интерполятора	Сброс
69000 - 69999	Индикация, блокировка NC-Start, останов на следующем конце кадра	Сброс

4.24.6 Определение заготовки (WORKPIECE)

Система управления должна знать форму и размер заготовки, чтобы отобразить ее при графическом моделировании. Поэтому пользователь может определить заготовки через интерфейс пользователя или прямо в программе ЧПУ. Определения заготовок сохраняются и при сбросах (конца программы / канала / ГРР). При следующем запуске системы управления они автоматически удаляются.

Синтаксис

```
WORKPIECE("<WP>", "<RefP>", "<ZeroOffset>", "<Type>", <Par5>,
<Par6>, ..., <Par12>)
```

Объяснение

`WORKPIECE(...)`:	Предопределенная процедура для определения заготовки	
	Остановка предварительной обработки:	да
	Автономно в кадре:	да
Параметры:		

1	"<WP>":	Название заготовки (опционально)		
		Тип данных:	STRING	
		Указание необходимо, только если в одном канале могут быть несколько заготовок. При отсутствии указания автоматически принимается "WORKP<n>", при этом <n> — номер объявляющего канала.		
2	"<RefP>":	Зажим (опционально; только для фрезерных станков)		
		Тип данных:	STRING	
		Диапазон значений:	"Стол"	Зажим на стационарном столе
			"A"	Зажим на круговой оси A
			"B"	Зажим на круговой оси B
			"C"	Зажим на круговой оси C
		Условие: Стол или круговая ось должны быть разрешены для зажима заготовки в соответствующих машинных данных (см. Руководство по вводу в эксплуатацию SINUMERIK Operate).		
3	"<ZeroOffset>":	Настраиваемое смещение нулевой точки для размещения заготовки (не программируется) Выбор настраиваемого смещения нулевой точки для размещения заготовки предлагается только при вводе заготовки через интерфейс пользователя. И напротив, при прямом определении заготовки в программе обработки детали заготовка всегда относится к действительному на соответствующий момент смещению нулевой точки.		
4	"<Type>":	Форма заготовки		
		Тип данных:	STRING	
		Диапазон значений:	"CYLINDER":	Цилиндр
			"PIPE":	Труба
			"RECTANGLE":	Прямоугольный параллелепипед по центру
			"BOX":	Прямоугольный параллелепипед
			"N_CORNER":	Многоугольник
5 ... 12	<Par5> ... <Par12>:	Параметры описания формы заготовки		
		Тип данных:	REAL	
		Количество необходимых параметров и их значение зависят от формы соответствующей заготовки и значения параметра битового формата. См.: • Таблица "Параметры описания формы заготовки" • Таблица "Параметры битового формата"		
WORPIECE():		Вызов WORKPIECE без параметров удаляет все определения заготовок.		
WORPIECE(<WP>):		Вызов WORKPIECE с именами деталей удаляет только эти определения заготовок.		

Таблица 4-6 Параметры описания формы заготовки

Форма заготовки	**Параметр**							
	<Par5>	**<Par6>**	**<Par7>**	**<Par8>**	**<Par9>**	**<Par10>**	**<Par11>**	**<Par12>**
Цилиндр	Параметры битового формата Действительное значение, интерпретируемое как кодированное в битах целочисленное число. Биты определяют значение следующих параметров (см. таблицу "Параметры битового формата").	Исходная точка Z_0	Длина Z_1	Размер обработки Z_B	Наружный диаметр d_0	-	Вращение вокруг круговой оси	-
Труба		Исходная точка Z_0	Длина Z_1	Размер обработки Z_B	Наружный диаметр d_0	Толщина стенки (инкр.) / внутренний диаметр d_1 (абс.)	Вращение вокруг круговой оси	-
Прямоугольный параллелепипед по центру		Исходная точка Z_0	Длина Z_1	Размер обработки Z_B	Ширина W	Длина L	Вращение вокруг круговой оси	-
Прямоугольный параллелепипед		Исходная точка Z_0	Длина Z_1	Размер обработки Z_B	X_0	Y_0	X_1	Y_1
Многоугольник		Исходная точка Z_0	Длина Z_1	Размер обработки Z_B	Количество углов	Размер ключа	Вращение вокруг круговой оси	-

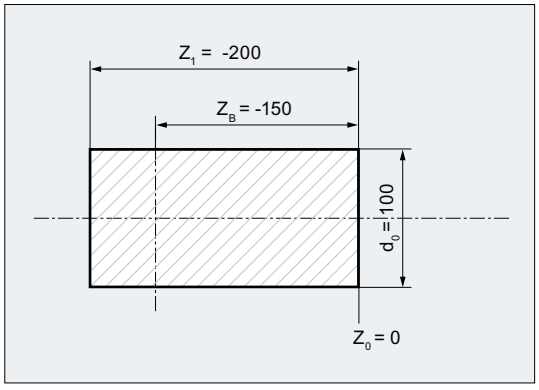
Таблица 4-7 Параметры битового формата

Бит	**Объяснение**	
4 (0x0010)	Прямоугольный параллелепипед: X_1	
	=0	инкр.
	=1	абс.
5 (0x0020)	Прямоугольный параллелепипед: Y_1	
	=0	инкр.
	=1	абс.
6 (0x0040)	Длина Z_1 (конечный размер)	
	=0	инкр.
	=1	абс.
Бит 7 (0x0080)	Размер обработки Z_B	
	=0	инкр.
	=1	абс.
Бит 8 (0x0100)	Труба: Толщина стенки / Внутренний диаметр	
	=0	инкр.
	=1	абс.

Бит	Объяснение	
9 (0x0200)	Многоугольник	
	=0	Размер ключа
	=1	Длина грани
12 (0x1000)	Зажим для токарных станков	
	=0	Главный шпиндель
	=1	Противошпиндель
13 (0x2000)	Противошпиндель	
	=0	с отражением
	=1	без отражения

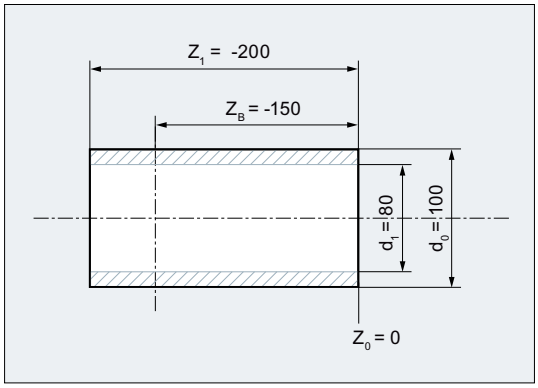
Примеры

Пример 1: Цилиндрическая заготовка на токарном станке



Программный код	Комментарий
...	
`WORKPIECE(,,,"CYLINDER",0,0,-200,-150,100)`	; **Определение заготовки:** ; Форма заготовки: Цилиндр ; Параметр битового формата =0 (бит не установлен) → Значения длины и размера обработки — инкрементальные, заготовка на главном шпинделе ; Исходная точка (`Z0)=0` ; Длина (`Z1)=-200` ; Размер обработки (`ZB)=-150` ; Наружный диаметр (`d0)=100`
...	

Пример 2: Трубообразная заготовка на токарном станке



Программный код	Комментарий
...	
WORKPIECE(,,,"PIPE",256,0,-200,-150,100,80)	; **Определение заготовки:** ; Форма заготовки: Труба ; Параметр битового формата =256(бит 8=1) → Значение внутреннего диаметра — абсолютное, значения длины и размера обработки — инкрементальные, заготовка на главном шпинделе ; Исходная точка (Z0)=0 ; Длина (Z1)=-200 ; Размер обработки (ZB)=-150 ; Наружный диаметр (d0)=100 ; Внутренний диаметр (d1)=80
...	

4.24.7 Переключение языкового режима (G290, G291)

Система управления позволяет считывать и обрабатывать программы обработки детали с внешних систем CNC. Для этого необходимо при вводе в эксплуатацию установить соответствующий языковой режим ЧПУ (диалект ISO).

Режим диалекта ISO можно активировать отдельно для каждого канала. Например, канал 1 может идти в режиме диалекта ISO, канал 2 в режиме SINUMERIK.

Переключение с режима SINUMERIK на режим диалекта ISO выполняется в программе ЧПУ командами G-группы 47. На активный инструмент, коррекции на инструмент и смещения нулевой точки переключение не влияет.

Синтаксис

```
G291
...
G290
```

Объяснение

G290:	Активировать языковой режим SINUMERIK	
	Автономно в кадре:	да
	Активность:	модально
G291:	Активировать языковой режим ISO	
	Автономно в кадре:	да
	Активность:	модально

Условия

Режим SINUMERIK

- Через машинные данные можно сохранить предустановки G-команд каждого канала.
- В режиме SINUMERIK невозможно программировать языковые команды диалектов ISO.

Режим диалекта ISO

- Через машинные данные можно настроить режим диалекта ISO в качестве установки системы управления по умолчанию. В этом случае система управления запускается в режиме диалекта ISO.
- Возможно программирование только G-команд диалекта ISO. Программировать функции SINUMERIK в режиме диалекта ISO невозможно.
- Комбинировать диалекта ISO и языка SINUMERIK в одном кадре ЧПУ невозможно.
- Переключение через G-команды между диалектом M ISO (фрезерная обработка) и диалектом T ISO (токарная обработка) невозможно.
- Возможен вызов подпрограмм, запрограммированных в режиме SINUMERIK.
- При необходимости использования функций SINUMERIK необходимо сначала переключиться в режим SINUMERIK. (см. пример).

Пример

Сжатие линейных кадров в режиме диалекта ISO

Программный код	Комментарий
`N5 G290`	; Активировать языковой режим `SINUMERIK`
`N10 COMPCAD`	`;` `COMPCAD` – команда языка `Siemens`, которая активирует функцию компрессора, заменяющую последовательные линейные кадры на полиномиальные кадры с максимально длинными траекториями.
`N15 G291`	; Активировать языковой режим `ISO`.
`N20 G01 X100 Y100 F1000`	; Поскольку `COMPCAD` была активирована в режиме `SINUMERIK`, она способна сжимать и линейные кадры в режиме диалекта `ISO`.
...	

Дополнительная информация

Описание функций "Диалекты ISO"

4.25 Собственные программы обработки резаньем

4.25.1 Функции поддержки для обработки резаньем

Для обработки резаньем предлагаются готовые циклы обработки. Кроме этого, с помощью перечисленных ниже функций можно создавать собственные программы обработки резаньем:

- Создание таблицы контуров (CONTPRON)
- Создание кодированной таблицы контуров (CONTDCON)
- Отключить подготовку контура (EXECUTE)
- Определить точку пересечения между двумя элементами контура (INTERSEC) (только для таблиц, созданных через CONTPRON).
- Покадровая обработка элементов контура таблицы (EXECTAB) (только для таблиц, созданных через CONTPRON).
- Вычисление данных окружности (CALCDAT)

Примечание

Эти функции могут использоваться не только для обработки резаньем, но и универсально.

Условия

Перед вызовом функций CONTPRON или CONTDCON должен быть:

- выполнен подвод к стартовой точке, обеспечивающей обработку без столкновений.
- коррекция радиуса резцов с `G40` должна быть отключена.

4.25.2 Создание таблицы контуров (CONTPRON)

С помощью `CONTPRON` включается подготовка контура. Вызванные после кадры ЧПУ не выполняются, а разбиваются на отдельные движения и сохраняются в таблицу контура. Каждому элементу контура соответствует строка в таблице в двухмерном поле таблицы контуров. Возвращается число полученных поднутрений.

Синтаксис

Включить подготовку контура:

CONTPRON(<таблица контура>,<режим обработки>,<поднутрения>,
<направление обработки>)

Отключить подготовку контура и одновременно вернуться в обычный режим обработки:
EXECUTE(<ОШИБКА>)

См. " Отключить подготовку контура (EXECUTE) (Страница 1065) "

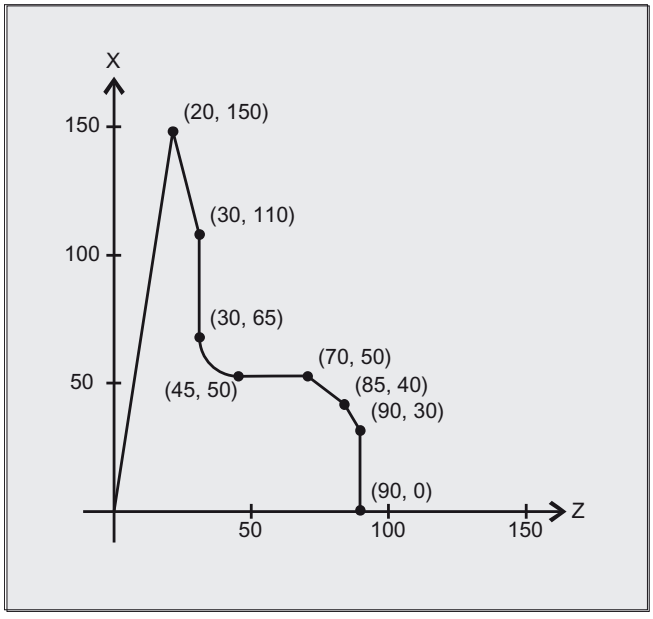
Объяснение

CONTPRON:	Предопределенная процедура для включения подготовки контура для создания таблицы контура		
<таблица контура>:	Имя таблицы контура		
<режим обработки>:	Параметры для режима обработки		
	Тип:	CHAR	
	Значение:	"G":	Продольная обточка: внутренняя обработка
		"L":	Продольная обточка: наружная обработка
		"N":	Торцевание: внутренняя обработка
		"P":	Торцевание: наружная обработка
<поднутрения>:	Переменная результата для числа возникших элементов поднутрения		
	Тип:	INT	
<направление обработки>:	Параметры для направления обработки		
	Тип:	INT	
	Значение:	0	Подготовка контура вперед (стандартное значение)
		1	Подготовка контура в обоих направлениях

Пример 1

Создание таблицы контура с:

- именем "KTAB"
- макс. 30 элементами контура (окружности, прямые)
- переменной для числа встречающихся элементов поднутрений
- переменной для сообщений об ошибках



Программа ЧПУ:

Программный код	Комментарий
`N10 DEF REAL KTAB[30,11]`	; Таблица контура с именем `KTAB` и макс. 30 элементами контура, значение параметра 11 (число колонок таблицы) это фиксированная величина.
`N20 DEF INT ANZHINT`	; Переменная для числа элементов поднутрений с именем `ANZHINT`.
`N30 DEF INT FEHLER`	; Переменная для квитирования ошибок (0=нет ошибок, 1=ошибка).
`N40 G18`	
`N50 CONTPRON (KTAB,"G",ANZHINT)`	; Включить подготовку контура.
`N60 G1 X150 Z20`	`; N60` до `N120:` Описание контура
`N70 X110 Z30`	
`N80 X50 RND=15`	
`N90 Z70`	
`N100 X40 Z85`	
`N110 X30 Z90`	
`N120 X0`	
`N130 EXECUTE(FEHLER)`	; Завершить заполнение таблицы контура, переключение на обычный программный режим.
`N140 …`	; Продолжить обработку таблицы.

Таблица контура KTAB:

Индекс Строка	Графа									
(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)

7	7	11	0	0	20	150	0	82.40535663	0	0
0	2	11	20	150	30	110	-1111	104.0362435	0	0
1	3	11	30	110	30	65	0	90	0	0
2	4	13	30	65	45	50	0	180	45	65
3	5	11	45	50	70	50	0	0	0	0
4	6	11	70	50	85	40	0	146.3099325	0	0
5	7	11	85	40	90	30	0	116.5650512	0	0
6	0	11	90	30	90	0	0	90	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

Объяснение содержания колонок:

(0) Указатель на следующий элемент контура (на номер строки таковой)

(1) Указатель на предшествующий элемент контура

(2) Кодировка режима контура для движения

Возможные значения для X = abc

a = 10^2 G90 = 0 G91 = 1

b = 10^1 G70 = 0 G71 = 1

c = 10^0 G0 = 0 G1 = 1 G2 = 2 G3 = 3

(3), (4) Начальная точка элементов контура

(3) = абсцисса, (4) = ордината в актуальной плоскости

(5), (6) Конечная точка элемента контура

(5) = абсцисса, (6) = ордината в актуальной плоскости

(7) Указатель макс./мин.: обозначает локальные максимум и минимум на контуре

(8) Макс. значение между элементом контура и абсциссой (при продольной обработке) или ординатой (при поперечной обработке). Угол зависит от запрограммированного режима обработки.

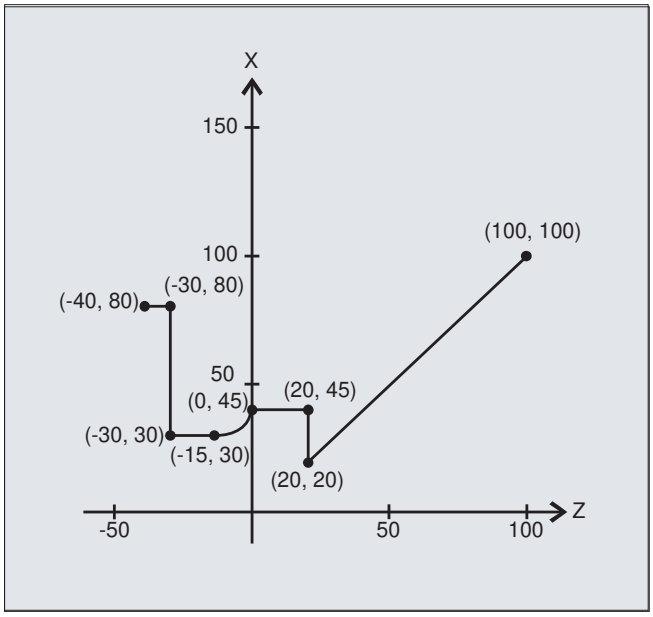
(9), (10) координаты центра элемента контура, если это круговой кадр.

(9) = абсцисса, (10) = ордината

Пример 2

Создание таблицы контура с

- именем KTAB
- макс. 92 элементами контура (окружности, прямые)
- режим работы: продольная обточка, наружная обработка
- подготовка вперед и назад



Программа ЧПУ:

Программный код	Комментарий
N10 DEF REAL KTAB[92.11]	; Таблица контура с именем KTAB и макс. 92 элементами контура, значение параметра 11 это фиксированная величина.
N20 DEF CHAR BT="L"	; Режим работы для CONTPRON: продольная обточка, наружная обработка
N30 DEF INT HE=0	; Число элементов поднутрения=0
N40 DEF INT MODE=1	; Подготовка вперед и назад
N50 DEF INT ERR=0	; Квитирование ошибок
...	
N100 G18 X100 Z100 F1000	
N105 CONTPRON(KTAB,BT,HE,MODE)	; Включить подготовку контура.
N110 G1 G90 Z20 X20	
N120 X45	
N130 Z0	
N140 G2 Z-15 X30 K=AC(-15) I=AC(45)	
N150 G1 Z-30	
N160 X80	
N170 Z-40	
N180 EXECUTE(ERR)	; Завершить заполнение таблицы контура, переключение на обычный программный режим.
...	

Таблица контура KTAB:

После завершения подготовки контура он доступен в обоих направлениях.

Индекс	Колонка										
Строка	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
0	6[1)]	7[2)]	11	100	100	20	20	0	45	0	0
1	0[3)]	2	11	20	20	20	45	-3	90	0	0
2	1	3	11	20	45	0	45	0	0	0	0
3	2	4	12	0	45	-15	30	5	90	-15	45
4	3	5	11	-15	30	-30	30	0	0	0	0
5	4	7	11	-30	30	-30	45	-1111	90	0	0
6	7	0[4)]	11	-30	80	-40	80	0	0	0	0
7	5	6	11	-30	45	-30	80	0	90	0	0
8	1[5)]	2[6)]	0	0	0	0	0	0	0	0	0
	...										
83	84	0[7)]	11	20	45	20	80	0	90	0	0
84	90	83	11	20	20	20	45	-1111	90	0	0
85	0[8)]	86	11	-40	80	-30	80	0	0	0	0
86	85	87	11	-30	80	-30	30	88	90	0	0
87	86	88	11	-30	30	-15	30	0	0	0	0
88	87	89	13	-15	30	0	45	-90	90	-15	45
89	88	90	11	0	45	20	45	0	0	0	0
90	89	84	11	20	45	20	20	84	90	0	0
91	83[9)]	85[10)]	11	20	20	100	100	0	45	0	0

Объяснение содержания колонок и примечаний к строкам 0, 1, 6, 8, 83, 85 и 91

Действуют названные в примере 1 пояснения содержания колонок.

Всегда в строке таблицы 0:

1) Предшествующий элемент: строка n содержит конец контура вперед

2) Следующий элемент: строка n это конец таблицы контура вперед

По одному разу в элементах контура вперед:

3) Предшествующий элемент: начало контура (вперед)

4) Следующий элемент: конец контура (вперед)

Всегда на строку конца таблицы контура (вперед) +1:

5) Предшествующий элемент: число поднутрений вперед

6) Следующий элемент: число поднутрений назад

По одному разу в элементах контура назад:

7) Следующий элемент: конец контура (назад)

8) Предшествующий элемент: начало контура (назад)

Всегда в последней строке таблицы:

9) Предшествующий элемент: строка n это начало таблицы контура (назад)

10) Следующий элемент: строка n содержит начало контура (назад)

Дополнительная информация

Разрешенные команды движения, система координат

Для программирования контура разрешены следующие команды G:

- G-группа 1: G0, G1, G2, G3

Дополнительно возможны:

- Закругление и фаска
- Программирование окружности через CIP и CT

Функции "Сплайн", "Полином" и "Резьба" приводят к ошибкам.

Изменения системы координат через включение фрейма между CONTPRON и EXECUTE не допускаются. То же самое действительно для переключения между G70 и G71 или между G700 и G710.

Переход гео-осей с GEOAX при подготовке таблицы контура приводит к ошибке.

Элементы поднутрения

Описание контура отдельных элементов поднутрения может осуществляться по выбору в подпрограмме или в отдельных кадрах.

Обработка резаньем независимо от запрограммированного направления контура

Подготовка контура с CONTPRON была расширена таким образом, что после ее вызова таблица контура доступна независимо от запрограммированного направления.

4.25.3 Создание кодированной таблицы контуров (CONTDCON)

При включенной с CONTDCON подготовкой контура вызванные после кадры ЧПУ сохраняются в таблице контура с 6 колонками в удобной для памяти кодировке. Каждому элементу контура соответствует строка таблицы контура. Зная указанные ниже правила кодировки можно, например, создавать для циклов из строк таблицы программы кода DIN. В строке таблицы с номером 0 сохраняются данные исходной точки.

Синтаксис

Включить подготовку контура:
CONTDCON(<таблица контура>,<направление обработки>)

Отключить подготовку контура и одновременно вернуться в обычный режим обработки:
EXECUTE(<ОШИБКА>)

См. " Отключить подготовку контура (EXECUTE) (Страница 1065) "

Объяснение

CONTDCON:	Предопределенная процедура для включения подготовки контура для создания кодированной таблицы контура		
<таблица контура>:	Имя таблицы контура		
<направление обработки>:	Параметры для направления обработки		
	Тип:	INT	
	Значение:	0	Подготовка контура согласно последовательности кадров контура (стандартное значение)
		1	недопустимо

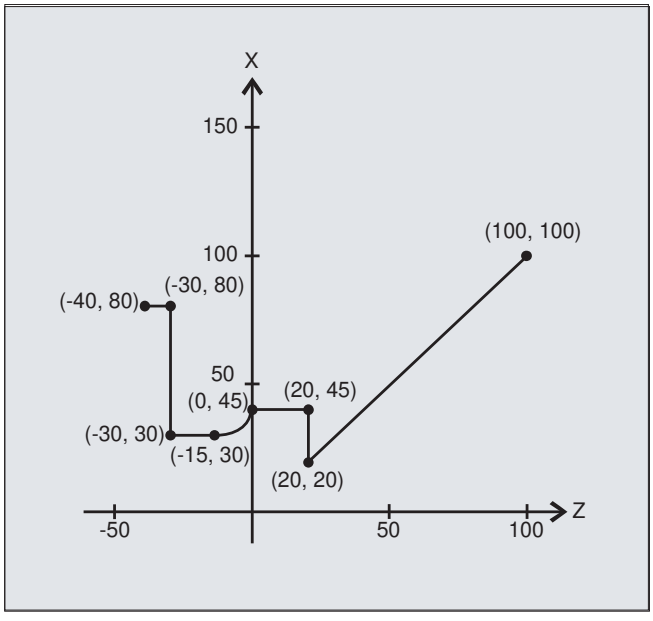
Примечание

Разрешенные для CONTDCON G-команды в табулируемом сегменте программы объемнее, чем при CONTPRON. Кроме этого, также сохраняются подачи и тип подачи на участок контура.

Пример

Создание таблицы контура с:

- именем "KTAB"
- элементами контура (окружности, прямые)
- Режим работы: токарная обработка
- Направление обработки: вперед



Программа ЧПУ:

Программный код	Комментарий
`N10 DEF REAL KTAB[9.6]`	; Таблица контура с именем `KTAB` и 9 строками. Они допускают 8 кадров контура. Значение параметра 6 (число колонок в таблице) это фиксированная величина.
`N20 DEF INT MODE = 0`	; Переменная для направления обработки. Стандартное значение 0: только в запрограммированном направлении контура.
`N30 DEF INT ERROR = 0`	; Переменная для квитирования ошибок.
...	
`N100 G18 G64 G90 G94 G710`	
`N101 G1 Z100 X100 F1000`	
`N105 CONTDCON (KTAB, MODE)`	; Вызов подготовки контура (`MODE` можно опустить).
`N110 G1 Z20 X20 F200`	; Описание контура.
`N120 G9 X45 F300`	
`N130 Z0 F400`	
`N140 G2 Z-15 X30 K=AC(-15) I=AC(45)F100`	
`N150 G64 Z-30 F600`	
`N160 X80 F700`	
`N170 Z-40 F800`	
`N180 EXECUTE(ERROR)`	; Завершить заполнение таблицы контура, переключение на обычный программный режим.
...	

Таблица контура KTAB:

	Индекс колонки					
	0	1	2	3	4	5
Индекс строки	Режим контура	Конечная точка абсцисса	Конечная точка ордината	Центр абсцисса	Центр ордината	Подача
0	30	100	100	0	0	7
1	11031	20	20	0	0	200
2	111031	20	45	0	0	300
3	11031	0	45	0	0	400
4	11032	-15	30	-15	45	100
5	11031	-30	30	0	0	600
6	11031	-30	80	0	0	700
7	11031	-40	80	0	0	800
8	0	0	0	0	0	0

Объяснение содержания колонок:

Строка 0: кодировки для **стартовой точки:**

Колонка 0: 10^0 (первая позиция): G0 = 0
10^1 (вторая позиция): G70 = 0, G71 = 1, G700 = 2, G710 = 3
Колонка 1: стартовая точка абсцисса
Колонка 2: стартовая точка ордината
Колонка 3-4: 0
Колонка 5: индекс строки последнего участка контура в таблице

Строки 1-n: записи **участков контура**

Колонка 0: 10^0 (первая позиция): G0 = 0, G1 = 1, G2 = 2, G3 = 3
10^1 (вторая позиция): G70 = 0, G71 = 1, G700 = 2, G710 = 3
10^2 (третья позиция): G90 = 0, G91 = 1
10^3 (четвертая позиция): G93 = 0, G94 = 1, G95 = 2, G96 = 3
10^4 (пятая позиция): G60 = 0, G44 = 1, G641 = 2, G642 = 3
10^5 (шестая позиция): G9 = 1
Колонка 1: конечная точка абсцисса
Колонка 2: конечная точка ордината
Колонка 3: центр абсцисса при круговой интерполяции
Колонка 4: центр ордината при круговой интерполяции
Колонка 5: подача

Дополнительная информация

Разрешенные команды движения, система координат

Для программирования контура разрешены следующие группы G и команды G:

G-группа 1: `G0`, `G1`, `G2`, `G3`
G-группа 10: `G60`, `G64`, `G641`, `G642`
G-группа 11: `G9`
G-группа 13: `G70`, `G71`, `G700`, `G710`
G-группа 14: `G90`, `G91`
G-группа 15: `G93`, `G94`, `G95`, `G96`, `G961`

Дополнительно возможны:

- Закругление и фаска
- Программирование окружности через `CIP` и `CT`

Функции "Сплайн", "Полином" и "Резьба" приводят к ошибкам.

Изменения системы координат через включение фрейма между `CONTDCON` и `EXECUTE` не допускаются. То же самое действительно для переключения между `G70` и `G71` или между `G700` и `G710`.

Переход гео-осей с `GEOAX` при подготовке таблицы контура приводит к ошибке.

Направление обработки

Создаваемая с помощью CONTDCON таблица контура предусмотрена для обработки резанием в запрограммированном направлении контура.

4.25.4 Определить точку пересечения между двумя элементами контура (INTERSEC)

INTERSEC определяет точку пересечения двух нормированных элементов контура из созданных с помощью CONTPRON таблиц контуров.

Синтаксис

<состояние>=INTERSEC(<таблица контура_1>[<элемент контура_1>],
<таблица контура_2>[<элемент контура_2>],<точка пересечения>,<режим обработки>)

Объяснение

INTERSEC:	Предопределенная функция для определения точки пересечения двух элементов контура из созданных с помощью CONTPRON таблиц контуров		
<состояние>:	Переменная для состояния точки пересечения		
	Тип:	BOOL	
	Значение:	TRUE	точка пересечения найдена
		false	точка пересечения не найдена
<таблица контура_1>:	имя первой таблицы контура		
<элемент контура_1>:	номер элемента контура первой таблицы контура		
<таблица контура_2>:	имя второй таблицы контура		
<элемент контура_2>:	номер элемента контура второй таблицы контура		
<точка пересечения>:	координаты точки пересечения в активной плоскости (G17 / G18 / G19)		
	Тип:	REAL	
<режим обработки>:	параметры для режима обработки		
	Тип:	INT	
	Значение:	0	вычисление точки пересечения в активной с параметром 2 плоскости (стандартное значение)
		1	вычисление точки пересечения независимо от переданной плоскости

Примечание

Учитывать, что переменные перед их использованием должны быть определены.

Для передачи контуров необходимо соблюдать определенные с CONTPRON значения:

Параметры	Значение
2	Кодировка режима контура для движения
3	Начальная точка контура абсцисса
4	Начальная точка контура ордината
5	Конечная точка контура абсцисса
6	Конечная точка контура ордината
9	Координата центра для абсциссы (только для кругового контура)
10	Координата центра для ординаты (только для кругового контура)

Пример

Определить точку пересечения элемента контура 3 таблицы TABNAME1 и элемента контура 7 таблицы TABNAME2. Координаты точки пересечения в активной плоскости сохраняются в переменной SCHNITT (1-й элемент = абсцисса, 2-й элемент = ордината). Если точки пересечения не существует, то выполняется переход на KEINSCH (точка пересечения не найдена).

Программный код	Комментарий
`DEF REAL TABNAME1[12,11]`	; Таблица контура 1
`DEF REAL TABNAME2[10,11]`	; Таблица контура 2
`DEF REAL ISCOORD[2]`	; Переменная для координат точки пересечения.
`DEF BOOL ISPOINT`	; Переменная для состояния точки пересечения.
`DEF INT MODE`	; Переменная для режима обработки.
…	
`MODE=1`	; Вычисление независимо от активной плоскости.
`N10 ISPOINT=INTERSEC(TABNAME1[3],TABNAME2[7], ISCOORD,MODE)`	; Вызов точки пересечения элементов контура.
`N20 IF ISPOINT==FALSE GOTOF KEINSCH`	; Переход на `KEINSCH`.
…	

4.25.5 Покадровая обработка элементов контура таблицы (EXECTAB)

С помощью EXECTAB возможна покадровая обработка элементов контура таблицы, которые, к примеру, были созданы с помощью CONTPRON .

Синтаксис

EXECTAB (<таблица контура>[<элемент контура>])

Объяснение

EXECTAB:	Предопределенная процедура для обработки элемента контура
<таблица контура>:	Имя таблицы контура
<элемент контура>:	Номер элемента контура

Пример

Элементы контура 0 до 2 таблицы KTAB должны быть обработаны покадрово.

Программный код	Комментарий
N10 EXECTAB(KTAB[0])	; Перемещение элемента 0 таблицы KTAB.
N20 EXECTAB(KTAB[1])	; Перемещение элемента 1 таблицы KTAB.
N30 EXECTAB(KTAB[2])	; Перемещение элемента 2 таблицы KTAB.

4.25.6 Вычисление данных окружности (CALCDAT)

С помощью CALCDAT из трех или четырех известных точек окружности можно вычислить радиус и координаты центра окружности. Указанные точки не должны совпадать.

В случае 4 точек, расположенных не точно на окружности, для центра окружности и радиуса формируется среднее значение.

Примечание

Правила вычисления для формирования среднего значения

Вычисление дуги окружности выполняется 4 раза:

1. с точками окружности 1, 2, 3
2. с точками окружности 1, 2, 4
3. с точками окружности 1, 3, 4
4. с точками окружности 2, 3, 4

Координаты центра окружности (значение абсциссы и значение ординаты) вычисляются через сложение значений абсциссы/ординаты, полученных в результате четырех вычислений дуги окружности, и деление суммы на 4.

Для вычисления радиуса берется корень из суммы четырех радиусов, полученных при вычислениях дуги окружности, и результат умножается на 0,5.

Синтаксис

<состояние>=CALCDAT(<точки окружности>[<число>,<вид>],<число>,<результат>)

Объяснение

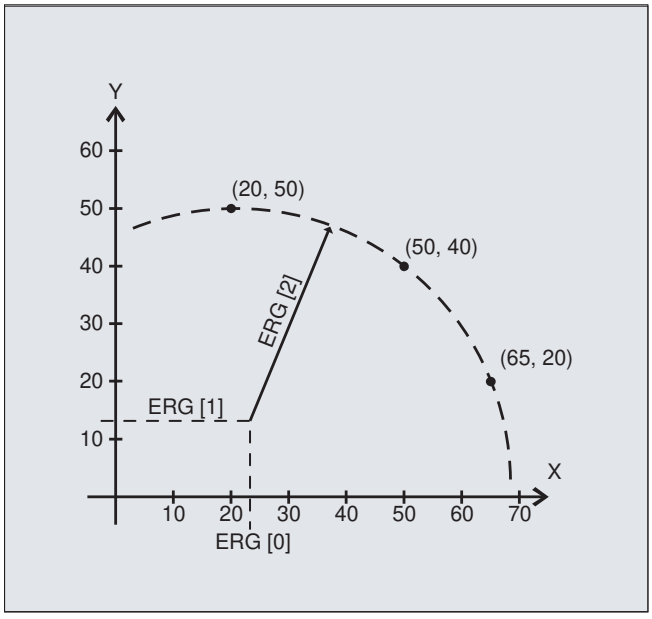
CALCDAT:	Предопределенная функция для вычисления радиуса и координат центра окружности из 3 или 4 точек		
<состояние>:	Переменная для состояния вычисления окружности		
	Тип:	BOOL	
	Значение:	TRUE	Указанный точки лежат на одной окружности.
		false	Указанный точки **не** лежат на одной окружности.
<точки окружности>[]:	Переменная для указания точек окружности с параметрами:		
	<число>:	Число точек окружности (3 или 4)	
	<вид>:	Вид указания координат, к примеру, 2 для 2-точечных координат	
<число>:	Параметр для числа используемых для вычисления точек (3 или 4)		
<результат>[3]:	Переменная для результата: Указание координат центра окружности и радиуса		
	0	Координата центра окружности: значение абсциссы	
	1	Координата центра окружности: значение ординаты	
	2	Радиус	

Примечание

Учитывать, что переменные перед их использованием должны быть определены.

Пример

Из трех точек необходимо определить, лежат ли они на круговом сегменте.



Программный код	Комментарий
`N10 DEF REAL PKT[3,2]=(20,50,50,40,65,20)`	; Переменная для указания точек окружности
`N20 DEF REAL ERG[3]`	; Переменная для результата:
`N30 DEF BOOL STATUS`	; Переменная для состояния.
`N40 STATUS=CALCDAT(PKT,3,ERG)`	; Вызов полученных данных окружности.
`N50 IF STATUS == FALSE GOTOF ERROR`	; Переход на ошибку.

4.25.7 Отключить подготовку контура (EXECUTE)

С помощью `EXECUTE` подготовка контура отключается и одновременно происходит возврат в обычный режим обработки.

Синтаксис

`EXECUTE` (<ОШИБКА>)

Объяснение

EXECUTE:	Предопределенная процедура для завершения подготовки контура	
<ошибка>:	Переменная для квитирования ошибки	
	Тип:	INT
	Значение переменной показывает, был ли контур подготовлен без ошибок:	
	0	ошибка
	1	нет ошибок

Пример

Программный код

```
...
N30 CONTPRON(...)
N40 G1 X... Z...
...
N100 EXECUTE(...)
...
```

4.26 Внешнее программирование циклов

4.26.1 Технологические циклы

4.26.1.1 Введение

Содержание

Эта глава содержит описание циклов для технологий токарной обработки, фрезерования, шлифования.

Структура

Описание цикла построено следующим образом:

- **Синтаксис**
  Имя цикла и последовательность вызова передаваемых параметров
- **Параметры**
  Таблица для объяснения отдельных параметров

Описание параметров

В таблице указаны следующие данные для параметра: Имя, описание, диапазон значений и зависимости относительно других параметров.

Графа для ссылки на параметр в маске облегчает нахождение запрограммированных на СЧПУ значений при обратном переводе вызовов циклов, созданных на внешних устройствах.

Параметр "только для интерфейса"

В таблице параметры обозначены как "только для интерфейса". Они не имеют значения для функций цикла. Они служат только для возможности полного обратного перевода вызовов циклов. Если они не запрограммированы, то обратный перевод цикла все же возможен, в этом случае поля обозначены соответствующими цветами и должны быть заполнены в маске.

Параметры "зарезервировано"

Параметры, описанные с "зарезервировано", должны быть запрограммированы со значением 0 или с незначимой запятой, чтобы согласование следующих параметров вызова с внутренними параметрами цикла совпадало. Исключение: для строковых параметров Значение "" или незначимая запятая.

Повторение циклов на образце позиции

Сверлильные и фрезерные циклы могут повторяться на образцах позиций (модальные вызовы). В этом случае записать перед циклом в той же строке `MCALL`, к примеру, `MCALL CYCLE83(...)`.

Примечание

Если определенные передаваемые параметры (к примеру, `<_VARI>`, `<_GMODE>`, `<_DMODE>`, `<_AMODE>`) запрограммированы косвенно как параметры, маска ввода при обратном переводе открывается, но не может быть сохранена, т. к. нет однозначного согласования с определенными полями выбора.

4.26.1.2 Обзор по технологиям

В следующей обзорной таблице перечислены все доступные технологические циклы с возможностью внешнего программирования с разбивкой по соответствующим технологиям:

Технология	Технологический цикл
Сверление	• CYCLE81 - сверление, центрование (Страница 1111) • CYCLE82 - сверление, цекование (Страница 1112) • CYCLE85 - развертывание (Страница 1120) • CYCLE86 - растачивание (Страница 1121) • CYCLE83 – Глубокое сверление 1 (Страница 1114) • CYCLE830 - Глубокое сверление 2 (Страница 1149) • CYCLE84 - нарезание внутренней резьбы без компенсирующего патрона (Страница 1117) • CYCLE840 - Нарезание внутренней резьбы с компенсирующим патроном (Страница 1160) • CYCLE78 - Сверлильное резьбофрезерование (Страница 1105) • CYCLE802 - любые позиции (Страница 1145) • HOLES1 – Образец позиции "Линия" (Страница 1070) • CYCLE801 – Образец позиции "Решетка или рамка" (Страница 1144) • HOLES2 – Образец позиции "Окружность или неполная окружность" (Страница 1071)
Токарная обработка	• CYCLE951 - обработка резанием (Страница 1171) • CYCLE930 - выточка (Страница 1165) • CYCLE940 – Канавка формы E и F, резьбовая канавка (Страница 1168) • CYCLE99 - нарезание резьбы резцом (Страница 1130) • CYCLE98 - цепочка резьб (Страница 1126) • CYCLE92 - отрез (Страница 1123)
Контурная обточка	• CYCLE62 - Вызов контура (Страница 1090) • CYCLE952 – Обработка резаньем / Обработка резаньем – остаток / Прорез / Прорез – остаток / Токарная выточка / Токарная выточка – остаток (Страница 1174)

Технология	**Технологический цикл**
Фрезерная обработка	• CYCLE61 - плоское фрезерование (Страница 1088) • POCKET3 – Прямоугольный карман (Страница 1073) • POCKET4 – Круговой карман (Страница 1075) • CYCLE76 – Прямоугольная цапфа (Страница 1101) • CYCLE77 – Круговая цапфа (Страница 1103) • CYCLE79 - многогранник (Страница 1108) • SLOT1 - Продольный паз (Страница 1078) • SLOT2 - кольцевая канавка (Страница 1081) • CYCLE899 – открытый паз (Страница 1162) • LONGHOLE - Продольный паз (Страница 1083) • CYCLE70 - резьбофрезерование (Страница 1095) • CYCLE60 – Гравирование (Страница 1085)
Контурное фрезерование	• CYCLE62 - Вызов контура (Страница 1090) • CYCLE72 - фрезерование траектории (Страница 1097) • CYCLE63 – Фрезерование контурного кармана / Контурный карман – остаточный материал / Фрезерование контурной цапфы / Контурная цапфа – остаточный материал (Страница 1091) • CYCLE64 - Предварительное сверление кармана (Страница 1093)
Шлифование	• CYCLE495 - Профилирование (Страница 1135) • CYCLE435 - установка системы координат правящего инструмента (Страница 1135) • CYCLE4071 - Продольное шлифование с подачей в точке возврата (Страница 1180) • CYCLE4072 - Продольное шлифование с подачей в точке возврата и сигналом отмены (Страница 1182) • CYCLE4073 - Продольное шлифование с непрерывной подачей на глубину (Страница 1186) • CYCLE4074 - Продольное шлифование с непрерывной подачей на глубину и сигналом отмены (Страница 1187) • CYCLE4075 - Плоское шлифование с подачей в точке возврата (Страница 1190) • CYCLE4077 - Плоское шлифование с подачей в точке возврата и сигналом отмены (Страница 1193) • CYCLE4078 - Плоское шлифование с непрерывной подачей на глубину (Страница 1197) • CYCLE4079 - Плоское шлифование с прерывистой подачей на глубину (Страница 1199)

Технология	Технологический цикл
Прочее	• CYCLE782 – Коррекция по загрузке (Страница 1137) • CYCLE800 – Поворот плоскости, поворот инструмента, точная установка инструмента (Страница 1140) • CYCLE806 — интерполированное вращение (Страница 1148) • CYCLE832 - High Speed Settings (Страница 1156)
Все	• GROUP_BEGIN — Начало блока программы (Страница 1201) • GROUP_END — Конец блока программы (Страница 1202) • GROUP_ADDEND — конец дополнения ввода (Страница 1202)

4.26.1.3 HOLES1 – Образец позиции "Линия"

Синтаксис

```
HOLES1(<SPCA>, <SPCO>, <STA1>, <FDIS>, <DBH>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1	X0	<SPCA>	REAL	Исходная точка для ряда отверстий в 1-й оси (абс.)		
2	Y0	<SPCO>	REAL	Исходная точка для ряда отверстий в 2-й оси (абс.)		
3	α0	<STA1>	REAL	Базовый угол поворота (угол к 1-й оси)		
4	L0	<FDIS>	REAL	Расстояние от 1-го отверстия до исходной точки		
5	l	<DBH>	REAL	Расстояние между отверстиями		
6	n	<NUM>	INT	Число отверстий		
7		<_VARI>	INT	зарезервировано		
8		<_UMODE>	INT	зарезервировано		
9		<_HIDE>	STRING [200]	Пропущенные позиции • макс. 198 знаков • Указание текущего номера позиции, к примеру, "1,3" (позиции 1 и 3 не выполняются)		
10		<_NSP>	INT	зарезервировано		
11		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)

4.26.1.4 HOLES2 – Образец позиции "Окружность или неполная окружность"

Синтаксис

```
HOLES2(<CPA>, <CPO>, <RAD>, <STA1>, <INDA>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)
```

Параметр

№	Параметр в маске	Параметр внутр.	Тип данных	Объяснение			
1	X0	<CPA>	REAL	Центр для окружности отверстий в 1-й оси (абс.) Исходная точка в 1-й оси			(при XY) (при XA, YB, ZC)
2	Y0	<CPO>	REAL	Центр для окружности отверстий во 2-й оси (абс.) Исходная точка во 2-й оси			(при XY) (при XA, YB, ZC)
3	r	<RAD>	REAL	Радиус окружности отверстий			(при XY)
4	α0	<STA1>	REAL	Начальный угол или 1-я позиция круговой оси			(при XY) (при XA, YB, ZC)
5	α1	<INDA>	REAL	Угол повторного включения (только для делительной окружности)			(при XY, XA, YB, ZC)
					< 0 =	по часовой стрелке	
					> 0 =	против часовой стрелки	
6	n	<NUM>	INT	Число позиций			

№	Параметр в маске	Параметр внутр.	Тип данных	Объяснение			
7		<_VARI>	INT	Режим обработки			
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано		
				ВТОРАЯ ПОЗИЦИЯ:	тип позиционирования		
					0 =	подвод в позицию - линейный	
					1 =	подвод в позицию - по круговой траектории	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано		
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	Образец окружности		
					0 =	режим совместимости, если INDA = 0 тогда полный круг, INDA <> 0 тогда делительная окружность)	
					1 =	Полный круг	
					2 =	Делительная окружность	
				ПЯТАЯ ПОЗИЦИЯ:	образец позиции с круговой осью		
					0 =	XY (без круговой оси)	(при XY)
					1 =	XA (ось X и круговая ось вокруг X)	(только при XA)
					2 =	YB (ось Y и круговая ось вокруг Y)	(только при YB)
					3 =	ZC (ось Z и круговая ось вокруг C)	(только при ZC)
				СЕДЬМАЯ ПОЗИЦИЯ + ШЕСТАЯ ПОЗИЦИЯ:	Смещение (при нескольких круговых осях вокруг одной и той же оси; если индекс слишком велик, то 1-й оси)		
					00 =	1. Ось A, B или C	
					01 =	2. Ось A, B или C	
					...		
					10 =	20. Ось A, B или C	
8		<_UMODE>	INT	зарезервировано			
9		<_HIDE>	STRING [200]	зарезервировано			
10		<_NSP>	INT	зарезервировано			
11		<_DMODE>	INT	Режим индикации			
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19		
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость	
					1 =	G17 (активна только в цикле)	
					2 =	G18 (активна только в цикле)	
					3 =	G19 (активна только в цикле)	

4.26.1.5 POCKET3 – Прямоугольный карман

Синтаксис

```
POCKET3(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_LENG>, <_WID>, <_CRAD>,
<_PA>, <_PO>, <_STA>, <_MID>, <_FAL>, <_FALD>, <_FFP1>, <_FFD>,
<_CDIR>, <_VARI>, <_MIDA>, <_AP1>, <_AP2>, <_AD>, <_RAD1>, <_DP1>,
<_UMODE>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1	RP	<_RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)		
3	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<_DP>	REAL	Глубина кармана (абс./инкр.), см. <_AMODE>)		
5	I	<_LENG>	REAL	Длина кармана (инкр., ввод со знаком)		
6	W	<_WID>	REAL	Ширина кармана (инкр., ввод со знаком)		
7	R	<_CRAD>	REAL	Радиус угла кармана		
8	X0	<_PA>	REAL	Исходная точка, 1-я ось (абс.)		
9	YO	<_PO>	REAL	Исходная точка, 2-я ось (абс.)		
10	α0	<_STA>	REAL	Угол поворота, угол между продольной осью (L) и 1-й осью		
11	DZ	<_MID>	REAL	Макс. подача на глубину		
12	UXY	<_FAL>	REAL	Чистовой припуск плоскость		
13	UZ	<_FALD>	REAL	Чистовой припуск глубина		
14	F	<_FFP1>	REAL	Подача в плоскости		
15	FZ	<_FFD>	REAL	Подача на глубину		
16		<_CDIR>	INT	Направление фрезерования:	0 =	Синхронный ход
					1 =	Противоход
17		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:		
					1 =	Черновая обработка
					2 =	Чистовая обработка
					4 =	Чистовая обработка края
					5 =	Снятие фаски
				ВТОРАЯ ПОЗИЦИЯ:		
					0 =	предварительное сверление, подача с G0
					1 =	вертикально, подача с G1
					2 =	по спирали
					3 =	качанием на продольной оси кармана
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
18	DXY	<_MIDA>	REAL	Макс. подача в плоскости, единица см. <_AMODE>		
19	L1	<_AP1>	REAL	Длина предварительной обработки (инкр.)		
20	W1	<_AP2>	REAL	Ширина предварительной обработки (инкр.)		
21	AZ	<_AD>	REAL	Глубина предварительной обработки (инкр.)		
22	ER	<_RAD1>	REAL	Радиус спиральной траектории при врезании по спирали		
	EW			Макс. угол врезания для качания		
23	EP	<_DP1>	REAL	Шаг спирали при врезании по спирали		
24		<_UMODE>	INT	зарезервировано		
25	FS	<_FS>	REAL	Ширина фаски (инкр.)		
26	ZFS	<_ZFS>	REAL	Глубина врезания (острие инструмента) при снятии фаски (абс./инкр.) см. <_AMODE>		
27		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки/только расчет стартовой точки	
					0 =	Режим совместимости
					1 =	обычная обработка
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	указание размеров через центр/угол	
					0 =	Режим совместимости
					1 =	указание размеров через центр
					2 =	указание размеров угловая точка, длина кармана +LENG/+WID
					3 =	указание размеров угловая точка, длина кармана -LENG/+WID
					4 =	указание размеров угловая точка, длина кармана +LENG/-WID
					5 =	указание размеров угловая точка, длина кармана -LENG/-WID
				ПЯТАЯ ПОЗИЦИЯ:	полная обработка/доводка	
					0 =	режим совместимости (<_AP1>, <_AP2> и <_AD> обрабатывать как прежде)
					1 =	Комплексная обработка
					2 =	Доработка

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
28		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	тип подачи: G-группа (G94/G95) для подачи в плоскости и на глубину	
					0 =	Режим совместимости
					1 =	G-команда как перед вызовом цикла. G94/G95 для подачи в плоскости и на глубину одновременно
				ТРЕТЬЯ ПОЗИЦИЯ:	---	зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой
29		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина кармана (Z1)	
					0 =	абсолютная (режим совместимости)
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	единица для подачи в плоскости (DXY)	
					0 =	мм
					1 =	% от диаметра инструмента
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски (ZFS)	
					0 =	абсолютная
					1 =	инкрементальная

4.26.1.6 POCKET4 – Круговой карман

Синтаксис

```
POCKET4(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_CDIAM>, <_PA>, <_PO>,
<_MID>, <_FAL>, <_FALD>, <_FFP1>, <_FFD>, <_CDIR>, <_VARI>, <_MIDA>,
<_AP1>, <_AD>, <_RAD1>, <_DP1>, <_UMODE>, <_FS>, <_ZFS>, <_GMODE>,
<_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1	RP	<_RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)		
3	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<_DP>	REAL	Глубина кармана (абс./инкр.), см. <_AMODE>		
5	⌀	<_CDIAM>	REAL	Диаметр кармана или радиус кармана, см. <_DMODE>		
6	X0	<_PA>	REAL	Исходная точка 1-й оси (абс.)		
7	Y0	<_PO>	REAL	Исходная точка 2-й оси (абс.)		
8	DZ	<_MID>	REAL	Макс. подача на глубину, см. <_VARI>= в плоскости Макс. шаг спирали, см. <_VARI> = по спирали		
9	UXY	<_FAL>	REAL	Чистовой припуск плоскость		
10	UZ	<_FALD>	REAL	Чистовой припуск глубина		
11	F	<_FFP1>	REAL	Подача для обработки поверхностей		
12	FZ	<_FFD>	REAL	Подача на глубину		
13		<_CDIR>	INT	Направление фрезерования	0 =	Синхронный ход
					1 =	Противоход
14		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	Обработка	
					1 =	Черновая обработка
					2 =	чистовая обработка
					4 =	Чистовая обработка края
					5 =	Снятие фаски
				ВТОРАЯ ПОЗИЦИЯ:	тип подачи (черновая и чистовая обработка)	
					0 =	предварительное сверление, подача с G0 (карман предварительно обработан)
					1 =	вертикально, подача с G1
					2 =	по спирали
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:		
					0 =	в плоскости
					1 =	по спирали
15	DXY	<_MIDA>	REAL	Макс. подача в плоскости, см. <_AMODE>, 0 = 0,8 x диаметр инструмента		
16	⌀	<_AP1>	REAL	Диаметр/радиус предварительной обработки (инкр.)		
17	AZ	<_AD>	REAL	Глубина предварительной обработки (инкр.)		
18	ER	<_RAD1>	REAL	Радиус спиральной траектории при врезании по спирали		
19	EP	<_DP1>	REAL	Шаг спирали при врезании по спиральной траектории		
20		<_UMODE>	INT	зарезервировано		
21	FS	<_FS>	REAL	Ширина фаски (инкр.)		

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
22	ZFS	<_ZFS>	REAL	Глубина врезания (острие инструмента) при снятии фаски (абс./инкр.) см. <_AMODE>		
23		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	обработка/расчет стартовой точки	
					0 =	Режим совместимости
					1 =	обычная обработка
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	зарезервировано	
				ПЯТАЯ ПОЗИЦИЯ:	полная обработка/доводка	
					0 =	режим совместимости (<_AP1> и <_AD> обрабатывать как прежде)
					1 =	Комплексная обработка
					2 =	Доработка
24		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	тип подачи: G-группа (G94/G95) для подачи в плоскости и на глубину	
					0 =	Режим совместимости
					1 =	G-команда как перед вызовом цикла. G94/G95 для подачи в плоскости и на глубину одновременно
				ТРЕТЬЯ ПОЗИЦИЯ:		
					0 =	режим совместимости (<_CDIAM>/<_AP1> вводится как радиус)
					1 =	<_CDIAM>/<_AP1> вводится как диаметр
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
25		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина кармана (Z1)	
					0 =	абсолютная (режим совместимости)
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	единица для ширины подачи (DXY)	
					0 =	мм
					1 =	% от диаметра инструмента
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски (ZFS)	
					0 =	абсолютная
					1 =	инкрементальная

4.26.1.7 SLOT1 - Продольный паз

Синтаксис

```
SLOT1 (<RTP>, <RFP>, <SDIS>, <_DP>, <_DPR>, <NUM>, <LENG>, <WID>,
<_CPA>, <_CPO>, <RAD>, <STA1>, <INDA>, <FFD>, <FFP1>, <_MID>,
<CDIR>, <_FAL>, <VARI>, <_MIDF>, <FFP2>, <SSF>, <_FALD>, <_STA2>,
<_DP1>, <_UMODE>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	RP	<RTP>	REAL	Плоскость отвода (абс.)
2	Z0	<RFP>	REAL	Исходная точка оси инструмента (абс.)
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)
4	Z1	<_DP>	REAL	Глубина паза (абс.)
5		<_DPR>	REAL	Глубина паза (инкр.), относительно Z0 (вводится без знака)
6		<NUM>	INT	Число пазов = 1
7	l	<LENG>	REAL	Длина паза
8	W	<WID>	REAL	Ширина паза
9	X0	<_CPA>	REAL	Исходная точка, 1-я ось плоскости
10	Y0	<_CPO>	REAL	Исходная точка, 2-я ось плоскости
11		<RAD>	REAL	зарезервировано
12	α	<STA1>	REAL	Угол поворота
13		<INDA>	REAL	зарезервировано
14	FZ	<FFD>	REAL	Подача на глубину

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
15	F	<FFP1>	REAL	Подача		
16	DZ	<_MID>	REAL	Макс. подача на глубину		
17		<CDIR>	INT	Направление фрезерования	0 =	Синхронный ход
					1 =	Противоход
18	UXY	<_FAL>	REAL	Чистовой припуск Плоскость или край паза		
19		<VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:		
					0 =	зарезервировано
					1 =	Черновая обработка
					2 =	Чистовая обработка
					4 =	чистовая обработка края (только обработка края)
					5 =	Снятие фаски
				ВТОРАЯ ПОЗИЦИЯ:	подвод	
					0 =	предварительное сверление, подача с G0 (паз подготовлен)
					1 =	вертикально, подача с G1
					2 =	по спирали
					3 =	качанием
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
20	DZF	<_MIDF>	REAL	зарезервировано		
21	FF	<FFP2>	REAL	зарезервировано		
22	SF	<SSF>	REAL	зарезервировано		
23	UZ	<_FALD>	REAL	Чистовой припуск глубина		
24	ER	<_STA2>	REAL	Радиус спиральной траектории при врезании по спирали		
	EW			Макс. угол врезания для качания		
25	EP	<_DP1>	REAL	Глубина врезания на оборот для спирали		
26		<_UMODE>	INT	зарезервировано		
27	FS	<_FS>	REAL	Ширина фаски (инкр.) при снятии фаски		
28	ZFS	<_ZFS>	REAL	Глубина врезания (острие инструмента) при снятии фаски (абс./инкр.) см. <_AMODE>		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
29		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки или только расчет стартовой точки	
					1 =	обычная обработка
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	указание размеров исходной точки, положения паза	
					0 =	центр
					1 =	слева внутри +L
					2 =	справа внутри -L
					3 =	левый край +L
					4 =	правый край -L
30		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	идентификатор версии ПО	
					1 =	расширение функций SLOT1
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой
31		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная глубина Z1 (абс./инкр.)	
					0 =	Совместимость
					1 =	Z1 (инкр.)
					2 =	Z1 (абс.)
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски ZFS	
					0 =	ZFS (абс.)
					1 =	ZFS (инкр.)

Примечание

Цикл, по сравнению с предшествующими версиями ПО, получил новые функции. Следствием этого явилось то, что определенные параметры более не отображаются в маске ввода (<NUM>, <RAD>, <INDA>). Несколько пазов на одном образце позиций могут быть запрограммированы посредством "MCALL" и вызова требуемого образца позиций, к примеру, HOLES2.

4.26.1.8 SLOT2 - кольцевая канавка

Синтаксис

```
SLOT2(<RTP>, <RFP>, <SDIS>, <_DP>, <_DPR>, <NUM>, <AFSL>, <WID>,
<_CPA>, <_CPO>, <RAD>, <STA1>, <INDA>, <FFD>, <FFP1>, <_MID>,
<CDIR>, <_FAL>, <VARI>, <_MIDF>, <FFP2>, <SSF>, <_FFCP>, <_UMODE>,
<_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	RP	<RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<RFP>	REAL	Исходная точка оси инструмента (абс.)		
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<_DP>	REAL	Глубина паза (абс.)		
5		<_DPR>	REAL	Глубина паза (инкр.), относительно Z0 (вводится без знака)		
6	n	<NUM>	INT	Количество канавок		
7	α1	<AFSL>	REAL	Апертурный угол канавки		
8	W	<WID>	REAL	Ширина паза		
9	X0	<_CPA>	REAL	Исходная точка = центр окружности, 1-я ось плоскости		
10	Y0	<_CPO>	REAL	Исходная точка = центр окружности, 2-я ось плоскости		
11	r	<RAD>	REAL	Радиус окружности		
12	α0	<STA1>	REAL	Начальный угол		
13	α2	<INDA>	REAL	Угол повторного включения		
14	FZ	<FFD>	REAL	Подача на глубину		
15	F	<FFP1>	REAL	Подача		
16	DZ	<_MID>	REAL	Макс. подача на глубину		
17		<CDIR>	INT	Направление фрезерования	0 =	Синхронный ход
					1 =	Противоход
18	UXY	<_FAL>	REAL	Чистовой припуск Плоскость или край паза		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
19		<VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:		
					0 =	Комплексная обработка
					1 =	Черновая обработка
					2 =	Чистовая обработка
					3 =	Чистовая обработка края
					5 =	Снятие фаски
				ВТОРАЯ ПОЗИЦИЯ:		
					0 =	промежуточное позиционирование с G0-прямой
					1 =	промежуточное позиционирование по круговой траектории
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:		
					0 =	режим совместимости, если <INDA> = 0, то полный круг, <INDA><> 0, то часть круга)
					1 =	Полный круг
					2 =	Делительная окружность
20	DZF	<_MIDF>	REAL	зарезервировано		
21		<FFP2>	REAL	зарезервировано		
22		<SSF>	REAL	зарезервировано		
23	FF	<_FFCP>	REAL	зарезервировано		
24		<_UMODE>	INT	зарезервировано		
25	FS	<_FS>	REAL	Ширина фаски (инкр.)		
26	ZFS	<_ZFS>	REAL	Глубина врезания (острие инструмента) при снятии фаски (абс./инкр.) см. <_AMODE>		
27		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки или только расчет стартовой точки	
					0 =	Режим совместимости
					1 =	обычная обработка

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
28		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	идентификатор версии ПО	
					1 =	функции SLOT2 от ПО 2.5
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой
29		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная глубина Z1 (абс./инкр.)	
					0 =	Совместимость
					1 =	Z1 (инкр.)
					2 =	Z1 (абс.)
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски ZFS	
					0 =	ZFS (абс.)
					1 =	ZFS (инкр.)

4.26.1.9 LONGHOLE - Продольный паз

Синтаксис

```
LONGHOLE(<RTP>, <RFP>, <SDIS>, <_DP>, <_DPR>, <NUM>, <LENG>, <_CPA>,
<_CPO>, <RAD>, <STA1>, <INDA>, <FFD>, <FFP1>, <MID>, <_VARI>,
<_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение
1	RP	<RTP>	REAL	Плоскость отвода (абс.)
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<_DP>	REAL	Глубина продольного паза (абс.)		
5		<_DPR>	REAL	Глубина продольного паза (инкр.), относительно Z0 (вводится без знака)		
6		<NUM>	INT	Число продольных пазов = 1		
7	l	<LENG>	REAL	Длина продольного паза		
8	X0	<_CPA>	REAL	Исходная точка, 1-я ось плоскости		
9	Y0	<_CPO>	REAL	Исходная точка, 2-я ось плоскости		
10		<RAD>	REAL	зарезервировано		
11	α0	<STA1>	REAL	Угол поворота		
12		<INDA>	REAL	зарезервировано		
13	FZ	<FFD>	REAL	Подача на глубину		
14	F	<FFP1>	REAL	Подача		
15	DZ	<MID>	REAL	Макс. подача на глубину		
16		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	тип подачи	
					1 =	вертикально с G1
					3 =	качанием
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
17		<_UMODE>	INT	зарезервировано		
18		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки или только расчет стартовой точки	
					0 =	режим совместимости
					1 =	обычная обработка
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	указание размеров исходной точки, положения паза	
					0 =	центр
					1 =	слева внутри +L
					2 =	справа внутри -L
					3 =	левый край +L
					4 =	правый край -L

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
19		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	тип подачи: G-группа (G94/G95) для подачи в плоскости и на глубину	
					0 =	режим совместимости
					1 =	G-команда как перед вызовом цикла. G94/G95 для подачи в плоскости и на глубину одновременно
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	идентификатор версии ПО	
					1 =	расширение функций LONGHOLE (указание размеров исходной точки)
20		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная глубина Z1 (абс./инкр.)	
					0 =	Совместимость
					1 =	Z1 (инкр.)
					2 =	Z1 (абс.)

Примечание

Цикл, по сравнению с предшествующими версиями ПО, получил новые функции. Следствием этого явилось то, что определенные параметры более не отображаются в маске ввода (<NUM>, <RAD>, <INDA>). Несколько пазов на одном образце позиций могут быть запрограммированы посредством "MCALL" и вызова требуемого образца позиций, например, HOLES2.

4.26.1.10 CYCLE60 – Гравирование

Синтаксис

```
CYCLE60(<_TEXT>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_PA>,
<_PO>, <_STA>, <_CP1>, <_CP2>, <_WID>, <_DF>, <_FFD>, <_FFP1>,
<_VARI>, <_CODEP>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение
1		<_TEXT>	STRING [200]	Гравируемый текст (макс. 100 символов)
2	RP	<_RTP>	REAL	Плоскость отвода (абс.)
3	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)
4	sc	<_SDIS>	REAL	Безопасное расстояние (аддитивно к базовой плоскости, вводится без знака)
5	Z1	<_DP>	REAL	Глубина (абс.), см. <_AMODE>
6	Z1	<_DPR>	REAL	Глубина (инкр.), см. <_AMODE>
7	X0	<_PA>	REAL	Исходная точка, 1-я ось плоскости (абс.) - прямоуг., см. <_VARI>
	R			Исходная точка, длина (радиус) - поляр., см. <_VARI>
8	Y0	<_PO>	REAL	Исходная точка, 2-я ось плоскости (абс.) - прямоуг., см. <_VARI>
	α0			Исходная точка, угол относительно 1-ой оси — полярн., см. <_VARI>
9	α1	<_STA>	REAL	Направление текста, угол текстовой линии относительно 1-ой оси), см. <_VARI>
10	XM	<_CP1>	REAL	Центр текстовой окружности, 1-я ось плоскости (абс.) - прямоуг., см. <_VARI>
	LM			Центр текстовой окружности, длина (радиус) относительно WNP - поляр., см. <_VARI>
11	YM	<_CP2>	REAL	Центр текстовой окружности, 2-я ось плоскости (абс.) - прямоуг., см. <_VARI>
	αM			Центр текстовой окружности, угол относительно 1-ой оси — полярн., см. <_VARI>
12	W	<_WID>	REAL	Высота символов (вводится без знака)
13	DX1 DX2	<_DF>	REAL	Интервал между символами / общая ширина, см. <_VARI>
	α2			Апертурный угол, см. <_VARI>
14	FZ	<_FFD>	REAL	Подача на глубину, см. <_DMODE>
15	F	<_FFP1>	REAL	Подача для обработки поверхностей

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
16		<_VARI>	INT	Обработка (ориентация и исходная точка гравируемого текста)		
				ПЕРВАЯ ПОЗИЦИЯ:	исходная точка	
					0 =	прямоуг.
					1 =	поляр.
				ВТОРАЯ ПОЗИЦИЯ:	ориентация текста	
					0 =	текст на одной линии
					1 =	текст на дуге окружности сверху
					2 =	текст на дуге окружности снизу
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	исходная точка текста горизонтально	
					0 =	слева
					1 =	по центру
					2 =	справа
				ПЯТАЯ ПОЗИЦИЯ:	исходная точка текста вертикально	
					0 =	внизу
					1 =	по центру
					2 =	сверху
				ШЕСТАЯ ПОЗИЦИЯ:	длина текста	
					0 =	интервал между символами
					1 =	общая ширина текста (только для линейного текста)
					2 =	апертурный угол (только для текста на дуге окружности)
				СЕДЬМАЯ ПОЗИЦИЯ:	центр окружности	
					0 =	прямоугольная (декартова)
					1 =	полярная
				ВОСЬМАЯ ПОЗИЦИЯ:	зеркальный шрифт	
					0 =	Совместимость
					1 =	зеркальный шрифт ВКЛ
					2 =	зеркальный шрифт ВЫКЛ
17		<_CODEP>	INT	Номер кодовой страницы для шрифта (в наст. время только 1252)		
18		<_UMODE>	INT	зарезервировано		
19		_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки/только расчет стартовой точки	
					0 =	Режим совместимости
					1 =	обычная обработка

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
20		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	тип подачи: G-группа (G94/G95) для подачи в плоскости и на глубину	
					0 =	Режим совместимости
					1 =	G-команда как перед вызовом цикла. G94/G95 для подачи в плоскости и на глубину одновременно
21		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная глубина (<_DP>, <_DPR>)	
					0 =	Совместимость
					1 =	инкрементальная (<_DPR>)
					2 =	абсолютная (<_DP>)

4.26.1.11 CYCLE61 - плоское фрезерование

Синтаксис

```
CYCLE61(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_PA>, <_PO>, <_LENG>,
<_WID>, <_MID>, <_MIDA>, <_FALD>, <_FFP1>, <_VARI>, <_LIM>,
<_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	RP	<_RTP>	REAL	Плоскость отвода (абс.)
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента, высота заготовки (абс.)
3	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)
4	Z1	<_DP>	REAL	Высота готовой детали (абс./инкр.), см. <_AMODE>
5	X0	<_PA>	REAL	Угловая точка 1 в 1-й оси (абс.)
6	Y0	<_PO>	REAL	Угловая точка 1 во 2-й оси (абс.)
7	X1	<_LENG>	REAL	Угловая точка 2 в 1-й оси (абс./инкр.), см. <_AMODE>
8	Y1	<_WID>	REAL	Угловая точка 2 во 2-й оси (абс./инкр.), см. <_AMODE>

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
9	DZ	<_MID>	REAL	Макс. подача на глубину		
10	DXY	<_MIDA>	REAL	Макс. подача в плоскости (единица, см. <_AMODE>)		
11	UZ	<_FALD>	REAL	Чистовой припуск глубина		
12	F	<_FFP1>	REAL	Подача обработки		
13		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	обработка	
					1 =	Черновая обработка
					2 =	Чистовая обработка
				ВТОРАЯ ПОЗИЦИЯ:	направление обработки	
					1 =	параллельно 1-й оси, одно направление
					2 =	параллельно 2-й оси, одно направление
					3 =	параллельно 1-й оси, разное направление
					4 =	параллельно 2-й оси, разное направление
14		<_LIM>	INT	Ограничения		
				ПЕРВАЯ ПОЗИЦИЯ:	ограничение 1-й оси минус	
					0 =	нет
					1 =	да
				ВТОРАЯ ПОЗИЦИЯ:	ограничение 1-й оси плюс	
					0 =	нет
					1 =	да
				ТРЕТЬЯ ПОЗИЦИЯ:	ограничение 2-й оси минус	
					0 =	нет
					1 =	да
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	ограничение 2-й оси плюс	
					0 =	нет
					1 =	да
15		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
16		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная глубина (<_DP>)	
					0 =	абсолютная
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	единица для подачи в плоскости (<_MIDA>)	
					0 =	мм
					1 =	% от диаметра инструмента
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	длина плоскости	
					0 =	инкрементальная
					1 =	абсолютная
				ПЯТАЯ ПОЗИЦИЯ:	ширина плоскости	
					0 =	инкрементальная
					1 =	абсолютная

4.26.1.12 CYCLE62 - Вызов контура

Синтаксис

```
CYCLE62(<_KNAME>,<_TYPE>, <_LAB1>, <_LAB2>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	PRG/CON	<_KNAME>	STRING [140]	Имя контура или имя подпрограммы, программирования не требуется при _TYPE = 2		
2		<_TYPE>	INT	Определение ввода контура		
					0 =	Подпрограмма
					1 =	Имя контура
					2 =	Метки
					3 =	Метки в подпрограмме
3	LAB1	<_LAB1>	STRING[32]	Метка 1, начало контура		
4	LAB2	<_LAB2>	STRING[32]	Метка 2, конец контура		

4.26.1.13 CYCLE63 – Фрезерование контурного кармана / Контурный карман – остаточный материал / Фрезерование контурной цапфы / Контурная цапфа – остаточный материал

Синтаксис

```
CYCLE63(<_PRG>, <_VARI>, <_RP>, <_Z0>, <_SC>, <_Z1>, <_F>, <_FZ>,
<_DXY>, <_DZ>, <_UXY>, <_UZ>, <_CDIR>, <_XS>, <_YS>, <_ER>, <_EP>,
<_EW>, <_FS>, <_ZFS>, <_TR>, <_DR>, <_UMODE>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1	PRG	<_PRG>	STRING [100]	Имя программы выборки		
2		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	технологическая обработка	
					1 =	Черновая обработка
					3 =	Чистовая обработка основания
					4 =	Чистовая обработка края
					5 =	Снятие фаски
				ВТОРАЯ ПОЗИЦИЯ:	тип подачи	
					0 =	Врезание по центру
					1 =	Спиральное врезание
					2 =	Маятниковое врезание
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	режим подъема	
					0 =	подъем на плоскость отвода
					1 =	подъем на исходную точку + безопасное расстояние
				ПЯТАЯ ПОЗИЦИЯ:	стартовая точка при черновой и чистовой обработке основания	
					0 =	автоматически
					1 =	вручную
3	RP	<_RP>	REAL	Плоскость отвода (абс.)		
4	Z0	<_Z0>	REAL	Исходная точка оси инструмента (абс.)		
5	sc	<_SC>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
6	Z1	<_Z1>	REAL	Конечная глубина, (см. <_AMODE> ПЕРВАЯ ПОЗИЦИЯ)		
7	F	<_F>	REAL	Подача в плоскости черновая/чистовая обработка		
8	FZ	<_FZ>	REAL	Подача на глубину		
9	DXY	<_DXY>	REAL	Подача в плоскости - единица, (см. <_AMODE> ВТОРАЯ ПОЗИЦИЯ)		
10	DZ	<_DZ>	REAL	Подача на глубину		

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
11	UXY	<_UXY>	REAL	Чистовой припуск плоскость		
12	UZ	<_UZ>	REAL	Чистовой припуск глубина		
13		<_CDIR>	INT	Направление фрезерования	0 =	Синхронный ход
					1 =	Противоход
14	XS	<_XS>	REAL	Стартовая точка X, абсолютная		
15	YS	<_YS>	REAL	Стартовая точка Y, абсолютная		
16	ER	<_ER>	REAL	Врезание по спирали: Радиус		
17	EP	<_EP>	REAL	Врезание по спирали: Шаг		
18	EW	<_EW>	REAL	Маятниковое врезание: Макс. угол врезания		
19	FS	<_FS>	REAL	Ширина фаски (инкр.) при снятии фаски		
20	ZFS	<_ZFS>	REAL	Глубина врезания острия инструмента при снятии фаски, (см. <_AMODE>ТРЕТЬЯ ПОЗИЦИЯ)		
21	tr	<_TR>	STRING[32]	Имя эталонного инструмента при обработке оставшегося материала		
22	DR	<_DR>	INT	D-номер эталонного инструмента при обработке оставшегося материала		
23		<_UMODE>	INT	зарезервировано		
24		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки/только расчет стартовой точки	
					0 =	обычная обработка (режима совместимости не требуется)
					1 =	обычная обработка
					2 =	зарезервировано

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
25		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	технологический режим	
					1 =	карман
					2 =	цапфа
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	обработка остаточного материала	
					0 =	нет
					1 =	да
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой
				ШЕСТАЯ ПОЗИЦИЯ:	Автоматическое имя программы	
					0 =	нет
					1 =	да
26		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная глубина (Z1)	
					0 =	абсолютная (режим совместимости)
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	единица для подачи в плоскости (DXY)	
					0 =	мм
					1 =	% от диаметра инструмента
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски (ZFS)	
					0 =	абсолютная
					1 =	инкрементальная
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано

