

## **СЧПУ MSHAK-CNC**

### **ИНСТРУКЦИЯ ПО ПРОГРАММИРОВАНИЮ**

**Фрезерная группа станков.**

**MSH 0.21.ИП**

(Редакция 4 – для версии 2.x)

**2003 г.  
г. Ереван**

## СОДЕРЖАНИЕ

<b>1. ОСНОВНЫЕ ПОЛОЖЕНИЯ.....</b>	<b>6</b>
1.1 СТРУКТУРА УПРАВЛЯЮЩЕЙ ПРОГРАММЫ (УП).....	6
1.2 НОМЕР КАДРА (N) .....	7
1.3 ПОДГОТОВИТЕЛЬНЫЕ ФУНКЦИИ (G).....	7
1.4 ЗАДАНИЕ ПЕРЕМЕЩЕНИЙ ПО ОСЯМ (X, Y, Z, U, V, W, A, B, C) .....	11
1.5 ЗАДАНИЕ ЦЕНТРА ДУГИ (I, J, K) И РАДИУСА (R) .....	11
1.6 ФУНКЦИЯ СКОРОСТИ ВРАЩЕНИЯ ШПИНДЕЛЯ (S).....	11
1.7 ФУНКЦИЯ ИНСТРУМЕНТА (T).....	12
1.8 НОМЕР КОРРЕКТОРА ИНСТРУМЕНТА (D).....	12
1.9 ФУНКЦИЯ СКОРОСТИ ПОДАЧИ (F).....	13
1.10 ФУНКЦИЯ МНОГОКРАТНОГО ПОВТОРА ОТРЕЗКА ПРОГРАММЫ (H).....	14
1.11 КОММЕНТАРИЙ (;).....	14
1.12 ПРОПУСК КАДРА (/).....	14
<b>2. ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ (M - КОДЫ).....</b>	<b>15</b>
2.1. ПРОГРАММИРУЕМЫЙ ОСТАНОВ (M00).....	15
2.2. ОСТАНОВ С ПОДТВЕРЖДЕНИЕМ (M01) .....	15
2.3. КОНЕЦ ПРОГРАММЫ (M02).....	16
2.4. ВРАЩЕНИЕ ШПИНДЕЛЯ ПО ЧАСОВОЙ СТРЕЛКЕ (M03).....	16
2.5. ВРАЩЕНИЕ ШПИНДЕЛЯ ПРОТИВ ЧАСОВОЙ СТРЕЛКИ (M04) .....	16
2.6. ОСТАНОВ ШПИНДЕЛЯ (M05) .....	16
2.7. СМЕНА ИНСТРУМЕНТА (M06).....	16
2.8. ВКЛЮЧЕНИЕ ОХЛАЖДЕНИЯ (M08) .....	17
2.9. ВЫКЛЮЧЕНИЕ ОХЛАЖДЕНИЯ (M09).....	17
2.10. ВОЗВРАТ ИЗ ПОДПРОГРАММЫ (M17) .....	17
2.11. ПОЗИЦИОНИРОВАНИЕ ШПИНДЕЛЯ (M18) .....	17
2.12. ОРИЕНТАЦИЯ ШПИНДЕЛЯ (M19) .....	18
2.13. КОНЕЦ ПОВТОРЯЮЩЕГОСЯ ОТРЕЗКА ПРОГРАММЫ (M20).....	18
2.14. ОСТАНОВ И ПЕРЕХОД В НАЧАЛО УПРАВЛЯЮЩЕЙ ПРОГРАММЫ (M30) .....	18
<b>3. ПОРЯДОК ПРОГРАММИРОВАНИЯ.....</b>	<b>19</b>
3.1 НАПРАВЛЕНИЕ ОСЕЙ .....	19
3.2 ОСНОВЫ ЗАДАНИЯ ПЕРЕМЕЩЕНИЙ ИНСТРУМЕНТА.....	19
3.3 ЗАДАНИЕ ПЕРЕМЕЩЕНИЙ ПО ОСЯМ .....	20
3.4 ЗАДАНИЕ РЕЖИМА СКОРОСТИ ПОДАЧИ (G93/G94/G95).....	21
3.5. СЛИЯНИЕ ДВИЖЕНИЙ ИНСТРУМЕНТА .....	21
3.5.1. Режим точного останова (G61).....	22
3.5.2. Функция отмены режима точного останова (Режим резания G64).....	22
3.5.3. Торможение (G9).....	23
3.6. ПАУЗА (G4) .....	23
3.7. АБСОЛЮТНЫЙ РАЗМЕР / РАЗМЕР В ПРИРАЩЕНИЯХ (G90/G91).....	23
3.8. ВЫБОР ДЮЙМОВОЙ/МЕТРИЧЕСКОЙ СИСТЕМЫ ЕДИНИЦ (G20/G21).....	24
3.9. ВЫБОР ПЛОСКОСТИ КРУГОВОЙ ИНТЕРПОЛЯЦИИ (G17/G18/G19) .....	24
3.10 ДВИЖЕНИЕ ДО СИГНАЛА ( G31 ) .....	25
3.11 МАСШТАБИРОВАНИЕ КООРДИНАТ (G22/G23/G24).....	26
3.12 ПОВОРОТ СИСТЕМЫ КООРДИНАТ / ОТМЕНА ПОВОРОТА (G68/G69).....	28
3.13 КОНТРОЛЬ СКОРОСТИ ВРАЩЕНИЯ ШПИНДЕЛЯ .....	29
3.13.1 Включение контроля скорости вращения шпинделя (G26) .....	29

3.13.2	Выключения контроля скорости вращения шпинделя (G25) .....	29
3.13.3	Установка максимальной скорости вращения шпинделя (G92).....	29
3.14	ЗЕРКАЛЬНАЯ ОБРАБОТКА (G150/G151) .....	30
<b>4.</b>	<b>ВЫБОР РАЗНОВИДНОСТИ ДВИЖЕНИЯ .....</b>	<b>31</b>
4.1	БЫСТРОЕ ПОЗИЦИОНИРОВАНИЕ (G0) .....	31
4.2	ЛИНЕЙНАЯ ИНТЕРПОЛЯЦИЯ (G1).....	31
4.3	КРУГОВАЯ ИНТЕРПОЛЯЦИЯ. (G2/G3) .....	32
4.3.1	Программирование дуги с помощью I,J,K .....	33
4.3.2	Программирование дуги с помощью R.....	35
4.4	СПИРАЛЬНАЯ ИНТЕРПОЛЯЦИЯ (G2/G3).....	35
<b>5.</b>	<b>СИСТЕМЫ КООРДИНАТ И РЕФЕРЕНТНАЯ ТОЧКА.....</b>	<b>36</b>
5.1	ОСНОВНЫЕ ПОЛОЖЕНИЯ.....	36
5.2	КООРДИНАТНАЯ СИСТЕМА СТАНКА .....	36
5.3	КООРДИНАТНАЯ СИСТЕМА ПРОГРАММЫ .....	36
5.4	ВЫБОР КООРДИНАТНОЙ СИСТЕМЫ .....	37
5.4.1	Выбор локальной системы координат (G52) .....	37
5.4.2	Отмена локальной системы координат (G152) .....	37
5.4.3	Выбор системы координат станка (G53).....	38
5.4.4	Выбор рабочей системы координат (G54-G59).....	38
5.5	РЕФЕРЕНТНАЯ ТОЧКА .....	39
5.5.1	Выход в референтную точку (G28) .....	39
5.5.2	Возврат из референтной точки (G29) .....	40
5.5.3	Выход ко 2-ой - 4-ой референтным точкам (G30).....	41
<b>6.</b>	<b>КОРРЕКЦИЯ НА РАДИУС ИНСТРУМЕНТА.....</b>	<b>42</b>
6.1	КОРРЕКЦИЯ НА РАДИУС ИНСТРУМЕНТА: ОТМЕНА/СЛЕВА/СПРАВА (G40/G41/G42) .....	42
6.2	СКОРОСТЬ КОРРЕКТИРУЮЩЕГО ДВИЖЕНИЯ .....	45
6.3	АКТИВИЗАЦИЯ РЕЖИМА КОРРЕКЦИИ .....	45
6.4	ДВИЖЕНИЕ ПРИ АКТИВНОМ РЕЖИМЕ КРИ .....	49
6.4.1	Обработка внутреннего угла .....	49
6.4.2	Обработка внешнего угла.....	49
6.4.3	Изменение направления коррекции.....	49
6.5	ОТМЕНА КОРРЕКЦИИ .....	55
<b>7.</b>	<b>КОРРЕКЦИЯ НА ПОЛОЖЕНИЕ ИНСТРУМЕНТА.....</b>	<b>61</b>
7.1	КОРРЕКЦИЯ НА ПОЛОЖЕНИЕ ИНСТРУМЕНТА +/- И ОТМЕНА (G43/G44/G49) ..	61
<b>8.</b>	<b>ПОСТОЯННЫЕ ЦИКЛЫ СВЕРЛЕНИЯ И РАСТОЧКИ. ....</b>	<b>62</b>
8.1.	ОТМЕНА ПОСТОЯННОГО ЦИКЛА G81-G86 (G80) .....	62
8.2.	ИГНОРИРОВАТЬ ПОСЛЕДУЮЩИЙ ПОСТОЯННЫЙ ЦИКЛ G81-G88 (G180).....	63
8.3.	ВОЗВРАТ ИЗ ПОСТОЯННОГО ЦИКЛА (G98/G99) .....	63
8.4.	СВЕРЛЕНИЕ С ВЫВОДОМ ИНСТРУМЕНТА НА БЫСТРОМ ХОДУ (G81).....	64
8.5.	РАСТАЧИВАНИЕ (СВЕРЛЕНИЕ) С ПАУЗОЙ ПЕРЕД ВЫВОДОМ (G82) .....	66
8.6.	СВЕРЛЕНИЕ С ПЕРИОДИЧЕСКИМ ВЫВОДОМ ИНСТРУМЕНТА (G83) .....	67
8.7.	СВЕРЛЕНИЕ С ПЕРИОДИЧЕСКИМ ОТСКОКОМ ИНСТРУМЕНТА (G183).....	70
8.8.	НАРЕЗАНИЕ РЕЗЬБЫ С ОТВОДОМ НА РАБОЧЕЙ ПОДАЧЕ И РЕВЕРСЕ (G84) .....	72
8.9.	РАСТАЧИВАНИЕ С ОТВОДОМ НА РАБОЧЕЙ ПОДАЧЕ БЕЗ РЕВЕРСА (G85) .....	74
8.10.	РАСТАЧИВАНИЕ С БЫСТРЫМ ОТВОДОМ И ОСТАНОВОМ ШПИНДЕЛЯ (G86) ...	75

<b>9.</b>	<b>МНОГООПЕРАЦИОННЫЕ ЦИКЛЫ.....</b>	<b>77</b>
9.1	ОБРАБОТКА ОТВЕРСТИЙ ПО КОНТУРУ ОКРУЖНОСТИ (G700).....	77
9.2	ОБРАБОТКА ОТВЕРСТИЙ ПО КОНТУРУ ДУГИ ОКРУЖНОСТИ (G701).....	78
9.3	ОБРАБОТКА ОТВЕРСТИЙ ПО ПРЯМОЛИНЕЙНОМУ КОНТУРУ (G702).....	79
<b>10.</b>	<b>ПАРАМЕТРИЧЕСКОЕ ПРОГРАММИРОВАНИЕ .....</b>	<b>80</b>
10.1	НАЗНАЧЕНИЕ.....	80
10.2	ВЫЗОВЫ ПОДПРОГРАММ .....	80
10.2.1	<i>Вызов подпрограмм с параметрами.....</i>	<i>80</i>
10.2.2	<i>Простой вызов подпрограмм.....</i>	<i>83</i>
10.2.3	<i>Программирование повторяющегося отрезка программы (H) .....</i>	<i>84</i>
10.3	ПЕРЕМЕННЫЕ (#).....	85
10.3.1	<i>Задание значения адреса совместно с выражением.....</i>	<i>87</i>
10.3.2	<i>Задание значения адресного слова как значения выражения.....</i>	<i>87</i>
10.4	ПЕРЕМЕННАЯ #0 - “НЕОПРЕДЕЛЕННО” .....	87
10.5	ЛОКАЛЬНЫЕ ПЕРЕМЕННЫЕ .....	88
10.6	ОБЩИЕ ПЕРЕМЕННЫЕ .....	90
10.7	СИСТЕМНЫЕ ПЕРЕМЕННЫЕ .....	90
10.7.1	<i>Корректоры инструментов #2000 - #2800, #6000-#7400.....</i>	<i>91</i>
10.7.2	<i>Аварийное прерывание выполнения УП с сообщением #3000.....</i>	<i>92</i>
10.7.3	<i>Системные таймеры #3001 - #3002 .....</i>	<i>92</i>
10.7.4	<i>Останов выполнения УП с сообщением #3006 .....</i>	<i>93</i>
10.7.5	<i>Модальная информация о G функциях в группе #4001 – #4020.....</i>	<i>93</i>
10.7.6	<i>Информация об адресных словах #4101 – #4126 .....</i>	<i>94</i>
10.7.7	<i>Командная координата, с учетом корректора инструмента #5001 – #500n</i>	<i>95</i>
10.7.8	<i>Текущая машинная координата #5021-#502n.....</i>	<i>96</i>
10.7.9	<i>Текущая координата в рабочей координатной системе #5041-#504n</i>	<i>96</i>
10.7.10	<i>Координата контакта с измерительным щупом #5061-#506n.....</i>	<i>97</i>
10.7.11	<i>Признак события - контакт с измерительным щупом #5070.....</i>	<i>97</i>
10.7.12	<i>Признак активности измерительного щупа в шпинделе #5071.....</i>	<i>97</i>
10.7.13	<i>Признак активности измерительного щупа на столе #5072 .....</i>	<i>97</i>
10.7.14	<i>Значения корректора радиуса текущего инструмента #5080.....</i>	<i>97</i>
10.7.15	<i>Значения корректоров текущего инструмента #5081-#508n .....</i>	<i>98</i>
10.7.16	<i>Значения корректора износа радиуса текущего инструмента #5090</i>	<i>98</i>
10.7.17	<i>Значения корректоров износа текущего инструмента #5091-#509n</i>	<i>98</i>
10.7.18	<i>Смещения текущей координатной системы #5201-#520n .....</i>	<i>98</i>
10.7.19	<i>Смещения координатной системы G54 #5221-#522n .....</i>	<i>99</i>
10.7.20	<i>Смещения координатной системы G55 #5241-#524n .....</i>	<i>99</i>
10.7.21	<i>Смещения координатной системы G56 #5261-#526n .....</i>	<i>99</i>
10.7.22	<i>Смещения координатной системы G57 #5281-#528n .....</i>	<i>99</i>
10.7.23	<i>Смещения координатной системы G58 #5301-#530n .....</i>	<i>99</i>
10.7.24	<i>Смещения координатной системы G59 #5321-#532n .....</i>	<i>99</i>
10.8	ФОРМАТИРОВАННЫЙ ВЫВОД.....	99
10.9	ВЫРАЖЕНИЯ.....	102
10.9.1	<i>Предложение присвоения.....</i>	<i>104</i>
10.9.2	<i>Адресное слово.....</i>	<i>105</i>
10.9.3	<i>Условное выражение .....</i>	<i>105</i>

---

10.9.4	Переход .....	106
10.9.4.1	Безусловный переход.....	106
10.9.4.2	Условный переход .....	107
10.10	МАКРООПРЕДЕЛЕНИЯ .....	108
10.11	ОРГАНИЗАЦИЯ БИБЛИОТЕКИ ВНЕШНИХ ПОДПРОГРАММ (ЦИКЛОВ).....	109
10.11.1	<i>Вызов внешних подпрограмм с помощью подготовительной функции G.</i> 109	
10.11.2	<i>Создание библиотеки подпрограмм.....</i>	110
10.11.3	<i>Визуализация цикла и интерактивный ввод параметров подпрограмм.....</i>	112
<b>11.</b>	<b>ПРИЛОЖЕНИЕ 1 - ПРИМЕР ПРИМЕНЕНИЯ ПАРАМЕТРИЧЕСКОЙ ПОДПРОГРАММЫ.....</b>	<b>113</b>

## 1. ОСНОВНЫЕ ПОЛОЖЕНИЯ

### 1.1 Структура управляющей программы (УП)

Для обработки детали на станке с СЧПУ необходимо задать траекторию перемещения инструмента и другие условия обработки. Данную программу называют программой обработки детали или управляющей программой (УП).

Структурную единицу УП составляет кадр. Кадр является последовательностью символов языка программирования.

Элементом кадра является слово, которое состоит из адреса и числового значения или переменной. Адрес представляет собой одну из алфавитных букв (А - Z) и определяет смысл последующего числового значения. Назначение адресов в УП приведено в табл. 1.

При задании числового параметра незначащие нули можно опускать. Числовой параметр адреса, если он не целое число, может иметь десятичную точку. Допускается не задавать десятичную точку, если после нее нет значащего числа;

**Пример:**

**Z - 123.23**  
Адрес Числовое значение  
**X #101**  
Адрес переменная

**Примечание:**

Порядок работы адресных слов с переменными приведено в разделах 10.3 и 10.9 данного руководства

*Таблица 1.*

Адрес	Назначение
<b>N</b>	Номер кадра
<b>G</b>	Подготовительная функция
<b>M</b>	Вспомогательная функция
<b>T</b>	Номер инструмента
<b>D</b>	Номер корректоров смещений и радиуса инструмента
<b>S</b>	Задание оборотов шпинделя
<b>F</b>	Задание подачи, шаг резьбы
<b>X, Y, Z</b>	Перемещения по линейным осям
<b>U, V, W</b>	Перемещения по дополнительным линейным осям
<b>A, B, C</b>	Перемещения по круговым осям
<b>I, J, K</b>	Параметры круговой интерполяции по осям X, Y, Z
<b>R</b>	Радиус круговой интерполяции
<b>H</b>	Повторение отрезка программы
<b>L</b>	Выполнение участка программы или вызов подпрограммы
<b>E</b>	Условный или безусловный переход
<b>#</b>	Переменная

**Примечание:**

В постоянных циклах назначение адресов может не совпадать с назначениями, приведенными в табл. 1.

Наряду с адресами в кадре можно использовать следующие символы:

Символ	Назначение
/	Пропуск кадра
;	Комментарий
–	Отрицательное значение числового параметра

**1.2 Номер кадра (N)**

При составлении УП, кадры можно отмечать при помощи адресного слова **Nxxx**, где xxx – это любое положительное целое число.

Адресное слово **N** не является номером кадра в традиционном его применении, а является меткой кадра, используемого для выполнения переходов и вызовов подпрограмм. Номера отмеченных кадров в УП не должны повторяться.

Необязательно применение номера кадра в кадре, однако при его использовании номер кадра является первым словом в кадре.

При составлении УП можно отмечать только кадры, которые необходимы для выполнения переходов, а нумерация кадров может быть произвольной.