4.26.1.14 CYCLE64 - Предварительное сверление кармана

Синтаксис

```
CYCLE64(<_PRG>, <_VARI>, <_RP>, <_Z0>, <_SC>, <_Z1>, <_F>, <_DXY>,
<_UXY>, <_UZ>, <_CDIR>, <_TR>, <_DR>, <_UMODE>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1	PRG	<_PRG>	STRING [100]	Имя программы сверления/центрования		
2		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	режим подъема	
					0 =	подъем на плоскость отвода
					1 =	подъем на исходную точку + безопасное расстояние
3	RP	<_RP>	REAL	Плоскость отвода (абс.)		
4	Z0	<_Z0>	REAL	Исходная точка (абс.)		
5	sc	<_SC>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
6	Z1	<_Z1>	REAL	Глубина сверления/центрования, (см. <_AMODE>ПЕРВАЯ ПОЗИЦИЯ)		
7	F	<_F>	REAL	Подача сверления/центрования		
8	DXY	<_DXY>	REAL	Подача в плоскости - единица, (см. <_AMODE>ВТОРАЯ ПОЗИЦИЯ)		
9	UXY	<_UXY>	REAL	Чистовой припуск плоскость		
10	UZ	<_UZ>	REAL	Чистовой припуск глубина		
11		<_CDIR>	INT	Направление фрезерования	0 =	Синхронный ход
					1 =	Противоход
12	tr	<_TR>	STRING[20]	Имя эталонного инструмента		
13	DR	<_DR>	INT	D-номер эталонного инструмента		
14		<_UMODE>	INT	зарезервировано		
15		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки/только расчет стартовой точки	
					0 =	обычная обработка (режима совместимости не требуется)
					1 =	обычная обработка
					2 =	зарезервировано

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
25		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	технологический режим	
					1 =	Предварительное сверление
					2 =	Центрование
				ТРЕТЬЯ ПОЗИЦИЯ:	---	зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ШЕСТАЯ ПОЗИЦИЯ:	Автоматическое имя программы	
					0 =	нет
					1 =	да
26		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления/центрования Z1	
					0 =	абсолютная (режим совместимости)
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	единица для подачи в плоскости (DXY)	
					0 =	мм
					1 =	% от диаметра инструмента

4.26.1.15 CYCLE70 - резьбофрезерование

Синтаксис

```
CYCLE70(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DIATH>, <_H1>, <_FAL>,
<_PIT>, <_NT>, <_MID>, <_FFR>, <_TYPTH>, <_PA>, <_PO>, <_NSP>,
<_VARI>, <_PITA>, <_PITM>, <_PTAB>, <_PTABA>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Параметры

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	RP	<_RTP>	REAL	Плоскость отвода (абс.)
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
3	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<_DP>	REAL	Длина резьбы (абс., инкр.), см. <_AMODE> Учитывать выход на основании отверстия (минимум половина шага)		
5	∅	<_DIATH>	REAL	Ном. диаметр резьбы		
6	H1	<_H1>	REAL	Глубина резьбы		
7	U	<_FAL>	REAL	Чистовой припуск		
8	P	<_PIT>	REAL	Шаг резьбы (выбор <_PITA>: мм, дюйм, MODUL, витков/дюйм)		
9	NT	<_NT>	INT	Число зубьев на режущей пластине Длина инструмента всегда относится к нижнему зубу!		
10	DXY	<_MID>	REAL	Макс. подача на проход резца <_MID> > <_H1>: все за один проход резца		
11	F	<_FFR>	REAL	Подача фрезерования		
12		<_TYPTH>	INT	Тип резьбы	0 =	внутренняя резьба
					1 =	наружная резьба
13	X0	<_PA>	REAL	Центр окружности, 1-я ось (абс.)		
14	Y0	<_PO>	REAL	Центр окружности, 2-я ось (абс.)		
15	αS	<_NSP>	REAL	Начальный угол (многозаходная резьба)		
16		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:		
					1 =	черновая обработка
					2 =	чистовая обработка
				ВТОРАЯ ПОЗИЦИЯ:		
					1 =	сверху вниз
					2 =	снизу вверх
				ТРЕТЬЯ ПОЗИЦИЯ:		
					0 =	правая резьба
					1 =	левая резьба
17		<_PITA>	INT	Нормирование шага резьбы		
					0 =	режим совместимости
					1 =	Шаг в мм
					2 =	шаг в витках на дюйм (TPI)
					3 =	шаг в дюймах
					4 =	шаг как MODUL
18		<_PITM>	STRING[15]	Строка как меркер для ввода шага резьбы (только для интерфейса)		
19		<_PTAB>	STRING[20]	Строка для таблицы резьб ("", "ISO", "BSW", "BSP", "UNC") (только для интерфейса)		
20		<_PTABA>	STRING[20]	Строка для выбора в таблице резьб (к примеру, "M 10", "M 12", ...) (только для интерфейса)		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
21		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	обработка/расчет стартовой точки	
					0 =	режим совместимости
					1 =	обычная обработка
22		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
23		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	Длина резьбы (<_DP>)	
					0 =	абсолютная
					1 =	инкрементальная

4.26.1.16 CYCLE72 - фрезерование траектории

Синтаксис

```
CYCLE72(<_KNAME>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_MID>, <_FAL>,
<_FALD>, <_FFP1>, <_FFD>, <_VARI>, <_RL>, <_AS1>, <_LP1>, <_FF3>,
<_AS2>, <_LP2>,<_UMODE>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение
1		<_KNAME>	STRING [141]	Имя подпрограммы контура
2	RP	<_RTP>	REAL	Плоскость отвода (абс.)
3	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)
4	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)
5	Z1	<_DP>	REAL	Конечная тока, конечная глубина (абс./инкр.), см. <_AMODE>
6	DZ	<_MID>	REAL	Макс. подача на глубину (инкр., вводится без знака)
7	UXY	<_FAL>	REAL	Чистовой припуск плоскость (инкр.), припуск на контуре края

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
8	UZ	<_FALD>	REAL	Чистовой припуск глубина (инкр.), припуск на основании (вводится без знака)		
9	FX	<_FFP1>	REAL	Подача на конуре		
10	FZ	<_FFD>	REAL	Подача на глубину (или подача в пространстве)		
11		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	обработка	
					1 =	черновая обработка
					2 =	чистовая обработка
					5 =	снятие фаски
				ВТОРАЯ ПОЗИЦИЯ:		
					0 =	промежуточные пути с G0
					1 =	промежуточные пути с G1
				ТРЕТЬЯ ПОЗИЦИЯ:	отвод на конце контура	
					0 =	отвод на конце контура на исходную точку
					1 =	отвод на конце контура на исходную точку + <_SDIS>
					2 =	отвод на конце контура на <_SDIS>
					3 =	нет отвода на конце контура, подвод в следующую стартовую точку с подачей по контуру
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	зарезервировано	
				ПЯТАЯ ПОЗИЦИЯ:	Обработка контура	
					0 =	обработка контура вперед
					1 =	обработка контура назад Ограничения при назад: • макс. 170 элементов контура (вкл. фаски и закругления) • обрабатываются только значения в плоскости (X/Y) и F
12		<_RL>	INT	Направление обработки		
					40 =	по центру контура (G40, подвод и отвод: прямая или вертикально)
					41 =	слева от контура (G41, подвод и отвод: прямая или окружность)
					42 =	справа от контура (G42, подвод и отвод: прямая или окружность)

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
13		<_AS1>	INT	Движение подвода к контуру		
				ПЕРВАЯ ПОЗИЦИЯ:		
					1 =	прямая
					2 =	четверть круга
					3 =	полукруг
					4 =	вертикальный подвод и отвод
				ВТОРАЯ ПОЗИЦИЯ:		
					0 =	последнее движение, в плоскости
					1 =	последнее движение, в пространстве
14	L1	<_LP1>	REAL	Путь подвода, или радиус подвода (инкр., вводится без знака)		
15	FZ	<_FF3>	REAL	Подача для промежуточных путей (G94/G95 как на контуре)		
16		<_AS2>	INT	Движение отвода от контура (не для вертикального подвода/отвода)		
				ПЕРВАЯ ПОЗИЦИЯ:		
					1 =	прямая
					2 =	четверть круга
					3 =	полукруг
				ВТОРАЯ ПОЗИЦИЯ:		
					0 =	последнее движение, в плоскости
					1 =	последнее движение, в пространстве
17	L2	<_LP2>	REAL	Путь отвода, или радиус отвода (инкр., вводится без знака)		
18		<_UMODE>	INT	зарезервировано		
19	FS	<_FS>	REAL	Ширина фаски (инкр.)		
20	ZFS	<_ZFS>	REAL	Глубина врезания (острие инструмента) при снятии фаски (абс./инкр.) см. <_AMODE>		
21		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки/только расчет стартовой точки	
					0 =	режим совместимости
					1 =	обычная обработка

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
22		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	тип подачи: G-группа (G94/G95) для подачи в плоскости и на глубину	
					0 =	режим совместимости
					1 =	G-команда как перед вызовом цикла. G94/G95 для подачи в плоскости и на глубину одновременно
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:		
					0 =	режим совместимости: Имя контура стоит в <_KNAME>
					1 =	имя контура программируется в CYCLE62 и передается в _SC_CONT_NAME
23		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная точка Z1 (<_DP>)	
					0 =	абсолютная (режим совместимости)
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	единица для подачи в плоскости	
					0 =	мм, дюйм
					1 =	зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски (<_ZFS>)	
					0 =	абсолютная
					1 =	инкрементальная

Примечание

Если следующие передаваемые параметры запрограммированы косвенно (как параметры), то обратный перевод маски ввода не выполняется:

<_VARI>, <_RL>, <_AS1>, <_AS2>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>

4.26.1.17 CYCLE76 – Прямоугольная цапфа

Синтаксис

```
CYCLE76(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_LENG>, <_WID>,
<_CRAD>, <_PA>, <_PO>, <_STA>, <_MID>, <_FAL>, <_FALD>, <_FFP1>,
<_FFD>, <_CDIR>, <_VARI>, <_AP1>, <_AP2>, <_FS>, <_ZFS>, <_GMODE>,
<_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	RP	<_RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)		
3	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<_DP>	REAL	Глубина цапфы (абс.)		
5		<_DPR>	REAL	Глубина цапфы (инкр.), относительно Z0 (вводится без знака)		
6	l	<_LENG>	REAL	Ширина цапфы, см. <_GMODE>(вводится без знака)		
7	W	<_WID>	REAL	Ширина цапфы, см. <_GMODE>(вводится без знака)		
8	r	<_CRAD>	REAL	Радиус угла цапфы (вводится без знака)		
9	X0	<_PA>	REAL	Исходная точка цапфы, 1-я ось плоскости (абс.)		
10	Y0	<_PO>	REAL	Исходная точка цапфы, 2-я ось плоскости (абс.)		
11	α0	<_STA>	REAL	Угол поворота, угол между продольной осью (L) и 1-й осью плоскости		
12	DZ	<_MID>	REAL	Макс. подача на глубину (инкр., вводится без знака)		
13	UXY	<_FAL>	REAL	Чистовой припуск плоскость (инкр.), припуск на контуре края		
14	UZ	<_FALD>	REAL	Чистовой припуск глубина (инкр.), припуск на основании (вводится без знака)		
15	FX	<_FFP1>	REAL	Подача на конуре		
16	FZ	<_FFD>	REAL	Подача на глубину		
17		<_CDIR>	INT	Направление фрезерования (вводится без знака)		
				ПЕРВАЯ ПОЗИЦИЯ:		
					0 =	синхронный ход
					1 =	противоход
18		<_VARI>	INT	Обработка		
				ПЕРВАЯ ПОЗИЦИЯ:		
					1 =	черновая обработка
					2 =	чистовая обработка
					5 =	снятие фаски
19	L1	<_AP1>	REAL	Длина цапфы-заготовки		
20	W1	<_AP2>	REAL	Ширина цапфы-заготовки		
21	FS	<_FS>	REAL	Ширина фаски (инкр.)		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
22	ZFS	<_ZFS>	REAL	Глубина врезания (острие инструмента) при снятии фаски (абс., инкр.) см. <_AMODE>		
23		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки или только расчет стартовой точки	
					0 =	режим совместимости
					1 =	обычная обработка
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	указание размеров цапфы через центр или угол	
					0 =	режим совместимости
					1 =	указание размеров через центр
					2 =	указание размера через угловую точку, цапфа +L +W
					3 =	указание размера через угловую точку, цапфа -L +W
					4 =	указание размера через угловую точку, цапфа +L -W
					5 =	указание размера через угловую точку, цапфа -L -W
				ПЯТАЯ ПОЗИЦИЯ:	полная обработка или доводка	
					0 =	режим совместимости
					1 =	комплексная обработка
					2 =	доработка
24		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	---	зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ:	---	зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	ввод: полный
					1 =	ввод: простой

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
25		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная глубина Z1 (DP)	
					0 =	совместимость
					1 =	инкрементальная
					2 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски (ZFS)	
					0 =	абсолютная
					1 =	инкрементальная

4.26.1.18 CYCLE77 – Круговая цапфа

Синтаксис

```
CYCLE77(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_CDIAM>, <_PA>,
<_PO>, <_MID>, <_FAL>, <_FALD>, <_FFP1>, <_FFD>, <_CDIR>, <_VARI>,
<_AP1>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	RP	<_RTP>	REAL	Плоскость отвода (абс.)
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)
3	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)
4	Z1	<_DP>	REAL	Глубина цапфы (абс.)
5		<_DPR>	REAL	Глубина цапфы (инкр.), относительно Z0 (вводится без знака)
6	∅	<_CDIAM>	REAL	Диаметр цапфы (вводится без знака)
7	X0	<_PA>	REAL	Исходная точка цапфы, 1-я ось плоскости (абс.)
8	Y0	<_PO>	REAL	Исходная точка цапфы, 2-я ось плоскости (абс.)
9	DZ	<_MID>	REAL	Макс. подача на глубину (инкр., вводится без знака)
10	UXY	<_FAL>	REAL	Чистовой припуск плоскость (инкр.), припуск на контуре края
11	UZ	<_FALD>	REAL	Чистовой припуск глубина (инкр.), припуск на основании (вводится без знака)
12	FX	<_FFP1>	REAL	Подача на конуре
13	FZ	<_FFD>	REAL	Подача на глубину

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
14		<_CDIR>	INT	Направление фрезерования (вводится без знака)		
				ПЕРВАЯ ПОЗИЦИЯ:		
					0 =	синхронный ход
					1 =	противоход
15		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	Обработка	
					1 =	черновая обработка до чистового припуска
					2 =	чистовая обработка (припуск X/Y/Z = 0)
					5 =	снятие фаски
16	∅1	<_AP1>	REAL	Диаметр цапфы-заготовки		
17	FS	<_FS>	REAL	Ширина фаски (инкр.)		
18	ZFS	<_ZFS>	REAL	Глубина врезания (острие инструмента) при снятии фаски (абс./инкр.) см. <_AMODE>		
19		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки/только расчет стартовой точки	
					0 =	режим совместимости
					1 =	обычная обработка
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	зарезервировано	
				ПЯТАЯ ПОЗИЦИЯ:	полная обработка/доводка	
					0 =	режим совместимости (<_AP1> обрабатывать как прежде)
					1 =	комплексная обработка
					2 =	доработка

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
20		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	---	зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ:	---	зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	ввод: полный
					1 =	ввод: простой
21		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная глубина Z1 (DP)	
					0 =	абсолютная (режим совместимости)
					1 =	инкрементальная
					2 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски (ZFS)	
					0 =	абсолютная
					1 =	инкрементальная

4.26.1.19 CYCLE78 - Сверлильное резьбофрезерование

Синтаксис

```
CYCLE78(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_ADPR>, <_FDPR>, <_LDPR>,
<_DIAM>, <_PIT>, <_PITA>, <_DAM>, <_MDEP>, <_VARI>, <_CDIR>, <_GE>,
<_FFD>, <_FRDP>, <_FFR>, <_FFP2>, <_FFA>, <_PITM>, <_PTAB>,
<_PTABA>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	RP	<_RTP>	REAL	Плоскость отвода (абс.)
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
3	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<_DP>	REAL	Конечная глубина сверления (абс./инкр.), см. <_AMODE>		
5		<_ADPR>	REAL	Глубина засверливания с уменьшенной подачей сверления (инкр.), действует при <_VARI> ПЯТАЯ ПОЗИЦИЯ		
6	D	<_FDPR>	REAL	Макс. подача на глубину (инкр.) D ≥ Z1 ⇒ подача на конечную глубину сверления D < Z1 ⇒ цикл глубокого сверления с несколькими подачами и удалением стружки		
7	ZR	<_LDPR>	REAL	Остаточная глубина сверления при сквозном сверлении (инкр.), с подачей FR		
8	∅	<_DIAM>	REAL	Ном. диаметр резьбы		
9	P	<_PIT>	REAL	Шаг резьбы как числовое значение		
10		<_PITA>	INT	Нормирование шага резьбы P		
					1 =	шаг в мм/об
					2 =	шаг в витках/дюйм
					3 =	шаг в дюймах/об
					4 =	шаг как MODUL
11	DF	<_DAM>	REAL	Величина/процент для каждой следующей подачи (дегрессия), см. <_AMODE>		
12	V1	<_MDEP>	REAL	Мин. подача (инкр.), действует только при дегрессии		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
13		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	Удаление стружки перед резьбофрезерованием	
					0 =	без удаления стружки перед резьбофрезерованием (действует только на конечной глубине сверления)
					1 =	удаление стружки перед резьбофрезерованием (действует только на конечной глубине сверления)
				ТРЕТЬЯ ПОЗИЦИЯ:	Правая/левая резьба	
					0 =	правая резьба
					1 =	левая резьба
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	остаточная глубина сверления с подачей сверления	
					0 =	нет остаточной глубины сверления с подачей сверления FR
					1 =	остаточная глубина сверления с подачей сверления FR
				ПЯТАЯ ПОЗИЦИЯ:	Засверливание с уменьшенной подачей	
					0 =	нет засверливания с уменьшенной подачей
					1 =	засверливание с уменьшенной подачей Подача засверливания = 0,3 F1, если F1 < 0,15 мм/об Подача засверливания = 0,1 мм/об, если F1 ≥0,15 мм/об
14		<_CDIR>	INT	Направление фрезерования	0 =	синхронный ход
					1 =	противоход
					4 =	противоход + синхронный ход (комбинация черновой + чистовой обработки)
15	Z2	<_GE>	REAL	Величина отвода перед резьбофрезерованием (инкр.)		
16	F1	<_FFD>	REAL	Подача сверления (мм/мин или дюймов/мин или мм/об)		
17	FR	<_FRDP>	REAL	Подача сверления для остаточной глубины сверления (мм/мин или мм/об)		
18	F2	<_FFR>	REAL	Подача для резьбофрезерования (мм/мин или мм/зуб)		
19	FS	<_FFP2>	REAL	Чистовая подача для <_CDIR> = 4 (мм/мин или мм/зуб)		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
20		<_FFA>	INT	Нормирование подач		
				ПЕРВАЯ ПОЗИЦИЯ:	подача сверления F1	
				ВТОРАЯ ПОЗИЦИЯ:	подача сверления для остаточной глубины сверления FR	
				ТРЕТЬЯ ПОЗИЦИЯ:	подача для резьбофрезерование F2	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	чистовая подача FS	
21		<_PITM>	STRING[15]	Строка как меркер для ввода шага резьбы (только для интерфейса)[1)]		
22		<_PTAB>	STRING[20]	Строка для таблицы резьб ("", "ISO", "BSW", "BSP", "UNC") (только для интерфейса)[1)]		
23		<_PTABA>	STRING[20]	Строка для выбора в таблице резьб (к примеру, "M 10", "M 12", ...) (только для интерфейса)[1)]		
24		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений), зарезервировано		
25		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
26		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления = конечная глубина сверления Z1 абс./инкр.	
					0 =	абсолютная
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	величина/процент DF для каждой следующей подачи (дегрессия)	
					0 =	значение
					1 =	процент (от 0.001 до 100 %)

Примечание

[1)] Параметры 21, 22 и 23 используются только при выборе резьбы в таблицах резьб маски ввода. Доступ к таблицам резьб через определение цикла при выполнении цикла невозможен.

4.26.1.20 CYCLE79 - многогранник

Синтаксис

```
CYCLE79(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_NUM>, <_SWL>, <_PA>,
<_PO>, <_STA>, <_RC>, <_AP1>, <_MIDA>, <_MID>, <_FAL>, <_FALD>,
```

```
<_FFP1>, <_CDIR>, <_VARI>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	RP	<_RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)		
3	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<_DP>	REAL	Глубина многогранника (абс./инкр.), см. <_AMODE>		
5	n	<_NUM>	INT	Число граней (1...n)		
6	SW / L	<_SWL>	REAL	Размер под ключ или длина грани (в зависимости от <_VARI>) ("SW" для размера под ключ, "L" для длины грани) Размер под ключ только при четном числе граней и одной грани		
7	X0	<_PA>	REAL	Исходная точка цапфы, 1-я ось (абс.)		
8	Y0	<_PO>	REAL	Исходная точка цапфы, 2-я ось (абс.)		
9	α0	<_STA>	REAL	Угол поворота центра грани к 1-й оси (ось X)		
10	R1/FS1	<_RC>	REAL	Закругление углов при <_NUM>> 2 (радиус/фаска, см. <_AMODE>) (инкр., ввод без знака) ("R1" для радиуса, "FS1" для фаски)		
11	∅	<_AP1>	REAL	Диаметр трубы цапфы		
12	DXY	<_MIDA>	REAL	Макс. ширина подачи (единица, см. <_AMODE>)		
13	DZ	<_MID>	REAL	Макс. подача на глубину		
14	UXY	<_FAL>	REAL	Чистовой припуск плоскость		
15	UZ	<_FALD>	REAL	Чистовой припуск глубина		
16	F	<_FFP1>	REAL	Подача обработки		
17		<_CDIR>	INT	Направление фрезерования	0 =	синхронный ход
					1 =	противоход
18		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	Обработка	
					1 =	черновая обработка
					2 =	чистовая обработка
					3 =	чистовая обработка края
					5 =	снятие фаски
				ВТОРАЯ ПОЗИЦИЯ:	Размер под ключ или длина граней	
					0 =	размер под ключ
					1 =	длина грани
19	FS	<_FS>	REAL	Ширина фаски (инкр.)		
20	ZFS	<_ZFS>	REAL	Глубина врезания (острие инструмента) при снятии фаски (абс./инкр.) см. <_AMODE>		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
21		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки или только расчет стартовой точки	
					1 =	обычная обработка
22		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	---	зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ:	---	зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	ввод: полный
					1 =	ввод: простой
23		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	конечная глубина (<_DP>)	
					0 =	абсолютная
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	единица для подачи в плоскости (<_MIDA>)	
					0 =	мм
					1 =	% от диаметра инструмента
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски (<_ZFS>)	
					0 =	абсолютная
					1 =	инкрементальная
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	закругление углов (<_RC>)	
					0 =	радиус
					1 =	фаска

4.26.1.21 CYCLE81 - сверление, центрование

Синтаксис

```
CYCLE81(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	RP	<RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<RFP>	REAL	Исходная точка (абс.)		
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1/Ø	<DP>	REAL	Глубина сверления (абс.) / диаметр центрования (абс.), см. <_GMODE>		
5	Z1	<DPR>	REAL	Глубина сверления (инкр.)		
6	DT	<DTB>	REAL	Время ожидания на конечной глубине сверления, см. <_AMODE>		
7		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	центрование относительно глубины/диаметра	
					0 =	совместимость, глубина
					1 =	диаметр
8		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
9		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления Z1 (абс./инкр.)	
					0 =	совместимость, из программирования DP/DPR
					1 =	инкрементальная
					2 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	время ожидания на конечной глубине сверления DT в секундах/оборотах	
					0 =	совместимость, из знака DTB (> 0 секунд или < 0 оборотов)
					1 =	в секундах
					2 =	в оборотах

4.26.1.22 CYCLE82 - сверление, цекование

Синтаксис

```
CYCLE82(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>, <_VARI>, <S_ZA>, <S_FA>, <S_ZD>, <S_FD>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	RP	<RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<RFP>	REAL	Исходная точка (абс.)		
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<DP>	REAL	Глубина сверления (абс.), см. <_AMODE>		
5	Z1	<DPR>	REAL	Глубина сверления (инкр.), см. <_AMODE>		
6	DT	<DTB>	REAL	Время ожидания на конечной глубине сверления, см. <_AMODE>		
7		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	глубина сверления относительно острия/хвостовика	
					0 =	совместимость, острие
					1 =	хвостовик
8		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	зарезервировано	
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	ввод: полный
					1 =	ввод: простой

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
9		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления Z1 (абс./инкр.)	
					0 =	совместимость из программирования DP/DPR
					1 =	инкрементальная
					2 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	время ожидания DT на конечной глубине сверления в секундах/оборотах	
					0 =	совместимость, из знака DT (>0 секунд/ < 0 оборотов)
					1 =	в секундах
					2 =	в оборотах
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина засверливания ZA абс./инкр.	
					0 =	инкрементальная
					1 =	абсолютная
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	нормирование подачи засверливания	
					0 =	в % от подачи сверления
					1 =	F/мин
					2 =	F/об
				ПЯТАЯ ПОЗИЦИЯ:	остаточная глубина сверления ZD абс./инкр.	
					0 =	инкрементальная
					1 =	абсолютная
				ШЕСТАЯ ПОЗИЦИЯ:	нормирование остаточной подачи сверления	
					0 =	в % от подачи сверления
					1 =	F/мин
					2 =	F/об
10		<_VARI>	INT	Режим обработки, засверливание/сквозное сверление		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	Сквозное сверление	
					0 =	сквозное сверление "Нет"
					1 =	сквозное сверление "Да"
				ПЯТАЯ ПОЗИЦИЯ:	Засверливание	
					0 =	засверливание "Нет"
					1 =	засверливание "Да"
11	ZA	<S_ZA>	REAL	Глубина засверливания инкрементальная относительно исходной точки или абсолютная (см. <_AMODE> ТРЕТЬЯ ПОЗИЦИЯ)		
12	FA	<S_FA>	REAL	Подача засверливания как значение или в % (в комбинации с <_AMODE>ЧЕТВЕРТАЯ ПОЗИЦИЯ)		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
13	ZD	<S_ZD>	REAL	Остаточная глубина сверления инкрементальная относительно конечной глубины сверления или абсолютная (в комбинации с <_AMODE> ПЯТАЯ ПОЗИЦИЯ)
14	FD	<S_FD>	REAL	Остаточная подача сверления как значение или в % (в комбинации с <_AMODE>ШЕСТАЯ ПОЗИЦИЯ)

4.26.1.23 CYCLE83 – Глубокое сверление 1

Синтаксис

```
CYCLE83(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <FDEP>, <FDPR>, <_DAM>,
<DTB>, <DTS>, <FRF>, <VARI>, <_AXN>, <_MDEP>, <_VRT>, <_DTD>,
<_DIS1>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1	RP	<RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<RFP>	REAL	Исходная точка (абс.)		
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<DP>	REAL	Конечная глубина сверления (абс.), см. <_AMODE>		
5	Z1	<DPR>	REAL	Конечная глубина сверления (инкр.), см. <_AMODE>		
6	D	<FDEP>	REAL	1. Глубина сверления (абс.), см. <_AMODE>		
7	D	<FDPR>	REAL	1. Глубина сверления (инкр.), см. <_AMODE>		
8	DF	<_DAM>	REAL	Величина дегрессии / процент для каждой следующей подачи, см. <_AMODE>		
9	DTB	<DTB>	REAL	Время ожидания на глубине сверления, см. <_AMODE>		
10	DTS	<DTS>	REAL	Время ожидания в начальной точке (только при удалении стружки), см. <_AMODE>		
11	FD1	<FRF>	REAL	Процент для подачи при первой подаче, см. <_AMODE>		
12		<VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	ломка стружки/удаление стружки	
					0 =	ломка стружки
					1 =	удаление стружки
13		<_AXN>	INT	Ось инструмента		
					0 =	3-я гео-ось
					1 =	1-я гео-ось
					2 =	2-я гео-ось
					> 2	3-я гео-ось
14	V1	<_MDEP>	REAL	Мин. подача на глубину (только при проценте для дегрессии)		

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
15	V2	<_VRT>	REAL	Величина отвода после каждой обработки (только при ломке стружки)		
					> 0	переменная величина отвода
					0 =	стандартное значение 1 мм
16	DT	<_DTD>	REAL	Время ожидания на конечной глубине сверления, см. <_AMODE>		
17	V3	<_DIS1>	REAL	Упреждающий зазор (только при удалении стружки, см. <_AMODE>		
18		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	глубина сверления относительно острия/хвостовика	
					0 =	острие
					1 =	хвостовик
19		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	---	зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ:	---	зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	ввод: полный
					1 =	ввод: простой

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
20		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления = конечная глубина сверления Z1 (абс./инкр.)	
					0 =	совместимость, из программирования <DP>/<DPR>
					1 =	инкрементальная
					2 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	время ожидания на глубине сверления DTB в секундах/оборотах	
					0 =	совместимость из знака DTB (> 0 секунд или < 0 оборотов)
					1 =	в секундах
					2 =	в оборотах
				ТРЕТЬЯ ПОЗИЦИЯ:	время ожидания в начальной точке DTS в секундах/оборотах	
					0 =	совместимость из знака DTS (> 0 секунд или < 0 оборотов)
					1 =	в секундах
					2 =	в оборотах
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	время ожидания на конечной глубине сверления DTD в секундах/оборотах	
					0 =	совместимость из знака DTD (> 0 секунд или < 0 оборотов)
					1 =	в секундах
					2 =	в оборотах
				ПЯТАЯ ПОЗИЦИЯ:	1. Глубина сверления D (абс./инкр.)	
					0 =	совместимость, из программирования <FDEPF>/<DPR>
					1 =	инкрементальная
					2 =	абсолютная
				ШЕСТАЯ ПОЗИЦИЯ:	Величина дегрессии / процент <_DAM> для каждой следующей подачи	
					0 =	совместимость, из знака <_DAM>(> 0 величина дегрессии или < 0 коэффициент от 0,001 до 1,0)
					1 =	Величина дегрессии
					2 =	процент (от 0,001 до 100 %)
				СЕДЬМАЯ ПОЗИЦИЯ:	упреждающий зазор V3 автоматический/ручной	
					0 =	совместимость, из знака <_DIS1> (= 0 автоматический или > 0 ручной)
					1 =	автоматический (вычисляется в цикле)

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
					2 =	ручной (запрограммированное значение)
				ВОСЬМАЯ ПОЗИЦИЯ:	коэффициент подачи для первой подачи <FRF> как коэффициент/процент	
					0 =	совместимость, как коэффициент (от 0,001 до 1,0, FRF = 0 означает 100 %)
					1 =	процент (от 0,001 до 999,999 %)

4.26.1.24 CYCLE84 - нарезание внутренней резьбы без компенсирующего патрона

Синтаксис

```
CYCLE84(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDAC>, <MPIT>,
<PIT>, <POSS>, <SST>, <SST1>, <_AXN>, <_PITA>, <_TECHNO>, <_VARI>,
<_DAM>, <_VRT>, <_PITM>, <_PTAB>, <_PTABA>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	RP	<RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<RFP>	REAL	Исходная точка (абс.)		
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<DP>	REAL	Глубина сверления = конечная глубина сверления (абс.), см. <_AMODE>		
5	Z1	<DPR>	REAL	Глубина сверления = конечная глубина сверления (инкр.), см. <_AMODE>		
6	DT	<DTB>	REAL	Время ожидания на глубине сверления в секундах		
7	SDE	<SDAC>	INT	Направление вращения после завершения цикла		
8		<MPIT>	REAL	Размер резьбы только для "ISO метрической" (шаг вычисляется при внутренней обработке)		
9	P	<PIT>	REAL	Шаг резьбы как значение, единицу см. <_PITA>		
10	αS[1)]	<POSS>	REAL	Позиция шпинделя для ориентированного останова шпинделя		
11	S	<SST>	REAL	Скорость шпинделя для нарезания внутренней резьбы		
12	SR	<SST1>	REAL	Скорость шпинделя для отвода		
13		<_AXN>	INT	Ось сверления	0 =	3. гео-ось
					1 =	1-я гео-ось
					2 =	2-я гео-ось
					≥ 3 =	3-я гео-ось

№	Параметр в маске	Параметр внутренний	Тип данных	Значение			
14		<_PITA>	INT	Единица измерения шага резьбы (обработка <PIT> и <MPIT>)			
					0 =	шаг в мм	- обработка <MPIT>/<PIT>
					1 =	шаг в мм	- обработка <PIT>
					2 =	шаг в TPI	- обработка <PIT> (витков резьбы на дюйм)
					3 =	шаг в дюймах	- обработка <PIT>
					4 =	MODUL	- обработка <PIT>
15		<_TECHNO>	INT	Технология[1)]			
				ПЕРВАЯ ПОЗИЦИЯ:	параметры точного останова		
					0 =	активны параметры точного останова как перед вызовом цикла	
					1 =	точный останов G601	
					2 =	точный останов G602	
					3 =	точный останов G603	
				ВТОРАЯ ПОЗИЦИЯ:	предуправление		
					0 =	активно с/без предуправления как перед вызовом цикла	
					1 =	с предуправлением FFWON	
					2 =	без предуправления FFWOF	
				ТРЕТЬЯ ПОЗИЦИЯ:	ускорение		
					0 =	SOFT/BRISK/DRIVE активно как перед вызовом цикла	
					1 =	с ограничением рывка SOFT	
					2 =	без ограничения рывка BRISK	
					3 =	уменьшенное ускорение DRIVE	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	MCALL шпиндельный режим		
					0 =	при MCALL снова активировать шпиндельный режим	
					1 =	при MCALL оставаться в управлении по положению	
16		<_VARI>	INT	Режим обработки			
				ПЕРВАЯ ПОЗИЦИЯ:	0 =	1 проход	
					1 =	ломка стружки (нарезание глубокой внутренней резьбы)	
					2 =	удаление стружки (нарезание глубокой внутренней резьбы)	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	ISO/SIEMENS режим для маски ввода не релевантно		
					0 =	вызов из ISO-совместимости	
					1 =	вызов из контекста SIEMENS	

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
17	D	<_DAM>	REAL	Макс. подача на глубину (только при удалении стружки/ломке стружки)		
18	V2	<_VRT>	REAL	Величина отвода после каждой обработки (только при ломке стружки), см. <_AMODE>		
19		<_PITM>	STRING[15]	Строка как меркер для ввода шага резьбы[2)]		
20		<_PTAB>	STRING[5]	Строка для таблицы резьб ("", "ISO", "BSW", "BSP", "UNC")[2)]		
21		<_PTABA>	STRING[20]	Строка для выбора в таблице резьб (к примеру, "M 10", "M 12", ...)[2)]		
22		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных геозначений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
23		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	режим совместимости (только для маски ввода для обратного перевода), если MD 52216 Бит0 = 1[1)]	
					0 =	технологические параметры отображаются (совместимость): действуют TECHNO-параметры
					1 =	технологические параметры не отображаются: действует технология "как запрограммировано перед вызовом цикла"
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
24		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления = конечная глубина сверления Z1 (абс./инкр.)	
					0 =	совместимость, из программирования <DP>/<DPR>
					1 =	инкрементальная
					2 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	направление вращения резьбы правое/левое	
					0 =	совместимость, из знака PIT/MPIT
					1 =	справа
					2 =	слева
				ПЯТАЯ ПОЗИЦИЯ:	зарезервировано	
				ШЕСТАЯ ПОЗИЦИЯ:	зарезервировано	
				СЕДЬМАЯ ПОЗИЦИЯ:	величина отвода после каждой обработки V2 вручную/автоматически	
					0 =	совместимость, из программирования <_VRT> (> 0 переменное значение или ≤ 0 стандартное значение 1 мм/0,0394 дюйма)
					1 =	автоматически (стандартное значение 1 мм/0,0394 дюйма)
					2 =	вручную (как запрограммировано в V2)

[1] Поля технологии могут быть скрыты в зависимости от установочных данных SD52216 $MCS_FUNCTION_MASK_DRILL

[2] Параметры 19, 20 и 21 используются только при выборе резьбы в таблицах резьб маски ввода. Доступ к таблицам резьб через определение цикла при выполнении цикла невозможен.

4.26.1.25 CYCLE85 - развертывание

Синтаксис

```
CYCLE85(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <FFR>, <RFF>,
<_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	RP	<RTP>	REAL	Плоскость отвода (абс.)
2	Z0	<RFP>	REAL	Исходная точка (абс.)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<DP>	REAL	Глубина сверления (абс.), см. <_AMODE>		
5	Z1	<DPR>	REAL	Глубина сверления (инкр.), см. <_AMODE>		
6	DT	<DTB>	REAL	Время ожидания на конечной глубине сверления, см. <_AMODE>		
7	F	<FFR>	REAL	Подача		
8	FR	<RFF>	REAL	Подача при отводе		
9		<_GMODE>	INT	зарезервировано		
10		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
11		<_AMODE>	INT	Альтернативный режим (сверление)		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления Z1 (абс./инкр.)	
					0 =	совместимость, из программирования DP/DPR
					1 =	инкрементальная
					2 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	время ожидания DT на конечной глубине сверления в секундах/оборотах	
					0 =	совместимость, из знака DT (> 0 секунд или < 0 оборотов)
					1 =	в секундах
					2 =	в оборотах

4.26.1.26 CYCLE86 - растачивание

Синтаксис

```
CYCLE86(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDIR>, <RPA>,
<RPO>, <RPAP>, <POSS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	RP	<RTP>	REAL	Плоскость отвода (абс.)
2	Z0	<RFP>	REAL	Исходная точка (абс.)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<DP>	REAL	Глубина сверления (абс.), см. <_AMODE>		
5	Z1	<DPR>	REAL	Глубина сверления (инкр.), см. <_AMODE>		
6	DT	<DTB>	REAL	Время ожидания на конечной глубине сверления, см. <_AMODE>		
7	DIR	<SDIR>	INT	Направление вращения шпинделя	3 =	M3
					4 =	M4
8	Dx	<RPA>	REAL	Величина отвода в направлении X		
9	DY	<RPO>	REAL	Величина отвода в направлении Y		
10	DZ	<RPAP>	REAL	Величина отвода в направлении Z		
11	SPOS	<POSS>	REAL	Позиция шпинделя для отвода (для ориентированного останова шпинделя, в градусах)		
12		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	режим подъема	
					0 =	подъем, совместимость
					1 =	Нет подъема
13		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
14		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления Z1 (абс./инкр.)	
					0 =	совместимость, из программирования <DP>/<DPR>
					1 =	инкрементальная
					2 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	время ожидания на конечной глубине сверления DT в секундах/оборотах	
					0 =	совместимость, из знака DT (> 0 секунд или < 0 оборотов)
					1 =	в секундах
					2 =	в оборотах

4.26.1.27 CYCLE92 - отрез

Синтаксис

```
CYCLE92(<_SPD>, <_SPL>, <_DIAG1>, <_DIAG2>, <_RC>, <_SDIS>, <_SV1>,
<_SV2>, <_SDAC>, <_FF1>, <_FF2>, <_SS2>, <_DIAGM>, <_VARI>, <_DN>,
<_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	X0	<_SPD>	REAL	Исходная точка (абс., всегда диаметр)		
2	Y0	<_SPL>	REAL	Исходная точка (абс.)		
3	X1	<_DIAG1>	REAL	Глубина для снижения скорости, см. <_AMODE>(ПЕРВАЯ ПОЗИЦИЯ)		
4	X2	<_DIAG2>	REAL	Конечная глубина, см. <_AMODE>(ВТОРАЯ ПОЗИЦИЯ)		
5	R/FS	<_RC>	REAL	Радиус закругления или ширина фаски, см. <_AMODE>(ЧЕТВЕРТАЯ ПОЗИЦИЯ)		
6	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
7	S	<_SV1>	REAL	Постоянная скорость шпинделя, см. <_AMODE> (ПЯТАЯ ПОЗИЦИЯ)		
	V			Постоянная скорость резания		
8	SV	<_SV2>	REAL	Макс. скорость при постоянной скорости резания		
9	DIR	<_SDAC>	INT	Направление вращения шпинделя	3 =	для M3
					4 =	для M4
10	F	<_FF1>	REAL	Подача до глубины для снижения скорости		
11	FR	<_FF2>	REAL	Уменьшенная подача до конечной глубины		
12	SR	<_SS2>	REAL	Уменьшенная скорость до конечной глубины		
13	XM	<_DIAGM>	REAL	Глубина выдвижения приемника деталей (абс., всегда диаметр)		
14		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	отвод	
					0 =	отвод на <_SPD> + <_SDIS>
					1 =	без отвода в конце
				ВТОРАЯ ПОЗИЦИЯ:	приемник деталей	
					0 =	нет, не выполнять M-команду
					1 =	да, вызов CUST_TECHCYC(101)- выдвинуть деталеприемник, CUST_TECHCYC(102)- закрыть деталеприемник
15		<_DN>	INT	D-номер для 2-го резца, если не запрограммирован ⇒ D+1		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
20		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
21		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина для снижения скорости (<_DIAG1>)	
					0 =	абсолютная, значение поперечной оси в диаметре
					1 =	инкрементальная, значение поперечной оси в радиусе
				ВТОРАЯ ПОЗИЦИЯ:	конечная глубина (<_DIAG2>)	
					0 =	абсолютная, значение поперечной оси в диаметре
					1 =	инкрементальная, значение поперечной оси в радиусе
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	радиус/фаска (<_RC>)	
					0 =	Радиус
					1 =	Фаска
				ПЯТАЯ ПОЗИЦИЯ:	скорость шпинделя/скорость резания (<_SV1>)	
					0 =	постоянная скорость шпинделя
					1 =	постоянная скорость резания

4.26.1.28 CYCLE95 - съём припуска вдоль контура

Синтаксис

```
CYCLE95(<NPP>, <MID>, <FALZ>, <FALX>, <FAL>, <FF1>, <FF2>, <FF3>,
<_VARI>, <DT>, <DAM>, <_VRT>, <_GMODE>, <_DMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	CON	<NPP>	STRING [140]	Имя контура		
2	D	<MID>	REAL	Макс. подача на глубину при черновой обработке, см. <_GMODE>		
3	UZ	<FALZ>	REAL	Чистовой припуск в Z		
4	UX	<FALX>	REAL	Чистовой припуск в X		
5	U	<FAL>	REAL	Чистовой припуск параллельно контуру (действует в обеих осях)		
6	F	<FF1>	REAL	Подача для черновой обработки		
7	FY	<FF2>	REAL	Подача врезания поднутрения		
8	FS	<FF3>	REAL	Чистовая подача		
9		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ и ВТОРАЯ позиция:		
					1 =	черновая обработка, вдоль, снаружи
					2 =	черновая обработка, поперек, снаружи
					3 =	черновая обработка, вдоль, внутри
					4 =	черновая обработка, поперек, внутри
					5 =	чистовая обработка, вдоль, снаружи
					6 =	чистовая обработка, поперек, снаружи
					7 =	чистовая обработка, вдоль, внутри
					8 =	чистовая обработка, поперек, внутри
					9 =	комплексная обработка, вдоль, снаружи
					10 =	комплексная обработка, поперек, снаружи
					11 =	комплексная обработка, вдоль, внутри
					12 =	комплексная обработка, поперек, внутри
				ТРЕТЬЯ ПОЗИЦИЯ:		
					0 =	с возвратом по контуру, без остаточных углов
					1 =	с прямым возвратом
					2 =	возврат только до прежней точки резания, возможны остаточные углы
10	DT	<DT>	REAL	Время ожидания при прерывании подачи		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
11	DI	<DAM>	REAL	Интервал прерываний подачи		
12	VRT	<_VRT>	REAL	Путь отвода от контура		
					0 =	внутри системы используется путь отвода в 1 мм независимо от дюймовой/метрической системы единиц
					> 0 =	путь отвода
13		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	Нормирование глубины подачи	
					0 =	глубина подачи рассчитывается согласно G-группе DIAMON/DIAMOF
					1 =	глубина подачи действует как значение радиуса (независимо от DIAMON/DIAMOF)
14		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:		
					0 =	режим совместимости: имя контура стоит в NPP
					1 =	имя контура программируется в CYCLE62 и передается в _SC_CONT_NAME

4.26.1.29 CYCLE98 - цепочка резьб

Синтаксис

```
CYCLE98(<_PO1>, <_DM1>, <_PO2>, <_DM2>, <_PO3>, <_DM3>, <_PO4>,
<_DM4>, <APP>, <ROP>, <TDEP>, <FAL>, <_IANG>, <NSP>, <NRC>, <NID>,
<_PP1>, <_PP2>, <_PP3>, <_VARI>, <_NUMTH>, <_VRT>, <_MID>, <_GDEP>,
<_IFLANK>, <_PITA>, <_PITM1>, <_PITM2>, <_PITM3>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	Z0	<_PO1>	REAL	Исходная точка в Z (абс.)
2	X0	<_DM1>	REAL	Исходная точка в X (абс.), в диаметре
3	Z1	<_PO2>	REAL	Промежуточная точка 1 в Z (абс./инкр.), см. <_AMODE>(ПЕРВАЯ ПОЗИЦИЯ)
4	X1	<_DM2>	REAL	Промежуточная точка 1 в X (абс./инкр.), см. <_AMODE>(ВТОРАЯ ПОЗИЦИЯ) или
	X1α			Наклон резьбы 1 (-90° до 90°) абс. всегда диаметр, инкр. всегда радиус
5	Z2	<_PO3>	REAL	Промежуточная точка 2 в Z (абс./инкр.), см. <_AMODE>(ТРЕТЬЯ ПОЗИЦИЯ)
6	X2	<_DM3>	REAL	Промежуточная точка 2 в X (абс./инкр.), см. <_AMODE>(ЧЕТВЕРТАЯ ПОЗИЦИЯ) или
	X2α			Наклон резьбы 2 (-90° до 90°) абс. всегда диаметр, инкр. всегда радиус
7	Z3	<_PO4>	REAL	Конечная точка в Z (абс./инкр.), см. <_AMODE> (ПЯТАЯ ПОЗИЦИЯ)
8	X3	<_DM4>	REAL	Конечная точка в X, (абс./инкр.), см. <_AMODE>(ШЕСТАЯ ПОЗИЦИЯ) или
	X3α			Наклон резьбы 3 (-90° до 90°) абс. всегда диаметр, инкр. всегда радиус
9	LW	<APP>	REAL	Заход резьбы (инкр., ввод без знака)
10	LR	<ROP>	REAL	Выход резьбы (инкр., ввод без знака)
11	H1	<TDEP>	REAL	Глубина резьбы (инкр., ввод без знака)
12	U	<FAL>	REAL	Чистовой припуск в X и Z
13	DP	<_IANG>	REAL	Наклон подачи как интервал или угол, см. <_AMODE> (СЕДЬМАЯ ПОЗИЦИЯ)
	αP			Наклон подачи действует согласно установке Параметр <_VARI>(ТРЕТЬЯ ПОЗИЦИЯ).
				Определение для <_VARI_ТРЕТЬЯ ПОЗИЦИЯ = 0 - режим совместимости:
				> 0 = боковая подача по одной стороне
				0 = подача вертикально в резьбе
				< 0 = боковая подача по разным сторонам
				Определение для _VARI_ТРЕТЬЯ ПОЗИЦИЯ<>0:
				> 0 = подача по положительной боковой стороне
				0 = подача по центру
				< 0 = подача по отрицательной боковой стороне
14	α0	<NSP>	REAL	Смещение начального угла для 1-го витка резьбы
15		<NRC>	INT	Число черновых проходов, см. <_VARI>(ПЯТАЯ ПОЗИЦИЯ)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
16	NN	<NID>	INT	Число холостых проходов		
17	P0	<_PP1>	REAL	Шаг резьбы на 1-м участке резьбы, см. <_PITA>		
18	P1	<_PP2>	REAL	Шаг резьбы на 2-м участке резьбы, см. <_PITA>		
19	P2	<_PP3>	REAL	Шаг резьбы на 3-м участке резьбы, см. <_PITA>		
20		<_VARI>	INT	Обработка		
				ПЕРВАЯ ПОЗИЦИЯ:	технология	
					1 =	наружная резьба с линейной подачей
					2 =	внутренняя резьба с линейной подачей
					3 =	наружная резьба с дегрессивной подачей, поперечное сечение резания остается постоянным
					4 =	наружная резьба с дегрессивной подачей, поперечное сечение резания остается постоянным
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	тип подачи	
					0 =	режим совместимости для <_IANG>
					1 =	подача с одной стороны
					2 =	попеременная подача
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	зарезервировано	
				ПЯТАЯ ПОЗИЦИЯ:	альтернативная подача на глубину	
					0 =	совместимость, задание числа черновых проходов (<_NRC>)
					1 =	задание значения для 1-й подачи (<_MID>)
				ШЕСТАЯ ПОЗИЦИЯ:	режим обработки	
					0 =	совместимость (черновая и чистовая обработка)
					1 =	черновая обработка
					2 =	чистовая обработка
					3 =	черновая и чистовая обработка
				СЕДЬМАЯ ПОЗИЦИЯ:	последовательность обработки для многозаходной резьбы	
					0 =	восходящая последовательность витков
					1 =	нисходящая последовательность витков
21	n	<_NUMTH>	INT	Число витков резьбы		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
22		<_VRT>	REAL	Интервал обратного хода (инкр.)		
					0 =	внутри системы используется путь отвода в 1 мм независимо от дюймовой/метрической системы единиц
					> 0 =	путь отвода
23	D1	<_MID>	REAL	Первая подача, см. <_VARI>(ПЯТАЯ ПОЗИЦИЯ)		
24	DA	<_GDEP>	REAL	Глубина смены захода (действует только при "многозаходная")		
					0 =	не учитывать глубину смены захода
					> 0 =	учитывать глубину смены захода
25		<_IFLANK>	REAL	Наклон подачи как ширина (только для интерфейса)		
26		<_PITA>	INT	Нормирование шага резьбы		
					0 =	режим совместимости для шага резьбы, обработка <_PP1> до <_PP3> как прежде согласно активной метрической/дюймовой системе
					1 =	шаг в мм
					2 =	шаг в TPI (витков резьбы на дюйм)
					3 =	шаг в дюймах
					4 =	MODUL
27		<_PITM1>	STRING[15]	Строка как меркер для ввода шага резьбы (только для интерфейса)		
28		<_PITM2>	STRING[15]	Строка как меркер для ввода шага резьбы (только для интерфейса)		
29		<_PITM3>	STRING[15]	Строка как меркер для ввода шага резьбы (только для интерфейса)		
30		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	---	зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ:	---	зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
31		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	1. промежуточная точка в Z (Z1)	
					0 =	абсолютная
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	1. промежуточная точка в X (X1)	
					0 =	абсолютная
					1 =	инкрементальная
					2 =	α
				ТРЕТЬЯ ПОЗИЦИЯ:	2. промежуточная точка в Z (Z2)	
					0 =	абсолютная
					1 =	инкрементальная
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	2. промежуточная точка в X (X2)	
					0 =	абсолютная
					1 =	инкрементальная
					2 =	α
				ПЯТАЯ ПОЗИЦИЯ:	конечная точка в Z (Z3)	
					0 =	абсолютная
					1 =	инкрементальная
				ШЕСТАЯ ПОЗИЦИЯ:	конечная точка в X (X3)	
					0 =	абсолютная
					1 =	инкрементальная
					2 =	α
				СЕДЬМАЯ ПОЗИЦИЯ:	выбор наклона подачи как угла или ширины	
					0 =	угол подачи <_IANG>
					1 =	наклон подачи <_IFLANK>
				ВОСЬМАЯ ПОЗИЦИЯ:	однозаходная/многозаходная	
					0 =	режим совместимости (начальный угол <_NSP>обрабатывается)
					1 =	однозаходная (со смещение начального угла <_NSP>)
					2 =	многозаходная

4.26.1.30 CYCLE99 - нарезание резьбы резцом

Синтаксис

```
CYCLE99(<_SPL>, <_SPD>, <_FPL>, <_FPD>, <_APP>, <_ROP>, <_TDEP>,
<_FAL>, <_IANG>, <_NSP>, <_NRC>, <_NID>, <_PIT>, <_VARI>, <_NUMTH>,
<_SDIS>, <_MID>, <_GDEP>, <_PIT1>, <_FDEP>, <_GST>, <_GUD>,
<_IFLANK>, <_PITA>, <_PITM>, <_PTAB>, <_PTABA>, <_DMODE>, <_AMODE>,
<_S_XRS>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1	Z0	<_SPL>	REAL	Исходная точка (абс.)		
2	X0	<_SPD>	REAL	Исходная точка (абс., всегда диаметр)		
3	Z1	<_FPL>	REAL	Конечная точка в комбинации с <_AMODE>(ПЕРВАЯ ПОЗИЦИЯ)		
4	X1	<_FPD>	REAL	Конечная точка, в комбинации с <_AMODE>(ВТОРАЯ ПОЗИЦИЯ)		
5	LW/LW2	<_APP>	REAL	Заход резьбы в комбинации с <_AMODE> (ТРЕТЬЯ ПОЗИЦИЯ) или Вход резьбы = выход резьбы в комбинации с <_AMODE>(ТРЕТЬЯ ПОЗИЦИЯ)		
6	LR	<_ROP>	REAL	Выход резьбы		
7	H1	<_TDEP>	REAL	Глубина резьбы		
8	U	<_FAL>	REAL	Чистовой припуск в X и Z		
9	DP	<_IANG>	REAL	Наклон подачи как интервал или угол, в комбинации с <_AMODE>(ЧЕТВЕРТАЯ ПОЗИЦИЯ)		
	αP				> 0 =	подача по положительной боковой стороне
					< 0 =	подача по отрицательной боковой стороне
					0 =	подача по центру
10	α0	<_NSP>	REAL	Смещение начального угла (действует только при "однозаходная")		
11	ND	<_NRC>	INT	Число черновых проходов, в комбинации с <_VARI>(ПЯТАЯ ПОЗИЦИЯ)		
12	NN	<_NID>	INT	Число холостых проходов		
13	P	<_PIT>	REAL	Шаг резьбы как значение, в комбинации с <_PITA>		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
14		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	технология	
					1 =	наружная резьба с линейной подачей
					2 =	внутренняя резьба с линейной подачей
					3 =	наружная резьба с дегрессивной подачей, поперечное сечение резания остается постоянным
					4 =	наружная резьба с дегрессивной подачей, поперечное сечение резания остается постоянным
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	тип подачи	
					1 =	подача с одной стороны
					2 =	попеременная подача
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	зарезервировано	
				ПЯТАЯ ПОЗИЦИЯ:	альтернативная подача на глубину	
					0 =	задание числа черновых проходов (<_NRC>)
					1 =	задание значения для 1-й подачи (<_MID>)
				ШЕСТАЯ ПОЗИЦИЯ:	режим обработки	
					1 =	Черновая обработка
					2 =	Чистовая обработка
					3 =	Черновая и чистовая обработка
				СЕДЬМАЯ ПОЗИЦИЯ:	последовательность обработки для многозаходной резьбы	
					0 =	восходящая последовательность витков
					1 =	нисходящая последовательность витков
15	n	<_NUMTH>	INT	Число витков резьбы		
16	VR	<_SDIS>	REAL	Интервал обратного хода, инкр.		
17	D1	<_MID>	REAL	Первая глубина подачи, в комбинации с <_VARI>(ПЯТАЯ ПОЗИЦИЯ)		
18	DA	<_GDEP>	REAL	Глубина смены захода (действует только при "многозаходная")		
					0 =	не учитывать глубину смены захода
					> 0 =	учитывать глубину смены захода

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
19	g	<_PIT1>	REAL	Изменение шага на оборот		
					0 =	постоянный шаг резьбы (G33)
					> 0 =	увеличение шага резьбы (G34)
					< 0 =	уменьшение шага резьбы (G35)
20		<_FDEP>	REAL	Глубина врезания (вводится без знака)		
21	N1	<_GST>	INT	Стартовый заход N1 = 1...N, в комбинации с <_AMODE>(ШЕСТАЯ ПОЗИЦИЯ)		
22		<_GUD>	INT	зарезервировано		
23		<_IFLANK>	REAL	Наклон подачи как ширина (только для интерфейса)		
24		<_PITA>	INT	Единица измерения шага резьбы (обработка PIT и/или MPIT)		
					0 =	шаг в мм - обработка MPIT/PIT
					1 =	шаг в мм - обработка PIT
					2 =	шаг в TPI - обработка PIT (витков резьбы на дюйм)
					3 =	шаг в дюймах - обработка PIT
					4 =	MODUL - обработка PIT
25		<_PITM>	STRING[15]	Строка как меркер для ввода шага резьбы (только для интерфейса)[1)]		
26		<_PTAB>	STRING[20]	Строка для таблицы резьб (только для интерфейса) [1)]		
27		<_PTABA>	STRING[20]	Строка для выбора таблицы резьб (только для интерфейса) [1)]		
28		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	тип резьбы	
					0 =	продольная резьба
					1 =	спиральная резьба
					2 =	коническая резьба
				ТРЕТЬЯ ПОЗИЦИЯ:	---	зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
29		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	длина резьбы в Z	
					0 =	абсолютная
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	длина резьбы в X	
					0 =	абсолютная, значение поперечной оси в диаметре
					1 =	инкрементальная, значение поперечной оси в радиусе
					2 =	α
				ТРЕТЬЯ ПОЗИЦИЯ:	нормирование захода/входного участка <_APP>	
					0 =	заход резьбы <_APP>
					1 =	вход резьбы = выход резьбы <_APP>= -<_ROP>
					2 =	задание входа резьбы <_APP> = -<_APP>
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	выбор наклона подачи как угла или ширины	
					0 =	угол подачи <_IANG>
					1 =	наклон подачи <_IFLANK>
				ПЯТАЯ ПОЗИЦИЯ:	однозаходная/многозаходная	
					0 =	однозаходная (со смещение начального угла <_NSP>)
					1 =	многозаходная
				ШЕСТАЯ ПОЗИЦИЯ:	стартовый заход <_GST>	
					0 =	полная обработка
					1 =	запустить обработку с этого витка
					2 =	обработать только этот виток
				СЕДЬМАЯ ПОЗИЦИЯ:	компенсация провисания при продольной резьбе	
					0 =	высота сегмента сферической резьбы XS
					1 =	радиус сферической резьбы RS
30	XS / RS	<_S_XRS>	REAL	Компенсация провисания при продольной резьбе в комбинации с <_AMODE>: СЕДЬМАЯ ПОЗИЦИЯ		

Примечание

[1] Параметры <_PITM>, <_PTAB> и <_PTABA> используются только при выборе резьбы в таблицах резьб маски ввода.
Доступ к таблицам резьб через определение цикла при выполнении цикла невозможен.

4.26.1.31 CYCLE435 - установка системы координат правящего инструмента

Синтаксис

```
CYCLE435(<_T>, <_DD>, <S_TA>, <S_DA>, <S_AD>, <S_AL>, <S_PVD>,
<S_PVL>, <S_PD>, <S_PL>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
1		<_T>	STRING[32]	Имя инструмента шлифовального круга		
2		<_DD>	INT	Номер резца шлифовального круга		
3		<S_TA>	STRING[32]	Исходная точка правящего инструмента - наименование правящего инструмента		
4		<S_DA>	INT	Номер резца правящего инструмента		
5		<S_AD>	REAL	Величина правки, диаметр		
6		<S_AL>	REAL	Величина правки, плоскость		
7		<S_PVD>	REAL	Смещение при профилировании, диаметр		
8		<S_PVL>	REAL	Смещение при профилировании, плоскость		
9		<S_PD>	REAL	Припуск при профилировании, диаметр		
10		<S_PL>	REAL	Припуск при профилировании, плоскость		
11		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	активный инструмент в конце цикла	
					0 =	правящий инструмент активен
					1 =	шайба активна

4.26.1.32 CYCLE495 - Профилирование

Синтаксис

```
CYCLE495(<_T>, <_DD>, <_SC>, <_F>, <_VARI>, <_D>, <_DX>, <_DZ>,
<S_PA>, <S_N>, <_DMODE>, <_AMODE>, <S_FW>, <S_HW>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1		<_T>	STRING[20]	Имя инструмента шлифовального круга
2		<_DD>	INT	Номер резца шлифовального круга
3		<_SC>	REAL	Величина отвода для обхода препятствий, инкрементальная
4		<_F>	REAL	Подача при профилировании

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
5		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	тип профилирования	
					1 =	параллельно осям
					2 =	параллельно контуру
				ВТОРАЯ ПОЗИЦИЯ:	Направление обработки	
					0 =	протягивающий возможно при положении резцов 1 - 4
					1 =	толкающий возможно при положении резцов 1 - 4
					2 =	чередующийся возможно при положении резцов 1 - 8
					3 =	начало → конец возможно при положении резцов 1 - 8
					4 =	конец → начало возможно при положении резцов 1 - 8
				ТРЕТЬЯ ПОЗИЦИЯ:	направление подачи	
					1 =	подача X- при G18 или Y- при G19
					2 =	подача X+ при G18 или Y+ при G19
					3 =	подача Z- при G18 и при G19
					4 =	подача Z+ при G18 и при G19
6		<_D>	REAL	Величина правки при профилировании параллельно оси		
7		<_DX>	REAL	Величина правки X при G18 или Y при G19 при профилировании параллельно контуру		
8		<_DZ>	REAL	Величина правки Z при G18 и G19 при профилировании параллельно контуру		
9		<S_PA>	REAL	Припуск на профилирование		
10		<S_N>	INT	Число ходов в программе профилирования		
11		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
12		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	выбор профилирования Заново/ Продолжить	
					1 =	заново
					2 =	продолжить
				ВТОРАЯ ПОЗИЦИЯ:	выбор припуска на профилирование	
					0 =	от контура заготовки до самой глубокой точки контура
					1 =	от контура заготовки до самой высокой точки контура
13		<S_FW>	REAL	Задний угол правящего инструмента		
14		<S_HW>	REAL	Угол зажима правящего инструмента		

Примечание

Интерпретация контура при профилировании посредством CYCLE495 происходит через предопределенную процедуру CONTDCON. Поскольку CONTDCON при активной коррекции радиуса инструмента (G41/G42) **недопустима**, необходимо до вызова CYCLE495 выключить при помощи G40 коррекцию радиуса инструмента.

4.26.1.33 CYCLE782 – Коррекция по загрузке

Примечание

Для использования CYCLE782 необходима лицензия на опцию "Интеллектуальная коррекция нагрузки".

Синтаксис

```
CYCLE782(<S_MODE>, <S_TESTAXIS>, <S_VALUE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1		<S_MODE>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	Выбор/отмена	
					0 =	Деактивация адаптации
					1 =	Активация адаптации
					2 =	Определение и активация базовой инерции оси (функция ввода в эксплуатацию)
				ВТОРАЯ ПОЗИЦИЯ:	Вариант измерения (только при <S_MODE> ТРЕТЬЯ ПОЗИЦИЯ = 0)	
					0 =	Нормальный режим
					1 =	Точный режим (например, с определением трения и т. п.)
				ТРЕТЬЯ	Измерение или задача значения инерции	
					0 =	Измерение В этом режиме выполняются движения перемещения для определения загрузки.
					1 =	Задача значения инерции В этом режиме не выполняются движения перемещения для определения загрузки. Вместо этого передается значение инерции, определенное ранее, например, посредством автоматической оптимизации (Automatische Servo Optimierung, AST).
				ЧЕТВЕРТАЯ ПОЗИЦИЯ	Индикация результата измерения	
					0 =	Экран результата измерения ВЫКЛ
					1 =	Экран результата измерения ВКЛ
				ПЯТАЯ ПОЗИЦИЯ	Длительность индикации результата измерения	
					1 =	Автоматическая отмена индикации через 8 сек
					3 =	Квитирование с пуском ЧПУ
				ШЕСТАЯ ПОЗИЦИЯ	Выбор, загрузка [1)]	
					0 =	Определение загрузки, всего: ось пустая + деталь
					1 =	Определение загрузки, раздельно: деталь
2		<S_TESTAXIS>	AXIS	Ось канала: • для которой необходимо активировать адаптации. • для которой необходимо определить инерцию (только при <S_MODE> ТРЕТЬЯ ПОЗИЦИЯ = 0)		
3		<S_VALUE>	REAL	Значение инерции (только при <S_MODE> ТРЕТЬЯ ПОЗИЦИЯ = 1)		

[1] Значение автоматически переносится из машинных данных 52212 $MCS_FUNCTION_MASK_TECH бит 18.

Примечание

Интерфейс пользователя не поддерживает функцию ввода в эксплуатацию "Определение и активация базовой инерции оси" (<S_MODE> EINERSTELLE = 2). Для этого необходимо программировать сам вызов цикла. Результат измерения записывается в машинные данные 53350 $MAS_ILC_BASE_VALUE.

Значение параметра "Выбор загрузки" (<S_MODE> HUNDERTTAUSENDERSTELLE) считывается интерфейсом пользователя из машинных данных 52212 $MCS_FUNCTION_MASK_TECH бит 18 и записывается в вызов цикла.

Загрузка может быть определена или задана как для линейных, так и для круговых осей. Масса для линейных осей и инерция для круговой оси всегда отображается на экране результата измерения или задается в интерфейсе пользователя.

Независимо от типа оси в машинные данные 53350 $MAS_ILC_BASE_VALUE и в адаптацию всегда записывается коррекция.

Примеры

Пример 1:

В начале программы ЧПУ необходимо определить загрузку оси MX1 и таким образом актуализировать адаптации. Результат должен быть отображен, индикацию необходимо квитировать посредством пуска ЧПУ.

Программный код

```
...
CYCLE782(31011,MX1)
...
```

Пример 2:

Значение инерции, определенное, например, посредством AST, необходимо принять по ходу программы в переменную _MY_VALUE, цикл загрузки необходимо вызвать с этим значением: тем самым включаются адаптации для оси MX2. Результат отображаться не должен.

Программный код

```
DEF REAL _MY_VALUE
_MY_VALUE=...
...
CYCLE782(101,MX2,_MY_VALUE)
...
```

Пример 3:

Адаптации для оси MX2 должны быть выключены.

Программный код
...
`CYCLE782(0,MX2)`
...