**Примечание:**

Не рекомендуется выполнять сквозную нумерацию всех кадров в УП, где количество кадров больше 10000, так как будет происходить резкое увеличение времени загрузки текста УП в буфер обработки - с нескольких секунд до нескольких минут. Это объясняется тем, что СЧПУ выполняет предварительный анализ текста УП.

**1.3 Подготовительные функции (G)**

Формат задания подготовительной функции (или **G**-коды) следующий:

**Gnnn**

Где: nnn- трехзначное число номер **G функции**.

Незначащие нули можно не задавать.

Подготовительные функции разбиты на группы. Описание групп подготовительных функций приведено в табл. 2. По умолчанию действующие **G функции** отмечены жирным шрифтом. Активация одной из подготовительных функций в любой из групп, кроме нулевой, приводит к автоматической отмене другой, ранее активированной или действующей по умолчанию функции. Активированная подготовительная функция действует до конца работы или до активации другой подготовительной функции данной группы. Исключение составляет группа 0. Функции этой группы действуют только в пределах кадра.

**Группа 0. Действующие в одном кадре**
*Таблица 2.*

G4	Пауза
G9	Точный останов в кадре
G28	Выход в референтную точку
G29	Возврат из референтной точки
G30	Выход ко 2-ой или 3-ей или 4-ой референтной точке
G31	Движение до касания
G33	Нарезание резьбы с постоянным шагом
G53	Выбор системы координат станка
G92	Установка максимальной скорости вращения шпинделя
G74	Многопроходный цикл нарезания торцовых канавок
G75	Многопроходный цикл нарезания цилиндрических канавок
G76	Многопроходный цикл нарезания резьбы
G77	Однопроходный цикл продольного точения
G78	Однопроходный цикл нарезания резьбы
G79	Однопроходный цикл поперечного точения
G171	Глубокое сверление
G172	Глубокое поперечное точение
G180	Отмена первого отверстия в циклах G81-G86, G183
G700	Обработка отверстий по контуру окружности
G701	Обработка отверстий по контуру дуги окружности
G702	Обработка отверстий по прямолинейному контуру

**Группа 1. Выбор разновидности движения**

<b>G0</b>	<b>Быстрое позиционирование</b>
G1	Линейная интерполяция
G2	Круговая интерполяция. Движение по часовой стрелке Спиральная интерполяция
G3	Круговая интерполяция. Движение против часовой стрелки Спиральная интерполяция

**Группа 2. Выбор плоскости**

<b>G17</b>	<b>Выбор плоскости XY</b>
G18	Выбор плоскости ZX
G19	Выбор плоскости ZY

**Группа 3. Размерность задания перемещений**

G20	Выбор дюймовой системы единиц
<b>G21</b>	<b>Выбор метрической системы единиц</b>

**Группа 4. Масштабирование системы координат**

G22	Увеличить масштаб
<b>G23</b>	<b>Отменить масштабирование</b>
G24	Уменьшить масштаб



**Группа 5. Контроль вращения шпинделя**

G25	Выключение контроля вращения скорости шпинделя
<b>G26</b>	<b>Включение контроля вращения скорости шпинделя</b>

**Группа 6. Коррекция на радиус инструмента**

<b>G40</b>	<b>Отмена коррекции на радиус инструмента</b>
G41	Коррекция на радиус инструмента слева
G42	Коррекция на радиус инструмента справа

**Группа 7. Коррекция на положение (длину) инструмента**

G43	Коррекция на длину инструмента – положительная
G44	Коррекция на длину инструмента – отрицательная
<b>G49</b>	<b>Отмена коррекции на положение инструмента</b>

**Группа 8. Локальная координатная система**

G52	Установка локальной системы координат
G152	Отмена локальной системы координат

**Группа 9. Рабочая координатная система (Задания смещений)**

<b>G54</b>	<b>Задание смещения 1–ой рабочей системы координат</b>
G55	Задание смещения 2–ой рабочей системы координат
G56	Задание смещения 3–ой рабочей системы координат
G57	Задание смещения 4–ой рабочей системы координат
G58	Задание смещения 5–ой рабочей системы координат
G59	Задание смещения 6–ой рабочей системы координат

**Группа 10. Режимы движения со слиянием**

G61	Режим точного останова
<b>G64</b>	<b>Режим резания (отмена режима точного останова)</b>

**Группа 11. Режимы задания размера диаметра/радиуса детали**

<b>G62</b>	<b>Режим задания размера по диаметру детали</b>
G63	Режим задания размера по радиусу детали

**Группа 12. Поворот системы координат**

G68	Поворот системы координат
<b>G69</b>	<b>Отмена поворота системы координат</b>

**Группа 13. Многопроходные токарные циклы**

G70	Чистовой замкнутый цикл
G170	Конец программы черногового многопроходного цикла
G71	Многопроходный черновой продольный замкнутый цикл
G72	Многопроходный черновой поперечный замкнутый цикл
G73	Многопроходный черновой копировальный цикл

**Группа 14. Сверлильно-расточные фрезерные циклы**

<b>G80</b>	<b>Отмена постоянного цикла</b>
G180	Не выполнять обработку в последующих постоянных циклах(G81-G88)
G81	Сверление с выводом инструмента на быстром ходу
G82	Растачивание (сверление) с паузой перед выводом
G83	Глубокое сверление с выводом сверла
G84	Нарезание резьбы с отводом на рабочей подаче и реверсе
G85	Растачивание с отводом на рабочей подаче без реверса
G86	Растачивание с быстрым отводом и остановом шпинделя
G183	Глубокое сверление со стружкодроблением

**Группа 15. Режим задания перемещений**

<b>G90</b>	<b>Абсолютный размер</b>
G91	Размер в приращениях

**Группа 16. Режим задания скорости подачи**

G93	Задание подачи по времени перемещения
<b>G94</b>	<b>Скорость подачи в мин.</b>
G95	Скорость подачи на оборот шпинделя

**Группа 17. Режим задания вращения шпинделя**

G96	Установка режима постоянной скорости резания
<b>G97</b>	<b>Отмена постоянной скорости резания</b>

**Группа 18. Точка возврата в сверлильных циклах**

<b>G98</b>	<b>Возврат в начальную точку постоянного цикла</b>
G99	Возврат в референтную точку из постоянного цикла

**Группа 19. Зеркальная обработка**

<b>G150</b>	<b>Отмена режима зеркальной обработки</b>
G151	Режим зеркальной обработки

**Группа 20. Режим стружкодробление**

G160	Стружкодробление по времени
G161	Стружкодробление по положению шпинделя
<b>G162</b>	<b>Отмена режима стружкодробление</b>

**Примечание:**

*Выделенные подготовительные функции работают в токарной версии ПО MSHAK-CNC. В фрезерной версии ПО можно эти функции сделать активными (разрешенными для использования). Описание работы токарных функций следует смотреть в документе “Инструкция по программированию. Токарная группа станков. MSH 0.20.ИП.” Установка разрешения/запрета работы этих функций выполняется изготовителем СЧПУ или станка.*



### 1.7 Функция инструмента (T)

Функция инструмента с адресом **T** используется для предварительного поиска инструмента в ячейках магазина и дополнительно, в зависимости от станка, установки инструмента в автооператор (манипулятор). Следующее за адресом **T** число определяет номер инструмента.

Формат:

**Tnnn**

Где: **nnn** – номер инструмента, принимает значение от 1 до 200

Для смены инструмента по программе используется вспомогательная функция **M6**.

#### Пример:

T5 ; начало поиска инструмента 5 в магазине и перевод в позицию  
; смены  
X50 Z60 ; продолжение отработки программы  
...  
M6 ; ожидание завершения поиска и смена инструмента

### 1.8 Номер корректора инструмента (D)

Адрес **D** определяет номер корректора, в котором заданы следующие параметры:

- 1 величины смещения инструмента по каждой координате, которые определены в СЧПУ
- 2 величина износа инструмента по каждой координате
- 3 величина радиуса инструмента
- 4 величина износа радиуса инструмента

Формат:

**Dnnn**

Где: **nnn** – номер корректора, принимает значение от 1 до 200

Эти параметры вводятся и редактируются на странице **Tool Offset** (Смещение Инструмента) программы **MSHAK-CNC** (см. Документ “СЧПУ MSHAK-CNC Руководство оператора”).

На фрезерных станках, как правило, используется коррекция на радиус и только смещение инструмента по оси **Z** - для компенсации длины инструмента, а параметры по остальным осям (X,Y) равны нулю.

Активизация смещения инструмента по осям выполняется функциями **G43,G44**. Отмена смещения инструмента по осям выполняется функцией **G49**.

Активизация смещения инструмента по радиусу выполняется функциями **G41,G42**. Отмена смещения инструмента по радиусу выполняется функцией **G40**.

В общем случае номер корректора не связан с номером инструмента. Но для удобства программирования, СЧПУ по умолчанию, после функции **M6** заданный корректор становится равным номеру инструмента. Однако в УП можно вне зависимости от номера инструмента при помощи **D** задать любой номер корректора.

Номер корректора **D** можно задать в любом кадре УП, или отдельным кадром, или в кадрах где есть функции **G41, G42,G43, G44**.

При задании **D**, если не активны ни одна из функций G41-G44, СЧПУ запоминает заданный номер корректора **D**, который будет в дальнейшем использоваться в этих функциях.

При задании нового **D**, если активна функция **G43** или **G44**, СЧПУ отменяет смещения по осям предыдущего **D**, и активизирует смещения нового.

**Пример:**

D5 ; задан корректор N 5  
G43 Z-30 ; активизация смещения по корректору N 5  
X30  
D7 ; активизация смещения по корректору N 7  
Z-50  
G49 ; отмена смещения по осям  
...  
G43 D4 ; активизация смещения по корректору N 4  
...

При задании нового **D**, если активна функция **G41** или **G42**, СЧПУ **не отменяет** смещения по радиусу для предыдущего номера **D**, но запоминает номер нового **D**, Он будет активизирован после отмены функцией **G40** и нового задания **G41** или **G42**.

**Пример:**

D2 ; задан корректор N 2  
G41 X30 ; активизация смещения по корректору N 2  
Y30  
X50  
D4 ; запоминание корректора N 4  
Y10  
X30  
G40Y0 ; отмена смещения по радиусу корректора N 2  
...  
G41 ; активизация смещения по корректору N 4  
...

**Внимание:**

В программе **MSHAK-CNC** до версии 2.0, при задании номера корректора, для компенсации длины инструмента использовался адрес **H**. А для компенсации радиуса инструмента использовался адрес **D**.

В версии 2.0 и выше в обоих случаях используется только адрес **D**. Поэтому, для пользователей предыдущих версий, следует в УП сделать изменения – заменить в кадрах **G43H\_** на **G43D\_**.

**Примечание:**

Величины смещений и радиуса, по запрограммированным номерам корректоров, следует вводить в ручном режиме до начала обработки УП.

### **1.9 Функция скорости подачи (F)**

Функция скорости подачи использует адрес **F**, за которым следует число, указывающее на скорость подачи при обработке. Со словом **F** можно задать и переменную #.(См. Раздел 10.3.)

Единица измерения скорости подачи может быть трех типов: время выполнения движения в секундах, перемещение в минуту и перемещение на оборот шпинделя. Выбор единицы измерения производится функциями **G93**, **G94** и **G95** соответственно. (См. Раздел 3.3.)

**Пример:**

#5=200  
F#5 ; подача 200 мм/мин  
F300 ; задание скорости резания 300мм/мин  
F300+#5 ; задание скорости резания 500мм/мин :300+#5=500

### 1.10 Функция многократного повтора отрезка программы (Н)

Многократный повтор отрезка управляющей программы задается с помощью адреса **Н** и вспомогательной функции **M20**.

Формат:

Нxxx,

Где: xxx -количество повторов отрезка УП

Значение адреса показывает количество повторов отрезка программы. Повторяющийся отрезок начинается с кадра, следующего за кадром **Н**. Конец отрезка программируется вспомогательной функцией **M20**.(подробнее смотри раздел 10.2.3)

### 1.11 Комментарий (;)

В программе можно использовать комментарии. Все символы, следующие в строке за знаком “;” игнорируются.

**Пример:**

X10 ; Это комментарий  
X15 ; Это тоже комментарий

### 1.12 Пропуск кадра (/)

В УП можно использовать специальным образом отмеченные кадры с помощью символа “/”.

При указании номеров кадров символ “/” ставится или после адреса Nxxxx, или вместо него.

**Пример:**

/X10 ; правильно  
N2/ X20 ; правильно  
/3 X30 ; неправильно  
/ N4 X30 ; неправильно

При выполнении УП программы, если выбран режим “**Пропуск отмеченных кадров**” программы MSHAK-CNC, то отмеченные кадры УП пропускаются.

При обычном режиме работы эти кадры выполняются.

## 2. ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ (М - КОДЫ)

Вспомогательные функции (или М-коды) программируются с помощью адресного слова М. Вспомогательные функции используются для управления программой и электроавтоматикой станка - включения/выключения шпинделя, охлаждающей жидкости, смены инструмента и др.

Таблица 3.

Обозначение	Назначение
<b>M00</b>	Программируемый останов
<b>M01</b>	Останов с подтверждением
<b>M02</b>	Конец программы
<b>M03</b>	Вращение шпинделя по часовой стрелке
<b>M04</b>	Вращение шпинделя против часовой стрелки
<b>M05</b>	Останов шпинделя
<b>M06</b>	Смена инструмента
<b>M08</b>	Включение охлаждения
<b>M09</b>	Отключение охлаждения
<b>M17</b>	Возврат из подпрограммы
<b>M18</b>	Позиционирование шпинделя на заданный угол
<b>M19</b>	Ориентация шпинделя
<b>M20</b>	Конец повторяющегося отрезка программы
<b>M30</b>	Останов и переход в начало управляющей программы
<b>M99</b>	Продолжить выполнение УП первого кадра

Вспомогательные функции, которые выполняют включение каких-либо операций (**M03**, **M04** и **M08**), выполняются в начале кадра перед командами движения. Остальные вспомогательные функции выполняются в конце кадра.

В табл. 3 приведен список обычно используемых вспомогательных функций.

### 2.1. Программируемый останов (M00)

Безусловный останов управляющей программы после выполнения движения содержащейся в текущем кадре. Состояние УП не изменяется до повторного нажатия кнопки **START** на пульте управления СЧПУ или клавиши **В НАЧАЛО**, для возврата к началу выполняемой УП.

### 2.2. Останов с подтверждением (M01)

Останов управляющей программы после выполнения движения содержащейся в текущем кадре, при условии, что установлен режим **“Останов с подтверждением”** с панели управления СЧПУ (см. Документ СЧПУ MSHAK-CNC Руководство оператора).

Пример:

X-2

X-4.

M1

; Останов выполнения программы на этом кадре, если

; задан режим **“Останов с подтверждением”** с пульта оператора

X5

### 2.3. Конец программы (M02)

Определяет конец выполнения управляющей программы, прекращает подачу охлаждающей жидкости и останавливает вращение шпинделя.

**Пример:**

```
G0X20Z50  
Z.5  
G0  
X0Z0  
M2
```

### 2.4. Вращение шпинделя по часовой стрелке (M03)

Запускает вращение шпинделя по часовой стрелке, используя текущее значение, заданное словом S.

**Пример:**

```
G54 G0 X-20 Z30  
S500M3
```

### 2.5. Вращение шпинделя против часовой стрелки (M04)

Запускает вращение шпинделя против часовой стрелки, используя текущее значение, заданное словом S.

**Пример:**

```
G54 G0 X-20 Z30  
S1500M4
```

### 2.6. Останов шпинделя (M05)

Останавливает вращение шпинделя. Выполняется после движений, содержащихся в кадре.

**Пример:**

```
G28 X0 Z0  
M5  
G4 P2  
M2
```

### 2.7. Смена инструмента (M06)

Выполняет смену инструмента между шпинделем и магазином инструментов.

По этой функции происходит:

- Позиционирование по осям в точку смены инструмента;
- Останов вращения шпинделя и ориентация шпинделя;
- Смена инструмента.

**Пример:**

```
T5 ; начало поиска инструмента 5 в магазине  
X50 Z60 ; продолжение отработки программы  
...  
M6 ; смена инструмента
```



**2.8. Включение охлаждения (M08)**

Включает подачу смазочно-охлаждающей жидкости (СОЖ).

**Пример:**

S300M3X20Z30G0  
G1X50Z44M8 ; Включить СОЖ  
G0Z-100

**2.9. Выключение охлаждения (M09)**

Выключает подачу смазочно-охлаждающей жидкости (СОЖ).

**Пример:**

S300M3X20Z30G0  
G1X50Z44  
M9M5G0Z-100

**2.10. Возврат из подпрограммы (M17)**

Определяет конец подпрограммы при ее вызове словом с адресом **L**.

**Пример:**

X5Z5 ; Основная программа  
L10 ; Вызов подпрограммы начинающейся с кадра N10  
X2Z8  
M2  
  
N10Z2 ; Подпрограмма с меткой кадра N10  
X10  
M17 ; Конец подпрограммы и возврат в основную программу

**2.11. Позиционирование шпинделя (M18)**

При помощи этой функции можно повернуть шпиндель на заданный угол

Формат:

**M18 Pnnn**

Где : nnn – угол поворота +/- 360 градусов.

Отсчет угла поворота выполняется относительно позиции шпинделя, на которую устанавливается шпиндель по функции M19.

**Пример:**

M18 P45 ; поворот шпинделя на 45 градусов

**2.12. Ориентация шпинделя (M19)**

Вспомогательная функция **M19** останавливает вращение шпинделя, выполняет его ориентацию.

**2.13. Конец повторяющегося отрезка программы (M20)**

Определяет конец повторяющегося отрезка программы при ее вызове словом с адресом **N**.

**Пример:**

```
N10 H2           ; выполнить отрезок программы до M20  2 раза
Z2X1
X1
M20             ; конец отрезка программы для повтора
```

**2.14. Останов и переход в начало управляющей программы (M30)**

Останавливает выполнение управляющей программы как **M2**, и возвращает указатель в начало программы.

**Пример:**

```
G1Z5
G0
X0Z0
M30
```

### 3. ПОРЯДОК ПРОГРАММИРОВАНИЯ

При обработке заготовки, в зависимости от конструкции станка, или инструмент может двигаться относительно обрабатываемой детали, или же заготовка (стол) двигаться относительно неподвижного инструмента. Движения инструмента выполняются по прямой линии или по дуге окружности в пределах допустимых границ координатных перемещений.

В этой инструкции при описании функций перемещений предполагается, что инструмент движется относительно заготовки.

#### 3.1 Направление осей

Координатные управляемые линейные оси  $X, Y, Z$  всегда создают правую трехмерную систему координат, в которой задаются позиции инструмента. При программировании траектории обработки всегда следует задавать перемещение инструмента и не следует беспокоиться о том, как будут выполняться физические перемещения узлов станка. СЧПУ автоматически скорректирует движение так, чтобы инструмент оказался в запрограммированной позиции заданной системы координат.

Приведенные в этом руководстве описания  $G$ -функций предполагают правую координатную систему в которой:

- ось  $Z$  всегда располагается параллельно оси шпинделя.

За положительное направление (из "-" в "+") оси  $Z$  принимается направление, которое приводит к увеличению расстояния между деталью и инструментом;

- ось  $X$ , как правило, задает продольное перемещение вдоль стола станка.

Ось  $Y$  задает поперечное перемещение вдоль стола станка. Оси  $X$  и  $Y$  расположены таким образом:

$X$  - вдоль стола станка

$Y$  - перпендикулярно  $X$  так, чтобы при повороте от положительного направления  $X$  к положительному направлению  $Y$  правый винт перемещался в положительном направлении  $Z$ ;

- дополнительные круговые оси  $A, B, C$  являются вращающимися координатными осями вокруг линейных осей  $X, Y, Z$  соответственно.

За положительное направление (из "-" в "+") поворотной оси принимается вращение по часовой стрелке, если смотреть на вращающуюся ось в положительном направлении для соответствующей ей линейной оси.

- дополнительные линейные оси  $U, V, W$  являются параллельными к осям  $X, Y, Z$ . Положительные и отрицательные направления дополнительных осей совпадают с направлением основных.

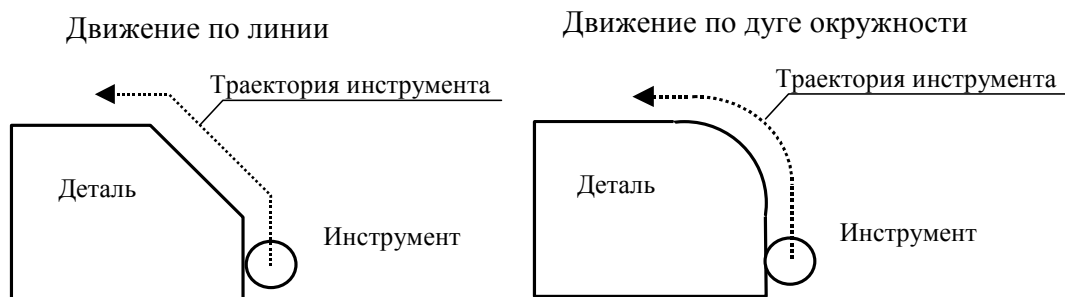
#### Примечание

*Станки, предназначенные для специальных видов обработки, могут иметь системы координат, построенные по иным правилам.*

#### 3.2 Основы задания перемещений инструмента

При обработке заготовки, инструмент движется относительно обрабатываемой детали по прямой линии или по дуге окружности внутри допустимых границ

координатных перемещений. В этом руководстве при описании функций предполагается, что инструмент движется относительно заготовки. Функция движения инструмента вдоль прямой линии и дуги, заданная начальными и конечными координатами, называется интерполяцией (см. рис 1).



**Рис. 1** Прямолинейное и круговое движение инструмента

Различают три основные функции интерполяции:

Движение инструмента по линии (**G01**) с заданной подачей или на быстрой скорости (**G00**);

Движение инструмента по дуге окружности по часовой стрелке (**G02**);

Движение инструмента по дуге окружности против часовой стрелки (**G03**).

Управление движением по осям выполняется скоординировано для точного получения требуемых перемещений.

### 3.3 Задание перемещений по осям

Позиция, заданная в предшествующем кадре, является начальной точкой, а позиция в текущем кадре - конечной точкой движения. Конечная позиция движения может быть как абсолютной (**G90**) (координаты точки в выбранной системе координат), так и относительной (**G91**) (приращение со знаком к значению текущей позиции).

Для задания конечной точки движения используется слово, состоящее из адреса **X**, **Y** или **Z** и числа с десятичной точкой.

Кроме трех основных осей могут быть заданы также дополнительные оси **A**, **B**, **C**, **U**, **V**, **W**.

Десятичную точку можно не задавать, если нет цифр после нее. В общем случае при задании координат, в этом документе будет использоваться типизированная форма **X\_Y\_Z\_** - где знак “\_” заменяет задание любого числа.

Движение задается словами с адресами позиций по осям с числовыми значениями. Единица длины устанавливается программно с помощью подготовительных функций группы **3**: **G20** – выбор дюймовой системы (задание перемещений в дюймах) и **G21** – выбор метрической системы (задание перемещений в мм).

Числовые значения подразумевают единицу длины (мм или дюйм).

#### Пример:

X5.2 Y0

X10 Y12.3 Z15.025

### 3.4. Задание режима скорости подачи (G93/G94/G95)

Заданная скорость движения инструмента относительно заготовки при обработке называется скоростью подачи (см. рис. 2).

При движениях задаваемых подготовительными функциями **G1**, **G2**, **G3**, скорость подачи задается в программе с помощью адресного слова **F**. Подготовительные функции **G93**, **G94**, **G95** являются взаимоисключающими в группе **16**. В каждой из них число, следующее за адресом **F**, указывает на величину подачи различным способом.

При **G94** указывается расстояние в единицах длины, которое необходимо пройти за одну минуту;

При **G95** указывается расстояние в единицах длины, которое необходимо пройти за один оборот шпинделя;

При **G93** указывается время в секундах, за которое необходимо выполнить движение в кадре.

Подготовительная функция **G94** устанавливается по умолчанию.

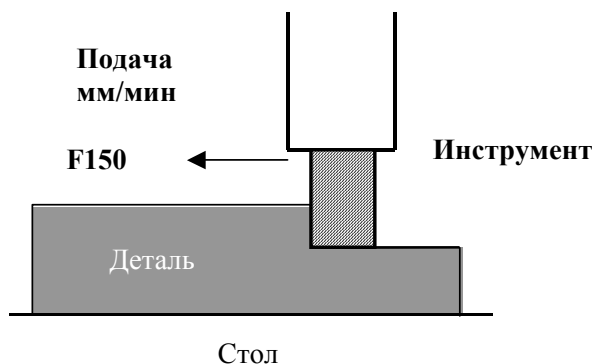


Рис. 2 Скорость подачи в мм/мин

Кроме перемещений с программируемой подачей, можно выполнять перемещения на быстрых скоростях, задаваемых подготовительной функцией **G0**. Скорость этих перемещений задается параметрами системы для каждой оси по отдельности. Таким образом, она в программе не указывается.

Заданную скорость подачи можно оперативно изменять процентом переопределения на панели управления оператора (см. Документ СЧПУ MSHAK-CNC Руководство оператора).

### 3.5. Слияние движений инструмента

В условиях многокоординатного движения при смене направления движения происходит слияние движений (см. рис. 3), что приводит к сглаживанию углов. При необходимости создания острого угла рекомендуется использование функции останова в конце кадра или паузы в кадре (**G61,G9,G04**). Эти подготовительные функции приводят к останову с точным позиционированием перед автоматическим продолжением следующего движения.

Состояние точного останова означает, что ось движения находится в положении, отличающемся от позиции заданной по программе на величину, не превышающую величину зоны позиционирования. Эта величина устанавливается системными параметрами СЧПУ для каждой оси в отдельности.

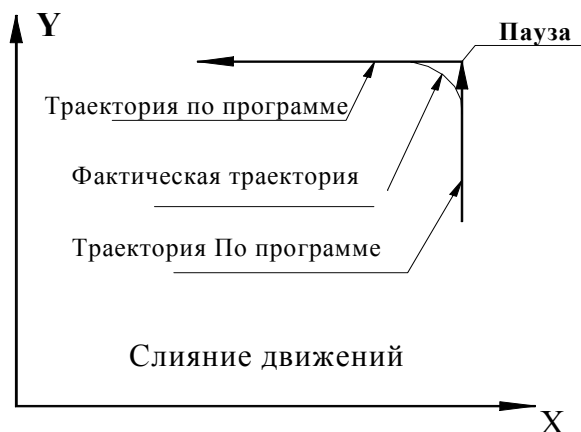


Рис. 3 Слияние движений

### 3.5.1. Режим точного останова (G61)

Активизация подготовительной функции **G61** приводит к останову движений между кадрами. В результате исключаются слияние движений и закругление угла (см. рис. 4). При этом выполняется торможение в конечной точке кадра обработки и переход к выполнению следующего кадра происходит только после точного позиционирования.

Подготовительная функция **G61** действует до активизации альтернативной функции **G64** (режим резания).

### 3.5.2. Функция отмены режима точного останова (Режим резания G64)

**G64** отменяет режим точного останова и выполняет слияние движений смежных кадров. Эта функция активна до ее отмены альтернативной функцией **G61** (режим точного останова).

#### Примечание:

*В режиме резания G64, движение с точным позиционированием в конце кадра происходит в следующих случаях:*

*в режиме быстрого позиционирования G0;*

*между кадрами движения установлена команда паузы G04;*

*следующий кадр не содержит команды движения.*

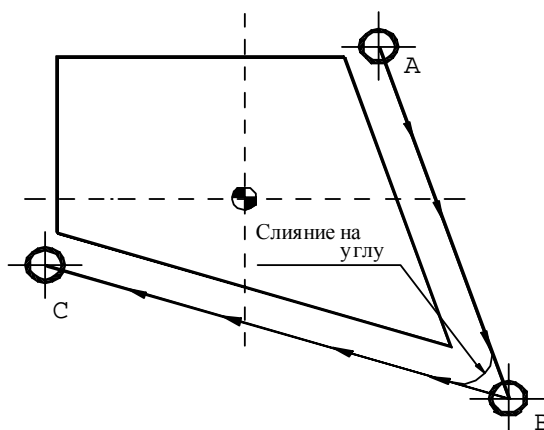


Рис. 4 Диаграмма точного останова

### 3.5.3. Торможение (G9)

Подготовительная функция **G9** выполняет точное позиционирование в конце заданного кадра так, что движение следующего кадра не сливается с движением текущего кадра. Функция **G9** не является модальной. Она действует только в текущем кадре, в отличие от функции **G61**, устанавливающей режим точного останова и являющейся модальной.

**Пример:**

G1 G64 X125 Y25 F100 ; установка режима движения со слиянием  
G9 X100 Y30 F50 ; точное позиционирование в кадре  
; отмена режима движения со слиянием в  
; текущем кадре

### 3.6. Пауза (G4)

Подготовительная функция **G4** задает паузу между движениями, задаваемыми функциями **G0/G1/G2/G3**.

Время паузы определяется в слове с адресом **P** в секундах.

Формат:

**G4** Pxxx,

Где:

Pxxx – время паузы в секундах. Диапазон значений: от 0.01 до 99999.999 сек.

**Пример:**

G0 S500 M3  
X0  
Z1 M8  
G4 P10 ; выдержка 10 сек

**Примечание:**

*Пауза может быть запрограммирована как в отдельном кадре, так и вместе с любой другой информацией. В последнем случае временная задержка выполняется после отработки всех перемещений и технологических команд, заданных в кадре.*

*По функции G4 прекращается движение только по осям, шпиндель продолжает вращение с заданной скоростью.*

### 3.7 Абсолютный размер / Размер в приращениях (G90/G91)

Задание перемещений по осям может осуществляться как в абсолютных значениях координат, так и в приращениях относительно текущей позиции. Эти установки осуществляются альтернативными подготовительными функциями **G90** и **G91** группы 15.

Функция **G90** устанавливает режим абсолютных перемещений, когда значения слов с адресами осей выражают абсолютные координаты в выбранной системе координат. Выбор системы координат осуществляется с помощью функций **G52**, **G53** и **G54-G59**.

Позиция в абсолютных координатах определяет движение инструмента к позиции определяемой в отношении нулевой точки координатной системы. Движение от точки **A** до точки **B** в абсолютных координатах задается путем указания координат конечной точки **B**.

Функция **G91** устанавливает режим задания размера перемещений в приращениях, когда значения слов с адресами осей выражают расстояния от текущих координат в выбранной системе координат. При этом знак заданной величины задает направление движения относительно текущей точки.

Позиция в относительных координатах определяет движение инструмента в отношении к текущей позиции инструмента. Движение от точки **A** до точки **B** задается разницей координат между **A** и **B** в виде числа со знаком.

### 3. 8. Выбор дюймовой/метрической системы единиц (G20/G21)

Выбор дюймовой/метрической системы единиц происходит с помощью подготовительных функций **G20** или **G21**. При задании метрической системы единиц **G21** в управляющей программе, все слова с адресами задания движения должны быть заданы в миллиметрах. При задании дюймовой системы единиц **G20** в управляющей программе, все слова задания движения должны быть заданы в дюймах. Разрешается во время выполнения программы менять систему исчисления, но при этом система исчисления для индикации координат движения, для значений смещений рабочих координат и инструмента не изменится и останется той, которая установлена интегратором или заводом изготовителем.

По умолчанию действует метрическая система единиц (**G21**).

#### Пример:

	; G21 действует по умолчанию
X20.5	; перемещение в координату X 20.5 мм
G20	; выбор дюймовой системы координат
Z5.1	; перемещение в координату Z 5.1 дюймов
G21	; выбор метрической системы координат
Z40X30	; перемещение в координату Z 40 мм и X 30 мм

### 3.9. Выбор плоскости круговой интерполяции (G17/G18/G19)

С помощью подготовительных функций **G17/G18/G19** выбирается плоскость программирования дуги при круговой интерполяции: **G17** - плоскость **XY**, **G18** - плоскость **XZ**, **G19** - плоскость **YZ** (см. рис. 5). При задании круговой интерполяции в плоскости **XY** должна быть активизирована подготовительная функция **G17**. При задании круговой интерполяции в плоскости **XZ** должна быть активизирована подготовительная функция **G18**. При задании круговой интерполяции в плоскости **YZ** должна быть активизирована подготовительная функция **G19**.

По умолчанию активна функция **G17**.



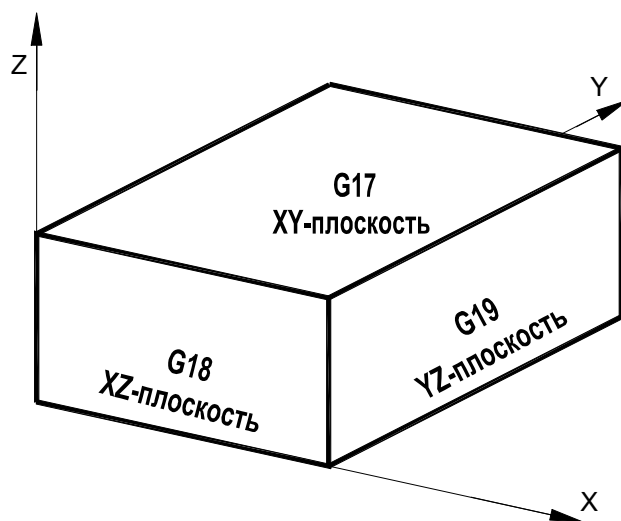


Рис. 5. Выбор плоскости круговой интерполяции

### 3.10 Движение до сигнала ( G31 )

При помощи этой функции задается движение до обнаружения сигнала срабатывания какого-либо датчика. Например, датчика касания измерительного щупа.

Формат:

G31 [X\_] [Y\_] [Z\_] [F\_]

Где:

**X,Y,Z** - определяют координаты конечной точки движения, до которой ожидается срабатывание сигнала датчика.

**F** - определяет векторную скорость перемещения.

Максимальная величина параметра **F** ограничивается системной (станочной) переменной, которая обычно равна **1000мм/мин**

**Примечание:**

*Параметры X,Y,Z,F могут быть не заданы.*

*Если какая - либо ось не задана, то перемещение по этой координате не происходит.*

*Если параметр F не задан, то скорость перемещения определяется значением системной (станочной) переменной. Рекомендуемое значение: 100мм/мин*

Когда **СЧПУ** обнаруживает срабатывание сигнала датчика, производит захват значений координат по осям **X,Y,Z** запоминает их значения в системных переменных и останавливает движение с торможением.

Фактическая точка останова может находиться на расстоянии небольшого перебега (до **1мм**) от точки касания из-за торможения осей.

Величина перебега зависит от параметра **F** – скорости перемещения.

Если во время движения не происходит срабатывания датчика, то движение останавливается в заданной в кадре конечной точке движения.

Функция **G31** не является модальной и действует только в одном кадре.

В зависимости от срабатывания сигнала датчика при **G31**, переменная **#5070** принимает соответствующее значение:

Значение 1 - сигнал датчика сработал;

Значение 0 - сигнал датчика не сработал.

**Примечание:**

*После выполнения G31, СЧПУ будет находиться в режиме быстрых перемещений G0.*

*Когда выполняется кадр с функцией G31, СЧПУ прекращает выполнение УП, ожидает завершения движения и после останова движения присваивает системным переменным #5061 -#5069 значения координат, в которых произошло срабатывание датчика.*

Координаты записываются с учетом смещений рабочей координатной системы, но без учета корректоров инструментов.

$$\#506n = K_{\text{маш}} - K_{\text{смещ}}$$

Где:

$K_{\text{маш}}$  - машинная координата по оси;

$K_{\text{смещ}}$  - заданное смещение по оси в текущей координатной системе (**G54-G59**)

Соответствие между осью и номером системной переменной приведено в таблице ниже.

Ось	Переменная
X	#5061
Y	#5062
Z	#5063
Z	#5064
V	#5065
W	#5066
A	#5067
B	#5068
C	#5069

Если по какой либо координате нет передачи данных, то переменная имеет значение “не определено” (см. **Раздел 10.4. Параметрического программирования**).

### **3.11 Масштабирование координат (G22/G23/G24)**

Все геометрические размеры в УП могут быть масштабированы, то есть по каждой координате может быть введен масштабный коэффициент, на который умножаются заданные в УП значения.

Масштабирование координат осуществляется посредством функций: **G22, G24**.

Формат:

**G24 X\_Y\_Z\_I\_J\_K\_**  
**G22 X\_Y\_Z\_I\_J\_K\_**

Где: **X, Y, Z** - центр масштабирования в программных координатах;  
**I, J, K** - масштабные коэффициенты без знака.

Центр текущей координатной системы удаляется (при **G22** - увеличение масштаба) или приближается (при **G24** - уменьшение масштаба) к центру масштабирования. Последующие программные перемещения умножаются (**G22**) или делятся (**G24**) на указанные в словах **I, J, K** масштабные коэффициенты. Для вычисления координат точки в рабочей системе координат на основе запрограммированных координат, система использует следующую формулу:

$$\begin{aligned} X_{\text{раб}} &= X + (X_{\text{прог}} - X) * I \\ Y_{\text{раб}} &= Y + (Y_{\text{прог}} - Y) * J \\ Z_{\text{раб}} &= Z + (Z_{\text{прог}} - Z) * K \end{aligned}$$

Где: **X<sub>раб</sub>, Y<sub>раб</sub>, Z<sub>раб</sub>** - координаты точки в рабочей системе координат;  
**X, Y, Z** - координаты центра масштабирования;  
**X<sub>прог</sub>, Y<sub>прог</sub>, Z<sub>прог</sub>** - запрограммированные координаты;  
**I, J, K** - коэффициенты масштабирования.

Отрицательный знак в масштабных коэффициентах отбрасывается.  
Подготовительная функция **G23** отменяет масштабирование.  
По умолчанию действует **G23**.