4.26.1.34 CYCLE800 – Поворот плоскости, поворот инструмента, точная установка инструмента

Синтаксис

```
CYCLE800(<_FR>, <_TC>, <_ST>, <_MODE>, <_X0>, <_Y0>, <_Z0>, <_A>,
<_B>, <_C>, <_X1>, <_Y1>, <_Z1>, <_DIR>, <_FR_I>, <_DMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1		<_FR>	INT	Режим отвода:	0 =	нет отвода
					1 =	отвод оси станка Z
					2 =	отвод оси станка Z и затем XY
					3 =	зарезервировано
					4 =	отвод в направлении инструмента макс.
					5 =	отвод в направлении инструмента инкр.
2		<_TC>	STRING[32]	Имя блока данных поворота:	""	"" (нет имени), если имеется только 1 блок данных поворота
					"0"	отмена блока данных поворота (удаление фрейма поворота)

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
3		<_ST>	INT	Состояние трансформаций		
				ПЕРВАЯ ПОЗИЦИЯ:		
					0 =	новая, повернутая плоскость удаляется и заново рассчитывается с текущими параметрами
					1 =	аддитивная, повернутая плоскость добавляется аддитивно к активной повернутой плоскости
				ВТОРАЯ ПОЗИЦИЯ:	слежение за острием инструмента да/нет (активно, то если функция установлена при вводе в эксплуатацию поворота)	
					0 =	не следить за острием инструмента
					1 =	следить за острием инструмента (TRAORI)
				ТРЕТЬЯ ПОЗИЦИЯ:	установить / точно установить инструмент (функция отображается в маске ввода ПОВОРОТ инструмента)	
					0 =	не устанавливать инструмент
					1 =	установить инструмент (преимущественно радиусные фрезы)
					2 =	точно установить **токарный** инструмент (если кинематика оси B для токарной технологии установлена при вводе поворота в эксплуатацию)
					3 =	точно установить **фрезерный** инструмент (если кинематика оси B для токарной технологии установлена при вводе поворота в эксплуатацию)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	внутренний параметр Поворот в JOG	
				ПЯТАЯ ПОЗИЦИЯ:	см. параметр Направление <_DIR>	
					0 =	поворот "да"
					1 =	поворот "нет", направление "минус" [3]
					2 =	поворот "нет", направление "плюс" [3]
				ШЕСТАЯ ПОЗИЦИЯ:	см. параметр Направление <_DIR>	
					0 =	Совместимость
					1 =	Выбор направления "минус" оптимизирован (только для интерфейса пользователя) [4]

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
					2 =	Выбор направления "плюс" оптимизирован (только для интерфейса пользователя) [4]
4		<_MODE> [5]	INT	Режим поворота: Обработка угла поворота и последовательности поворота (битовая кодировка!)		
				Бит: 7 6	0 0:	поворот каждой осью -> см. параметры <_A>, <_B>, <_C>
					0 1:	пространственный угол -> см. параметры <_A>, <_B>[1]
					1 0:	угол проецирования -> см. параметры <_A>, <_B>, <_C>[1]
					1 1:	режим поворота Круговые оси напрямую -> см. параметры <_A>, <_B>[1]
				Бит: 5 4 3 2 1 0 (для пространственных углов без значения!)	x x x x 0 1	1-й поворот _A вокруг X
					x x x x 1 0	1-й поворот _A вокруг Y
					x x x x 1 1	1-й поворот _A вокруг Z
					x x 0 1 x x	2-й поворот _B вокруг X
					x x 1 0 x x	2-й поворот _B вокруг Y
					x x 1 1 x x	2-й поворот _B вокруг Z
					0 1 x x x x	3-й поворот _C вокруг X
					1 0 x x x x	3-й поворот _C вокруг Y
					1 1 x x x x	3-й поворот _C вокруг Z
5	X0	<_X0>	REAL	Исходная точка X перед вращением		
6	Y0	<_Y0>	REAL	Исходная точка Y перед вращением		
7	Z0	<_Z0>	REAL	Исходная точка Z перед вращением		
8	X(A)	<_A>	REAL	1-е вращение согласно настройкам параметров <_MODE>		
9	Y(B)	<_B>	REAL	2-е вращение согласно настройкам параметров <_MODE>		
10	Z(C)	<_C>	REAL	3-е вращение согласно настройкам параметров <_MODE>		
11	X1	<_X1>	REAL	Исходная точка X после вращения		
12	Y1	<_Y1>	REAL	Исходная точка Y после вращения		
13	Z1	<_Z1>	REAL	Исходная точка Z после вращения		
14	- или +	<_DIR>	INT	Запустить движение перемещения круговых осей (по умолчанию = -1!)		
					-1 =	позиционировать на меньшее значение круговой оси 1 или 2 [2]
					+1 =	позиционировать на большее значение круговой оси 1 или 2 [2]
					0 =	нет поворота (только расчет фрейма поворота) [1] [3]
15	FR	<_FR_I>	REAL	Значение (инкр.) отвод в направлении инструмента инкрементальный		

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
16		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	представление бета-значения при точной установке инструмента	
					0 =	Значение
					1 =	Указатель

Примечание

Если следующие передаваемые параметры запрограммированы косвенно (как параметры), то обратный перевод маски ввода не выполняется: <_FR>, <_ST>, <_TC>, <_MODE>, <_DIR>

[1] Выбор возможен, если функция установлена при ВВОДЕ В ЭКСПЛУАТАЦИЮ ПОВОРОТА.

[2] Выбор возможен, если при ВВОДЕ В ЭКСПЛУАТАЦИЮ ПОВОРОТА зависимость направления перемещения установлена на круговую ось 1 или 2.

Нет поля выбора при Зависимость направления перемещения нет

[3] Выбор поворота "нет" может быть скрыт SD 55221 Бит 0

Поворот "нет" направление "минус" соответствует <_DIR>= 0 и _ST ПЯТАЯ ПОЗИЦИЯ = 1

Поворот "нет" направление "плюс" соответствует <_DIR>= 0 и _ST ПЯТАЯ ПОЗИЦИЯ = 2

[4] Выбор направления круговой оси 1 или 2 осуществляется и тогда, когда круговая ось с зависимостью направления перемещения находится в позиции полюса (значение позиции равно нулю).

[5] Пример кодировки: вращение каждой осью, последовательность вращения ZYX

Двоичная: 00011011; десятичная: 27

Идентификаторы осей XYZ соответствуют гео-осям канала ЧПУ. Повороты вокруг осей XYZ могут выполняться по отдельности. Например, последовательность вращения вокруг ZXZ в одном вызове CYCLE800 не разрешена

4.26.1.35 CYCLE801 – Образец позиции "Решетка или рамка"

Синтаксис

```
CYCLE801(<_SPCA>, <_SPCO>, <_STA>, <_DIS1>, <_DIS2>, <_NUM1>,
<_NUM2>, <_VARI>, <_UMODE>, <_ANG1>, <_ANG2>, <_HIDE>, <_NSP>,
<_DMODE>)
```

Параметр

№	Параметр в маске	Параметр внутр.	Тип данных	Объяснение		
1	X0	<_SPCA>	REAL	Исходная точка для образца позиций (решетка/рамка) в 1-й оси (абс.)		
2	Y0	<_SPCO>	REAL	Исходная точка для образца позиций (решетка/рамка) во 2-й оси (абс.)		
3	α0	<_STA>	REAL	Базовый угол поворота (угол к 1-й оси)	< 0 =	Вращение по часовой стрелке
					> 0 =	Вращение против часовой стрелки
4	L1	<_DIS1>	REAL	Расстояние между колонками (интервал позиции 1-й оси, вводится без знака)		
5	L2	<_DIS2>	REAL	Расстояние между рядами (интервал позиции 2-й оси, вводится без знака)		
6	N1	<_NUM1>	INT	Число колонок		
7	N2	<_NUM2>	INT	Число рядов		
8		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	образец позиции	
					0 =	Решетка
					1 =	Рамка
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
9		<_UMODE>	INT	зарезервировано		
10	αX	<_ANG1>	REAL	Угол среза к 1-й оси (наклонное положение рядов относительно 1-й оси)		
					< 0 =	измерение по часовой стрелке (от 0 до -90 градусов)
					> 0 =	измерение против часовой стрелки (от 0 до 90 градусов)
11	αY	<_ANG2>	REAL	Угол среза ко 2-й оси (наклонное положение колонок относительно 2-й оси)		
					< 0 =	измерение по часовой стрелке (от 0 до -90 градусов)
					> 0 =	измерение против часовой стрелки (от 0 до 90 градусов)

№	Параметр в маске	Параметр внутр.	Тип данных	Объяснение		
12		<_HIDE>	STRING [200]	Пропущенные позиции • макс. 198 знаков • Указание текущего номера позиции, к примеру, "1,3" (позиции 1 и 3 не выполняются)		
13		<_NSP>	INT	зарезервировано		
14		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)

4.26.1.36 CYCLE802 - любые позиции

Синтаксис

```
CYCLE802(<_XA>, <_YA>, <_X0>, <_Y0>, <_X1>, <_Y1>, <_X2>, <_Y2>,
<_X3>, <_Y3>, <_X4>, <_Y4>, <_X5>, <_Y5>, <_X6>, <_Y6>, <_X7>, <_Y7>,
<_X8>, <_Y8>, <_VARI>, <_UMODE>, <_DMODE>, <S_ABA>, <S_AB0>,
<S_AB1>, <S_AB2>, <S_AB3>, <S_AB4>, <S_AB5>, <S_AB6>, <S_AB7>,
<S_AB8>)
```

Параметр

№	Параметр в маске	Параметр внутр.	Тип данных	Значение		
1		<_XA>	INT	Альтернатива для всех X-позиций (9-значное десятичное значение) Число разрядов: 876543210 (разряд соответствует позиции сверления Xn)		
				Значение разряда:	1 =	абсолютное (1-я запрограммированная позиция всегда абсолютная!)
					2 =	инкрементальное
2		<_YA>	INT	Альтернатива для всех Y-позиций (9-значное десятичное значение) Число разрядов: 876543210 (разряд соответствует позиции сверления Yn)		
				Значение разряда:	1 =	абсолютное (1-я запрограммированная позиция всегда абсолютная!)
					2 =	инкрементальное
3	X0	<_X0>	REAL	1. позиция X		
4	Y0	<_Y0>	REAL	1. позиция Y		

№	Параметр в маске	Параметр внутр.	Тип данных	Значение
5	X1	<_X1>	REAL	2. позиция X
6	Y1	<_Y1>	REAL	2. позиция Y
7	X2	<_X2>	REAL	3. позиция X
8	Y2	<_Y2>	REAL	3. позиция Y
9	X3	<_X3>	REAL	4. позиция X
10	Y3	<_Y3>	REAL	4. позиция Y
11	X4	<_X4>	REAL	5. позиция X
12	Y4	<_Y4>	REAL	5. позиция Y
13	X5	<_X5>	REAL	6. позиция X
14	Y5	<_Y5>	REAL	6. позиция Y
15	X6	<_X6>	REAL	7. позиция X
16	Y6	<_Y6>	REAL	7. позиция Y
17	X7	<_X7>	REAL	8. позиция X
18	Y7	<_Y7>	REAL	8. позиция Y
19	X8	<_X8>	REAL	9. позиция X
20	Y8	<_Y8>	REAL	9. позиция Y

№	Параметр в маске	Параметр внутр.	Тип данных	Значение		
21		<_VARI>	INT	Обработка		
				ТРЕТЬЯ ПОЗИЦИЯ:	(только для вызова из Jobshop) (пока обрабатываются только 0 и 2)	
					0 =	нет зажима шпинделя
					1 =	зажим шпинделя только при вертикальном врезании с G00 или G01
					2 =	зажим шпинделя во время всей обработки
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	зарезервировано	
				ПЯТАЯ ПОЗИЦИЯ:	образец позиции с круговой осью/без нее – комбинация осей (с <_VARI> ШЕСТАЯ ПОЗИЦИЯ)	
					0 =	XY (только XY без круговой оси, совместимость)
					1 =	X,Y или Z и круговая ось: XA, YB, ZC (1 круговая ось с гео-осью, вокруг которой вращается круговая ось)
					2 =	XY и круговая ось: XYA, XYB, XYC (1 круговая ось с 1-й и 2-й гео-осями, без TRACYL)
				ШЕСТАЯ ПОЗИЦИЯ:	Круговая ось	
					0 =	без круговой оси (только XY, совместимость)
					1 =	ось A (круговая ось вокруг X)
					2 =	ось B (круговая ось вокруг Y)
					3 =	ось C (круговая ось вокруг Z)
				ВОСЬМАЯ ПОЗИЦИЯ + СЕДЬМАЯ ПОЗИЦИЯ:	образец позиции с круговой осью – смещение (при нескольких круговых осях вокруг одной и той же оси; если индекс слишком велик, то 1-й оси)	
					00 =	1-я ось A, B или C или при совместимости
					01 =	2-я ось A, B или C
					...	
					19 =	20-я ось A, B или C
22		<_UMODE>	INT	Выбор зажимаемого шпинделя: (только для вызова из Jobshop) (вызов цикла пользователя CUST_TECHCYC)		
					3 =	зажим/отпускание главного шпинделя
					23 =	зажим/отпускание противошпинделя

№	Параметр в маске	Параметр внутр.	Тип данных	Значение		
23		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
24		<S_ABA>	INT	Альтернатива для всех AB-позиций (9-значное десятичное значение) Число разрядов: 876543210 (разряд соответствует позиции ABn)		
				Значение разряда:	1 =	абсолютное (1-я запрограммированная позиция всегда абсолютная!)
					2 =	инкрементальное
25	A0	<S_AB0>	REAL	1-я позиция круговой оси при образце позиции с круговой осью (в комбинации с <_VARI>)		
26	A1	<S_AB1>	REAL	2-я позиция круговой оси при образце позиции с круговой осью		
27	A2	<S_AB2>	REAL	3-я позиция круговой оси при образце позиции с круговой осью		
28	A3	<S_AB3>	REAL	4-я позиция круговой оси при образце позиции с круговой осью		
29	A4	<S_AB4>	REAL	5-я позиция круговой оси при образце позиции с круговой осью		
30	A5	<S_AB5>	REAL	6-я позиция круговой оси при образце позиции с круговой осью		
31	A6	<S_AB6>	REAL	7-я позиция круговой оси при образце позиции с круговой осью		
32	A7	<S_AB7>	REAL	8-я позиция круговой оси при образце позиции с круговой осью		
33	A8	<S_AB8>	REAL	9-я позиция круговой оси при образце позиции с круговой осью		

Примечание

Ненужные позиции параметров X1/Y1/A1 — X8/Y8/A8 можно пропустить. Но альтернативные значения для <_XA>, <_YA> и <S_ABA> должны быть указаны полностью для всех 9 позиций.

При образцах позиции XA, YB или ZC (одна гео-ось и круговая ось) ось плоскости обработки, которая не перемещается через образец позиции (при G17 и XA это ось Y), нужно позиционировать перед вызовом цикла.

4.26.1.37 CYCLE806 — интерполированное вращение

Примечание

Интерполированное вращение при помощи CYCLE806 является лицензируемой опцией!

Синтаксис

```
CYCLE806(<_MODE>, <S_XC>, <S_YC>, <_STA>, <_SV1>, <_SV2>, <_SDIR>,
<_AMODE>, <_DMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1		<_MODE>	INT	Режим		
				ПЕРВАЯ ПОЗИЦИЯ:	Выбор/отмена	
					0 =	Отменить выбор интерполированного вращения
					1 =	Выбрать интерполированное вращение
2	XC	<S_XC>	REAL	Центр вращения X		
3	YC	<S_YC>	REAL	Центр вращения Y		
4	Гамма	<_STA>	REAL	Угол гамма (вращение вокруг оси инструмента); зарезервировано		
5	S	<_SV1>	REAL	постоянная частота вращения шпинделя или постоянная скорость резания (см. <_AMODE> EINER)		
6	SV	<_SV2>	REAL	Максимальное число оборотов при постоянной скорости резания (зарезервировано)		
7		<_SDIR>	INT	Зарезервировано		
					0 =	Направление вращения из инструмента
					3 =	Направление вращения M3
					4 =	Направление вращения M4
8		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	Движение шпинделя (<_SV1>)	
					0 =	постоянная скорость шпинделя
					1 =	постоянная скорость резания
9		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/18/19	
					1 =	G17 (активна только в цикле)
10	TC	<S_TRA>	STRING	• Имя трансформации • пусто: использовать 1-ю кинематическую трансформацию TRAINT		

4.26.1.38 CYCLE830 - Глубокое сверление 2

Синтаксис

```
CYCLE830(<RTP>, <RFP>, <SDIS>, <_DP>, <FDEP>, <_DAM>, <DTB>, <DTS>,
<FRF>, <VARI>, <_MDEP>, <_VRT>, <_DTD>, <_DIS1>, <S_FP>, <S_SDAC2>,
```

```
<S_SV2>, <S_FB>, <_SDAC>, <_SV1>, <S_SPOS>, <S_ZA>, <S_FA>, <S_ZP>,
<S_FS>, <S_ZS1>, <S_ZS2>, <S_N>, <S_ZD>, <S_FD>, <S_FR>, <S_SDAC3>,
<S_SV3>, <S_CON>, <S_COFF>, <_GMODE>, <_DMODE>, <_AMODE>,
<S_AMODE2>, <S_AMODE3>, <S_ZPV>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	RP	<RTP>	REAL	Плоскость отвода (абс.)
2	Z0	<RFP>	REAL	Опорная точка (абс.)
3	sc	<SDIS>	REAL	Безопасное расстояние (аддитивно к исходной точке, без знака)
4	Z1	<_DP>	REAL	Конечная глубина сверления абс./инкр. (см. <_AMODE> ПЕРВАЯ ПОЗИЦИЯ)
5	D	<FDEP>	REAL	1. Глубина сверления для ломки/удаления стружки, абсолютная или инкрементальная, относительно исходной точки при засверливании/без засверливания или относительно глубины базового отверстия <_AMODE> ПЯТАЯ ПОЗИЦИЯ)
6	DF	<_DAM>	REAL	Величина/процент для каждой следующей подачи, величина дегрессии или процент, (см. <_AMODE> ШЕСТАЯ ПОЗИЦИЯ)
7	DTB	<DTB>	REAL	Время ожидания на каждой глубине сверления (см. <_AMODE> ВТОРАЯ ПОЗИЦИЯ)
8	DTS	<DTS>	REAL	Время ожидания при удалении стружки в начальной точке (см. <_AMODE> ТРЕТЬЯ ПОЗИЦИЯ)
9	FD1	<FRF>	REAL	Процент для подачи при первой подаче, см. (см. <_AMODE> ВОСЬМАЯ ПОЗИЦИЯ)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
10		<VARI>	INT	Обработка		
				ПЕРВАЯ ПОЗИЦИЯ:	ломка стружки/удаление стружки	
					0 =	в одном проходе
					1 =	ломка стружки
					2 =	удаление стружки
					3 =	ломка стружки и удаление стружки
				ВТОРАЯ ПОЗИЦИЯ:	Отвод при удалении стружки	
					0 =	на глубину базового отверстия
					1 =	на безопасное расстояние
				ТРЕТЬЯ ПОЗИЦИЯ:	Мягкое врезание	
					0 =	нет
					1 =	да
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	Сквозное сверление	
					0 =	нет
					1 =	да
				ПЯТАЯ ПОЗИЦИЯ:	Засверливание/базовое отверстие	
					0 =	без засверливания
					1 =	с засверливанием
					2 =	с базовым отверстием
				ШЕСТАЯ ПОЗИЦИЯ:	Отвод	
					0 =	на глубину базового отверстия
					1 =	на плоскость отвода
11	V1	<_MDEP>	REAL	Мин. подача на глубину инкрементальная (только при проценте для дегрессии)		
12	V2	<_VRT>	REAL	Величина отвода после каждой обработки инкрементальная (только при ломке стружки)		
					0 =	стандартное значение 1 мм
					> 0 =	переменная величина отвода
13	DT	<_DTD>	REAL	Время ожидания на конечной глубине сверления, см. <_AMODE> ЧЕТВЕРТАЯ ПОЗИЦИЯ)		
14	V3	<_DIS1>	REAL	Упреждающий зазор инкрементальный, только при удалении стружки (см. <_AMODE> СЕДЬМАЯ ПОЗИЦИЯ)		
15	FP	<S_FP>	REAL	Подача для ввода в базовое отверстие, как значение или в % (в комбинации с <S_AMODE2> ТРЕТЬЯ ПОЗИЦИЯ)		
16		<S_SDAC2>	INT	Направление вращения шпинделя при подводе		
					3 =	M3
					4 =	M4
					5 =	M5 (по умолчанию)
17	SP	<S_SV2>	REAL	Подвод с	постоянной скоростью шпинделя (см. <S_AMODE2> ВОСЬМАЯ ПОЗИЦИЯ)	
	V4				постоянной скоростью резания	
				Скорость шпинделя в % от скорости сверления		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
18	F	<S_FB>	REAL	Подача сверления (см. <S_AMODE2> ПЕРВАЯ ПОЗИЦИЯ)		
19		<_SDAC>	REAL	Направление вращения шпинделя при сверлении		
					3 =	M3
					4 =	M4
20	S	<_SV1>	REAL	Сверление с	постоянной скоростью шпинделя (см. <S_AMODE2> СЕДЬМАЯ ПОЗИЦИЯ)	
	V5				постоянной скоростью резания	
21	SPOS	<S_SPOS>	REAL	Позиция шпинделя, только при подводе с M5		
22	ZA	<S_ZA>	REAL	Глубина засверливания инкрементальная относительно исходной точки или абсолютная (см. <S_AMODE3> ПЕРВАЯ ПОЗИЦИЯ)		
23	FA	<S_FA>	REAL	Подача засверливания как значение или в % (в комбинации с <S_AMODE2> ВТОРАЯ ПОЗИЦИЯ)		
24	ZP	<S_ZP>	REAL	Базовое отверстие инкрементальное относительно исходной точки, абсолютное или как коэффициент диаметра сверла (см. <S_AMODE3> ВТОРАЯ ПОЗИЦИЯ)		
25	FS	<S_FS>	REAL	Подача врезания как значение или в % (в комбинации с <S_AMODE2> ЧЕТВЕРТАЯ ПОЗИЦИЯ)		
26	ZS1	<S_ZS1>	REAL	Глубина каждого врезания с постоянной подачей (инкр.)		
27	ZS2	<S_ZS2>	REAL	Глубина каждого врезания для увеличения подачи (инкр.)		
28	n	<S_N>	INT	Число ходов стружколомания перед каждым удалением стружки		
29	ZD	<S_ZD>	REAL	Остаточная глубина сверления инкрементальная относительно конечной глубины сверления или абсолютная (в комбинации с <S_AMODE3> ТРЕТЬЯ ПОЗИЦИЯ)		
30	FD	<S_FD>	REAL	Остаточная подача сверления как значение или в % (в комбинации с <S_AMODE2> ПЯТАЯ ПОЗИЦИЯ)		
31	FR	<S_FR>	REAL	Подача при отводе (в комбинации с <S_AMODE2> ШЕСТАЯ ПОЗИЦИЯ)		
32		<S_SDAC3>	INT	Направление вращения шпинделя при отводе		
					3 =	M3
					4 =	M4
					5 =	M5
33	SR	<S_SV3>	REAL	Отвод с	постоянной скоростью шпинделя (см. <S_AMODE2> ДЕВЯТАЯ ПОЗИЦИЯ)	
	V6				постоянной скоростью резания	
				Скорость шпинделя в % от скорости сверления		
34	СОЖ вкл	<S_CON>	STRING[10]	СОЖ вкл, M-команда или вызов подпрограммы		
35	СОЖ выкл	<S_COFF>	STRING[10]	СОЖ выкл, M-команда или вызов подпрограммы		
36		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	глубина сверления относительно острия/хвостовика	
					0 =	Острие
					1 =	Хвостовик

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
37		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	зарезервировано	
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
38		<_AMODE>	INT	Альтернативный режим 1		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления = конечная глубина сверления Z1 абс./инкр.	
					0 =	инкрементальная
					1 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	время ожидания на каждой глубине сверления DTB в секундах/оборотах	
					0 =	в секундах
					1 =	в оборотах
				ТРЕТЬЯ ПОЗИЦИЯ:	Время ожидания для удаления стружки DTS в секундах/оборотах	
					0 =	в секундах
					1 =	в оборотах
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	время ожидания на конечной глубине сверления DT в секундах/оборотах	
					0 =	в секундах
					1 =	в оборотах
				ПЯТАЯ ПОЗИЦИЯ:	1. Глубина сверления D (абс./инкр.)	
					0 =	инкрементальная
					1 =	абсолютная
				ШЕСТАЯ ПОЗИЦИЯ:	величина/процент DF для каждой следующей подачи (дегрессия)	
					0 =	Значение
					1 =	процент (от 0,001 до 100 %)
				СЕДЬМАЯ ПОЗИЦИЯ:	упреждающий зазор V3 автоматический/ручной	
					0 =	автоматический (вычисляется в цикле)
					1 =	ручной (запрограммированное значение)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
39		<S_AMODE2>	INT	Альтернативный режим 2		
				ПЕРВАЯ ПОЗИЦИЯ:	ПЕРВАЯ ПОЗИЦИЯ: Подача сверления F	
					0 =	F/мин
					1 =	F/об
				ВТОРАЯ ПОЗИЦИЯ:	нормирование подачи засверливания FA	
					0 =	в % от подачи сверления
					1 =	F/мин
					2 =	F/об
				ТРЕТЬЯ ПОЗИЦИЯ:	нормирование подачи для ввода в базовое отверстие FP	
					0 =	в % от подачи сверления
					1 =	F/мин
					2 =	F/об
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	нормирование подачи врезания FS	
					0 =	в % от подачи сверления
					1 =	F/мин
					2 =	F/об
				ПЯТАЯ ПОЗИЦИЯ:	нормирование подачи сквозного сверления FD	
					0 =	в % от подачи сверления
					1 =	F/мин
					2 =	F/об
				ШЕСТАЯ ПОЗИЦИЯ:	Подача при отводе FR	
					0 =	F/мин
					1 =	Ускоренный ход
				СЕДЬМАЯ ПОЗИЦИЯ:	сверление - скорость шпинделя/ скорость резания (S/V5)	
					0 =	постоянная скорость шпинделя
					1 =	постоянная скорость резания
				ВОСЬМАЯ ПОЗИЦИЯ:	подвод со скорость шпинделя/скоростью резания (SP/V4)	
					0 =	постоянная скорость шпинделя
					1 =	постоянная скорость резания
					2 =	скорость шпинделя в % от скорости сверления
				ДЕВЯТАЯ ПОЗИЦИЯ:	отвод - скорость шпинделя/скорость резания (SR/V6)	
					0 =	постоянная скорость шпинделя
					1 =	постоянная скорость резания
					2 =	скорость шпинделя в % от скорости сверления

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
40		<S_AMODE3>	INT	Альтернативный режим 3		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина засверливания ZA абс./инкр.	
					0 =	инкрементальная
					1 =	абсолютная
				ВТОРАЯ ПОЗИЦИЯ:	глубина базового отверстия ZP	
					0 =	инкрементальная
					1 =	абсолютная
					2 =	коэффициент диаметра сверла
				ТРЕТЬЯ ПОЗИЦИЯ:	остаточная глубина сверления ZD инкр./абс.	
					0 =	инкрементальная
					1 =	абсолютная
41	ZPV	<S_ZPV>	REAL	Упреждающий зазор, инкрементальный, на глубину базового отверстия		

4.26.1.39 CYCLE832 - High Speed Settings

Синтаксис

```
CYCLE832(<S_TOL>, <S_TOLM>, <S_OTOL>)
```

Примечание

CYCLE832 не освобождает изготовителя станка от необходимости выполнения задач по оптимизации при вводе станка в эксплуатацию. Это относится к оптимизации участвующих в обработке осей и настройкам NCU (предуправление, ограничение рывка и т.д.).

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1	Допуск	<S_TOL>	REAL	Допуск контура Допуск контура соответствует осевому допуску гео-осей.		
2		<S_TOLM>	INT	Режим обработки (технология)		
				ПЕРВАЯ ПОЗИЦИЯ:		
					0 =	Отмена
					1 =	чистовая обработка (Finish)
					2 =	получистовая обработка (Semifinish)
					3 =	черновая обработка (Rough)
					4 =	Точная чистовая обработка (Precision)
				ВТОРАЯ ПОЗИЦИЯ:		
					0 =	совместимость[1] или нет допуска ориентации
					1 =	Допуск ориентации в параметре <S_OTOL>
				ТРЕТЬЯ ... СТОТЫСЯЧНАЯ	занято по причине Совместимость	
				СЕДЬМАЯ ПОЗИЦИЯ:		
					0 =	Совместимость. Автоматически используется наилучшая из доступных функций изготовления пресс-форм: • Опция Top Surface не активна: ⇒ Advanced Surface • Опция Top Surface активна: ⇒ Top Surface со сглаживанием
					1 =	Top Surface без сглаживания
					2 =	Top Surface со сглаживанием
3	ORI-допуск	<S_OTOL>	REAL	Допуск ориентации или обозначение версии CYCLE832 Параметр допуска для ориентации инструмента. Необходим для выполнения высокоскоростной программы обработки на станках с динамической трансформацией ориентации (например, 5-осевая обработка). Параметр <S_OTOL> **нужно** запрограммировать. Это же относится и к приложениям на 3-осевых станках для программ без ориентации инструмента (<S_OTOL> = 1).		

[1] Допуск ориентации берется из установочных данных циклов SD55451 ... SD55454 (допуск ориентации для динамического режима ...) или SD55445 ... SD55449 (допуск контура для динамического режима ...), умножается на коэффициент из SD55441 ... SD55444.
Дополнительная информация: Руководство по вводу в эксплуатацию SINUMERIK Operate

Ввод открытым текстом

Для улучшения читабельности вызова цикла, параметр <S_TOLM> (Режим обработки) может быть введен и открытым текстом. Открытые тексты не имеют языковой зависимости. Могут вводиться следующие данные:

_OFF	Для	0	Отмена
_FINISH	Для	1	Чистовая обработка
_SEMIFIN	Для	2	Предварительная чистовая обработка
_ROUGH	Для	3	Черновая обработка
_PRECISION	Для	4	Точная чистовая обработка
_ORI_FINISH	Для	11	Получистовая обработка с вводом допуска ориентации
_ORI_SEMIFIN	Для	12	Предварительная чистовая обработка с вводом допуска ориентации
_ORI_ROUGH	Для	13	Черновая обработка с вводом допуска ориентации
_ORI_PRECISION	Для	14	Чистовая обработка с вводом допуска ориентации
_TOP_SURFACE_SMOOTH_OFF	Для	1000000	Top Surface без сглаживания
_TOP_SURFACE_SMOOTH_ON	Для	2000000	Top Surface со сглаживанием

Пример составления открытого текста, вводимого в Top Surface, приведен ниже:

```
CYCLE832(0.1, _TOP_SURFACE_SMOOTH_OFF+_ORI_FINISH, 1)
```

Примечание

Открытые тексты взяты из имен группы G-функций 59 (динамический режим для интерполяции траектории). Этими открытыми текстами при использовании четко различаются 3-осевые станки и станки с многоосевой трансформацией ориентации (TRAORI).

Сброс CYCLE832

При отмене CYCLE832 параметр <S_TOL> должен передаваться с нулем.

Пример: `CYCLE832(0,0,1)`

Синтаксис `CYCLE832()` также разрешен для отмены CYCLE832.

Примеры

Пример 1: CYCLE832 на 3-осевом станке без трансформации ориентации

a) Вызов цикла с вводом открытого текста

Программный код	Комментарий
G710	; Метрическая система единиц.
CYCLE832(0.004,_FINISH,1)	; Вызов CYCLE832 с: допуск контура = 0,004 мм, режим обработки: чистовая обработка
...	; Выполнение программы высокоскоростной обработки

b) Вызов цикла без ввода открытого текста

Программный код	Комментарий
G710	; см. выше
CYCLE832(0.004,1,1)	; см. выше
...	; см. выше

Пример 2: CYCLE832 на 5-осевом станке с трансформацией ориентации

a) Вызов цикла и отмена с вводом открытого текста

Программный код	Комментарий
G710	; Метрическая система единиц.
TRAORI	; Активировать трансформацию ориентации.
CYCLE832(0.3,_ORI_ROUGH,0.8)	; Вызов CYCLE832 с: допуск контура = 0,3 мм, режим обработки: черновая обработка с вводом допуска ориентации, допуск ориентации = 0,8 градуса
...	; Выполнение программы высокоскоростной обработки
CYCLE832(0,_OFF,1)	; Допуск контура = 0, режим обработки: Отмена CYCLE832, допуск ориентации = 0 градусов

b) Вызов цикла и отмена без ввода открытого текста

Программный код	Комментарий
G710	; см. выше
TRAORI	; см. выше
CYCLE832(0.3,13,0.8)	; см. выше
...	; см. выше
CYCLE832(0,0,1)	; см. выше

4.26.1.40 CYCLE840 - Нарезание внутренней резьбы с компенсирующим патроном

Синтаксис

```
CYCLE840(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDR>, <SDAC>,
<ENC>, <MPIT>, <PIT>, <_AXN>, <_PITA>, <_TECHNO>, <_PITM>, <_PTAB>,
<_PTABA>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение			
1	RP	<RTP>	REAL	Плоскость отвода (абс.)			
2	Z0	<RFP>	REAL	Исходная точка (абс.)			
3	sc	<SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)			
4	Z1	<DP>	REAL	Глубина сверления (абс.), см. <_AMODE>			
5	Z1	<DPR>	REAL	Глубина сверления (инкр.), см. <_AMODE>			
6	DT	<DTB>	REAL	Время ожидания на глубине сверления/на безопасном расстоянии после отвода в секундах, см. <ENC>			
7		<SDR>	INT	Направление вращения для отвода			
8	SDE	<SDAC>	INT	Направление вращения после завершения цикла			
9		<ENC>	INT	Нарезание внутренней резьбы с датчиком шпинделя (G33)/нарезание внутренней резьбы без датчика шпинделя (G63)			
					0 =	с датчиком шпинделя	- шаг из <MPIT>/<PIT> - без DT
					20 =	с датчиком шпинделя	- шаг из <MPIT>/<PIT> - с DT после отвода на безопасное расстояние
					11 =	без датчика шпинделя	- шаг из <MPIT>/<PIT> - с DT на глубине сверления
					1 =	без датчика шпинделя	- шаг из запрограммированной подачи - с DT на глубине сверления (подача = скорость · шаг)
10		<MPIT>	REAL	Размер резьбы только для "ISO метрической" (шаг вычисляется при внутренней обработке) Диапазон значений: 3 до 48 (для M3 до M48), как альтернатива <PIT>			
11		<PIT>	REAL	Шаг резьбы как значение, единицу см. <_PITA> Диапазон значений: > 0, как альтернатива MPIT			

№	Параметр в маске	Параметр внутренний	Тип данных	Значение			
12		<_AXN>	INT	Ось сверления	0 =	3-я гео-ось	
					1 =	1-я гео-ось	
					2 =	2-я гео-ось	
					≥ 3 =	3-я гео-ось	
13		<_PITA>	INT	Единица измерения шага резьбы (обработка <PIT> и <MPIT>)			
					0 =	шаг в мм	- обработка <MPIT>/<PIT>
					1 =	шаг в мм	- обработка <PIT>
					2 =	шаг в TPI	- обработка <PIT> (витков резьбы на дюйм)
					3 =	шаг в дюймах	- обработка <PIT>
					4 =	MODUL	- обработка <PIT>
14		<_TECHNO>	INT	Технология[1]			
				ПЕРВАЯ ПОЗИЦИЯ:	параметры точного останова		
					0 =	активны параметры точного останова как перед вызовом цикла	
					1 =	точный останов G601	
					2 =	точный останов G602	
					3 =	точный останов G603	
				ВТОРАЯ ПОЗИЦИЯ:	предуправление		
					0 =	активно с/без предуправления как перед вызовом цикла	
					1 =	с предуправлением FFWON	
					2 =	без предуправления FFWOF	
15		<_PITM>	STRING[15]	Строка как меркер для ввода шага резьбы[2]			
16		<_PTAB>	STRING[5]	Строка для таблицы резьб ("", "ISO", "BSW", "BSP", "UNC")[2]			
17		<_PTABA>	STRING[20]	Строка для выбора в таблице резьб (к примеру, "M 10", "M 12", ...)[2]			
18		<_GMODE>	INT	зарезервировано			

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
19		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	режим совместимости (только для маски ввода для обратного перевода), если MD 52216 Бит0 = 1[1)]	
					0 =	технологические параметры отображаются (совместимость): действуют TECHNO-параметры
					1 =	технологические параметры не отображаются: действует технология "как запрограммировано перед вызовом цикла"
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой
20		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина сверления Z1 (абс./инкр.)	
					0 =	совместимость, из программирования <DP>/<DPR>
					1 =	инкрементальная
					2 =	абсолютная

[1)] Поля технологии могут быть скрыты в зависимости от установочных данных SD52216 MCS_FUNCTION_MASK_DRILL

[2)] Параметры 15, 16 и 17 используются только при выборе резьбы в таблицах резьб маски ввода. Доступ к таблицам резьб через определение цикла при выполнении цикла невозможен!

4.26.1.41 CYCLE899 – открытый паз

Синтаксис

```
CYCLE899(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_LENG>, <_WID>, <_PA>,
<_PO>, <_STA>, <_MID>, <_MIDA>, <_FAL>, <_FALD>, <_FFP1>, <_CDIR>,
<_VARI>, <_GMODE>, <_DMODE>, <_AMODE>, <_UMODE>, <_FS>, <_ZFS>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
1	RP	<_RTP>	REAL	Плоскость отвода (абс.)		
2	Z0	<_RFP>	REAL	Исходная точка оси инструмента (абс.)		
3	sc	<_SDIS>	REAL	Безопасное расстояние (вводится аддитивно к исходной точке, без знака)		
4	Z1	<_DP>	REAL	Глубина паза (абс./инкр.), см.<_AMODE>		
5	l	<_LENG>	REAL	Длина канавки (инкр.)		
6	W	<_WID>	REAL	Ширина канавки (инкр.)		
7	X0	<_PA>	REAL	Исходная точка/стартовая позиция 1-й оси (абс.)		
8	Y0	<_PO>	REAL	Исходная точка/стартовая позиция 2-й оси (абс.)		
9	α0	<_STA>	REAL	Угол поворота к 1-й оси		
10	DZ	<_MID>	REAL	Макс. глубина подачи (инкр.), только для вихревого фрезерования		
11	DXY	<_MIDA>	REAL	Макс. подача в плоскости, см. <_AMODE>		
12	UXY	<_FAL>	REAL	Чистовой припуск плоскость		
13	UZ	<_FALD>	REAL	Чистовой припуск глубина		
14	F	<_FFP1>	REAL	Подача		
15		<_CDIR>	INT	Направление фрезерования		
				ПЕРВАЯ ПОЗИЦИЯ:		
					0 =	Синхронный ход
					1 =	Противоход
					4 =	Попеременно
16		<_VARI>	INT	Обработка		
				ПЕРВАЯ ПОЗИЦИЯ:		
					1 =	Черновая обработка
					2 =	Чистовая обработка
					3 =	Чистовая обработка основания
					4 =	Чистовая обработка края
					5 =	Предварительная чистовая обработка
					6 =	Снятие фаски
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:		
					1 =	Вихревое фрезерование
					2 =	Врезное фрезерование

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
17		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки/только расчет стартовой точки	
					1 =	обычная обработка
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	указание размеров через центр / кромку	
					0 =	указание размеров через центр
					1 =	указание размеров через кромку "слева" ("-" направление 1-й оси)
					2 =	указание размеров через кромку "справа" ("+" направление 1-й оси)
18		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	---	зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ:	---	зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	Ввод: полный
					1 =	Ввод: простой
19		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	глубина канавки Z1	
					0 =	абсолютная
					1 =	инкрементальная
				ВТОРАЯ ПОЗИЦИЯ:	единица для подачи в плоскости (<_MIDA>) DXY	
					0 =	мм
					1 =	% от диаметра инструмента
				ТРЕТЬЯ ПОЗИЦИЯ:	глубина врезания при снятии фаски ZFS	
					0 =	абсолютная
					1 =	инкрементальная
20		<_UMODE>	INT	зарезервировано		

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение
21	FS	<_FS>	REAL	Ширина фаски (инкр.)
22	ZFS	<_ZFS>	REAL	Глубина врезания (острие инструмента) при снятии фаски (абс./инкр.) см. <_AMODE>)

4.26.1.42 CYCLE930 - выточка

Синтаксис

```
CYCLE930(<_SPD>, <_SPL>, <_WIDG>, <_WIDG2>, <_DIAG>, <_DIAG2>,
<_STA>, <_ANG1>, <_ANG2>, <_RCO1>, <_RCI1>, <_RCI2>, <_RCO2>,
<_FAL>, <_IDEP1>, <_SDIS>, <_VARI>, <_DN>, <_NUM>, <_DBH>, <_FF1>,
<_NR>, <_FALX>, <_FALZ>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	X0	<_SPD>	REAL	Исходная точка в поперечной оси (всегда диаметр)
2	Z0	<_SPL>	REAL	Исходная точка в продольной оси
3	B1	<_WIDG>	REAL	Ширина выточки внизу
4	B2	<_WIDG2>	REAL	Ширина выточки вверху (только для интерфейса)
5	T1	<_DIAG>	REAL	Глубина выточки в исходной точке, при абс. и продольной обработке = диаметр, в остальных случаях инкр.
6	T2	<_DIAG2>	REAL	Глубина выточки напротив исходной точки (только для интерфейса), при абс. и продольной обработке = диаметр, в остальных случаях инкр.
7	α0	<_STA>	REAL	Угол наклона (-180 ≤ <_STA> ≤ 180)
8	α1	<_ANG1>	REAL	Угол профиля 1 (0 ≤ <_ANG1> < 90) на определенной через исходную точку стороне выточки
9	α2	<_ANG2>	REAL	Угол профиля 2 (0 ≤ <_ANG2> < 90) напротив исходной точки
10	R1/FS1	<_RCO1>	REAL	Радиус закругления или ширина фаски 1, снаружи на исходной точке
11	R2/FS2	<_RCI1>	REAL	Радиус закругления или ширина фаски 2, внутри на исходной точке
12	R3/FS3	<_RCI2>	REAL	Радиус закругления или ширина фаски 3, внутри напротив исходной точки
13	R4/FS4	<_RCO2>	REAL	Радиус закругления или ширина фаски 4, снаружи напротив исходной точки
14	U	<_FAL>	REAL	Чистовой припуск в X и Z, см. <_VARI> (ПЯТАЯ ПОЗИЦИЯ) (вводится без знака)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
15	D	<_IDEP1>	REAL	Макс. подача на глубину при врезании (вводится без знака)		
					0 =	1-й проход резца прямо на всю глубину
					> 0 =	1-й проход резца <_IDEP1>, 2-й проход 2 · <_IDEP1> и т.д.
16	sc	<_SDIS>	REAL	Безопасное расстояние (вводится без знака)		
17		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	технологическая обработка	
					1 =	черновая обработка
					2 =	чистовая обработка
					3 =	черновая и чистовая обработка
				ТРЕТЬЯ ПОЗИЦИЯ:	положение продольное/поперечное снаружи/внутри +Z/+Z или +X/-X	
					1 =	продольное/снаружи +Z
					2 =	поперечное/внутри -X
					3 =	продольное/внутри +Z
					4 =	поперечное/внутри +X
					5 =	продольное/снаружи -Z
					6 =	поперечное/снаружи -X
					7 =	продольное/внутри -Z
					8 =	поперечное/снаружи +X
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	положение исходной точки	
					0 =	исходная точка сверху
					1 =	исходная точка снизу
				ПЯТАЯ ПОЗИЦИЯ:	определение действия чистовых припусков	
					0 =	чистовой припуск U параллельно контуру
					1 =	чистовые припуски UX и UZ отдельно
18		<_DN>	INT	D-номер для 2-го резца инструмента		
					> 0 =	D-номер для коррекции на инструмент 2-го резца прорезного инструмента
					0 =	2-й резец не запрограммирован
19	N	<_NUM>	INT	Число выточек (0 = 1 выточка)		
20	DP	<_DBH>	REAL	Расстояние между выточками (необходимо, только если <_NUM> > 1)		
21	F	<_FF1>	REAL	Подача		

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
22		<_NR>	INT	Идентификатор для формы выточки соответствует вертикальным программным клавишам для выбора формы		
					0 =	боковые поверхности 90° без фасок/закруглений
					1 =	наклонные боковое стороны с фасками/закруглениями (без α0)
					2 =	как 1, но на конусе (с α0)
23	UX	<_FALX>	REAL	Чистовой припуск в оси X, см. <_VARI> (ПЯТАЯ ПОЗИЦИЯ) (вводится без знака)		
24	UZ	<_FALZ>	REAL	Чистовой припуск в оси Z, см. <_VARI> (ПЯТАЯ ПОЗИЦИЯ) (вводится без знака)		
25		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
26		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	указание размеров Глубина (только для интерфейса)	
					0 =	на исходной точке
					1 =	напротив исходной точки
				ВТОРАЯ ПОЗИЦИЯ:	глубина	
					0 =	абсолютная
					1 =	инкрементальная
				ТРЕТЬЯ ПОЗИЦИЯ:	указание размеров Ширина (только для интерфейса)	
					0 =	на наружном диаметре (сверху)
					1 =	на внутреннем диаметре (снизу)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	радиус/фаска 1 (<_RCO1>)	
					0 =	радиус
					1 =	фаска
				ПЯТАЯ ПОЗИЦИЯ:	радиус/фаска 2 (<_RCI1>)	
					0 =	радиус
					1 =	фаска
				ШЕСТАЯ ПОЗИЦИЯ:	радиус/фаска 3 (<_RCI2>)	
					0 =	радиус
					1 =	фаска
				СЕДЬМАЯ ПОЗИЦИЯ:	радиус/фаска 4 (<_RCO2>)	
					0 =	радиус
					1 =	фаска

4.26.1.43 CYCLE940 – Канавка формы E и F, резьбовая канавка

С помощью цикла CYCLE940 можно запрограммировать различные канавки. Частично они очень сильно различаются в параметрировании.

Дополнительные колонки в таблице показывают, какой параметр для какой формы канавки необходим. Они соответствуют вертикальным программным клавишам выбора в маске цикла:

- E: канавка формы E
- F: канавка формы F
- A-D: резьбовая канавка DIN (формы A-D)
- t: резьбовая канавка (свободное определение формы)

Синтаксис

```
CYCLE940(<_SPD>, <_SPL>, <_FORM>, <_LAGE>, <_SDIS>, <_FFP>, <_VARI>,
<_EPD>, <_EPL>, <_R1>, <_R2>, <_STA>, <_VRT>, <_MID>, <_FAL>,
<_FALX>, <_FALZ>, <_PITI>, <_PTAB>, <_PTABA>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Прогр. для формы				Объяснение		
				E	F	A-D	T			
1	X0	<_SPD>	REAL	x	x	x	x	Исходная точка в поперечной оси (всегда диаметр)		
2	Z0	<_SPL>	REAL	x	x	x	x	Исходная точка продольной оси (абс.)		
3	FORM	<_FORM>	CHAR	x	x	x	x	Форма канавки (прописные буквы, например, "T") Выбор, из какой таблицы должны быть взяты значения канавки		
									A =	снаружи, источник DIN76, A = обычная
									B =	снаружи, источник DIN76, B = короткая
									C =	внутри, источник DIN76, C = обычная
									D =	внутри, источник DIN76, D = короткая
									E =	источник DIN509
									f =	источник DIN509
									T=	свободная форма
4	LAGE	<_LAGE>	INT	x	x	x	x	Положение канавки (параллельно Z)	0 =	снаружи +Z: ____|
									1 =	снаружи -Z: |____/
									2 =	внутри +Z: /-----|
									3 =	внутри -Z: |-----\
5	sc	<_SDIS>		x	x	x	x	Безопасное расстояние (инкр.)		
6	F	<_FFP>		x	x	x	x	Подача обработки (мм/об)		
7		<_VARI>	INT	-	-	x	x	Режим обработки		
								ПЕРВАЯ ПОЗИЦИЯ:	обработка	
									1 =	черновая обработка
									2 =	чистовая обработка
									3 =	черновая + чистовая
								ВТОРАЯ ПОЗИЦИЯ:	стратегия обработки	
									0 =	параллельно контуру
									1 =	продольная
				Канавки формы E и F всегда обрабатываются за один проход как при чистовой обработке.						
8	X1	<_EPD>		x	x	-	-	Припуск X (абс./инкр.), см. <_AMODE>		
				-	-	-	x	Глубина канавки (абс./инкр.), см. <_AMODE>		

№	Параметр в маске	Параметр внутренний	Тип данных	Прогр. для формы				Объяснение			
9	Z1	<_EPL>		-	x	-	-	Припуск Z			
				-	-	-	x	Ширина канавки (абс./инкр.), см. <_AMODE>			
10	R1	<_R1>		-	-	-	x	Радиус закругления на наклоне			
11	R2	<_R2>		-	-	-	x	Радиус закругления в углу			
12	α	<_STA>		-	-	x	x	Угол врезания			
13	VX	<_VRT>		x	x	-	-	Поперечная подача X (абс./инкр.), см. <_AMODE>			
				-	-	x	x	Поперечная подача X при чистовой обработке, (абс./инкр.), см. <_AMODE>			
14	D	<_MID>		-	-	x	x	Подача на глубину			
15	U	<_FAL>		-	-	x	x	Чистовой припуск параллельно контуру, см. <_AMODE>			
16	UX	<_FALX>		-	-	x	x	Чистовой припуск x			
17	UZ	<_FALZ>		-	-	x	x	Чистовой припуск Z			
18	P	<_PITI>	INT	-	-	x	-	Выбор шага, форма A-D, соответствует M1 ... M68			
								0 = 0.20 1 = 0.25 2 = 0.30 3 = 0.35 4 = 0.40 5 = 0.45	6 = 0.50 7 = 0.60 8 = 0.70 9 = 0.75 10 = 0.80 11 = 1.00	12 = 1.25 13 = 1.50 14 = 1.75 15 = 2.00 16 = 2.50 17 = 3.00	18 = 3.50 19 = 4.00 20 = 4.50 21 = 5.00 22 = 5.50 23 = 6.00
				x	x	-	-	Выбор радиуса/глубины, форма E, F			
								0 = 0.6 · 0.3 1 = 1.0 · 0.4 2 = 1.0 · 0.2 3 = 1.6 ··0.3	4 = 2.5 · 0.4 5 = 4.0 · 0.5 6 = 0.4 · 0.2 7 = 0.6 · 0.2	8 = 0.1 · 0.1 9 = 0.2 ··0.1	
19		<_PTAB>	STRING [5]					Строка для таблицы резьб ("", "ISO", "BSW", "BSP", "UNC") (только для интерфейса)			
20		<_PTABA>	STRING [20]					Строка выбора в таблице значений резьбы (например, "M 10", "M 12", ...) (только для поверхности)			
21		<_DMODE>	INT					Режим индикации			
				x	x	x	x	ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19		
									0 =	совместимость, остается активной действующая перед вызовом цикла плоскость	
									1 =	G17 (активна только в цикле)	
									2 =	G18 (активна только в цикле)	
									3 =	G19 (активна только в цикле)	

№	Параметр в маске	Параметр внутренний	Тип данных	Прогр. для формы				Объяснение		
22		<_AMODE>	INT					Альтернативный режим		
				x	x	-	x	ПЕРВАЯ ПОЗИЦИЯ:	Параметр <_EPD> припуск X или глубина канавки	
									0 =	абсолютная (всегда диаметр)
									1 =	инкрементальная
				x	x	-	x	ВТОРАЯ ПОЗИЦИЯ:	Параметр <_EPL> припуск Z или ширина канавки	
									0 =	абсолютная
									1 =	инкрементальная
				x	x	x	x	ТРЕТЬЯ ПОЗИЦИЯ:	Параметр <_VRT> поперечная подача X	
									0 =	абсолютная (всегда диаметр)
									1 =	инкрементальная
				-	-	x	x	ЧЕТВЕРТАЯ ПОЗИЦИЯ:	чистовой припуск	
									0 =	чистовой припуск параллельно контуру, см. <_FAL>
									1 =	чистовой припуск отдельно (<_FALX>/<_FALZ>)

4.26.1.44 CYCLE951 - обработка резанием

Синтаксис

```
CYCLE951(<_SPD>, <_SPL>, <_EPD>, <_EPL>, <_ZPD>, <_ZPL>, <_LAGE>,
<_MID>, <_FALX>, <_FALZ>, <_VARI>, <_RF1>, <_RF2>, <_RF3>, <_SDIS>,
<_FF1>, <_NR>, <_DMODE>, <_AMODE>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Значение
1	X0	<_SPD>	REAL	Исходная точка (абс., всегда диаметр)
2	Z0	<_SPL>	REAL	Исходная точка (абс.)
3	X1	<_EPD>	REAL	Конечная точка
4	Z1	<_EPL>	REAL	Конечная точка
5	XM α1 α2	<_ZPD>	REAL	Промежуточная точка, см. <_DMODE>(ВТОРАЯ ПОЗИЦИЯ)
6	ZM α1 α2	<_ZPL>	REAL	Промежуточная точка, см. <_DMODE>(ВТОРАЯ ПОЗИЦИЯ)

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
7	Положение	<_LAGE>	INT	Положение обрабатываемого угла	0 =	снаружи/сзади
					1 =	снаружи/спереди
					2 =	внутри/сзади
					3 =	внутри/спереди
8	D	<_MID>	REAL	Макс. подача на глубину при врезании		
9	UX	<_FALX>	REAL	Чистовой припуск в X		
10	UZ	<_FALZ>	REAL	Чистовой припуск в Z		
11		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	направление резания (продольное или поперечное) в системе координат	
					1 =	продольная
					2 =	поперечная
				ВТОРАЯ ПОЗИЦИЯ:		
					1 =	черновая обработка до чистового припуска
					2 =	чистовая обработка
				ТРЕТЬЯ ПОЗИЦИЯ:	зарезервировано	
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	зарезервировано	
				ПЯТАЯ ПОЗИЦИЯ:	Зарезервировано	
12	R1/FS1	<_RF1>	REAL	Радиус закругления или ширина фаски 1, см. <_AMODE>(ПЯТАЯ ПОЗИЦИЯ)		
13	R2/FS2	<_RF2>	REAL	Радиус закругления или ширина фаски 2, см. <_AMODE>(ШЕСТАЯ ПОЗИЦИЯ)		
14	R3/FS3	<_RF3>	REAL	Радиус закругления или ширина фаски 3, см. <_AMODE> (СЕДЬМАЯ ПОЗИЦИЯ)		
15	sc	<_SDIS>	REAL	Безопасное расстояние		
16	F	<_FF1>	REAL	Подача для черновой/чистовой обработки		
17		<_NR>	INT	Идентификатор для типа резания (соответствует вертикальным программным клавишам для выбора формы):		
					0 =	обработка резаньем 1, угол 90 градусов без фасок/закруглений
					1 =	обработка резаньем 2, угол 90 градусов с фасками/закруглениями
					2 =	обработка резаньем 3, любой угол с фасками/закруглениями

№	Параметр в маске	Параметр внутренний	Тип данных	Значение		
18		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	форма ввода <_ZPD>/<_ZPL>	
					0 =	Xm/Zm
					1 =	Xm/α1
					2 =	Xm/α2
					3 =	α1/Zm
					4 =	α2/Zm
					5 =	α1/α2
21		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	промежуточная точка в X	
					0 =	абсолютная, значение поперечной оси в диаметре
					1 =	инкрементальная, значение поперечной оси в радиусе
				ВТОРАЯ ПОЗИЦИЯ:	промежуточная точка в Z	
					0 =	абсолютная
					1 =	инкрементальная
				ТРЕТЬЯ ПОЗИЦИЯ:	конечная точка в X	
					0 =	абсолютная, значение поперечной оси в диаметре
					1 =	инкрементальная, значение поперечной оси в радиусе
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	конечная точка в Z	
					0 =	абсолютная
					1 =	инкрементальная
				ПЯТАЯ ПОЗИЦИЯ:	радиус/фаска 1	
					0 =	радиус
					1 =	фаска
				ШЕСТАЯ ПОЗИЦИЯ:	радиус/фаска 2	
					0 =	радиус
					1 =	фаска
				СЕДЬМАЯ ПОЗИЦИЯ:	радиус/фаска 3	
					0 =	радиус
					1 =	фаска

4.26.1.45 CYCLE952 – Обработка резаньем / Обработка резаньем – остаток / Прорез / Прорез – остаток / Токарная выточка / Токарная выточка – остаток

Синтаксис

```
CYCLE952(<_PRG>, <_CON>, <_CONR>, <_VARI>, <_F>, <_FR>, <_RP>, <_D>,
<_DX>, <_DZ>, <_UX>, <_UZ>, <_U>, <_U1>, <_BL>, <_XD>, <_ZD>, <_XA>,
<_ZA>, <_XB>, <_ZB>, <_XDA>, <_XDB>, <_N>, <_DP>, <_DI>, <_SC>,
<_DN>, <_GMODE>, <_DMODE>, <_AMODE>, <_PK>, <_DCH>, <_FS>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение
1	PRG	<_PRG>	STRING[100]	Имя программы обработки резаньем
2	CON	<_CON>	STRING[100]	Имя программы, из которой считывается обновленный контур заготовки (при обработке остатков материала)
3	CONR	<_CONR>	STRING[100]	Имя программы, в которую записывается обновленный контур заготовки (см. <_AMODE>ПЯТАЯ ПОЗИЦИЯ)

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
4		<_VARI>	INT	Режим обработки		
				ПЕРВАЯ ПОЗИЦИЯ:	тип резания	
					1 =	продольная
					2 =	поперечная
					3 =	параллельно контуру
				ВТОРАЯ ПОЗИЦИЯ:	технологическая обработка (см. <_GMODE>ТРЕТЬЯ ПОЗИЦИЯ)	
					1 =	черновая обработка
					2 =	чистовая обработка
					3 =	зарезервировано
					4 =	черновая обработка двухканальная
					5 =	чистовая обработка двухканальная
				ТРЕТЬЯ ПОЗИЦИЯ:	Направление обработки	
					1 =	направление обработки X -
					2 =	направление обработки X +
					3 =	направление обработки Z -
					4 =	направление обработки Z +
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	направление подачи	
					1 =	снаружи X -
					2 =	внутри X +
					3 =	торцевая сторона Z -
					4 =	задняя сторона Z +
				ПЯТАЯ ПОЗИЦИЯ:	определение действия чистовых припусков	
					0 =	чистовые припуски UX и UZ отдельно
					1 =	чистовой припуск U параллельно контуру
				ШЕСТАЯ ПОЗИЦИЯ:	возврат по контуру	
					0 =	совместимость, автоматический возврат по контуру
					1 =	с возвратом по контуру
					2 =	с прямым возвратом
					3 =	автоматический возврат по контуру
				СЕДЬМАЯ ПОЗИЦИЯ:	поднутрения	
					0 =	место не обрабатывается при прорезе, - остаток и токарная выточка, - остаток
					1 =	обработка поднутрений
					2 =	не обрабатывать поднутрения

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
				ВОСЬМАЯ ПОЗИЦИЯ:	за/перед центром вращения	
					0 =	обработка перед центром вращения
					1 =	зарезервировано
5	F	<_F>	REAL	Подача для черновой/чистовой обработки		
	FZ			Подача, абсцисса, токарная выточка		
6	FR	<_FR>	REAL	Подача для врезания в поднутрения при черновой обработке		
	FX			Подача, ордината, токарная выточка		
7	RP	<_RP>	REAL	Плоскость отвода при внутренней обработке (абс., всегда диаметр)		
8	D	<_D>	REAL	Подача черновой обработки, (см. <_AMODE>ПЕРВАЯ ПОЗИЦИЯ)		
9	Dx	<_DX>	REAL	Подача X (см. <_AMODE>ПЕРВАЯ ПОЗИЦИЯ)		
10	DZ	<_DZ>	REAL	Подача Z (см. <_AMODE>ПЕРВАЯ ПОЗИЦИЯ)		
11	UX	<_UX>	REAL	Чистовой припуск X (см. <_VARI>ПЯТАЯ ПОЗИЦИЯ)		
12	UZ	<_UZ>	REAL	Чистовой припуск Z (см. <_VARI> ПЯТАЯ ПОЗИЦИЯ)		
13	U	<_U>	REAL	Чистовой припуск параллельно контуру, см. <_VARI>ПЯТАЯ ПОЗИЦИЯ)		
14	U1	<_U1>	REAL	Дополнительный чистовой припуск при чистовой обработке (см. <_AMODE>ЧЕТВЕРТАЯ ПОЗИЦИЯ)		
15	BL	<_BL>	INT	Определение заготовки	1 =	цилиндр с припуском
					2 =	припуск к контуру готовой детали
					3 =	контур заготовки указан
16	XD	<_XD>	REAL	Определение заготовки X (см. <_AMODE>ШЕСТАЯ ПОЗИЦИЯ)		
17	ZD	<_ZD>	REAL	Определение заготовки Z (см. <_AMODE> СЕДЬМАЯ ПОЗИЦИЯ)		
18	XA	<_XA>	REAL	Граница 1 X (абс., всегда диаметр)		
19	ZA	<_ZA>	REAL	Граница 1 Z (абс.)		
20	XB	<_XB>	REAL	Граница 2 X (см. <_AMODE> ВОСЬМАЯ ПОЗИЦИЯ)		
21	ZB	<_ZB>	REAL	Граница 2 Z (см. <_AMODE>ДЕВЯТАЯ ПОЗИЦИЯ)		
22	XDA	<_XDA>	REAL	Граница врезания 1 для 1-й позиции врезания на торцовой стороне (абс., всегда диаметр)		
23	XDB	<_XDB>	REAL	Граница врезания 2 для 1-й позиции врезания на торцовой стороне (абс., всегда диаметр)		
24	N	<_N>	INT	Количество выточек		
25	DP	<_DP>	REAL	Интервал выточек	Продольная выточка: параллельно оси Z	
					Поперечная выточка: параллельно оси X	
26	DI	<_DI>	REAL	Интервал для прерывания подачи	0 =	нет прерывания
					> 0 =	с прерыванием
27	sc	<_SC>	REAL	Безопасное расстояние для обхода препятствий, инкрементальное		
28	D2	<_DN>	INT	D-номер для 2-го резца, если не запрограммирован ⇒ D+1		

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
29		<_GMODE>	INT	Геометрический режим (нормирование запрограммированных гео-значений)		
				ПЕРВАЯ ПОЗИЦИЯ:	зарезервировано	
				ВТОРАЯ ПОЗИЦИЯ:	зарезервировано	
				ТРЕТЬЯ ПОЗИЦИЯ:	выбор обработки/только расчет стартовой точки	
					0 =	обычная обработка (режима совместимости не требуется)
					1 =	обычная обработка
					2 =	расчет стартовой позиции - нет обработки (только для вызова из ShopMill/ShopTurn)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	ограничение	
					0 =	нет
					1 =	да
				ПЯТАЯ ПОЗИЦИЯ:	ввести границу 1 X	
					0 =	нет
					1 =	да
				ШЕСТАЯ ПОЗИЦИЯ:	ввести границу 2 X	
					0 =	нет
					1 =	да
				СЕДЬМАЯ ПОЗИЦИЯ:	ввести границу 1 Z	
					0 =	нет
					1 =	да
				ВОСЬМАЯ ПОЗИЦИЯ:	ввести границу 2 Z	
					0 =	нет
					1 =	да

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
30		<_DMODE>	INT	Режим индикации		
				ПЕРВАЯ ПОЗИЦИЯ:	плоскость обработки G17/G18/G19	
					0 =	совместимость, остается активной действующая перед вызовом цикла плоскость
					1 =	G17 (активна только в цикле)
					2 =	G18 (активна только в цикле)
					3 =	G19 (активна только в цикле)
				ВТОРАЯ ПОЗИЦИЯ:	технологический режим	
					1 =	съём припуска вдоль контура
					2 =	прорезание контура
					3 =	токарная выточка
				ТРЕТЬЯ ПОЗИЦИЯ:	обработка остаточного материала	
					0 =	нет
					1 =	да
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	---	зарезервировано
				ПЯТАЯ ПОЗИЦИЯ:	Технологическое масштабирование внутри масок цикла (Страница 1203)	
					0 =	ввод: полный
					1 =	ввод: простой
				ШЕСТАЯ ПОЗИЦИЯ:	Автоматическое имя программы	
					0 =	нет
					1 =	да

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
31		<_AMODE>	INT	Альтернативный режим		
				ПЕРВАЯ ПОЗИЦИЯ:	выбор подачи	
					0 =	подача DX и DZ при типе резания параллельно контуру
					1 =	подача D
				ВТОРАЯ ПОЗИЦИЯ:	стратегия подачи	
					0 =	переменная глубина резания (90 ... 100 %)
					1 =	постоянная глубина резания
				ТРЕТЬЯ ПОЗИЦИЯ:	подрез	
					0 =	равномерно
					1 =	выравнивание по кромкам
				ЧЕТВЕРТАЯ ПОЗИЦИЯ:	Выбор припуска контура U1, двойной чистовой	
					0 =	нет
					1 =	да
				ПЯТАЯ ПОЗИЦИЯ:	выбор обновления заготовки	
					0 =	нет
					1 =	да
				ШЕСТАЯ ПОЗИЦИЯ:	выбор припуска к заготовке XD	
					0 =	абсолютная, значение поперечной оси в диаметре
					1 =	инкрементальная, значение поперечной оси в радиусе
				СЕДЬМАЯ ПОЗИЦИЯ:	выбор припуска к заготовке ZD	
					0 =	абсолютная
					1 =	инкрементальная
				ВОСЬМАЯ ПОЗИЦИЯ:	выбор границы 2 XB	
					0 =	абсолютная, значение поперечной оси в диаметре
					1 =	инкрементальная, значение поперечной оси в радиусе
				ДЕВЯТАЯ ПОЗИЦИЯ:	выбор границы 2 ZB	
					0 =	абсолютная
					1 =	инкрементальная
				ДЕСЯТАЯ ПОЗИЦИЯ:		
					0 =	главный канал
					1 =	добавочный канал
32		<_PK>	INT	Номер смежного канал, если на станке есть более 2 каналов		
33	DCH	<_DCH>	REAL	Сдвиг канала		
34	FS	<_FS>	REAL	Подача чистовой обработки при комплексной обработке		

4.26.1.46 CYCLE4071 - Продольное шлифование с подачей в точке возврата

Синтаксис

```
CYCLE4071(<S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>, <S_H>, <S_A1>,
<S_A2>)
```

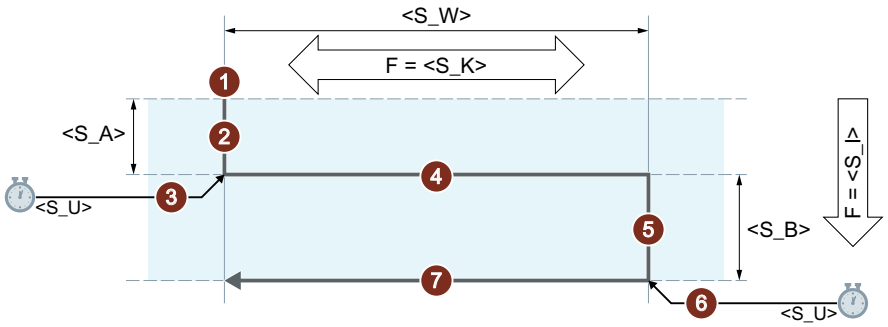
Параметр

№	Параметр	Тип данных	Объяснение
1	<S_A>	REAL	Глубина подачи в начале
2	<S_B>	REAL	Глубина подачи в конце
3	<S_W>	REAL	Ширина шлифования
4	<S_U>	REAL	Время выхаживания
5	<S_I>	REAL	Подача на глубину
6	<S_K>	REAL	Подача для поперечной подачи
7	<S_H>	INT	Число повторов
8	<S_A1>	AXIS	Ось подачи (опция)или 1-я Гео-ось
9	<S_A2>	AXIS	Ось качания (опция) или 2-я Гео-ось

Функция

Цикл служит для обработки повторяющихся подач. При этом глубина подачи в начале и в конце может быть разной. Во время подачи происходит движение по касательной.

Процесс



① Начало цикла в текущей позиции оси качания.
② Перемещение оси подачи на глубину подачи в начале <S_A> с глубиной подачи <S_I>.
③ Выхаживание со временем выхаживания <S_U>.
④ Перемещение оси качания с шириной шлифования <S_W> в качестве пути перемещения и подача для поперечной подачи <S_K>.
⑤ Перемещение оси подачи на глубину подачи в конце <S_B> с глубиной подачи <S_I>.
⑥ Выхаживание со временем выхаживания <S_U>.
⑦ Перемещение качающейся оси с шириной шлифования <S_W> в качестве пути перемещения в начальную точку и подача для поперечной подачи <S_K>.

Обозначает повторяющиеся этапы процессов.

Процесс повторяется до тех пор, пока не будет достигнуто запрограммированное количество повторов <S_H>.

Примечание

Прервать этот процесс отдельным кадром невозможно.

Пример

Выполнение двух маятниковых движений со следующими параметрами цикла:

- Начальная глубина подачи: 0,02 мм
- Конечная глубина подачи: 0,01 мм
- Ход: 100 мм
- Время выхаживания: 1 с
- Скорость подачи: 1 мм/мин
- Поперечная подача: 1000 мм/мин
- Повторы: 2
- Ось качания и ось подачи: стандартные гео-оси

Программный код

```
N10 T1 D1
```

Программный код

```
N20 CYCLE4071(0.02,0.01,100,1,1,1000,2)
N30 M30
```

4.26.1.47 CYCLE4072 - Продольное шлифование с подачей в точке возврата и сигналом отмены

Синтаксис

```
CYCLE4072(<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>,
<S_H>, <S_A1>, <S_A2>)
```

Параметр

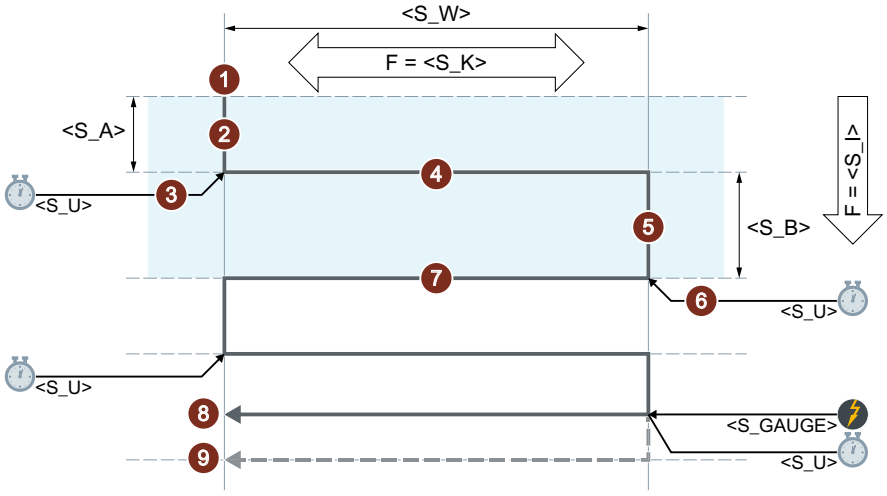
№	Параметр	Тип данных	Объяснение
1	<S_GAUGE>	STRING	Условия отмены подачи на глубину: 1. Номер быстрого входа 2. Логическое выражение
2	<S_A>	REAL	Глубина подачи в начале
3	<S_B>	REAL	Глубина подачи в конце
4	<S_W>	REAL	Ширина шлифования
5	<S_U>	REAL	Время выхаживания
6	<S_I>	REAL	Подача на глубину
7	<S_K>	REAL	Подача для поперечной подачи
8	<S_H>	INT	Число повторов
9	<S_A1>	AXIS	Ось подачи (опция)или 1-я Гео-ось
10	<S_A2>	AXIS	Ось качания (опция) или 2-я Гео-ось

Функция

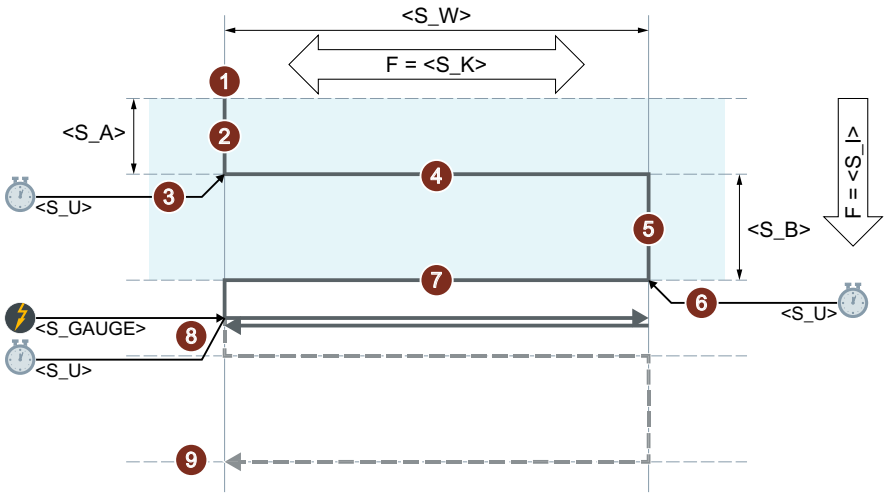
Цикл служит для обработки повторяющихся подач с учетом внешнего сигнала отмены. Глубина подачи в начале и в конце может различаться. Во время подачи происходит движение по касательной. Подача на глубину отменяется, если выполняется условие отмены. После отмены подачи на глубину всегда выполняется полный ход.

Процесс

Отмена подачи на глубину в конце



Отмена подачи на глубину в начале



① Начало цикла в текущей позиции оси качания.
② Перемещение оси подачи на глубину подачи в начале <S_A> с глубиной подачи <S_I>.
③ Выхаживание со временем выхаживания <S_U>.
④ Перемещение оси качания с шириной шлифования <S_W> в качестве пути перемещения и подача для поперечной подачи <S_K>.
⑤ Перемещение оси подачи на глубину подачи в конце <S_B> с глубиной подачи <S_I>.
⑥ Выхаживание со временем выхаживания <S_U>.
⑦ Перемещение оси качания с шириной шлифования <S_W> в качестве пути перемещения в начальную точку и подача для поперечной подачи <S_K>.
⑧ Сигнал отмены: Обработка завершается при достижении следующей стартовой точки.
⑨ Без сигнала отмены: Процесс повторяется до тех пор, пока не будет достигнуто запрограммированное количество повторов <S_H>.

Обозначает повторяющиеся этапы процессов.

Примечание

Прервать этот процесс отдельным кадром невозможно.

Ресурсы

В качестве ресурсов цикл использует синхронное действие с выходом за границы кадров и переменную синхронного действия. Синхронное действие динамически определяется из свободной области диапазона синхронных действий (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). В качестве переменной синхронного действия используется SYG_IS[1].

Примеры

Пример 1: Осциллирующее движение с двумя ходами

Параметры цикла:

- Начальная глубина подачи: 0,02 мм
- Конечная глубина подачи: 0,01 мм
- Ход: 100 мм
- Время выхаживания: 1 с
- Скорость подачи: 1 мм/мин
- Поперечная подача: 1000 мм/мин
- Повторы: 2
- Ось качания и ось подачи: стандартные гео-оси

Сигнал отмены: быстрый вход 1 ($A_IN[1])

Программный код

```
N10 T1 D1
N20 CYCLE4072("1",0.02,0.01,100,1,1,1000,2)
N30 M30
```

Пример 2: Осциллирующее движение с двумя ходами

Параметры цикла:

- Начальная глубина подачи: 0,02 мм
- Конечная глубина подачи: 0,01 мм
- Ход: 100 мм
- Время выхаживания: 1 с
- Скорость подачи: 1 мм/мин
- Поперечная подача: 1000 мм/мин
- Повторы: 2
- Ось качания и ось подачи: стандартные гео-оси

Сигнал отмены: Переменная $A_DBR[20] < 0,01

Программный код

```
N10 T1 D1
N20 CYCLE4072("($A_DBR[20]<0.01)",0.02,0.01,100,1,1,1000,2)
N30 M30
```

4.26.1.48 CYCLE4073 - Продольное шлифование с непрерывной подачей на глубину

Синтаксис

```
CYCLE4073(<S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>, <S_A1>, <S_A2>)
```

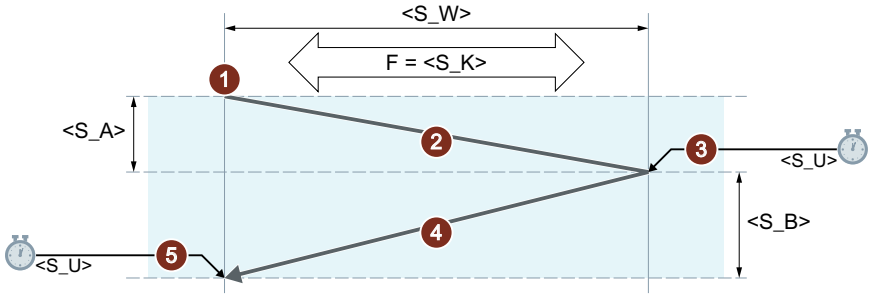
Параметр

№	Параметр	Тип данных	Объяснение
1	<S_A>	REAL	Глубина подачи в начале
2	<S_B>	REAL	Глубина подачи в конце
3	<S_W>	REAL	Ширина шлифования
4	<S_U>	REAL	Время выхаживания
5	<S_K>	REAL	Подача для поперечной подачи
6	<S_H>	INT	Число повторов
7	<S_A1>	AXIS	Ось подачи (опция)или 1-я Гео-ось
8	<S_A2>	AXIS	Ось качания (опция) или 2-я Гео-ось

Функция

Цикл служит для обработки повторяющихся подач. При этом подача на глубину от начала к концу и от конца к началу может быть разной.

Процесс



① Начало цикла в текущей позиции оси качания с глубиной подачи 0.

② Перемещение оси качания с шириной шлифования <S_W> в качестве пути перемещения и подача для поперечной подачи <S_K> с непрерывным увеличением глубины подачи до значения глубины подачи в начале <S_A>.

③ Выхаживание со временем выхаживания <S_U>.

④ Перемещение оси качания с шириной шлифования <S_W> в качестве пути перемещения в начальную точку и подача для поперечной подачи <S_K> с непрерывным увеличением глубины подачи до значения глубины подачи в конце <S_B>.

⑤ Выхаживание со временем выхаживания <S_U>.

Обозначает повторяющиеся этапы процессов.

Процесс повторяется до тех пор, пока не будет достигнуто запрограммированное количество повторов <S_H>.

Примечание

Прервать этот процесс отдельным кадром невозможно.

Пример

Осциллирующее движение с двумя ходами

Параметры цикла:

- Начальная глубина подачи: 0,02 мм
- Конечная глубина подачи: 0,01 мм
- Ход: 100 мм
- Время выхаживания: 1 с
- Поперечная подача: 1000 мм/мин
- Повторы: 2
- Ось качания и ось подачи: стандартные гео-оси

Программный код

```
N10 T1 D1
N20 CYCLE4073(0.02,0.01,100,1,1000,2)
N30 M30
```

4.26.1.49 CYCLE4074 - Продольное шлифование с непрерывной подачей на глубину и сигналом отмены

Синтаксис

```
CYCLE4074(<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>,
<S_A1>, <S_A2>)
```

Параметр

№	Параметр	Тип данных	Объяснение
1	<S_GAUGE>	STRING	Условия отмены подачи на глубину: 1. Номер быстрого входа 2. Логическое выражение
2	<S_A>	REAL	Глубина подачи в начале
3	<S_B>	REAL	Глубина подачи в конце
4	<S_W>	REAL	Ширина шлифования
5	<S_U>	REAL	Время выхаживания
6	<S_K>	REAL	Подача для поперечной подачи

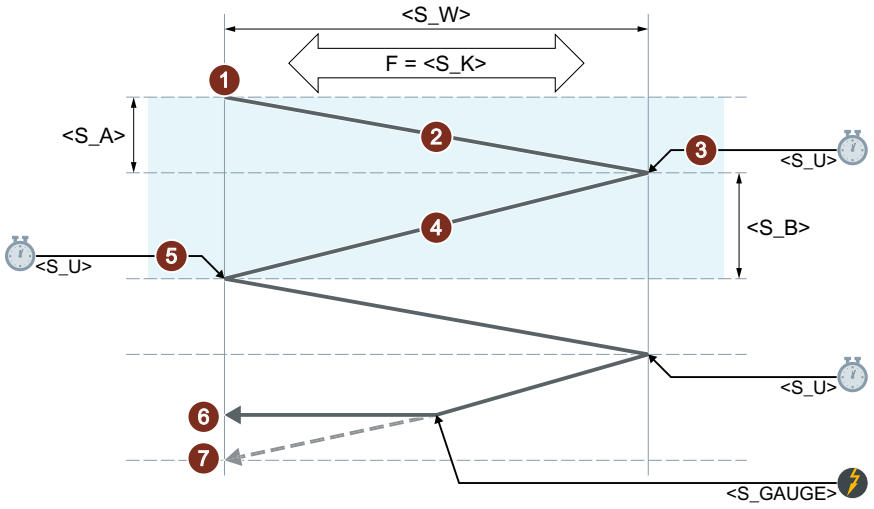
№	Параметр	Тип данных	Объяснение
7	<S_H>	INT	Число повторов
8	<S_A1>	AXIS	Ось подачи (опция)или 1-я Гео-ось
9	<S_A2>	AXIS	Ось качания (опция) или 2-я Гео-ось

Функция

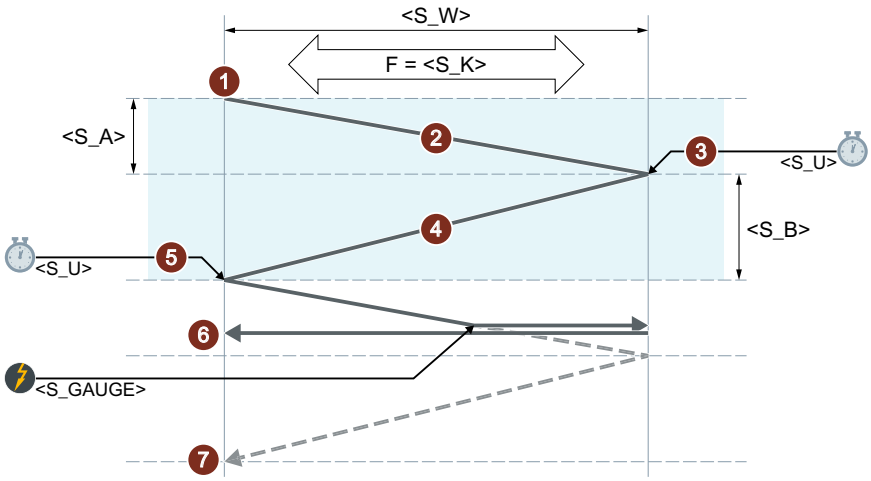
Цикл служит для обработки повторяющихся подач с учетом, например, внешнего сигнала отмены. Глубина подачи в начале и в конце может различаться. Подача на глубину отменяется, если выполняется условие отмены. После отмены подачи на глубину всегда выполняется полный ход.

Процесс

Отмена подачи на глубину от конца к началу



Отмена подачи на глубину от начала к концу



① Начало цикла в текущей позиции оси качания с глубиной подачи 0.

② Перемещение оси качания с шириной шлифования <S_W> в качестве пути перемещения и подача для поперечной подачи <S_K> с непрерывным увеличением глубины подачи до значения глубины подачи в начале <S_A>.

③ Выхаживание со временем выхаживания <S_U>.

④ Перемещение оси качания с шириной шлифования <S_W> в качестве пути перемещения в начальную точку и подача для поперечной подачи <S_K> с непрерывным увеличением глубины подачи до значения глубины подачи в конце <S_B>.

⑤ Выхаживание со временем выхаживания <S_U>.

⑥ Сигнал отмены: Подача на глубину прерывается. Обработка завершается при достижении следующей стартовой точки.

⑦ Без сигнала отмены: Процесс повторяется до тех пор, пока не будет достигнуто запрограммированное количество повторов <S_H>.

Обозначает повторяющиеся этапы процессов.

Примечание

Прервать этот процесс отдельным кадром невозможно.

Ресурсы

В качестве ресурсов цикл использует синхронное действие с выходом за границы кадров и переменную синхронного действия. Синхронное действие динамически определяется из свободной области диапазона синхронных действий (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). В качестве переменной синхронного действия используется SYG_IS[1].

Примеры

Пример 1: Осциллирующее движение с двумя ходами

Параметры цикла:

- Начальная глубина подачи: 0,02 мм
- Конечная глубина подачи: 0,01 мм
- Ход: 100 мм
- Время выхаживания: 1 с
- Поперечная подача: 1000 мм/мин
- Повторы: 2
- Ось качания и ось подачи: стандартные гео-оси

Сигнал отмены: быстрый вход 1 ($A_IN[1])

Программный код