**Примечание:**

*Если не задан центр масштабирования по одной из координат, то для этой координаты центр масштабирования берется равным текущей позиции.  
Если не задан коэффициент масштабирования по одной из координат, то для этой координаты коэффициент масштабирования становится равным нулю.*

**Пример:**

G54G90F500G17  
G0X20Y20 ;Обработать квадрат стороной 10 мм  
G1 G90  
X30  
Y30  
G91X-10  
Y-10  
G22 X25 Y25 I2 J2 ; Увеличить масштаб в два раза  
; центр квадрата в координате X25 Y25.  
G90X20Y20 ; обработать той же программой  
X30 ; квадрат стороной 20 мм  
Y30  
G91X-10  
Y-10  
M2

### 3.12 Поворот системы координат / Отмена поворота (G68/G69)

При помощи функции **G68** можно выполнить поворота системы координат на произвольный угол относительно заданной точки в действующей плоскости, определяемой функциями **G17-G19**. В этом случае становится возможным, не изменяя программу, производить обработку детали, повернутой относительно начальных координат под определенным углом. При разработке УП, когда обрабатываемая деталь содержит идентичные фрагменты, повернутые относительно детали, время программирования и величина программы могут быть уменьшены за счет подготовки подпрограммы этих фрагментов и ее вызова после поворота.

Формат:

**G68 X\_ Y\_ Z\_ R\_**

Где: **R** - угол поворота в градусах в виде десятичного числа со знаком. Угол поворота может быть задан в пределах  $-360$  до  $+360$  градусов.

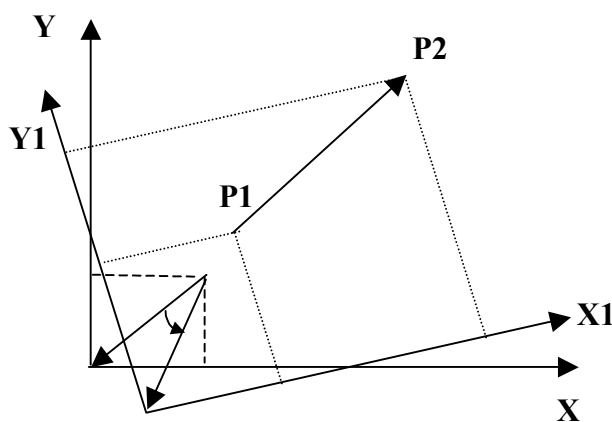
Положительным углом является угол, отсчитываемый против часовой стрелки.

**X, Y, Z** – координаты центра поворота в плоскости

Если координата **X,Y,Z** не задана, то значение это координаты принимается равным нулю.

Пример:

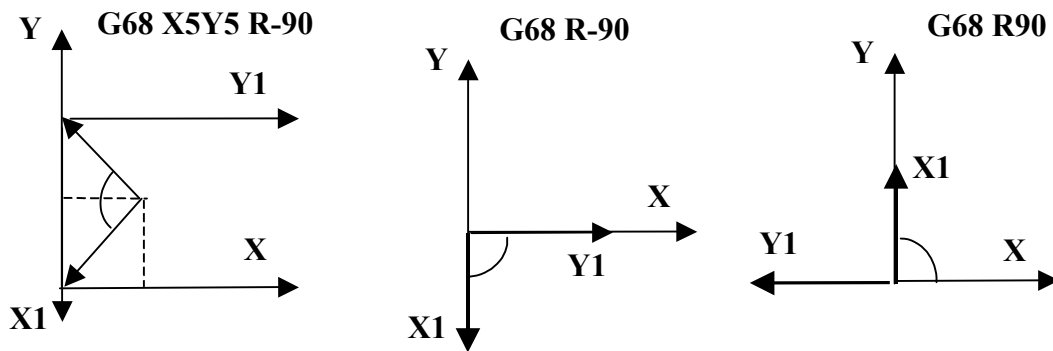
G17 G68 X10 Y10 R20



До задания функции **G68**, должна быть определена плоскость поворота с помощью одной из функций **G17, G18, G19**, которая может быть задана в том же кадре, что и **G68**. После активации функции **G68** работают все смещения инструмента, включая коррекцию на фрезе и длину инструмента.

После поворота координаты задаются уже в новой, повернутой системе координат. Возврат к исходной системе координат осуществляется заданием функции **G69**.

Примеры поворотов приведены ниже



### 3.13 Контроль скорости вращения шпинделя

#### 3.13.1 Включение контроля скорости вращения шпинделя (G26)

Подготовительная функция **G26** включает режим СЧПУ “контроль вращения шпинделя”.

Формат:

**G26**

После функции **G26**, при выполнении кадров программы, где заданы команды на вращение шпинделя, СЧПУ будет ожидать достижения скорости вращения шпинделя до заданной величины и только после этого перейдет к выполнению следующих кадров. Пока скорость вращения шпинделя не будет равна заданной, продолжение выполнения УП будет приостановлено.

#### 3.13.2 Выключения контроля скорости вращения шпинделя (G25)

Подготовительная функция **G25** отменяет режим СЧПУ “контроль вращения шпинделя”.

Формат:

**G25**

После функции **G25**, при выполнении кадров программы, где заданы команды на вращение шпинделя, СЧПУ *не будет* ожидать достижения скорости вращения шпинделя до заданной величины, а сразу перейдет к выполнению следующих кадров.

#### 3.13.3 Установка максимальной скорости вращения шпинделя (G92)

С помощью функции **G92** можно устанавливать максимальную скорость вращения шпинделя.

Формат:

**G92 S\_**

Где: **S\_** - указывает максимальное число оборотов в минуту.

#### Примечание:

*При активном режиме постоянной скорости резания(G96), увеличение скорости вращения шпинделя ограничено значением S, заданной в G92.*

### 3.14 Зеркальная обработка (G150/G151)

Подготовительная функция **G151** устанавливает режим зеркальной обработки

Формат:

**G151 X\_ Y\_ Z\_**

Где **X,Y,Z** - координаты центра зеркального преобразования.

Значение параметра определяет координату центра зеркального преобразования, т. е. показывает линию, относительно которой выполняется зеркальное преобразование. Координату центра зеркального преобразования имеет смысл задавать только в режиме абсолютных координат (**G90**). В режиме приращений (**G91**) значение параметра игнорируется, и, в качестве координаты центра зеркального преобразования, устанавливается текущая координата.

Если какая-либо ось зеркального преобразования не задана, то система выполняет отмену зеркального преобразования по этой оси.

Режим зеркального преобразования отменяется подготовительной функцией **G150**.

#### Примеры:

	; Задание режима зеркальной обработки при <b>G90</b>
G1 G90 X7 Y5	; движение в координату X7,Y5
G151 X5	; координата X7,Y5
	; центр зеркальной обработки по оси X, координата X5
X7	; движение в координату X3,Y5
Y15	; движение в координату X3,Y15
X10	; движение в координату X0,Y15
G91 X2	; движение в координату X-2,Y15
G150	

	; Задание режима зеркальной обработки при <b>G91</b>
G1 G91 X7 Y5	; движение в координату X7,Y5
G151 X5	; координата X7,Y5
	; центр зеркальной обработки по оси X, координата X7
G90 X7	; координата X7,Y5
G91 X2	; движение в координату X5,Y5
G90 Y15	; движение в координату X5,Y15
X10	; движение в координату X4,Y15
G150	

## 4. ВЫБОР РАЗНОВИДНОСТИ ДВИЖЕНИЯ

### 4.1 Быстрое позиционирование (G0)

Подготовительная функция **G0** используется для позиционирования инструмента из текущей программной точки к следующей программной точке, на максимальной скорости хода по всем осям.

Формат:

**G0 X\_ Y\_ Z\_**

**G0** является модальной функцией группы 1. Она отменяется при активации любой другой альтернативной функции этой группы. Одновременно можно запрограммировать перемещение по всем осям. Каждая ось может иметь самостоятельную величину ускорения/торможения в конечной точке и может иметь различную максимальную скорость позиционирования. Из всех осей заданных в кадре, только для оси с наибольшим значением отношения расстояния - скорость (расстояние/скорость), движение будет выполняться на максимальной скорости. Для других осей скорости движения будут замедлены так, чтобы движения по всем осям завершились почти одновременно. В этом случае, получается приблизительно линейное движение, но без линейной интерполяции.

#### Пример:

G54 G21 G40

S2500 M03

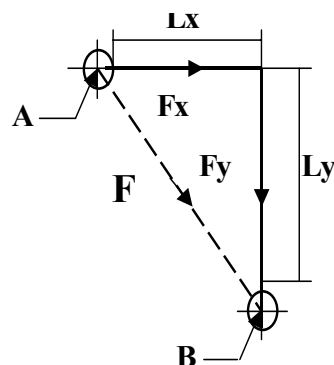
G0 X5 Y40 Z100 ; режим позиционирования

### 4.2 Линейная Интерполяция (G1)

Подготовительная функция **G1** линейно интерполирует позицию инструмента от текущей точки **A** до программируемой точки **B** (см. рис 6)

Формат:

**G1 X\_ Y\_ Z\_ F\_**



**Рис. 6 Линейная интерполяция**

Скорости перемещений по осям рассчитываются исходя из значения скорости подачи, указанной в функции подачи с адресом **F**, и являются составляющими вектора скорости инструмента в пути, определяемыми по формулам:

$$F_x = F \frac{L_x}{\sqrt{L_x^2 + L_y^2}}; F_y = F \frac{L_y}{\sqrt{L_x^2 + L_y^2}}; F_z = F \frac{L_z}{\sqrt{L_x^2 + L_z^2}},$$

где  $L_x$ ,  $L_y$  и  $L_z$  – расстояния пройденных путей по координатам X, Y и Z соответственно. Если в кадре скорость подачи  $F$  не указывается, то движение выполняется с подачей, установленной ранее.

Линейное движение может сливаться с движением следующего кадра. Если кадр с функцией **G1** содержит функцию паузы **G4**, то отменится слияние и произойдет точный останов в конце текущего кадра. Если активен режим точного останова (**G61**), то слияние движений между кадрами не произойдет до тех пор, пока не выбран Режим резания (**G64**).

Подготовительная функция **G1** является модальной в группе 1. Она отменяется при активизации любой другой функции группы 1.

**Пример:**

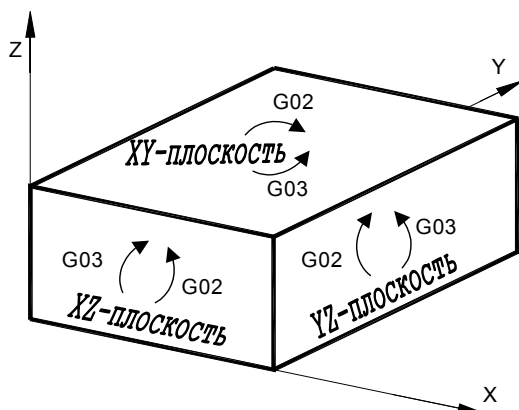
X11.2 Y-125 F20 G1 ;линейная интерполяция  
 G61 Y20.5 F20 ;точный останов в конце кадра  
 X39  
 X42 Y-120  
 G64 X43 Y-100 ;режим резания - движения со слиянием  
 X50 Y-80

### 4.3 Круговая интерполяция. (G2/G3)

Подготовительные функций **G2** и **G3** используются для задания движения инструмента по дуге окружности. Подготовительные функций **G2** используются для задания круговой интерполяции по часовой стрелке, а **G3** – против часовой стрелке.

Для выполнения круговой интерполяции, предварительно должна быть задана плоскость интерполяции подготовительными функциями **G17/G18/G19**.

Направление движения для **G2** (по часовой стрелке) и **G3** (против часовой стрелки) можно определить, смотря на плоскости интерполяции (XY, XZ, YZ) с положительного направления оси перпендикулярной этой плоскости (см. рис. 7).



**Рис. 7. Определение направления движения при G2 и G3**

Скорость движения инструмента определяется функцией подачи – слово с адресом

**F** и равна векторной сумме скоростей подач по осям:  $F_t = \sqrt{f_x^2 + f_y^2}$ .

Круговая интерполяция программируется, основываясь на трех исходных параметрах:



- начальная точка дуги;
- конечная точка дуги;
- центр дуги.

Начальной точкой круговой интерполяции **G2/ G3** является конечное положение инструмента, определенное предыдущим движением **G1** или **G2, G3**.

Конечная точка определяется заданными координатами осей в адресах **X** и **Y** в кадре **G2/G3**.

Центр дуги задается двумя вариантами:

- заданием центра окружности с помощью адресов **I,J,K**;
- заданием радиуса окружности с помощью адресного слова **R**.

**Примечание:**

- *Подготовительные функции G2 и G3 действуют модально, т.е. если их не отменить другой функцией группы 1, то задание движения в последующих кадрах будет восприниматься как задание круговой интерполяции;*
- *Если в кадре не заданы параметры центра дуги I,J,K или R, и задано перемещение, то будет выполняться линейное движение (линейная интерполяция).*

**4.3.1 Программирование дуги с помощью I,J,K**

Формат команды с заданием **I,J,K**:

**G2 (или G3) X\_ Y\_ I\_ J\_** - для плоскости **XY**;

**G2 (или G3) X\_ Z\_ I\_ K\_** - для плоскости **XZ**;

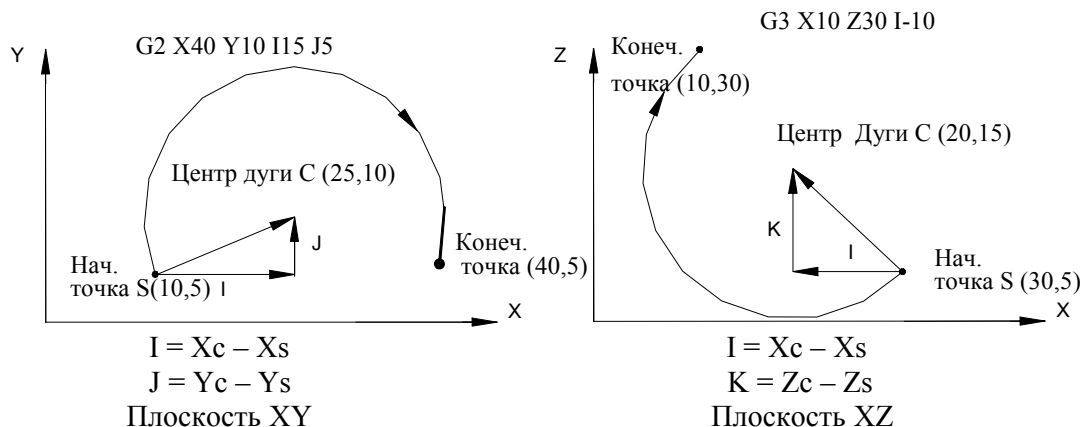
**G2 (или G3) Y\_ Z\_ J\_ K\_** - для плоскости **YZ**.

При задании центра дуги, адресные слова **I,J,K** определяют координату центра дуги относительно начальной точки и вычисляются, как разность координаты центра дуги и координаты начальной точки дуги (см. рис. 8).

Значение **I,J,K** задается в приращениях независимо от режима работы программы – задания движений в абсолютных размерах (**G90**) или в приращениях (**G91**).

Если один из параметров **I,J,K** не задан, то он берется равным нулю

При программировании дуги с помощью **I,J,K** можно задать не только дугу окружности, но и полную окружность.



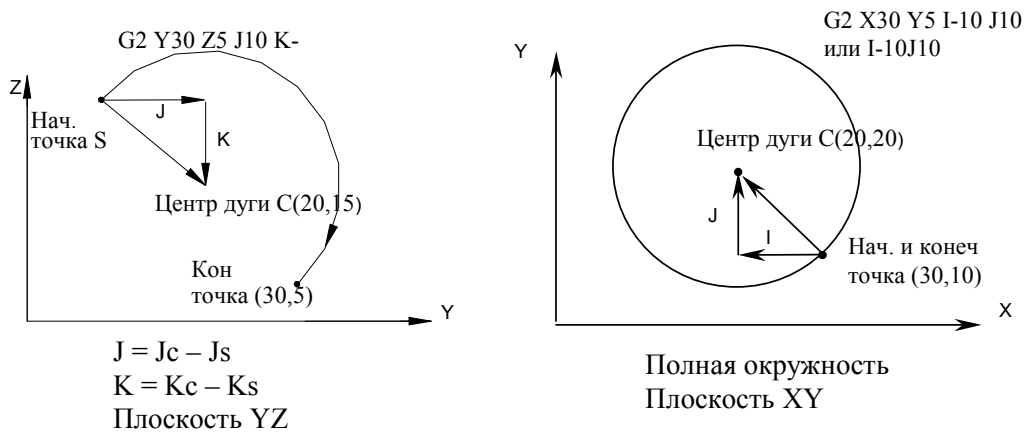


Рис. 8. Определение I, J, K

**Примеры:**

G54 G90 F200  
G0 X0 Y0 Z0  
X10

; пример для плоскости XY с заданием I, J  
G17

G2 X0 Y-10 I-10 ; дуга  $0^0-90^0$

X-10 Y0 J10 ; дуга  $90^0-180^0$

X0 Y10 I10 ; дуга  $180^0-270^0$

X10 Y0 J-10 ; дуга  $270^0-360^0$

X-10 I-10 ; дуга  $0^0-180^0$

X10 I10 ; дуга  $180^0-360^0$

I-10 ; полная окружность

G54 G90 F200  
G0 X0 Y0 Z0  
X10

; пример для плоскости XZ с заданием I, K  
G18

G3 X0 Z-10 I-10 ; дуга  $0^0-90^0$

X-10 Z0 K10 ; дуга  $90^0-180^0$

X0 Z10 I10 ; дуга  $180^0-270^0$

X10 Z0 K-10 ; дуга  $270^0-360^0$

X-10 I-10 ; дуга  $0^0-180^0$

X10 I10 ; дуга  $180^0-360^0$

I-10 ; полная окружность

G54 G90 F200  
G0 X0 Y0 Z0  
Y10

; пример для плоскости YZ с заданием J, K  
G19

G2 Y0 Z-10 J-10 ; дуга  $0^0-90^0$

Y-10 Z0 K10 ; дуга  $90^0-180^0$

Y0 Z10 J10 ; дуга  $180^0-270^0$

Y10 Z0 K-10 ; дуга  $270^0-360^0$

Y-10 J-10 ; дуга  $0^0-180^0$

Y10 J10 ; дуга 180<sup>0</sup>-360<sup>0</sup>  
 J-10 ; полная окружность

### 4.3.2 Программирование дуги с помощью R

Формат команды с заданием R:

G2 (или G3) X\_ Y\_ R\_ - для плоскости XY

G2 (или G3) X\_ Z\_ R\_ - для плоскости XZ

G2 (или G3) Y\_ Z\_ R\_ - для плоскости YZ

Центр дуги определяется заданием радиуса дуги при помощи адреса R. Положительное значение R задает короткую дугу ( $\leq 180^\circ$ ), а отрицательное – длинную ( $> 180^\circ$ ) (см. рис 9).

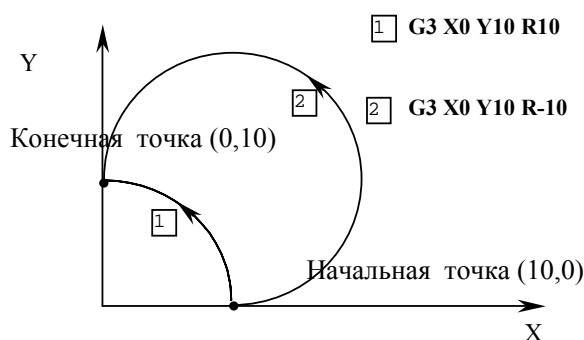


Рис. 9. Задание дуги окружности при помощи R

При задании дуги окружности с помощью адреса R, нельзя программировать полную окружность. Дуга окружности в кадре должна быть меньше 359.9<sup>0</sup>. Полную окружность следует задать с помощью двух последовательных кадров.

**Пример:**

G17 G0 X10 Y0

F100 G3 X0 Y-10 R-10 ; дуга 0<sup>0</sup> - 270<sup>0</sup>

G3 X10 Y0 R10 ; дуга 270<sup>0</sup> - 360<sup>0</sup>

### 4.4 Спиральная интерполяция (G2/G3)

Если в кадре круговой интерполяции G2 или G3 задано перемещение по оси, перпендикулярное плоскости круговой интерполяции, то выполняется движение по спиральной интерполяции с величиной подачи по оси спирали равной: F \* (длина линии по оси/длина дуги).

Например, для плоскости XY спиральная интерполяция записывается в виде:

G17

G2 X\_ Y\_ Z\_ I\_ J\_ F\_

## 5. СИСТЕМЫ КООРДИНАТ И РЕФЕРЕНТНАЯ ТОЧКА

### 5.1 Основные положения

Задание перемещения осей всегда производится в какой-либо системе координат. СЧПУ позволяет задавать позиции рабочего органа в системе координат связанной со станком или с деталью.

**Система координат станка** определяется при проектировании станка и учитывает предельные размеры перемещений. **Система координат детали** определяется программой ЧПУ и учитывает особенности геометрии обрабатываемой детали.

Управление движением осуществляется в первой системе координат. Поэтому для обработки заготовки по программе, записанной в системе координат детали, эти две координатные системы должны быть связаны математически.

### 5.2 Координатная система станка

Точка - *ноль станка*, является стандартной точкой на станке. Физически координаты этой точки устанавливаются при разработке станка. Система координат с началом в точке ноль называется **система координат станка**. После включения СЧПУ, **система координат станка** устанавливается после выполнения команды выхода в ноль по всем осям.

### 5.3 Координатная система программы

Координаты программы задаются в **системе координат детали** с помощью подготовительных функций G54 - G59.

При помощи задания смещений относительно **системы координат станка**  $W_{off}$  (иногда называется смещением нуля станка), определяется положение начала **системы координат детали** (см. рис.10). Внутри **системы координат детали** локальные смещения  $L_{off}$  определяют начало **локальной системы координат**. При отсутствии рабочего и локального смещений, станочные и программные координаты совпадают.

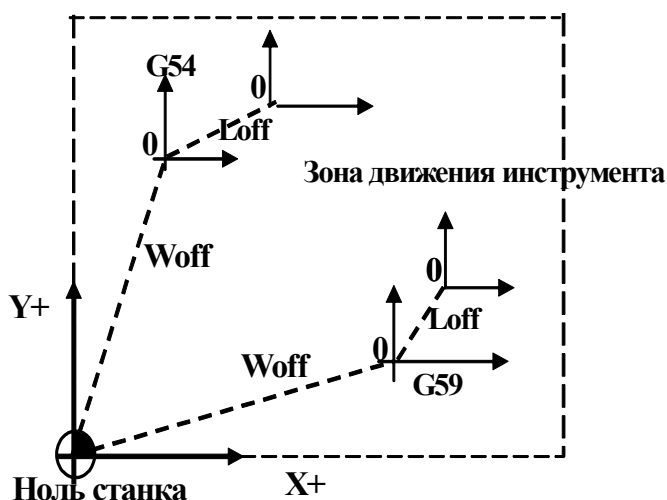


Рис. 10 Системы координат

## 5.4 Выбор координатной системы

### 5.4.1 Выбор локальной системы координат (G52)

Когда программирование осуществляется в *системе координат детали*, иногда удобно иметь временную координатную систему внутри рабочей системы координат. Такая система координат называется локальной.

Формат задания локальной системы координат:

**G52 X\_ Y\_ Z\_**

При помощи **G52** будет задана *локальная система координат (X'Y'Z')*, начало которой будет смещено относительно рабочей *XYZ* так, что текущая точка инструмента в локальной системе координат примет значение заданных в адресах *X,Y,Z* величин.

Например, при задании **G52X100Y100**, локальная система координат установится относительно рабочей смещенной на вектор **A** (100,60) (см.рис. 11) и текущая точка инструмента станет равной X100, Y100, вместо X200, Y160.



**Рис. 11 Установка локальной системы координат**

При отсутствии одного из параметров X,Y,Z, координаты локальной системы по этой оси не смещаются.

После установки локальной системы координат все последующие команды движения выполняются в локальной системе координат. Локальная система координат может быть изменена заданием новой функции **G52** с новыми значениями адресов *X,Y,Z* в рабочей системе координат.

### 5.4.2 Отмена локальной системы координат (G152)

Функция **G152** отменяет локальную систему рабочих координат и восстанавливает активную предыдущую рабочую систему координат.

При смене рабочей системы координат при помощи **G54-G59**, локальная система координат отменяется автоматически.

#### Пример:

G0G17G54

X20Y20Z20 ; координаты инструмента X20Y20Z20

G52 X5 Y5 ; установка локальной системы координат

; координаты инструмента X5Y5Z20

X30 Y30 ; движение в локальной системе координат  
Z40  
G152 ; отмена локальной системы координат  
M2

#### 5.4.3 *Выбор системы координат станка (G53)*

Подготовительная функция **G53**, позволяет программировать движение непосредственно в системе координат станка.

Обычно при работе применяется рабочая система координат **G54-G59**, но в случаях, когда необходимо переместиться к определенной точке координатной системы станка, отменив все активные смещения инструмента и рабочих координат, применяется подготовительная функция **G53**. Эта функция не отменяет другие установки и действует только в текущем кадре. Из функций интерполяции могут быть использованы только **G00** или **G01**.

Формат задания движения в системе координат станка:

**G53 X\_Y\_Z\_**

Инструмент переместится в точку с абсолютными координатами станка указанными в кадре. Скорость перемещения определяется в зависимости от действующей функции **G0** или **G1**. При **G1** скорость перемещения можно задавать в текущем кадре.

После кадра с функцией **G53** автоматически восстанавливаются текущие активные смещения инструмента и рабочей системы координат.

#### Пример:

G54 X0 Y0 ; движение в X0 Z0 в рабочих координатах G54  
G0 G53 X0 Y0 ; выход "в ноль" станка  
X20 ; движение в X20 рабочих координатах G54

#### 5.4.4 *Выбор рабочей системы координат (G54-G59)*

СЧПУ имеет шесть рабочих систем координат, которые привязываются к координатам станка при помощи задания рабочих смещений. После задания смещений рабочим системам координат, любая из них может быть активизирована при помощи подготовительных функций **G54 -G59**.

Рабочая система координат 1 - **G54**

Рабочая система координат 2 - **G55**

Рабочая система координат 3 - **G56**

Рабочая система координат 4 - **G57**

Рабочая система координат 5 - **G58**

Рабочая система координат 6 - **G59**

Рабочая координатная система определяется путем указания расстояний (заданных смещений) по каждой оси от собственной нулевой точки до нулевой точки станка (см. рис.11). Смещения задаются и запоминаются на экране рабочих смещений СЧПУ (см. Руководство оператора).

#### Пример:

G55 G00 X20.0 Z100.0  
X 40.0 Z20.0

В приведенном выше примере быстрое позиционирование осуществляется в точку (X=20.0; Z=100.0) и далее (X=40.0; Z=20.0) в рабочей системе координат 2(**G55**). Точка на станке, на которую будет позиционирован инструмент, зависит от значения заданного смещения нуля рабочей системы координат 2.

Установки рабочих систем координат запоминаются в файле данных Coords.dat и сохраняются на диске. При включении СЧПУ все последние установки смещений рабочих систем координат из этого файла восстанавливаются.

### 5.5 Референтная точка

Кроме нуля станка возникает необходимость иметь определенные фиксированные позиции, связанные с особенностями оборудования, например сменой инструмента или сменой паллетов. Эти позиции называются референтными точками и, в частности, могут совпадать с нулем станка.

Переход к референтной точке может быть осуществлен двумя способами:

- Ручным - при выполнении ручных операций;
- Автоматическим - из управляющей программы.

Выход в референтную точку вручную может выполнить оператор после выполнения операций выхода в ноль станка. Функция автоматического выхода в референтную точку может использоваться для движения инструмента в позицию, где происходит, например, смена инструмента, или в позицию, где станок будет ожидать продолжения выполнения программы, или начать выполнять программу снова.

#### 5.5.1 Выход в референтную точку (G28)

Формат:

**G28 X\_Y\_Z\_**

Где **X,Y,Z** координаты промежуточной точки **B** в текущей рабочей системе координат.

Со скоростью быстрого перемещения **G0** инструмент из текущей точки **A** (см. рис. 12) перемещается в заданную промежуточную точку **B**, а затем на высокой скорости позиционирует в референтную точку **R**. Промежуточная точка **B** запоминается для использования подготовительной функцией **G29**.

Выход к референтной точке может быть выполнен по любой одной или любыми двумя или тремя осями одновременно. При этом движение выполняется только по заданным в функции **G28** осям.

**G28 X\_Y\_Z\_** ; движение по трем осям X,Y,Z в референтную точку  
**G28 Z\_Y\_** ; движение по двум осям Z,Y в референтную точку  
**G28 Y\_** ; движение по одной оси Y в референтную точку

Допускается последовательно задавать в кадрах подготовительную функцию **G28**.

#### Пример:

G91 G28 Z0  
G91 G28 X0 Y0

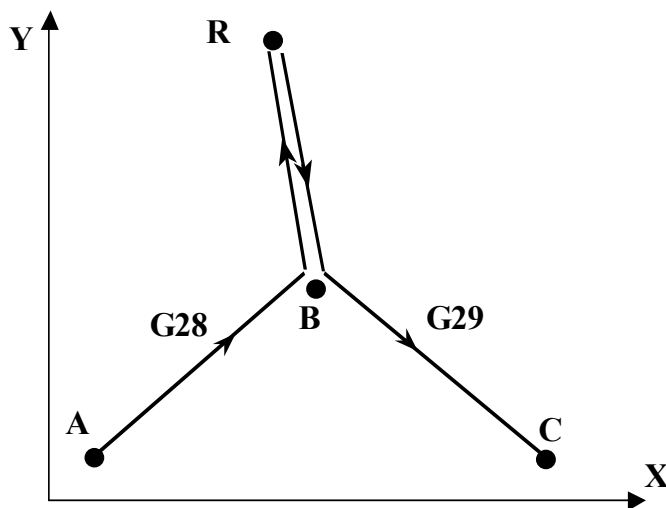


Рис. 12 Выход в референтную точку через промежуточную

Цель такого задания **G28** - первый кадр для движения по оси **Z**, чтобы увести инструмент на безопасное место. Второй кадр для движения по осям **X** и **Y**. Координаты референтной точки устанавливаются интегратором и не могут быть изменены со стороны оператора.

Пример:

```
F500
G01X-10Z-10
X-40Z-10
G28 X50 Z100
```

### 5.5.2 Возврат из референтной точки (G29)

Формат:

**G29 X\_Y\_Z\_**

Где:

**X,Y,Z** - координаты конечной точки **C** в текущей рабочей системе координат.

Инструмент из референтной точки **R** на высокой скорости позиционирует к промежуточной точке **B**, указанной в кадре с функций **G28** или **G30**, а затем на высокой скорости перемещается к определенной в кадре точке **C** (см. рис. 12) .

Возврат из референтной точки может быть выполнен по любой одной или любыми двумя или тремя осями одновременно. При этом движение выполняется только по заданным в функции **G29** осям.

**G28 X\_Y\_ Z\_** ; движение по трем осям X,Y,Z из референтной точки  
**G28 Z\_Y\_** ; движение по двум осям Z,Y из референтной точки  
**G28 Y\_** ; движение по одной оси Y из референтной точки

Допускается последовательно задавать в кадрах подготовительную функцию **G29**.



**Пример:**

F500  
G1 X-10Y-10  
X-40 Y-10  
G28 X50 Y100  
M0  
G29 X-30 Y-5  
G1 X-50 Y-30

**5.5.3 Выход ко 2-ой - 4-ой референтным точкам (G30)**

Подготовительная функция **G30** работает также как функция **G28**, но используется для задания выхода ко 2-ой - 4-ой референтным точкам.

Формат:

**G30 P\_ X\_ Y\_ Z\_**

Где: **X,Y,Z** координаты промежуточной точки в текущей рабочей системе координат;

**P** – задает номер референтной точки и принимает значение 1,2,3,4.

При задании **P1** выполняется выход к первой референтной точке, которая аналогична референтной точке для **G28**.

При задании **P2**, **P3** или **P4**, инструмент движется ко 2-ой, 3-ей или 4-ой референтной точке, через промежуточную точку, заданную словами **X,Y,Z**. Заданная точка запоминается для последующего использования в кадре с подготовительной функцией **G29**.

Координаты референтных точек 2,3,4 задаются интегратором при настройке параметров станка, но могут быть изменены, при необходимости, с помощью системных параметров редактора (см. **Руководство оператора**).

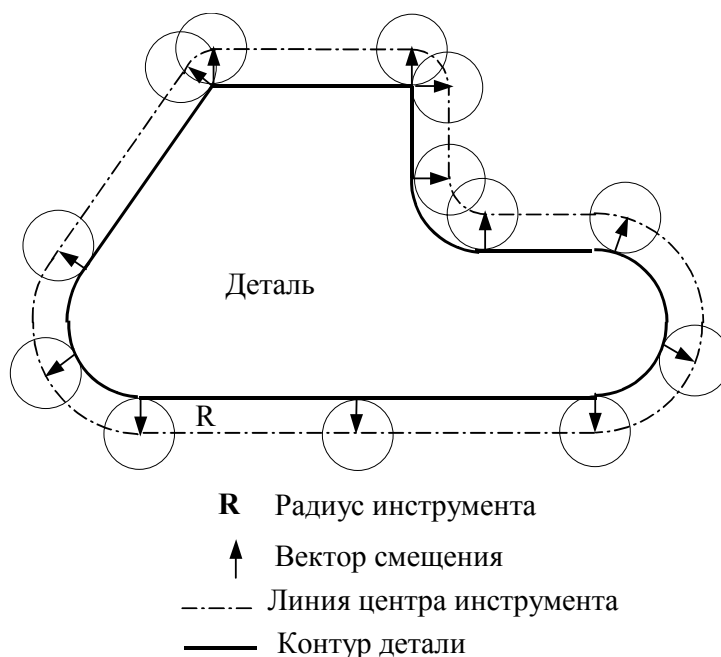
**Пример:**

F500  
G1 X-10Y-10  
X-40 Y-10  
G30 P3 X50 Y100 ; выход к 3-ей референтной точке  
M0  
G29 X-30 Y-5  
G1 X-50 Y-30

## 6. КОРРЕКЦИЯ НА РАДИУС ИНСТРУМЕНТА

В процессе обработки по запрограммированному контуру, состоящему из линий и дуг, в зависимости от направления резания, инструмент должен быть постоянно ориентированным к обрабатываемой поверхности и быть смещенным, что необходимо для формирования заданного контура поверхности (см. **Рис. 13**). При этом обычно требуется расчет эквидистантного контура, по нормали к касательной поверхности.

Коррекция на радиус инструмента используется для учета радиуса инструмента для выполнения эквидистантных движений по программируемому контуру.



**Рис. 13. Коррекция на радиус инструмента**

### 6.1 Коррекция на радиус инструмента: отмена/слева/справа (G40/G41/G42)

Во время движения по контуру, при активизации режима **Коррекции на Радиус Инструмента (КРИ)**, СЧПУ начинает выполнять расчет эквидистантного контура и смещает инструмент на величину его радиуса, и выполняет движения перпендикулярно к текущей линии поверхности программируемого контура детали с учетом направления движения инструмента и плоскости коррекции (см. **Рис. 13**). Таким образом, можно не меняя управляющую программу, обрабатывать один и тот же контур детали с инструментами, имеющими разные радиусы.

Для реализации КРИ предусмотрены подготовительные функции **группы 6**, состоящие из следующих альтернативных функций:

- **G40** Отмена коррекции на радиус инструмента;
- **G41** Коррекция на радиус инструмента слева;
- **G42** Коррекция на радиус инструмента справа.

Формат:

**G41/G42 X\_Y\_D\_R\_** ; для плоскости XY

**G40 X\_Y\_R\_**

**G41/G42 X\_Z\_D\_R\_** ; для плоскости XZ  
**G40 X\_Z\_R\_**

**G41/G42 Y\_Z\_D\_R\_** ; для плоскости YZ  
**G40 Y\_Z\_R\_**

**D\_** - номер корректора, в котором заданы радиус инструмента. Этот параметр может отсутствовать. В этом случае устанавливается или предыдущий заданный корректор или, если была задана функция смены инструмента по M6, то номер корректора **D** устанавливается равным номеру текущего инструмента **T**.

**R\_** - определяет величину радиуса дуги и активизирует режим, по которой:

- если в кадре задана функция **G41** или **G42**, вход в режим эквидистантного движения будет выполнен по дуге окружности **R** ;
- если в кадре задана функция **G40**, выход из режима эквидистантного движения будет выполнен по дуге окружности **R**.

Этот параметр может отсутствовать, и в этом случае выполняется вход в режим и выход из режима эквидистантного движения будет выполнен по линии.

Для активации коррекции должны быть определены несколько параметров:

- плоскость коррекции,
- величина коррекции - радиус инструмента,
- направление коррекции.

**Плоскость.** Плоскость коррекции определяется с помощью задания одной из подготовительных функций **G17,G18,G19**. Плоскость коррекции должна быть определена заранее до активизации режима коррекции

**Величина коррекции.** Величина коррекции - радиус инструмента, устанавливается, используя параметр заданного корректора **D** (см. Раздел.1.8).

**Направление.** Направление и начало коррекции на радиус определяется выбором подготовительных функций **G41/G42**. Коррекция отменяется с помощью функции **G40**.

**G41** – коррекция слева от контура, смещает инструмент влево от детали, если смотреть по направлению движения инструмента (см. Рис. 14а).