```
N10 T1 D1
N20 CYCLE4074("1",0.02,0.01,100,1,1000,2)
N30 M30
```

Пример 2: Осциллирующее движение с двумя ходами

Параметры цикла:

- Начальная глубина подачи: 0,02 мм
- Конечная глубина подачи: 0,01 мм
- Ход: 100 мм
- Время выхаживания: 1 с
- Поперечная подача: 1000 мм/мин
- Повторы: 2
- Ось качания и ось подачи: стандартные гео-оси

Сигнал отмены: Переменная $A_DBR[20] < 0,01

Программный код

```
N10 T1 D1
N20 CYCLE4074("($A_DBR[20]<0.01)",0.02,0.01,100,1,1000,2)
N30 M30
```

4.26.1.50 CYCLE4075 - Плоское шлифование с подачей в точке возврата

Синтаксис

```
CYCLE4075(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>,
<S_A2>)
```

Параметр

№	Параметр	Тип данных	Объяснение
1	<S_I>	REAL	Глубина подачи в начале
2	<S_J>	REAL	Глубина подачи в конце
3	<S_K>	REAL	Общая глубина подачи
4	<S_A>	REAL	Ширина шлифования
5	<S_R>	REAL	Подача на глубину
6	<S_F>	REAL	Подача для поперечной подачи
7	<S_P>	REAL	Время выхаживания
8	<S_A1>	AXIS	Ось подачи (опция)
9	<S_A2>	AXIS	Качающаяся ось (опция)

Функция

Цикл служит для выполнения обработки с общей глубиной подачи в несколько этапов подачи. Значения глубины подачи в начале и в конце могут быть разными. Во время подачи происходит движение по касательной.

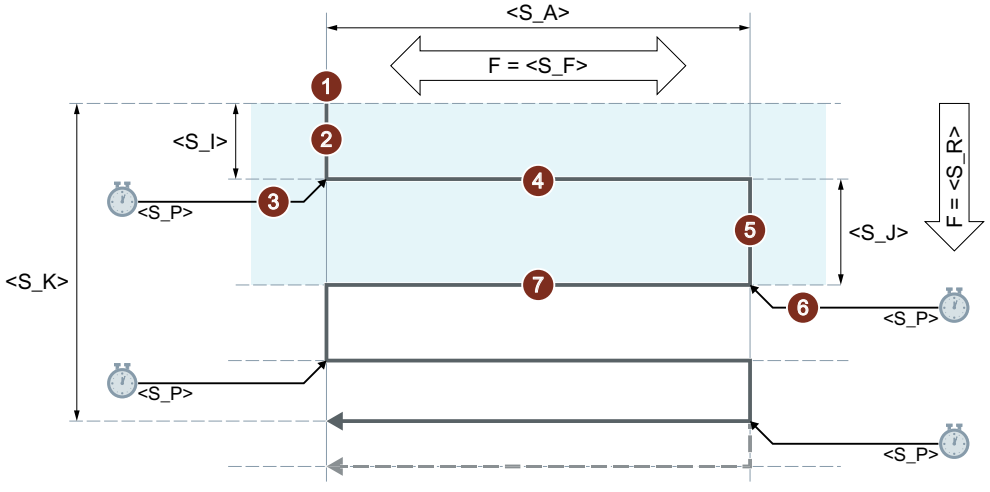
Параметры пути P1 - P4 могут иметь отрицательное или положительное значение.

Данные оси подачи и/или оси качания не обязательны. Если один или оба параметра не заданы, то цикл использует обе первые гео-оси канала.

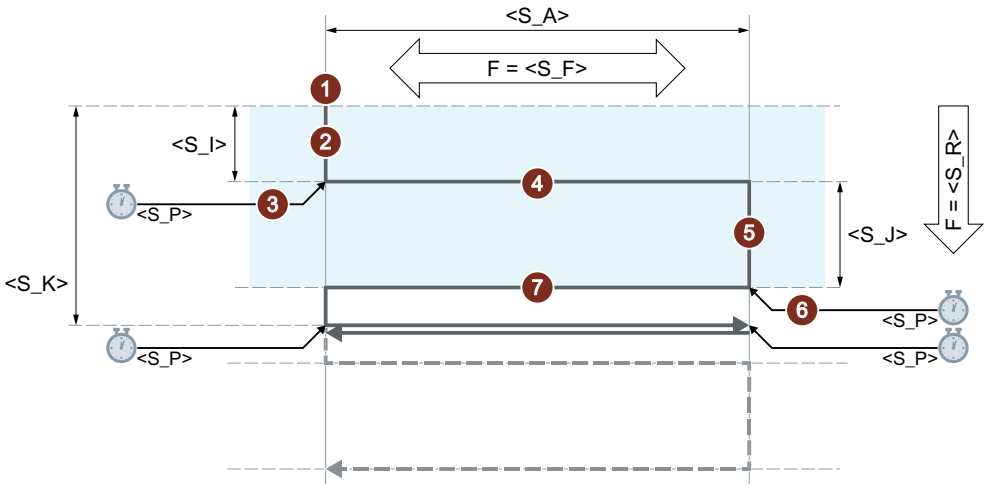
Если сумма глубины подачи в начале и в конце равна 0, или общая глубина подачи равна 0, то выполняется только один ход выхаживания.

Процесс

Общая глубина подачи достигается при подаче на второй точке возврата



Общая глубина подачи достигается при подаче на первой точке возврата



① Начало цикла в текущей позиции оси качания.

② Перемещение оси подачи на глубину подачи в начале <S_I> с глубиной подачи <S_R>.

③ Выхаживание со временем выхаживания <S_P>.

④ Перемещение оси качания с шириной шлифования <S_A> в качестве пути перемещения и подача для поперечной подачи <S_F>.

⑤ Перемещение оси подачи на глубину подачи в конце <S_J> с глубиной подачи <S_R>.

⑥ Выхаживание со временем выхаживания <S_P>.

⑦ Перемещение оси качания с шириной шлифования <S_A> в качестве пути перемещения в начальную точку и подача для поперечной подачи <S_F>.

Обозначает повторяющиеся этапы процессов.

Процесс повторяется до тех пор, пока не будет достигнута общая глубина подачи <S_K>. Последний ход в этом случае разделяется неравномерно.

Примечание

Прервать этот процесс отдельным кадром невозможно.

Пример

Осциллирующее движение со следующими данными:

- глубина подачи 0,02 мм в начале
- глубина подачи 0,01 мм в конце
- общая глубина подачи 1 мм
- ход 100 мм
- подача на глубину 1 мм/мин
- поперечная подача 1000 мм/мин
- время выхаживания 1 с
- стандартные гео-оси

Программный код

```
N10 T1 D1
N20 CYCLE4075(0.02,0.01,1,100,1,1000,1)
N30 M30
```

4.26.1.51 CYCLE4077 - Плоское шлифование с подачей в точке возврата и сигналом отмены

Синтаксис

```
CYCLE4077(<S_GAUGE>, <S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>,
<S_P>, <S_A1>, <S_A2>)
```

Параметр

№	Параметр	Тип данных	Объяснение
1	<S_GAUGE>	STRING	Условие отмены подачи на глубину: • Номер быстрого входа • Логическое выражение
2	<S_I>	REAL	Глубина подачи в начале
3	<S_J>	REAL	Глубина подачи в конце
4	<S_K>	REAL	Общая глубина подачи
5	<S_A>	REAL	Ширина шлифования
6	<S_R>	REAL	Подача на глубину
7	<S_F>	REAL	Подача для поперечной подачи
8	<S_P>	REAL	Время выхаживания
9	<S_A1>	AXIS	Ось подачи (опция)
10	<S_A2>	AXIS	Качающаяся ось (опция)

Функция

Цикл служит для выполнения обработки с общей глубиной подачи в несколько этапов подачи. Значения глубины подачи в начале и в конце могут быть разными. Во время подачи происходит движение по касательной. Подача на глубину прекращается, когда сигнал отмены быстрого входа равен 1, или выполняется условие отмены. После отмены выполняется полный ход.

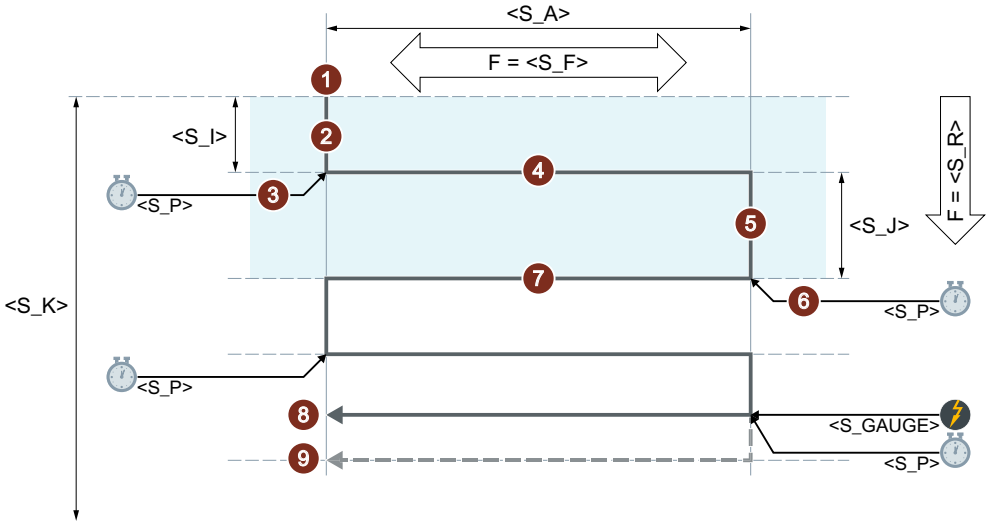
Параметры пути P2 - P5 могут иметь отрицательное или положительное значение.

Данные оси подачи и/или оси качания не обязательны. Если один или оба параметра не заданы, то цикл использует обе первые гео-оси канала.

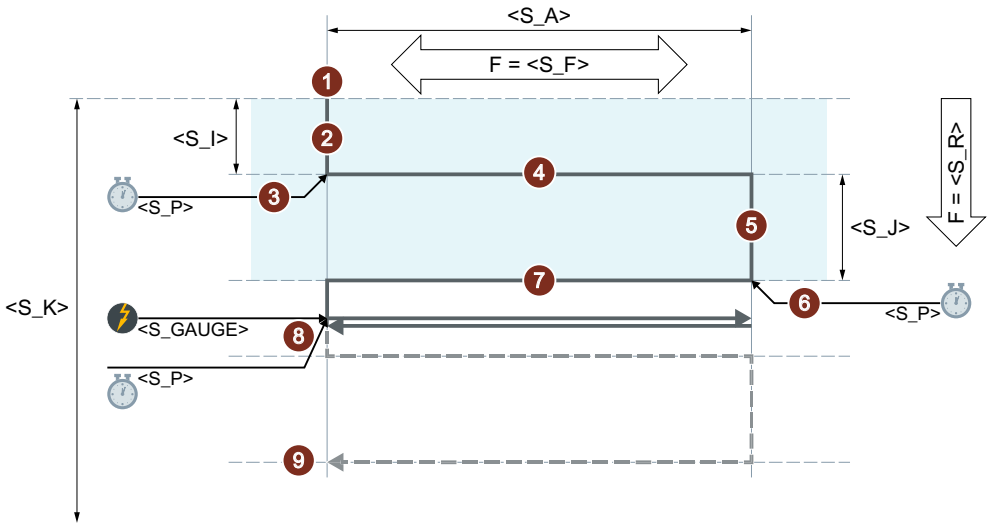
Если сумма глубины подачи в начале и в конце равна 0, или общая глубина подачи равна 0, то выполняется только один ход выхаживания.

Процесс

Отмена подачи на глубину в конце



Отмена подачи на глубину в начале



① Начало цикла в текущей позиции оси качания.
② Перемещение оси подачи на глубину подачи в начале <S_I> с глубиной подачи <S_R>.
③ Выхаживание со временем выхаживания <S_P>.
④ Перемещение оси качания с шириной шлифования <S_A> в качестве пути перемещения и подача для поперечной подачи <S_F>.
⑤ Перемещение оси подачи на глубину подачи в конце <S_J> с глубиной подачи <S_R>.
⑥ Выхаживание со временем выхаживания <S_P>.
⑦ Перемещение оси качания с шириной шлифования <S_A> в качестве пути перемещения в начальную точку и подача для поперечной подачи <S_F>.
⑧ Сигнал отмены: Обработка завершается при достижении следующей стартовой точки.
⑨ Без сигнала отмены: Процесс повторяется до тех пор, пока не будет достигнута общая глубина подачи <S_K>. Последний ход в этом случае разделяется неравномерно.

Обозначает повторяющиеся этапы процессов.

Примечание

Прервать этот процесс отдельным кадром невозможно.

Ресурсы

В качестве ресурсов цикл использует синхронное действие с выходом за границы кадров и переменную синхронного действия. Синхронное действие динамически определяется из свободной области диапазона синхронных действий (CUS.DIR - 1 ..., CMA.DIR - 1000 ... , CST.DIR – 1199 ...). В качестве переменной синхронного действия используется SYG_IS[1].

Примеры

Пример 1

Осциллирующее движение со следующими данными:

- глубина подачи 0,02 мм в начале
- глубина подачи 0,01 мм в конце
- общая глубина подачи 1 мм
- ход 100 мм
- подача на глубину 1 мм/мин
- поперечная подача 1000 мм/мин
- время выхаживания 1 с
- стандартные гео-оси

Сигнал отмены: быстрый вход 1 ($A_IN[1])

Программный код

```
N10 T1 D1
N20 CYCLE4077("1",0.02,0.01,1,100,1,1000,1)
N30 M30
```

Пример 2

Осциллирующее движение со следующими данными:

- глубина подачи 0,02 мм в начале
- глубина подачи 0,01 мм в конце
- общая глубина подачи 1 мм
- ход 100 мм
- подача на глубину 1 мм/мин
- поперечная подача 1000 мм/мин
- время выхаживания 1 с
- стандартные гео-оси

Сигнал отмены: Dualport-RAM-переменная 20 меньше, чем 0,01 ($A_DBR[20] < 0,01)

Программный код

```
N10 T1 D1
N20 CYCLE4077("($A_DBR[20]<0.01)",0.02,0.01,1,100,1,1000,1)
N30 M30
```

4.26.1.52 CYCLE4078 - Плоское шлифование с непрерывной подачей на глубину

Синтаксис

```
CYCLE4078(<S_I>, <S_J>, <S_K>, <S_A>, <S_F>, <S_P>, <S_A1>, <S_A2>)
```

Параметр

№	Параметр	Тип данных	Объяснение
1	<S_I>	REAL	Глубина подачи от начала к концу
2	<S_J>	REAL	Глубина подачи от конца к началу
3	<S_K>	REAL	Общая глубина подачи
4	<S_A>	REAL	Ширина шлифования
5	<S_F>	REAL	Подача
6	<S_P>	REAL	Время выхаживания
7	<S_A1>	AXIS	Ось подачи (опция)
8	<S_A2>	AXIS	Качающаяся ось (опция)

Функция

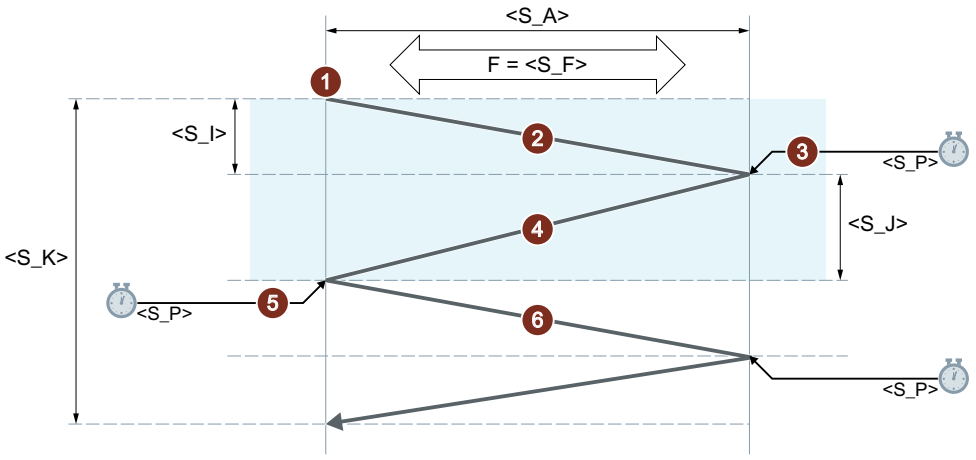
Цикл служит для выполнения обработки с общей глубиной подачи путем непрерывной подачи. Глубина подача от начала к концу и от конца к началу может различаться.

Параметры пути P1 - P4 могут иметь отрицательное или положительное значение.

Данные оси подачи и/или оси качания не обязательны. Если один или оба параметра не заданы, то цикл использует обе первые гео-оси канала.

Если сумма глубины подачи P1 и P2 равна 0, или общая глубина подачи равна 0, то выполняется только один ход выхаживания.

Процесс



① Начало цикла в текущей позиции оси качания с глубиной подачи 0.

② Перемещение оси качания с шириной шлифования <S_A> в качестве пути перемещения и подача <S_F> с непрерывным увеличением глубины подачи до значения глубины подачи в начале <S_I>.

③ Выхаживание со временем выхаживания <S_P>.

④ Перемещение оси качания с шириной шлифования <S_A> в качестве пути перемещения в начальную точку и подача <S_F> с непрерывным увеличением глубины подачи до значения глубины подачи в конце <S_J>.

⑤ Выхаживание со временем выхаживания <S_P>.

⑥ Перемещение оси качания с шириной шлифования <S_A> в качестве пути перемещения в начальную точку и подача <S_F>.

Обозначает повторяющиеся этапы процессов.

Процесс повторяется до тех пор, пока не будет достигнута общая глубина подачи <S_K>. Последний ход в этом случае разделяется неравномерно.

Примечание

Прервать этот процесс отдельным кадром невозможно.

Пример

Осциллирующее движение со следующими данными:

- глубина подачи 20 мм в начале
- глубина подачи 10 мм в конце
- общая глубина подачи 100 мм
- ход 100 мм
- Подача 1000 мм/мин
- время выхаживания 1 с
- стандартные гео-оси

Программный код

```
N10 T1 D1
N20 CYCLE4078(20,10,100,100,1000,1)
N30 M30
```

4.26.1.53 CYCLE4079 - Плоское шлифование с прерывистой подачей на глубину

Синтаксис

```
CYCLE4079(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>,
<S_A2>)
```

Параметр

№	Параметр	Тип данных	Объяснение
1	<S_I>	REAL	Глубина подачи в начале
2	<S_J>	REAL	Глубина подачи в конце
3	<S_K>	REAL	Общая глубина подачи
4	<S_A>	REAL	Ширина шлифования
5	<S_R>	REAL	Подача на глубину
6	<S_F>	REAL	Подача для поперечной подачи
7	<S_P>	REAL	Время выхаживания
8	<S_A1>	AXIS	Ось подачи (опция)
9	<S_A2>	AXIS	Качающаяся ось (опция)

Функция

Цикл служит для выполнения обработки с общей глубиной подачи в несколько этапов подачи. Значения глубины подачи в начале и в конце могут быть разными. Во время подачи происходит движение по касательной.

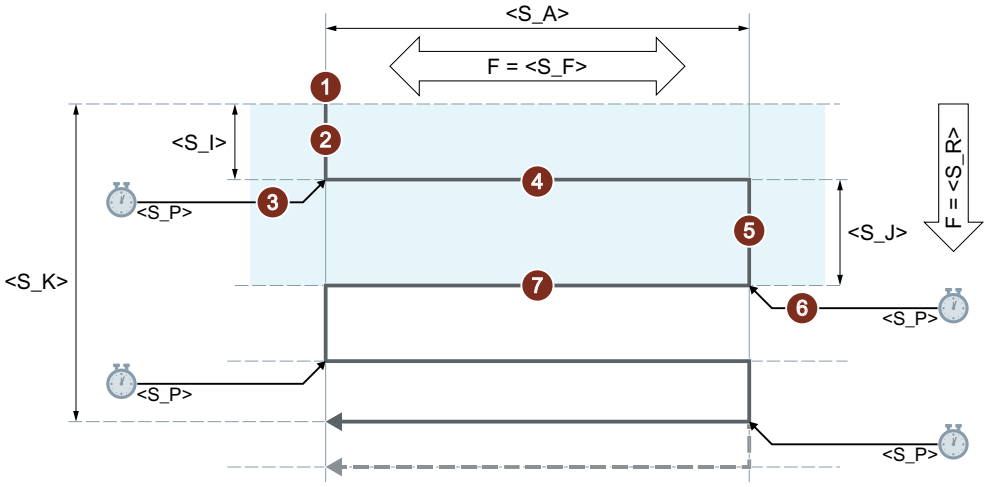
Параметры пути P1 - P4 могут иметь отрицательное или положительное значение.

Данные оси подачи и/или оси качания не обязательны. Если один или оба параметра не заданы, то цикл использует обе первые гео-оси канала.

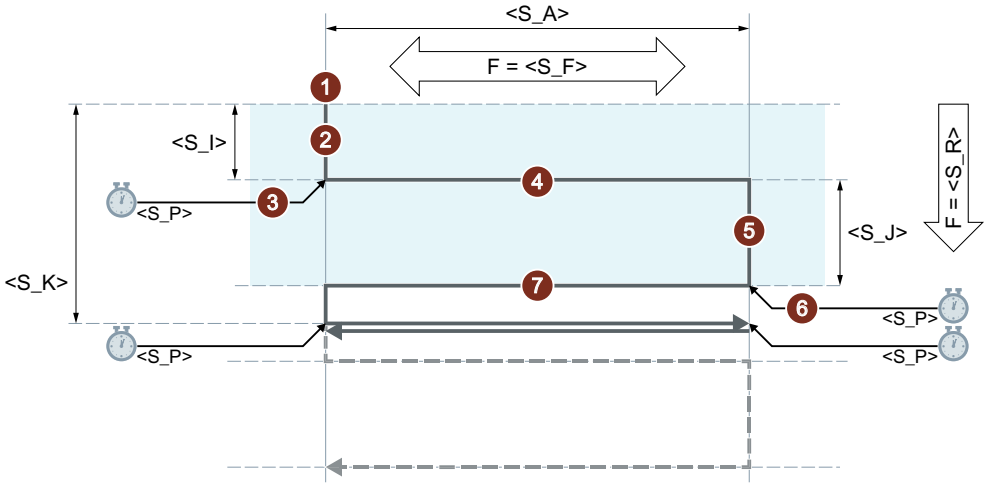
Если сумма глубины подачи в начале и в конце равна 0, или общая глубина подачи равна 0, то выполняется только один ход выхаживания.

Процесс

Общая глубина подачи достигается при подаче на второй точке возврата



Общая глубина подачи достигается при подаче на первой точке возврата



① Начало цикла в текущей позиции оси качания.

② Перемещение оси подачи на глубину подачи в начале <S_I> с глубиной подачи <S_R>.

③ Выхаживание со временем выхаживания <S_P>.

④ Перемещение оси качания с шириной шлифования <S_A> в качестве пути перемещения и подача для поперечной подачи <S_F>.

⑤ Перемещение оси подачи на глубину подачи в конце <S_J> с глубиной подачи <S_R>.

⑥ Выхаживание со временем выхаживания <S_P>.

⑦ Перемещение оси качания с шириной шлифования <S_A> в качестве пути перемещения в начальную точку и подача для поперечной подачи <S_F>.

Обозначает повторяющиеся этапы процессов.

Процесс повторяется до тех пор, пока не будет достигнута общая глубина подачи <S_K>. Последний ход в этом случае разделяется неравномерно.

Примечание

Прервать этот процесс отдельным кадром невозможно.

Пример

Осциллирующее движение со следующими данными:

- глубина подачи 0,02 мм в начале
- глубина подачи 0,01 мм в конце
- общая глубина подачи 1 мм
- ход 100 мм
- подача на глубину 1 мм/мин
- поперечная подача 1000 мм/мин
- время выхаживания 1 с
- стандартные гео-оси

Программный код

```
N10 T1 D1
N20 CYCLE4079(0.02,0.01,1,100,1,1000,1)
N30 M30
```

4.26.1.54 GROUP_BEGIN — Начало блока программы

Синтаксис

```
GROUP_BEGIN(<_LEVEL>, <_NAME>, <_SP>, <_MODE>, <S_ICON>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение	
1		<_LEVEL>	INT	Плоскость	
				0 =	Главная плоскость
				1 =	Вспомогательный уровень 1
2		<_NAME>	STRING[128]	Имя блока	
3		<_SP>	INT	Шпиндель	
				0 =	Шпинделя нет
				1 =	Главный шпиндель
				2 =	Противошпиндель

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение		
4		<_MODE>	INT	Режим		
				Бит 0	=1	GROUP_ADDEND существует
				Бит 1	=1	ShopTurn: автоматический отвод (движение к точке смены инструмента)
				Бит 12	зарезервировано	
				Бит 13	зарезервировано	
5		<S_ICON>	STRING[32]	Название иконки (только для интерфейса пользователя)		

4.26.1.55 GROUP_END — Конец блока программы

Синтаксис

```
GROUP_END(<_LEVEL>, <_SP>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение	
1		<_LEVEL>	INT	Плоскость	
				0 =	Главная плоскость
				1 =	Вспомогательный уровень 1
2		<_SP>	INT	Шпиндель	
				0 =	Шпинделя нет
				1 =	Главный шпиндель
				2 =	Противошпиндель

4.26.1.56 GROUP_ADDEND — конец дополнения ввода

Синтаксис

```
GROUP_ADDEND(<_LEVEL>, <_SP>)
```

Параметр

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение	
1		<_LEVEL>	INT	Плоскость	
				0 =	Главная плоскость
				1 =	Вспомогательный уровень 1

№	Параметр в маске	Параметр внутренний	Тип данных	Объяснение	
2		<_SP>	INT	Шпиндель	
				0 =	Шпинделя нет
				1 =	Главный шпиндель
				2 =	Противошпиндель

4.26.1.57 Граничные условия

Технологическое масштабирование внутри масок цикла

Если технологическое масштабирование активно, для различных масок цикла можно выбрать упрощенный ввод, при котором отображаются только самые важные параметры цикла.

Упрощенный ввод возможен, например, в следующих масках цикла:

Технология	Маска цикла
Сверление	Глубокое сверление
	Нарезание внутренней резьбы
Фрезерная обработка	Прямоугольный карман
	Контурное фрезерование: карман
Токарная обработка	Нарезание резьбы резцом: продольное
	Токарная обработка контура: обработка резаньем
	Токарная обработка контура: выточка
	Токарная обработка контура: токарная выточка

Для соответствующих масок цикла в интерфейсе пользователя есть опции "Ввод: **простой**" и "Ввод: **полный**".

Неотображаемые параметры цикла

Либо не отображаемые при упрощенном вводе параметры цикла предустанавливаются с фиксированными, технологически целесообразными, но не изменяемыми значениями. Либо параметрам цикла через специфические для канала установочные данные цикла присваиваются параметрируемые значения. См. ниже абзац "Ввод в эксплуатацию" > "Спец. для канала установочные данные циклов"

Переключение "Ввод: полный" > "Ввод: простой"

Если одна из масок цикла заполняется при настройке "Ввод полный", а затем переключается на "Ввод простой", то при генерировании вызова цикла для более не отображаемых параметров используются значения по умолчанию или значения установочных данных.

Ввод в эксплуатацию

Спец. для канала машинные данные конфигурации

С помощью машинных данных активируется возможность технологического масштабирования внутри масок цикла:

MD52210 $MCS_FUNCTION_MASK_DISP, бит 9 = 1 (отображать выбор "Ввод простой")

Спец. для канала установочные данные циклов

Если упрощенный ввод внутри масок цикла активен, значения для определенных параметров цикла можно задавать через следующие установочные данные:

Номер	Идентификатор	Объяснение
SD55300	$SCS_EASY_SAFETY_CLEARANCE	Безопасное расстояние
SD55301	$SCS_EASY_DWELL_TIME	Время ожидания
SD55305	$SCS_EASY_DRILL_DEEP_FD1	Глубокое сверление: процент: 1-я подача
SD55306	$SCS_EASY_DRILL_DEEP_DF	Глубокое сверление: процент: подача
SD55307	$SCS_EASY_DRILL_DEEP_V1	Глубокое сверление: мин. подача на глубину
SD55308	$SCS_EASY_DRILL_DEEP_V2	Глубокое сверление: величина отвода
SD55309	$SCS_EASY_THREAD_RETURN_DIST	Нарезание резьбы резцом: интервал обратного хода

4.26.2 Измерительные циклы

Измерительные циклы – это специальные подпрограммы Siemens для решения различных задач измерения. Как и любые другие циклы, измерительные циклы тоже можно адаптировать к конкретной проблеме с помощью параметров.

Измерительные циклы можно использовать для измерений в следующих областях и технологиях:

- измерение инструментов для токарной обработки / фрезерования
- измерение деталей при токарной обработке / фрезеровании

Дополнительная информация

Руководство по программированию "Измерительные циклы"

4.26.2.1 Обзор параметров измерительных циклов

Параметры измерительного цикла CYCLE973

```
PROC CYCLE973(INT S_MVAR,INT S_PRNUM,INT S_CALNUM,REAL S_SETV,INT S_MA,INT S_MD,REAL
S_FA,REAL S_TSA,REAL S_VMS,INT S_NMSP,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Таблица 4-8 Параметры вызывающей функции CYCLE973 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения (default=0012103)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: калибровка на поверхности, кромке или в пазу 0 = длина на поверхности /кромке (в WCS) с известным заданным значением 1 = радиус на поверхности (в WCS) с известным заданным значением 2 = длина в пазу (в MCS), см. S_CALNUM 3 = радиус в пазу (в MCS), см. S_CALNUM
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано 0 = 0
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано 0 = 0
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: выбор измерительной оси и направления измерения при калибровке [2)] 0 = нет данных (при калибровке поверхность на основании паза нет выбора измерительной оси и направления измерения) [4)] 1 = указать выбор измерительной оси и направления измерения, см. S_MA, S_MD (одно направление измерения в одной измерительной оси) 2 = указать выбор измерительной оси, см. S_MA (два направления измерения в одной измерительной оси)
				ПЯТАЯ ПОЗИЦИЯ: полученное позиционное отклонение (наклонное положение измерительного щупа) [2), 3)] 0 = определить позиционное отклонение 1 = не определять позиционное отклонение
				ШЕСТАЯ ПОЗИЦИЯ: зарезервировано 0 = 0
				EINEMILLION: адаптация длины инструмента [7)] 0 = без адаптации длины инструмента (только точки срабатывания) 1 = с адаптацией длины инструмента
2	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (default=1)	
3		S_CALNUM	Номер калибровочного паза при калибровке в пазу (default=1) [5)]	
4		S_SETV	Заданное значение при калибровке на поверхности	
5	X0	S_MA	Измерительная ось (номер оси) [6)] (default=1)	
			Значения:	1 = 1-я ось плоскости (для G18 Z) 2 = 2-я ось плоскости (для G18 X) 3 = 3-я ось плоскости (для G18 Y) [6)]
6	+-	S_MD	Направление измерения (default=1)	
			Значения:	0 = положительное направление измерения 1 = отрицательное направление измерения
7	DFA	S_FA	Путь измерения	
8	TSA	S_TSA	Доверительная область	
9	VMS	S_VMS	Переменная скорость измерения при калибровке [2)]	
10	Измерения	S_NMSP	Число измерений на одном месте [2)] (default=1)	

№	Параметры масок	Параметры цикла	Значение	
11		S_MCBIT	зарезервировано	
12		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
13		_AMODE	Альтернативный режим	

1) все значения по умолчанию = 0 или обозначенные как default=x

2) Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Релевантно только при калибровке в двух осевых направлениях

4) Только измерительная ось и направление измерения определяются автоматически из положения резцов (SL) измерительного щупа. SL=8 → -X , SL=7 → -Z

5) Номер калибровочного паза (n) ссылается на следующие общие параметры настройки (все позиции в MCS):
при положении резцов SL = 7:
SD54615 $SNS_MEA_CAL_EDGE_BASE_AX1[n] Позиция основания паза в 1-й оси плоскости (при G18 Z)
SD54621 $SNS_MEA_CAL_EDGE_PLUS_DIR_AX2[n] Позиция стенки паза в положительном направлении 2-й оси плоскости (при G18 X)
SD54622 $SNS_MEA_CAL_EDGE_MINUS_DIR_AX2[n] Позиция стенки паза в отрицательном направлении 2-й оси плоскости
при положении резцов SL = 8:
SD54619 $SNS_MEA_CAL_EDGE_BASE_AX2[n] Позиция основания паза во 2-й оси плоскости
SD54620 $SNS_MEA_CAL_EDGE_UPPER_AX2[n] Позиция верхней кромки паза во 2-й оси плоскости (только для предварительного позиционирования измерительного щупа)
SD54617 $SNS_MEA_CAL_EDGE_PLUS_DIR_AX1[n] Позиция стенки паза в положительном направлении 1-й оси плоскости
SD54618 $SNS_MEA_CAL_EDGE_MINUS_DIR_AX1[n] Позиция стенки паза в отрицательном направлении 1-й оси плоскости
Указание:
Значения позиций стенки паза +- допускают грубое определение.
Значение ширины паза по разности значений позиций стенки паза необходимо определить точно (индикатор часового типа).
При калибровке в пазу предполагается, что длина инструмента измерительного щупа калиброванной оси = 0.
Значения позиций для основания паза также необходимо определить на станке точно (не размеры по чертежу).

6) Измерительная ось S_MA=3 при калибровке на плоскости и на токарном станке с реальной 3-й осью плоскости (для G18 Y).

7) Адаптация длины инструмента при выверке длины в пазе или длин на поверхности.
Контактная измерительная головка в токарных станках может быть описана с 2 длинами (X Z).
Измерительный щуп для токарной обработки, тип 580
Положение резца 7: При выверке длины по выбору корректируется Z-длина.
Измерительный щуп для токарной обработки, тип 580
Положение резца 8: При выверке длины корректируется по выбору X-длина
Для вариантов измерения радиуса в пазу или радиуса на поверхности коррекция длины инструмента не выполняется.
Всегда сохраняются только соответствующие точки срабатывания.

Параметры измерительного цикла CYCLE974

```
PROC CYCLE974(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,INT S_MA,REAL
S_FA,REAL S_TSA,REAL S_STA1,INT S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL S_TZL,REAL
S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT
_AMODE,INT _DP)
```

Таблица 4-9 Параметры вызывающей функции CYCLE974 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: 0 = измерение торцовой поверхности 1 = внутреннее измерение 2 = наружное измерение
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ: Цель коррекции 0 = только измерение (без коррекции WO или без коррекции инструмента) 1 = измерение, определение и коррекция WO (см. S_KNUM) [3)] 2 = измерение и коррекция инструмента (см. S_KNUM1)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: зарезервировано
				ПЯТАЯ ПОЗИЦИЯ: измерение с или без реверса главного шпинделя (токарный шпиндель) 0 = измерение без реверса 1 = измерение с реверсом
2	Выбор	S_KNUM	Коррекция в смещение нулевой точки (WO), базовое WO или базовое отношение [2)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 99 номеров смещения нулевой точки или 1 до макс. 16 номеров базового смещения
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Коррекция в WO, базовое WO или базовое отношение 0 = коррекция настраиваемого WO 1 = коррекция спец. для канала базового WO 2 = коррекция базового отношения 3 = коррекция глобального базового WO 9 = коррекция активного WO или при G500 последнего активного спец. для канала базового WO
				ПЯТАЯ ПОЗИЦИЯ: Коррекция грубая или точная в WO, базовое WO или базовое отношение 0 = коррекция точная [6)] 1 = коррекция грубая

№	Параметры масок	Параметры цикла	Значение	
3	Выбор	S_KNUM1	Коррекция в коррекции инструмента [2), 4)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ:
				ТРЕТЬЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 999 D-номеров (номеров резцов) при коррекции инструмента; в суммарной и отладочной коррекции также S_DLNUM
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: 0 или однозначный D-номер
				ПЯТАЯ ПОЗИЦИЯ: 0 или однозначный D-номер от 1 до макс. 32000, если однозначные D-номера установлены в MD
				ШЕСТАЯ ПОЗИЦИЯ: коррекция инструмента [2)] 0 = нет данных (коррекция в геометрии инструмента) 1 = коррекция - длина L1 2 = коррекция - длина L2 3 = коррекция - длина L3 4 = коррекция - радиус
				СЕДЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2)] 0 = нет данных (коррекция износа длины инструмента) 1 = коррекция инструмента - суммарная коррекция (SK) [5)] коррекция инструмента прибавляется к существующей SK 2 = коррекция инструмента - отладочная коррекция (EK) [5)] EK (новая) = EK (старая) + SK (старая) поправка, SK (новая) = 0 3 = коррекция инструмента - отладочная коррекция (EK) [5)] коррекция инструмента прибавляется к существующей EK 4 = коррекция инструмента - геометрия
				ВОСЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2)] 0 = нет данных (коррекция в геометрии инструмента обычная, без инверсии) 1 = коррекция с инверсией
				ДЕВЯТАЯ ПОЗИЦИЯ: Коррекция на инструмент 0 = Коррекция на инструмент без инструментов DUPLO 1 = Коррекция на инструмент: инструмент DUPLO (_DP)
4	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (default=1)	
5	X0	S_SETV	Заданное значение	
6	X	S_MA	Измерительная ось (номер оси) (default=1)	
			Значения:	1 = 1-я ось плоскости (для G18 Z) 2 = 2-я ось плоскости (для G18 X) 3 = 3-я ось плоскости (для G18 Y) [5)]
7	DFA	S_FA	Путь измерения	
8	TSA	S_TSA	Доверительная область	
9	α	S_STA1	Начальный угол при измерении с реверсом	
10	Измерения	S_NMSP	Число измерений на одном месте [2)] (default=1)	
11	T	S_TNAME	Имя инструмента [2)]	
12	DL	S_DLNUM	Отладочная суммарная коррекция DL-номер [5)]	
13	ST	_DP	Номер подлежащего коррекции однотипного инструмента (номер DUPLO)	
14	TZL	S_TZL	Коррекция нуля [2), 4)]	

№	Параметры масок	Параметры цикла	Значение	
15	DIF	S_TDIF	Контроль разности размеров [2), 4)]	
16	TUL	S_TUL	Верхняя граница допуска (инкрементальная к заданному значению) [4)]	
17	TLL	S_TLL	Нижняя граница допуска (инкрементальная к заданному значению) [4)]	
18	TMV	S_TMV	Область коррекции при формировании среднего значения [2)]	
19	FW	S_K	Весовой коэффициент для формирования среднего значения [2)]	
20	EVN	S_EVNUM	Номер памяти эмпирических средних значений [2), 7)]	
21		S_MCBIT	зарезервировано	
22		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
23		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: допуск на размер да/нет 0 = нет 1 = да

1) все значения по умолчанию = 0 или обозначенные как default=x

2) Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Коррекция в WO возможна только при измерении без реверса.

4) При коррекции инструмента учитывать в спец. для канала MD 20360 TOOL_PARAMETER_DEF_MASK Бит0 и Бит1

5) Только если функция "Отладочная суммарная коррекция" установлена в общих MD 18108 $MN_MM_NUM_SUMCORR . Дополнительно в общих MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK должна быть установка Бит8=1.

6) Если "точное" WO в MD не установлено, коррекцию выполняют по "грубому" WO.

7) Формирование эмпирического среднего значения возможно только при коррекции инструмента
Диапазон значений памяти эмпирических средних значений:
1 до 20 номеров (n) памяти эмпирических значений, см. спец. для канала SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
10000 до 200000 номеров (n) памяти эмпирических значений, см. спец. для канала SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

Параметры измерительного цикла CYCLE994

```
PROC CYCLE994(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,INT S_MA,REAL
S_SZA,REAL S_SZO,REAL S_FA,REAL S_TSA,INT S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL
S_TZL,REAL S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT
_DMODE,INT _AMODE,INT _DP)
```

Таблица 4-10 Параметры вызывающей функции CYCLE994 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: измерение внутри или снаружи (default = 1) 1 = измерение внутри 2 = измерение снаружи
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ: Цель коррекции 0 = только измерение (без коррекции WO или без коррекции инструмента) 1 = измерение, определение и коррекция WO (см. S_KNUM) [3)] 2 = измерение и коррекция инструмента (см. S_KNUM1)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Диапазон обхода 0 = без диапазона обхода 1 = обходная ось, 1-я ось плоскости (при G18 Z) Измерительная ось см. S_MA. 2 = обходная ось, 2-я ось плоскости (при G18 X). Измерительная ось см. S_MA. 3 = обходная ось, 3-я ось плоскости (для G18 Y). Измерительная ось см. S_MA. [8)]
2	Выбор	S_KNUM	Коррекция смещения нулевой точки (WO), базового WO или базового отношения [2)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 99 номеров смещения нулевой точки или 1 до макс. 16 номеров базового смещения
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Коррекция WO, базового смещения или базового отношения 0 = коррекция настраиваемого WO 1 = коррекция спец. для канала базового WO 2 = коррекция базового отношения 3 = коррекция глобального базового WO 9 = коррекция активного WO или при G500 в последнее активное спец. для канала базовое WO
				ПЯТАЯ ПОЗИЦИЯ: Коррекция грубая или точная в WO, базовое WO или базовое отношение 0 = коррекция точная [6)] 1 = коррекция грубая

№	Параметры масок	Параметры цикла	Значение	
3	Выбор	S_KNUM1	Коррекция в коррекции инструмента [2), 4)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ:
				ТРЕТЬЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 999 D-номеров (номеров резцов) при коррекции инструмента в суммарной и отладочной коррекции также S_DLNUM
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: 0 или однозначные D-номера
				ПЯТАЯ ПОЗИЦИЯ: 0 или однозначные D-номера от 1 до макс. 32000, если однозначные D-номера установлены в MD
				ШЕСТАЯ ПОЗИЦИЯ: коррекция инструмента [2)] 0 = нет данных (коррекция в геометрии инструмента) 1 = коррекция - длина L1 2 = коррекция - длина L2 3 = коррекция - длина L3 4 = коррекция - радиус
				СЕДЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2)] 0 = нет данных (коррекция износа длины инструмента) 1 = коррекция инструмента - суммарная коррекция (SK) [5)] коррекция инструмента прибавляется к существующей SK 2 = коррекция инструмента - отладочная коррекция (EK) [5)] EK (новая) = EK (старая) + SK (старая) поправка, SK (новая) = 0 3 = коррекция инструмента - отладочная коррекция (EK) [5)] коррекция инструмента прибавляется к существующей EK 4 = коррекция инструмента - геометрия
				ВОСЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2)] 0 = нет данных (коррекция в геометрии инструмента обычная, без инверсии) 1 = коррекция с инверсией
				ДЕВЯТАЯ ПОЗИЦИЯ: Коррекция на инструмент 0 = Коррекция на инструмент без инструментов DUPLO 1 = Коррекция на инструмент: инструмент DUPLO (_DP)
4	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (default=1)	
5	X0	S_SETV	Заданное значение	
6	X	S_MA	Номер измерительной оси (по умолчанию=1)	
			Значения:	1 = 1-я ось плоскости (для G18 Z) 2 = 2-я ось плоскости (для G18 X) 3 = 3-я ось плоскости (для G18 Y) [8)]
7	X1	S_SZA	Ширина обхода в измерительной оси	
8	Y1	S_SZO	Ширина обхода в обходной оси	
9	DFA	S_FA	Путь измерения	
10	TSA	S_TSA	Доверительная область	
11	Измерения	S_NMSP	Число измерений на одном месте [2)] (default=1)	
12	T	S_TNAME	Имя инструмента [2)]	
13	DL	S_DLNUM	Отладочная суммарная коррекция DL-номер [5)]	

№	Параметры масок	Параметры цикла	Значение	
14	ST	_DP	Номер подлежащего коррекции однотипного инструмента (номер DUPLO)	
15	TZL	S_TZL	Коррекция нуля [2), 4)]	
16	DIF	S_TDIF	Контроль разности размеров [2), 4)]	
17	TUL	S_TUL	Верхняя граница допуска (инкрементальная к заданному значению) [4)]	
18	TLL	S_TLL	Нижняя граница допуска (инкрементальная к заданному значению) [4)]	
19	TMV	S_TMV	Область коррекции при формировании среднего значения [2)]	
20	FW	S_K	Весовой коэффициент для формирования среднего значения [2)]	
21	EVN	S_EVNUM	Номер памяти эмпирических значений [2), 7)]	
22		S_MCBIT	зарезервировано	
23		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
24		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: допуск на размер да/нет 0 = нет 1 = да

1) все значения по умолчанию = 0 или обозначенные как default=x

2) Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Коррекция в WO возможна только при измерении без реверса.

4) При коррекции инструмента учитывать MD канала 20360 TOOL_PARAMETER_DEF_MASK

5) Только если функция "Отладочная суммарная коррекция" установлена в общих MD 18108 $MN_MM_NUM_SUMCORR . Дополнительно должны быть установлены общие MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK Бит8=1.

6) Если "точное" WO в MD не установлено, коррекцию выполняют по "грубому" WO.

7) Формирование эмпирического среднего значения возможно только при коррекции инструмента
Диапазон значений памяти эмпирических средних значений:
1 до 20 номеров (n) памяти эмпирических значений, см. спец. для канала SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
10000 до 200000 номеров (n) памяти эмпирических значений, см. спец. для канала SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

8) Если Y-ось есть на станке

Параметры измерительного цикла CYCLE976

```
PROC CYCLE976(INT S_MVAR,INT S_PRNUM,REAL S_SETV,REAL S_SETV0,INT S_MA,INT S_MD,REAL
S_FA,REAL S_TSA,REAL S_VMS,REAL S_STA1,INT S_NMSP,INT S_SETV1,INT _DMODE,INT _AMODE)
```

Таблица 4-11 Параметры вызывающей функции CYCLE976 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения (default=1000)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: калибровка на поверхности, калибровочной сфере или в калибровочном кольце [2)] 0 = длина на поверхности с известным заданным значением 1 = радиус в калибровочном кольце с известным диаметром (заданное значение) и известным центром. 2 = радиус в калибровочном кольце с известным диаметром (заданное значение) и неизвестным центром 3 = радиус и длина на калибровочной сфере 4 = радиус на кромке с известным заданным значением. Учитывать выбор измерительной оси и направление измерения. [3)] 5 = радиус между двумя кромками с известным заданным значением и расстоянием между кромками. Необходимо выбрать измерительную ось.
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано 0 = 0
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано 0 = 0
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: выбор измерительной оси и направление измерения при калибровке 0 = нет данных (не требуется выбора измерительной оси и направления измерения) [8)] 1 = указать выбор измерительной оси и направления измерения, см. S_MA, S_MD (одно направление измерения в одной измерительной оси) 2 = указать выбор измерительной оси, см. S_MA (два направления измерения в одной измерительной оси)
				ПЯТАЯ ПОЗИЦИЯ: полученное позиционное отклонение (наклонное положение измерительного щупа) [2)] 0 = определить позиционное отклонение измерительного щупа[6)] 1 = не определять позиционное отклонение
				ШЕСТАЯ ПОЗИЦИЯ: калибровка параллельно осям или под углом 0 = калибровка параллельно осям в активной WCS 1 = калибровка под углом [7)]
				СЕДЬМАЯ ПОЗИЦИЯ: определение длины инструмента при калибровке на поверхности или на сфере 0 = без определения длины инструмента 1 = определение длины инструмента [4)] 2 = калибровка оси подачи на сфере, определение длины инструмента, внесение разности измерений длины инструмента в данные калибровки
2	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (default=1)	
3		S_SETV	Заданное значение	
4	Z0	S_SETV0	Заданное значение отношения длины при калибровке сфера	

№	Параметры масок	Параметры цикла	Значение	
5	X / Y / Z	S_MA	Измерительная ось (номер оси) [2), 6)] (default=1)	
			Значения:	1 = 1-я ось плоскости (для G17 X) 2 = 2-я ось плоскости (для G17 Y) 3 = 3-я Ось плоскости (для G17 Z)
6	+ -	S_MD	Направление измерения [2), 6)]	
			Значения:	0 = положительное 1 = отрицательное
7	DFA	S_FA	Путь измерения	
8	TSA	S_TSA	Доверительная область	
9	VMS	S_VMS	Переменная скорость измерения при калибровке [2)]	
10	α	S_STA1	Начальный угол [2), 5)]	
11	Измерения	S_NMSP	Число измерений на одном месте [2)] (default=1)	
12	X0	S_SETV1	Опорная точка кромки при выверке между 2 кромками [3)]	
13		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
14		_AMODE	Альтернативный режим	

1) все значения по умолчанию = 0 или обозначенные как default=x

2) Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) При калибровке "Радиус в калибровочном кольце" диаметр и центр кольца должны быть известны (4 направления измерения).
При калибровке "Радиус на двух кромках" должен быть известен только интервал кромок в направлении измерительной оси (2 направления измерения).
При калибровке "Радиус на кромке" должно быть известно заданное значение поверхности.

4) Вариант измерения только калибровка на поверхности (длина на поверхности) исправленная длина инструмента получается из S_MD и S_MA.

5) Только для варианта измерения "Калибровочное кольцо, ... и известном центре" (S_MVAR=1xxx02).

6) Измерительная ось только для варианта измерения S_MVAR=0 или =xx1x01 или =xx2x01 или =20000
Вариант измерения: "Калибровка на поверхности" → выбор измерительной оси и направления измерения
или на "Калибровочное кольцо, ... и известном центре" → выбор направления оси и выбор измерительной оси и направления измерения
или на "Калибровочное кольцо, ... и известном центре" → выбор двух направлений оси и выбор измерительной оси
или "Определение длины измерительного щупа" → S_MA=3 → 3-я Ось плоскости (для G17 Z)

7) Вариант измерения только калибровка в калибровочном кольце или на калибровочной сфере
При "Калибровке на калибровочной сфере" при измерении под углом сфера обходится по экватору.

8) При калибровке "Радиус в калибровочном кольце" с неизвестным центром четыре направления измерения в плоскости (при G17 +-X +-Y).
При калибровке "Длина на поверхности" в минусовом направлении оси инструмента (при G17 -Z).

Параметры измерительного цикла CYCLE978

```
PROC CYCLE978(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL
S_TSA,INT S_MA,INT S_MD,INT S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL S_TZL,REAL
S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT
_AMODE,INT _DP)
```

Таблица 4-12 Параметры вызывающей функции CYCLE978 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: элемент контура 0 = измерение поверхности
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ: Цель коррекции 0 = только измерение (без коррекции WO или без коррекции инструмента) 1 = измерение, определение и коррекция WO (см. S_KNUM) 2 = измерение и коррекция инструмента (см. S_KNUM1)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: зарезервировано
				ПЯТАЯ ПОЗИЦИЯ: измерение с/без реверса шпинделя или точная установка измерительного щупа в направлении срабатывания [9)] 0 = измерение без реверса шпинделя, без точной установки измерительного щупа 1 = измерение с реверсом шпинделя 2 = точная установка измерительного щупа в направлении срабатывания
2	Выбор	S_KNUM	Коррекция смещения нулевой точки (WO), базового WO или базового отношения [2)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 99 номеров смещения нулевой точки или 1 до макс. 16 номеров базового смещения
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Коррекция WO, базового смещения или базового отношения 0 = коррекция настраиваемого WO 1 = коррекция спец. для канала базового WO 2 = коррекция базового отношения 3 = коррекция глобального базового WO 9 = коррекция активного WO или при G500 в последнее активное спец. для канала базовое WO
				ПЯТАЯ ПОЗИЦИЯ: Коррекция грубая или точная в WO, базовое WO или базовое отношение 0 = коррекция точная [6)] 1 = коррекция грубая

№	Параметры масок	Параметры цикла	Значение	
3	Выбор	S_KNUM1	Коррекция в коррекции инструмента [2]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ:
				ТРЕТЬЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 999 D-номеров (номеров резцов) при коррекции инструмента, в суммарной и отладочной коррекции также S_DLNUM
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: 0 или однозначные D-номера
				ПЯТАЯ ПОЗИЦИЯ: 0 или однозначные D-номера 1 до макс. 32000, если однозначные D-номера установлены в MD
				ШЕСТАЯ ПОЗИЦИЯ: коррекция инструмента [2] 0 = нет данных (коррекция в геометрии инструмента) 1 = коррекция - длина L1 2 = коррекция - длина L2 3 = коррекция - длина L3 4 = коррекция - радиус
				СЕДЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2] 0 = нет данных (коррекция износа радиуса инструмента) 1 = коррекция инструмента - суммарная коррекция (SK) [5] коррекция инструмента прибавляется к существующей SK 2 = коррекция инструмента - отладочная коррекция (EK) [5] EK (новая) = EK (старая) + SK (старая) поправка, SK (новая) = 0 3 = коррекция инструмента - отладочная коррекция (EK) [5] коррекция инструмента прибавляется к существующей EK 4 = коррекция инструмента - геометрия
				ВОСЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2] 0 = нет данных (коррекция в геометрии инструмента обычная, без инверсии) 1 = коррекция с инверсией
				ДЕВЯТАЯ ПОЗИЦИЯ: Коррекция на инструмент 0 = Коррекция на инструмент без инструментов DUPLO 1 = Коррекция на инструмент: инструмент DUPLO (_DP)
4	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (диапазон значений: от 1 до 40)	
5	X0	S_SETV	Заданное значение	
6	DFA	S_FA	Путь измерения	
7	TSA	S_TSA	Доверительная область	
8	X	S_MA	Номер измерительной оси [7] (диапазон значений 1 до 3)	
			Значения:	1 = 1-я ось плоскости (для G17 X) 2 = 2-я ось плоскости (для G17 Y) 3 = 3-я ось плоскости (для G17 Z) измерение в направлении инструмента
9		S_MD	Направление измерения измерительной оси	
			Значения:	1 = положительное направление измерения 2 = отрицательное направление измерения
10	Измерения	S_NMSP	Число измерений на одном месте [2] (диапазон значений 1 до 9)	
11	TR	S_TNAME	Имя инструмента [3]	

№	Параметры масок	Параметры цикла	Значение	
12	DL	S_DLNUM	Отладочная суммарная коррекция DL-номер [5]	
13	ST	_DP	Номер подлежащего коррекции однотипного инструмента (номер DUPLO)	
14	TZL	S_TZL	Коррекция нуля [2), 3)]	
15	DIF	S_TDIF	Контроль разности размеров [2), 3)]	
16	TUL	S_TUL	Верхняя граница допуска (инкрементальная к заданному значению) [3)]	
17	TLL	S_TLL	Нижняя граница допуска (инкрементальная к заданному значению) [3)]	
18	TMV	S_TMV	Область коррекции при формировании среднего значения [2)]	
19	FW	S_K	Весовой коэффициент для формирования среднего значения [2)]	
20	EVN	S_EVNUM	Блок данных памяти эмпирических значений [2), 8)]	
21		S_MCBIT	зарезервировано	
22		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
23		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: допуск на размер да/нет 0 = нет 1 = да

1) все значения по умолчанию = 0 или обозначенные как диапазон значений a до b

2) Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Только при коррекции в инструменте, в ином случае параметр = ""

4) Только при коррекции в инструменте и допуске на размер "да", в ином случае параметр = 0

5) Только если функция "Отладочная суммарная коррекция" установлена в общих MD 18108 $MN_MM_NUM_SUMCORR . Дополнительно в общих MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK должна быть установка Бит8=1.

6) Если "точное" WO в MD не установлено, коррекцию выполняют по "грубому" WO.

7) Коррекция в геометрии инструмента:
- при измерении в плоскости (S_MA=1 или S_MA=2) Коррекция в радиусе инструмента
- при измерении в направлении инструмента (S_MA=3) Коррекция в длине инструмента L1

8) Формирование эмпирического среднего значения возможно при коррекции инструмента и коррекции в WO
Диапазон значений памяти эмпирических средних значений:
от 1 до 20 номеров (n) памяти эмпирических значений, см. спец. для канала SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
от 10000 до 200000 номеров (n) памяти эмпирических значений, см. спец. для канала SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

9) При измерении с реверсом шпинделя радиус/диаметр измерительного щупа должен быть точно определен. Это должно быть выполнено с помощью варианта калибровки CYCLE976 радиус на кольце или на кроме или на сфере. Иначе результат измерения будет неправильным.

Параметры измерительного цикла CYCLE998

```
PROC CYCLE998(INT S_MVAR,INT S_KNUM,INT S_RA,INT S_PRNUM,REAL S_SETV,REAL S_STA1,REAL
S_INCA,REAL S_FA,REAL S_TSA,INT S_MA,INT S_MD,REAL S_ID,REAL S_SETV0,REAL S_SETV1,REAL
S_SETV2,REAL S_SETV3,INT S_NMSP,INT S_EVNUM,INT _DMODE,INT _AMODE)
```

Таблица 4-13 Параметры вызывающей функции CYCLE998 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения (default=5)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: элемент контура 5 = измерение кромки (один угол) 6 = измерение плоскости (два угла)
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ: Цель коррекции 0 = только измерение, без коррекции WO 1 = измерение, определение и коррекция WO (см. S_KNUM)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: зона защиты 0 = без учета зоны защиты 1 = с учетом зоны защиты
				ПЯТАЯ ПОЗИЦИЯ: измерение с реверсом шпинделя (дифф. измерение) 0 = измерение без реверса шпинделя 1 = измерение с реверсом шпинделя
				ШЕСТАЯ ПОЗИЦИЯ: измерение под углом или параллельно осям 0 = измерение под углом 1 = измерение параллельно осям
2	Выбор	S_KNUM	Коррекция смещения нулевой точки (WO), базового WO или базового отношения [2)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 99 номеров смещения нулевой точки или 1 до макс. 16 номеров базового смещения
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Коррекция WO, базового смещения или базового отношения 0 = коррекция настраиваемого WO 1 = коррекция спец. для канала базового WO 2 = коррекция базового отношения 9 = коррекция активного WO или при G500 в последнее активное спец. для канала базовое WO
				ПЯТАЯ ПОЗИЦИЯ: Коррекция грубая или точная в WO, базовое WO или базовое отношение [3)] 0 = коррекция точная 1 = коррекция грубая

№	Параметры масок	Параметры цикла	Значение	
3	A, B, C	S_RA	Цель коррекции вращение координат или круговая ось [14]	
			Значения:	0 = Цель коррекции вращение координат вокруг оси, получаемой из параметра S_MA [4] >0 = Цель коррекции круговая ось. Номер оси канала круговой оси (предпочтительно круглый стол). Угловая коррекция выполняется в линейной доле WO круговой оси.
4	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (default=1)	
5	DX / DY / DZ	S_SETV	Путь (инкрементальный) от исходной позиции до точки измерения P1 измерительной оси (S_MA) [5]	
6	α	S_STA1	Заданный угол при "Выверка кромки" или "Выверка плоскости" вокруг 1-й оси плоскости (для G17 X) [9]	
7	β	S_INCA	Заданный угол при "Выверка плоскости" вокруг 2-й оси плоскости (для G17 Y) [9]	
8	DFA	S_FA	Путь измерения	
9	TSA	S_TSA	Доверительная область Контроль углового сдвига к заданному значению угла [градус] [6]	
10	X / Y / Z	S_MA	Измерительная ось, смещенная ось [7] (default=201)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: номер измерительной оси 1 = 1-я ось плоскости (для G17 X) 2 = 2-я ось плоскости (для G17 Y) 3 = 3-я Ось плоскости (для G17 Z)
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ: номер смещенной оси 1 = 1-я ось плоскости (для G17 X) 2 = 2-я ось плоскости (для G17 Y) 3 = 3-я Ось плоскости (для G17 Z)
11	+-	S_MD	Направление измерения измерительной оси [8]	
			Значения:	0 = направление измерения определяется из заданного значения и фактической позиции измерительной оси (совместимость) 1 = положительное направление измерения 2 = отрицательное направление измерения
12	L2	S_ID	При варианте измерения "Точная установка кромки": расстояние (инкрементальное) между точками измерения P1 и P2 в смещенной оси (значение >0) Для варианта измерения "Точная установка плоскости" действуют перечисленные ниже параметры.	
13	L2	S_SETV0	Расстояние между точками измерения P1 и P2 в 1-й оси плоскости [10]	
14		S_SETV1	Расстояние между точками измерения P1 и P2 во 2-й оси плоскости [11], [12]	
15	L3x	S_SETV2	Расстояние между точками измерения P1 и P3 в 1-й оси плоскости [11]	
16	L3y	S_SETV3	Расстояние между точками измерения P1 и P3 во 2-й оси плоскости [10]	
17	Измерения	S_NMSP	Число измерений на одном месте [2] (default=1)	
18		S_EVNUM	Блок данных памяти эмпирических значений [2], [13]	

№	Параметр-ры масок	Параметр-ры цикла	Значение	
19		_DMODE	Режим отображения	
			Значе-ния:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызо-вом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
20		_AMODE	Зарезервировано (альтернативный режим)	

1) все значения по умолчанию = 0 или обозначенные как default=x

2) Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) WO "точное", только если цель коррекции – круговая ось и MD 52207 $MCS_AXIS_USAGE_ATTRIB[n] бит 6 = 1. Если WO не установлено в MD, коррекцию выполняют по "грубому" WO.

4) Пример для коррекции во вращении координат: S_MA=102 Измерительная ось Y смещенная ось X дает вращение координат вокруг Z (при G17)

5) Значение релевантно только при зоне защиты "да" (S_MVAR ЧЕТВЕРТАЯ ПОЗИЦИЯ = 1)

6) При позиционировании от точки измерения P1 к точке измерения P2 в смещенной оси угла в параметрах S_STA1 и S_TSA складываются.

7) Номер измерительной оси должен отличаться от номера смещенной оси (к примеру, 101 не разрешено)

8) Направление измерения только "Точная установка кромки" и "Измерение параллельно осям" (S_MVAR=10x105)

9) Угловой диапазон S_STA1 ±45 градусов при "Точной установке кромки"
Угловой диапазон S_STA1 0 до +60 градусов и S_INCA ±30 градусов при "Точной установке плоскости"

10) При варианте измерения "Точная установка плоскости" и "Точная установка кромки"

11) При варианте измерения "Измерение плоскости" и "Измерение параллельно осям"

12) Не для версии измерительных циклов SW04.04.

13) Формирование эмпирического значения при коррекции в WO; диапазон значений памяти средних эмпирических значений:
1 до 20 номер (n) памяти эмпирических значений см. спец. для канала SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]

14) Параметр S_RA значим только для варианта измерения "Выверка кромки" (S_MVAR xxxxx5).

Параметры измерительного цикла CYCLE977

```
PROC CYCLE977(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,REAL S_SETV0,REAL
S_SETV1,REAL S_FA,REAL S_TSA,REAL S_STA1,REAL S_ID,REAL S_SZA,REAL S_SZO,INT S_MA,INT
S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL S_TZL,REAL S_TDIF,REAL S_TUL,REAL S_TLL,REAL
S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT _AMODE,REAL S_XM,REAL_S_YM,INT _DP)
```

Таблица 4-14 Параметры вызывающей функции CYCLE977 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: элемент контура (диапазон значений 1 до 6) 1 = измерение отверстия 2 = измерение цапфы (вала) 3 = измерение паза 4 = измерение перемычки 5 = измерение прямоугольника, внутри 6 = измерение прямоугольника, снаружи
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ: Цель коррекции 0 = только измерение (без коррекции WO или без коррекции инструмента) 1 = измерение, определение и коррекция WO (см. S_KNUM) 2 = измерение и коррекция инструмента (см. S_KNUM1)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: зона защиты 0 = без учета зоны защиты 1 = с учетом зоны защиты
				ПЯТАЯ ПОЗИЦИЯ: измерение с/без реверса шпинделя (дифф. измерение) или точная установка измерительного щупа в направлении срабатывания 0 = измерение без реверса шпинделя, без точной установки измерительного щупа 1 = измерение с реверсом шпинделя 2 = точная установка измерительного щупа в направлении срабатывания
2	Выбор	S_KNUM	Коррекция смещения нулевой точки (WO), базового WO или базового отношения [2)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 99 номеров смещения нулевой точки или 1 до макс. 16 номеров базового смещения
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Коррекция WO, базового смещения или базового отношения 0 = коррекция настраиваемого WO 1 = коррекция спец. для канала базового WO 2 = коррекция базового отношения 3 = коррекция глобального базового WO 9 = коррекция активного WO или при G500 в последнее активное спец. для канала базовое WO
				ПЯТАЯ ПОЗИЦИЯ: Коррекция грубая или точная в WO, базовое WO или базовое отношение 0 = коррекция точная [6)] 1 = коррекция грубая

№	Параметры масок	Параметры цикла	Значение	
3	Выбор	S_KNUM1	Коррекция в коррекции инструмента [2]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ:
				ТРЕТЬЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 999 D-номеров (номеров резцов) при коррекции инструмента; в суммарной и отладочной коррекции также S_DLNUM
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: 0 или однозначные D-номера
				ПЯТАЯ ПОЗИЦИЯ: 0 или однозначные D-номера 1 до макс. 32000, если однозначные D-номера установлены в MD
				ШЕСТАЯ ПОЗИЦИЯ: коррекция инструмента [2] 0 = нет данных (коррекция радиус инструмента) 1 = коррекция - длина L1 2 = коррекция - длина L2 3 = коррекция - длина L3 4 = коррекция - радиус
				СЕДЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2] 0 = нет данных (коррекция износа радиуса инструмента) 1 = коррекция инструмента — суммарная коррекция (SK) [5] коррекция инструмента прибавляется к существующей SK 2 = коррекция инструмента — отладочная коррекция (EK) [5] EK (новая) = EK (старая) + SK (старая) поправка, SK (новая) = 0 3 = коррекция инструмента — отладочная коррекция (EK) [5] коррекция инструмента прибавляется к существующей EK 4 = коррекция инструмента — геометрия
				ВОСЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2] 0 = нет данных (коррекция в геометрии инструмента обычная, без инверсии) 1 = коррекция с инверсией
				ДЕВЯТАЯ ПОЗИЦИЯ: Коррекция на инструмент 0 = Коррекция на инструмент без инструментов DUPLO 1 = Коррекция на инструмент: инструмент DUPLO (_DP)
4	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (диапазон значений: от 1 до 40)	
5	X0	S_SETV	Заданное значение	
6	X0	S_SETV0	заданное значение для прямоугольника 1-й оси плоскости (X для G17)	
7	Y0	S_SETV1	заданное значение для прямоугольника 2-й оси плоскости (Y для G17)	
8	XM	S_XM	Установка заданного значения для средней точки: гео-ось X	
9	YM	S_YM	Установка заданного значения для средней точки: гео-ось Y	
10	DFA	S_FA	Путь измерения	
11	TSA	S_TSA	Доверительная область	
12	α 0	S_STA1	Начальный угол	

№	Параметры масок	Параметры цикла	Значение	
13	DZ	S_ID	Инкрементальная величина 1. Инкрементальная подача 3-й оси плоскости (Z при G17) Направление подачи через знак S_ID. При измерении цапфы, перемычки и прямоугольника снаружи с S_ID определяется уменьшение высоты измерения. 2. Учет зоны защиты При измерении отверстия, паза и прямоугольника внутри и зоны защиты с S_ID определяется высота обхода.	
14	X1	S_SZA	Диаметр или длина (ширина) зоны защиты [7]	
15	Y1	S_SZO	При измерении прямоугольника: ширина зоны защиты 2-й оси плоскости	
16	X	S_MA	номер измерительной оси [7] (только при измерении паза или перемычки)	
			Значения:	1 = 1-я ось плоскости (для G17 X) 2 = 2-я ось плоскости (для G17 Y)
17	ST	_DP	Номер подлежащего коррекции однотипного инструмента (номер DUPLO)	
18	Измерения	S_NMSP	Число измерений на одном месте [2] (диапазон значений 1 до 9)	
19	TR	S_TNAME	Имя инструмента [2]	
20	DL	S_DLNUM	Отладочная суммарная коррекция DL-номер [5]	
21	TZL	S_TZL	Коррекция нуля [2, 4]	
22	DIF	S_TDIF	Контроль разности размеров [2, 4]	
23	TUL	S_TUL	Верхняя граница допуска (инкрементальная к заданному значению) [4]	
24	TLL	S_TLL	Нижняя граница допуска (инкрементальная к заданному значению) [4]	
25	TMV	S_TMV	Область коррекции при формировании среднего значения [2]	
26	FW	S_K	Весовой коэффициент для формирования среднего значения [2]	
27		S_EVNUM	Блок данных памяти эмпирических средних значений [2, 8]	
28		S_MCBIT	зарезервировано	
29		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
30		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: допуск на размер да/нет 0 = нет 1 = да

[1] все значения по умолчанию = 0 или обозначенные как диапазон значений a до b

[2] Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

[3] Только при коррекции в инструменте, в ином случае параметр = ""

[4] Только при коррекции в инструменте и допуске на размер "да", в ином случае параметр = 0

[5] Только если функция "Отладочная суммарная коррекция" установлена в общих MD 18108 $MN_MM_NUM_SUMCORR . Дополнительно в общих MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK должна быть установка Бит8=1.

[6] Если "точное" WO в MD не установлено, коррекцию выполняют по "грубому" WO.

[7] Диаметр или ширина зоны защиты внутри отверстия или паза.
Диаметр или ширина зоны защиты вне цапфы или перемычки

[8] Формирование эмпирического среднего значения возможно только при коррекции инструмента
Диапазон значений памяти эмпирических средних значений:
1 до 20 номеров (n) памяти эмпирических значений, см. спец. для канала SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
10000 до 200000 номеров (n) памяти эмпирических значений, см. спец. для канала SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

Параметры измерительного цикла CYCLE961

```
PROC CYCLE961(INT S_MVAR,INT S_KNUM,INT S_PRNUM,REAL S_SETV0,REAL S_SETV1,REAL
S_SETV2,REAL S_SETV3,REAL S_SETV4,REAL S_SETV5,REAL S_SETV6,REAL S_SETV7,REAL S_SETV8,REAL
S_SETV9,REAL S_STA1,REAL S_INCA,REAL S_ID,REAL S_FA,REAL S_TSA,INT S_NMSP,INT S_MCBIT,INT
_DMODE,INT _AMODE)
```

Таблица 4-15 Параметры вызывающей функции CYCLE961 [1]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения (default ≥ 6)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: элемент контура 5 = настройка прямого внутреннего угла, ввод заданного значения угол и расстояния A1 до A3 6 = настройка прямого внешнего угла, ввод заданного значения угла и расстояния A1 до A3 7 = настройка внутреннего угла, ввод угла и расстояний A1 до A4 8 = настройка внешнего угла, ввод угла и расстояний A1 до A4
				ВТОРАЯ ПОЗИЦИЯ: установка заданного значения как расстояния или через четыре точки 0 = установка заданного значения как расстояния (полярно) 1 = установка заданного значения через четыре точки (точки измерения P1 до P4)
				ТРЕТЬЯ ПОЗИЦИЯ: Цель коррекции 0 = только измерение (без коррекции WO или без коррекции инструмента) 1 = измерение, определение и коррекция WO (см. S_KNUM
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: зона защиты 0 = без учета зоны защиты (препятствие) 1 = с учетом зоны защиты (препятствие), см. S_ID
				ПЯТАЯ ПОЗИЦИЯ: положение угла в WCS 0 = положение угла определяется через параметр S_STA1 (совместимость) 1 = положение 1 угла в позиционированной начальной точке измерения[6] 2 = положение 2 угла, значения расстояния в 1-й оси плоскости (при G17 X) отрицательные (см. S_SETV0, S_SETV1) 3 = положение 3 угла, значения расстояния в 1-й и 2-й оси плоскости (при G17 XY) отрицательные (см. S_SETV0 до S_SETV3) 4 = положение 4 угла, расстояния во 2-й оси плоскости (при G17 Y) отрицательные (см. S_SETV2, S_SETV3)

№	Параметры масок	Параметры цикла	Значение	
2	Выбор	S_KNUM	Коррекция смещения нулевой точки (WO), базового WO или базового отношения [2]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 99 номеров смещения нулевой точки или 1 до макс. 16 номеров базового смещения
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Коррекция WO, базового смещения или базового отношения 0 = коррекция настраиваемого WO 1 = коррекция спец. для канала базового WO 2 = коррекция базового отношения 9 = коррекция активного WO или при G500 последнего активного спец. для канала базового WO
				ПЯТАЯ ПОЗИЦИЯ: Коррекция грубая или точная в WO, базовое WO или базовое отношение 0 = коррекция точная [5] 1 = коррекция грубая
3	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (диапазон значений: от 1 до 40)	
4	L1/X1	S_SETV0	Расстояние L1 между полюсом и точкой измерения P1 в направлении 1-й оси плоскости (при G17 X) [3] (если расстояние L1 = 0, то автоматически вычисляется L1 = M_SETV1 / 2) или начальная точка P1x 1-й оси плоскости (при G17 X)[4]	
5	L2/Y1	S_SETV1	Расстояние L2 между полюсом и точкой измерения P2 в направлении 1-й оси плоскости [3] или начальная точка P1y 2-й оси плоскости (при G17 Y)[4]	
6	L3/X2	S_SETV2	Расстояние L3 между полюсом и точкой измерения P3 в направлении 2-й оси плоскости [3] (Если расстояние L3 = 0, то при непрямом угле автоматически вычисляется L3 = M_SETV3 / 2) или начальная точка P2x 1-й оси плоскости[4]	
7	L4/Y2	S_SETV3	Расстояние L4 между полюсом и точкой измерения P3 в направлении 2-й оси плоскости при непрямом угле [3] или начальная точка P2y 2-й оси плоскости[4]	
8	XP/X3	S_SETV4	Положение полюса в 1-й оси плоскости [3] или начальная точка P3x 1-й оси плоскости [4]	
9	XP/Y3	S_SETV5	Положение полюса во 2-й оси плоскости [3] или начальная точка P3y 2-й оси плоскости [4]	
10	X4	S_SETV6	Начальная точка P4x 1-й оси плоскости [4]	
11	Y4	S_SETV7	Начальная точка P4y 2-й оси плоскости [4]	
12	X0	S_SETV8	Заданное значение измеренного угла в 1-й оси плоскости при коррекции в WO	
13	Y0	S_SETV9	Заданное значение измеренного угла во 2-й оси плоскости при коррекции в WO	
14	α0	S_STA1	Начальный угол от положительного направления 1-й оси плоскости к базовой кромке детали в MCS (+-270 градусов)	
15	α1	S_INCA	Угол между базовыми кромками детали при измерении не прямого угла [7]	
16	DZ	S_ID	Величина подачи на высоте измерения для каждой точки измерения при активной защищенной области (см. S_MVAR).	