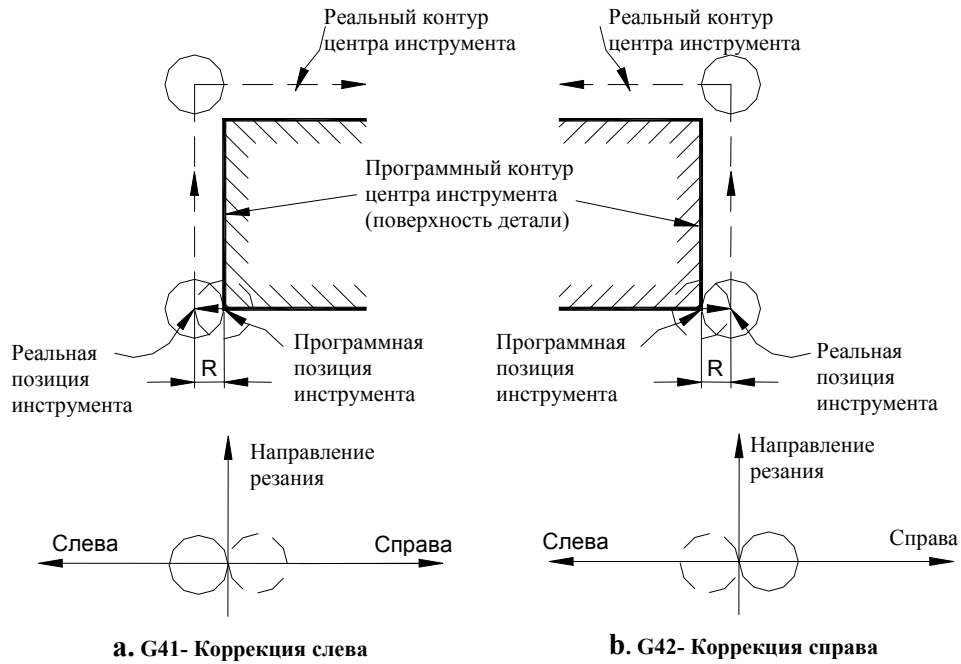
**G42** – коррекция справа от контура, смещает инструмент направо от детали, если смотреть по направлению движения инструмента (см. Рис. 14b.)

Два фактора должны учитываться для правильного выбора команд **G41** и **G42** для режима коррекции:

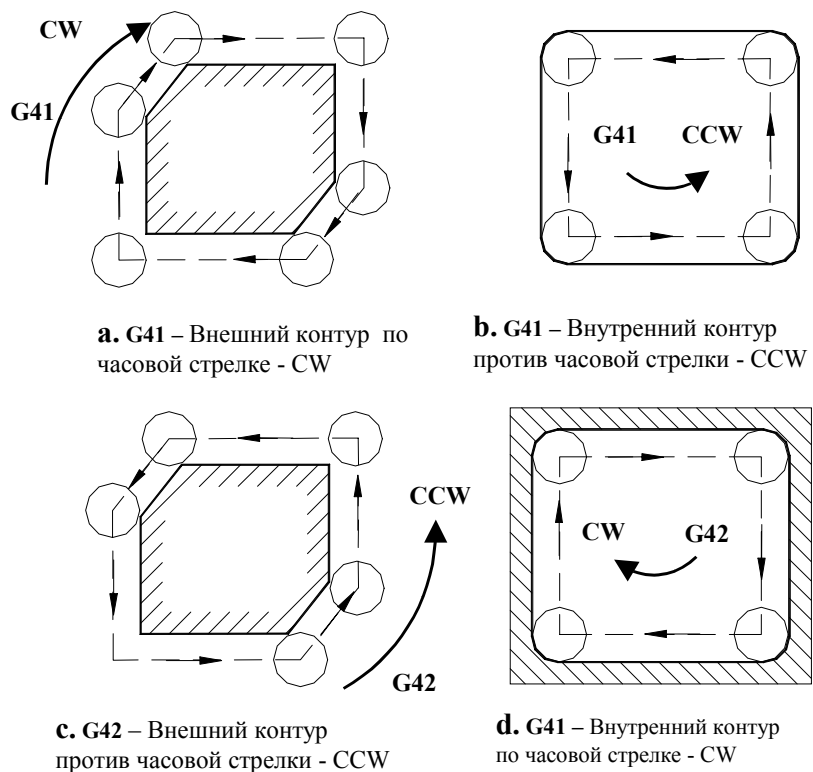
- направление движения инструмента по часовой стрелке или против нее;
- внутренняя или наружная обработка.

**G41** – коррекция слева от контура, используется для наружной обработки при движении по часовой стрелке или для внутренней обработки при движении против часовой стрелки (см. Рис. 15а и 15b).

**G42** – коррекция справа от контура, используется для наружной обработки при движении против часовой стрелки или для внутренней обработки при движении по часовой стрелке (см. Рис. 15с и 15d).



**Рис. 14** Направление коррекция радиуса инструмента



**Рис. 15** Направление обработки для G41 и G42

Режим коррекции на радиус инструмента, установленный однажды, остается активным, пока не будет отменен с помощью **G40**.

При активизации режима коррекции, СЧПУ берет значение радиуса инструмента из параметра активного корректора **D** (см. Раздел.1.8).

### 6.2 Скорость корректирующего движения

Скорость движения центра инструмента на пути коррекции остается той же что и задано функцией подачи в адресе **F**. При движении по дуге, это означает, что скорость перемещения режущей кромки (часть инструмента, контактирующая с заготовкой) будет отличаться от запрограммированной на величину отношения  $R_{\text{tool}}/R_{\text{arc}}$ , где:

$R_{\text{tool}}$  – радиуса инструмента;

$R_{\text{arc}}$  – радиус дуги.

### 6.3 Активизация режима коррекции

Любые действия изменения режима коррекции (активизация, отмена, изменение направления коррекции) выполняются только при активной функции **G1**, и посредством задания движения после функции изменения режима коррекции.

В кадре активизации, где заданы функции **G41** или **G42**, или сразу вслед за ним, необходимо совершить установочные движения для входа на эквидистантную траекторию. При этом инструмент проходит вектор пути (см **Рис. 16, 17 и 17А**) в зависимости от угла наклона и типа линии (прямая или дуга) первого и второго программных движений после **G41, G42**.

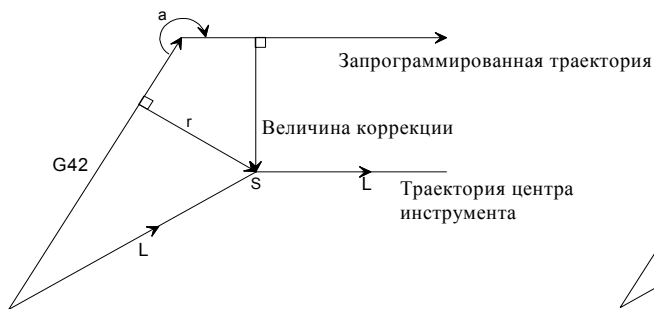
Рекомендуется перед началом коррекции позиционировать инструмент таким образом, чтобы коррекция выполнялась по нормали к поверхности. Необходимо, чтобы центр радиуса инструмента был удален от обрабатываемой поверхности, как минимум, на величину радиуса инструмента.

При задание параметра **R** с функцией **G41/G42**, выполняется вход на эквидистантную траекторию по дуге окружности (см. **рис. 17А**). При этом инструмент, по линии подходит к поверхности заготовки на расстоянии  $r + R$  по одной оси (  $r$  - это радиус инструмента заданный в текущем корректоре **D** ) и смещенно на расстояние **R** по другой оси и затем выполняется круговая интерполяция по дуге окружности радиуса **R**.

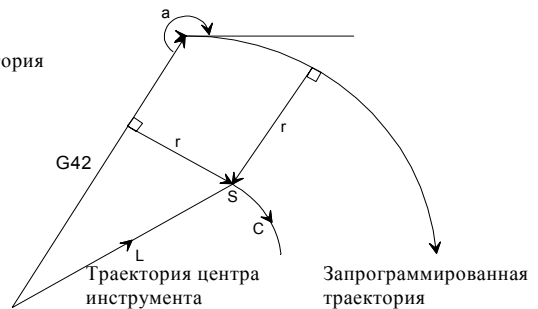
## Начало Коррекции

а) Для внутреннего угла

1) Линия → Линия

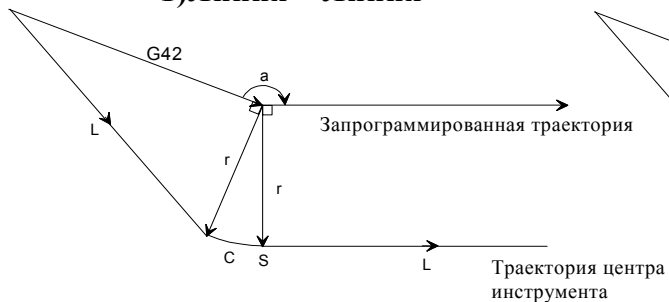


2) Линия → Дуга

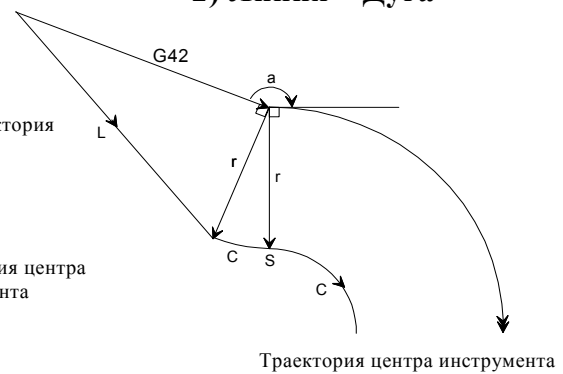


б) Для внешнего угла

1) Линия → Линия



2) Линия → Дуга



S = Точка Сопряжения

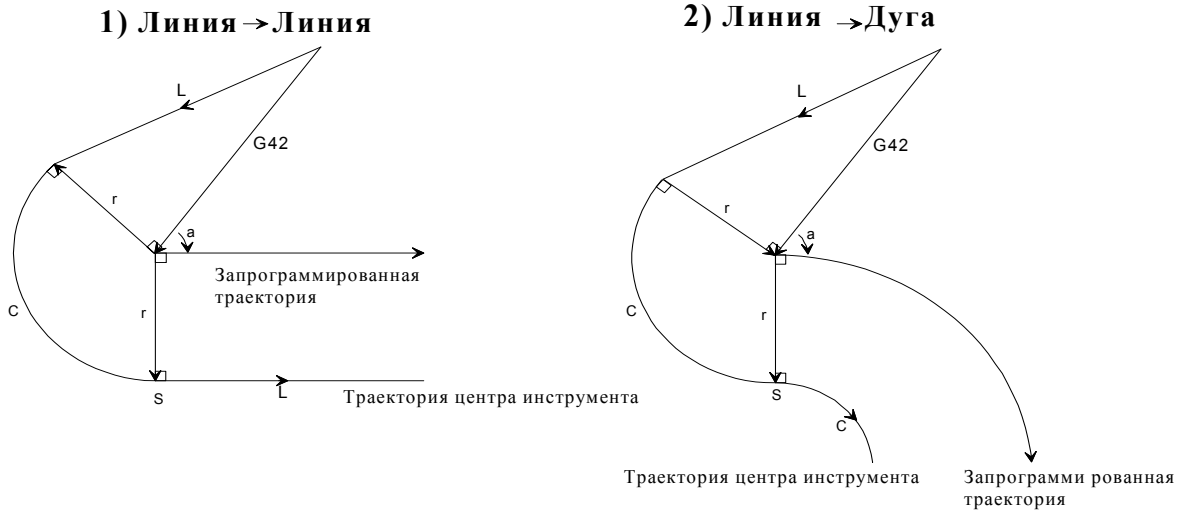
L = Линия

C = Дуга

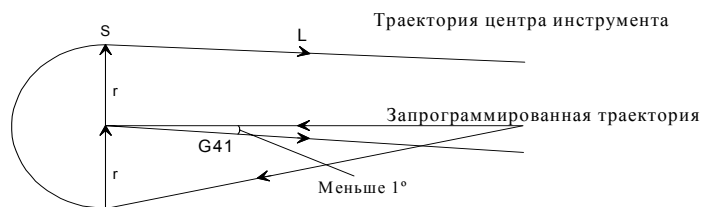
Рис. 16 Вход в эквидистантное движение при активизации коррекции на радиус инструмента. Варианты: а) – [1, 2] и б) – [1, 2]

## Начало Коррекции

с) Для внешнего острого угла



d) Для внешнего острого угла меньше чем  $1^\circ$   
Линия → Линия



S = Точка сопряжения  
L = Линия  
C = Дуга

**Рис. 17** Вход в эквидистантное движение при активизации коррекции на радиус инструмента. Варианты: с) – [1, 2] и d)

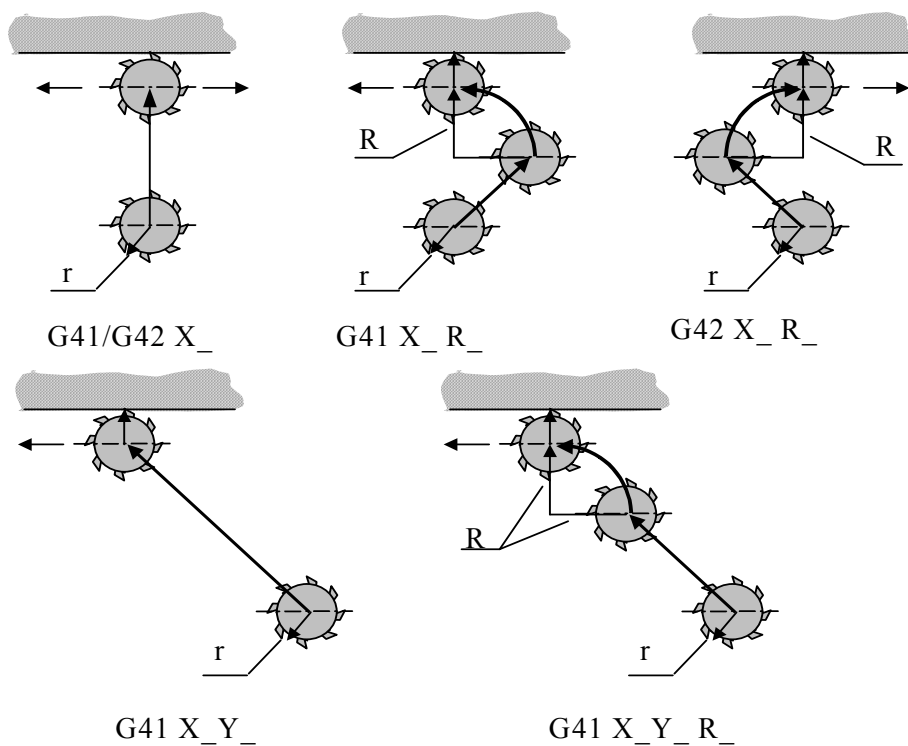


Рис 17А. Вход на эквидистатную траекторию по дуге окружности. Примеры



## **6.4 Движение при активном режиме КРИ**

При активном режиме компенсации радиуса инструмента, СЧПУ выполняет дополнительные движения между двумя кадрами движения (см. Рис 18-20), в зависимости от угла наклона и типа линии (прямая или дуга) между этими кадрами.

### **6.4.1      *Обработка внутреннего угла***

Во время режима коррекции, при переходах движений на внутренних углах, в вершинах внутренних углов выполняется слияние движений. Чем больше время ускорения при обработке угла, тем значительнее закругление угла. Закругление угла начинается и кончается на расстоянии пути коррекции  $F \cdot T/2$  от пересечения сторон геометрического угла (см. Рис. 3). Здесь  $F$ - скорость подачи, а  $T$ - время разгона или торможения при движении, установленное интегратором для станка. Обычно в СЧПУ устанавливается время  $T$  в диапазоне **0.01-0.1 сек.**

Когда применяется режим точного останова на внутреннем угле, то происходит коррекция без слияния движений в вершине угла.

### **6.4.2      *Обработка внешнего угла***

При обработке внешнего угла совершается дополнительное движение по дуге окружности для обхода вокруг угла (см. Рис.19, 20). Начальными и конечными точками дуги являются точки, смещенные от запрограммированной точки геометрической вершины угла, на величину радиуса инструмента. Дуга имеет центр в заданной программой точке угла.

При покадровом выполнении программы и активном режиме коррекции, сначала производится дополнительное движение по дуге вокруг внешнего угла, затем выполняется останов программы.

#### **Примечание:**

*Для внешнего угла, с изменением угла меньше чем на 1 градус, дополнительное движение по дуге не выполняется, просто происходит слияние движений (см. рис. 3,4).*

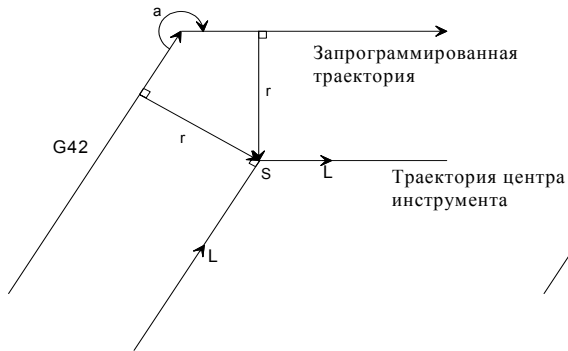
### **6.4.3      *Изменение направления коррекции***

Когда изменяется направление коррекции (с левого на правое или наоборот), изменяется местоположение конечной точки движения (удаляется или сближается) для того, чтобы следующее движение начиналось с правильной стороны угла. Путь движения к этой точке не изменяется. Когда изменение коррекции вводится на участке линейного движения, то путь коррекции инструмента будет выполняться по диагонали к запрограммированному пути инструмента. Когда изменение коррекции вводится на участке движения по дуге окружности, то коррекция пути инструмента будет выполняться по спирали (см. Рис 21, 22).

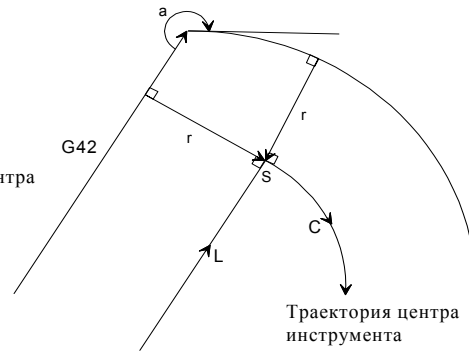
## Режим Коррекции

а) Для внутреннего угла

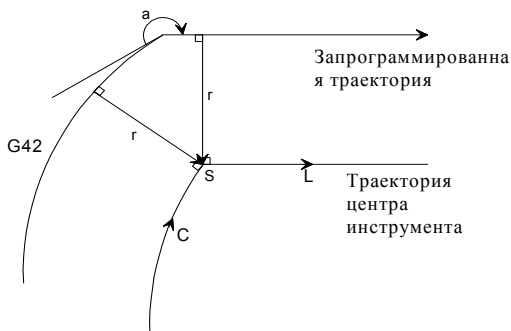
1) Линия → Линия



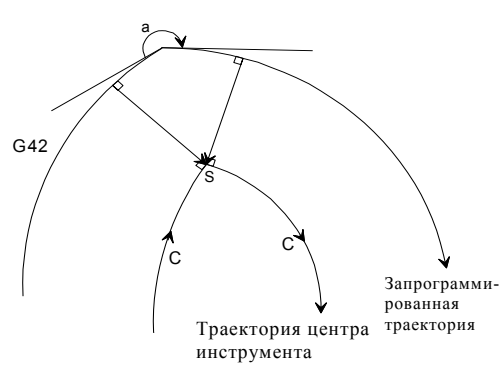
2) Линия → Дуга



3) Дуга → Линия



4) Дуга → Дуга



5) Для внутреннего острого угла Линия → Линия

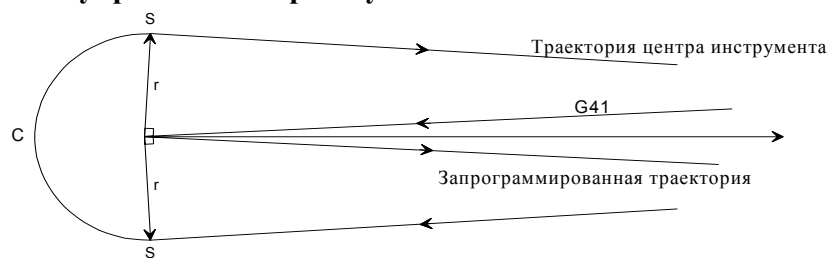
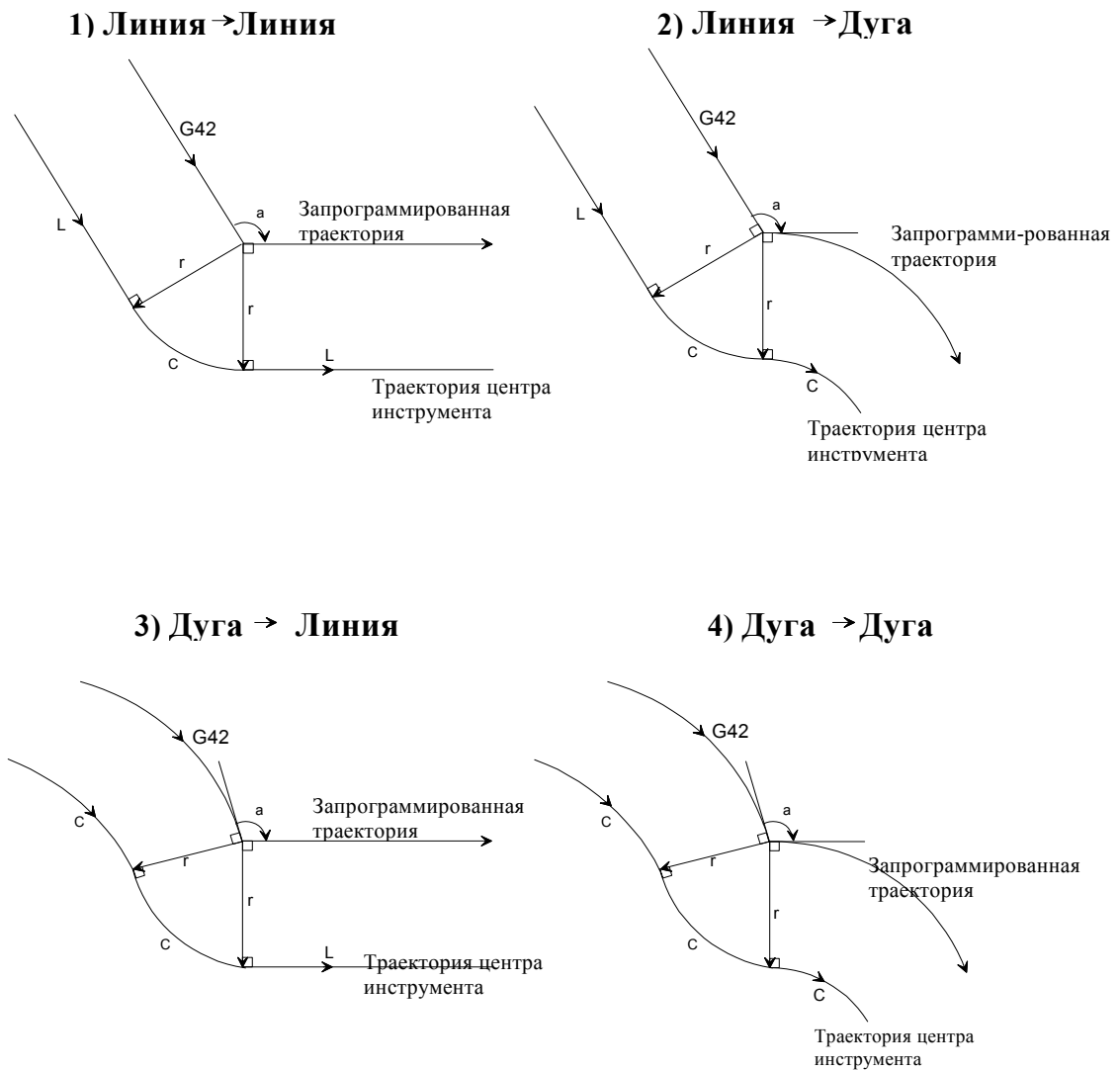


Рис. 18 Режим эквидистантного движения с коррекцией на инструмент.  
Варианты: а) – [1, 2, 3, 4, 5]

## Режим Коррекции

б) Для внешнего тупого угла



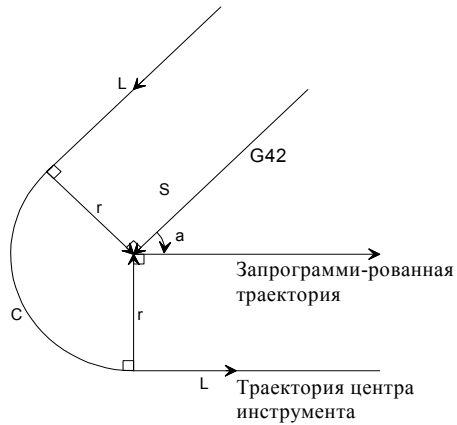
Когда угол изменения направления меньше  $1^\circ$  ( $a > 179^\circ$ ), сегмент дуги не добавляется. Происходит простое движение со слиянием.

**Рис. 19** Режим эквидистантного движения с коррекцией на инструмент.  
 Варианты: б) – [1, 2, 3, 4]

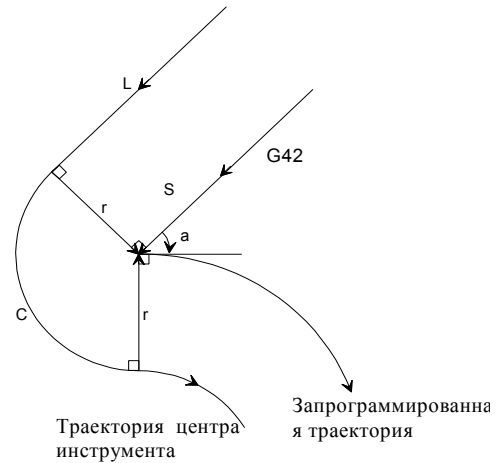
## Режим Коррекции

с) Для внешнего острого угла

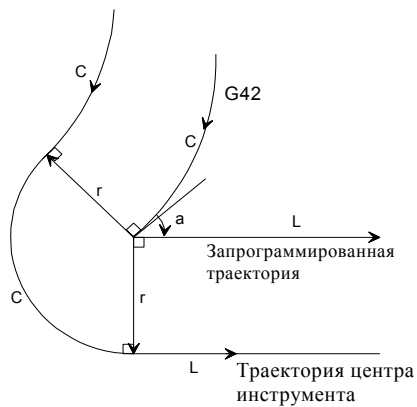
1) Линия → Линия



2) Линия → Дуга



3) Дуга → Линия



4) Дуга → Дуга

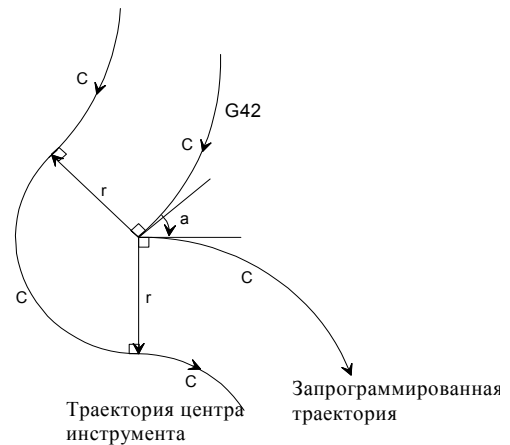


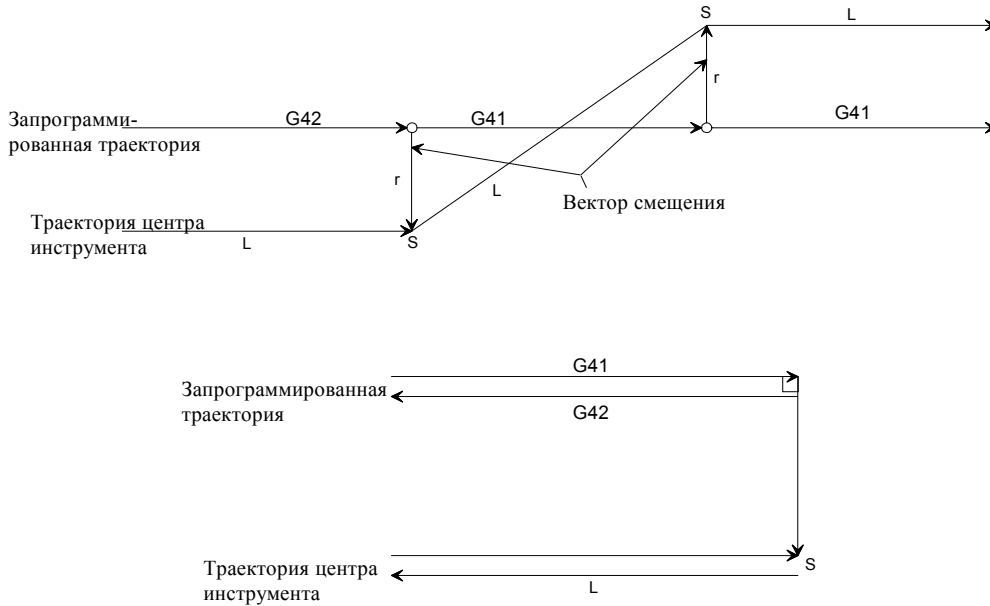
Рис. 20 Режим эквидистантного движения с коррекцией на инструмент.  
Варианты: с) – [1, 2, 3, 4]



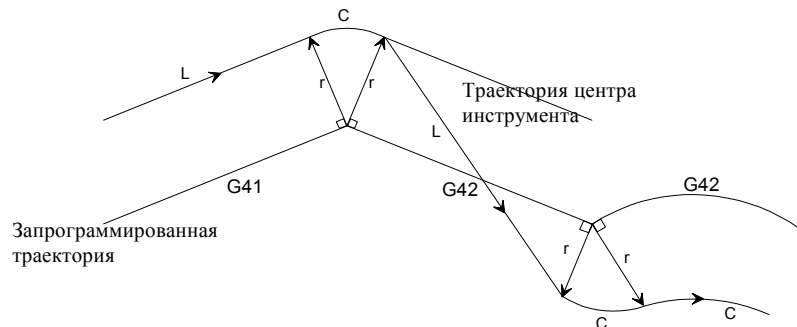
## Изменение Направления Коррекции

### 5) Пересечение отсутствует

#### 1. Линия → Линия



#### 2. Линия → Дуга



$r$  = Радиус компенсации  
 $S$  = Точка сопряжения  
 $L$  = Линия  
 $C$  = Дуга

**Рис. 22 Эквидистантное движение при изменении направления коррекции на инструмент (с левого на правое и наоборот). Варианты: 1., 2.**

### 6.5 Отмена коррекции

Режим коррекции отменяется с помощью функции **G40**.

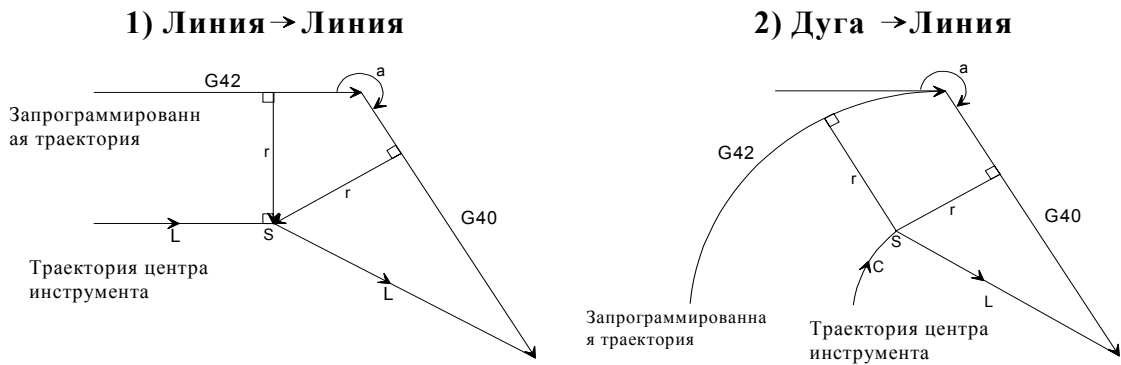
При отмене режима коррекции, должно быть обеспечено движение для выхода с эквидистантной траектории (см. рис. 23, 24 и 24А). Если такое движение пропущено после **G40**, то СЧПУ не отменит коррекцию на радиус, пока кадр с движением по одной или двум осям в плоскости коррекции не будет выполнен.

***Внимание:*** Так как отмена режима КРИ вызывает движение по одной или двум осям, не рекомендуется выполнять эту отмену в кадрах, где выполняется процесс резания

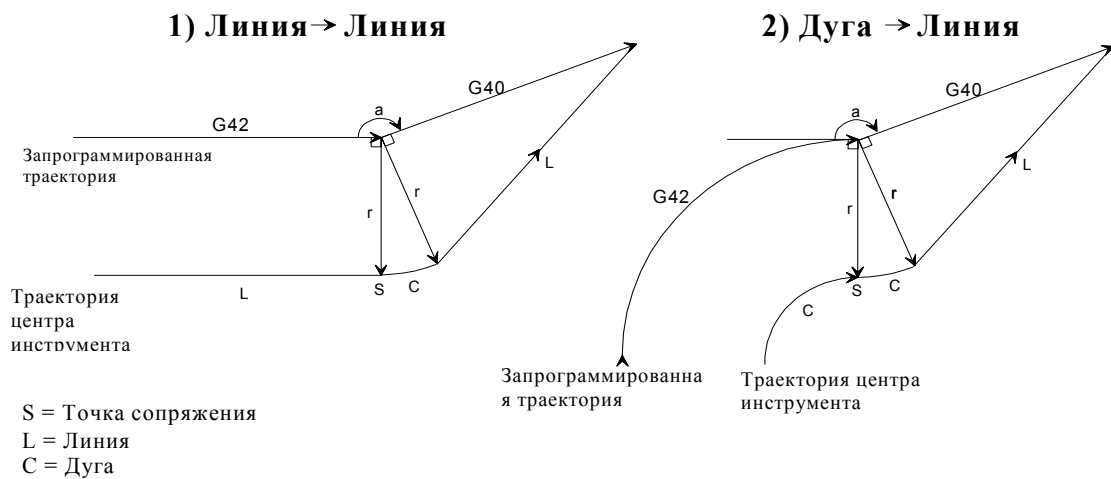
При задании параметра **R** с функцией **G40**, выход из режима эквидистантного движения выполняется по дуге окружности (см. рис. 24А). При этом инструмент, по дуге окружности радиуса **R**, отходит от поверхности заготовки и затем по линии отходит в заданную в функции **G40** координату.