№	Параметры масок	Параметры цикла	Значение	
17	DFA	S_FA	Путь измерения	
18	TSA	S_TSA	Доверительная область Контроль углового сдвига к заданному значению угла [градус]	
19	Измерения	S_NMSP	Число измерений на одном месте [2] (диапазон значений 1 до 9) [2]	
20		S_MCBIT	зарезервировано	
21		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
22		_AMODE	Альтернативный режим	

[1] все значения по умолчанию = 0 или обозначенные как диапазон значений a до b

[2] Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

[3] Ввод точек измерения в полярных координатах с учетом начального угла S_STA1 для точки измерения 3 или 4 углового приращения S_INCA.

[4] Ввод точек измерения в прямоугольной системе координат (задача через 4 точки)

[5] Если "точное" WO в MD не установлено, коррекцию выполняют по "грубому" WO.

[6] Диапазон значений угла S_INCA: от -180 до +180 градусов

[7] Начальный угол S_STA1, диапазон значений: прямой угол: +- 90 градусов, произвольный угол: +- 45 градусов

Параметры измерительного цикла CYCLE979

```
PROC CYCLE979(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL
S_TSA,REAL S_CPA,REAL S_CPO,REAL S_STA1,REAL S_INCA,INT S_NMSP,STRING[32] S_TNAME,REAL
S_DLNUM,REAL S_TZL,REAL S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT
S_MCBIT,INT _DMODE,INT _AMODE,REAL S_XM,REAL S_YM,INT _DP)
```

Таблица 4-16 Параметры вызывающей функции CYCLE979 [0)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: элемент контура 1 = измерение отверстия 2 = измерение цапфы (вала)
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ: Цель коррекции 0 = только измерение (без коррекции WO или без коррекции инструмента) 1 = измерение, определение и коррекция WO (см. S_KNUM) 2 = измерение и коррекция инструмента (см. S_KNUM1)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: число точек измерения 0 = 3 точки измерения 1 = 4 точки измерения
				ПЯТАЯ ПОЗИЦИЯ: измерение с/без реверса шпинделя (дифф. измерение) или точная установка измерительного щупа в направлении срабатывания 0 = измерение без реверса шпинделя, без точной установки измерительного щупа 1 = измерение с реверсом шпинделя 2 = точная установка измерительного щупа в направлении срабатывания
2	Выбор	S_KNUM	Коррекция смещения нулевой точки (WO), базового WO или базового отношения [2)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 99 номеров смещения нулевой точки или 1 до макс. 16 номеров базового смещения
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Коррекция WO, базового смещения или базового отношения 0 = коррекция настраиваемого WO 1 = коррекция спец. для канала базового WO 2 = коррекция базового отношения 3 = коррекция глобального базового WO 9 = коррекция активного WO или при G500 в последнее активное спец. для канала базовое WO
				ПЯТАЯ ПОЗИЦИЯ: Коррекция грубая или точная в WO, базовое WO или базовое отношение 0 = коррекция точная [6)] 1 = коррекция грубая

№	Параметры масок	Параметры цикла	Значение	
3	Выбор	S_KNUM1	Коррекция в коррекции инструмента [2]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ:
				ТРЕТЬЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 999 D-номеров (номеров резцов) при коррекции инструмента; в суммарной и отладочной коррекции также S_DLNUM
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: 0 или однозначные D-номера
				ПЯТАЯ ПОЗИЦИЯ: 0 или однозначные D-номера 1 до макс. 32000 если однозначные D-номера установлены в MD
				ШЕСТАЯ ПОЗИЦИЯ: коррекция инструмента [2] 0 = нет данных (коррекция в радиусе инструмента) 1 = коррекция - длина L1 2 = коррекция - длина L2 3 = коррекция - длина L3 4 = коррекция - радиус
				СЕДЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2] 0 = нет данных (коррекция износа радиуса инструмента) 1 = коррекция инструмента - суммарная коррекция (SK) [5] коррекция инструмента прибавляется к существующей SK 2 = коррекция инструмента - отладочная коррекция (EK) [5] EK (новая) = EK (старая) + SK (старая) поправка, SK (новая) = 0 3 = коррекция инструмента - отладочная коррекция (EK) [5] коррекция инструмента прибавляется к существующей EK 4 = коррекция инструмента - геометрия
				ВОСЬМАЯ ПОЗИЦИЯ: коррекция инструмента [2] 0 = нет данных (коррекция в геометрии инструмента обычная, без инверсии) 1 = коррекция с инверсией
				ДЕВЯТАЯ ПОЗИЦИЯ: Коррекция на инструмент 0 = Коррекция на инструмент без инструментов DUPLO 1 = Коррекция на инструмент: инструмент DUPLO (_DP)
4	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (диапазон значений: от 1 до 40)	
5	X0	S_SETV	Заданное значение	
6	DFA	S_FA	Путь измерения	
7	TSA	S_TSA	Доверительная область	
8	X0	S_CPA	Центр 1-й оси плоскости (при G17 X)	
9	Y0	S_CPO	Центр 2-й оси плоскости (при G17 Y)	
10	XM	S_XM	Установка заданного значения для средней точки: гео-ось X	
11	YM	S_YM	Установка заданного значения для средней точки: гео-ось Y	
12	alpha 0	S_STA1	Начальный угол [7]	
13	alpha 1	S_INCA	Угловое приращение [8]	
14	Измерения	S_NMSP	Число измерений на одном месте [1] (диапазон значений от 1 до 9)	
15	ST	_DP	Номер подлежащего коррекции однотипного инструмента (номер DUPLO)	
16	T	S_TNAME	Имя инструмента [2]	

№	Параметры масок	Параметры цикла	Значение	
17	DL	S_DLNUM	Отладочная суммарная коррекция DL-номер [1), 4)]	
18	TZL	S_TZL	Коррекция нуля [1), 2)]	
19	DIF	S_TDIF	Контроль разности размеров [1), 2)]	
20	TUL	S_TUL	Верхняя граница допуска (инкрементальная к заданному значению) [2)]	
21	TLL	S_TLL	Нижняя граница допуска (инкрементальная к заданному значению) [2)]	
22	TMV	S_TMV	Область коррекции при формировании среднего значения [1)]	
23	FW	S_K	Весовой коэффициент для формирования среднего значения [1)]	
24		S_EVNUM	Блок данных памяти эмпирических значений [1), 6)]	
25		S_MCBIT	зарезервировано	
26		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
27		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: допуск на размер да/нет 0 = нет 1 = да

0) Все значения по умолчанию = 0 или обозначенные как диапазон значений a до b.

1) Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

2) Только при коррекции в инструменте, в ином случае параметр = ""

3) Только при коррекции в инструменте и допуске на размер "да", в ином случае параметр = 0

4) Только если функция "Отладочная суммарная коррекция" установлена в общих MD 18108 $MN_MM_NUM_SUMCORR .

5) Если "точное" WO в MD не установлено, коррекцию выполняют по "грубому" WO.

6) Формирование эмпирического среднего значения возможно только при коррекции инструмента
Диапазон значений памяти эмпирических средних значений:
1 до 20 номеров (n) памяти эмпирических значений, см. спец. для канала SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
10000 до 200000 номеров (n) памяти эмпирических значений, см. спец. для канала SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

7) Диапазон значений начального угла -360 до +360 градусов

8) Диапазон значений углового приращения >0 до ≤90 градусов при 4 точках измерения или >0 до ≤120 градусов при 3 точках измерения.

Параметры измерительного цикла CYCLE997

```
PROC CYCLE997 (INT S_MVAR,INT S_KNUM,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL S_TSA,REAL
S_STA1,REAL S_INCA,REAL S_SETV0,REAL S_SETV1,REAL S_SETV2,REAL S_SETV3,REAL S_SETV4,REAL
S_SETV5,REAL S_SETV6,REAL S_SETV7,REAL S_SETV8,REAL S_TNVL,INT S_NMSP,INT S_MCBIT,INT
_DMODE,INT _AMODE)
```

Таблица 4-17 Параметры вызывающей функции CYCLE997 [1), 2)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения (default =9)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: элемент контура 9 = измерение сферы
				ВТОРАЯ ПОЗИЦИЯ: повторение измерения 0 = без повторения измерения 1 = с повторением измерения
				ТРЕТЬЯ ПОЗИЦИЯ: Цель коррекции 0 = только измерение (без коррекции WO) 1 = измерение, определение и коррекция WO (см. S_KNUM)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Стратегия измерения 0 = измерение параллельно осям, без начального угла, ориентация измерительного щупа в соответствии с SD55740, бит 1 1 = измерение с обходом, с начальным углом, ориентация измерительного щупа в соответствии с SD55740, бит 1 2 = измерение с обходом, с начальным углом, измерительный щуп ориентирован в направлении срабатывания 3 = измерение параллельно осям, с начальным углом, ориентация измерительного щупа в соответствии с SD55740, бит 1 4 = измерение параллельно осям, с начальным углом, измерительный щуп ориентирован в направлении срабатывания
				ПЯТАЯ ПОЗИЦИЯ: число измеряемых сфер 0 = измерить одну сферу 1 = измерить три сферы
				ШЕСТАЯ ПОЗИЦИЯ: число точек измерения, только при измерении под углом (соблюдать стратегию измерения: ЧЕТВЕРТАЯ позиция > 0) 0 = три точки измерения при измерении под углом (обход сферы) 1 = четыре точки измерения при измерении под углом (обход сферы)
				СЕДЬМАЯ ПОЗИЦИЯ: определение заданного диаметра сферы 0 = без определения заданного диаметра сферы 1 = определение заданного диаметра сферы

№	Параметры масок	Параметры цикла	Значение	
2	Выбор	S_KNUM	Коррекция в смещение нулевой точки (WO), базовое смещение или базовое отношение [3]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ:
				ВТОРАЯ ПОЗИЦИЯ: 0 = нет коррекции 1 до макс. 99 номеров смещения нулевой точки или 1 до макс. 16 номеров базового смещения
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Коррекция в WO, базовое WO или базовое отношение 0 = коррекция в настраиваемое WO 1 = коррекция в спец. для канала базовое WO 2 = коррекция в базовое отношение 3 = коррекция в глобальное базовое WO [7] 9 = коррекция в активное WO или при G500 в последнее активное спец. для канала базовое WO
				ПЯТАЯ ПОЗИЦИЯ: Коррекция в WO, базовое WO или базовое отношение (грубая или точная) 0 = коррекция точная [6] 1 = коррекция грубая
3	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (диапазон значений: от 1 до 40)	
4		S_SETV	Диаметр сфер(ы) [4]	
5	DFA	S_FA	Путь измерения	
6	TSA	S_TSA	Доверительная область	
7	alpha 0	S_STA1	Начальный угол при измерении под углом	
8	alpha 1	S_INCA	Угловое приращение при измерении под углом	
9	X1	S_SETV0	Заданное положение 1-й сферы 1-й оси плоскости (при G17 X) при измерении 3 сфер	
10	Y1	S_SETV1	Заданное положение 1-й сферы 2-й оси плоскости (при G17 Y) при измерении 3 сфер	
11	Z1	S_SETV2	Заданное положение 1-й сферы 3-й оси плоскости (при G17 Z) при измерении 3 сфер	
12	X2	S_SETV3	Заданное положение 2-й сферы 1-й оси плоскости при измерении 3 сфер	
13	Y2	S_SETV4	Заданное положение 2-й сферы 2-й оси плоскости при измерении 3 сфер	
14	Z2	S_SETV5	Заданное положение 2-й сферы 3-й оси плоскости при измерении 3 сфер	
15	X3	S_SETV6	Заданное положение 3-й сферы 1-й оси плоскости при измерении 3 сфер	
16	Y3	S_SETV7	Заданное положение 3-й сферы 2-й оси плоскости при измерении 3 сфер	
17	Z3	S_SETV8	Заданное положение 3-й сферы 3-й оси плоскости при измерении 3 сфер	
18	TVL	S_TNVL	Предельное значение для искажения треугольника (сумма отклонений) при измерении 3 сфер [5]	
19	Измерения	S_NMSP	Число измерений на одном месте [2] (диапазон значений 1 до 9)	
20		S_MCBIT	зарезервировано	

№	Параметры масок	Параметры цикла	Значение	
21		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
22		_AMODE	Альтернативный режим	

1) все значения по умолчанию = 0 или обозначенные как диапазон значений a до b

2) Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Промежуточное позиционирование при обходе сферы по экватору

4) Измерение с 3 сферами: Для всех сфер действует одинаковый заданный диаметр (_SETV)

5) Значение по умолчанию для S_TNVL=1.2
Коррекция в WO: коррекция в WO выполняется, только если полученное значение искажения ниже предельного значения S_TNVL .

6) Если "точное" WO в MD не установлено, коррекцию выполняют по "грубому" WO.

7) При варианте измерения "Измерение трех сфер" коррекция в глобальный базовый фрейм невозможна (S_KNUM = 3001 до 3016), т.к. у фрейма отсутствует компонент вращения.

Параметры измерительного цикла CYCLE995

```
PROC CYCLE995 (INT S_MVAR,INT S_KNUM,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL S_TSA,REAL
S_STA1,REAL S_INCA,REAL S_DZ,REAL S_SETV0,REAL S_SETV1,REAL S_SETV2,REAL S_TUL,REAL
S_TZL,INT S_NMSP,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Таблица 4-18 Параметры вызывающей функции CYCLE995 [1)]

№	Параметры масок	Параметры цикла	Объяснение	
1		S_MVAR	Вариант измерения (default=5)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: элемент контура 5 = геометрия шпинделя (параллельность оси инструмента)
				ВТОРАЯ ПОЗИЦИЯ: повторение измерения 1 = с повторением измерения
				ТРЕТЬЯ ПОЗИЦИЯ: нет цели коррекции 0 = только измерение
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Стратегия измерения 2 = измерение под углом, точно установить измерительный щуп в направлении срабатывания
				ПЯТАЯ ПОЗИЦИЯ: число измеряемых сфер 0 = измерение сферы
				ШЕСТАЯ ПОЗИЦИЯ: число точек измерения 1 = 4 точки измерения при измерении под углом (обход сферы)
				СЕДЬМАЯ ПОЗИЦИЯ: определение заданного диаметра сферы 0 = без определения заданного диаметра сферы 1 = определение заданного диаметра сферы
2	Выбор	S_KNUM	цель коррекции 0 = 0	
3	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (диапазон значений: от 1 до 40)	
4	DM	S_SETV	Диаметр калибровочной сферы [4)]	
5	DFA	S_FA	Длина измерения	
6	TSA	S_TSA	Доверительная область [5)]	
7	alpha 0	S_STA1	Начальный угол при измерении под углом [3)]	
8		S_INCA	Угловое приращение при измерении под углом [2)]	
9	DZ	S_DZ	Расстояние 1-го измерения P1 относительно 2-го измерения P2 на стержне измерительного щупа	
10		S_SETV0	Заданное положение сферы 1-й оси плоскости (при G17 X) [2)]	
11		S_SETV1	Заданное положение сферы 2-й оси плоскости (при G17 Y) [2)]	
12		S_SETV2	Заданное положение сферы 3-й оси плоскости (при G17 Z) [2)]	
13	TUL	S_TUL	Верхняя граница допуска углового отклонения	
14	TZL	S_TZL	Область нулевой коррекции [1), 4)]	
15	Число	S_NMSP	Число измерений на одном месте [2)] (диапазон значений 1 до 9)	
16		S_MCBIT	зарезервировано [2)]	

№	Параметры масок	Параметры цикла	Объяснение	
17		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
18		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: допуск на размер да/нет 0 = нет 1 = да

Все значения по умолчанию = 0 или обозначенные как диапазон значений a до b

1) Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE.

2) Параметры в настоящее время не используются и не отображаются в маске ввода.
Параметр углового приращения S_INCA постоянно установлен ан 90 градусов.

3) Диапазон значений начального угла -360 до +360 градусов

4) при допуске на размер да:
Если измеренные углы меньше, чем значение области нулевой коррекции TZL, то результирующие параметры для углов (_OVR[2], _OVR[3]) и отклонений (_OVR[7], _OVR[8]) устанавливаются равными нулю.
TZL отображается через общие SD54760 $SNS_MEA_FUNCTION_MASK_PIECE Бит25=1.
(выбор разрешения коррекции нуля при измерении наклона шпинделя)

5) Параметр TSA относится к 1-му измерению калибровочной сферы.

Параметры измерительного цикла CYCLE996

```
PROC CYCLE996(INT S_MVAR,INT S_TC,INT S_PRNUM,REAL S_SETV,REAL S_STA1,REAL S_SETV0,REAL
S_SETV1,REAL S_SETV2,REAL S_SETV3,REAL S_SETV4,REAL S_SETV5,REAL S_TNVL,REAL S_FA,REAL
S_TSA,INT S_NMSP,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Таблица 4-19 Параметры вызывающей функции CYCLE996 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения (default=1)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: последовательность измерений 0 = расчет кинематики (выбор с: Индикация результатов, протокол, изменение блока данных поворота, при необходимости, с квитированием пользователем), см. _AMODE 1 = 1. Измерение 2 = 2. Измерение 3 = 3. Измерение
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано 0 = 0
				ТРЕТЬЯ ПОЗИЦИЯ: вариант измерения для 1-ого - 3-его измерения 0 = измерение калибровочной сферы параллельно осям 1 = измерение калибровочной сферы под углом без отслеживания шпинделя [3)] 2 = измерение калибровочной сферы и отслеживание шпинделя в направлении срабатывания измерительного щупа [3)] 3 = измерение параллельно осям, с начальным углом [8)] 4 = измерение параллельно осям, с начальным углом, отслеживание шпинделя в направлении срабатывания измерительного щупа [8)]
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: расчет цели коррекции для кинематики [4)] 0 = только измерение. Блоки данных поворота рассчитываются, но остаются без изменений 1 = рассчитать блок данных поворота. Блоки данных поворота изменяются, при необходимости после квитирования оператора [4)]
				ПЯТАЯ ПОЗИЦИЯ: измерительная ось (круговая ось 1 или 2) или разомкнутая/замкнутая векторная цепочка при расчете кинематики 0 = замкнутая векторная цепочка (только при расчете кинематики) 1 = круговая ось 1 (только для 1-го - 3-го измерения) 2 = круговая ось 2 (только для 1-го - 3-го измерения) [5)] 3 = разомкнутая векторная цепочка (только при расчете кинематики)
				ШЕСТАЯ ПОЗИЦИЯ: нормирование круговой оси 1 для расчета кинематики 0 = без нормирования круговой оси 1 1 = нормирование в направлении 1-й оси плоскости (для G17 X) 2 = нормирование в направлении 2-й оси плоскости (для G17 Y) 3 = нормирование в направлении 3-й Ось плоскости (для G17 Z)
				СЕДЬМАЯ ПОЗИЦИЯ: нормирование круговой оси 2 для расчета кинематики [5)] 0 = без нормирования круговой оси 2 1 = нормирование в направлении 1-й оси плоскости (для G17 X) 2 = нормирование в направлении 2-й оси плоскости (для G17 Y) 3 = нормирование в направлении 3-й Ось плоскости (для G17 Z)
				ВОСЬМАЯ ПОЗИЦИЯ: Журнал 0 = нет журнала 1 = журнал с рассчитанными векторами (Toolcarrier) и 1-й динамической 5-осевой трансформацией (TRAORI(1)), если установлена в MD.
2		S_TC	Номер блока данных поворота (Toolcarrier)	

№	Параметры масок	Параметры цикла	Значение	
3	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (default=1)	
4		S_SETV	Диаметр калибровочной сферы	
5	alpha 0	S_STA1	Начальный угол при измерении под углом	
6	alpha 0	S_SETV0	Значение позиции круговой оси 1 (для ручной или полуавтоматической круговой оси)	
7	alpha 1	S_SETV1	Значение позиции круговой оси 2 (для ручной или полуавтоматической круговой оси) [6]	
8	XN	S_SETV2	Значение позиции для нормирования круговой оси 1	
9	XN	S_SETV3	Значение позиции для нормирования круговой оси 2 [6]	
10	delta	S_SETV4	Значение допуска векторов смещения I1 до I4	
11	delta	S_SETV5	Значение допуска векторов круговых осей V1 и V2	
12	TVL	S_TNVL	Предельное значение углового сегмента круговой оси (диапазон значений 1 до 60 градусов), (default=20) [7]	
13	DFA	S_FA	Путь измерения	
14	TSA	S_TSA	Доверительная область	
15	Измерения	S_NMSP	Число измерений на одном месте [2] (default=1)	
16		S_MCBIT	зарезервировано	
17		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
18		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: контроль допуска да/нет 0 = нет 1 = да: обработка значений допуска векторов S_SETV4, S_SETV5
				ВТОРАЯ ПОЗИЦИЯ: квитирование оператора при вводе вычисленных векторов в блок данных поворота [4] 0 = да: оператор должен квитировать изменения 1 = нет: вычисленные векторы вносятся сразу же (действует только при ТРЕТЬЕЙ и ЧЕТВЕРТОЙ позиции = 0)
				ТРЕТЬЯ ПОЗИЦИЯ: индикация результата измерения [5] 0 = нет 1 = да
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: возможность редактирования индикации результата измерения 0 = нет 1 = да, и возможность редактирования (действует только при ТРЕТЬЕЙ позиции = 1)

[1] все значения по умолчанию = 0 или обозначенные как default=x

[2] Отображение в зависимости от общих SD54760 $SNS_MEA_FUNCTION_MASK_PIECE.

[3] С помощью этого варианта возможно, к примеру, измерение на калибровочной сфере в 90-градусных позициях кинематики, без столкновений к крепежным стержнем калибровочной сферы. Начальный угол S_STA1 (0 до

360 градусов) может быть задан. Угловое приращение при обходе сферы равно 90 градусам.
В качестве подачи на круговой траектории используются спец. для канала SD55634 $SCS_MEA_FEED_PLANE_VALUE

[4] Перед вводом выполняется запрос оператору с M0. Только с NC-Start векторы вводятся.
Если программа измерений отменяется с RESET , то вычисленные векторы не вносятся.
Векторы вносятся только в том случае, если допуск векторов смещения при расчете не превышен.

[5] Индикация результатов измерения только для варианта измерения вычисления кинематики.
Если необходимо отобразить результат измерения и после 1-го - 3-го измерения, это осуществляется через ввод спец. для канала SD55613 $SCS_MEA_RESULT_DISPLAY.

[6] Круговая ось 2 только в кинематике с двумя круговыми осями

[7] Предельное значение углового сегмента круговой оси. Диапазон значений `S_TNVL` между 20 и 60 градусами. При значениях `S_TNVL` < 20 градусов возможны неточности, обусловленные погрешностями измерения в микронном диапазоне измерительного щупа. Если предельное значение превышено, следует сообщение об ошибке 61430 с индикацией мин. значения для предельного значения.

[8] Отслеживание шпинделя в направлении срабатывания измерительного щупа, если SD54760 бит 17 = 1

Параметры измерительного цикла CYCLE9960

```
PROC CYCLE9960(INT S_MVAR,STRING[40] S_TNAME,INT S_PRNUM,REAL S_SETV,REAL S_SETV1,REAL
S_START_RA1,REAL S_END_RA1,INT S_CMEA_RA1,REAL S_POS_RA2,REAL S_SETV2,REAL
S_START_RA2,REAL S_END_RA2,INT S_CMEA_RA2,REAL S_POS_RA1,REAL S_SETV4,REAL S_FA,REAL
S_TSA,INT S_NMSP,INT S_DMODE,INT S_AMODE,INT S_KNUM)
```

Таблица 4-20 Параметры вызывающей функции CYCLE9960 [1)]

№	Параметры масок	Параметры цикла	Значение	
1	Выбор	S_MVAR	Вариант измерения (default=1)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: Вариант измерения 0 = измерение и расчет кинематики (выбор с: индикация результатов, протокол, изменение блока данных качания (см. четвертую позицию S_MVAR)) 1 = Измерение эталонной головки 2 = Коррекция головки по эталонной головке 3 = Измерение и расчет опорных точек (E996)
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ: Вариант измерения при измерении сферы 2 = измерение калибровочной сферы и отслеживание шпинделя в направлении срабатывания измерительного щупа 4 = измерение параллельно осям, с начальным углом, отслеживание шпинделя в направлении срабатывания измерительного щупа [2)]
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Расчет цели коррекции для кинематики [3)] 0 =измерение и расчет. Блоки данных рассчитываются и остаются без изменений 1 = Рассчитанные блоки данных изменяются при необходимости после квитирования оператором 2 = Уже измеренная кинематика рассчитывается и изменяется при необходимости после квитирования оператором.
				ПЯТАЯ ПОЗИЦИЯ: измерительная ось (круговая ось 1 или 2) 1 = Измерение и расчет всех имеющихся круговых осей 4 = Измерение и расчет только круговой оси 1 5 = Измерение и расчет только круговой оси 2
2		S_TNAME	Имя трансформации (блок данных качания или трансформации на основе кинематических цепей)	
3	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (default=1)	
4		S_SETV	Диаметр калибровочной сферы	
5	alpha 1	S_SETV1	Начальный угол при измерении под углом для 1-й круговой оси	
6			S_START_RA1 : Начальный угол 1-й круговой оси	
7			S_END_RA1 : Конечный угол 1-й круговой оси	
8			S_CMEA_RA1 : Количество измерений 1-й круговой оси, 3 для Измерение и расчет кинематики. Для опорных точек возможно до 12 измерений.	
9			S_POS_RA2 : Позиция 2-й круговой оси при измерении 1-й круговой оси [4)]	
10	alpha 2	S_SETV2	Начальный угол при измерении под углом для 2-й круговой оси [4)]	
11			S_START_RA2 : Начальный угол 2-й круговой оси [4)]	
12			S_END_RA2 : Конечный угол 2-й круговой оси [4)]	
13			S_CMEA_RA2 : Количество измерений 2-й круговой оси, 3 для Измерение и расчет кинематики. Для опорных точек возможно до 12 измерений.[4)]	
14			S_POS_RA1 : Позиция 1-й круговой оси при измерении 2-й круговой оси	
15	delta	S_SETV4	Значение допуска векторов смещения	
16	DFA	S_FA	Длина измерения	

№	Параметры масок	Параметры цикла	Значение	
17	TSA	S_TSA	Доверительная область	
18	Число	S_NMSP	Число измерений на одном месте [5] (default=1)	
19		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: Плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
20		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: Контроль допусков: да/нет 0 = нет 1 = да: Обработка значений допуска векторов -> S_SETV4, S_SETV5
				ВТОРАЯ ПОЗИЦИЯ: автоматическая калибровка 0 = нет автоматической калибровки 1 = автоматическая калибровка
				ТРЕТЬЯ ПОЗИЦИЯ: Автоматика начального угла 0 = используется настроенный начальный угол 1 = автоматический расчет начального угла, в каждой точке измерения
21		S_KNUM	Номер подлежащего коррекция смещения нулевой точки для «Измерение эталонной головки»	
			Значения	ПЕРВАЯ ПОЗИЦИЯ: Коррекция всегда вносится в грубое смещение, точное смещение удаляется
				ВТОРАЯ ПОЗИЦИЯ: 1..99 Номер настраиваемого смещения нулевой точки (1=G54)
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: 9 коррекция, активное настраиваемое смещение нулевой точки

[1] Все значения по умолчанию = 0 или обозначенные как default=xx

[2] Отслеживание шпинделя в направлении срабатывания измерительного щупа, если SD54760 бит 17 = 1

[3] Перед вводом выполняется запрос оператору с M0. Векторы вносятся только при пуске ЧПУ. Если программа измерений прерывается при помощи сброса (RESET), рассчитанные векторы не вносятся. Векторы вносятся, только если допуск векторов смещения при расчете не превышен.

[4] Круговая ось 2 только в кинематиках с двумя круговыми осями

[5] Отображение в зависимости от общей SD54760 $SNS_MEA_FUNCTION_MASK_PIECE.

Параметры измерительного цикла CYCLE982

```
PROC CYCLE982(INT S_MVAR,INT S_KNUM,INT S_PRNUM,INT S_MA,INT S_MD,REAL S_ID,REAL S_FA,REAL
S_TSA,REAL S_VMS,REAL S_STA1,REAL S_CORA,REAL S_TZL,REAL S_TDIF,INT S_NMSP,INT S_EVNUM,INT
S_MCBIT,INT _DMODE,INT _AMODE)
```

Таблица 4-21 Параметры вызывающей функции CYCLE982 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: калибровка / измерение 0 = калибровка измерительного щупа инструмента 1 = отдельное измерение инструмента [3)] 2 = многократное измерение инструмента, определение длин и радиуса инструмента (для фрезерного инструмента)
				ВТОРАЯ ПОЗИЦИЯ: калибровка или измерение в MCS или WCS 0 = по отношению к станку [4)] 1 = по отношению к детали
				ТРЕТЬЯ ПОЗИЦИЯ: измерение с или без реверса для фрезерных инструментов 0 = измерение без реверса 1 = измерение с реверсом
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: цель коррекции для фрезерных инструментов 0 = определение длины или длины и радиуса (см. S_MVAR 1-я позиция) 1 = определение радиуса, если S_MVAR 1-я позиция = 1 2 = определение длины и радиуса (торец), если S_MVAR 1-я позиция = 1 или 2 3 = определение верхнего диска (задняя сторона) дисковой фрезы и длины и радиуса [5)]
				ПЯТАЯ ПОЗИЦИЯ: положение фрезерного инструмент или сверла 0 = осевое положение фрезерного инструмента или сверла, радиус во 2-й оси плоскости (для G18 X) [7)] 1 = радиальное положение фрезерного инструмента или сверла, радиус в 1-й оси плоскости (для G18 Z) [7)]
				ШЕСТАЯ ПОЗИЦИЯ: инкрементальная калибровка или измерение 0 = нет данных 1 = инкрементальная калибровка или измерение
				СЕДЬМАЯ ПОЗИЦИЯ: позиционировать шпиндель на начальный угол S_STA1 (только при измерении фрезерного инструмента) 0 = не позиционировать шпиндель 1 = позиционировать шпиндель на начальный угол S_STA1
2	Выбор	S_KNUM	Вариант коррекции [2)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: Коррекция на инструмент 0 = нет данных (коррекция инструмента в геометрию) 1 = коррекция инструмента в износ
3	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа) (default=1)	
4	X0	S_MA	Измерительная ось	
			Значения:	1 = 1-я ось плоскости (для G18 Z) 2 = 2-я ось плоскости (для G18 X)
5	+-	S_MD	Направление измерения	
			Значения:	0 = без выбора (направление измерения определяется из фактического значения) 1 = положительное 2 = отрицательное

№	Параметры масок	Параметры цикла	Значение	
6	Z2	S_ID	Смещение	
7	DFA	S_FA	Путь измерения	
8	TSA	S_TSA	Доверительная область	
9	VMS	S_VMS	Переменная скорость измерения при калибровке [2)]	
10	alpha1	S_STA1	Начальный угол при измерении фрезерных инструментов	
11	alpha2	S_CORA	Угол поправки при измерении фрезерных инструментов с реверсом [8)]	
12	TZL	S_TZL	Коррекция нуля при измерении инструментов. При калибровке S_TZL = 0	
13	DIF	S_TDIF	Контроль разности размеров	
14	Измерения	S_NMSP	Число измерений на одном месте [2)] (default=1)	
15	EVN	S_EVNUM	Номер памяти эмпирических средних значений [2), 9)]	
16		S_MCBIT	зарезервировано	
17		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)
				ВТОРАЯ ПОЗИЦИЯ: положение резцов у токарных и фрезерных инструментов (только для индикации в масках ввода 1 до 9)
				ТРЕТЬЯ ПОЗИЦИЯ: Тип инструмента 0 = токарный инструмент 1 = фреза 2 = сверло
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Стратегия подвода относительно измерительного щупа инструмента 0 = ПЛЮС [X/Z]; X при осевом расположении инструмента, Z при радиальном расположении инструмента 1 = МИНУС [X/Z]; X при осевом расположении инструмента, Z при радиальном расположении инструмента
18		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: зарезервировано
				ВТОРАЯ ПОЗИЦИЯ: зарезервировано
				ТРЕТЬЯ ПОЗИЦИЯ: зарезервировано
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: Подвод к исходной позиции после процесса измерения при калибровке и отдельном измерении (см. S_MVAR - EINER) 0 = инструмент установлен со смещением на DFA относительно кромки измерительного щупа 1 = Подвод к исходной позиции

[1)] все значения по умолчанию = 0 или обозначенные как default=x

[2)] Отображение в зависимости от общих SD54762 _MEA_FUNCTION_MASK_TOOL

[3)] Измерение токарного или фрезерного инструмента или сверла. Измерительная ось в параметре S_MA Спецификация для токарных инструментов через положение резцов 1...8, у фрезерных инструментов через ТРЕТЬЮ и ЧЕТВЕРТУЮ позицию в параметре S_MVAR.

[4] Измерение и калибровка выполняются в базовой кинематической системе (MCS при отключенной кинематической трансформации).

[5] Не для инкрементального измерения

[6] Только для многократного измерения `S_MVAR=x2x02` или `x3x02` (пример дисковой или пазовой фрезы)

[7] Если спец. для канала SD42950 $SC_TOOL_LENGTH_TYPE = 2, то компоненты длин инструмента согласуются как у токарных инструментов

[8] Только при измерении с реверсом `S_MVAR=xx1x1`

[9] Формирование эмпирического значения
Диапазон памяти эмпирических значений: 1 до 20 номер (n) памяти эмпирических значений см. спец. для канала SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1].

Параметры измерительного цикла CYCLE971

```
PROC CYCLE971(INT S_MVAR,INT S_KNUM,INT S_PRNUM,INT S_MA,INT S_MD,REAL S_ID,REAL S_FA,REAL
S_TSA,REAL S_VMS,REAL S_TZL,REAL S_TDIF,INT S_NMSP,REAL S_F1,REAL S_S1,REAL S_F2,REAL
S_S2,REAL S_F3,REAL S_S3,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Таблица 4-22 Параметры вызывающей функции CYCLE971 [1)]

№	Параметры масок	Параметры цикла	Значение	
1		S_MVAR	Вариант измерения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: 0 = калибровка измерительного щупа инструмента 1 = измерение инструмента с остановленным шпинделем (длина или радиус) 2 = измерение инструмента с вращающимся шпинделем (длина или радиус), см. параметр S_F1 до S_S4
				ВТОРАЯ ПОЗИЦИЯ: измерение в MCS или WCS 0 = измерение в MCS (по отношению к станку), измерение инструмента или калибровка измерительного щупа инструмента 1 = измерение в WCS (по отношению к детали), измерение инструмента или калибровка измерительного щупа инструмента
				ТРЕТЬЯ ПОЗИЦИЯ: Индивидуальная проверка зубьев 0 = нет 1 = да
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: 0 = 0
				ПЯТАЯ ПОЗИЦИЯ: инкрементальная калибровка или измерение 0 = нет данных 1 = инкрементальная калибровка или измерение
				ШЕСТАЯ ПОЗИЦИЯ: автоматическая калибровка измерительного щупа инструмента 0 = нет автоматический калибровки измерительного щупа инструмента 1 = автоматическая калибровка измерительного щупа инструмента
				СЕДЬМАЯ ПОЗИЦИЯ: калибровка в плоскости с реверсом шпинделя 0 = калибровка в плоскости без реверса шпинделя 1 = калибровка в плоскости с реверсом шпинделя
2	Выбор	S_KNUM	Вариант коррекции [2)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: Коррекция на инструмент 0 = нет данных (коррекция инструмента в геометрию) 1 = коррекция инструмента в износ
3	пикт.+ число	S_PRNUM	Номер массива параметров измерительного щупа (не номер измерительного щупа)	
4	X0	S_MA	Измерительная ось, смещенная ось [4)]	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: номер измерительной оси 1 = 1-я ось плоскости (для G17 X) 2 = 2-я ось плоскости (для G17 Y) 3 = 3-я Ось плоскости (для G17 Z)
				ВТОРАЯ ПОЗИЦИЯ: 0 = 0
				ТРЕТЬЯ ПОЗИЦИЯ: номер смещенной оси 0 = несмещенная ось 1 = 1-я ось плоскости (для G17 X) 2 = 2-я ось плоскости (для G17 Y)

№	Параметры масок	Параметры цикла	Значение	
5	+-	S_MD	Направление измерения	
			Значения:	0 = без выбора (направление измерения определяется из фактического значения) 1 = положительное 2 = отрицательное
6	V	S_ID	Смещение	
			Значения:	0 = для инструментов без смещения >0 = • Калибровка: Смещение сказывается на 3-й оси плоскости (при G17 Z), если диаметр калибровочного инструмента больше верхнего значения диаметра измерительного щупа. Здесь инструмент смещается из центра измерительного щупа на радиус инструмента, минус значение из S_ID. Смещенная ось дополнительно указывается в S_MA . • Измерение: У инструмента с несколькими резцами указать смещение длины инструмента и высшей точки резца при измерении радиуса или смещение радиуса инструмента к высшей точке резца при измерении длины инструмента.
7	DFA	S_FA	Путь измерения	
8	TSA	S_TSA	Доверительная область	
9	VMS	S_VMS	Переменная скорость измерения при калибровке [2)]	
10	TZL	S_TZL	Коррекция нуля (только при измерении инструмента)	
11	DIF	S_TDIF	Контроль разности размеров при измерении инструмента (S_MVAR=xx1 или S_MVAR=xx2)	
12	Измерения	S_NMSP	Число измерений на одном месте [2)]	
13	F1	S_F1	1-я подача для касания с вращающимся шпинделем [2)]	
14	S1	S_S1	1-я частота вращения для касания с вращающимся шпинделем [2)]	
15	F2	S_F2	2-я подача для касания с вращающимся шпинделем [2)]	
16	S2	S_S2	2-я частота вращения для касания с вращающимся шпинделем [2)]	
17	F3	S_F3	3-я подача для касания с вращающимся шпинделем [2)]	
18	S3	S_S3	3-я частота вращения для касания с вращающимся шпинделем [2)]	
19	EVN	S_EVNUM	Номер памяти эмпирических значений [2)]	
20		S_MCBIT	Маска _CBITs или _CHBITs	
21		_DMODE	Режим отображения	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: плоскость обработки G17/G18/G19 0 = совместимость, сохраняется активность плоскости, активной перед вызовом цикла 1 = G17 (активность только в цикле) 2 = G18 (активность только в цикле) 3 = G19 (активность только в цикле)

№	Параметры масок	Параметры цикла	Значение	
22		_AMODE	Альтернативный режим	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: измерение смещения инструмента по радиусу 1 = нет 2 = да
				ВТОРАЯ ПОЗИЦИЯ: измерение направления смещения инструмента по радиусу в 3-й оси плоскости (при G17 Z) 1 =положительное 2 = отрицательное
				ТРЕТЬЯ ПОЗИЦИЯ: смещение инструмента при измерении длины или при калибровке измерительного щупа в 3-й оси 0 = совместимость, авто 1 = нет 2 = да
				ЧЕТВЕРТАЯ ПОЗИЦИЯ: направление смещения инструмента при измерении длины по оси смещения (cv/ S_MA HUNDERTER) 1 =положительное 2 = отрицательное

1) все значения по умолчанию = 0 или обозначенные как default=x

2) Отображение в зависимости от общих SD54762 MEA_FUNCTION_MASK_TOOL

3) Только при коррекции в инструменте и допуске на размер "да", в ином случае параметр = 0

4) При автоматическом измерении (S_MVAR=1x00xx) измерительная ось, смещенная ось не отображается ⇒ S_MA=0.

Параметры измерительного цикла CYCLE150

```
PROC CYCLE150(INT S_PICT,INT S_PROT,STRING[160] S_PATH) SAVE
ACTBLOCNO DISPLOF
```

Таблица 4-23 Параметры вызывающей функции CYCLE150

№	Параметры масок	Параметры цикла	Значение	
1	Окно результатов измерения	S_PICT	Выбор индикации результатов (по умолчанию = 0)	
			Значения:	ПЕРВАЯ ПОЗИЦИЯ: 0 = окно результатов измерения ВЫКЛ 1 = окно результатов измерения ВКЛ
				ВТОРАЯ ПОЗИЦИЯ: Выбор режима индикации (значения аналогично SD55613) 1 = отображение окна результатов измерения - автоматическое закрытие через 8 с 3 = отображение окна результатов измерения - квитирование при пуске ЧПУ 4 = отображение окна результатов измерения - только для предупреждений (61303 ... 61306)
2		S_PROT	Выбор ведения журнала (по умолчанию = 0)	

№	Параметры масок	Параметры цикла	Значение	
	Журнал		Значения:	ПЕРВАЯ ПОЗИЦИЯ: Выбор журнал ВЫКЛ / ВКЛ / последнее измерение 0 = журнал ВЫКЛ 1 = журнал ВКЛ 2 = запись последнего измерения
	Тип журнала			ВТОРАЯ ПОЗИЦИЯ: Выбор типа журнала 0 = стандартный журнал 1 = пользовательский журнал (произвольно определяемый)
	Формат журнала			ТРЕТЬЯ ПОЗИЦИЯ: Выбор формата журнала 0 = текстовый формат 1 = табличный формат (для импорта в Excel)
	Данные журнала			ЧЕТВЕРТАЯ ПОЗИЦИЯ: Выбор перезаписи или добавления 0 = создать 1 = добавить
	Журнал Место сохранения			ПЯТАЯ ПОЗИЦИЯ: Выбор места хранения журнала 0 = в виде УП 1 = директория
3		S_PATH	Путь к файлу журнала в соответствии с выбором места хранения журнала (указание полного пути или только имени файла, например: "//NC:/WKS.DIR/NAME.WPD" или "MESSPROTOKOLL.TXT"	

Таблицы

5

5.1 Операторы

Примечание

Циклы

В списке операторов содержатся все циклы, которые могут появиться в программе ЧПУ (G-код), т. е. которые можно программировать в редакторе программ через маски или которые требуется программировать при шлифовании без программной поддержки. Не учтены циклы, все еще представленные в системе управления из соображений совместимости, но которые больше не поддаются программированию в редакторе программ SINUMERIK Operate ("Циклы совместимости").

Операторы A ... C

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
:	O	Номер главного кадра ЧПУ, конец метки перехода, связывающий оператор		+		PM-NC
*	O	Оператор для умножения		+		PM-NC
+	O	Оператор для сложения		+		PM-NC
-	O	Оператор для вычитания		+		PM-NC
<	O	Оператор сравнения, меньше		+		PM-NC
<<	O	Связывающий оператор для строк		+		PM-NC
<=	O	Оператор сравнения, меньше равно		+		PM-NC
=	O	Оператор присваивания		+		PM-NC
>=	O	Оператор сравнения, больше равно		+		PM-NC
/	O	Оператор для деления		+		PM-NC
/0 ... ... /7		Кадр пропускается (1-й уровень пропуска)° ... ... Кадр пропускается (8-й уровень пропуска)		+		PM-NC
A	A	Имя оси	м/п	+		PM-NC
A2	A	Ориентация инструмента: Угол RPY или эйлеров угол	п	+		PM-NC
A3	A	Ориентация инструмента: 1. компонента вектора направления	п	+		PM-NC
A4	A	Ориентация инструмента: 1. компонента вектора нормали к плоскости на начале кадра	п	+		PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
A5	A	Ориентация инструмента: 1. компонента вектора нормали к плоскости на конце кадра	п	+		PM-NC
A6	A	Ориентация инструмента: 1. компонента вектора направления для оси вращения конуса	п	+		PM-NC
A7	A	Ориентация инструмента: 1. компонента вектора для промежуточной ориентации на боковой поверхности конуса	п	+		PM-NC
ABS	F	Абсолютное значение (величина)		+	+	PM-NC
AC	K	Абсолютное указание размеров координат/позиций	п	+		PM-NC
ACC	K	Управление актуальным осевым ускорением	м	+	+	PM-NC
ACCLIMA	K	Управление актуальным максимальным осевым ускорением	м	+	+	PM-NC
ACN	K	Абсолютное указание размера для круговых осей, подвод к позиции в отрицательном направлении	п	+		PM-NC
ACOS	F	Арккосинус (тригонометрическая функция)		+	+	PM-NC
ACP	K	Абсолютное указание размера для круговых осей, подвод к позиции в положительном направлении	п	+		PM-NC
ACTBLOCNO	P	Вывод актуального номера кадра ошибки, даже в случае активности "актуальная индикация кадра подавлена" (DISPLOF)!		+		PM-NC
ADDFRAME	F	Учет и возможная активация измеренного фрейма		+	-	PM-NC, FM-B
ADIS	A	Интервал перешлифовки для траекторных функций G1, G2, G3, ...	м	+		PM-NC
ADISPOS	A	Интервал перешлифовки для ускоренного хода G0	м	+		PM-NC
ADISPOSA	P	Размер окна допуска для IPOBRKA	м	+	+	PM-NC
AFISOF	P	Выключить автоматическое переключение цепочек фильтров	м	+	-	PM-NC
AFISON	P	Включить автоматическое переключение цепочек фильтров	м	+	-	PM-NC
ALF	A	Угол быстрого отвода	м	+		PM-NC
AMIRROR	G	Программируемое отражение	п	+		PM-NC
AND	K	Логическая И		+		PM-NC
ANG	A	Угол линии контура	п	+		PM-NC
AP	A	Полярный угол	м/п	+		PM-NC
APR	K	Чтение / индикация защиты доступа		+		PM-NC
APRB	K	Чтение права доступа, BTSS		+		PM-NC
APRP	K	Чтение права доступа, программа обработки детали		+		PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
APW	K	Запись защиты доступа		+		PM-NC
APWB	K	Запись права доступа, BTSS		+		PM-NC
APWP	K	Запись права доступа, программа обработки детали		+		PM-NC
APX	K	Определение защиты доступа для выполнения указанного элемента языка		+		PM-NC
AR	A	Апертурный угол	м/п	+		PM-NC
AROT	G	Программируемое вращение	п	+		PM-NC
AROTS	G	Программируемые вращения фреймов с пространственными углами	п	+		PM-NC
AS	K	Макро-определение		+		PM-NC
ASCALE	G	Программируемое масштабирование	п	+		PM-NC
ASIN	F	Математическая функция, арксинус		+	+	PM-NC
ASPLINE	G	Акима-сплайн	м	+		PM-NC
ATAN2	F	Арктангенс2		+	+	PM-NC
ATOL	A	Спец. для оси допуск для функций компрессора, сглаживания ориентации и типов перешлифовки	м	+		PM-NC
ATRANS	G	Аддитивное программируемое смещение нулевой точки	п	+		PM-NC
AUXFUDEL	P	Удалить вспомогательную функцию спец. для канала из глобального списка		+	-	FM-B
AUXFUDELG	P	Удалить все вспомогательные функции группы вспомогательных функций спец. для канала из глобального списка		+	-	FM-B
AUXFUMSEQ	P	Определить последовательность вывода для вспомогательных функций M		+	-	FM-B
AUXFUSYNC	P	Создать из глобального списка вспомогательных функций полный кадр программы обработки детали для спец. для канала SERUPRO-Ende-ASUP		+	-	FM-B
AX	K	Переменный идентификатор оси	м/п	+		PM-NC
AXCTSWE	P	Вращать осевой контейнер		+	-	PM-NC
AXCTSWEC	P	Отмена разрешения на вращение осевого контейнера		+	+	PM-NC
AXCTSWED	P	Вращать осевой контейнер (командная переменная для ввода в эксплуатацию!)		+	-	PM-NC
AXIS	K	Идентификатор оси, адрес оси		+		PM-NC
AXNAME	F	Преобразует входную строку в идентификатор оси		+	-	PM-NC
AXSTRING	F	Преобразует строку номер шпинделя		+	-	PM-NC
AXTOCHAN	P	Затребовать ось для определенного канала. Возможно из программы ЧПУ и из синхронного действия.		+	+	PM-NC
AXTOSPI	F	Преобразует идентификатор оси в индекс шпинделя		+	-	PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
B	A	Имя оси	м/п	+		PM-NC
B2	A	Ориентация инструмента: Угол RPY или эйлеров угол	п	+		PM-NC
B3	A	Ориентация инструмента: компонента вектора направления	п	+		PM-NC
B4	A	Ориентация инструмента: 2. компонента вектора нормали к плоскости на начале кадра	п	+		PM-NC
B5	A	Ориентация инструмента: 2. компонента вектора нормали к плоскости на конце кадра	п	+		PM-NC
B6	A	Ориентация инструмента: 2. компонента вектора направления для оси вращения конуса	п	+		PM-NC
B7	A	Ориентация инструмента: 2. компонента вектора для промежуточной ориентации на боковой поверхности конуса	п	+		PM-NC
B_AND	O	Побитовая И		+		PM-NC
B_OR	O	Побитовая ИЛИ		+		PM-NC
B_NOT	O	Побитовое отрицание		+		PM-NC
B_XOR	O	Побитовая исключающая "ИЛИ"		+		PM-NC
BAUTO	G	Определение первого сегмента сплайна через следующие 3 точки	м	+		PM-NC
BLOCK	K	Определяет вместе с кодовым словом TO обрабатываемый программный блок в косвенном вызове подпрограммы		+		PM-NC
BLSYNC	K	Выполнение обработчика прерываний должно начаться только при следующей смене кадра		+		PM-NC
BNAT [6]	G	Натуральный переход к первому сплайн-кадру	м	+		PM-NC
BOOL	K	Тип данных: Значение истинности TRUE/FALSE или 1/0		+		PM-NC
BOUND	F	Проверяет, находится ли значение в пределах определенного диапазона значений. Тождество возвращает проверяемое значение.		+	+	PM-NC
BRISK [6]	G	Скачкообразное ускорение по траектории	м	+		PM-NC
BRISKA	P	Включение скачкообразного ускорения по траектории для запрограммированных осей		+	-	PM-NC
BSPLINE	G	B-сплайн	м	+		PM-NC
BTAN	G	Тангенциальный переход к первому сплайн-кадру	м	+		PM-NC
C	A	Имя оси	м/п	+		PM-NC
C2	A	Ориентация инструмента: Угол RPY или эйлеров угол	п	+		PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
C3	A	Ориентация инструмента: 3. компонента вектора направления	п	+		PM-NC
C4	A	Ориентация инструмента: 3. компонента вектора нормали к плоскости на начале кадра	п	+		PM-NC
C5	A	Ориентация инструмента: 3. компонента вектора нормали к плоскости на конце кадра	п	+		PM-NC
C6	A	Ориентация инструмента: 3. компонента вектора направления для оси вращения конуса	п	+		PM-NC
C7	A	Ориентация инструмента: 3. компонента вектора для промежуточной ориентации на боковой поверхности конуса	п	+		PM-NC
CAC	K	Абсолютный подвод к позиции		+		PM-NC
CACN	K	Абсолютный подвод к зафиксированному в таблице значению в отрицательном направлении		+		PM-NC
CACP	K	Абсолютный подвод к зафиксированному в таблице значению в положительном направлении		+		PM-NC
CADAPTOF	P	Выключить коррекцию нагрузки		+	-	PM-NC
CADAPTON	P	Включить коррекцию нагрузки		+	-	PM-NC
CALCDAT	F	Вычисляет радиус и центр окружности из 3 или 4 точек		+	-	PM-NC
CALCFIR	P	Корректирование FIR-фильтра по динамическому режиму		+	-	PM-NC
CALCPOSI	F	Проверка нарушения защищенной области, ограничения рабочего поля и программных лимитов		+	-	PM-NC
CALL	K	Косвенный вызов подпрограммы		+		PM-NC
CALLPATH	P	Программируемый маршрут поиска для вызовов подпрограмм		+	-	PM-NC
CANCEL	P	Отменить модальное синхронное действие		+	-	FM-SA
CANCELSUB	P	Отменить актуальный уровень подпрограммы		+	+	FM-SA
CASE	K	Условное ветвление программы		+		PM-NC
CDC	K	Прямой подвод к позиции		+		PM-NC
CDOF [6]	G	Выключить контроль столкновений	м	+		PM-NC
CDOF2	G	Выключить контроль столкновений, при периферийном 3D-фрезеровании	м	+		PM-NC
CDON	G	Включить контроль столкновений	м	+		PM-NC
CFC [6]	G	Постоянная подача на контуре	м	+		PM-NC
CFIN	G	Постоянная подача только для внутреннего изгиба, не для наружного изгиба	м	+		PM-NC
CFINE	F	Присвоение точного смещения ФРЕЙМ-переменной		+	-	PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
CFTCP	G	Постоянная подача в исходной точке резцов инструмента, траектория центра	м	+		PM-NC
Chan	K	Спецификация сферы действия данных		+		PM-NC
CHANDATA	P	Установить номера канала для обращения к данным канала		+	-	PM-NC
CHAR	K	Тип данных: Символ ASCII		+		PM-NC
CHF	A	Фаска; значение = длина фаски	п	+		PM-NC
CHKDM	F	Проверка однозначности в магазине		+	-	FM-TM
CHKDNO	F	Проверка однозначности номеров D		+	-	PM-NC
CHR	A	Фаска; значение = длина фаски в направлении движения		+		PM-NC
CIC	K	Инкрементальный подвод к позиции		+		PM-NC
CIP	G	круговая интерполяция через промежуточную точку	м	+		PM-NC
CLEARM	P	Сброс одной/нескольких меток для координации канала		+	+	PM-NC
CLRINT	P	Выключение прерываний		+	-	PM-NC
CMIRROR	F	отражение на оси координат		+	-	PM-NC
COARSEA	K	Конец движения при достижении "Точного останова грубого"	м	+		PM-NC
COLLPAIR	F	Проверка на отношение к соударяемой паре		+		PM-NC
COMPCAD	G	Включить функцию компрессора COMPCAD	м	+		PM-NC
COMPCURV	G	Включить функцию компрессора COMPCURV	м	+		PM-NC
COMPLETE		Управляющий оператор для выгрузки и загрузки данных		+		PM-NC
COMPOF [6]	G	Выключить сжатие кадров УП	м	+		PM-NC
COMPON	G	Включить функцию компрессора COMPON	м	+		PM-NC
COMPSURF	G	Включить функцию компрессора COMPSURF	м	+		PM-NC
CONTDCON	P	Включить декодирование контура в табличной форме		+	-	PM-NC
CONTPRON	P	Включение эталонной подготовки		+	-	PM-NC
CORROF	P	Все активные наложенные движения отключаются.		+	-	PM-NC
CORRTC	F	Модифицирование векторов смещения или векторов направления ориентируемых инструментальных суппортов после замеров станка.		+	-	PM-NC
CORRTRAFO	F	Изменение векторов смещения или векторов осей ориентации в кинематической модели станка		+	-	PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
COS	F	Косинус (тригон. функция)		+	+	PM-NC
COUPDEF	P	Определение структуры электронного редуктора / соединения синхронных шпинделей		+	-	PM-NC
COUPDEL	P	Удалить структуру электронного редуктора		+	-	PM-NC
COUPOF	P	Выключить структуру электронного редуктора / синхронную шпиндельную пару		+	-	PM-NC
COUPOFS	P	Отключение структуры электронного редуктора / синхронной шпиндельной пары с остановкой ведомого шпинделя		+	-	PM-NC
COUPON	P	Включить структуру электронного редуктора / синхронную шпиндельную пару		+	-	PM-NC
COUPONC	P	Включение структуры электронного редуктора/пары синхронных шпинделей с передачей предшествующего программирования		+	-	PM-NC
COUPRES	P	Сбросить структуру электронного редуктора		+	-	PM-NC
CP [6]	G	Движение по траектории	м	+		PM-NC
CPBC	K	Базовое соединение: Критерий смены кадров		+	+	FM-A
CPDEF	K	Базовое соединение: Создание модуля сопряжения		+	+	FM-A
CPDEL	K	Базовое соединение: Удаление модуля сопряжения		+	+	FM-A
CPFMOF	K	Базовое соединение: Поведение ведомой оси при полном выключении		+	+	FM-A
CPFMON	K	Базовое соединение: Поведение ведомой оси при включении		+	+	FM-A
CPFMSON	K	Базовое соединение: Режим синхронизации		+	+	FM-A
CPFPOS	K	Базовое соединение: синхронная позиция ведомой оси		+	+	FM-A
CPFRS	K	Базовое соединение: Исходная система координат		+	+	FM-A
CPLA	K	Базовое соединение: Определение ведущей оси		+	-	FM-A
CPLCTID	K	Базовое соединение: Номер таблицы кривых		+	+	FM-A
CPLDEF	K	Базовое соединение: Определение ведущей оси и создание модуля сопряжения		+	+	FM-A
CPLDEL	K	Базовое соединение: Удаление ведущей оси модуля сопряжения		+	+	FM-A
CPLDEN	K	Базовое соединение: Знаменатель коэффициента связи		+	+	FM-A

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
CPLINSC	K	Базовое соединение: Коэффициент масштабирования для входного значения ведущей оси		+	+	FM-A
CPLINTR	K	Базовое соединение: Значение смещения для входного значения ведущей оси		+	+	FM-A
CPLNUM	K	Базовое соединение: Числитель коэффициента связи		+	+	FM-A
CPLOF	K	Базовое соединение: Выключение ведущей оси модуля сопряжения		+	+	FM-A
CPLON	K	Базовое соединение: Включение ведущей оси модуля сопряжения		+	+	FM-A
CPLOUTSC	K	Базовое соединение: Коэффициент масштабирования для выходного значения соединения		+	+	FM-A
CPLOUTTR	K	Базовое соединение: Значение смещения для выходного значения соединения		+	+	FM-A
CPLPOS	K	Базовое соединение: синхронная позиция ведущей оси		+	+	FM-A
CPLSETVAL	K	Базовое соединение: Источник соединения		+	+	FM-A
CPMALARM	K	Базовое соединение: Блокировка вывода спец. относящихся к соединению аварийных сообщений		+	+	FM-A
CPMBRAKE	K	Базовое соединение: Поведение ведомой оси при определенных сигналах и командах останова		+	-	FM-A
CPMPRT	K	Базовое соединение: Поведение соединения при запуске программы обработки детали под поиск кадра через тестирование программы		+	+	FM-A
CPMRESET	K	Базовое соединение: Поведение соединения при RESET		+	+	FM-A
CPMSTART	K	Базовое соединение: Поведение соединения при запуске программы обработки детали		+	+	FM-A
CPMVDI	K	Базовое соединение: Реакция ведомой оси на определенные интерфейсные сигналы ЧПУ/PLC		+	+	FM-A
CPOF	K	Базовое соединение: Выключение модуля сопряжения		+	+	FM-A
CPON	K	Базовое соединение: Включение модуля сопряжения		+	+	FM-A
CPRECOF [6]	G	Выключение программируемой точности контура	м	+		PM-NC
CPRECON	G	Включение программируемой точности контура	м	+		PM-NC
CPRES	K	Базовое соединение: Активирует сконфигурированные данные соединения синхронных шпинделей		+	-	FM-A

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
CPROT	P	Включить/выключить спец. для канала защищенную область		+	-	PM-NC
CPROTDEF	P	Определение специфической для канала защищенной области		+	-	PM-NC
CPSETTYPE	K	Базовое соединение: Тип соединения		+	+	FM-A
CPSYNCOP	K	Базовое соединение: Пороговое значение синхронизации позиций "Грубо"		+	+	FM-A
CPSYNCOP2	K	Базовое соединение: Пороговое значение синхронизации позиций "Грубо" 2		+	+	FM-A
CPSYNCOV	K	Базовое соединение: Пороговое значение синхронизации скорости "Грубо"		+	+	FM-A
CPSYNFIP	K	Базовое соединение: Пороговое значение синхронизации позиций "Точно"		+	+	FM-A
CPSYNFIP2	K	Базовое соединение: Пороговое значение синхронизации позиций "Точно" 2		+	+	FM-A
CPSYNFIV	K	Базовое соединение: Пороговое значение синхронизации скорости "Точно"		+	+	FM-A
CR	A	Радиус окружности	п	+		PM-NC
CROT	F	Вращение актуальной системы координат		+	-	PM-NC
CROTS	F	Программируемые вращения фрейма с пространственными углами (вращение в указанных осях)	п	+	-	PM-NC
CRPL	F	Вращение фрейма в произвольной плоскости		+	-	FM-B
CSCALE	F	Коэффициент масштабирования для нескольких осей		+	-	PM-NC
CSPLINE	F	Кубический сплайн	м	+		PM-NC
CT	G	Окружность с тангенциальным переходом	м	+		PM-NC
CTAB	F	Определить позицию ведомой оси на основе позиции ведущей оси из таблицы кривых		+	+	PM-NC
CTABDEF	P	Включить определение таблицы		+	-	PM-NC
CTABDEL	P	Удалить таблицу кривых		+	-	PM-NC
CTABEND	P	Выключить определение таблицы		+	-	PM-NC
CTABEXISTS	F	Проверяет таблицу кривых с номером n		+	+	PM-NC
CTABFNO	F	Число еще возможных таблиц кривых в памяти		+	+	PM-NC
CTABFPOL	F	Число еще возможных полиномов в памяти		+	+	PM-NC
CTABFSEG	F	Число еще возможных сегментов кривых в памяти		+	+	PM-NC
CTABID	F	Выводит номер n-ной таблицы кривых		+	+	PM-NC
CTABINV	F	Определить позицию ведущей оси на основе позиции ведомой оси из таблицы кривых		+	+	PM-NC
CTABISLOCK	F	Возвращает состояние блокировки таблицы кривых с номером n		+	+	PM-NC
CTABLOCK	P	Удаление и перезапись, блокировать		+	+	PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
CTABMEMTYP	F	Возвращает память, в которой создана таблица кривых с номером n.		+	+	PM-NC
CTABMPOL	F	Число макс. возможных полиномов в памяти		+	+	PM-NC
CTABMSEG	F	Число макс. возможных сегментов кривых в памяти		+	+	PM-NC
CTABNO	F	Число определенных таблиц кривых в SRAM или DRAM		+	+	FM-A
CTABNOMEM	F	Число определенных таблиц кривых в SRAM или DRAM		+	+	PM-NC
CTABPERIOD	F	Возвращает периодичность таблицы кривых с номером n		+	+	PM-NC
CTABPOL	F	Число уже использованных полиномов в памяти		+	+	PM-NC
CTABPOLID	F	Число использованных таблицей кривых с номером n криволинейных полиномов		+	+	PM-NC
CTABSEG	F	Число уже использованных сегментов кривой в памяти		+	+	PM-NC
CTABSEGID	F	Число использованных таблицей кривых с номером n сегментов кривой		+	+	PM-NC
CTABSEV	F	Выводит конечное значение ведомой оси сегмента таблицы кривых		+	+	PM-NC
CTABSSV	F	Выводит стандартное значение ведомой оси сегмента таблицы кривых		+	+	PM-NC
CTABTEP	F	Выводит значение ведущей оси на конце таблицы кривых		+	+	PM-NC
CTABTEV	F	Выводит значение ведомой оси на конце таблицы кривых		+	+	PM-NC
CTABTMAX	F	Выводит макс. значение ведомой оси таблицы кривых		+	+	PM-NC
CTABTMIN	F	Выводит мин. значение ведомой оси таблицы кривых		+	+	PM-NC
CTABTSP	F	Выводит значение ведущей оси на начале таблицы кривых		+	+	PM-NC
CTABTSV	F	Выводит значение ведомой оси на начале таблицы кривых		+	+	PM-NC
CTABUNLOCK	P	Снятие блокировки удаления и перезаписи		+	+	PM-NC
CTOL	A	Допуск для функций компрессора, сглаживания ориентации и типов перешлифовки	м	+	-	PM-NC
CTOLG0	A	Допуск контура при движениях ускоренного хода	м	+	-	PM-NC
CTRANS	F	Смещение нулевой точки для нескольких осей		+	-	PM-NC
CUT2D [6]	G	Коррекция радиуса 2D	м	+		PM-NC
CUT2DD	G	Коррекция радиуса 2½D относительно дифференциального инструмента	м	+		PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
CUT2DF	G	Коррекция радиуса 2D, относительно текущего фрейма (наклонная плоскость)	м	+		PM-NC
CUT2DFD	G	Коррекция радиуса 2½D относительно дифференциального инструмента, относительно текущего фрейма (наклонная плоскость)	м	+		PM-NC
CUT3DC	G	Коррекция радиуса 3D для периферийного фрезерования	м	+		PM-NC
CUT3DCC	G	Коррекция радиуса 3D для периферийного фрезерования с учетом ограничивающей поверхности с коррекцией радиуса 3D: Контур на поверхности обработки	м	+		PM-NC
CUT3DCCD	G	Коррекция радиуса 3D для периферийного фрезерования с учетом ограничивающей поверхности с дифференциальным инструментом на траектории центра инструмента: Подача к ограничивающей поверхности	м	+		PM-NC
CUT3DCD	G	Коррекция радиуса 3D относительно дифференциального инструмента для периферийного фрезерования	м	+		PM-NC
CUT3DF	G	Коррекция радиуса 3D для торцового фрезерования с изменением ориентации	м	+		PM-NC
CUT3DFD	G	Коррекция радиуса 3D относительно дифференциального инструмента для торцового фрезерования с изменением ориентации	м	+		PM-NC
CUT3DFF	G	Коррекция радиуса 3D для торцового фрезерования с постоянной ориентацией. Ориентация инструмента — это определенное через G17 – G19 и в некоторых случаях повернутое через фрейм направление.	м	+		PM-NC
CUT3DFS	G	Коррекция радиуса 3D для торцового фрезерования с постоянной ориентацией. Ориентация инструмента определена через G17 – G19 и не подвергается воздействию фреймов.	м	+		PM-NC
CUTCONOF [6]	G	Выключить постоянную коррекцию радиуса	м	+		PM-NC
CUTCONON	G	Включить постоянную коррекцию радиуса	м	+		PM-NC
CUTMOD	A	Включение изменения данных коррекции для вращающихся инструментов (в сочетании с ориентируемыми инструментальными суппортами)	м	+		PM-NC
CUTMODK	A	Включение изменения данных коррекции для вращающихся инструментов (в сочетании с трансформациями ориентации, определенными посредством кинематических цепочек)	м	+		PM-NC
CYCLE60	C (T)	Цикл гравирования		+		PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
CYCLE61	C (T)	Плоское фрезерование		+		PM-NC
CYCLE62	C (T)	Вызов контура		+		PM-NC
CYCLE63	C (T)	Фрезерование контурного кармана		+		PM-NC
CYCLE64	C (T)	Предварительное сверление контурного кармана		+		PM-NC
CYCLE70	C (T)	Резьбофрезерование		+		PM-NC
CYCLE72	C (T)	Фрезерование траектории		+		PM-NC
CYCLE76	C (T)	Фрезерование прямоугольной цапфы		+		PM-NC
CYCLE77	C (T)	Фрезерование круговой цапфы		+		PM-NC
CYCLE78	C (T)	Сверлильное резьбофрезерование		+		PM-NC
CYCLE79	C (T)	Многогранник		+		PM-NC
CYCLE81	C (T)	Сверление, центрование		+		PM-NC
CYCLE82	C (T)	Сверление, зенкование		+		PM-NC
CYCLE83	C (T)	Глубокое сверление		+		PM-NC
CYCLE84	C (T)	Нарезание внутренней резьбы без компенсирующего патрона		+		PM-NC
CYCLE85	C (T)	Развертывание		+		PM-NC
CYCLE86	C (T)	Растачивание		+		PM-NC
CYCLE92	C (T)	Отрез		+		PM-NC
CYCLE95	C (T)	Съём припуска вдоль контура		+		PM-NC
CYCLE98	C (T)	Цепочка резьб		+		PM-NC
CYCLE99	C (T)	Нарезание резьбы резцом		+		PM-NC
CYCLE116	C (M)	Расчет центра и радиуса окружности		+		PM-MC
CYCLE119	C (M)	Определение положения в пространстве		+		PM-MC
CYCLE150	C (M)	Индикация / регистрация результатов измерения		+		PM-MC
CYCLE435	C (T)	Вычислить позицию инструмента для правки		+		PM-NC
CYCLE495	C (T)	Профилирование		+		PM-NC
CYCLE750	C (A)	Внутренний рабочий цикл для CYCLE751 ... CYCLE759 (содержит команду MMC для прямого вызова функции)		-		FM-A
CYCLE751	C (A)	Открыть / выполнить / закрыть сеанс оптимизации		M		FM-A
CYCLE752	C (A)	Добавить ось к сеансу оптимизации		M		FM-A
CYCLE753	C (A)	Выбрать режим оптимизации		M		FM-A
CYCLE754	C (A)	Добавить / удалить блок данных		M		FM-A
CYCLE755	C (A)	Архивировать / восстановить блок данных		M		FM-A
CYCLE756	C (A)	Активировать результаты оптимизации		M		FM-A
CYCLE757	C (A)	Сохранить данные оптимизации		M		FM-A
CYCLE758	C (A)	Изменить значение параметра		M		FM-A
CYCLE759	C (A)	Считать значение параметра		M		FM-A
CYCLE782	C (T)	Согласовать по загрузке		+		PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
CYCLE800	C (T)	Поворот		+		PM-NC
CYCLE801	C (T)	Решетка или матрица		+		PM-NC
CYCLE802	C (T)	Любые позиции		+		PM-NC
CYCLE806	C (T)	Интерполированное вращение		+		PM-NC
CYCLE830	C (T)	Глубокое сверление 2		+		PM-NC
CYCLE832	C (T)	High Speed Settings		+		PM-NC
CYCLE840	C (T)	Нарезание внутренней резьбы с компенсирующим патроном		+		PM-NC
CYCLE899	C (T)	Фрезерование открытой канавки		+		PM-NC
CYCLE930	C (T)	Выточка		+		PM-NC
CYCLE940	C (T)	Формы канавки		+		PM-NC
CYCLE951	C (T)	Обработка резаньем		+		PM-NC
CYCLE952	C (T)	Прорезание контура		+		PM-NC
CYCLE961	C (M)	Определение положения угла детали (внутри или снаружи) и использование как смещения нулевой точки		+		PM-MC
CYCLE971	C (M)	Калибровка измерительного щупа инструмента, измерение длины инструмента и/или радиуса инструмента (только для технологии фрезерования)		+		PM-MC
CYCLE973	C (M)	Калибровка измерительного щупа детали на поверхности детали или в пазу (только для технологии токарной обработки)		+		PM-MC
CYCLE974	C (M)	Определение нулевой точки детали в выбранной измерительной оси, коррекция инструмента с помощью 1-точечного измерения (только для технологии токарной обработки)		+		PM-MC
CYCLE976	C (M)	Калибровка измерительного щупа детали в калибровочном кольце или на калибровочной сфере полностью в рабочей плоскости или на кромке для определенной оси и направления		+		PM-MC
CYCLE977	C (M)	Определение центра плоскости, а также ширины или диаметра		+		PM-MC
CYCLE978	C (M)	Измерение положения кромки в системе координат детали		+		PM-MC
CYCLE979	C (M)	Измерение центра плоскости, измерение радиуса круговых сегментов		+		PM-MC
CYCLE982	C (M)	Калибровка измерительного щупа инструмента, измерение токарного, сверлильного и фрезеровального инструмента (только для технологии токарной обработки)		+		PM-MC
CYCLE994	C (M)	Определение нулевой точки детали в выбранной измерительной оси с помощью 2-точечного измерения (только для технологии токарной обработки)		+		PM-MC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
CYCLE995	C (M)	Измерение наклона шпинделя на станке		+		PM-MC
CYCLE996	C (M)	Определение данных, релевантных для кинематических трансформаций, с использованием круговых осей		+		PM-MC
CYCLE997	C (M)	Определение центра и диаметра сферы, измерение центров трех распределенных сфер		+		PM-MC
CYCLE998	C (M)	Определение углового положения поверхности (плоскости) относительно рабочей плоскости, определение угла кромок в системе координат детали		+		PM-MC
CYCLE4071	C (T)	Продольное шлифование с подачей в точке возврата		+		PM-NC
CYCLE4072	C (T)	Продольное шлифование с подачей в точке возврата и сигналом отмены		+		PM-NC
CYCLE4073	C (T)	Продольное шлифование с непрерывной подачей		+		PM-NC
CYCLE4074	C (T)	Продольное шлифование с непрерывной подачей и сигналом отмены		+		PM-NC
CYCLE4075	C (T)	Плоское шлифование с подачей в точке возврата		+		PM-NC
CYCLE4077	C (T)	Плоское шлифование с подачей в точке возврата и сигналом отмены		+		PM-NC
CYCLE4078	C (T)	Плоское шлифование с непрерывной подачей		+		PM-NC
CYCLE4079	C (T)	Плоское шлифование с прерывистой подачей		+		PM-NC
CYCLE9960	C (M)	Полное измерение кинематики		+		PM-MC

Операторы D ... F

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
D	A	Номер коррекции инструмента		+		PM-NC
D0	A	При D0 коррекции для инструмента не действуют.		+		PM-NC
DAC	K	Абсолютное покадровое спец. для оси программирование диаметра	п	+		PM-NC
DC	K	Абсолютное указание размеров для круговых осей, прямой подвод к позиции	п	+		PM-NC
DCI	K	Присвоить класс данных I (= Individual) (только SINUMERIK 828D!)		+		PM-NC
DCM	K	Присвоить класс данных M (= Manufacturer) (только SINUMERIK 828D!)		+		PM-NC
DCU	K	Присвоить класс данных U (= User) (только SINUMERIK 828D!)		+		PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1][2][3][4][5] **Объяснения см. Экспликация (Страница 1247).**						
DEF	K	Определение переменных		+		PM-NC
DEFAULT	K	Ветвь в ветвлении CASE		+		PM-NC
DEFINE	K	Кодовое слово для макроопределений		+		PM-NC
DELAYFSTOF	P	Определить конец области задержки останова	м	+	-	PM-NC
DELAYFSTON	P	Определить начало области задержки останова	м	+	-	PM-NC
DELDL	F	Удалить аддитивные коррекции		+	-	PM-NC
DELDTG	P	Стирание остатка пути		-	+	FM-SA
DELETE	P	Удалить указанный файл. Имя файла может быть указано с маршрутом и идентификатором файла.		+	-	PM-NC
DELMLOWNER	F	Удалить место в собственном магазине инструмента		+	-	FM-TM
DELMLRES	F	Удалить резервирование места в магазине		+	-	FM-TM
DELMT	P	Удалить мультиинструмент		+	-	FM-TM
DELOBJ	F	Удаление элементов кинематических цепочек, защищенных областей, элементов защищенных областей, соударяемых пар и данных трансформаций		+		PM-NC
DELT	P	Удалить инструмент		+	-	FM-TM
DELTC	P	Удалить блок данных инструментального суппорта		+	-	FM-TM
DELTOOLENV	F	Удалить блоки данных для описания инструментального окружения		+	-	PM-NC
DIACYCOFA	K	Спец. для оси модальное программирование диаметра: ВЫКЛ в циклах	м	+		FM-A
DIAM90	G	Программирование диаметра для G90, программирование радиуса для G91	м	+		PM-NC
DIAM90A	K	Спец. для оси модальное программирование диаметра для G90 и AC, программирование радиуса для G91 и IC	м	+		PM-NC
DIAMCHAN	K	Передача всех осей из MD функций осей в состояние канала программирования диаметра		+		PM-NC
DIAMCHANA	K	Применение состояния канала программирования диаметра		+		PM-NC
DIAMCYCOF	G	Спец. для канала программирование диаметра: ВЫКЛ в циклах	м	+		FM-A
DIAMOF [6]	G	Программирование диаметра: ВЫКЛ Первичная установка см. Изготовитель станка	м	+		PM-NC
DIAMOFA	K	Спец. для оси модальное программирование диаметра: ОТКЛ Первичная установка см. Изготовителя станка	м	+		PM-NC
DIAMON	G	Программирование диаметра: ВКЛ	м	+		PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1][2][3][4][5] **Объяснения см. Экспликация (Страница 1247).**						
DIAMONA	K	Спец. для оси модальное программирование диаметра: ВКЛ Активация см. Изготовитель станка	м	+		PM-NC
DIC	K	Относительное покадровое спец. для оси программирование диаметра	п	+		PM-NC
DILF	A	Путь обратного хода (длина)	м	+		PM-NC
Disable	P	Прерывание ВЫКЛ.		+	-	PM-NC
DISC	A	Выступ переходного круга коррекции радиуса инструмента	м	+		PM-NC
DISCL	A	Расстояние от конечной точки быстрого движения подачи до плоскости обработки		+		PM-NC
DISPLOF	PA	Подавление актуальной индикации кадра		+		PM-NC
DISPLON	PA	Отменить подавление актуальной индикации кадра		+		PM-NC
DISPR	A	Рассогласование траектории Repos	п	+		PM-NC
DISR	A	Интервал Repos	п	+		PM-NC
DISRP	A	Расстояние от плоскости отвода до плоскости обработки при мягком подводе и отводе		+		PM-NC
DITE	A	Выход резьбы	м	+		PM-NC
DITS	A	Вход резьбы	м	+		PM-NC
DIV	K	Целочисленное деление		+		PM-NC
DL	A	Выбрать зависящую от места аддитивную коррекцию инструмента (DL, суммарная отладочная коррекция)	м	+		PM-NC
DO	K	Синхронное действие: запуск действий при выполненном условии		-	+	FM-SA
DRFOF	P	Выключение смещений маховичком (DRF)	м	+	-	PM-NC
DRIVE	G	Ускорение по траектории в зависимости от скорости	м	+		PM-NC
DRIVEA	P	Включение ломаной характеристики ускорения для запрограммированных осей		+	-	PM-NC
DRVPRD	P	Чтение параметров привода		+	-	PM-NC
DRVPWR	P	Запись параметров привода		+	-	PM-NC
DYNFINISH	G	Динамика для чистовой обработки	м	+		PM-NC
DYNNORM [6]	G	Обычная динамика	м	+		PM-NC
DYNPOS	G	Динамика для режима позиционирования, нарезание внутренней резьбы	м	+		PM-NC
DYNPREC	G	Динамика для чистовой обработки точной	м	+		PM-NC
DYNROUGH	G	Динамика для черновой обработки	м	+		PM-NC
DYNSEMIFIN	G	Динамика для предварительной чистовой обработки	м	+		PM-NC
DZERO	P	Обозначает все номера D блока TO как недействительные		+	-	PM-NC
EAUTO	G	Определение последнего сегмента сплайна через последние 3 точки	м	+		PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1][2][3][4][5] **Объяснения см. Экспликация (Страница 1247).**						
EGDEF	P	Определение электронного редуктора		+	-	PM-NC
EGDEL	P	Удалить определение соединения для ведомой оси		+	-	PM-NC
EGOFC	P	Непрерывное выключение электронного редуктора		+	-	PM-NC
EGOFS	P	Селективное выключение электронного редуктора		+	-	PM-NC
EGON	P	Включение электронного редуктора		+	-	PM-NC
EGONSYN	P	Включение электронного редуктора		+	-	PM-NC
EGONSYNE	P	Включение электронного редуктора, с задаче режима пуска		+	-	PM-NC
ELSE	K	Программа ЧПУ: ветвление программы при невыполненном условии IF		+	-	PM-NC
ELSE	K	Синхронное действие: запуск действий при невыполненном условии		-	+	FM-SA
Enable	P	Прерывание ВКЛ.		+	-	PM-NC
ENAT [6]	G	Натуральный криволинейный переход к следующему кадру перемещения	м	+		PM-NC
ENDFOR	K	Конечная строка цикла вычисления FOR		+		PM-NC
ENDIF	K	Конечная строка ветвления IF		+		PM-NC
ENDLABEL	K	Метка конца для повторений программы обработки детали через REPEAT		+		PM-NC, FM-B
ENDLOOP	K	Конечная строка бесконечного программного цикла LOOP		+		PM-NC
ENDPROC	K	Конечная строка программы с начальной строкой PROC		+		
ENDWHILE	K	Конечная строка цикла WHILE		+		PM-NC
ESRR	P	Спараметрировать автономный отвода привода ESR в приводе		+		PM-NC
ESRS	P	Спараметрировать автономный останов привода ESR в приводе		+		PM-NC
ETAN	G	Тангенциальный криволинейный переход к следующему кадру перемещения на начале сплайна	м	+		PM-NC
EVERY	K	Выполнить синхронное действие при переходе условия с FALSE на TRUE		-	+	FM-SA
EX	K	Кодовое значение для присвоения значения при экспоненциальной форме записи		+		PM-NC
EXECSTRING	P	Передача переменной String с исполняемой строкой программы обработки детали		+	-	PM-NC
EXECTAB	P	Обработать элемент из таблицы движений		+	-	PM-NC
EXECUTE	P	Выполнение программы ВКЛ		+	-	PM-NC
EXP	F	Показательная функция ex		+	+	PM-NC
EXTCALL	A	Выполнить внешнюю подпрограмму		+	+	PM-NC
EXTCLOSE	P	Для записи закрыть открытое внешнее устройство/файл		+	-	PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1][2][3][4][5] **Объяснения см. Экспликация (Страница 1247).**						
EXTERN	K	Публикация подпрограммы с передачей параметров		+		PM-NC
EXTOPEN	P	Открыть внешнее устройство/файл для канала для записи		+	-	PM-NC
F	A	Значение подачи (вместе с G4 с F программируется и время ожидания)		+	+	PM-NC
FA	K	Осевая подача	м	+	+	PM-NC
FAD	A	Подача на ход для мягкого подвода и отвода		+		PM-NC
FALSE	K	Логическая постоянная: неверно		+	+	PM-NC
FB	A	Покадровая подача		+		PM-NC
FCTDEF	P	Определение функции полинома		+	-	PM-NC
FCUB	G	подача изменяется по кубическому сплайну	м	+		PM-NC
FD	A	Подача по траектории для наложения маховичка	п	+		PM-NC
FDA	K	Осевая подача для наложения маховичка	п	+		PM-NC
FENDNORM [6]	G	Замедление на углах ВЫКЛ	м	+		PM-NC
FFWOF [6]	G	Предуправление ВЫКЛ	м	+		PM-NC
FFWON	G	Предуправление Вкл	м	+		PM-NC
FGREF	K	Исходный радиус для круговых осей или коэффициенты соотношения траекторий для осей ориентации (векторная интерполяция)	м	+		PM-NC
FGROUP	P	Определение оси(ей) с подачей по траектории		+	-	PM-NC
FI	K	Параметры для доступа к данным фрейма: Точное смещение		+		PM-NC
FIFOCTRL	G	Управление буфером предварительной обработки	м	+		PM-NC
FILEDATE	P	Выводит дату последнего обращения по записи к файлу		+	-	PM-NC
FILEINFO	P	Выводит сумму FILEDATE, FILESIZE, FILESTAT и FILETIME		+	-	PM-NC
FILESIZE	P	Выводит актуальный размер файла		+	-	PM-NC
FILESTAT	P	Выводит состояние файла прав чтения, записи, исполнения, индикации, удаления (rwxsd)		+	-	PM-NC
FILETIME	P	Выводит время последнего обращения по записи к файлу		+	-	PM-NC
FINEA	K	Конец движения при достижении "Точного останова точного"	м	+		PM-NC
FL	K	Предельная скорость для синхронных осей	м	+		PM-NC
FLIM	A	Коррекция максимальной скорости движения по траектории	м	+	-	PM-NC

Оператор	Тип [1]	Значение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1][2][3][4][5] **Объяснения см. Экспликация (Страница 1247).**						
FLIN	G	подача изменяется линейно	м	+		PM-NC
FMA	K	Несколько осевых подач	м	+		PM-NC
FNORM [6]	G	обычная подача по DIN66025	м	+		PM-NC
FOC	K	Действующее покадрово ограничение момента/силы	п	-	+	FM-SA
FOCOF	K	Выключение модального ограничения момента/силы	м	-	+	FM-SA
FOCON	K	Включение модального ограничения момента/силы	м	-	+	FM-SA
FOR	K	Цикл вычисления с фиксированным числом проходов		+		PM-NC
FP	A	Фиксированная точка: номер фикс. точки, к которой осуществляется подвод	п	+		PM-NC
FPO	K	Запрограммированная через полином характеристика подачи		+		PM-NC
FPR	P	Обозначение круговой оси		+	-	PM-NC
FPRAOF	P	Выключение окружной подачи		+	-	PM-NC
FPRAON	P	Включение окружной подачи		+	-	PM-NC
FRAME	K	Тип данных для определения систем координат		+		PM-NC
FRC	A	Подача для радиуса и фаски	п	+		PM-NC
FRCM	A	Модальная подача для радиуса и фаски	м	+		PM-NC
FROM	K	Операция выполняется, если условие выполнено один раз и пока активно синхронное действие		-	+	FM-SA
FTOC	P	Изменить точную коррекцию инструмента		-	+	FM-SA
FTOCOF [6]	G	действующая online точная коррекция инструмента ВЫКЛ	м	+		PM-NC
FTOCON	G	действующая online точная коррекция инструмента ВКЛ	м	+		PM-NC
FXS	K	Наезд на жесткий упор вкл	м	+	+	PM-NC
FXST	K	Граница момента для наезда на жесткий упор	м	+	+	PM-NC
FXSW	K	Окно контроля для наезда на жесткий упор		+	+	PM-NC
FZ	K	Подача на зуб	м	+		PM-NC

Операторы G ... L

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
G0	G	Линейная интерполяция с ускоренным ходом (движение ускоренного хода)	м	+		PM-NC
G1 [6]	G	Линейная интерполяция с подачей	м	+		PM-NC
G2	G	Круговая интерполяция по часовой стрелке	м	+		PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
G3	G	Круговая интерполяция против часовой стрелки	м	+		PM-NC
G4	G	Время ожидания, заранее определено по времени	п	+		PM-NC
G5	G	Шлифование с врезанием с угловой подачей	п	+		PM-NC
G7	G	Компенсационное движение при шлифовании с врезанием с угловой подачей	п	+		PM-NC
G9	G	Точный останов - снижение скорости	п	+		PM-NC
G17 [6]	G	Выбор рабочей плоскости X/Y	м	+		PM-NC
G18	G	Выбор рабочей плоскости Z/X	м	+		PM-NC
G19	G	Выбор рабочей плоскости Y/Z	м	+		PM-NC
G25	G	Нижнее ограничение рабочего поля	п	+		PM-NC
G26	G	Верхнее ограничение рабочего поля	п	+		PM-NC
G33	G	Резьбонарезание с постоянным шагом	м	+		PM-NC
G34	G	Резьбонарезание с линейной увеличивающимся шагом	м	+		PM-NC
G35	G	Резьбонарезание с линейной уменьшающимся шагом	м	+		PM-NC
G40 [6]	G	Коррекция радиуса инструмента ВЫКЛ.	м	+		PM-NC
G41	G	Коррекция радиуса инструмента слева от контура	м	+		PM-NC
G42	G	Коррекция радиуса инструмента справа от контура	м	+		PM-NC
G53	G	Подавление актуального смещения нулевой точки (покадрово)	п	+		PM-NC
G54	G	1-е настраиваемое смещение нулевой точки	м	+		PM-NC
G55	G	2-е настраиваемое смещение нулевой точки	м	+		PM-NC
G56	G	3-е настраиваемое смещение нулевой точки	м	+		PM-NC
G57	G	4-е настраиваемое смещение нулевой точки	м	+		PM-NC
G58 (840D sl)	G	Абсолютное программируемое смещение нулевой точки (грубое смещение)	п	+		PM-NC
G58 (828D)	G	5-е настраиваемое смещение нулевой точки	м	+		PM-NC
G59 (840D sl)	G	Аддитивное программируемое смещение нулевой точки (точное смещение)	п	+		PM-NC
G59 (828D)	G	6-е настраиваемое смещение нулевой точки	м	+		PM-NC
G60 [6]	G	Точный останов - снижение скорости	м	+		PM-NC
G62	G	Задержка на внутренних углах при активной коррекции радиуса инструмента (G41, G42)	м	+		PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
G63	G	Нарезание внутренней резьбы с компенсирующим патроном	п	+		PM-NC
G64	G	Режим управления траекторией	м	+		PM-NC
G70	G	Дюймовое указание размеров для геометрических данных (длины)	м	+	+	PM-NC
G71 [6]	G	Метрическое указание размеров для геометрических данных (длины)	м	+	+	PM-NC
G74	G	Движение к точке реферирования	п	+		PM-NC
G75	G	Подвод к базисной точке	п	+		PM-NC
G90 [6]	G	Абсолютное указание размера	м/п	+		PM-NC
G91	G	Указание составного размера	м/п	+		PM-NC
G93	G	Обратная по времени подача 1/мин	м	+		PM-NC
G94 [6]	G	Линейная подача F в мм/мин или дюймов/мин и градусах/мин	м	+		PM-NC
G95	G	Окружная подача F в мм/об или дюймов/оборот	м	+		PM-NC
G96	G	Окружная подача (как у G95) и постоянная скорость резания	м	+		PM-NC
G97	G	Окружная подача и постоянная частота вращения шпинделя (постоянная скорость резания ВЫКЛ)	м	+		PM-NC
G110	G	Программирование полюса относительно последней запрограммированной заданной позиции	п	+		PM-NC
G111	G	Программирование полюса относительно нулевой точки актуальной системы координат детали	п	+		PM-NC
G112	G	Программирование полюса относительно последнего действующего полюса	п	+		PM-NC
G140 [6]	G	Направление подвода SAR установлено через G41/G42	м	+		PM-NC
G141	G	Направление подвода SAR слева от контура	м	+		PM-NC
G142	G	Направление подвода SAR справа от контура	м	+		PM-NC
G143	G	Направление подвода SAR зависимое от касательных	м	+		PM-NC
G147	G	Мягкий подвод по прямой	п	+		PM-NC
G148	G	Мягкий отвод по прямой	п	+		PM-NC
G153	G	Подавление актуальных фреймов включая базовый фрейм	п	+		PM-NC
G247	G	Мягкий подвод по четверти круга	п	+		PM-NC
G248	G	Мягкий отвод по четверти круга	п	+		PM-NC
G290 [6]	G	Переключение на режим SINUMERIK ВКЛ.	м	+		FM-TM
G291	G	Переключение на режим ISO2/3 ВКЛ.	м	+		FM-TM

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
G331	G	Нарезание внутренней резьбы без компенсирующего патрона, положительный шаг, правый ход	м	+		PM-NC
G332	G	Нарезание внутренней резьбы без компенсирующего патрона, отрицательный шаг, левый ход	м	+		PM-NC
G335	G	Нарезание резьбы с выпуклым профилем по часовой стрелке	м	+		PM-NC
G336	G	Нарезание резьбы с выпуклым профилем против часовой стрелки	м	+		PM-NC
G340 [6]	G	Кадр подвода пространственный (по глубине и в плоскости одновременно (спираль))	м	+		PM-NC
G341	G	Подача сначала в вертикальной оси (z), потом подвод в плоскости	м	+		PM-NC
G347	G	Мягкий подвод по полукругу	п	+		PM-NC
G348	G	Мягкий отвод по полукругу	п	+		PM-NC
G450 [6]	G	Переходная окружность	м	+		PM-NC
G451	G	Точка пересечения эквидистант	м	+		PM-NC
G460 [6]	G	Включение контроля столкновения для кадра подвода и отвода	м	+		PM-NC
G461	G	Вставка окружности в кадре КРИ	м	+		PM-NC
G462	G	Вставка прямой в кадре КРИ	м	+		PM-NC
G500 [6]	G	Отключение всех устанавливаемых фреймов, базовые фреймы активны	м	+		PM-NC
G505 ... G599	G	5. ... 99. Настраиваемое смещение нулевой точки	м	+		PM-NC
G601 [6]	G	Смена кадра при точном останове точном	м	+		PM-NC
G602	G	Смена кадра при точном останове грубом	м	+		PM-NC
G603	G	Смена кадра при конце кадра IPO	м	+		PM-NC
G621	G	Замедление на всех углах	м	+		PM-NC
G641	G	Режим управления траекторией с перешлифовкой по критерию пути (= программируемый интервал перешлифовки)	м	+		PM-NC
G642	G	Режим управления траекторией с перешлифовкой с соблюдением определенных допусков	м	+		PM-NC
G643	G	Режим управления траекторией с перешлифовкой с соблюдением определенных допусков (внутри кадра)	м	+		PM-NC
G644	G	Режим управления траекторией с перешлифовкой с макс. возможной динамикой	м	+		PM-NC
G645	G	Режим управления траекторией с перешлифовкой углов с тангенциальными переходами кадров с соблюдением определенных допусков	м	+		PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
G700	G	Дюймовое указание размеров для геометрических и технологических данных (длины, подача)	м	+	+	PM-NC
G710 [6]	G	Метрическое указание размеров для геометрических и технологических данных (длины, подача)	м	+	+	PM-NC
G810 [6], ..., G819	G	Зарезервированная для пользователя OEM G-группа		+		PM-NC
G820 [6], ..., G829	G	Зарезервированная для пользователя OEM G-группа		+		PM-NC
G931	G	Задача подачи через время перемещения, отключить постоянную скорость движения по траектории	м	+		
G942	G	Линейная подача и постоянная скорость резания или "замораживание" частоты вращения шпинделя	м	+		
G952	G	Окружная подача и постоянная скорость резания или "замораживание" частоты вращения шпинделя	м	+		
G961	G	Линейная подача (как у G94) и постоянная скорость резания	м	+		PM-NC
G962	G	Линейная подача или окружная подача и постоянная скорость резания	м	+		PM-NC
G971	G	Линейная подача и постоянная частота вращения шпинделя (постоянная скорость резания ВЫКЛ)	м	+		PM-NC
G972	G	Линейная подача или окружная подача и постоянная частота вращения шпинделя (постоянная скорость резания ВЫКЛ)	м	+		PM-NC
G973	G	Окружная подача без ограничения частоты вращения шпинделя и постоянная частота вращения шпинделя (G97 без LIMS для режима ISO)	м	+		PM-NC
GEOAX	P	Согласовать с гео-осями 1 - 3 новые оси канала		+	-	PM-NC
GET	P	Переход разрешенной оси между каналами		+	+	PM-NC
GETACTT	F	Определяет активный инструмент из группы инструментов с одинаковым именем		+	-	FM-TM
GETACTTD	F	Определяет для абсолютного номера D соответствующий номер T		+	-	PM-NC
GETD	P	Прямой переход оси между каналами		+	-	PM-NC
GETDNO	F	Выводит номер D резца (CE) инструмента (T)		+	-	PM-NC
GETEXET	P	Чтение установленного номера T		+	-	FM-TM
GETFREELOC	P	Поиск свободного места в магазинах для данного инструмента		+	-	FM-TM
GETSELT	P	Вывести предварительно выбранный номер T		+	-	FM-TM

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
GETT	F	Определить номер T для имени инструмента		+	-	FM-TM
GETTCOR	F	Считать длину или компоненты длины инструмента		+	-	PM-NC
GETTENV	F	Чтение номеров T, D и DL		+	-	PM-NC
GETVARAP	F	Чтение права доступа к системной переменной/переменной пользователя		+	-	PM-NC
GETVARDFT	F	Чтение стандартного значения системной переменной/переменной пользователя		+	-	PM-NC
GETVARLIM	F	Чтение предельных значений системной переменной/переменной пользователя		+	-	PM-NC
GETVARPHU	F	Чтение физической единицы системной переменной/переменной пользователя		+	-	PM-NC
GETVARTYP	F	Чтение типа данных системной переменной/переменной пользователя		+	-	PM-NC
GFRAME0 ... GFRAME100	G	Активация фрейма шлифования <n> для сохранения данных в канале	м	+		PM-NC
GOTO	K	Оператор перехода сначала вперед, потом назад (сначала в направлении конца программы, а после к началу программы)		+		PM-NC
GOTOB	K	Оператор перехода назад (в направлении начала программы)		+		PM-NC
GOTOC	K	Как GOTO, но подавление ошибки 14080 "Цель перехода не найдена"		+		PM-NC
GOTOF	K	Оператор перехода вперед (в направлении конца программы)		+		PM-NC
GOTOS	K	Возврат в начало программы		+		PM-NC
GP	K	Кодовое слово для косвенного программирования атрибутов позиции		+		PM-NC
GROUP_ ADDEND	C (T)	Конец – дополнение подведения		+		PM-NC
GROUP_BEGIN	C (T)	Начало – блок программы		+		PM-NC
GROUP_END	C (T)	Конец – блок программы		+		PM-NC
GWPSOF	P	Отмена постоянной окружной скорости круга (SUG)	п	+	-	PM-NC
GWPSON	P	Выбор постоянной окружной скорости круга (SUG)	п	+	-	PM-NC
H...	A	Вывод вспомогательных функций на PLC		+	+	PM-NC, FM-B
HOLES1	C (T)	Ряд отверстий		+		PM-NC
HOLES2	C (T)	Окружность отверстий		+		PM-NC
I	A	Параметры интерполяции	п	+		PM-NC
I1	A	Координата промежуточной точки	п	+		PM-NC
IC	K	Ввод составного размера	п	+		PM-NC
ICYCOF	P	Выполнить все кадры технологического цикла после ICYCOF в одном такте IPO		+	+	FM-SA

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
ICYCON	P	Выполнить каждый кадр технологического цикла после ICYCON в отдельном такте IPO		+	+	FM-SA
ID	K	Обозначение для модальных синхронных действий	м	-	+	FM-SA
IDS	K	Обозначение для модальных статических синхронных действий		-	+	FM-SA
IF	K	Ввод условного перехода в программе обработки детали / технологическом цикле		+	+	PM-NC
INDEX	F	Определить индекс символа во входной строке		+	-	PM-NC
INICF	K	Инициализация переменных при NEWCONF		+		PM-NC
INIPO	K	Инициализация переменных при PowerOn		+		PM-NC
INIRE	K	Инициализация переменных при Reset		+		PM-NC
INIT	P	Выбор определенной программы ЧПУ для выполнения в определенном канале		+	-	PM-NC
INITIAL		Создание файла INI по всем областям		+		PM-NC
INT	K	Тип данных: Целочисленное значение со знаком		+		PM-NC
INTERSEC	F	Вычислить точку пересечения между двумя элементами контура		+	-	PM-NC
INVCCW	G	Движение по эвольвенте, против часовой стрелки	м	+		PM-NC
INVCW	G	Движение по эвольвенте, по часовой стрелке	м	+		PM-NC
INVFRAME	F	Вычислить из фрейма инверсный фрейм		+	-	FM-B
IP	K	Переменный параметр интерполяции		+		PM-NC
IPOBRKA	P	Критерий движения от точки применения рампы торможения	м	+	+	
IPOENDA	K	Конец движения при достижении "IPO-Stop"	м	+		PM-NC
IPTRLOCK	P	"Заморозить" начало непригодного для поиска сегмента программы на следующем кадре функций станка.	м	+	-	PM-NC
IPTRUNLOCK	P	Установить конец непригодного для поиска сегмента программы на актуальный на момент прерывания кадр.	м	+	-	PM-NC
IR	A	Координата центра окружности (направление X) при нарезании резьбы с выпуклым профилем		+		PM-NC
ISAXIS	F	Проверить, указанная как параметр гео-ось это 1		+	-	PM-NC
ISD	A	Глубина врезания	м	+		PM-NC
ISFILE	F	Проверить, есть ли файл в прикладной памяти ЧПУ		+	-	PM-NC
ISNUMBER	F	Проверить, может ли входная строка быть преобразована в число		+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
ISOCALL	K	Косвенный вызов запрограммированной на языке ISO программы		+		PM-NC
ISVAR	F	Проверить, содержит ли передаваемый параметр известную в ЧПУ переменную		+	-	PM-NC
J	A	Параметры интерполяции	п	+		PM-NC
J1	A	Координата промежуточной точки	п	+		PM-NC
JERKA	P	Активировать установленные через MD параметры ускорения для запрограммированных осей		+	-	
JERKLIM	K	Коррекция максимального рывка оси	м	+		PM-NC
JERKLIMA	K	Уменьшение или увеличение макс. осевого рывка ведомой оси	м	+	+	PM-NC
JR	A	Координата центра окружности (направление Y) при нарезании резьбы с выпуклым профилем		+		PM-NC
K	A	Параметры интерполяции	п	+		PM-NC
K1	A	Координата промежуточной точки	п	+		PM-NC
KONT	G	Обход контура при коррекции инструмента	м	+		PM-NC
KONTC	G	Подвод/отвод с полиномом с постоянным изгибом	м	+		PM-NC
KONTT	G	Подвод/отвод с полиномом с постоянным касанием	м	+		PM-NC
KR	A	Координата центра окружности (направление Z) при нарезании резьбы с выпуклым профилем		+		PM-NC
I	A	Номер подпрограммы	п	+	+	PM-NC
LEAD	A	Угол предварения 1. Ориентация инструмента 2. Полином ориентации	м	+		PM-NC
LEADOF	P	Осевое соединение по главному значению ВЫКЛ		+	+	PM-NC
LEADON	P	Осевое соединение по главному значению ВКЛ		+	+	PM-NC
LENTOAX	F	Выводит информацию по согласованию длин инструмента L1, L2 и L3 активного инструмента с абсциссой, ординатой и аппликатой		+	-	PM-NC
LFOF [6]	G	Быстрый обратный ход для резьбонарезания ВЫКЛ	м	+		PM-NC
LFON	G	Быстрый обратный ход для резьбонарезания ВКЛ	м	+		PM-NC
LFPOS	G	Обратный ход обозначенной с POLFMASK или POLFMLIN оси на запрограммированную с POLF абсолютную позицию оси	м	+		PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
LFTXT [6]	G	Плоскость движения обратного хода при быстром отводе определяется из касательной к траектории и актуального направления инструмента	м	+		PM-NC
LFWP	G	Плоскость движения обратного хода при быстром отводе определяется через актуальную рабочую плоскость (G17/G18/G19)	м	+		PM-NC
LIFTFAST	K	Быстрый отвод		+		PM-NC
LIMS	K	Ограничение частоты вращения при G96/G961 и G97	м	+		PM-NC
LLI	K	Нижнее предельное значение переменной		+		PM-NC
LN	F	Натуральный логарифм		+	+	PM-NC
LOCK	P	Блокировать синхронное действие с ID (остановить технологический цикл)		-	+	FM-SA
LONGHOLE	C (T)	Продольный паз		+		PM-NC
LOOP	K	Начало бесконечного цикла		+		PM-NC

Операторы M ... R

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
M0		Запрограммированная остановка		+	+	PM-NC
M1		Остановка по выбору		+	+	PM-NC
M2		Конец главной программы (как M30)		+	+	PM-NC
M3		Правое направление вращения шпинделя		+	+	PM-NC
M4		Левое направление вращения шпинделя		+	+	PM-NC
M5		Остановка шпинделя		+	+	PM-NC
M6		Смена инструмента		+	+	PM-NC
M17		Конец подпрограммы		+	+	PM-NC
M19		Позиционирование шпинделя на введенную в SD43240 позицию		+	+	PM-NC
M30		Конец главной программы (как M2)		+	+	PM-NC
M40		Автоматическое переключение редуктора		+	+	PM-NC
M41 ... M45		Ступень редуктора 1 ... 5		+	+	PM-NC
M70		Переход в осевой режим		+	+	PM-NC
MASLDEF	P	Определить структуру осей Master/Slave		+	+	PM-NC
MASLDEL	P	Разъединить структуру осей Master/Slave и удалить определение структуры		+	+	PM-NC
MASLOF	P	Отключение временного соединения		+	+	PM-NC
MASLOFS	P	Отключение временного соединения с автоматическим остановом оси Slave		+	+	PM-NC
MASLON	P	Включение временного соединения		+	+	PM-NC
MATCH	F	Поиск строки в строке		+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
MAXVAL	F	Большее значение двух переменных (арифм. функция)		+	+	PM-NC
MCALL	K	Модальный вызов подпрограммы		+		PM-NC
MEAC	K	Осевое непрерывное измерение без стирания остатка пути	п	+	+	PM-NC
MEAFRAME	F	Вычисление фрейма из точек измерения		+	-	PM-NC
MEAS	A	Измерение со стиранием остатка пути	п	+		PM-NC
MEASA	K	Осевое измерение со стиранием остатка пути	п	+	+	PM-NC
MEASURE	F	Метод вычислений для измерения детали и инструмента		+	-	FM-TE
MEAW	A	Измерение без стирания остатка пути	п	+		PM-NC
MEAWA	K	Осевое измерение без стирания остатка пути	п	+	+	PM-NC
MI	K	Доступ к данным фрейма: Отражение		+		PM-NC
MINDEX	F	Определить индекс символа во входной строке		+	-	PM-NC
MINVAL	F	Меньшее значение двух переменных (арифм. функция)		+	+	PM-NC
MIRROR	G	Программируемое отражение	п	+		PM-NC
MMC	P	Интерактивный вызов диалоговых окон на HMI из программы обработки детали		+	-	PM-NC
MOD	K	Выделение дробной части		+		PM-NC
MODAXVAL	F	Определить позицию модуло круговой оси модуло		+	-	PM-NC
MOV	K	Запустить позиционирующую ось		-	+	FM-SA
MOVT	A	Указать конечную точку движения перемещения в направлении инструмента				FM-B
MSG	P	Программируемые сообщения	м	+	-	PM-NC
MVTOOL	P	Языковая команда для движения инструмента		+	-	FM-TM
n	A	Номер вспомогательного кадра ЧПУ		+		PM-NC
NAMETOINT	F	Определение индекса системной переменной		+		PM-NC
ЧПУ	K	Спецификация сферы действия данных		+		PM-NC
NEWCONF	P	Применить измененные машинные данные (соответствует "Активировать машинные данные")		+	-	PM-NC
NEWMT	F	Создать новый мультиинструмент		+	-	FM-TM
NEWT	F	Создать новый инструмент		+	-	FM-TM
NORM [6]	G	Обычная установка в начальной, конечной точке при коррекции инструмента	м	+		PM-NC
NOT	K	Логическая НЕ (отрицание)		+		PM-NC
NPROT	P	Специфическая для станка защищенная область ВКЛ/ВЫКЛ		+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
NPROTDEF	P	Определение специфической для станка защищенной области		+	-	PM-NC
NUMBER	F	Преобразовать входную строку в число		+	-	PM-NC
OEMIPO1	G	Интерполяция OEM 1	м	+		PM-NC
OEMIPO2	G	Интерполяция OEM 2	м	+		PM-NC
OF	K	Кодовое слово в ветвлении CASE		+		PM-NC
OFFN	A	Припуск к запрограммированному контуру	м	+		PM-NC
OMA1	A	Адрес OEM 1	м	+		PM-NC
OMA2	A	Адрес OEM 2	м	+		PM-NC
OMA3	A	Адрес OEM 3	м	+		PM-NC
OMA4	A	Адрес OEM 4	м	+		PM-NC
OMA5	A	Адрес OEM 5	м	+		PM-NC
OR	K	Логический оператор, операция ИЛИ		+		PM-NC
ORIAXES	G	Линейная интерполяция осей станка или осей ориентации	м	+		PM-NC
ORIAXPOS	G	Угол ориентации через виртуальные оси ориентации с позициями круговых осей	м	+		PM-NC
ORIC [6]	G	Изменения ориентации на наружных углах накладываются на вставляемый круговой кадр	м	+		PM-NC
ORICONCCW	G	Интерполяция на боковой поверхности окружности против часовой стрелки	м	+		PM-NC, FM-TR
ORICONCW	G	Интерполяция на боковой поверхности окружности по часовой стрелке	м	+		PM-NC, FM-TR
ORICONIO	G	Интерполяция на боковой поверхности окружности с указанием промежуточной ориентации	м	+		PM-NC, FM-TR
ORICONTO	G	Интерполяция на боковой поверхности окружности в тангенциальном переходе (указание конечной ориентации)	м	+		PM-NC, FM-TR
ORICURVE	G	Интерполяция ориентации с задачей движения двух контактных точек инструмента	м	+		PM-NC, FM-TR
ORID	G	Изменения ориентации выполняются перед круговым кадром	м	+		PM-NC
ORIEULER [6]	G	Угол ориентации через эйлеров угол	м	+		PM-NC
ORIMKS	G	Ориентация инструмента в системе координат станка	м	+		PM-NC
ORIPATH	G	Ориентация инструмента относительно траектории	м	+		PM-NC
ORIPATHS	G	Ориентация инструмента относительно траектории, изгиб в характеристике ориентации сглаживается	м	+		PM-NC
ORIPLANE	G	Интерполяция в плоскости (соответствует ORIVECT) Большая круговая интерполяция	м	+		PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
ORIRESET	P	Первичная установка ориентации инструмента с макс. 3 осями ориентации		+	-	PM-NC
ORIROTA [6]	G	Угол поворота к заданному абсолютно направлению вращения	м	+		PM-NC
ORIROTC	G	Тангенциальный вектор вращения к касательной к траектории	м	+		PM-NC
ORIROTR	G	Угол поворота относительно плоскости между стартовой и конечной ориентацией	м	+		PM-NC
ORIROTT	G	Угол поворота относительно изменения вектора ориентации	м	+		PM-NC
ORIRPY	G	Угол ориентации через угол RPY (XYZ)	м	+		PM-NC
ORIRPY2	G	Угол ориентации через угол RPY (ZYX)	м	+		PM-NC
ORIS	A	Изменение ориентации	м	+		PM-NC
ORISOF [6]	G	Сглаживание характеристики ориентации ВЫКЛ	м	+		PM-NC
ORISOLH	F	Вычисление ориентаций		+		PM-NC
ORISON	G	Сглаживание характеристики ориентации ВКЛ	м	+		PM-NC
ORIVECT [6]	G	Большая круговая интерполяция (идентично ORIPLANE)	м	+		PM-NC
ORIVIRT1	G	Угол ориентации через виртуальные оси ориентации (определение 1)	м	+		PM-NC
ORIVIRT2	G	Угол ориентации через виртуальные оси ориентации (определение 1)	м	+		PM-NC
ORIWKS [6]	G	Ориентация инструмента в системе координат детали	м	+		PM-NC
OS	K	Вкл/выкл качания		+		PM-NC
OSB	K	Качание: Начальная точка	м	+		PM-NC
OSC	G	Постоянное сглаживание ориентации инструмента	м	+		PM-NC
OSCILL	K	AXIS: 1 - 3 оси подачи	м	+		PM-NC
OSCTRL	K	Опции качания	м	+		PM-NC
OSD	G	Перешлифовка ориентации инструмента через задачу длины перешлифовки с SD	м	+		PM-NC
OSE	K	Конечная точка качания	м	+		PM-NC
OSNSC	K	Качание: Количество выхаживаний	м	+		PM-NC
OSOF [6]	G	Сглаживание ориентации инструмента ВЫКЛ	м	+		PM-NC
OSP1	K	Качание: левая точка возврата	м	+		PM-NC
OSP2	K	Правая точка возврата качания	м	+		PM-NC
OSS	G	Сглаживание ориентации инструмента на конце кадра	м	+		PM-NC
OSSE	G	Сглаживание ориентации инструмента на начале кадра и на конце кадра	м	+		PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
OST	G	Перешлифовка ориентации инструмента через задачу углового допуска в градусах с SD (макс. отклонение от запрограмм. характеристики ориентации)	м	+		PM-NC
OST1	K	Качание: точка останова в левой точке возврата	м	+		PM-NC
OST2	K	Качание: точка останова в правой точке возврата	м	+		PM-NC
OTOL	A	Допуск ориентации для функций компрессора, сглаживания ориентации и типов перешлифовки	м	+	-	PM-NC
OTOLG0	A	Допуск ориентации при движениях ускоренного хода	м	+	-	PM-NC
OVR	K	Коррекция частоты вращения	м	+		PM-NC
OVRA	K	Осевая коррекция частоты вращения	м	+	+	PM-NC
OVRRAP	K	Коррекция ускоренного хода	м	+		PM-NC
P	A	Число прогонов подпрограммы		+		PM-NC
PACCLIM	A	Коррекция максимального ускорения по траектории	м	+	-	PM-NC
PAROT	G	Точная установка системы координат детали на детали	м	+		PM-NC
PAROTOF [6]	G	Отключить относящееся к детали вращение фрейма	м	+		PM-NC
PCALL	K	Вызвать подпрограммы с абсолютным указанием пути и передачей параметров		+		PM-NC
PDELAYOF	G	Задержка при штамповке ВЫКЛ	м	+		PM-NC
PDELAYON [6]	G	Задержка при штамповке ВКЛ	м	+		PM-NC
PHI	K	Угол поворота ориентации вокруг оси направления конуса		+		PM-NC
PHU	K	Физическая единица переменной		+		PM-NC
PL	A	1. B-сплайн: Расстояние между узловыми точками 2. Полиномиальная интерполяция: длина интервала параметров при полиномиальной интерполяции	п	+		PM-NC
PM	K	В минуту		+		PM-NC
po	K	Коэффициент полинома при полиномиальной интерполяции	п	+		PM-NC
POCKET3	C (T)	Фрезерование прямоугольного кармана		+		PM-NC
POCKET4	C (T)	Фрезерование кругового кармана		+		PM-NC
POLF	K	Переходная позиция LIFTFAST	м	+		PM-NC
POLFA	P	Запустить переходную позицию отдельных осей с $AA_ESR_TRIGGER	м	+	+	PM-NC
POLFMASK	P	Разрешить оси для обратного хода без связи между осями	м	+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
POLFMLIN	P	Разрешить оси для обратного хода с линейной связью между осями	м	+	-	PM-NC
POLY	G	полиномиальная интерполяция	м	+		PM-NC
POLYPATH	P	Полиномиальная интерполяция с возможностью выбора для групп осей AXIS или VECT	м	+	-	PM-NC
PON	G	Штамповка ВКЛ	м	+		PM-NC
PONS	G	Штамповка ВКЛ в такте IPO	м	+		PM-NC
Pos	K	Позиционировать ось		+	+	PM-NC
POSA	K	Позиционировать ось за границу кадра		+	+	PM-NC
POSM	P	Позиционирование магазина		+	-	FM-TM
POSMT	P	Переместить мультиинструмент в зажиме на номер места		+	-	FM-TM
POSP	K	Посегментное позиционирование (качание)		+		PM-NC
POSRANGE	F	Определить, находится ли актуальная интерполированная заданная позиция оси в окне вокруг заданной контрольной позиции		+	+	FM-SA
POT	F	Квадрат (арифметическая функция)		+	+	PM-NC
PR	K	На оборот		+		PM-NC
PREPRO	PA	Обозначить подпрограммы с подготовкой		+		PM-NC
PRESETON	P	Установка фактического значения с потерей состояния реферирования		+	+	PM-NC
PRESETONS	P	Установка фактического значения без потери состояния реферирования		+	+	PM-NC
Prio	K	Кодовое слово для установки приоритета при обработке прерываний		+		PM-NC
PRLOC	K	Инициализация переменных при Reset только после локального изменения		+		PM-NC
PROC	K	Первый оператор программы		+		PM-NC
PROTA	P	Запрос повторного расчета модели столкновений		+		PM-NC
PROTD	F	Определение интервала между двумя защищенными областями		+		PM-NC
PROTS	P	Установка состояния защищенной области		+		PM-NC
PSI	K	Апертурный угол конуса		+		PM-NC
PTP	G	Позиционное движение (PTP-перемещение)	м	+		PM-NC
PTPG0	G	Позиционное движение только для G0, в остальных случаях CP	м	+		PM-NC
PTPWOC	G	Позиционное движение без компенсирующих движений, обусловленных изменением ориентации	м	+		PM-NC
PUNCHACC	P	Зависящее от пути ускорение при вырубке		+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
PUTFTOC	P	Точная коррекция инструмента для параллельной правки		+	-	PM-NC
PUTFTOCF	P	Точная коррекция инструмента в зависимости от установленной с FCTDEF функции для параллельной правки		+	-	PM-NC
PW	A	B-сплайн, вес точки	п	+		PM-NC
Qu	K	Быстрый вывод дополнительной (вспомогательной) функции		+		PM-NC
R ...	A	R-параметр и как устанавливаемый идентификатор адреса и с числовым расширением		+		PM-NC
RAC	K	Абсолютное покадровое спец. для оси программирование радиуса	п	+		PM-NC
RDISABLE	P	Блокировка загрузки		-	+	FM-SA
READ	P	Считывает в указанном файле одну или несколько строк и сохраняет считанную информацию в поле		+	-	PM-NC
REAL	K	Тип данных: Переменная с плавающей запятой со знаком (вещественные числа)		+		PM-NC
REDEF	K	Переопределение системных переменных, переменных пользователя и языковых команд ЧПУ		+		PM-NC
RELEASE	P	Разрешить оси станка для перехода оси		+	+	PM-NC
REP	K	Кодовое слово для инициализации всех элементов поля с одним и тем же значением		+		PM-NC
REPEAT	K	Повторение программного цикла		+		PM-NC
REPEATB	K	Повторение программной строки		+		PM-NC
REPOSA	G	Повторный подвод к контуру линейный всеми осями	п	+		PM-NC
REPOSH	G	Повторный подвод к контуру по полукругу	п	+		PM-NC
REPOSHA	G	Повторный подвод к контуру всеми осями; геометрические оси по полукругу	п	+		PM-NC
REPOSL	G	Повторный подвод к контуру линейный	п	+		PM-NC
REPOSQ	G	Повторный подвод к контуру по четверти круга	п	+		PM-NC
REPOSQA	G	Повторный подвод к контуру линейно всеми осями; геометрические оси по четверти круга	п	+		PM-NC
RESETMON	P	Языковая команда для активации заданного значения		+	-	FM-TM
RET	P	Конец подпрограммы		+	+	PM-NC
RETB	P	Конец подпрограммы		+	+	PM-NC
RIC	K	Относительное покадровое спец. для оси программирование радиуса	п	+		PM-NC
RINDEX	F	Определить индекс символа во входной строке		+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
RMB	G	Повторный подвод к начальной точке кадра	м	+		PM-NC
RMBBL	G	Повторный подвод к начальной точке кадра	п	+		PM-NC
RME	G	Повторный подвод к конечной точке кадра	м	+		PM-NC
RMEBL	G	Повторный подвод к конечной точке кадра	п	+		PM-NC
RMI [6]	G	Повторный подвод к точке прерывания	м	+		PM-NC
RMIBL [6]	G	Повторный подвод к точке прерывания	п	+		PM-NC
RMN	G	Повторный подвод к следующей точке траектории	м	+		PM-NC
RMNBL	G	Повторный подвод к следующей точке траектории	п	+		PM-NC
RND	A	Закругление угла контура	п	+		PM-NC
RNDM	A	Модальное закругление	м	+		PM-NC
ROT	G	Программируемое вращение	п	+		PM-NC
ROTS	G	Программируемые вращения фрейма с пространственными углами	п	+		PM-NC
ROUND	F	Округление мест после запятой		+	+	PM-NC
ROUNDUP	F	Округление входного значения в сторону увеличения		+	+	PM-NC
RP	A	Полярный радиус	м/п	+		PM-NC
RPL	A	Вращение в плоскости	п	+		PM-NC
RT	K	Параметры для доступа к данным фрейма: Вращение		+		PM-NC
RTLIOF	G	G0 без линейной интерполяции (интерполяция отдельной оси)	м	+		PM-NC
RTLION [6]	G	G0 с линейной интерполяцией	м	+		PM-NC

Операторы S ... Z

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
S	A	Частота вращения шпинделя (для G4, G96/G961 другое значение)	м/п	+	+	PM-NC
Save	PA	Атрибут для сохранения информации при вызовах подпрограмм		+		PM-NC
SBLOF	P	Подавление отдельного кадра		+	-	PM-NC
SBLON	P	Отменить подавление отдельного кадра		+	-	PM-NC
sc	K	Параметры для доступа к данным фрейма: Масштабирование		+		PM-NC
SCALE	G	Программируемое масштабирование	п	+		PM-NC
SCC	K	Выборочное согласование поперечной оси с G96/G961/G962. Идентификаторами оси могут быть гео-ось, ось канала или ось станка.		+		PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
SCPARA	K	Программирование блока параметров Servo		+	+	PM-NC
SD	A	Порядок сплайна	п	+		PM-NC
set	K	Кодовое слово для инициализации всех элементов поля с перечисленными значениями		+		PM-NC
SETAL	P	Установить аварийное сообщение		+	+	PM-NC
SETDNO	F	Согласовать номер D резца (CE) инструмента (T)		+	-	PM-NC
SETINT	K	Определение, какой обработчик прерываний должен быть активирован, если вход ЧПУ готов		+		PM-NC
SETM	P	Установка маркеров в собственном канале		+	+	PM-NC
SETMS	P	Переключение на установленный в машинных данных мастер-шпиндель		+	-	PM-NC
SETMS(n)	P	Шпиндель n должен считаться мастер-шпинделем		+		PM-NC
SETMTH	P	Установить номер ведущего инстр. суппорта		+	-	FM-TM
SETPIECE	P	Учитывать количество штук для всех инструментов, которые согласованы со шпинделем.		+	-	FM-TM
SETTA	P	Активация инструмента из структуры износа		+	-	FM-TM
SETTCOR	F	Изменение компонентов инструмента с учетом всех граничных условий		+	-	PM-NC
SETTIA	P	Деактивация инструмента из структуры износа		+	-	FM-TM
SF	A	Смещение стартовой точки для резьбонарезания	м	+		PM-NC
SIN	F	Синус (тригонометрическая функция)		+	+	PM-NC
SIRELAY	F	Активировать спараметрированные со SIRELIN, SIRELOUT и SIRELTIME функции безопасности		-	+	FM-SI
SIRELIN	P	Инициализировать входные величины функционального блока		+	-	FM-SI
SIRELOUT	P	Инициализировать выходные величины функционального блока		+	-	FM-SI
SIRELTIME	P	Инициализировать таймер функционального блока		+	-	FM-SI
SLOT1	C (T)	Продольный паз		+		PM-NC
SLOT2	C (T)	Кольцевая канавка		+		PM-NC
SOFT	G	Ускорение по траектории с ограничением рывка	м	+		PM-NC
SOFTA	P	Включение осевого ускорения с ограничением рывка для запрограммированных осей		+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
SON	G	Вырубка ВКЛ	м	+		PM-NC
SONS	G	Вырубка ВКЛ в такте IPO	м	+		PM-NC
SPATH [6]	G	Соотношение траекторий для осей FGROUP это длина дуги	м	+		PM-NC
SPCOF	P	Переключение мастер-шпинделя или шпинделя(ей) из режима управления по положению в режим управления по частоте вращения	м	+	-	PM-NC
SPCON	P	Переключение мастер-шпинделя или шпинделя(ей) из режима управления по частоте вращения в режим управления по положению	м	+	-	PM-NC
SPI	F	Преобразует номер шпинделя в идентификатор оси		+	-	PM-NC
SPIF1 [6]	G	Быстрые входы/выходы ЧПУ для штамповки/вырубки Байт 1	м	+		FM-TE
SPIF2	G	Быстрые входы/выходы ЧПУ для штамповки/вырубки Байт 2	м	+		FM-TE
SPLINEPATH	P	Определение соединения сплайнов		+	-	PM-NC
SPN	A	Кол-во участков пути на кадр	п	+		PM-NC
SPOF [6]	G	Ход ВЫКЛ, штамповка, вырубка ВЫКЛ	м	+		PM-NC
SPOS	K	Позиция шпинделя	м	+	+	PM-NC
SPOSA	K	Позиция шпинделя за границы кадра	м	+		PM-NC
SPP	A	Длина участка пути	м	+		PM-NC
SPRINT	F	Возвращает отформатированную входную строку		+		PM-NC
SQRT	F	Квадратный корень (арифметическая функция) (square root)		+	+	PM-NC
SR	A	Путь обратного хода качания для синхронного действия	п	+		PM-NC
SRA	K	Путь обратного хода качания при внешнем входе осевом для синхронного действия	м	+		PM-NC
ST	A	Время выхаживания качания для синхронного действия	п	+		PM-NC
STA	K	Время выхаживания качания осевого для синхронного действия	м	+		PM-NC
START	P	Запуск выбранных программ в нескольких каналах одновременно из текущей программы		+	-	PM-NC
STARTFIFO [6]	G	Выполнение; параллельное заполнение буфера предварительной обработки	м	+		PM-NC
STAT		Положение шарниров	п	+		PM-NC
STOLF	A	Коэффициент допуска G0	м	+		PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
STOPFIFO	G	Остановка обработки; заполнение буфера предварительной обработки, до распознания STARTFIFO, пока буфер предварительной обработки не будет заполнен или до конца программы	м	+		PM-NC
stopre	P	Остановка предварительной обработки до выполнения всех подготовленных кадров главного хода		+	-	PM-NC
STOPREOF	P	Отменить остановку предварительной обработки		-	+	FM-SA
String	K	Тип данных: строка символов		+		PM-NC
STRINGIS	F	Проверяет имеющуюся языковую среду ЧПУ и специально относящиеся к этой команде имена циклов ЧПУ, переменные пользователя, макросы и имена меток, существуют ли они, определены ли они или активны.		+	-	PM-NC
STRLEN	F	Определить длину строки		+	-	PM-NC
SUBSTR	F	Определить индекс символа во входной строке		+	-	PM-NC
SUPA	G	Подавление актуального смещения нулевой точки, вкл. запрограммированные смещения, системные фреймы, смещения маховичком (DRF), внешнее смещение нулевой точки и наложенное движение	п	+		PM-NC
SUPD	G	Подавление активных коррекций инструмента	п	+	-	PM-NC
SVC	K	Скорость резания инструмента	м	+		PM-NC
SYNFCT	P	Обработка полинома в зависимости от условия в синхронном действии движения		-	+	FM-SA
SYNR	K	Чтение переменной выполняется синхронного, т.е. на момент времени выполнения		+		PM-NC
SYNRW	K	Чтение и запись переменной выполняется синхронного, т.е. на момент времени выполнения		+		PM-NC
SYNW	K	Запись переменной выполняется синхронного, т.е. на момент времени выполнения		+		PM-NC
T	A	Вызвать инструмент (смена только если установлено в машинных данных; в ином случае необходима команда M6)		+		PM-NC
TAN	F	Тангенс (тригонометрическая функция)		+	+	PM-NC
TANG	P	Тангенциальное управление: Определить соединение		+	-	PM-NC
TANGDEL	P	Тангенциальное управление: Удаление соединения		+	-	PM-NC
TANGOF	P	Тангенциальное управление: Выключение соединения		+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
TANGON	P	Тангенциальное управление: Включить соединение		+	-	PM-NC
TCA (828D: _TCA)	P	Выбор инструмента / смена инструмента независимо от состояния инструмента		+	-	FM-TM
TCARR	A	Запросить инструментальный суппорт (номер "m")		+		PM-NC
TCI	P	Поместить инструмент из буфера в магазин		+	-	FM-TM
TCOABS [6]	G	Определение компонентов длин инструмента из текущей ориентации инструмента	м	+		PM-NC
TCOFR	G	Определить компоненты длин инструмента из ориентации активного фрейма	м	+		PM-NC
TCOFRX	G	Определение ориентации инструмента активного фрейма при выборе инструмента, инструмент показывает в направлении X	м	+		PM-NC
TCOFRY	G	Определение ориентации инструмента активного фрейма при выборе инструмента, инструмент показывает в направлении Y	м	+		PM-NC
TCOFRZ	G	Определение ориентации инструмента активного фрейма при выборе инструмента, инструмент показывает в направлении Z	м	+		PM-NC
THETA	A	Угол поворота	п	+		PM-NC
TILT	A	Боковой угол	м	+		PM-NC
TLIFT	P	Тангенциальное управление: Включение создания промежуточного кадра		+	-	PM-NC
TML	P	Выбор инструмента с номером места в магазине		+	-	FM-TM
TMOF	P	Отменить контроль инструмента		+	-	PM-NC
TMON	P	Выбрать контроль инструмента		+	-	PM-NC
TO	K	Обозначает конечное значение в цикле вычисления FOR		+		PM-NC
TOFF	A	Смещение длин инструмента в направлении компонента длин инструмента, действующего параллельно указанной в индексе гео-оси.	м	+		PM-NC
TOFFL	A	Смещение длин инструмента в направлении компонента длин инструмента L1, L2 или L3	м	+		PM-NC
TOFFLR	A	Одновременное смещение длины инструмента и радиуса инструмента	м	+		PM-NC
TOFFOF	P	Сбросить онлайн-коррекцию длины инструмента		+	-	PM-NC
TOFFON	P	Активировать онлайн-коррекцию длины инструмента		+	-	PM-NC
TOFFR	A	Смещение радиуса инструмента	м	+		PM-NC
TOFRAME	G	Точно установить ось Z СКД через вращение фрейма параллельно ориентации инструмента	м	+		PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
TOFRAMEX	G	Точно установить ось X СКД через вращение фрейма параллельно ориентации инструмента	м	+		PM-NC
TOFRAMEY	G	Точно установить ось Y СКД через вращение фрейма параллельно ориентации инструмента	м	+		PM-NC
TOFRAMEZ	G	как TOFRAME	м	+		PM-NC
TOLOWER	F	Преобразовать буквы строки в строчные буквы		+	-	PM-NC
TOOLENV	F	Сохранить все состояния, важные для нормирования находящихся в памяти параметров инструмента		+	-	PM-NC
TOOLGNT	F	Определить число инструментов одной группы инструментов		+	-	FM-TM
TOOLGT	F	Определить T-номер инструмента из группы инструментов		+	-	FM-TM
TOROT	G	Точно установить ось Z СКД через вращение фрейма параллельно ориентации инструмента	м	+		PM-NC
TOROTOF [6]	G	Вращение фрейма в направлении инструмента ВЫКЛ	м	+		PM-NC
TOROTX	G	Точно установить ось X СКД через вращение фрейма параллельно ориентации инструмента	м	+		PM-NC
TOROTY	G	Точно установить ось Y СКД через вращение фрейма параллельно ориентации инструмента	м	+		PM-NC
TOROTZ	G	как TOROT	м	+		PM-NC
TOUPPER	F	Преобразовать буквы строки в прописные буквы		+	-	PM-NC
TOWBCS	G	Значения износа в базовой кинематической системе (BKS)	м	+		PM-NC
TOWKCS	G	Значения износа в системе координат инструментальной головки при кинематической трансформации (отличается от MCS вращением инструмента)	м	+		PM-NC
TOWMCS	G	Значения износа в системе координат станка (MCS)	м	+		PM-NC
TOWSTD [6]	G	Значение по умолчанию для коррекции по длине инструмента	м	+		PM-NC
TOWTCS	G	Значения износа в системе координат инструмента (исходная точка инстр. суппорта T на креплении суппорта)	м	+		PM-NC
TOWWCS	G	Значения износа в системе координат детали (WCS)	м	+		PM-NC
tr	K	Компонент смещения переменной фрейма		+		PM-NC
TRAANG	P	Трансформация наклонной оси		+	-	PM-NC
TRACON	P	Каскадированная трансформация		+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
TRACYL	P	Цилиндр: трансформация боковой поверхности		+	-	PM-NC
TRAFOOF	P	Отключить активные в канале трансформации		+	-	PM-NC
TRAFOON	P	Включение трансформаций, определенных посредством кинематических цепей		+	-	PM-NC
TRAILOF	P	Осевая синхронная буксировка ВЫКЛ		+	+	PM-NC
TRAILON	P	Осевая синхронная буксировка ВКЛ		+	+	PM-NC
TRANS	G	Абсолютное программируемое смещение нулевой точки	п	+		PM-NC
Transmit	P	Полярная трансформация (обработка торцовых поверхностей)		+	-	PM-NC
TRAORI	P	4-, 5-осевая трансформация, базовая трансформация		+	-	PM-NC
true	K	Логическая постоянная: истинно		+		PM-NC
TRUNC	F	Отсечение мест после запятой		+	+	PM-NC
Tu		Осевой угол	п	+		PM-NC
Turn	A	Число витков для спиральной линии	п	+		PM-NC
ULI	K	Верхнее предельное значение переменной		+		PM-NC
UNLOCK	P	Разрешить синхронное действие с ID (продолжить технологический цикл)		-	+	FM-SA
UNTIL	K	Условие для завершения цикла REPEAT		+		PM-NC
UPATH	G	Соотношение траекторий для осей FGROUP это параметр кривой	м	+		PM-NC
Var	K	Кодовое слово: тип передачи параметров		+		PM-NC
VELOLIM	K	Коррекция максимальной скорости оси или частоты вращения шпинделя	м	+	-	PM-NC
VELOLIMA	K	Уменьшение или увеличение макс. скорости ведомой оси	м	+	+	PM-NC
WAITC	P	Ожидание выполнения критерия смены кадра соединения для оси/шпинделя		+	-	PM-NC
WAITE	P	Ожидание конца программы в другом канале.		+	-	PM-NC
WAITENC	P	Ожидание синхронизированных или восстановленных позиций осей		+	-	PM-NC
WAITM	P	Ожидание метки в указанном канале; завершить предшествующий кадр с точным остановом.		+	-	PM-NC
WAITMC	P	Ожидание метки в указанном канале; точный останов только если другие каналы еще не достигли метки		+	-	PM-NC
WAITP	P	Ожидание конца перемещения позиционирующей оси		+	-	PM-NC
WAITS	P	Ожидание достижения позиции шпинделя		+	-	PM-NC
WALCS0 [6]	G	Ограничение рабочего поля WCS отключено	м	+	-	PM-NC

Оператор	Тип [1]	Объяснение	W [2]	TP [3]	SA [4]	Описание см. [5]
[1] [2] [3] [4] [5] **Объяснения см. Экспликация (Страница 1247).**						
WALCS1	G	Группа 1 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALCS2	G	Группа 2 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALCS3	G	Группа 3 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALCS4	G	Группа 4 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALCS5	G	Группа 5 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALCS6	G	Группа 6 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALCS7	G	Группа 7 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALCS8	G	Группа 8 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALCS9	G	Группа 9 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALCS10	G	Группа 10 ограничений рабочего поля СКД активна	м	+	-	PM-NC
WALIMOF	G	Ограничение рабочего поля BKS ВЫКЛ	м	+	-	PM-NC
WALIMON [6]	G	Ограничение рабочего поля BKS ВКЛ	м	+	-	PM-NC
WHEN	K	Операция будет выполнена один раз, если условие выполнено один раз.		-	+	FM-SA
WHENEVER	K	Операция выполняется циклически в каждом такте интерполяции, если условие выполнено.		-	+	FM-SA
WHILE	K	Начало программного цикла WHILE		+		PM-NC
write	P	Записать текст в файловую систему. Добавляет кадр в конце указанного файла.		+	-	PM-NC
WRTPR	P	Запись строки символов в переменную BTSS		+	-	PM-NC
X	A	Имя оси	м/п	+	+	PM-NC
XOR	O	Логическая исключающая ИЛИ		+		PM-NC
Y	A	Имя оси	м/п	+	+	PM-NC
Z	A	Имя оси	м/п	+	+	PM-NC

Легенда

[1] Тип оператора:

A Адрес

Идентификатор, которому присваивается значение (к примеру, OVR=10). Также есть несколько адресов, которые включают или выключают функцию без присвоения значения (к примеру, CPLON и CPLOF).

C (A) Цикл AST

Предопределенная программа ЧПУ для дополнительной автоматической оптимизации с AST (= автоматическая оптимизация Servo) Согласование с конкретной ситуацией оптимизации осуществляется по параметрам, которые передаются циклу при вызове.

C (M) Цикл измерения

Предопределенная программа ЧПУ, в которой универсально запрограммирован определенный процесс измерения, например, определение внутреннего диаметра цилиндрической детали. Согласование с конкретной ситуацией измерения осуществляется по параметрам, которые передаются циклу при вызове.

C (T) Технологический цикл

Предопределенная программа ЧПУ, в которой универсально запрограммирован определенный процесс обработки, например, нарезание внутренней резьбы или фрезерование кармана. Адаптация к конкретной ситуации обработки осуществляется через параметры, которые передаются циклу при вызове.

F Предопределенная функция (выводит возвращаемое значение)

Вызов предопределенной функции может быть указан как операнд в выражении.

G G-команда

G-команды разделены на группы. В одном кадре можно записать только одну G-команду группы. G-команда может действовать модально (до отмены через другую команду этой же группы), или она действует только для кадра, в котором она стоит (действует покадрово).

K Кодовое слово

Идентификатор, определяющий синтаксис кадра. Кодовому слову значение не присваивается и включение/выключение функции ЧПУ с помощью кодового слова невозможно.

Примеры: Управляющие структуры (IF, ELSE, ENDIF, WHEN, ...), выполнение программы (GOTOB, GOTO, RET ...)

O Оператор

Оператора для математической операции, операции сравнения или логической операции

P Предопределенная процедура (не выводит возвращаемое значение)

PA Атрибут программы

Атрибуты программы стоят в конце строки определения подпрограммы:
`PROC` <имя программы>(...) <атрибут программы>

Они определяют поведение при выполнении подпрограммы.

[2] Активность оператора:

м модально

п покадрово

[3] Возможность программирования в программе обработки детали:
+ программируемые
- не программируемые
M программируется только изготовителем станка

[4] Возможность программирования в синхронных действиях:
+ программируемые
- не программируемые
T программируется только в технологических циклах

[5] Ссылка на документ, содержащий подробное описание оператора:
FM-A Описание функций – Оси и шпиндели
FM-B Описание функций – Основные функции
FM-SA Описание функций "Синхронные действия"
FM-SI Описание функций Safety Integrated
FM-TE Описание функций – Технологии
FM-TM Описание функций "Управление инструментом"
FM-TR Справочник по функциям — Трансформации
PM-MC Руководство по программированию — Измерительные циклы
PM-NC Руководство по программированию — Программирование ЧПУ

[6] Стандартная установка в начале программы (в состоянии СЧПУ при поставке, когда еще ничего другого не запрограммировано).

5.2 Адреса

5.2.1 Буквы адреса

Буква	Объяснение	Цифровое расширение
A	Настраиваемый идентификатор адреса	x
B	Настраиваемый идентификатор адреса	x
C	Настраиваемый идентификатор адреса	x
D	Включение/выключение коррекции длин инструмента, резца инструмента	
E	Настраиваемый идентификатор адреса	x
F	Подача Время ожидания в секундах	x
g	G-команда	
H	Функция H	x
I	Настраиваемый идентификатор адреса	x
J	Настраиваемый идентификатор адреса	x
K	Настраиваемый идентификатор адреса	x

Буква	Объяснение	Цифровое расширение
L	Подпрограммы, -вызов	
M	Функция M	x
N	Номер вспомогательного кадра	
O	свободно	
P	Число прогонов программы	
Q	Настраиваемый идентификатор адреса	x
R	Идентификатор переменной (R-параметр) Настраиваемый идентификатор адреса (без числового расширения)	x
S	Значение шпинделя Время ожидания в оборотах шпинделя	x x
T	Номер инструмента	x
U	Настраиваемый идентификатор адреса	x
V	Настраиваемый идентификатор адреса	x
W	Настраиваемый идентификатор адреса	x
X	Настраиваемый идентификатор адреса	x
Y	Настраиваемый идентификатор адреса	x
Z	Настраиваемый идентификатор адреса	x
%	Символ начала и разделения при передаче файлов	
:	Номер главного кадра	
/	Символ пропуска кадра	

5.2.2 Постоянные адреса

Фиксированные адреса без осевого расширения

Идентификатор адреса	Тип адреса	модально/покадрово	G70/G71	G700/G710	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	Qu	Тип данных присвоенного значения
D	Номер коррекции	м								x	без знака INT
F	Подача, время ожидания	м, п	x							x	без знака REAL
FLIM	Максимальная скорость движения по траектории	м									без знака REAL

Идентификатор адреса	Тип адреса	модально/покадрово	G70/G71	G700/G710	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	Qu	Тип данных присвоенного значения
g	G-команда	см. список Gфункций									без знака INT
H	Вспомогательные функции	п								x	M: без знака INT H: REAL
I	Номер подпрограммы	п									без знака INT
M	Вспомогательные функции	п								x	M: без знака INT H: REAL
N	Номер кадра	п									без знака INT
OVR	Процентовка	м									без знака REAL
OVRRAP	Процентовка для скорости ускоренного хода	м									без знака REAL
P	Число прогонов подпрограммы	п									без знака INT
PACCLIM	Максимальное ускорение по траектории	м									без знака REAL
S	Шпиндель, время ожидания	м, п								x	без знака REAL
SCC	Согласование поперечной оси с G96 /G961/G962	м									REAL
SPOS	Позиция шпинделя	м				x	x	x			REAL

Идентификатор адреса	Тип адреса	модально/покадрово	G70/G71	G700/G710	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	Qu	Тип данных присвоенного значения
SPOSA	Позиция шпинделя за границы кадра	м				x	x	x			REAL
T	Номер инструмента	м								x	без знака INT

Фиксированные адреса с осевым расширением

Идентификатор адреса	Тип адреса	модально/покадрово	G70/G71	G700/G710	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	Qu	Тип данных присвоенного значения
ACC	Осевое ускорение	м									без знака REAL
ACCLIMA	Осевое ограничение ускорения ведомой оси	м									без знака REAL
AX	Переменный идентификатор оси	[1]	x	x	x	x	x	x			REAL
FA	Осевая подача	м	x							x	без знака REAL
FDA	Осевая подача для наложения маховичка	п	x								без знака REAL
FGREF	Исходный радиус	м	x	x							без знака REAL
FL	Осевая предельная подача	м	x								без знака REAL
FMA	Осевая синхронная подача	м									без знака REAL
FOC	Покадровое движение с ограниченным моментом	п									REAL

Идентификатор адреса	Тип адреса	модально/покадрово	G70/G71	G700/G710	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	Qu	Тип данных присвоенного значения
FOCOF	Модальное движение с ограниченным моментом ВЫКЛ	м									REAL
FOCON	Модальное движение с ограниченным моментом ВКЛ	м									REAL
FXS	Наезд на жесткий упор ВКЛ	м									без знака INT
FXST	Граница момента для наезда на жесткий упор	м									REAL
FXSW	Окно контроля для наезда на жесткий упор	м									REAL
IP	Переменный параметр интерполяции	п	x	x	x	x	x				REAL
JERKLIM	Осевое ограничение рывка	м									без знака REAL
JERKLIMA	Осевое ограничение рывка ведомой оси	м									без знака REAL
MEAC	Циклическое измерение	п									INT режим и 1 - 4 пусковые события
MEASA	Осевое измерение со стиранием остатка пути	п									INT режим и 1 - 4 пусковые события

Идентификатор адреса	Тип адреса	модально/покадрово	G70/G71	G700/G710	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	Qu	Тип данных присвоенного значения
MEAWA	Осевое измерение без стирания остатка пути	п									INT режим и 1 - 4 пусковые события
MOV	Запустить позиционирующую ось	м	x	x	x	x	x	x	x		REAL
OS	ВКЛ/ВЫКЛ качания	м									без знака INT
OSB	Стартовая точка качания	м	x	x	x	x	x	x			REAL
OSCILL	Согласование оси для качания, включить качание	м									Ось: 1 - 3 оси подачи
OSCTRL	Опции качания	м									без знака INT: опции установки, без знака INT: опции сброса
OSE	Конечная точка качания	м	x	x	x	x	x	x			REAL
OSNSC	Число выхаживаний качания	м									без знака INT
OSP1	Левая точка возврата (качание)	м	x	x	x	x	x	x			REAL
OSP2	Правая точка возврата (качание)	м	x	x	x	x	x	x			REAL
OST1	Время удержания в левой точке возврата (качание)	м									REAL
OST2	Время удержания в правой точке возврата (качание)	м									REAL

Идентификатор адреса	Тип адреса	модально/покадрово	G70/G71	G700/G710	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	Qu	Тип данных присвоенного значения
OVRA	Осевая процентовка	м	x								без знака REAL
PO	Коэффициент полинома	п	x	x		x	x	x			без знака REAL
POLF	Позиция LIFTFAST	м	x	x							без знака REAL
POS	Позиционирующая ось	м	x	x	x	x	x	x	x		REAL
POSA	Позиционирующая ось за границы кадра	м	x	x	x	x	x	x	x		REAL
POSP	Посегментное позиционирование (качание)	м	x	x	x	x	x	x			REAL: конечная позиция Real: длина фрагмента INT: опция
STA	Осевое время выхаживания	м									без знака REAL
SRA	Путь отвода при внешнем осевом входе	м									без знака REAL
VELOLIM	Осевое ограничение скорости	м									без знака REAL
VELOLIMA	Осевое ограничение скорости ведомой оси	м									без знака REAL

[1] Абсолютные конечные точки: Модальные, инкрементальные конечные точки: покадрово, в иных случаях модально/покадрово в зависимости от определяющей синтаксис G-команды

5.2.3 Устанавливаемые адреса

Идентификатор адреса (стандартная настройка)	Тип адреса	модально/покадрово	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Макс. число	Тип данных присвоенного значения
Осевые значения и конечные точки											
X, Y, Z, A, B, C	Ось	[1]	x	x	x	x				8	REAL
AP	Полярный угол	м/п [1]	x	x	x					1	REAL
RP	Полярный радиус	м/п [1]	x	x	x					1	без знака REAL
Ориентация инструмента											
A2, B2, C2	Угол Эйлера или угол RPY	п								3	REAL
A3, B3, C3	Компоненты вектора направления	п								3	REAL
A4, B4, C4	Компоненты вектора нормали к плоскости в начале кадра	п								3	REAL
A5, B5, C5	Компоненты вектора нормали к плоскости на конце кадра	п								3	REAL
A6, B6, C6	Компоненты вектора направления для оси вращения конуса	п								3	REAL
A7, B7, C7	Компоненты вектора для промежуточной ориентации на боковой поверхности конуса	п								3	REAL
LEAD	Угол предварения	м								1	REAL
THETA	Угол поворота Вращение вокруг направления инструмента	м		x	x					1	REAL
TILT	Боковой угол	м								1	REAL

Идентификатор адреса (стандартная настройка)	Тип адреса	модально/покадрово	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Макс. число	Тип данных присвоенного значения
ORIS	Изменение ориентации инструмента (относительно траектории)	м								1	REAL
Параметры интерполяции											
I, J, K	Параметры интерполяции Координата промежуточной точки	п		x [2)]	x [2)]					3	REAL
I1, J1, K1		п	x	x	x					3	REAL
RPL	Вращение в плоскости	п								1	REAL
CR	Радиус окружности	п								1	без знака REAL
AR	Апертурный угол	п								1	без знака REAL
TURN	Число витков для спиральной линии	п								1	без знака INT
PL	Длина интервала параметра	п								1	без знака REAL
PW	Вес точки	п								1	без знака REAL
SD	Порядок сплайна	м								1	без знака INT
TU	Осевой угол	п								1	без знака INT
STAT	Положение шарниров	м								1	без знака INT
SF	Смещение стартовой точки для резьбонарезания	м								1	REAL
DISCL	Безопасное расстояние SAR	п								1	без знака REAL
DISR	Repos-расстояние / SAR-расстояние	п								1	без знака REAL
DISPR	Рассогласование траектории Repos	п								1	без знака REAL
ALF	Угол быстрого отвода	м								1	без знака INT

Идентификатор адреса (стандартная настройка)	Тип адреса	модально/покадрово	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Макс. число	Тип данных присвоенного значения
DILF	Длина быстрого отвода	м								1	REAL
FP	Фиксированная точка: номер фикс. точки, к которой осуществляется подвод	п								1	без знака INT
RNDM	Модальное закругление	м								1	без знака REAL
RND	Покадровое закругление	п								1	без знака REAL
CHF	Покадровая фаска	п								1	без знака REAL
CHR	Фаска а первоначальном направлении движения	п								1	без знака REAL
ANG	Угол линии контура	п								1	REAL
ISD	Глубина врезания	м								1	REAL
DISC	Превышение переходной окружности коррекция радиуса инструмента	м								1	без знака REAL
OFFN	Смещение нормального контура	м								1	REAL
DITS	Входной участок резьбы	м								1	REAL
DITE	Выходной участок резьбы	м								1	REAL
Критерии перешлифовки											
ADIS	Интервал перешлифовки	м								1	без знака REAL
ADISPOS	Интервал перешлифовки для ускоренного хода	м								1	без знака REAL
Измерение											

Идентификатор адреса (стандартная настройка)	Тип адреса	модально/покадрово	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Макс. число	Тип данных присвоенного значения
MEAS	Измерение с контактным щупом	п								1	без знака INT
MEAW	Измерение контактным щупом без стирания остатка пути	п								1	без знака INT
Поведение оси, шпинделя											
LIMS	Ограничение скорости шпинделя	м								1	без знака REAL
COARSEA	Параметры смены кадра: точный останов грубый осевой	м									
FINEA	Параметры смены кадра: точный останов точный осевой	м									
IPOENDA	Параметры смены кадра: Стоп интерполятора осевой	м									
DIACYCOFA	Поперечная ось: осевое программирование диаметра ВЫКЛ в циклах	м									
DIAM90A	Поперечная ось: осевое программирование диаметра при G90	м									
DIAMCHAN	Поперечная ось: прием всех поперечных осей в состояние канала программирования диаметра	м									

Идентификатор адреса (стандартная настройка)	Тип адреса	модально/покадрово	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Макс. число	Тип данных присвоенного значения
DIAMCHANA	Поперечная ось: применение состояния канала программирования диаметра	м									
DIAMOFA	Поперечная ось: осевое программирование диаметра ВЫКЛ	м									
DIAMONA	Поперечная ось: осевое программирование диаметра ВКЛ	м									
GP	Позиция: косвенное программирование атрибутов позиции	м									
Подачи											
FAD	Скорость медленного движения подачи	п						x		1	без знака REAL
FD	Подача по траектории для наложения маховичка	п								1	без знака REAL
FRC	Подача для радиуса и фаски	п								1	без знака REAL
FRCM	Модальная подача для радиуса и фаски	м								1	без знака REAL
FB	Покадровая подача	п								1	без знака REAL
Вырубка/штамповка											
SPN	Кол-во участков пути на кадр	п								1	INT
SPP	Длина участка пути	м								1	REAL
Шлифование											
ST	Время выхаживания	п								1	без знака REAL

Идентификатор адреса (стандартная настройка)	Тип адреса	модально/покадрово	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Макс. число	Тип данных присвоенного значения
SR	Путь обратного хода	п								1	без знака REAL
Выбор инструмента											
TCARR	Инструментальный суппорт	м								1	INT
Управление инструментом											
DL	Суммарная коррекция инструмента	м								1	INT
Адреса OEM											
OMA1	Адрес OEM 1	м		x	x	x				1	REAL
OMA2	Адрес OEM 2	м		x	x	x				1	REAL
OMA3	Адрес OEM 3	м		x	x	x				1	REAL
OMA4	Адрес OEM 4	м		x	x	x				1	REAL
OMA5	Адрес OEM 5	м		x	x	x				1	REAL
Прочее											
CUTMOD	Изменение данных коррекции для вращающихся инструментов (в сочетании с ориентируемыми инструментальными суппортами)	м									INT
CUTMODK	Изменение данных коррекции для вращающихся инструментов (в сочетании с трансформациями ориентации, определяемыми посредством кинематических цепей)	м									STRING
TOFF	Смещение длин инструмента параллельно указанной геооси	м									REAL

Идентификатор адреса (стандартная настройка)	Тип адреса	модально/покадрово	G90/G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Макс. число	Тип данных присвоенного значения
TOFFL	Смещение длин инструмента в направлении компонента длин инструмента L1, L2 или L3	м									REAL
TOFFR	Смещение радиуса инструмента	м									REAL
TOFFLR	Одновременное смещение длины инструмента и радиуса инструмента	м									REAL

1) Абсолютные конечные точки: Модальные, инкрементальные конечные точки: покадрово, в иных случаях модально/покадрово в зависимости от определяющей синтаксис G-команды

2) В качестве центров окружности параметры IPO действуют инкрементально. С AC они могут быть запрограммированы абсолютно. При других значениях (например, шаг резьбы) изменение адреса игнорируется.

5.3 G-команды

5.3.1 G-команды

G-команды разделены на G-группы. В программах обработки деталей и синхронных действиях в одном кадре можно записать только одну G-команду одной G-группы. G-команда может действовать модально или покадрово.

Модально: до программирования другой G-команды той же G-группы.

5.3.2 G-группа 1: Действующие модально команды движения

G-команда	№ 1)	Объяснение	MD20150 2)	W 3)	STD 4)	
					SAG	MH
G0	1	Движение ускоренного хода	+	м		
G1	2	Линейная интерполяция	+	м	x	
G2	3	Круговая интерполяция по часовой стрелке	+	м		
G3	4	Круговая интерполяция против часовой стрелки	+	м		
CIP	5	Круговая интерполяция через промежуточную точку	+	м		

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ASPLINE	6	Акима-сплайн	+	м		
BSPLINE	7	B-сплайн	+	м		
CSPLINE	8	Кубический сплайн	+	м		
POLY	9	Полиномиальная интерполяция	+	м		
G33	10	Резьбонарезание с постоянным шагом	+	м		
G331	11	Нарезание внутренней резьбы	+	м		
G332	12	Отвод (нарезание внутренней резьбы)	+	м		
OEMIPO1	13	Зарезервировано	+	м		
OEMIPO2	14	Зарезервировано	+	м		
CT	15	Окружность с тангенциальным переходом	+	м		
G34	16	Резьбонарезание с линейной увеличивающимся шагом	+	м		
G35	17	Резьбонарезание с линейной уменьшающимся шагом	+	м		
INVCW	18	Эвольвентная интерполяция по часовой стрелке	+	м		
INVCCW	19	Эвольвентная интерполяция против часовой стрелки	+	м		
G335	20	Нарезание резьбы с выпуклым профилем по часовой стрелке	+	м		
G336	21	Нарезание резьбы с выпуклым профилем против часовой стрелки	+	м		

5.3.3 G-группа 2: Действующие не модально движения, время ожидания

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G4	1	Время ожидания, заранее определено по времени	-	п		
G63	2	Нарезание внутренней резьбы без синхронизации	-	п		
G74	3	Реферирование с синхронизацией	-	п		
G75	4	Подвод к фиксированной точке	-	п		
REPOSL	5	Повторный подвод к контуру линейный	-	п		
REPOSQ	6	Повторный подвод к контуру по четверти круга	-	п		
REPOSH	7	Повторный подвод к контуру по полукругу	-	п		
REPOSA	8	Повторный подвод к контуру линейный всеми осями	-	п		
REPOSQA	9	Повторный подвод к контуру всеми осями, геометрические оси по четверти круга	-	п		
REPOSHA	10	Повторный подвод к контуру всеми осями, геометрические оси по полукругу	-	п		
G147	11	Подвод к контуру по прямой	-	п		
G247	12	Подвод к контуру по четверти круга	-	п		
G347	13	Подвод к контуру по полукругу	-	п		

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G148	14	Выход из контура по прямой	-	п		
G248	15	Выход из контура по четверти круга	-	п		
G348	16	Выход из контура по полукругу	-	п		
G5	17	Шлифование с врезанием с угловой подачей	-	п		
G7	18	Компенсационное движение при шлифовании с врезанием с угловой подачей	-	п		

5.3.4 G-группа 3: Программируемый фрейм, ограничение рабочего поля и программирование полюса

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
TRANS	1	TRANSLATION: программируемое смещение	-	п		
ROT	2	ROTATION: программируемое вращение	-	п		
SCALE	3	Scale: программируемое масштабирование	-	п		
MIRROR	4	Mirror: программируемое отражение	-	п		
ATRANS	5	Аддитивная TRANSLATION: аддитивное программируемое смещение	-	п		
AROT	6	Аддитивная ROTATION: Программируемое вращение	-	п		
ASCALE	7	Аддитивная SCALE: программируемое масштабирование	-	п		
AMIRROR	8	Аддитивная MIRROR: Программируемое отражение	-	п		
-	9	свободно	-	-		
G25	10	Минимальное ограничение рабочего поля/ограничение частоты вращения шпинделя	-	п		
G26	11	Максимальное ограничение рабочего поля/ограничение частоты вращения шпинделя	-	п		
G110	12	Программирование полюса относительно последней запрограммированной заданной позиции	-	п		
G111	13	Программирование полюса относительно нулевой точки текущей системы координат детали	-	п		
G112	14	Программирование полюса относительно последнего действующего полюса	-	п		
G58	15	Абсолютное программируемое смещение нулевой точки	-	п		
G59	16	Аддитивное программируемое смещение нулевой точки	-	п		
ROTS	17	Вращение с пространственным углом	-	п		
AROTS	18	Аддитивное вращение с пространственным углом	-	п		

5.3.5 G-группа 4: FIFO

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
STARTFIFO	1	Старт FIFO Выполнение с параллельным заполнением буфера предварительной обработки	+	м	x	
STOPFIFO	2	Стоп FIFO, Остановка обработки; заполнение памяти предварительной обработки до распознавания STARTFIFO, заполнения памяти предварительной обработки или конца программы	+	м		
FIFOCTRL	3	Включение автоматического управления буфером предварительной обработки	+	м		

5.3.6 G-группа 6: Выбор плоскости

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G17	1	Выбор плоскости 1-й – 2-й гео-оси	+	м	x	
G18	2	Выбор плоскости 3-й – 1-й гео-оси	+	м		
G19	3	Выбор плоскости 2-й – 3-й гео-оси	+	м		

5.3.7 G-группа 7: коррекция радиуса инструмента

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G40	1	Нет коррекции радиуса инструмента	+	м	x	
G41	2	Коррекция радиуса инструмента слева от контура	-	м		
G42	3	Коррекция радиуса инструмента справа от контура	-	м		

5.3.8 G-группа 8: Настраиваемое смещение нулевой точки

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G500	1	Отключение настраиваемого смещения нулевой точки (G54 ... G57, G505 ... G599)	+	м	x	
G54	2	1-е настраиваемое смещение нулевой точки	+	м		

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G55	3	2-е настраиваемое смещение нулевой точки	+	м		
G56	4	3-е настраиваемое смещение нулевой точки	+	м		
G57	5	4-е настраиваемое смещение нулевой точки	+	м		
G505	6	5-е настраиваемое смещение нулевой точки	+	м		
...	...	...	+	м		
G599	100	99-е настраиваемое смещение нулевой точки	+	м		
С помощью G-команд этой G-группы активируется соответствующий настраиваемый фрейм пользователя $P_UIFR[ ]. G54 соответствует фрейм $P_UIFR[1], G505 соответствует фрейм $P_UIFR[5]. Число настраиваемых фреймов пользователя и тем самым число G-команд этой G-группы может настраиваться через MD28080 $MC_MM_NUM_USER_FRAMES.						

5.3.9 G-группа 9: Подавление фрейма и коррекции инструмента

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G53	1	Блокировка текущих фреймов: программируемый фрейм включая системный фрейм для TOROT и TOFRAME и активный настраиваемый фрейм (G54 ... G57, G505 ... G599)	-	п		
SUPA	2	Как G153, включая подавление системных фреймов для установки фактического значения, касания, внешнего смещения нулевой точки, PAROT, включая смещения маховичком (DRF), [внешнее смещение нулевой точки], наложенное движение	-	п		
G153	3	Как G53, включая подавление всех спец. для канала и/или глобальных базовых фреймов NCU	-	п		
SUPD	4	Подавление активных коррекций инструмента	-	п		

5.3.10 G-группа 10: Точный останов – режим управления траекторией

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G60	1	Точный останов	+	м	x	
G64	2	Режим управления траекторией	+	м		
G641	3	Режим управления траекторией с перешлифовкой по критерию пути (= программируемый интервал перешлифовки)	+	м		
G642	4	Режим управления траекторией с перешлифовкой с соблюдением определенных допусков	+	м		

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G643	5	Режим управления траекторией с перешлифовкой с соблюдением определенных допусков (внутри кадра)	+	м		
G644	6	Режим управления траекторией с перешлифовкой с макс. возможной динамикой	+	м		
G645	7	Режим управления траекторией с перешлифовкой углов с тангенциальными переходами кадров с соблюдением определенных допусков	+	м		

5.3.11 G-группа 11: Не модальный точный останов

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G9	1	Точный останов	-	п		

5.3.12 G-группа 12: Критерии смены кадра для точного останова (G60/G9)

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G601	1	Смена кадра при точном останове точном	+	м	x	
G602	2	Смена кадра при точном останове грубом	+	м		
G603	3	Смена кадра при конце кадра IPO	+	м		

5.3.13 G-группа 13: Дюймовая/метрическая постановка размеров детали

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G70	1	Дюймовая система ввода (длины)	+	м		
G71	2	Метрическая система ввода мм (длины)	+	м	x	
G700	3	Дюймовая система ввода дюймы, дюймов/мин (длины + скорость + системные переменные)	+	м		
G710	4	Метрическая система ввода мм; мм/мин (длины + скорость + системные переменные)	+	м		

5.3.14 G-группа 14: Абсолютная/инкрементальная постановка размеров детали

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G90	1	Указание абсолютного размера	+	м	x	
G91	2	Указание составного размера	+	м		

5.3.15 G-группа 15: Тип подачи

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G93	1	Обратная по времени подача 1/мин	+	м		
G94	2	Линейная подача в мм/мин, дюймов/мин	+	м	x	
G95	3	Окружная подача в мм/об, дюймов/об	+	м		
G96	4	Окружная подача (как у G95) и постоянная скорость резания	+	м		
G97	5	Окружная подача и постоянная частота вращения шпинделя (постоянная скорость резания ВЫКЛ)	+	м		
G931	6	Задача подачи через время перемещения, отключить постоянную скорость движения по траектории	+	м		
G961	7	Линейная подача (как у G94) и постоянная скорость резания	+	м		
G971	8	Линейная подача и постоянная частота вращения шпинделя (постоянная скорость резания ВЫКЛ)	+	м		
G942	9	Линейная подача и постоянная скорость резания или постоянная скорость шпинделя	+	м		
G952	10	Окружная подача и постоянная скорость резания или постоянная скорость шпинделя	+	м		
G962	11	Линейная подача или окружная подача и постоянная скорость резания	+	м		
G972	12	Линейная подача или окружная подача и постоянная частота вращения шпинделя (постоянная скорость резания ВЫКЛ)	+	м		
G973	13	Окружная подача без ограничения частоты вращения шпинделя и постоянная частота вращения шпинделя (G97 без LIMS для режима ISO)	+	м		

5.3.16 G-группа 16: Процентовка подачи на внутреннем и наружном изгибе

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CFC	1	Постоянная подача на контуре действует на внутреннем и наружном изгибе	+	м	x	
CFTCP	2	Постоянная подача в исходной точке резца инструмента (траектория центра)	+	м		
CFIN	3	Постоянная подача на внутреннем изгибе, ускорение на наружном изгибе	+	м		

5.3.17 G-группа 17: Режим подвода/отвода, коррекция на инструмент

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
NORM	1	Положение нормали в начальной/конечной точке	+	м	x	
KONT	2	Обход контура в начальной/конечной точке	+	м		
KONTT	3	Подвод/отвод с постоянным касанием	+	м		
KONTC	4	Подвод/отвод с постоянным изгибом	+	м		

5.3.18 G-группа 18: Поведение на углах, коррекция на инструмент

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G450	1	Переходная окружность (инструмент обходит углы детали по круговой траектории)	+	м	x	
G451	2	Точка пересечения эквидистант (свободное резание инструмента в углу детали)	+	м		

5.3.19 G-группа 19: Криволинейный переход для начала сплайна

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
BNAT	1	Натуральный криволинейный переход к первому сплайн-кадру	+	м	x	
BTAN	2	Тангенциальный криволинейный переход к первому сплайн-кадру	+	м		
BAUTO	3	Определение первого сегмента сплайна через следующие 3 точки	+	м		

5.3.20 G-группа 20: Криволинейный переход для конца сплайна

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ENAT	1	Натуральный криволинейный переход к следующему кадру перемещения	+	м	x	
ETAN	2	Тангенциальный криволинейный переход к следующему кадру перемещения	+	м		
EAUTO	3	Определение последнего сегмента сплайна через последние 3 точки	+	м		

5.3.21 G-группа 21: Профиль ускорения

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
BRISK	1	Скачкообразное ускорение по траектории	+	м	x	
SOFT	2	Ускорение по траектории с ограничением рывка	+	м		
DRIVE	3	Зависимое от скорости ускорение по траектории	+	м		

5.3.22 G-группа 22: Тип коррекции на инструмент

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CUT2D	1	Коррекция радиуса 2½D	+	м	x	
CUT2DF	2	Коррекция радиуса 2½D относительно текущего фрейма (наклонная плоскость)	+	м		
CUT3DC	3	Коррекция радиуса 3D для периферийного фрезерования	+	м		
CUT3DF	4	Коррекция радиуса 3D для торцового фрезерования с изменением ориентации	+	м		
CUT3DFS	5	Коррекция радиуса 3D для торцового фрезерования с постоянной ориентацией. Ориентация инструмента определена через G17 – G19 и не подвергается воздействию фреймов.	+	м		
CUT3DFF	6	Коррекция радиуса 3D для торцового фрезерования с постоянной ориентацией. Ориентация инструмента — это определенное через G17 – G19 и в некоторых случаях повернутое через фрейм направление.	+	м		
CUT3DCC	7	Коррекция радиуса 3D для периферийного фрезерования с учетом ограничивающей поверхности с коррекцией радиуса 3D: контур на поверхности обработки	+	м		

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CUT3DCCD	8	Коррекция радиуса 3D для периферийного фрезерования с учетом ограничивающей поверхности с дифференциальным инструментом на траектории центра инструмента: подача к ограничивающей поверхности	+	м		
CUT2DD	9	Коррекция радиуса 2½D относительно дифференциального инструмента	+	м		
CUT2DFD	10	Коррекция радиуса 2½D относительно дифференциального инструмента, относительно текущего фрейма (наклонная плоскость)	+	м		
CUT3DCD	11	Коррекция радиуса 3D относительно дифференциального инструмента для периферийного фрезерования	+	м		
CUT3DFD	12	Коррекция радиуса 3D относительно дифференциального инструмента для торцового фрезерования с изменением ориентации	+	м		

5.3.23 G-группа 23: Контроль столкновений на внутренних контурах

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CDOF	1	Контроль столкновения ВЫКЛ	+	м	x	
CDON	2	Контроль столкновения ВКЛ	+	м		
CDOF2	3	Контроль столкновения ВЫКЛ при периферийном фрезеровании 3D	+	м		

5.3.24 G-группа 24: Предуправление

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
FFWOF	1	Предуправление ВЫКЛ	+	м	x	
FFWON	2	Предуправление ВКЛ	+	м		

5.3.25 G-группа 25: Референция ориентации инструмента

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIWKS	1	ориентация инструмента в системе координат детали (WCS)	+	м	x	
ORIMKS	2	ориентация инструмента в системе координат станка (MCS)	+	м		

5.3.26 G-группа 26: Режим повторного подвода для REPOS (действует модально)

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
RMB	1	повторный подвод к начальной точке кадра	-	м		
RMI	2	повторный подвод к точке прерывания	-	м	x	
RME	3	повторный подвод к конечной точке кадра	-	м		
RMN	4	повторный подвод к следующей точке траектории	-	м		

5.3.27 G-группа 27: Коррекция на инструмент при изменении ориентации на наружных углах

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIC	1	изменения ориентации на наружных углах накладываются на вставляемый круговой кадр	+	м	x	
ORID	2.	изменения ориентации выполняются перед круговым кадром	+	м		

5.3.28 G-группа 28: Ограничение рабочего поля

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
WALIMON	1	Ограничение рабочего поля ВКЛ	+	м	x	
WALIMOF	2	Ограничение рабочего поля ВЫКЛ	+	м		

5.3.29 G-группа 29: Программирование радиуса/диаметра

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
DIAMOF	1	Действующее модально, спец. для канала программирование диаметра ВЫКЛ При выключении начинает действовать спец. для канала программирование радиуса.	+	м	x	
DIAMON	2	Действующее модально, независимое спец. для канала программирование диаметра ВКЛ Действие не зависит от запрограммированного режима указания размера (G90/G91).	+	м		
DIAM90	3	Действующее модально, зависимое спец. для канала программирование диаметра ВКЛ Действие зависит от запрограммированного режима указания размера (G90/G91).	+	м		
DIAMCYCOF	4	Действующее модально, спец. для канала программирование диаметра при обработке цикла ВЫКЛ	+	м		

5.3.30 G-группа 30: Сжатие кадров УП

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
COMPOF	1	Компрессор кадров ЧПУ ВЫКЛ	+	м	x	
COMPON	2	Функция компрессора COMPON ВКЛ	+	м		
COMPCURV	3	Функция компрессора COMPCURV ВКЛ	+	м		
COMPCAD	4	Функция компрессора COMPCAD ВКЛ	+	м		
COMPSURF	5	Функция компрессора COMPSURF ВКЛ	+	м		

5.3.31 G-группа 31: OEM-G-команды

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G810	1	OEM-G-команда	-	м		
G811	2	OEM-G-команда	-	м		
G812	3	OEM-G-команда	-	м		
G813	4	OEM-G-команда	-	м		
G814	5	OEM-G-команда	-	м		
G815	6	OEM-G-команда	-	м		
G816	7	OEM-G-команда	-	м		
G817	8	OEM-G-команда	-	м		

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G818	9	OEM-G-команда	-	м		
G819	10	OEM-G-команда	-	м		
Две группы G зарезервированы для пользователя OEM. Тем самым он открывает программирование установленных им функций.						

5.3.32 G-группа 32: OEM-G-команды

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G820	1	OEM - G-команда	-	м		
G821	2	OEM - G-команда	-	м		
G822	3	OEM - G-команда	-	м		
G823	4	OEM - G-команда	-	м		
G824	5	OEM - G-команда	-	м		
G825	6	OEM - G-команда	-	м		
G826	7	OEM - G-команда	-	м		
G827	8	OEM - G-команда	-	м		
G828	9	OEM - G-команда	-	м		
G829	10	OEM - G-команда	-	м		
Две группы G зарезервированы для пользователя OEM. Тем самым он открывает программирование установленных им функций.						

5.3.33 G-группа 33: Настраиваемая точная коррекция на инструмент

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
FTOCOF	1	действующая online точная коррекция инструмента ВЫКЛ	+	м	x	
FTOCON	2	действующая online точная коррекция инструмента ВКЛ	-	м		

5.3.34 G-группа 34: Сглаживание ориентации инструмента

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
OSOF	1	Сглаживание ориентации инструмента ВЫКЛ	+	м	x	
OSC	2	Постоянное сглаживание ориентации инструмента	+	м		

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
OSS	3	Сглаживание ориентации инструмента на конце кадра	+	м		
OSSE	4	Сглаживание ориентации инструмента на начале и конце кадра	+	м		
OSD	5	Перешлифовка внутри кадра с задачей длины хода	+	м		
OST	6	Перешлифовка внутри кадра с задачей углового допуска	+	м		

5.3.35 G-группа 35: Штамповка и вырубка

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
SPOF	1	Ход ВЫКЛ, штамповка и вырубка ВЫКЛ	+	м	x	
SON	2	Вырубка ВКЛ	+	м		
PON	3	Штамповка ВКЛ	+	м		
SONS	4	Вырубка ВКЛ в такте IPO	-	м		
PONS	5	Штамповка ВКЛ в такте IPO	-	м		

5.3.36 G-группа 36: Штамповка с задержкой

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
PDELAYON	1	Задержка при штамповке ВКЛ	+	м	x	
PDELAYOF	2	Задержка при штамповке ВЫКЛ	+	м		

5.3.37 G-группа 37: Профиль подачи

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
FNORM	1	обычная подача по DIN66025	+	м	x	
FLIN	2	подача изменяется линейно	+	м		
FCUB	3	подача изменяется по кубическому сплайну	+	м		

5.3.38 G-группа 38: Назначение быстрых входов/выходов для штамповки/вырубки

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
SPIF1	1	Быстрые входы/выходы ЧПУ для штамповки/вырубки Байт 1	+	м	x	
SPIF2	2	Быстрые входы/выходы ЧПУ для штамповки/вырубки Байт 2	+	м		

5.3.39 G-группа 39: Программируемая точность контура

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CPRECOF	1	Программируемая точность контура ВЫКЛ	+	м	x	
CPRECON	2	Программируемая точность контура ВКЛ	+	м		

5.3.40 G-группа 40: Постоянная коррекция на радиус инструмента

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CUTCONOF	1	Постоянная коррекция радиуса инструмента ВЫКЛ	+	м	x	
CUTCONON	2	Постоянная коррекция радиуса инструмента ВКЛ	+	м		

5.3.41 G-группа 41: Резьбонарезание с возможностью прерывания

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
LFOF	1	Прерываемое резьбонарезание ВЫКЛ	+	м	x	
LFON	2	Прерываемое резьбонарезание ВКЛ	+	м		

5.3.42 G-группа 42: Инструментальный суппорт

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
TCOABS	1	Определение компонентов длин инструмента из текущей ориентации инструмента	+	м	x	
TCOFR	2	Определить компоненты длин инструмента из ориентации активного фрейма	+	м		
TCOFRZ	3	Определение ориентации инструмента активного фрейма при выборе инструмента, инструмент показывает в направлении Z	+	м		
TCOFRY	4	Определение ориентации инструмента активного фрейма при выборе инструмента, инструмент показывает в направлении Y	+	м		
TCOFRX	5	Определение ориентации инструмента активного фрейма при выборе инструмента, инструмент показывает в направлении X		м		

5.3.43 G-группа 43: Направление подвода SAR

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G140	1	Направление подвода SAR установлено через G41/G42	+	м	x	
G141	2	Направление подвода SAR слева от контура	+	м		
G142	3	Направление подвода SAR справа от контура	+	м		
G143	4	Направление подвода SAR зависимое от касательных	+	м		

5.3.44 G-группа 44: Разделение пути SAR

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G340	1	Пространственный кадр подвода, т. е. подача на глубину и подвод в плоскости в одном кадре	+	м	x	
G341	2	Сначала подача в вертикальной оси (Z), потом подвод в плоскости	+	м		

5.3.45 G-группа 45: Соотношение траекторий осей FGROUP

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
SPATH	1	Соотношение траекторий для осей FGROUP это длина дуги	+	м	x	
UPATH	2	Соотношением траекторий для осей FGROUP является параметр кривой	+	м		

5.3.46 G-группа 46: Выбор плоскости для быстрого отвода

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
LFTXT	1	Плоскость определяется из касательной к траектории и актуальной ориентации инструмента	+	м	x	
LFWP	2	Плоскость определяется через актуальную рабочую плоскость (G17/G18/G19)	+	м		
LFPOS	3	Осевой отвод на позицию	+	м		

5.3.47 G-группа 47: Переключение режима для внешнего кода УП

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G290	1	Активировать языковой режим SINUMERIK	+	м	x	
G291	2	Активировать языковой режим ISO	+	м		

5.3.48 G-группа 48: Режим подвода/отвода при коррекции на радиус инструмента

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G460	1	Контроль столкновения для кадра подвода и отвода ВКЛ	+	м	x	
G461	2	Продолжение граничного кадра дугой окружности, если нет точки пересечения в кадре КРИ	+	м		
G462	3	Продолжение граничного кадра прямой, если нет точки пересечения в кадре КРИ	+	м		

5.3.49 G-группа 49: Позиционное движение

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CP	1	Движение по траектории	+	м	x	
PTP	2	Позиционное движение (движение синхронной оси)	+	м		
PTPG0	3	Позиционное движение только для G0, в остальных случаях CP	+	м		
PTPWOC	4	Позиционное движение без компенсирующих движений, обусловленных изменением ориентации	+	м		

5.3.50 G-группа 50: Программирование ориентации

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIEULER	1	Угол ориентации через эйлеров угол	+	м	x	
ORIRPY	2	Угол ориентации через угол RPY (последовательность вращения XYZ)	+	м		
ORIVIRT1	3	Угол ориентации через виртуальные оси ориентации (определение 1)	+	м		
ORIVIRT2	4	Угол ориентации через виртуальные оси ориентации (определение 2)	+	м		
ORIAXPOS	5	Угол ориентации через виртуальные оси ориентации с позициями круговых осей	+	м		
ORIRPY2	6	Угол ориентации через угол RPY (последовательность вращения ZYX)	+	м		

5.3.51 G-группа 51: Тип интерполяции программирования ориентации

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIVECT	1	Большая круговая интерполяция (идентично ORIPLANE)	+	м	x	
ORIAXES	2	Линейная интерполяция осей станка или осей ориентации	+	м		
ORIPATH	3	Путь ориентации инструмента относительно траектории	+	м		
ORIPLANE	4	Интерполяция в плоскости (идентична ORIVECT)	+	м		
ORICONCW	5	Интерполяция на боковой поверхности конуса по часовой стрелке	+	м		
ORICONCCW	6	Интерполяция на боковой поверхности конуса против часовой стрелки	+	м		

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORICONIO	7	Интерполяция на боковой поверхности конуса с указанием промежуточной ориентации	+	м		
ORICONTO	8	Интерполяция на боковой поверхности окружности с тангенциальным переходом	+	м		
ORICURVE	9	Интерполяция с дополнительной пространственной кривой для ориентации	+	м		
ORIPATHS	10	Ориентация инструмента относительно траектории, изгиб в характеристике ориентации сглаживается	+	м		

5.3.52 G-группа 52: Вращения фрейма по отношению к детали

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
PAROTOF	1	Относящееся к детали вращение фрейма ВЫКЛ	+	м	x	
PAROT	2	Относящееся к детали вращение фрейма ВКЛ Система координат детали точно устанавливается на детали.	+	м		

5.3.53 G-группа 53: Вращения фрейма по отношению к инструменту

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
TOROTOF	1	Относящееся к инструменту вращение фрейма ВЫКЛ	+	м	x	
TOROT	2	Точно установить ось Z WCS через вращение фрейма параллельно ориентации инструмента	+	м		
TOROTZ	3	как TOROT	+	м		
TOROTY	4	Точно установить ось Y WCS через вращение фрейма параллельно ориентации инструмента	+	м		
TOROTX	5	Точно установить ось X WCS через вращение фрейма параллельно ориентации инструмента	+	м		
TOFRAME	6	Точно установить ось Z WCS через вращение фрейма параллельно ориентации инструмента	+	м		
TOFRAMEZ	7	как TOFRAME	+	м		
TOFRAMEY	8	Точно установить ось Y WCS через вращение фрейма параллельно ориентации инструмента	+	м		
TOFRAMEX	9	Точно установить ось X WCS через вращение фрейма параллельно ориентации инструмента	+	м		

5.3.54 G-группа 54: Вращение вектора при программировании полинома

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIROTA	1	Абсолютный поворот вектора	+	м	x	
ORIROTR	2	Относительный поворот вектора	+	м		
ORIROTT	3	Тангенциальный поворот вектора	+	м		
ORIROTC	4	Тангенциальный вектор вращения к касательной к траектории	+	м		

5.3.55 G-группа 55: Движение ускоренного хода с/без линейной интерполяции

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
RTLION	1	Движение ускоренного хода с линейной интерполяцией ВКЛ	+	м	x	
RTLIOF	2	Движение ускоренного хода с линейной интерполяцией ВЫКЛ Движение ускоренного хода выполняется с интерполяцией отдельной оси.	+	м		

5.3.56 G-группа 56: Учет износа инструмента

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
TOWSTD	1	Значение по умолчанию для коррекции по длине инструмента	+	м	x	
TOWMCS	2	Значения износа в системе координат станка (MCS)	+	м		
TOWWCS	3	Значения износа в системе координат детали (WCS)	+	м		
TOWBCS	4	Значения износа в базовой кинематической системе (BKS)	+	м		
TOWTCS	5	Значения износа в системе координат инструмента (исходная точка инструментального суппорта T на зажиме инструмента)	+	м		
TOWKCS	6	Значения износа в системе координат инструментальной головки при кинематической трансформации (отличается от MCS вращением инструмента)	+	м		

5.3.57 G-группа 57: Замедление на углах

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
FENDNORM	1	Замедление на углах ВЫКЛ	+	м	x	
G62	2	Замедление на внутренних углах при активной коррекции радиуса инструмента (G41/ G42)	+	м		
G621	3	Замедление на всех углах	+	м		

5.3.58 G-группа 59: Динамический режим для интерполяции траектории

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
DYNNORM	1	Обычная динамика	+	м	x	
DYNPOS	2	Режим позиционирования, нарезание внутренней резьбы	+	м		
DYNROUGH	3	Черновая обработка	+	м		
DYNSEMIFIN	4	Предварительная чистовая обработка	+	м		
DYNFINISH	5	Чистовая обработка	+	м		
DYNPREC	6	Точная чистовая обработка	+	м		

5.3.59 G-группа 60: Ограничение рабочего поля

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
WALCS0	1	Ограничение рабочего поля WCS ВЫКЛ	+	м	x	
WALCS1	2	Группа ограничений рабочего поля WCS 1 активна	+	м		
WALCS2	3	Группа ограничений рабочего поля WCS 2 активна	+	м		
WALCS3	4	Группа ограничений рабочего поля WCS 3 активна	+	м		
WALCS4	5	Группа ограничений рабочего поля WCS 4 активна	+	м		
WALCS5	6	Группа ограничений рабочего поля WCS 5 активна	+	м		
WALCS6	7	Группа ограничений рабочего поля WCS 6 активна	+	м		
WALCS7	8	Группа ограничений рабочего поля WCS 7 активна	+	м		
WALCS8	9	Группа ограничений рабочего поля WCS 8 активна	+	м		
WALCS9	10	Группа ограничений рабочего поля WCS 9 активна	+	м		
WALCS10	11	Группа ограничений рабочего поля WCS 10 активна	+	м		

5.3.60 G-группа 61: Сглаживание ориентации инструмента

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORISOF	1	Сглаживание ориентации инструмента ВЫКЛ	+	м	x	
ORISON	2	Сглаживание ориентации инструмента ВКЛ	+	м		

5.3.61 G-группа 62: Режим повторного подвода для REPOS (действует не модально)

G-команда	№ [1]	Объяснение	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
RMBBL	1	Повторный подвод к начальной точке кадра	-	п		
RMIBL	2	Повторный подвод к точке прерывания	-	п	x	
RMEBL	3	Повторный подвод к конечной точке кадра	-	п		
RMNBL	4	Повторный подвод к следующей точке траектории	-	п		

5.3.62 G-группа 64: Цикловые фреймы

G-команда	№ [1]	Объяснение **Активный фрейм шлифования в канале $P_GFRAME =**	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
GFRAME[0]	1	Фрейм шлифования сохранения данных $P_GFR[0] (нулевой фрейм)	+	м	x	
GFRAME[1]	2	Фрейм шлифования сохранения данных $P_GFR[1]	+	м		
GFRAME[2]	3	Фрейм шлифования сохранения данных $P_GFR[2]	+	м		
...	...		+	м		
GFRAME[100]	101	Фрейм шлифования сохранения данных $P_GFR[100]	+	м		

Легенда

1) Внутренний номер (например, для интерфейса PLC)

2) Возможность конфигурирования G-команды как положения сброса группы G при запуске, Reset или завершении программы обработки детали (с MD20150 $MC_GCODE_RESET_VALUES):

+ конфигурируемая

- не конфигурируемая

3) Активность G-команды:

м модально (с выходом за границы кадров)

п покадрово

4) Положение сброса, см. следующие машинные данные:

- MD20149 $MC_GCODE_RESET_S_VALUES (положение сброса G-групп (fix))
- MD20150 $MC_GCODE_RESET_VALUES (положение сброса G-групп)
- MD20151 $MC_GCODE_RESET_S_MODE (сброс G-групп (fix))
- MD20152 $MC_GCODE_RESET_MODE (сброс G-групп)
- MD20154 $MC_EXTERN_GCODE_RESET_VALUES (положение сброса G-групп в режиме ISO)
- MD20156 $MC_EXTERN_GCODE_RESET_MODE (сброс внешних G-групп)

SAG Стандартная настройка **S**iemens **AG**

MH Стандартная настройка изготовителя станка (**M**aschinen**h**ersteller) (см. указания изготовителя станка)

5.4 Предопределенные процедуры

Вызов предопределенной процедуры запускает выполнение предопределенной функции ЧПУ. Предопределенная процедура, в отличие от предопределенной функции, **не** выводит возвращаемого значения.

Система координат					
Идентификатор	**Параметр**				**Объяснение**
	1.	**2.**	**3. - 15.**	**4. - 16.**	
PRESETON	AXIS *): идентификатор оси ось станка	REAL: Preset-смещение G700/G710 контекст	как 1 ...	как 2 ...	Установка фактического значения для запрограммированных осей с потерей состояния реферирования
PRESETONS	AXIS *): идентификатор оси ось станка	REAL: Preset-смещение G700/G710 контекст	как 1 ...	как 2 ...	Установка фактического значения для запрограммированных осей без потери состояния реферирования

Система координат		
Идентификатор	**Параметр**	**Объяснение**
DRFOF		Удаление смещения DRF для всех согласованных с каналом осей

*) Вместо идентификаторов осей станка могут стоять идентификаторы геометрических или дополнительных осей, если возможно однозначное отображение.

Группы осей					
Идентификатор	**Параметр**				**Объяснение**
GEOAX	**1.**	**2.**	**3. / 5.**	**4. / 6.**	Выбор параллельной системы координат
	INT: номер гео-оси 1 - 3	AXIS: идентификатор оси канала	как 1	как 2	
FGROUP	**1. – 8.**				Переменная исходная точка значения F: определение осей, к которым относится подача по траектории Макс. количество осей: 8 С помощью FGROUP ( ) без указания параметров активируется стандартная настройка для исходной точки значения F.
	AXIS: идентификатор оси канала				
SPLINEPATH	**1.**	**2. - 9.**			Определение соединения сплайнов Макс. количество осей: 8
	INT: соединение сплайнов (должно быть 1)	AXIS: идентификатор гео- или доп. оси			
POLYPATH	**1.**	**2.**			Включение полиномиальной интерполяции для выбранных групп осей
	STRING	STRING			

Совместное движение (буксировка)							
Идентификатор	**Параметр**						**Объяснение**
	1.	**2.**	**3.**	**4-й**	**5.**	**6.**	
TANG	AXIS: имя оси Ведомая ось	AXIS: Ведущая ось 1	AXIS: Ведущая ось 2	REAL: Коэффициент связи	CHAR: Опция: "B": Слежение в BKS "W": Слежение в WCS	CHAR Оптимизация: "S": Стандартный "P": автом. с путем скругления, угловой допуск	Тангенциальное управление: Определить соединение из двух указанных ведущих осей определяется касательная для слежения. Коэффициент связи указывает связь между изменением угла касательной и отслеживаемой осью. Он, как правило, равен 1.
TANGON	AXIS: имя оси Ведомая ось	REAL: Угол смещения	REAL: путь скругления	REAL: угловой допуск			Тангенциальное управление: Включить соединение
TANGOF	AXIS: имя оси Ведомая ось						Тангенциальное управление: Выключение соединения
TLIFT	AXIS: отслеживаемая ось						Тангенциальное управление: Включение создания промежуточного кадра
TRAILON	AXIS: Ведомая ось	AXIS: Ведущая ось	REAL: Коэффициент связи				Включение осевой синхронной буксировки
TRAILOF	AXIS: Ведомая ось	AXIS: Ведущая ось					Выключение осевой синхронной буксировки
TANGDEL	AXIS: Ведомая ось						Тангенциальное управление: Удаление соединения

Таблицы кривых						
Идентификатор	**Параметр**					**Объяснение**
	1.	**2.**	**3.**	**4-й**	**5.**	
CTABDEF	AXIS: ведомая ось	AXIS: ведущая ось	INT: номер таблицы	INT: поведения на границах области определения	STRING: указание места сохранения	Определение таблиц ВКЛ Следующие кадры движения определяют таблицу кривых.
CTABEND	AXIS: ведомая ось	AXIS: ведущая ось	INT: номер таблицы	INT: поведения на границах области определения		Определение таблиц ВЫКЛ
CTABDEL	INT: номер таблицы n	INT: номер таблицы m	STRING: указание места сохранения			Удалить таблицу кривых
CTABLOCK	INT: номер таблицы n					Блокирует таблицу кривых с номером n, т. е. эта таблица не может быть удалена/заменена.
CTABUNLOCK	INT: номер таблицы n					Снова разрешает защищенную с CTABLOCK таблицу с номером n
LEADON	AXIS: ведомая ось	AXIS: ведущая ось	INT: номер таблицы			Соединение по главному значению ВКЛ
LEADOF	AXIS: ведомая ось	AXIS: ведущая ось				Соединение по главному значению ВЫКЛ

Профиль ускорения осевой		
Идентификатор	**Параметр**	**Объяснение**
	1. – 8.	
BRISKA	AXIS	Включение скачкообразного осевого ускорения для запрограммированных осей
SOFTA	AXIS	Включение осевого ускорения с ограничением рывка для запрограммированных осей
DRIVEA	AXIS	Включение ломаной характеристики ускорения для запрограммированных осей
JERKA	AXIS	Установленная через машинные данные $MA_AX_JERK_ENABLE характеристика ускорения действует для запрограммированных осей.

Окружная подача			
Идентификатор	**Параметр**		**Объяснение**
FPRAON	**1.**	**2.**	Осевая окружная подача ВКЛ
	AXIS: ось, для которой включается окружная подача	AXIS: ось/шпиндель, от которой берется окружная подача. Если ось не запрограммирована, то окружная подача берется от мастер-шпинделя.	
FPRAOF	**1. - n.**		Окружная подача осевая ВЫКЛ
	AXIS: оси, для которых выключается окружная подача		Окружная подача может быть отключена одновременно для нескольких осей. Может быть запрограммировано столько осей, сколько допускается на кадр.
FPR	**1.**		Выбор круговой оси/шпинделя, которые являются производными для окружной подачи траектории при G95. Настройка с помощью FPR действует модально.
	AXIS: ось/шпиндель, от которой берется окружная подача. Если ось не запрограммирована, то окружная подача берется от мастер-шпинделя.		

Трансформации				
Идентификатор	**Параметр**			**Объяснение**
	1.	**2.**	**3.**	
TRACYL	REAL: рабочий диаметр	INT: номер трансформации		Цилиндр: трансформация боковой поверхности На канал может быть установлено несколько трансформаций. Номер трансформации показывает, какая трансформация должна быть активирована. Если 2-ой параметр отсутствует, то активируется установленная через машинные данные структура трансформаций.
TRANSMIT	INT: номер трансформации			Transmit: Полярная трансформация На канал может быть установлено несколько трансформаций. Номер трансформации показывает, какая трансформация должна быть активирована. Если параметр отсутствует, то активируется установленная через машинные данные структура трансформаций.

Трансформации				
Идентификатор	**Параметр**			**Объяснение**
	1.	**2.**	**3.**	
TRAANG	REAL: угол	INT: номер трансформации		Трансформация наклонной оси На канал может быть установлено несколько трансформаций. Номер трансформации показывает, какая трансформация должна быть активирована. Если 2-ой параметр отсутствует, то активируется установленная через машинные данные структура трансформаций. Если угол не программируется TRAANG (,2) или TRAANG то последний угол действует модально.
TRAORI	INT: номер трансформации			4-, 5-осевая трансформация На канал может быть установлено несколько трансформаций. Номер трансформации показывает, какая трансформация должна быть активирована.
TRACON	INT: номер трансформации	REAL: прочие параметры зависят от машинных данных.		Каскадированная трансформация Значение параметров зависит от типа каскадирования.
TRAFOOF				Отключить трансформацию
TRAFOON	STRING: имя блока данных трансформации	REAL: базовый или рабочий диаметр (только TRACYL)	BOOL: с коррекцией стенки паза / без коррекции стенки паза (только TRACYL)	Включение трансформаций, определенных посредством кинематических цепей

Шпиндель			
Идентификатор	**Параметр**		**Объяснение**
	1	**2. - n.**	
SPCON	INT: номер шпинделя	INT: номер шпинделя	переключение в режим ориентации шпинделя
SPCOF	INT: номер шпинделя	INT: номер шпинделя	переключение в режим управления числом оборотов шпинделя
SETMS	INT: номер шпинделя		Объявление шпинделя мастер-шпинделем для текущего канала С SETMS( ) без указания параметров начинает действовать предварительная установка через машинные данные.

Шлифование		
Идентификатор	**Параметр**	**Объяснение**
	1.	
GWPSON	INT: номер шпинделя	Постоянная окружная скорость круга ВКЛ Если номер шпинделя не программируется, то для шпинделя активного инструмента выбирается окружная скорость круга.
GWPSOF	INT: номер шпинделя	Постоянная окружная скорость круга ВЫКЛ Если номер шпинделя не программируется, то для шпинделя активного инструмента отключается окружная скорость круга.
TMON	INT: номер T	Спец. для шлифования контроль инструмента ВКЛ Если номер T не программируется, то включается контроль для активного инструмента.
TMOF	INT: номер T	Контроль инструмента ВЫКЛ Если номер T не программируется, то выключается контроль для активного инструмента.

Обработка резаньем					
Идентификатор	**Параметр**				**Объяснение**
	1.	**2.**	**3.**	**4-й**	
CONTPRON	REAL [,11]: таблица контуров	CHAR: Режим обработки	INT: кол-во поднутрений	INT: состояние вычисления	Включение эталонной подготовки Вызываемые в дальнейшем программы контура или кадры ЧПУ разделяются на отдельные движения и сохраняются в таблице контуров. Количество поднутрений возвращается.
CONTDCON	REAL [, 6]: таблица контуров	INT: Направление обработки			Декодирование контура Кадры контура, закодированный строкой таблицы на кадр для экономии памяти, сохраняются в названной таблице.
EXECUTE	INT: состояние ошибки				Включение выполнения программы Таким образом, осуществляется переключение из режима эталонной подготовки или после создания защищенной области к нормальной обработке программы.

Обработка таблицы		
Идентификатор	**Параметр**	**Объяснение**
	1.	
EXECTAB	REAL [11]: элемент из таблицы движений	Обработать элемент из таблицы движений

Защищенные области						
Идентификатор	**Параметр**					**Объяснение**
	1.	**2.**	**3.**	**4-й**	**5.**	
CPROTDEF	INT: номер защищенной области	BOOL: TRUE: Ориентированная на инструмент защищенная область	INT: 0: 4-й и 5-й параметры не обрабатываются 1: 4-й параметр обрабатывается 2: 5-й параметр обрабатывается 3: 4-й и 5-й параметры обрабатываются	REAL: ограничение в плюсовом направлении	REAL: ограничение в минусовом направлении	Определение спец. для канала защищенной области
NPROTDEF	INT: номер защищенной области	BOOL: TRUE: Ориентированная на инструмент защищенная область	INT: 0: 4-й и 5-й параметры не обрабатываются 1: 4-й параметр обрабатывается 2: 5-й параметр обрабатывается 3: 4-й и 5-й параметры обрабатываются	REAL: ограничение в плюсовом направлении	REAL: ограничение в минусовом направлении	Определение спец. для станка защищенной области

Защищенные области						
Идентификатор	**Параметр**					**Объяснение**
	1.	**2.**	**3.**	**4-й**	**5.**	
CPROT	INT: номер защищенной области	INT: опция 0: защищенная область выкл 1: предварительная активация защищенной области 2: защищенная область вкл 3: предварительно активировать защищенную область с условным остановом, только для активных защищенных областей	REAL: смещение защищенной области в первой гео-оси	REAL: смещение защищенной области во второй гео-оси	REAL: смещение защищенной области в третьей гео-оси	Спец. для канала защищенная область ВКЛ/ВЫКЛ
NPROT	INT: номер защищенной области	INT: опция 0: защищенная область выкл 1: предварительная активация защищенной области 2: защищенная область вкл 3: предварительно активировать защищенную область с условным остановом, только для активных защищенных областей	REAL: смещение защищенной области в первой гео-оси	REAL: смещение защищенной области во второй гео-оси	REAL: смещение защищенной области в третьей гео-оси	Спец. для канала защищенная область ВКЛ/ВЫКЛ

Предварительная обработка / отдельный кадр		
Идентификатор	**Параметр**	**Объяснение**
STOPRE		Остановка предварительной обработки до выполнения всех подготовленных кадров главного хода
SBLOF		Подавить покадровую обработку
SBLON		Снять подавление покадровой обработки

Прерывания		
Идентификатор	**Параметр**	**Объяснение**
	1.	
Disable	INT: номер входа прерываний	Обработчик прерываний, согласованный с указанным аппаратным входом, деактивируется. Быстрый отвод также не осуществляется. Установленное с помощью SETINT согласование между аппаратным входом и обработчиком прерываний сохраняется и может снова быть активировано с помощью ENABLE.
Enable	INT: номер входа прерываний	Реактивация деактивированного с DISABLE согласования обработчика прерываний.
CLRINT	INT: номер входа прерываний	Удаление согласования обработчиков прерываний и атрибутов с входом прерываний. Таким образом, обработчик прерываний выключается. При возникновении прерывания реакция не следует.

Синхронные действия		
Идентификатор	**Параметр**	**Объяснение**
	1. - n.	
CANCEL	INT: номер синхронного действия	Отмена модального синхронного действия с указанным ID. Может быть указано и несколько ID, разделенных запятой.
CANCELSUB		Отменить актуальный уровень подпрограммы

Определение функций					
Идентификатор	**Параметр**				**Объяснение**
	1.	**2.**	**3.**	**4.-7.**	
FCTDEF	INT: номер функции	REAL: нижнее предельное значение	REAL: верхнее предельное значение	REAL: коэффициенты a0-a3	Определение функции полинома Она обрабатывается в SYNFCT или PUTFTOCF.

Коммуникация						
Идентификатор	**Параметр**					**Объяснение**
	1.	**2.**				
MMC	STRING: команда	CHAR: Режим квитирования*) "N": без квитирования "S": синхронное квитирование "A": асинхронное квитирование				Команда на интерпретатор команд HMI для конфигурирования окон через программу ЧПУ
	1.	**2.**	**3.**	**4.**	**5.**	
DRVPRD	VAR INT: состояние	AXIS: имя оси станка	INT: номер параметра привода	INT: индекс параметра привода	VAR REAL: считанное значение параметра привода	Чтение параметров привода на уровне программы обработки детали
DRVPWR	VAR INT: состояние	AXIS: имя оси станка	INT: номер параметра привода	INT: индекс параметра привода	REAL: записываемое значение параметра привода	Запись параметров привода на уровне программы обработки детали

*) Команды квитируются по требованию исполняющего компонента (канал, ЧПУ ...).

Координация программ					
Идентификатор	**Параметр**				**Объяснение**
INIT	**1.**	**2.**	**3.**		Выбор программы ЧПУ для выполнения в канале
	INT: номер канала **или** имя канала из MD20000*)	STRING: указание пути	CHAR: режим квитирования**)		
	1. - n.				

Координация программ			
Идентификатор	**Параметр**		**Объяснение**
START	INT: номер канала **или** имя канала из MD20000*)		Запуск выбранных программ в нескольких каналах одновременно из текущей программы Эта команда не действует в собственном канале.
WAITE	INT: номер канала **или** имя канала из MD20000*)		Ожидание завершения программы в одном или нескольких других каналах
	1.	**2. - n.**	
WAITM	INT: номер метки	INT: номер канала **или** имя канала из MD20000*)	Ожидание достижения метки в указанных каналах Предшествующий кадр завершается с точным остановом.
WAITMC	INT: номер метки	INT: номер канала **или** имя канала из MD20000*)	Ожидание достижения метки в указанных каналах Точный останов запускается только в том случае, если другие каналы еще не достигли метки.
	1. - n.		
SETM	INT: номер метки		Установка одной или нескольких меток для координации каналов Это не затрагивает обработку в собственном канале.
CLEARM	INT: номер метки		Удаление одной или нескольких меток для координации каналов Это не затрагивает обработку в собственном канале.
	1. - n.		
WAITP	AXIS: идентификатор оси		Ожидание, пока указанные позиционирующие оси, прежде запрограммированные с POSA, достигнут своей запрограммированной конечной точки
WAITS	INT: номер шпинделя		Ожидание, пока указанные шпиндели, прежде запрограммированные со SPOSA, достигнут своей запрограммированной конечной точки

Координация программ					
Идентификатор	**Параметр**				**Объяснение**
RET	**1.**	**2.**	**3.**	**4.**	Конец подпрограммы без вывода функции на PLC При указании 1-го параметра (цель перехода) возврат осуществляется сначала на кадр после кадра вызова. Затем в зависимости от программирования (RET или RETB) поиск цели выполняется по следующей стратегии: • RET: Поиск в направлении конца программы. Если этот поиск не дал результата, далее выполняется поиск в направлении начала программы. • RETB:: Поиск в направлении начала программы. Если этот поиск не дал результата, далее выполняется поиск в направлении конца программы.
	INT (или STRING): цель перехода (кадр №. / метка) для возврата	INT: 0: возврат на цель перехода из 1-го Пар. > 0: возврат на следующий кадр	INT: число пропускаемых уровней подпрограмм	BOOL: возврат на первый кадр в главной программе	
RETB	INT (или STRING): цель перехода (кадр №. / метка) для возврата	INT: 0: возврат на цель перехода из 1-го Пар. > 0: возврат на следующий кадр	INT: число пропускаемых уровней подпрограмм	BOOL: возврат на первый кадр в главной программе	
	1. - n.				
GET	AXIS: идентификатор оси ***)				Назначение оси(ей) станка Указанные оси с помощью RELEASE должны быть разрешены в другом канале.
GETD	AXIS: идентификатор оси ***)				Прямое назначение оси(ей) станка Указанные оси **не** должны быть разрешены с RELEASE.
RELEASE	AXIS: идентификатор оси ***)				Разрешение оси(ей) станка
	1.	**2.**	**3.**	**4.**	
PUTFTOC	REAL: поправка	INT: номер параметра	INT: номер канала **или** имя канала из MD20000*)	INT: Номер шпинделя	Изменение точной коррекции инструмента
PUTFTOCF	INT: № функции	VAR REAL: Исходное значение	INT: номер параметра	INT: номер канала **или** имя канала из MD20000*)	Изменение точной коррекции инструмента в зависимости от установленной с помощью FCTDEF функции (полином макс. 3-го порядка) При FCTDEF указать используемый здесь №.

Координация программ					
Идентификатор	**Параметр**				**Объяснение**
AXTOCHAN	**1.**	**2.**	**3. - n.**	**4. - m.**	Передать оси в другие каналы
	AXIS: идентификатор оси	INT: номер канала **или** имя канала из MD20000*)	как 1 ...	как 2 ...	

*) Вместо номеров каналов могут программироваться и определенные через MD20000 $MC_CHAN_NAME имена каналов.

**) Команды квитируются по требованию исполняющего компонента (канал, ЧПУ ...).

***) Вместо оси с помощью функции SPI может быть соответственно запрограммирован шпиндель: например, GET(SPI(1))

Доступ к данным		
Идентификатор	**Параметр**	**Объяснение**
CHANDATA	**1.** INT: номер канала	Настройка номера канала для доступа к данным канала (допускается только в блоке инициализации). Последующие обращения относятся к установленному с помощью CHANDATA каналу.
NEWCONF		Применить измененные машинные данные

Сообщения			
Идентификатор	**Параметр**		**Объяснение**
	1.	**2.**	
MSG	STRING: сообщение	INT: исполнение	Вывести любую строку символов как сообщение на интерфейс пользователя
WRTPR	STRING: строка символов	INT: исполнение	Запись строки символов в переменную BTSS

Доступ к файлам						
Идентификатор	**Параметр**					**Объяснение**
READ	**1.**	**2.**	**3.**	**4.**	**5.**	Чтение файлов из файловой системы
	VAR INT: ошибка	CHAR[160]: имя файла	INT: начальная строка читаемой области файла	INT: число читаемых строк	VAR CHAR[255]: поле переменной, в которое сохраняется считанная информация	
write	**1.**	**2.**	**3.**	**4.**		Записать кадр в файловую систему (или на внешнее устройство/файл)
	VAR INT: ошибка	CHAR[160]: имя файла	STRING: устройство/файл для внешнего вывода	CHAR[200]: кадр		
DELETE	**1.**	**2.**				Удалить файл
	VAR INT: ошибка	CHAR[160]: имя файла				

Аварийные сообщения			
Идентификатор	**Параметр**		**Объяснение**
	1.	**2.**	
SETAL	INT: номер аварийного сообщения (аварийные сообщения циклов)	STRING: строка символов	Установить аварийное сообщение К номеру аварийного сообщения дополнительно может быть указана строка символов с макс. 4 параметрами. Доступны следующие предопределенные параметры: %1 = номер канала %2 = номер кадра, метка %3 = индекс текста для аварийных сообщений циклов %4 = дополнительный параметр аварийного сообщения

Управление инструментом				
Идентификатор	**Параметр**			**Объяснение**
	1.	**2.**		

Управление инструментом							
Идентификатор	**Параметр**						**Объяснение**
DELDL	INT: T-№	INT: D-№					Удаление всех суммарных коррекций резца (или инструмента, если D не указывается)
DELT	STRING [32]: идентификатор инструмента	INT: № гнезда					Удалить инструмент Номер гнезда может не указываться.
DELTC	INT: № блока данных n	INT: № блока данных m					Удалить номера блоков данных инструмента n до m
DZERO							Установка недействительными D-№ всех инструментов согласованного с каналом блока TO
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
GETFREELOC	VAR INT: № магазина (Возвращаемое значение)	VAR INT: № места (Возвращаемое значение)	INT: T-№	INT: № реф. магазина	CHAR: данные зависят от 4-го параметра	INT: режим резервирования	Поиска свободного места для инструмента
	1.	**2.**					
GETSELT	VAR INT: T-№ (Возвращаемое значение)	INT: № шпинделя					Выводит T-номер предварительно выбранного для шпинделя инструмента
GETEXET	VAR INT: T-№ (Возвращаемое значение)	INT: № шпинделя					Выводит T-номер активного с точки зрения программы ЧПУ инструмента
GETTENV	STRING: имя инструментального окружения	INT ARRAY[3]: возвращаемые значения					Считывает сохраненные в инструментальном окружении T-, D- и DL-номера
	1.	**2.**	**3.**	**4.**			

Управление инструментом						
Идентификатор	**Параметр**					**Объяснение**
POSM	INT: № места, на которое должно быть выполнено позиционирование	INT: № магазина, который должен двигаться	INT: № места внутреннего магазина	INT: № магазина внутреннего магазина		Позиционирование магазина
RESETMON	VAR INT: состояние = результат операции (возвращаемое значение)	INT: внутренний T-№	INT: D-№ инструмента	INT: опциональный параметр в битовой кодировке		Установка фактического значения инструмента на заданное значение
SETDNO	**1.**	**2.**	**3.**			Установить номер коррекции (D) резца инструмента (T)
	INT: T-№	INT: № резца	INT: D-№			
SETMTH	**1.**					Установка Nr. инструментального суппорта
	INT: № зажима инструмента					
SETPIECE	**1.**	**2.**				Уменьшение счетчика деталей шпинделя Здесь пользователь может обновить данные контроля числа изделий участвующих в процессе обработки инструментов.
	INT: значение, на которое происходит уменьшение	INT: № шпинделя				
	1.	**2.**	**3.**	**4.**		
SETTA	VAR INT: состояние = результат операции (возвращаемое значение)	INT: Магазин №	INT: № структуры износа	INT: подгруппа инструмента		Активация инструмента из структуры износа
SETTIA	VAR INT: состояние = результат операции (возвращаемое значение)	INT: Магазин №	INT: № структуры износа	INT: подгруппа инструмента		Деактивация инструмента из структуры износа

Управление инструментом							
Идентификатор	**Параметр**						**Объяснение**
TCA	**1.**	**2.**	**3.**				Выбор/смена инструмента независимо от состояния инструмента
	STRING[32]: идентификатор инструмента	INT: № гнезда	INT: № зажима инструмента				
TCI	**1.**	**2.**					Переместить инструмент из буфера в магазин
	INT: № буфера	INT: № зажима инструмента					
MVTOOL	**1.**	**2.**	**3.**	**4.**	**5.**		Языковая команда для движения инструмента
	INT: состояние	INT: Магазин №	INT: № места	INT: магазин № после движения	INT: место назначения № после движения		

Ориентация инструмента				
Идентификатор	**Параметр**			**Объяснение**
	1.	**2.**	**3.**	
ORIRESET	REAL: исходное положение 1-й гео-оси	REAL: исходное положение 2-й гео-оси	REAL: исходное положение 3-й гео-оси	Исходное положение ориентации инструмента

Синхронный шпиндель							
Идентификатор	**Параметр**						**Объяснение**
	1.	**2.**	**3.**	**4-й**	**5.**	**6.**	
COUPDEF	AXIS: ведомый шпиндель	AXIS: ведущий шпиндель	REAL: числитель передаточного числа	REAL: знаменатель передаточного числа	STRING[8]: режим смены кадра	STRING[2]: тип соединения	Определить структуру синхронных шпинделей
COUPDEL	AXIS: ведомый шпиндель	AXIS: ведущий шпиндель					Удалить структуру синхронных шпинделей

Синхронный шпиндель							
Идентификатор	**Параметр**						**Объяснение**
	1.	**2.**	**3.**	**4-й**	**5.**	**6.**	
COUPRES	AXIS: ведомый шпиндель	AXIS: ведущий шпиндель					Сбросить параметры соединения на сконфигурированные значения MD и SD
COUPON	AXIS: ведомый шпиндель	AXIS: ведущий шпиндель	REAL: позиция включения ведомого шпинделя				Включить соединение синхронных шпинделей Если для ведомого шпинделя указывается позиция включения (угловое смещение между FS и LS, относящееся абсолютно или инкрементально к позиции нуля градусов ведущего шпинделя в положительном направлении вращения), то соединение включается только после перехода через указанную позицию.
COUPONC	AXIS: ведомый шпиндель	AXIS: ведущий шпиндель					Включить соединение синхронных шпинделей С COUPONC при включении соединения применяется текущая действующая скорость ведомого шпинделя (M3/M4 S...).
COUPOF	AXIS: ведомый шпиндель	AXIS: ведущий шпиндель	REAL: позиция выключения ведомого шпинделя (абсолютная)	REAL: позиция выключения ведущего шпинделя (абсолютная)			Выключить соединение синхронных шпинделей Если позиции указываются, то связь разрывается только после того, как пройдены все указанные позиции. Ведомый шпиндель продолжает вращаться с последним числом оборотов перед выключением соединения.

Синхронный шпиндель							
Идентификатор	**Параметр**						**Объяснение**
	1.	**2.**	**3.**	**4-й**	**5.**	**6.**	
COUPOFS	AXIS: ведомый шпиндель	AXIS: ведущий шпиндель	REAL: позиция выключения ведомого шпинделя (абсолютная)				Отключение соединения синхронных шпинделей с остановкой ведомого шпинделя Если позиция указываются, то связь разрывается только после того, как указанная позиция пройдена
WAITC	AXIS: ведомый шпиндель	STRING [8]: режим смены кадра	AXIS: ведомый шпиндель	STRING[8]: режим смены кадра			Ожидать выполнения критерия смены кадров соединения для шпинделей (макс. 2) Если параметры смены кадров не указываются, то действуют указанные при определении с COUPDEF параметры смены кадра.

Электронный редуктор					
Идентификатор	**Параметр**				**Объяснение**
EGDEL	**1.** AXIS: ведомая ось				Удалить определение соединения для ведомой оси
EGDEF	**1.** AXIS: ведомая ось	**2./4./6./8./10.** AXIS: ведущая ось	**3./5./7./9./11.** INT: тип соединения		Определение электронного редуктора

Электронный редуктор									
Идентификатор	**Параметр**								**Объяснение**
EGON	**1.**	**2.**	**3. / 6. / 9. / 12. / 15.**	**4. / 7. / 10. / 13. / 16.**	**5. / 8. / 11. / 14. / 17.**				Электронный редуктор ВКЛ без синхронизации
	AXIS: ведомая ось	STRING: режим смены кадра	AXIS: ведущая ось	REAL: числитель коэффициента связи	REAL: знаменатель коэффициента связи				
EGONSYN	**1.**	**2.**	**3.**	**4. / 8. / 12. / 16. / 20.**	**5. / 9. / 13. / 17. / 21.**	**6. / 10. / 14. / 18. / 22.**	**7. / 11. / 15. / 19. / 23.**		Электронный редуктор ВКЛ с синхронизацией
	AXIS: ведомая ось	STRING: режим смены кадра	REAL: синхронная позиция ведомой оси	AXIS: ведущая ось	REAL: синхронная позиция ведущей оси	REAL: числитель коэффициента связи	REAL: знаменатель коэффициента связи		
EGONSYNE	**1.**	**2.**	**3.**	**4.**	**5. / 9. / 13. / 17. / 21.**	**6. / 10. / 14. / 18. / 22.**	**7. / 11. / 15. / 19. / 23.**	**8. / 12. / 16. / 20. / 24.**	Электронный редуктор ВКЛ с синхронизацией и и установка режима подвода
	AXIS: ведомая ось	STRING: режим смены кадра	REAL: синхронная позиция ведомой оси	STRING: режим подвода	AXIS: ведущая ось	REAL: синхронная позиция ведущей оси	REAL: числитель коэффициента связи	REAL: знаменатель коэффициента связи	
EGOFS	**1.**	**2. - n.**							Селективное выключение электронного редуктора
	AXIS: ведомая ось	AXIS: ведущая ось							
EGOFC	**1.**								Выключение электронного редуктора (вариант только для шпинделей)
	AXIS: ведомый шпиндель								

Вырубка					
Идентификатор	**Параметр**				**Объяснение**
	1.	**2.**	**3.**	**4-й**	
PUNCHAAC	REAL: мин. расстояние между отверстиями	REAL: начальное ускорение	REAL: макс. расстояние между отверстиями	REAL: конечное ускорение	Активировать зависимое от хода ускорение

Информационные функции в пассивной файловой системе				
Идентификатор	**Параметр**			**Объяснение**
	1.	**2.**	**3.**	
FILEDATE	VAR INT: сообщение об ошибке	CHAR[160]: имя файла	VAR CHAR[8]: дата в формате "дд.мм.гг"	Выводит дату последнего обращения к файлу по записи
FILETIME	VAR INT: сообщение об ошибке	CHAR[160]: имя файла	VAR CHAR[8]: время в формате "чч.мм.сс"	Выводит время последнего обращения к файлу по записи
FILESIZE	VAR INT: сообщение об ошибке	CHAR[160]: имя файла	VAR INT: размер файла	Выводит текущий размер файла
FILESTAT	VAR INT: сообщение об ошибке	CHAR[160]: имя файла	VAR CHAR[5]: данные в формате "rwxsd"	Выводит состояние файла касательно следующих прав: • Чтение (r: read) • Запись (w: write) • Исполнение (x: execute) • Индикация (s: show) • Удаление (d: delete)
FILEINFO	VAR INT: сообщение об ошибке	CHAR[160]: имя файла	VAR CHAR[32]: данные в формате "rwxsd nnnnnnnn дд.мм.гг чч:мм:сс"	Выводит для файла суммарную информацию, которая может быть выгружена через FILEDATE, FILETIME, FILESIZE и FILESTAT

Осевой контейнер		
Идентификатор	**Параметр**	**Объяснение**
	1. - n.	
AXCTSWE	AXIS: осевой контейнер	Вращать осевой контейнер
AXCTSWED	AXIS: осевой контейнер	Вращать осевой контейнер (командная переменная для ввода в эксплуатацию!)
AXCTSWEC:	AXIS: осевой контейнер	Отмена разрешения на вращение осевого контейнера

Master/Slave-соединение		
Идентификатор	**Параметр**	**Объяснение**
	1. - n.	
MASLON	AXIS: идентификатор оси	Включить Master/Slave-соединение
MASLOF	AXIS: идентификатор оси	Разорвать Master/Slave-соединение
MASLOFS	AXIS: идентификатор оси	Разорвать Master/Slave-соединение и автоматически остановить Slave-шпиндели
MASLDEF	AXIS: идентификатор оси	Определить Master/Slave-соединение Последняя ось это Master-ось.
MASLDEL	AXIS: идентификатор оси	Разорвать соединение Master/Slave и удалить определение структуры

Коррекция длин инструмента Online			
Идентификатор	**Параметр**		**Объяснение**
	1.	**2.**	
TOFFON	AXIS: направление коррекции	REAL: значение смещения в направлении коррекции	Активировать коррекцию длин инструмента Online в указанном направлении коррекции
TOFFOF	AXIS: направление коррекции		Сбросить коррекцию длин инструмента Online в указанном направлении коррекции

SERUPRO		
Идентификатор	**Параметр**	**Объяснение**
IPTRLOCK		Начало непригодного для поиска сегмента программы
IPTRUNLOCK		Конец непригодного для поиска сегмента программы

Отвод				
Идентификатор	**Параметр**			**Объяснение**
	1. - n.			
POLFMASK	AXIS: имя гео- или оси станка			Разрешить оси для быстрого отвода (без связи между осями)
POLFMLIN	AXIS: имя гео- или оси станка			Разрешить оси для линейного быстрого отвода
POLFA	**1.**	**2.**	**3.**	Позиция отвода для отдельных осей
	AXIS: идентификатор оси канала	INT: тип	REAL: значение	

Предотвращение столкновений			
Идентификатор	**Параметр**		**Объяснение**
	1.		
PROTA	STRING: "R"		Запрос повторного расчета модели столкновений
PROTS	**1.**	**2. - n.**	Установка состояния защищенной области
	CHAR: состояние	STRING: имя защищенной области	

Интеллектуальная коррекция нагрузки					
Идентификатор	**Параметр**				**Объяснение**
CADAPTON	**1.**	**2.**	**3.**	**4.**	Включить коррекцию нагрузки
	INT: состояние	AXIS: имя оси станка	INT: входная величина	REAL: входное значение (опция)	
CADAPTOF	**1.**	**2.**	**3.**		Выключить коррекцию нагрузки
	INT: состояние	AXIS: имя оси станка	INT: входная величина		

Адаптация рывка		
Идентификатор	**Параметр**	**Объяснение**
CALCFIR		Корректирование FIR-фильтра по динамическому режиму CALCFIR используется в цикле изготовителя CUST_832.
AFISON		Включить автоматическое переключение цепочек фильтров
AFISOF		Выключить автоматическое переключение цепочек фильтров

5.5 Предопределенные процедуры в синхронных действиях

Следующие предопределенные процедуры доступны только в синхронных действиях?

Синхронные процедуры		
Идентификатор	**Параметр**	**Объяснение**
STOPREOF		Отменить остановку предварительной обработки Синхронное действие с командой STOPREOF вызывает остановку предварительной обработки после следующего кадра вывода (= кадр на главный ход). Остановка предварительной обработки отменяется с завершением кадра вывода или при выполнении условия STOPREOF. Все операторы синхронного действия с командой STOPREOF после этого считаются обработанными.
RDISABLE		блокировка загрузки
DELDTG	**1.**	Стирание остатка пути Синхронное действие с командой DELDTG вызывает остановку предварительной обработки после следующего кадра вывода (= кадр на главный ход). Остановка предварительной обработки отменяется с завершением кадра вывода или при выполнении первого условия DELDTG. В $AA_DELT[<ось>] можно найти осевое удаление от заданной точки при осевом стирании остатка пути, в $AC_DELT – остаточный ход траектории.
	AXIS: ось для осевого стирания остатка пути (опция). Если оси нет, то запускается стирание остатка пути для хода траектории.	

Координация программ технологических циклов		
Идентификатор	**Параметр**	**Объяснение**
	1.	

Координация программ технологических циклов		
Идентификатор	**Параметр**	**Объяснение**
LOCK	INT: ID синхронного действия, которое должно быть заблокировано	Блокировать синхронное действие с ID или остановить технологический цикл Может быть запрограммирован один или несколько ID.
UNLOCK	INT: ID синхронного действия, которое должно быть разрешено	Разрешить синхронное действие с ID или продолжить технологический цикл Может быть запрограммирован один или несколько ID.
ICYCON		Выполнить каждый кадр технологического цикла после ICYCON в отдельном такте IPO
ICYCOF		Выполнить все кадры технологического цикла после ICYCOF в одном такте IPO

Полиномиальные функции					
Идентификатор	**Параметр**				**Объяснение**
SYNFCT	**1.**	**2.**	**3.**		Если в синхронном действии движения выполнено условие, то обрабатывается определенный через первое выражение полином на входных переменных. После значение ограничивается вниз и вверх и присваивается входной переменной.
	INT: номер полиномиальной функции, определенной с FCTDEF	VAR REAL: переменная результата *)	VAR REAL: входная переменная **)		
FTOC	**1.**	**2.**	**3.**	**4.** / **5.**	Изменение точной коррекции инструмента в зависимости от определенной с FCTDEF функции (полином макс. 3-о порядка). При FCTDEF необходимо указать используемый здесь номер.
	INT: номер полиномиальной функции, определенной с FCTDEF	VAR REAL: входная переменная **)	INT: длина 1, 2, 3	INT: номер канала / INT: номер шпинделя	

*) В качестве переменной результата разрешены только специальные системные переменные (см. Описание функций "Синхронные действия").

**) В качестве входной переменной допустимы только специальные системные переменные (см. Описание функций "Синхронные действия").

5.6 Предопределенные функции

Вызов предопределенной функции запускает выполнение предопределенной функции ЧПУ, которая, в отличие от предопределенной процедуры, возвращает значение. Вызов предопределенной функции может быть указан как операнд в выражении.

Система координат						
Идентификатор	**Возвращаемое значение**	**Параметр**				**Объяснение**
		1.	**2.**	**3. - 15.**	**4. - 16.**	
CTRANS	FRAME	AXIS: идентификатор оси	REAL: Смещение	AXIS: идентификатор оси	REAL: Смещение	Смещение: смещение нулевой точки ГРУБОЕ для нескольких осей
CFINE	FRAME	AXIS: идентификатор оси	REAL: Смещение	AXIS: идентификатор оси	REAL: Смещение	Смещение: смещение нулевой точки ТОЧНОЕ для нескольких осей
CSCALE	FRAME	AXIS: идентификатор оси	REAL: коэффициент масштабирования	AXIS: идентификатор оси	REAL: коэффициент масштабирования	Scale: Коэффициент масштабирования для нескольких осей
		1.	**2.**	**3-я и 5-я**	**4-я и 6-я**	
CROT	FRAME	AXIS: идентификатор оси	REAL: вращение	AXIS: идентификатор оси	REAL: Вращение	Вращение: вращение текущей системы координат Макс. число параметров: 6 (по одному идентификатору оси и значению на геометрическую ось).
CROTS	FRAME	AXIS: идентификатор оси	REAL: Вращение с пространственным углом	AXIS: идентификатор оси	REAL: Вращение с пространственным углом	Вращение: Вращение текущей системы координат с пространственным углом Макс. число параметров: 6 (по одному идентификатору оси и значению на геометрическую ось).
CMIRROR		**1.**	**2. - 8.**			Mirror: отражение на оси координат
	FRAME	AXIS	AXIS			

Система координат						
Идентификатор	**Возвращаемое значение**	**Параметр**				**Объяснение**
		1.	**2.**			
CRPL	FRAME	INT: ось вращения	REAL: угол поворота			Вращение фрейма в произвольной плоскости
ADDFRAME	INT: 0: OK 1: указание цели (строка) неправильное 2: целевой фрейм не сконфигурирован 3: Вращение во фрейме не разрешено	FRAME: аддитивный измеренный или вычисленный фрейм	STRING: специфицированный целевой фрейм			Вычисляет целевой фрейм, специфицированный строкой Целевой фрейм вычисляется таким образом, что новый общий фрейм получается как соединение старого общего фрейма с переданным фреймом.
INVFRAME	FRAME	**1.** FRAME				Вычисляет из фрейма инверсный фрейм Соединение фрейма с его инверсным фреймом всегда дает нулевой фрейм
MEAFRAME	FRAME	**1.** REAL[3,3]: координаты измеренных точек в пространстве	**2.** REAL[3,3]: координаты заданных точек	**3.** VAR REAL: переменная, возвращающая информацию о качестве вычисления ФРЕЙМА		Вычисление фрейма из 3 точек измерения в пространстве

Геометрические функции					
Идентификатор	**Возвращаемое значение**	**Параметр**			**Объяснение**
		1.	**2.**	**3.**	
CALCDAT	BOOL: состояние ошибки	VAR REAL [n, 2]: Таблица (абсцисса, ордината) точек 1 - n	INT: Количество точек	VAR REAL [3]: результат: Абсцисса, ордината и радиус вычисленного центра круга	Вычисляет координаты центра окружности и радиус из 3 или 4 точек. Точки должны быть различными.
INTERSEC	BOOL: состояние ошибки	VAR REAL [11]: первый элемент контура	VAR REAL [11]: второй элемент контура	VAR REAL [2]: Вектор результата для координат точки пересечения: абсцисса и ордината	Рассчитывает координаты точки пересечения между двумя элементами контура. Статус ошибки указывает на то, была ли найдена точка пересечения.

Функции таблиц кривых								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTAB	REAL: позиция ведомой оси	REAL: позиция ведущей оси	INT: номер таблицы	VAR REAL[ ]: результат подъем	AXIS: ведомая ось для масштабирования	AXIS: ведущая ось для масштабирования		Определяет позицию ведомой оси для указанной позиции ведущей оси по таблице кривых Если параметры 4/5 не запрограммированы, то расчет выполняется со стандартным масштабированием.
CTABINV	REAL: позиция ведущей оси	REAL: позиция ведомой оси	REAL: главная позиция	INT: номер таблицы	VAR REAL[ ]: результат подъем	AXIS: ведомая ось для масштабирования	AXIS: ведущая ось для масштабирования	Определяет позицию ведущей оси для указанной позиции ведомой оси по таблице кривых. Если параметры 5/6 не запрограммированы, то расчет выполняется со стандартным масштабированием.
CTABID	INT: номер таблицы кривых	INT: номер элемента в памяти	STRING: место сохранения: "SRAM", "DRAM"					Определяет номер таблицы кривых, помещенный под указанным номером в память

Функции таблиц кривых								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABISLOCK	INT: состояние блокировки	INT: номер таблицы						Определяет состояние блокировки таблицы кривых: > 0: таблица заблокирована 1: CTABLOCK 2: активное соединение 3: CTABLOCK и активное соединение 0: Таблица не заблокирована -1: Таблица не существует
CTABEXISTS	INT: наличие	INT: номер таблицы						Определяет наличие таблицы кривых в статической или динамической памяти ЧПУ: 0: false 1: TRUE
CTABMEMTYP	INT: место в памяти	INT: номер таблицы						Определяет место таблицы кривых в памяти: 1: DRAM 0: SRAM -1: Таблица не существует
CTABPERIOD	INT: периодичность	INT: номер таблицы						Определяет периодичность таблицы кривых: 0: не периодическая 1: периодическая по ведущей оси 2: периодическая по ведущей и ведомой оси -1: Таблица не существует
CTABNO	INT: количество таблиц кривых							Определяет количество определенных таблиц кривых (в статической и динамической памяти ЧПУ)

Функции таблиц кривых								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABNOMEM	INT: количество таблиц кривых	STRING: место сохранения: "SRAM", "DRAM"						Определяет количество определенных таблиц кривых в указанной памяти
CTABFNO	INT: число таблиц	STRING: место сохранения: "SRAM", "DRAM"						Определяет количество таблиц кривых, которое можно добавить в указанную память
CTABSEG	INT: число сегментов кривой	STRING: место сохранения: "SRAM", "DRAM"	STRING: тип сегмента: "L": линейный "P": полиномиальный					Определяет количество используемых сегментов кривых указанного типа в указанной памяти >=0: Количество -1: недействительный тип памяти Если параметр 2 не программируется, то выводится сумма линейных и полиномиальных сегментов.
CTABFSEG	INT: число сегментов кривой	STRING: место сохранения: "SRAM", "DRAM"	STRING: тип сегмента: "L": линейный "P": полиномиальный					Определяет количество сегментов кривой указанного типа, которое можно добавить в указанную память >=0: Количество -1: недействительный тип памяти
CTABSEGID	INT: число сегментов кривой	INT: номер таблицы	STRING: тип сегмента: "L": линейный "P": полиномиальный					Определяет количество сегментов кривой указанного типа, используемое в таблице кривых >=0: Количество -1: Таблица не существует

Функции таблиц кривых								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABMSEG	INT: число сегментов кривой	STRING: место сохранения: "SRAM", "DRAM"	STRING: тип сегмента: "L": линейный "P": полиномиальный					Определяет количество макс. возможных сегментов кривой указанного типа сегмента в указанной памяти >=0: Количество -1: Таблица не существует
CTABPOL	INT: число полиномов кривой	STRING: место сохранения: "SRAM", "DRAM"						Определяет количество использованных полиномов кривой в указанной памяти >=0: Количество -1: Таблица не существует
CTABPOLID	INT: число полиномов кривой	INT: номер таблицы						Определяет количество полиномов кривой, которые используются таблицей кривых >=0: Количество -1: Таблица не существует
CTABFPOL	INT: число полиномов кривой	STRING: место сохранения: "SRAM", "DRAM"						Определяет количество макс. возможных полиномов кривой в указанной памяти: >=0: Количество -1: Таблица не существует
CTABMPOL	INT: число полиномов кривой	STRING: место сохранения: "SRAM", "DRAM"						Определяет количество макс. возможных полиномов кривой в указанной памяти: >=0: Количество -1: Таблица не существует
CTABSSV	REAL: позиция ведомой оси	REAL: позиция ведущей оси	INT: номер таблицы	VAR REAL[ ]: результат подъем	AXIS: ведомая ось для масштабирования	AXIS: ведущая ось для масштабирования		Определяет позицию ведомой оси в начале сегмента кривой, относящегося к указанному значению ведущей оси.

Функции таблиц кривых								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABSEV	REAL: позиция ведомой оси	REAL: позиция ведущей оси	INT: номер таблицы	VAR REAL[ ]: результат подъем	AXIS: ведомая ось для масштабирования	AXIS: ведущая ось для масштабирования		Определяет позицию ведомой оси в конце сегмента кривой, относящегося к указанному значению ведущей оси.
CTABTSV	REAL: позиция ведомой оси	INT: номер таблицы	VAR REAL[ ]: результат подъем начало таблицы	AXIS: ведомая ось				Определяет позицию ведомой оси в начале таблицы кривых.
CTABTEV	REAL: позиция ведомой оси	INT: номер таблицы	VAR REAL[ ]: результат подъем конец таблицы	AXIS: ведомая ось				Определяет позицию ведомой оси в конце таблицы кривых.
CTABTSP	REAL: позиция ведущей оси	INT: номер таблицы	VAR REAL[ ]: результат подъем начало таблицы	AXIS: ведущая ось				Определяет позицию ведущей оси в начале таблицы кривых.
CTABTEP	REAL: позиция ведущей оси	INT: номер таблицы	VAR REAL[ ]: результат подъем конец таблицы	AXIS: ведущая ось				Определяет позицию ведущей оси в конце таблицы кривых.
CTABTMIN	REAL: минимальное значение	INT: номер таблицы	REAL: интервал главного значения нижняя граница	REAL: интервал главного значения верхняя граница	AXIS: ведомая ось	AXIS: ведущая ось		Определяет мин. значение ведомой оси во всей области определения таблицы кривых или в определенном интервале.

Функции таблиц кривых								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABTMAX	REAL: максимальное значение	INT: номер таблицы	REAL: интервал главного значения нижняя граница	REAL: интервал главного значения верхняя граница	AXIS: ведомая ось	AXIS: ведущая ось		Определяет макс. значение ведомой оси во всей области определения таблицы кривых или в определенном интервале.
Примечание: Функции таблиц кривых могут быть запрограммированы и в синхронных действиях.								

Осевые функции						
Идентификатор	**Возвращаемое значение**	**Параметр**				
		1.	**2.**	**3.**	**4.**	**Объяснение**
AXNAME	AXIS: идентификатор оси	STRING [ ]: входная строка				Преобразует входную строку в идентификатор оси
AXSTRING	STRING[ ]: имя оси	AXIS: идентификатор оси				Преобразует идентификатор оси в строку
ISAXIS	BOOL: ось имеется (TRUE) или нет (FALSE)	INT: номер гео-оси (1 до 3)				Проверяет, имеется ли указанная как параметр гео-ось 1 до 3 согласно машинным данным MD20050 $MC_AXCONF_GEOAX_ASSIGN_TAB
SPI	AXIS: идентификатор оси	INT: номер шпинделя				Преобразует номер шпинделя в идентификатор оси
AXTOSPI	INT: номер шпинделя	AXIS: идентификатор оси				Преобразует идентификатор оси в номер шпинделя
MODAXVAL	REAL: значение по модулю	AXIS: идентификатор оси	REAL: позиция оси			Рассчитывает из введенной позиции оси остаток по модулю Если указанная ось не является осью по модулю, то позиция оси возвращается без изменений.
POSRANGE	BOOL: заданная позиция в окне позиций (TRUE) или нет (FALSE)	AXIS: идентификатор оси	REAL: контрольная позиция в системе координат	REAL: ширина окна позиций	INT: система координат	Определяет, находится ли заданная позиция оси в окне округ заданной контрольной позиции

Управление инструментом					
Идентификатор	**Возвращаемое значение**	**Параметр**			**Объяснение**
		1.	**2.**	**3.**	
CHKDM	INT: состояние: Результат испытания	INT: номер магазина	INT: D-номер		Проверяет однозначность D-номера в рамках магазина
CHKDNO	INT: состояние: Результат испытания	INT: номер T для 1-го инструмента	INT: номер T для 2-го инструмента	INT: D-номер	Проверяет однозначность D-номера
GETACTT	INT: состояние	INT: номер T	STRING [32]: имя инструмента		Определяет активный инструмент из группы инструментов с одинаковым именем
GETACTTD	INT: состояние: Результат испытания	VAR INT: найденный T-номер (возвращаемое значение)	INT: D-номер		Определяет соответствующий T-номер для абсолютного D-номера
GETDNO	INT: D-номер	INT: номер T	INT: номер резца		Определяет D-номер резца инструмента T
GETT	INT: номер T	STRING [32]: имя инструмента	INT: номер гнезда		Определяет T-номер к имени инструмента
NEWT	INT: номер T	STRING [32]: имя инструмента	INT: номер гнезда		Создает новый инструмент (предоставление данных инструмента) Номер гнезда может отсутствовать.
TOOLENV	INT: состояние	STRING: имя			Сохраняет инструментальное окружение с указанным именем в статической памяти ЧПУ
DELTOOLENV	INT: состояние	STRING: имя			Удаляет инструментальное окружение с указанным именем из статической памяти ЧПУ Удаляет все инструментальные окружения, если имя не указано.
GETTENV	INT: состояние	STRING: имя	VAR INT: T-номер [0] D-номер [1] DL-номер [2]		Определяет T-номер D-номер и DL-номер по инструментальному окружению с указанным именем

арифметика					
Идентификатор	**Возвращаемое значение**	**Параметр**			**Объяснение**
		1.	**2.**	**3.**	
SIN	REAL	REAL			Синус
ASIN	REAL	REAL			Арксинус

арифметика					
Идентификатор	**Возвращаемое значение**	**Параметр**			**Объяснение**
		1.	**2.**	**3.**	
COS	REAL	REAL			Косинус
ACOS	REAL	REAL			Арккосинус
TAN	REAL	REAL			Тангенс
ATAN2	REAL	REAL	REAL		Арктангенс 2
SQRT	REAL	REAL			Квадратный корень
POT	REAL	REAL			Квадрат
TRUNC	REAL	REAL			Целочисленная доля
ROUND	REAL	REAL			Округление
ROUNDUP	REAL	REAL			Округление
ABS	REAL	REAL			Абсолютное значение
LN	REAL	REAL			Натуральный логарифм
exp	REAL	REAL			Показательная функция e^x
MINVAL	REAL	REAL	REAL		Определяет наименьшее значение двух параметров
MAXVAL	REAL	REAL	REAL		Определяет наибольшее значение двух параметров
BOUND	REAL: состояние проверки	REAL: нижняя граница	REAL: верхняя граница	REAL: опорное значение	Определяет, укладывается ли опорное значение в диапазон.
Примечание: Арифметические функции могут быть запрограммированы и в синхронных действиях. В этом случае расчет и обработка этих арифметических функций осуществляется на главном ходе. Для расчетов и в качестве буфера можно использовать и параметр синхронных действий $AC_PARAM[<n>].					

Строковые функции (STRING)					
Идентификатор	**Возвращаемое значение**	**Параметр**			**Объяснение**
		1.	**2.**	**3.**	
ISNUMBER	BOOL	STRING: входная строка			Проверяет, можно ли преобразовать входную строку в число
NUMBER	REAL	STRING: входная строка			Преобразует входную строку в число
TOUPPER	STRING	STRING: входная строка			Преобразует входную строку в прописные буквы
TOLOWER	STRING	STRING: входная строка			Преобразует входную строку в строчные буквы
STRLEN	INT	STRING: входная строка			Определяет длину входной строки до конца строки (/0)

Строковые функции (STRING)					
Идентификатор	**Возвращаемое значение**	**Параметр**			**Объяснение**
		1.	**2.**	**3.**	
INDEX	INT	STRING: входная строка	CHAR: символ поиска		Определяет место символа во входной строке слева направо. 1-й символ строки слева имеет индекс 0.
RINDEX	INT	STRING: входная строка	CHAR: символ поиска		Определяет место символа во входной строке справа налево. 1-й символ строки справа имеет индекс 0.
MINDEX	INT	STRING: входная строка	STRING: символ поиска		Определяет место одного из символов, указанных во 2-м параметре во входной строке слева направо. 1-й символ входной строки слева имеет индекс 0.
SUBSTR	STRING	STRING: входная строка	INT	INT	Определяет подстроку входной строки, описанную началом (2-й параметр) и количеством символов (3-й параметр).
SPRINT	STRING	STRING: входная строка			Определяет отформатированную входную строку

Функции для измерительных циклов								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CALCPOSI	INT: состояние	REAL[3]: исходная позиция в WCS	REAL[3]: инкрементальное указание пути относительно исходной позиции	REAL[5]: мин. расстояния до границ контроля	REAL[3]: поле возврата для возм. инкр. пути	BOOL: пересчет системы единиц да/нет	INT: тип граничного контроля	Проверяет, могут ли исходя из имеющейся стартовой точки гео-оси пройти заданный путь, не нарушив границы осей В том случае, если заданный путь не может быть пройден без нарушений, то возвращается макс. допустимое значение.
GETTCOR	INT: состояние	REAL [11]:	STRING: компонент длины инструмента : Система координат	STRING: имя инструментального окружения	INT: внутренний T-№ инструмента	INT: номер резца (D-№) инструмента	INT: номер зависящей от места коррекции (DL-№ инструмента))	Определяет длины инструмента и компоненты длин инструмента из инструментального окружения или текущего окружения
LENTOAX	INT: состояние	INT[3]: согласование гео-осей	REAL[3]: матрица для отображения длин инструмента в системе координат	STRING: система координат для согласования				Находит информацию по согласованию длин инструмента L1, L2, L3 активного инструмента с абсциссой, ординатой, аппликатой Управление согласованием с геометрическими осями осуществляется через фреймы и активную плоскость (G17 - G19).

SETTCOR	INT: состояние	**1.**	**2.**	**3.**	**4.**	**5.**	**6.**	**7.**	**8.**	**9.**	
		REAL [3]: вектор коррекции в пространстве	STR.: идентификатор компонента	INT: исправл. компонент(ы) 0 - 11	INT: тип операции записи 0 - 3	INT: индекс геооси	STRING: имя инструментального окружения	INT: внутр. T-№ инструмента	INT: D-№ инструмента	INT: DL-№ инструмента	Изменяет компоненты инструмента с учетом граничных условий, входящих в нормирование отдельных компонентов

Прочие функции								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
STRINGIS	INT: информация о строке	STRING: имя проверяемого элемента						Проверяет, доступна ли указанная строка как элемент языка программирования ЧПУ в текущей языковой среде
ISVAR	BOOL: переменная известна да/нет	STRING: имя переменной						Проверяет, содержит ли передаваемый параметр известную в ЧПУ переменную (машинные данные, установочные данные, системная переменная, общие переменные как GUD)
GETVARTYP	INT: тип данных	STRING: имя переменной						Определяет тип данных системной переменной / переменной пользователя
GETVARPHU	INT: Числовое значение физической единицы	STRING: имя переменной						Определяет физические единицы системной переменной / переменной пользователя
GETVARAP	INT: степень защиты для обращений	STRING: имя переменной	STRING: тип доступа					Определяет права доступа к системной переменной / переменной пользователя

Прочие функции								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
GETVARLIM	INT: состояние	STRING: имя переменной	CHAR: указывает, какое предельное значение должно быть выгружено	VAR REAL: возврат предельного значения				Определяет нижнее / верхнее предельного значения системной переменной / переменной пользователя
GETVARDFT	INT: состояние	STRING: имя переменной	VAR REAL/ STRING/ FRAME: возврат стандартного значения	INT: индекс на первую размерность (как опция)	INT: индекс на вторую размерность (как опция)	INT: индекс на третью размерность (как опция)		Определяет стандартное значение системной переменной / переменной пользователя
COLLPAIR	INT: результат проверки	STRING: имя 1-й защищенной области	STRING: имя 2-й защищенной области	BOOL: блокировка аварийных сообщений (опция)				Проверяет на отношение к соударяемой паре
PROTD	REAL: интервал между двумя защищенными областями	STRING: имя 1-й защищенной области	STRING: имя 2-й защищенной области	VAR REAL: возвращаемое значение: 3-мерный вектор расстояния	BOOL: Система единиц для интервала и вектора расстояния (опция)			Рассчитывает интервал между двумя указанными защищенными областями
DELOBJ	INT: номер ошибки	STRING: тип удаляемого компонента	INT: начальный индекс удаляемых компонентов (опция)	INT: конечный индекс удаляемых компонентов (опция)	BOOL: блокировка аварийных сообщений (опция)			Удаляет элементы кинематических цепочек, защищенные области, элементы защищенных областей, соударяемые пары и данные трансформаций

Прочие функции								
Идентификатор	**Возвращаемое значение**	**Параметр**						**Объяснение**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
NAMETOINT	INT: индекс системной переменной	STRING: имя поля системных переменных	STRING: цепочка символов/имя	BOOL: блокировка аварийных сообщений (опция)				Определяет на основе цепочки символов соответствующий индекс системной переменной
ORISOLH	INT: номер ошибки	INT: управляет поведением функции	REAL: первый угол	REAL: второй угол				Помогает пользователю настраивать положения круговых осей станка так, чтобы токарный инструмент устанавливался в определенное, не зависящее от кинематики положение относительно детали. Условие: активна 6-осевая трансформация, параметры которой заданы посредством кинематических цепей.
CORRTRAFO	INT: номер ошибки	REAL: вектор коррекции	INT: модифицируемый элемент	INT: режим коррекции	BOOL: блокировка аварийных сообщений (опция)			Модифицирование векторов смещения или векторов направления осей ориентации в кинематической модели станка.
CORRTC	INT: номер ошибки	REAL: вектор коррекции	INT: модифицируемый элемент	INT: режим коррекции	BOOL: блокировка аварийных сообщений (опция)			Модифицирование векторов смещения или векторов направления ориентируемых инструментальных суппортов после замеров станка.

5.7 Текущий язык в HMI

Следующая таблица содержит все доступные на интерфейсе языки.

Текущий установленный язык может быть запрошен в программе обработки детали или в синхронных действиях через следующую системную переменную:

$AN_LANGUAGE_ON_HMI = <значение>

<значение>	Язык	Краткое обозначение
1	Немецкий (Германия)	DEU
2	Французский	FRA
3	Английский (Соединенное королевство)	ENG
4	Испанский	ESP
6	Итальянский	ITA
7	Голландский	NLD
8	Китайский (упрощенный)	CHS
9	Шведский	SVE
18	Венгерский	HUN
19	Финский	FIN
28	Чешский	CSY
50	Португальский (бразильский)	PTB
53	Польский	PLK
55	Датский	DAN
57	Русский	RUS
68	Словацкий	SKY
72	Румынский	ROM
80	Китайский (традиционный)	CHT
85	Корейский	KOR
87	Японский	JPN
89	Турецкий	TRK

Примечание

Обновление $AN_LANGUAGE_ON_HMI выполняется:

- после запуска системы.
- после сброса (Reset) ЧПУ и/или PLC.
- после переключения на другое ЧПУ в рамках M2N.
- после переключения языка на HMI.

Приложение

A

A.1 Список сокращений

A	
A	Выход
AFIS	Automatic Filter Switch: Автоматическое переключение фильтров
ASCII	American Standard Code for Information Interchange: американский стандарт кода для передачи информации
ASIC	Application Specific Integrated Circuit: схема пользователя
ASUP	Асинхронная подпрограмма
AUTO	Режим работы "Автоматика"
AUXFU	Auxiliary Function: Вспомогательная функция
AWL	Список операторов

B	
BAG	Группа режимов работы (ГРР)
BCD	Binary Coded Decimals: закодированные двоичным кодом десятичные числа
BICO	Binector Connector
BIN	Binary Files: двоичные файлы
BKS	Базовая кинематическая система
BM	Оперативное сообщение
BO	Binector Output
BTSS	Интерфейс пульта оператора

C	
CLC	Регулировка дистанции
CNC	Computerized Numerical Control: Computerunterstützte numerische Steuerung
COM	Communication
CP	Communication Processor
CPU	Central Processing Unit: центральный процессор
CST	Configured Stop: Сконфигурированный останов

D	
DB	Блок данных (PLC)
DBB	Байт блока данных (PLC)
DBD	Двойное слово блока данных (PLC)
DBW	Слово блока данных (PLC)
DBX	Бит блока данных (PLC)

D	
DDS	Drive Data Set: Блок данных привода
DIR	Directory: Директория
DO	Drive Object
DRF	Differential Resolver Function: функция дифференциального преобразования координат (маховичок)
DRY	Dry Run: Подача пробного хода
DW	Слово данных
DWORD	двойное слово (сейчас 32 бита)

E	
E	Вход
EES	Execution from External Storage
E/A	Ввод/вывод
ESR	Расширенный останов и отвод
ETC	ETCКлавиша расширения ">"; расширение панели программных клавиш в том же меню

F	
FB	Функциональный блок (PLC)
FC	Function Call: Функциональный блок (PLC)
FDD	Feed Disable: Блокировка подачи
FdStop	Feed Stop: остановка подачи
FIFO	First In First Out: память, работающая без указания адреса, данные которой считываются в последовательности их сохранения
FM	Сообщение об ошибке
FUP	Функциональные схемы FBD (метод программирования для PLC)
FW	Firmware

g	
GEO	Геометрия, например, гео-ось
GP	Главная программа (PLC)
GUD	Global User Data: глобальные данные пользователя

H	
HEX	сокращение для шестнадцатеричного числа
HiFu	Вспомогательная функция
HMI	Human Machine Interface: интерфейс пользователя SINUMERIK
HSA	Привод главного движения
HT	Handheld Terminal
HW	Аппаратное обеспечение

I	
IBN	Ввод в эксплуатацию
INC	Increment: размер шага
INI	Initializing Data: данные инициализации
IPO	Interpolator

J	
JOG	Jogging: отладочный режим

K	
KOP	Релейно–контактные схемы (LAD) (метод программирования для PLC)

I	
LED	Light Emitting Diode: светодиод
LMS	Система измерения положения
LR	Регулятор положения

M	
Main	Main program: главная программа (OB1, PLC)
MCP	Machine Control Panel: Станочный пульт
MD	Машинные данные
MDA	Manual Data Automatic: ручной ввод
MDS	Motor Data Set: Блок данных двигателя
MELDW	Информационное слово
MKS	Система координат станка
MM	Motor Module
MPF	Main Program File: Главная программа (ЧПУ)
MPI	Multi Point Interface
MSTT	Станочный пульт

N	
NC	Numerical Control: Числовое программное управление (ЧПУ) с подготовкой кадра, диапазоном перемещения и т. п.
NCU	Numerical Control Unit: аппаратный блок ЧПУ
NCK	Numerical Control Kernel
NCSD	NC Start Disable
NST	Интерфейсный сигнал
NV	Смещение нулевой точки
NX	Numerical Extension: модуль расширения осей

O	
OB	Организационный модуль в PLC
OEM	Original Equipment Manufacturer
OP	Operation Panel: панель оператора

P	
PCU	PC Unit: PC-Box (блок ВУ)
PG	Программатор
PLC	Programmable Logic Control: контроллер
PN	PROFINET
PO	Power On
POS	Позиция/позиционирование
PPO	Parameter Prozessdaten Objekt ; циклическая телеграмма данных при передаче с PROFIBUS–DP и профилем "Электрические приводы с регулируемой скоростью"
PPU	Panel Processing Unit (центральный аппаратный компонент панельной СЧПУ, например, SINUMERIK 828D)
PROFIBUS	Process Field Bus: последовательная шина данных
PRT	Тест программы
PTP	Point to Point: точка-точка
PZD	Данные процесса: компонент данных процесса PPO

R	
REF	Функция подвода в референтную точку
REPOS	Функция репозиционирования
RESU	Retrace support
RID	Read In Disable
RP	R-параметр, предопределенная переменная пользователя

S	
SA	Синхронное действие
SBL	Single Block: отдельный кадр
SBT	Safe Brake Test
SCC	Safety Control Channel
SCL	Structured Control Language
SD	Settingdatum или установочные данные
SDI	Safe Direction
SERUPRO	Search-Run by Program Test: Поиск кадра через программный тест
SIC	Safety Info Channel
SKP	Skip: функция для пропуска кадров программы обработки деталей
SLP	Safe Limited Position
SLS	Safely Limited Speed

S	
SMI	Sensor Module Integrated
SOS	Safe Operating Stop
SPF	Sub Program File: подпрограмма (ЧПУ)
SS1	Safe Stop 1
SS2	Safe Stop 2
STO	Safe Torque Off
STW	Управляющее слово
SUG	Окружная скорость круга
SW	ПО

T	
TCU	Thin Client Unit
TM	Terminal Module (SINAMICS)
TO	Tool Offset: Коррекция инструмента
TOA	Tool Offset Active: обозначение (тип файла) для коррекций инструмента
TOFF	Коррекция длин инструмента Online
TRANSMIT	Transform Milling Into Turning: трансформация координат для фрезерной обработки на токарном станке

U	
UP	Подпрограмма
USB	Universal Serial Bus

V	
VDI	Внутренний коммуникационный интерфейс между ЧПУ и PLC

W	
WKS	Система координат детали
WPD	Work Piece Directory: директория детали
WZ	Инструмент
WZV	Управление инструментом

Z	
ZSW	Слово состояния (привода)

Указатель

$

$A_PROBE, 636, 643
$A_PROBE_LIMITED, 644
$AA_ACC, 134
$AA_ATOL, 880
$AA_COUP_ACT
    при буксировке, 935
    при осевом соединении по главному значению, 957
$AA_ESR_ENABLE, 1017
$AA_FGREF, 117
$AA_FGROUP, 117
$AA_G0MODE, 189
$AA_LEAD_SP, 957
$AA_LEAD_SV, 957
$AA_MM, 637
$AA_MM1...4, 644
$AA_MW, 637
$AA_MW1...4, 644
$AC_ACT_PROG_NET_TIME, 1025
$AC_ACTUAL_PARTS, 1028
$AC_AXCTSWA, 921
$AC_AXCTSWE, 922
$AC_CTOL, 880
$AC_CTOL_G0_ABS, 193
$AC_CUT_INV, 834
$AC_CUTMOD, 833
$AC_CUTMOD_ANG, 833
$AC_CUTMODK, 833
$AC_CUTTING_TIME, 1024
$AC_CYCLE_TIME, 1024
$AC_DELAYFST, 891
$AC_ESR_TRIGGER, 1017
$AC_F_TYPE, 150
$AC_FGROUP_MASK, 117
$AC_FZ, 150
$AC_MEA, 637, 644
$AC_OLD_PROG_NET_TIME, 1025
$AC_OLD_PROG_NET_TIME_COUNT, 1025
$AC_OPERATING_TIME, 1024
$AC_OTOL, 880
$AC_OTOL_ G0_ABS, 193
$AC_PROG_NET_TIME_TRIGGER, 1025
$AC_REPOS_PATH_MODE, 899
$AC_REQUIRED_PARTS, 1028
$AC_S_TYPE, 102
$AC_SMAXVELO, 902
$AC_SMAXVELO_INFO, 902
$AC_SPECIAL_PARTS, 1028
$AC_STOLF, 193
$AC_SVC, 102
$AC_TOFF, 92
$AC_TOFFCR, 92
$AC_TOFFL, 92
$AC_TOFFR, 92
$AC_TOTAL_PARTS, 1028
$AC_TRAFO_CORR_ELEM_P, 761
$AC_TRAFO_CORR_ELEM_T, 761
$AC_TRAFO_ORIAX_LOC, 762
$AN_AXCTAS, 922
$AN_AXCTSWA, 921
$AN_ESR_TRIGGER, 1017
$AN_LANGUAGE_ON_HMI, 1365
$AN_POWERON_TIME, 1024
$AN_SETUP_TIME, 1024
$NT_CLOSE_CHAIN_T, 762
$NT_CNTRL, 761
$NT_CORR_ELEM_P, 761
$NT_CORR_ELEM_T, 761
$NT_NAME, 754
$NT_ROT_AX_NAME, 829
$NT_TRAFO_INDEX, 754
$P_ACTBFRAME, 675
$P_AD, 833
$P_AEP, 290
$P_APDV, 290
$P_APR, 290
$P_BFRAME, 675
$P_CHBFRAME, 674
$P_CHBFRMASK, 675
$P_CTOL, 880
$P_CTOL_ G0_ABS, 194
$P_CUT_INV, 834
$P_CUTMOD, 833
$P_CUTMOD_ANG, 833
$P_CUTMOD_ERR, 835
$P_CUTMODK, 833
$P_DELAYFST, 891
$P_F_TYPE, 150
$P_FGROUP_MASK, 118
$P_FZ, 150
$P_GWPS, 109
$P_IFRAME, 676
$P_IS_EES_PATH, 582
$P_NCBFRAME, 674
$P_NCBFRMASK, 675

$P_ORI_DIFF, 824
$P_ORI_POS, 824
$P_ORI_SOL, 825
$P_ORI_STAT, 828
$P_OTOL, 880
$P_OTOL_ G0_ABS, 194
$P_PATH, 581
$P_PFRAME, 676
$P_PROG, 581
$P_PROGPATH, 581
$P_S_TYPE, 103
$P_SIM, 647, 931, 1029
$P_STACK, 581
$P_STOLF, 193
$P_SUBPAR, 514
$P_SVC, 103
$P_TOFF, 92
$P_TOFFCR, 92
$P_TOFFL, 92
$P_TOFFR, 92
$P_TOOLENV, 843
$P_TOOLENVN, 843
$P_WORKAREA_CS_COORD_SYSTEM, 364
$P_WORKAREA_CS_LIMIT_MINUS, 365
$P_WORKAREA_CS_LIMIT_PLUS, 365
$P_WORKAREA_CS_MINUS_ENABLE, 364
$P_WORKAREA_CS_PLUS_ENABLE, 364
$PA_ATOL, 880
$PA_FGREF, 117
$PA_FGROUP, 118
$SA_LEAD_TYPE, 956
$SC_PA_ACTIV_IMMED, 602
$SN_PA_ACTIV_IMMED, 602
$TC_CARR_CORR_ELEM, 816
$TC_CARR1...14, 803
$TC_CARR13, 810
$TC_CARR14, 810
$TC_CARR18...23, 803
$TC_CARR18[m], 807
$TC_DP1 ... 25, 763
$TC_TP_MAX_VELO, 99

*****

* (функция вычисления), 455

/

/ (функция вычисления), 455

+

+ (функция вычисления), 455

<

< (оператор сравнения), 457
<< (связывающий оператор), 464
<= (оператор сравнения), 457
<> (оператор сравнения), 457

=

== (оператор сравнения), 457

>

> (оператор сравнения), 457
>= (оператор сравнения), 457

A

ABS, 455
AC, 157
ACC, 133
ACCLIMA, 872
ACN, 164
ACOS, 455
ACP, 164
ACTBLOCNO, 525
ACTFRAME, 653
ADIS, 305
ADISPOS, 305
ADISPOSA, 648
AFISOF, 883
AFISON, 883
ALF
  для быстрого отвода от контура, 567
  для быстрого отвода при резьбонарезании, 239
AMIRROR, 337
AND, 457
ANG, 219
ANG1, 219
ANG2, 219
AP, 182
APR, 425
APRB, 425
APRP, 425

APW, 425
APWB, 425
APWP, 425
AR
Программирование окружности, 202
AROT, 325
AROTS, 331
AS, 505
ASCALE, 334
ASIN, 455
ASPLINE, 614
ATAN2, 455
ATOL, 876
ATRANS, 319
AV, 964
AX, 914
AXCTSWE, 920
AXCTSWEC, 920
AXCTSWED, 920
AXIS, 408
AXNAME, 463
AXSTRING, 914
AXTOCHAN, 912
AXTOSPI, 914
A-сплайн, 621

B

B_AND, 457
B_NOT, 457
B_OR, 457
B_XOR, 457
BAUTO, 614
BFRAME, 653
BKS, 42
BLOCK, 551
BLSYNC, 561
BNAT, 614
BNS, 43
BOOL, 408
BOUND, 437
BRISK, 870
BRISKA, 870
BSPLINE, 614
BTAN, 614
B-сплайн, 622

C

CAC, 613
CACN, 613
CACP, 613
CADAPTOF, 924
CADAPTON, 924
CALL, 550
CALLPATH, 553
CASE, 483
CDC, 613
CDOF, 294
CDOF2, 294
CDON, 294
CFC, 138
CFIN, 138
CFINE, 662
CFTCP, 138
Chan, 408
CHANDATA, 582
CHAR, 408
CHF, 254
CHKDNO, 800
CHR, 254
CIC, 613
CIP, 206
CLEARM, 499
CLRINT, 563
COARSE, 964
COARSEA, 648
COMPCAD, 626
COMPCURV, 626
COMPLETE, 582
COMPOF, 626
COMPON, 626
COMPSURF, 626
CONTDCON, 1057
CONTPRON, 1051
CORROF, 346
CORRTC, 813
CORRTRAFO, 755
COS, 455
COUPDEF, 964
COUPDEL, 964
COUPOF, 964
COUPOFS, 964
COUPON, 964
COUPONC, 964
COUPRES, 964
CP, 733
CPBC, 976
CPDEF, 975
CPDEL, 975
CPFMOF, 978
CPFMON, 978
CPFMSON, 977

CPFPOS + CPOF, 978
CPFPOS + CPON, 976
CPFRS, 976
CPLA, 975
CPLCTID, 975
CPLDEF, 975
CPLDEL, 975
CPLDEN, 975
CPLINSC, 980
CPLINTR, 980
CPLNUM, 975
CPLOF, 975
CPLON, 975
CPLOUTSC, 980
CPLOUTTR, 980
CPLPOS, 976
CPLSETVAL, 976
CPMALARM, 981
CPMBRAKE, 981
CPMPRT, 980
CPMRESET, 979
CPMSTART, 980
CPMVDI, 981
CPOF, 975
CPON, 975
CPRECOF, 881
CPRECON, 881
CPROT, 599
CPROTDEF, 595
CPSETTYPE, 981
CPSYNCOP, 981
CPSYNCOP2, 981
CPSYNCOV, 981
CPSYNFIP, 981
CPSYNFIP2, 981
CPSYNFIV, 981
CR, 200
CROTS, 331
CSPLINE, 614
CT, 208
CTAB, 946
CTABDEF, 936
CTABDEL, 942
CTABEND, 936
CTABEXISTS, 942
CTABFNO, 950
CTABFPOL, 950
CTABFSEG, 950
CTABID, 944
CTABINV, 946
CTABISLOCK, 944
CTABLOCK, 943
CTABMEMTYP, 944
CTABMPOL, 950
CTABMSEG, 950
CTABNO, 950
CTABNOMEM, 950
CTABPERIOD, 944
CTABPOL, 950
CTABPOLID, 950
CTABSEG, 950
CTABSEGID, 950
CTABSEV, 946
CTABSSV, 946
CTABTEP, 946
CTABTEV, 946
CTABTMAX, 946
CTABTMIN, 946
CTABTSP, 946
CTABTSV, 946
CTABUNLOCK, 943
CTOL, 876
CTOLG0, 190
CTRANS, 662
CUT2D, 295
CUT2DD, 295
CUT2DF, 295
CUT2DFD, 295
CUT3DC, 778
CUT3DCC, 789
CUT3DCCD, 789
CUT3DCD, 778
CUT3DF, 783
CUT3DFD, 783
CUT3DFF, 783
CUT3DFS, 783
CUTCONOF, 298
CUTCONON, 298
CUTMOD, 830
CUTMODK, 830
CYCLE4071
    внешнее программирование, 1180
CYCLE4072
    внешнее программирование, 1182
CYCLE4073
    внешнее программирование, 1186
CYCLE4074
    внешнее программирование, 1187
CYCLE4075
    внешнее программирование, 1190
CYCLE4077
    внешнее программирование, 1193
CYCLE4078
    внешнее программирование, 1197

CYCLE4079
внешнее программирование, 1199
CYCLE435 - установка системы координат правящего инструмента
внешнее программирование, 1135
CYCLE495 - Профилирование
внешнее программирование, 1135
CYCLE60 - гравирование
внешнее программирование, 1085
CYCLE61 - плоское фрезерование
внешнее программирование, 1088
CYCLE62 - вызов контура
внешнее программирование, 1090
CYCLE63 – Фрезерование контурного кармана / Контурный карман – остаточный материал / Фрезерование контурной цапфы / Контурная цапфа – остаточный материал
внешнее программирование, 1091
CYCLE64 - предварительное сверление кармана
внешнее программирование, 1093
CYCLE70 - резьбофрезерование
внешнее программирование, 1095
CYCLE72 - фрезерование траектории
внешнее программирование, 1097
CYCLE76 - прямоугольная цапфа
внешнее программирование, 1101
CYCLE77 - круговая цапфа
внешнее программирование, 1103
CYCLE78 - Сверлильное резьбофрезерование
внешнее программирование, 1105
CYCLE782 – Коррекция по загрузке
внешнее программирование, 1137
CYCLE79 - многогранник
внешнее программирование, 1108
CYCLE800 – Поворот плоскости, поворот инструмента, точная установка инструмента
внешнее программирование, 1140
CYCLE801 – Образец позиции "Решетка или рамка"
внешнее программирование, 1144
CYCLE802 - любые позиции
внешнее программирование, 1145
CYCLE806 — интерполированное вращение
внешнее программирование, 1149
CYCLE81 - центрование
внешнее программирование, 1111
CYCLE82 - сверление
внешнее программирование, 1112
CYCLE83 – Глубокое сверление 1
внешнее программирование, 1114
CYCLE830 - Глубокое сверление 2
внешнее программирование, 1149
CYCLE832 - High Speed Settings
внешнее программирование, 1156
CYCLE84 - нарезание внутренней резьбы без компенсирующего патрона
внешнее программирование, 1117
CYCLE840 - нарезание внутренней резьбы с компенсирующим патроном
внешнее программирование, 1160
CYCLE85 - развертывание
внешнее программирование, 1120
CYCLE86 - растачивание
внешнее программирование, 1121
CYCLE899 – открытый паз
внешнее программирование, 1162
CYCLE92 - отрез
внешнее программирование, 1123
CYCLE930 - выточка
внешнее программирование, 1165
CYCLE940 – Канавка формы E и F, резьбовая канавка
внешнее программирование, 1169
CYCLE95 - съём припуска вдоль контура
внешнее программирование, 1124
CYCLE951- обработка резаньем
внешнее программирование, 1171
CYCLE952 – Обработка резаньем / Обработка резаньем – остаток / Прорез / Прорез – остаток / Токарная выточка / Токарная выточка – остаток
внешнее программирование, 1174
CYCLE98 - цепочка резьб
внешнее программирование, 1126
CYCLE99 - нарезание резьбы резцом
внешнее программирование, 1130
C-сплайн, 623

D

D..., 84
D0, 84
DAC, 172
DC, 164
DEF, 408
DEFAULT, 483
DEFINE ... AS, 505
DELAYFSTOF, 888
DELAYFSTON, 888
DELDL, 767
DELETE, 589
DELOBJ, 745
DELTOOLENV, 841
DIACYCOFA, 172
DIAM90, 170

DIAM90A, 172
DIAMCHAN, 172
DIAMCHANA, 172
DIAMCYCOF, 170
DIAMOF, 170
DIAMOFA, 172
DIAMON, 170
DIAMONA, 172
DIC, 172
DILF, 239
DIN 66217, 40
Disable, 563
DISC, 276
DISCL, 279
DISPLOF, 525
DISPLON, 525
DISPR, 893
DISR, 279
DISRP, 279
DITE, 237
DITS, 237
DIV, 455
dl, 765
DO, 992
DRFOF, 348
DRIVE, 870
DRIVEA, 870
DRVPRD, 929
DRVPWR, 929
DV, 964
DYNFINISH, 874
DYNNORM, 874
DYNPOS, 874
DYNPREC, 874
DYNROUGH, 874
DYNSEMIFIN, 874

E

Easy XML, 1035
EAUTO, 614
EES, 573
EG
    Электронный редуктор, 957
EGDEF, 958
EGDEL, 963
EGOFC, 962
EGOFS, 962
EGON, 959
EGONSYN, 959
EGONSYNE, 959
ELSE, 493, 992
Enable, 563
ENAT, 614
ENDFOR, 495
ENDIF, 493
ENDLABEL, 485
ENDLOOP, 495
ENDWHILE, 497
ENS, 43
ESR, 1016
ESRR, 1022
ESRS, 1021
ETAN, 614
EVERY, 992
EXECSTRING, 453
EXECTAB, 1062
EXECUTE, 1065
exp, 455
EXTCALL
    для SINUMERIK 840D sl, 555
EXTCLOSE, 1039
EXTERN, 544
EXTOPEN, 1039

F

F...
    для подачи, 110
    при линейной интерполяции, 194
    при резьбонарезании G34 G35, 236
FA, 129
FAD, 279
false, 408
FB, 144
FCTDEF, 774
FCUB, 865
FD, 135
FDA, 135
Feedback senden, 23
FENDNORM, 647
FFWOF, 875
FFWON, 875
FGREF, 110
FGROUP, 110
FIFOCTRL, 886
FILEDATE, 593
FILEINFO, 593
FILESIZE, 593
FILESTAT, 593
FILETIME, 593
FINE, 964
FINEA, 648
FL, 110

FLIM, 904
FLIN, 865
FMA, 141
FNORM, 865
FOR, 495
FP, 367
FPO, 865
FPR, 129
FPRAOF, 129
FPRAON, 129
FRAME, 408
FRC, 254
FRCM, 254
FROM, 992
FTOCOF, 778
FTOCON, 778

G

G1, 194
G110, 181
G111, 181
G112, 181
G140, 279
G141, 279
G142, 279
G143, 279
G147, 279
G148, 279
G153
  отключить для фрейма, 345
  при смещении нулевой точки, 150
G17, 153
G18, 153
G19, 153
G2, 197
G247, 279
G248, 279
G25
  Ограничение рабочего поля, 360
  Ограничение скорости шпинделя, 109
G26
  Ограничение рабочего поля, 360
  Ограничение скорости шпинделя, 109
G290, 1049
G291, 1049
G3, 197
G33, 229
G335, 243
G336, 243
G34, 236
G340, 279
G341, 279
G347, 279
G348, 279
G35, 236
G4, 376
G40, 260
G41, 260
G42, 260
G450, 276
G451, 276
G460, 290
G461, 290
G462, 290
G5, 729
G500
  при смещении нулевой точки, 150
G505 ... G599, 150
G53
  отключить для фрейма, 345
  при смещении нулевой точки, 150
G54 ... G57, 150
G58, 323
G59, 323
G60, 302
G601, 302
G602, 302
G603, 302
G62, 647
G621, 647
G64, 305
G641, 305
G642, 305
G643, 305
G644, 305
G645, 305
G7, 729
G70, 166
G700, 166
G71, 166
G710, 166
G74, 366
G75, 367
G810 ... G819, 646
G820 ... G829, 646
G9, 302
G90, 157
G91, 159
G93, 110
G94, 110
G95, 110
G96, 103
G961, 103

G962, 103
G97, 103
G971, 103
G972, 103
G973, 103
GEOAX, 916
GET, 908
GETACTTD, 802
GETD, 908
GETDNO, 801
GETTCOR, 844
GETTENV, 842
GETVARAP, 442
GETVARDFT, 444
GETVARDIM, 443
GETVARLIM, 443
GETVARPHU, 441
GETVARTYP, 445
GFRAME0 ... GFRAME100, 350
GOTO, 479
GOTOB, 479
GOTOC, 479
GOTOF, 479
GOTOS, 478
GP, 450
GROUP_ADDEND — конец дополнения ввода
внешнее программирование, 1202
GROUP_BEGIN — Начало блока программы
внешнее программирование, 1201
GROUP_END — Конец блока программы
внешнее программирование, 1202
GUD, 409
GWPSOF, 108
GWPSON, 108
G-группа
Технология, 874
G-код
запрограммировать косвенно, 449
G-команды, 1302

H

High Speed Settings — CYCLE832
внешнее программирование, 1156
HOLES1 – Образец позиции "Линия"
внешнее программирование, 1070
HOLES2 – Образец позиции "Окружность или неполная окружность"
внешнее программирование, 1071

I

I...
при круговой интерполяции, 197
при резьбонарезании G33, 229
при резьбонарезании G34 G35, 236
IC, 159
ID, 992
IDS, 992
IF, 493
IFRAME, 653
INDEX, 466
INICF, 408
INIPO, 408
INIRE, 408
INIT, 499
INITIAL, 582
INITIAL_INI, 582
INT, 408
INTERSEC, 1061
INVCCW, 214
INVCW, 214
IPOBRKA, 648
IPOENDA, 648
IPOSTOP, 964
IPTRLOCK, 891
IPTRUNLOCK, 891
IR, 243
ISAXIS, 914
ISFILE, 592
ISNUMBER, 463
ISOCALL, 552
ISVAR, 439

J

J...
при круговой интерполяции, 197
при резьбонарезании G34 G35, 236
JERKLIM, 903
JERKLIMA, 872
JR, 243

K

K...
при круговой интерполяции, 197
при резьбонарезании G33, 229
при резьбонарезании G34 G35, 236
KONT, 270

KONTC, 270
KONTT, 270
KR, 243

L

L..., 542
LEAD, 693
LEADOF, 952
LEADON, 952
LENTOAX, 862
LF, 50
LFOF, 239
LFON, 239
LFPOS, 239
LFTXT, 239
LFWP, 239
LIFTFAST, 565
LIMS, 103
Line Feed, 50
Link
    Lead-Link-ось, 386
    -оси, 384
    -переменные, 406
LLI, 421
LN, 455
LONGHOLE - продольный паз
    внешнее программирование, 1083
LookAhead, 310
LOOP, 495
LUD, 409

M

M..., 354
M0, 354
M1, 354
M17, 529
M19
    при позиционировании шпинделя, 123
    Функции M, 354
M2, 354
M3, 93
M30, 529
M4, 93
M40, 354
M41, 354
M42, 354
M43, 354
M44, 354
M45, 354
M5, 93
M6, 69
M70, 123
MASLDEF, 988
MASLDEL, 988
MASLOF, 988
MASLOFS, 988
MASLON, 988
MATCH, 466
MAXVAL, 437
MCALL, 548
MD10010, 499
MD10210, 813
MD10240, 168
MD10260, 166
MD10280, 499
MD10651, 244
MD10710, 239
MD15800, 405
MD18104, 840
MD18116, 841
MD18156, 405
MD20360, 848
MD24558, 850
MD24658, 850
MD9440, 86
MEAC, 637
MEAFRAME, 668
MEAS, 635
MEASA, 637
MEAW, 635
MEAWA, 637
MINDEX, 466
MINVAL, 437
MIRROR, 337
MMC, 1034
MOD, 455
MODAXVAL, 914
MPF, 572
MSG, 357
mySupport-Dokumentation, 23

N

NAMETOINT, 749
NCK, 408
NEWCONF, 1030
NOC, 964
NORM, 270
NOT, 457
NPROT, 599
NPROTDEF, 595

NUMBER, 463
NUT, 704

O

OEMIPO1/2, 646
OFFN, 260
OMA1 ... OMA5, 646
OpenSSL, 26
OR, 457
ORIAXES, 702
ORIC, 794
ORICONCCW, 704
ORICONCW, 704
ORICONIO, 704
ORICONTO, 704
ORICURVE, 708
ORID, 794
ORIEULER, 702
ORIMKS, 700
ORIPATH, 715
ORIPATHS, 715
ORIPLANE, 704
ORIRESET(A, B, C), 691
ORIROTA, 711
ORIROTC
при вращение ориентации инструмента, 711
при интерполяции вращения инструмента, 717
ORIROTR, 711
ORIROTT, 711
ORIRPY, 702
ORIRPY2, 702
ORIS, 794
ORISOF, 723
ORISOLH, 819
ORISON, 723
ORIVECT, 702
ORIVIRT1, 702
ORIVIRT2, 702
ORIWKS, 700
OS, 993
OSB, 993
OSC, 794
OSCILL, 998
OSCTRL, 993
OSD, 794
OSE, 993
OSNSC, 993
OSOF, 794
OSP1, 993
OSP2, 993
OSS, 794
OSSE, 794
OST, 794
OST1, 993
OST2, 993
OTOL, 876
OTOLG0, 190
OVR, 132
OVRA, 132
OVRRAP, 132

P

p..., 546
P_ACTFRAME, 676
PACCLIM, 906
PAROT, 342
PAROTOF, 342
PCALL, 553
PDELAYOF, 1005
PDELAYON, 1005
PFRAME, 653
PHI
Полиномы ориентации, 710
при ориентации вдоль боковой поверхности конуса, 704
PHU, 422
PL
при полиномиальной интерполяции, 627
при сплайн-интерполяции, 614
PLC
-оси, 384
PM, 279
po, 627
PO[PHI]
Полиномы ориентации, 710
при вращение ориентации инструмента, 715
при ориентации вдоль боковой поверхности конуса, 704
PO[PSI]
Полиномы ориентации, 710
при вращение ориентации инструмента, 715
при ориентации вдоль боковой поверхности конуса, 704
PO[THT]
Полиномы ориентации, 710
при вращение ориентации инструмента, 715
PO[XH]
Полиномы ориентации, 710
при задаче ориентации двух контактных точек, 708

PO[YH]
Полиномы ориентации, 710
при задаче ориентации двух контактных точек, 708
PO[ZH]
Полиномы ориентации, 710
при задаче ориентации двух контактных точек, 708
POCKET3 - Прямоугольный карман
внешнее программирование, 1073
POCKET4 - Круговой карман
внешнее программирование, 1075
POLF
для быстрого отвода при резьбонарезании, 239
для управляемого ЧПУ отвода, 1017
POLFA, 1017
POLFMASK
для быстрого отвода при резьбонарезании, 239
для управляемого ЧПУ отвода, 1017
POLFMLIN
для быстрого отвода при резьбонарезании, 239
для управляемого ЧПУ отвода, 1017
POLY, 627
POLYPATH, 627
PON, 1013
PONS, 1005
Pos, 118
POSA, 118
POSFS, 964
POSP, 118
POT, 455
PR, 279
PREPRO, 528
PRESETON, 665
PRESETONS, 666
Prio, 561
PRLOC, 408
Process DataShare, 1039
PROTA, 750
PROTD, 752
PROTS, 751
PSI
Полиномы ориентации, 710
при ориентации вдоль боковой поверхности конуса, 704
PTP, 733
PTPG0, 733
PTPWOC, 733
PUD, 409
PUNCHACC, 1005
PUTFTOC, 777
PUTFTOCF, 776
PW, 614

Q

Qu, 353

R

RAC, 172
READ, 590
REAL, 408
REDEF, 414
RELEASE, 908
REP, 431
REPEAT, 485
REPEATB, 485
REPOSA, 893
REPOSH, 893
REPOSHA, 893
REPOSL, 893
REPOSQ, 893
REPOSQA, 893
RET, 530
RET (параметрируемая), 531
RETB (параметрируемая), 538
RG, 404
RIC, 172
RINDEX, 466
RMBBL, 893
RMEBL, 893
RMIBL, 893
RMNBL, 893
RND, 254
RNDM, 254
ROT, 325
ROTS, 331
ROUND, 455
ROUNDUP, 460
RP, 182
RPL, 325
Run MyScreens, 1035
R-параметры (параметры для расчета)
Глобальный, 404
Спец. для канала, 403

S

S, 93

Save, 519
SBLOF, 520
SBLON, 520
SCALE, 334
SCC, 103
SCPARA, 923
SD, 614
SD41610, 762
SD41611, 762
SD42010, 238
SD42440, 160
SD42442, 160
SD42450$SC_CONTPREC, 882
SD42451 $SC_CONTPREC_G00_ABS, 882
SD42460 $SC_MINFEED (мин. подача по траектории при CPRECON), 882
SD42465, 311
SD42466, 311
SD42475, 721
SD42476, 721
SD42477, 721
SD42900, 769
SD42910, 769
SD42920, 770
SD42930, 770
SD42935, 772
SD42940, 773
SD42984, 832
SD43240, 125
SD43250, 125
set, 431
SETAL, 1043
SETDNO, 801
SETINT, 561
SETM, 499
SETMS, 93
SETTCOR, 850
SF, 229
Siemens Industry Online Support
  Приложение, 25
SIN, 455
SINUMERIK, 19
SLOT1 - Продольный паз
  внешнее программирование, 1078
SLOT2 - кольцевая канавка
  внешнее программирование, 1081
SOFT, 870
SOFTA, 870
SON, 1005
SONS, 1005
SPCOF, 122
SPCON, 122
SPF, 572
SPI, 914
SPIF1, 1005
SPIF2, 1005
SPLINEPATH, 625
SPN, 1010
SPOF, 1005
SPOS, 123
SPOSA, 123
SPP, 1010
SPRINT, 469
SQRT, 455
SR, 141
SRA, 141
ST, 141
STA, 141
START, 499
STARTFIFO, 886
STAT, 734
STOLF, 190
STOPFIFO, 886
stopre, 886
STRING, 408
STRINGIS, 1031
STRLEN, 466
SUBSTR, 467
SUG, 108
SUPA
  отключить для фрейма, 345
  при смещении нулевой точки, 150
SVC, 97
SYNR, 408
SYNRW, 408
SYNW, 408

T

T0, 68
TAN, 455
TANG, 982
TANGDEL, 987
TANGOF, 986
TANGON, 985
TCARR, 808
TCOABS, 808
TCOFR, 808
TCOFRX, 808
TCOFRY, 808
TCOFRZ, 808

THETA
при вращение ориентации инструмента, 711
при интерполяции вращения инструмента, 717
TILT, 693
TLIFT, 984
TMOF, 1015
TMON, 1015
TOFF, 88
TOFFL, 88
TOFFLR, 88
TOFFOF, 817
TOFFON, 817
TOFFR, 88
TOFRAME, 342
TOFRAMEX, 342
TOFRAMEY, 342
TOFRAMEZ, 342
TOLOWER, 465
TOOLENV, 838
TOROT, 342
TOROTOF, 342
TOROTX, 342
TOROTY, 342
TOROTZ, 342
TOUPPER, 465
TOWBCS, 771
TOWKCS, 771
TOWMCS, 771
TOWSTD, 771
TOWTCS, 771
TOWWCS, 771
TRAANG
с программируемым углом, 727
TRACON, 731
TRACYL, 724
TRAFOOF, 745
TRAFOON, 754
TRAILOF, 932
TRAILON, 932
TRANS, 319
Transmit, 724
TRAORI, 690
TRUE, 408
TRUNC, 455
Tu, 738
TURN, 212

U

ULI, 421
UNTIL, 497

V

VELOLIM, 901
VELOLIMA, 872

W

WAB, 279
WAITC, 964
WAITE, 499
WAITENC, 922
WAITM, 499
WAITMC, 499
WAITP, 118
WAITS, 123
WALCS<n>, 363
WALCS0, 363
WALIMOF, 360
WALIMON, 360
WHEN, 992
WHEN-DO, 1002
WHENEVER, 992
WHENEVER-DO, 1002
WHILE, 497
WORKPIECE, 1045
WRITE, 585
WRTPR, 358
WCS, 44
точная установка на детали, 342

X

X..., 179
XML пользователя, 1035
XOR, 457

Y

Y..., 179

Z

Z..., 179

A

Абсолютный размер, 34

Аварийные сообщения
устанавливать в программе ЧПУ, 1043
Автоматический указатель прерываний, 893
Автоматического разделения пути, 1010
Адрес
Присваивание значения, 51
Адреса, 389
Адреса OEM, 646
Адресация, 575
Асинхронное качание, 993
Атрибуты позиций
запрограммировать косвенно, 450

Б

Базовая кинематическая система, 42
Базовая система нулевой точки, 43
Базовое смещение, 43
Бесконечный цикл, 495
Блок программы
-повторение, 485
Буквы адреса, 1289
Буксировка, 932
Буксируемые оси, 934
Быстрый отвод
Резьбонарезание, 239
Быстрый отвод от контура, 565

В

Веб-страницы третьих лиц, 21
Ведомая ось
при осевом соединении по главному значению, 952
Ведущая ось
при осевом соединении по главному значению, 952
Вектор направления, 696
Вектор ориентации THETA, 711
внешнее программирование, 1201
Внешнее смещение нулевой точки, 664
Внутренняя остановка предварительной обработки, 378
Вращение
вектора ориентации, 711
программируемое, 325
Время обработки, 1024
Время ожидания, 376
Вспомогательная точка фрезы, 782
Вывод
на внешнее устройство/файл, 1039
Вывод вспомогательной функции
быстрый, 353
в режиме управления траекторией, 354
Свойства вспомогательных функций, 351
Вызов контура - CYCLE62
внешнее программирование, 1090
Вызов подпрограммы с указанием пути и параметрами, 553
Вырубка, 1005
Высокоуровневый язык программирования ЧПУ, 48
Выточка - CYCLE930
внешнее программирование, 1165
Вычисление хода, 388

Г

Геометрия
-оси, 379
Гео-ось
переключить, 916
Главная запись, 175
Глобальная память программ обработки деталей (GDIR), 573
Глубина вложенности
управляющих структур, 492
Глубина врезания, 782
Глубокое сверление 1 – CYCLE83
внешнее программирование, 1114
Глубокое сверление 2 - CYCLE830
внешнее программирование, 1149
Гравирование - CYCLE60
внешнее программирование, 1085
Граничные условия для трансформаций, 744

Д

Данные окружности
вычислить, 1063
Движение "от точки к точке", 732
Движение "от точки к точке" в декартовой системе координат, 732
Двоичная постоянная, 394
Декартовы координаты, 32
Десятичная постоянная, 393
Деталь
-главная директория, 572
-директории, 572
-контур, 178
-Нулевая точка, 38
-счетчик, 1027

Дополнительные оси, 380
Допуски G0, 190

Ж

Жесткий упор, 371

З

Заданная точка, 178
Зажимной момент
-жесткий упор, 374
Закругление, 254
Замедление на внутренних углах, 648
Замедление на всех углах, 648
Запрограммированная остановка, 356
Защищенные области, 595
Значение S
Интерпретация, 95

И

Идентификатор, 46
для системных переменных, 56
для специальных числовых данных, 56
для цепочки символов, 56
Имя диска, 576
Имя подпрограммы в коде DIN, 579
Имя файла, 578
Индекс поля, 433
Индикация кадра
подавить, 525
Инициализация
полей, 431
Инкрементальный размер, 36
Инструмент
-Группа, 74
-коррекция длин, 71
-коррекция радиуса, 72
-ориентация, 794
-острие, 73
-память коррекций, 73
-параметры, 763
-резец, 84
смена с M6, 69
смена с номером T, 68
-тип, 74
-Типовой номер, 74
-Точка смены, 39
Инструментальный суппорт
-Исходная тока, 39
-кинематика, 803
-ориентируемый, 808
Интерполированное вращение — CYCLE806
внешнее программирование, 1149
Интерполяция вектора вращения, 712
Исходные точки, 38
Исходный радиус, 116

К

Кадр, 47
-ende LF, 56
-Длина, 50
-Конец, 50
-Номер, 50
Последовательность операторов, 50
пропустить, 52
Кадр останова, 892
Канавка формы E и F, резьбовая канавка – CYCLE940
внешнее программирование, 1169
Канал
-оси, 382
Качание
Асинхронное, 993
Асинхронное качание, 993
Синхронное качание, 998
управлять через синхронное действие, 998
Частичная подача, 1001
Кинематика
подвижная, 807
Кольцевая канавка - SLOT2
внешнее программирование, 1081
Команда, 47
-оси, 383
Команда движения, 178
Комментарии, 51
Компонент фрейма
FI, 658
tr, 658
Конец блока программы — GROUP_END
внешнее программирование, 1202
Конец дополнения ввода — GROUP_ADDEND
внешнее программирование, 1202
Коническая резьба, 235
Контроль столкновений, 294
Контур
-вычислитель, 220
-кодирование, 1057
повторный подвод, 893
подвести/выйти, 270
-подготовка, 1051

-таблица, 1051
-элемент, 177
Координаты
Декартовы, 32
Полярные, 33
цилиндрические, 183
Коррекции на инструмент
аддитивные, 765
Коррекции радиуса инструмента
Замедление на углах, 647
Коррекция
Длины инструмента, 71
-плоскость, 297
Радиус инструмента, 72
Коррекция длин инструмента Online, 817
Коррекция инструмента
Память коррекций, 762
Система координат для значений износа, 771
-Смещение, 88
Коррекция по загрузке – CYCLE782
внешнее программирование, 1137
Коррекция радиуса инструмента
CUT2DF, 297
на наружных углах, 276
Косвенное программирование
G-кодов, 449
адресов, 447
атрибутов позиций, 450
строк программы обработки детали, 453
Косоугольная трансформация (TRAANG)
с программируемым углом, 727
Коэффициент масштабирования, 334
Коэффициент полинома, 628
коэффициент связи, 932
Критерий окончания движения
программируемый, 648
Круговая интерполяция
Винтовая интерполяция, 212
с промежуточной и конечной точкой, 206
Круговая цапфа - CYCLE77
внешнее программирование, 1103
Круговой карман - POCKET4
внешнее программирование, 1075

Л

Левая резьба, 231
Линейная
-интерполяция, 194
Логические операторы, 457
Любые позиции - CYCLE802
внешнее программирование, 1145

М

Макрос, 505
Маршрут поиска
при вызове подпрограммы, 580
Программируемый маршрут поиска, 554
Массив, 431
Мастер-шпиндель, 381
Матричный штрих-код, 25
Маховичок
-наложение, 135
Маятниковое движение
Область возврата, 1001
Подача в точке возврата, 1003
Точка возврата, 1001
Метка, 485
Метка перехода
при переходах в программе, 480
при повторении программных блоков, 485
Многогранник - CYCLE79
внешнее программирование, 1108
модальное действие, 49

Н

Набор символов, 56
Наградка, 83
Нарезание внутренней резьбы без
компенсирующего патрона - CYCLE84
внешнее программирование, 1117
Нарезание внутренней резьбы с
компенсирующим патроном - CYCLE840
внешнее программирование, 1160
Нарезание резьбы резцом - CYCLE99
внешнее программирование, 1130
Начало блока программы — GROUP_BEGIN, 1201
не зависящий от регистра, 574
номер D
свободно присвоить, 800
Номер резца, 800
Номера D
переименовать, 801
проверить, 800
Нулевой фрейм, 151
Нулевые точки
при токарной обработке, 176

О

Обработка резаньем
поддерживающие функции, 1051
Обработка резаньем - CYCLE951
внешнее программирование, 1171
Обработка резаньем / Обработка резаньем – остаток / Прорез / Прорез – остаток / Токарная выточка / Токарная выточка – остаток – CYCLE952
внешнее программирование, 1174
Обработчик прерываний
Быстрый отвод от контура, 565
Выключение/включение, 563
Движение отвода, 567
Новое согласование, 563
Программируемое направление перемещения, 567
Удаление, 563
Образец позиции "Линия" – HOLES1
внешнее программирование, 1070
Образец позиции "Окружность или неполная окружность" – HOLES2
внешнее программирование, 1071
Образец позиции "Решетка или рамка" – CYCLE801
внешнее программирование, 1144
Обучение, 24
Общий регламент по защите данных, 26
Ограничение рабочего поля
в BKS, 360
Округление, 460
Оперативная память, 582
Оператор, 47
Операторы сравнения, 457
Определение заготовки, 1045
Ориентация инструмента
относительно траектории, 714
Ориентация инструмента относительно траектории, 714
Ориентируемые инструментальные суппорта, 803
Осевое соединение по главному значению, 952
Осевой
контейнер, 385
оси
Link-, 384
Оси
Lead-Link-ось, 386
PLC-, 384
Буксируемая, 934
Геометрические, 379
Главный-, 379
дополнительные-, 380
Канал-, 382
Командные, 383
Позиционирующие, 382
Синхронные, 383
Станок-, 381
Траектория, 382
Оси вращения
Векторы направления, 803
Векторы расстояния, 803
Угол поворота, 803
Оси ориентации, 702
Оставшееся время
для детали, 1026
Останов
автономный для привода, 1021
управляемый ЧПУ, 1021
Остановка
запрограммированная, 356
на конце цикла, 356
по выбору, 356
Остановка по выбору, 356
Остановка предварительной обработки
внутренняя, 378
Острие фрезы, 782
Ось
-переход, 908
-типы, 378
Отвод
автономный для привода, 1022
-направление при резьбонарезании, 240
управляемый ЧПУ, 1017
Отдельный кадр
-подавление, 520
Открытый паз – CYCLE899
внешнее программирование, 1162
Отрез - CYCLE92
внешнее программирование, 1123

П

память
Оперативная-, 582
Память
Предварительная обработка-, 886
Программная-, 570
Память коррекций, 762
Параметр
-передача при вызове подпрограммы, 512, 544
Фактический-, 512
Формальный-, 512

параметры
Инструмент, 763
Параметры измерительного цикла
CYCLE961, 1224
CYCLE971, 1243
CYCLE973, 1205
CYCLE974, 1207
CYCLE976, 1213
CYCLE977, 1221
CYCLE978, 1215
CYCLE979, 1227
CYCLE982, 1240
CYCLE994, 1210
CYCLE995, 1233
CYCLE996, 1235
CYCLE9960, 1238
CYCLE997, 1230
CYCLE998, 1218
Первичная установка ориентации инструмента ORIRESET, 692
Переключаемые гео-оси, 916
Переменная
Преобразование типов, 446
Переменные
определенные пользователем, 408
Преобразование типа, 462
Переход
на метки перехода, 479
на начало программы, 478
Плоское фрезерование - CYCLE61
внешнее программирование, 1088
Поворот плоскости, поворот инструмента, точная установка инструмента – CYCLE800
внешнее программирование, 1140
Повторение программного блока
с помощью косвенного программирования CALL, 551
Подача
для позиционирующих осей, 129
для синхронных осей, 114
для траекторных осей, 113
Единицы измерения, 115
-коррекция, 132
обратная по времени, 114
Правила, 110
-процентовка, 137
с наложением маховичка, 135
-скорость, 194
Подготовка контура
Квитирование ошибок, 1065
Подпрограмма
-возврат параметрируемый (RET ...), 531
-возврат параметрируемый (RETB...), 538
-вызов без передачи параметров, 542
-вызов косвенный, 550
-вызов модальный, 548
-вызов с передачей параметров, 544
-имя, 509
-повторение, 546
Применение, 508
Программируемый маршрут поиска, 554
Позиционирующие оси, 382
покадровое действие, 49
Поле
-определение, 431
-элемент, 431
Полином-знаменатель, 631
Полиномиальная интерполяция, 627
Полюс, 181
Полярная трансформация, 683
Полярные координаты, 33
Полярный радиус, 33
Полярный угол, 34
Поперечная ось, 177
Постоянная, 393
Постоянная INTEGER, 393
Постоянная REAL, 393
Правая резьба, 231
Правило правой руки, 40
Предварительная обработка
-память, 886
Предварительное сверление контурного кармана - CYCLE64
внешнее программирование, 1093
Представление EES, 575
Представление NCK, 575
Приложение "Siemens Industry Online Support", 25
Присваивание значения, 51
Программа
-"шапка", 57
Адресация, 575
-ветвление, 483
-время выполнения, 1024
-имя, 46
Инициализация, 583
-Конец, 49
-память, 572
-переходы, 479
-повторение, 546
Симуляция, 1029
Программа инициализации, 583
Программа ЧПУ
создать, 55
Программирование диаметра, 170

Программирование линии контура, 219
Программирование окружности
с апертурным углом и центром, 202
с полярными координатами, 204
с радиусом и конечной точкой, 200
с центром и конечной точкой, 198
Типы интерполяции, 197
Программирование ориентации, 702
Программирование радиуса, 170
Программирование ЧПУ
Набор символов, 56
Программная память
Стандартные директории, 571
Типы файлов, 572
Программный цикл
Конечный цикл, 495
Счетный цикл, 495
Цикл IF, 493
Цикл REPEAT, 497
Цикл WHILE, 497
Продольный паз - LONGHOLE
внешнее программирование, 1083
Продольный паз - SLOT1
внешнее программирование, 1078
Пропускаемые кадры, 52
Профилирование - CYCLE495
внешнее программирование, 1135
Процедура
предопределенная, 1324
Прямоугольная цапфа - CYCLE76
внешнее программирование, 1101
Прямоугольный карман - POCKET3
внешнее программирование, 1073
Путь директории, 577

P

Рабочая плоскость, 37
Рабочий цикл
-характеристика управляющих структур, 492
Радиус
эффективный, 116
Развертывание - CYCLE85
внешнее программирование, 1120
Растачивание - CYCLE86
внешнее программирование, 1121
Расширенное написание адреса, 390
Регистрация и поиск непригодных для поиска областей, 892
Режим управления траекторией, 305
Режим ускорения, 870
Резцы
-исходная тока, 300
-номер, 85
положение, 73
-положение, релевантное, 300
-радиус, 73
-центр, 73
-число у контурных инструментов, 296
Резьба
-многозаходная, 230
-направление вращения, 231
-нарезание G33, 229
-нарезание G34 G35, 236
-цепочка, 230
-шаг, 236
Резьбофрезерование - CYCLE70
внешнее программирование, 1095
Референтная точка, 39
Реферирование, 366
Рывок
-коррекция, 903
-ограничение, 870

C

Сверление - CYCLE82
внешнее программирование, 1112
Сверлильное резьбофрезерование - CYCLE78
внешнее программирование, 1105
Сверло, 77
Сглаживание
характеристики ориентации, 723
Символ 0, 462
Симуляция, 1029
Симуляция главного значения, 956
Сингулярные позиции, 701
Синхронное качание
Движение подачи, 1002
Обработка такт IPO, 1004
Определение подач, 1001
Подача в области возврата, 1002
Синхронные действия, 1002
Следующая частичная подача, 1004
Согласование качающейся оси и оси подачи, 1001
Синхронность по положению, 964
Синхронность по положению с угловым смещением, 964
Синхронность по частоте вращения, 964
Синхронные
-оси, 383

Синхронный ход
грубый, 967
точный, 967
Синхронный шпиндель
-определить пару, 969
Соединение, 963
Система единиц, 166
Система координат
Базовая, 42
Обзор, 40
Система координат детали, 44
Система координат станка, 40
Система координат станка (MCS), 40
Система нулевой точки
Базовая, 43
Настраиваемая, 43
Системные фреймы, 674
Скорость движения по траектории
Максимальное значение, 904
Скорость инструмента
макс., 99
Скорость резания (постоянная), 103
Смещение
Длины инструмента, 88
Радиус инструмента, 88
Смещение нулевой точки
Внешнее смещение, 664
Настраиваемый, 150
программируемое, 319
Смещение стартовой точки
при резьбонарезании, 230
Соединение
базовое, 974
строк, 464
Соединение по главному значению
Синхронизация ведущей и ведомой оси, 955
Соединение по фактическому и заданному значению, 955
Соединение по заданному значению, 967
Соединение по скорости, 967
Соединение по фактическому значению, 967
Соединение сплайнов, 625
Сообщения, 357
Сопровождение изделий, 24
Составной размер, 36
Состояние связи
при буксировке, 935
Состояние соединения
при осевом соединении по главному значению, 957
Специальные инструменты, 82
Специальные символы, 56
Спиральная интерполяция, 212
Спиральная резьба, 234
сплайн
-типы, 621
Сплайн
-интерполяция, 614
Стандартный объем, 21
Станки
-Нулевая точка, 38
Станок
-оси, 381
Стартовая точка, 39
Стартовая точка-заданная точка, 178
Строка
-длина, 466
-операции, 461
-соединение, 464
форматировать, 469
Структура буксировок, 932
Структура глав, 20
Сфера действия, 20
Сферическая резьба, 243
Счетный цикл, 495
Съём припуска вдоль контура - CYCLE95
внешнее программирование, 1124

T

Техническая поддержка, 24
Тип кинематики, 807
Тип соединения, 967
Токарные инструменты, 80
Торцовое фрезерование, 699
Точка обработки фрезы, 782
Точка упора, 38
Точка/угол подвода, 272
Точность контура
программируемая, 881
Точный останов, 302
Траекторные оси, 382
Трансформации
Кинематические трансформации, 680
Последовательная связь трансформаций, 680
связанные, 731
Трансформация ориентации, 679
Трех-, четырех- и пятиосевая трансформация, 690
Трансформации координат (фреймы), 44
Трансформация боковой поверхности цилиндра, 684
Трансформация ориентации TRAORI
Базовая 5/6-осевая трансформация, 683

Варианты программирования ориентации, 691
Движения ориентации, 681
Кинематика станка, 682
Программирование ориентации, 691
Трансформация с качающейся линейной осью, 689

У

Угол RPY, 696
Угол контура
закруглить, 254
снять фаску, 254
Угол Эйлера, 696
Указание пути, 576
Указание размеров
в диаметре, 170
в радиусе, 170
Возможности, 157
для круговых осей и шпинделей, 164
Указание составного размера, 159
Управляющие
-структуры, 491
Уровни пропуска, 53
ускорение по траектории
Максимальное значение, 906
Установка системы координат правящего инструмента - CYCLE435
внешнее программирование, 1135

Ф

Файл
-информация, 593
Фаска, 254
Фиксированная точка
Подвод, 367
Формат RS232, 47
Фрезерные инструменты, 75
Фрезерование контурного кармана / Контурный карман – остаточный материал / Фрезерование контурной цапфы / Контурная цапфа – остаточный материал – CYCLE63
внешнее программирование, 1091
Фрезерование траектории - CYCLE72
внешнее программирование, 1097
Фрейм
- масштабирование, 334
вызвать, 659
выключить, 345
-операторы, 316
-отражение, 337
-цепочка, 678
Фрейм-компонент
MI, 658
RT, 658
sc, 658
Фрейм-переменная
Вызов трансформаций координат, 650
Предопределенные фрейм-переменные, 653
Присваивание значений, 656
Фреймы, 314
глобальные, 672
присвоить, 660
Системный-, 674
Спец. для канала, 673
Цепочки фреймов, 661
Функции M, 354
Функции OEM, 646
Функция
предопределенная, 1350

Ц

Центрование - CYCLE81
внешнее программирование, 1111
Цепочка резьб - CYCLE98
внешнее программирование, 1126
Цепочки фильтров заданного значения положения, 883
Цепь детали, 760
Цепь инструмента, 760
Цилиндрическая резьба, 234
Цилиндрические координаты, 183
Цифровое расширение, 390

Ч

Чтение/запись параметров привода, 929

Ш

Шаговая скорость, 97
Шестнадцатеричная постоянная, 393
Шлифовальные инструменты, 78
Шлифование с врезанием с угловой подачей, 729
Шпиндель
Главный-, 381
-направление вращения, 93
-ограничение скорости, 109

-переход, 908
Позиционирование, 123
Функции M, 356
-частота вращения, 93
Штамповка, 1005

Э

Эвольвента, 214
Электронный редуктор, 957
Элемент контура
обработать, 1062

Я

Языковой режим, 1049