## Отмена Коррекции

### а) Для внутреннего угла



### б) Для внешнего угла

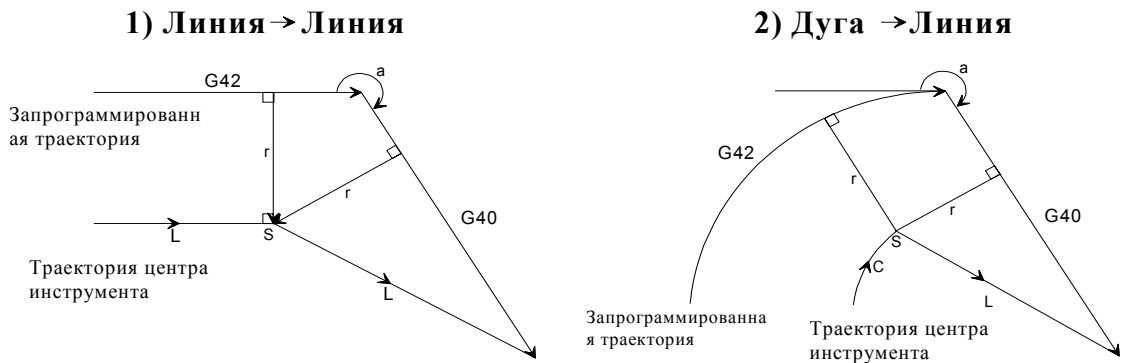


**Рис. 23 Выход из эквидистантного движения при отмене коррекции на инструмент. Варианты: а) – [1, 2] и б) – [1, 2]**

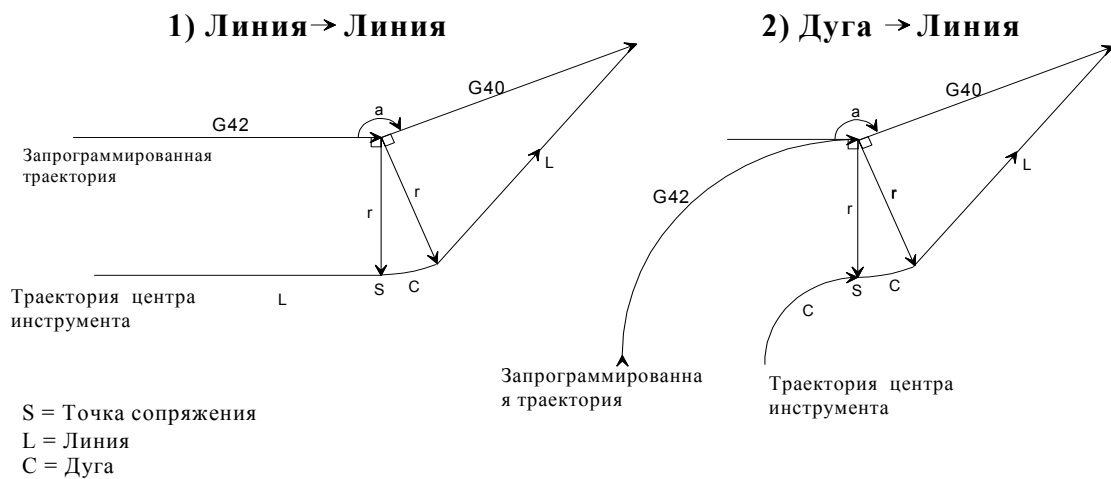


## Отмена Коррекции

### а) Для внутреннего угла



### б) Для внешнего угла



**Рис. 24 Выход из эквидистантного движения при отмене коррекции на инструмент. Варианты: с) – [1, 2] и d)**

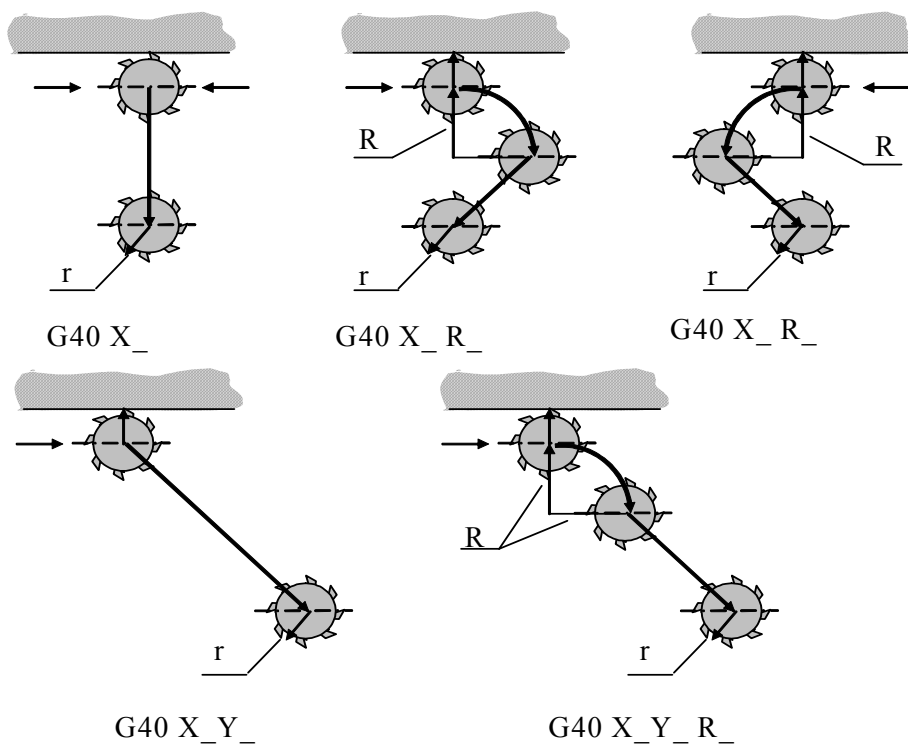
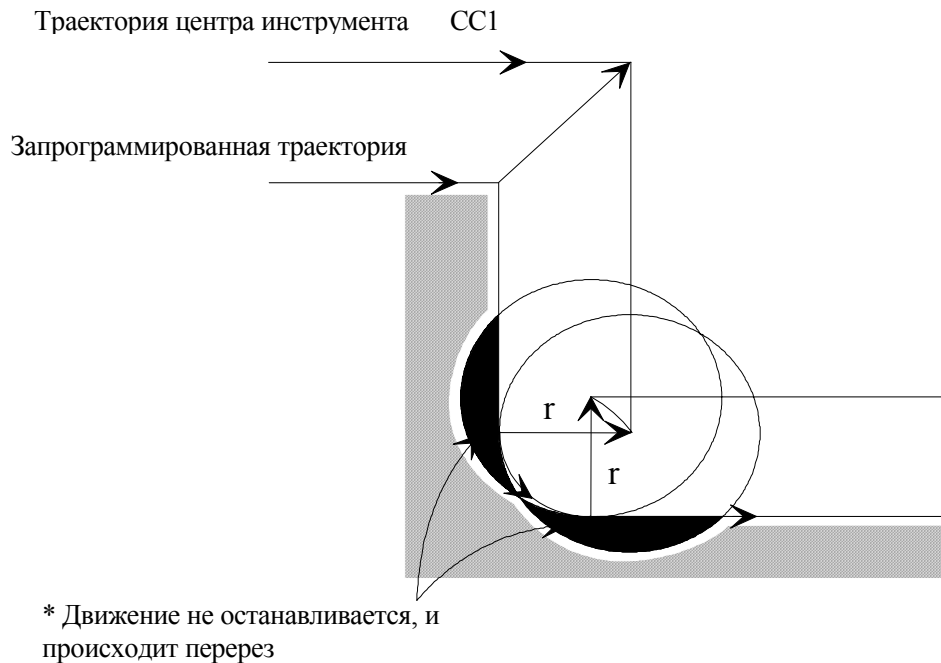


Рис 24А. Выход с эквидистантной траектории по дуге окружности. Примеры

## Перерез при коррекции

1) Обработка внутреннего угла по радиусу, меньшего диаметра инструмента



2) Обработка паза, когда ширина меньше диаметра инструмента

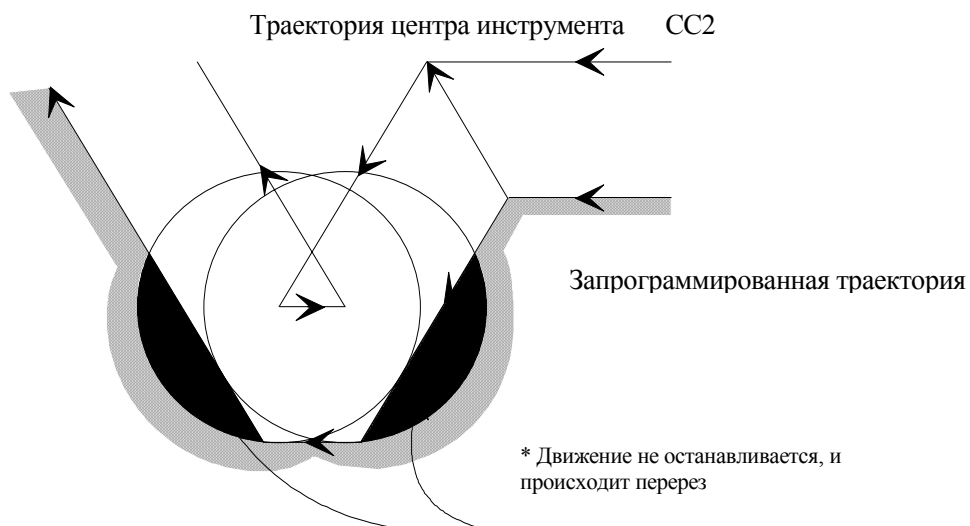
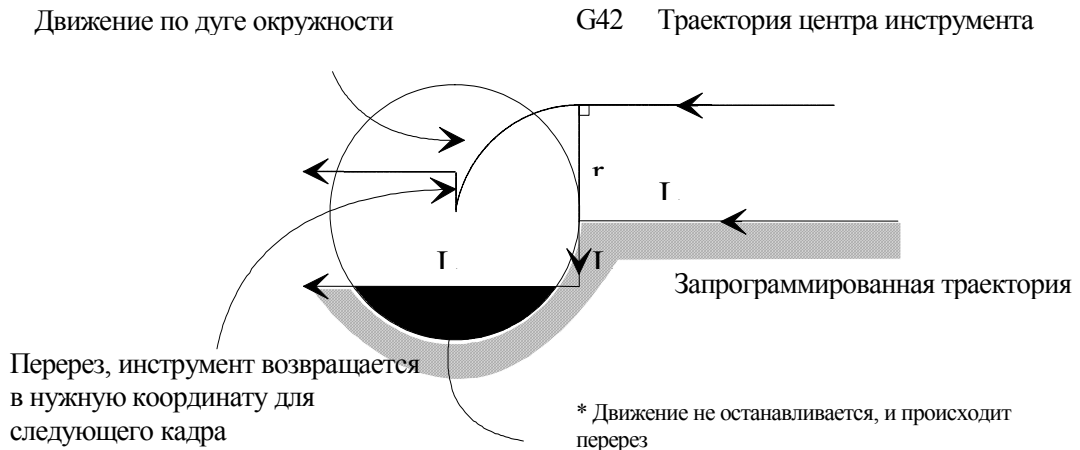


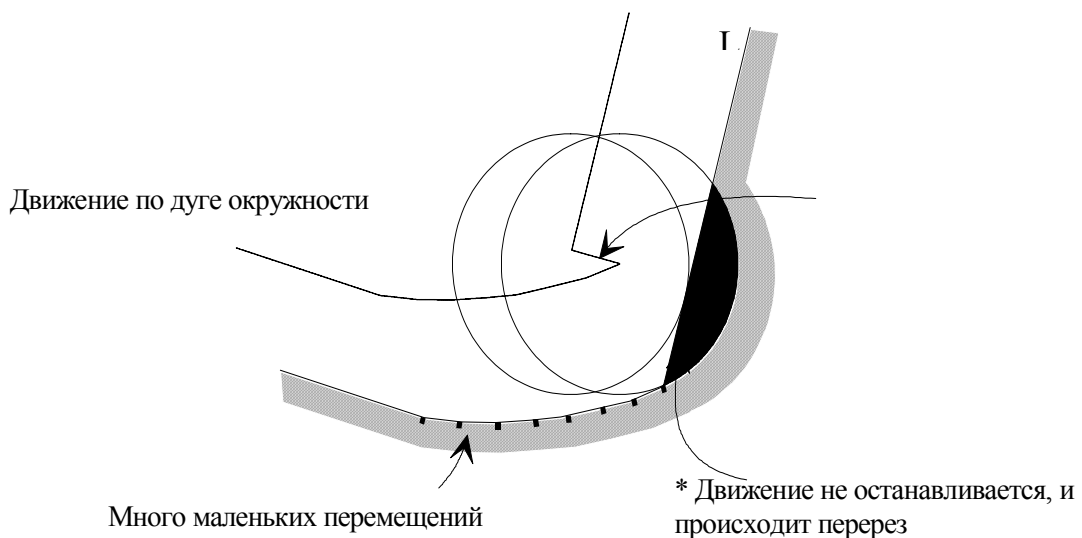
Рис 25 Перерез при коррекции на инструмент

## Перерез при коррекции

### 3) Движение в кадре меньше, чем радиус инструмента



### 4. Обработка малыми перемещениями



СЧПУ вычисляет траекторию на 2 кадра вперед. Поэтому, если радиус коррекции инструмента значительно больше, чем перемещения в кадре, может произойти перерез.

Рис 26 Перерез при коррекции на инструмент

## 7. КОРРЕКЦИЯ НА ПОЛОЖЕНИЕ ИНСТРУМЕНТА

### 7.1 Коррекция на положение инструмента +/- и отмена (G43/G44/G49)

Программный нуль является центром координатной системы детали, и используется при написании программы и при обработке заготовки. При смене инструмента координата центра инструмента по осям **X** и **Y** не изменяется. Но так как инструменты могут быть разной длины, изменится расстояние от кончика режущей кромки до программного нуля по оси **Z**. Поэтому каждый инструмент будет иметь отличное от других расстояние от кончика режущей кромки до поверхности детали.

Коррекция положения инструмента используется для компенсации разницы программной и реальной положений инструмента.

В основном используется коррекция инструмента по длине, которая действует по оси **Z**. Обычно для фрезных станков значения смещений по осям **X, Y** не задают и оставляют их равными нулю.

Активация корректора инструмента по длине выполняется функциями **G43, G44**.

Формат:

**G43 Z\_ D\_** ; коррекция длины инструмента в положительном направлении

**G44 Z\_ D\_** ; коррекция длины инструмента в отрицательном направлении

**G49** ; отмена коррекции

Где: **Z\_** - координата точки, в которую будет выполняться движение. Этот параметр может отсутствовать

**D\_** - номер корректора, в котором заданы значения смещений для коррекции положения инструмента. Этот параметр может отсутствовать. В этом случае устанавливается или предыдущий заданный корректор или, если была задана функция смены инструмента по **M6**, то номер корректора **D** устанавливается равным номеру текущего инструмента **T**. Более подробно см п. 1.8.

Параметр **D** может быть задан в любом месте УП

По функции **G43** значения смещений корректора **D** прибавляются к текущим координатам, а по функции **G44** значения смещений корректора **D** вычитаются из текущих координат.

Активизацию коррекции на положение инструмента необходимо делать при первом подходе инструмента к заготовке после смены инструмента.

Подготовительные функции **G43/G44** взаимоисключающие. Установленные однажды они действуют до тех пор, пока не будут переустановлены или отменены. Подготовительная функция **G49** отменяет коррекцию на длину инструмента.

#### Пример:

...

T2M6

G54 G17 X20 Y20

G43

G0 Z30 ; активизация корректора D2 , по номеру инструмента

G1 F200 Z-50

G0Z30

T5M6

G43 D7 Z20 ; активизация корректор D7

G0 Z30

G1 F200 Z-50

## 8. ПОСТОЯННЫЕ ЦИКЛЫ СВЕРЛЕНИЯ И РАСТОЧКИ.

Постоянные циклы G80 – G86 упрощают программирование благодаря использованию одной подготовительной функции для задания таких станочных операций, которые требуют несколько кадров программы СЧПУ. Постоянные циклы G80 – G86 предназначены для сверления, растачивания и нарезания резьбы.

Эти циклы используют ось **Z** как ось сверления и расточки, и плоскость **X,Y** (G17) как плоскость позиционирования. Постоянные циклы (см Рис. 27) представляют последовательности ряда операций приведенных ниже:

1. Позиционирование на начальном уровне на быстром ходу по осям **X,Y** в центр отверстия;
2. Позиционирование на быстром ходу на опорный уровень (**R** - уровень) по оси **Z**;
3. Обработка отверстия на рабочей подаче на заданную глубину (**Z**-уровень глубины обработки) по оси **Z**;
4. Обработка дна отверстия – выполнение дополнительных операций (пауза, стоп вращения шпинделя, реверс);
5. Возврат по оси **Z** на опорный уровень **R**;
6. Возврат по оси **Z** на начальный уровень.

### Примечание:

*Начальный уровень определяется координатой **Z** предыдущего кадра перед заданием постоянного цикла.*

*Опорный уровень **R** - это позиция по координате **Z**, с которого начинается рабочая подача для обработки.*

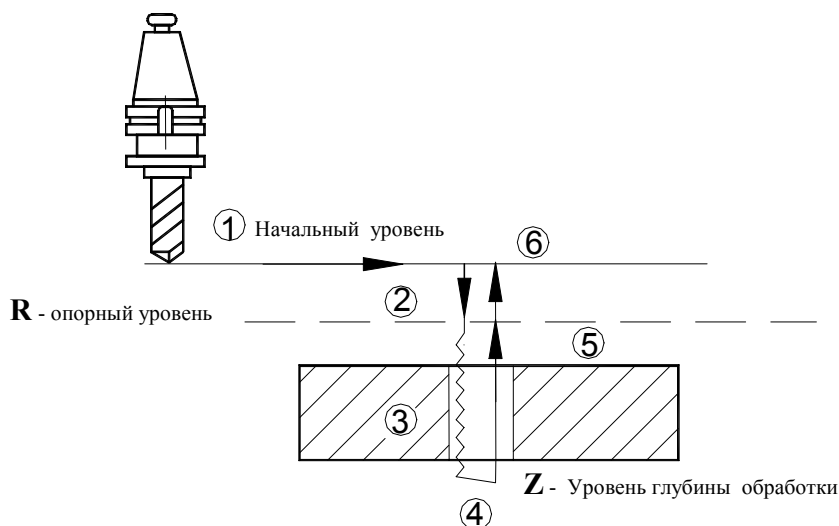


Рис. 27. Типовая последовательность операций для постоянного цикла

### 8.1. Отмена постоянного цикла G81-G86 (G80)

Функция **G80** отменяет действие активного постоянного цикла.

## 8.2. Игнорировать последующий постоянный цикл G81-G88 (G180)

Подготовительная функция G180 блокирует выполнение следующего за ней постоянного цикла (G81-G88).

### Примечание:

*В последующем постоянном цикле происходит перемещение в позицию X, Y и выход в опорный уровень по координате Z. Непосредственно цикл не выполняется.*

*Следующие постоянные циклы выполняются целиком.*

## 8.3. Возврат из постоянного цикла (G98/G99)

Подготовительные функции G98 и G99 определяют уровень возврата инструмента по оси Z после обработки: на опорный уровень (R) или на начальный уровень. Когда активна подготовительная функция G98, то возврат инструмента осуществляется в начальный уровень Z. В противном случае, когда активна подготовительная функция G99, возврат осуществляется на опорный уровень R.

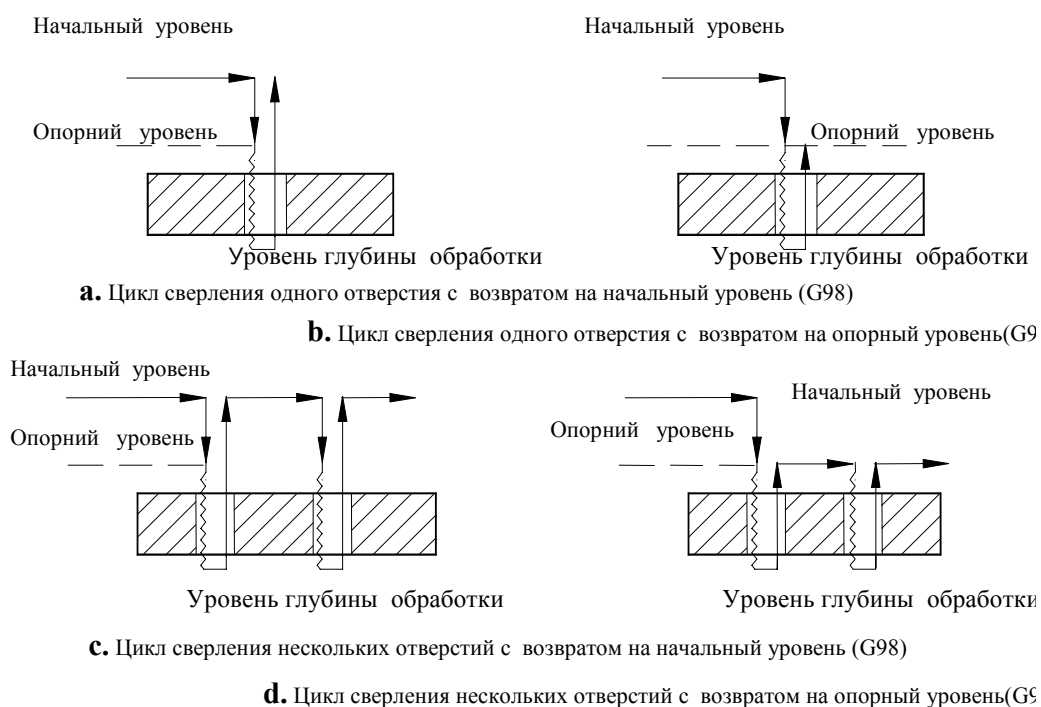
### Примечание:

*По умолчанию активна подготовительная функция G98.*

*Действие подготовительной функции G98/G99 распространяется на постоянные циклы сверления и расточки и на многооперационные циклы.*

На рис. 28а и рис. 28с приведена последовательность движений с возвратом на начальный уровень (G98) при обработке одного или нескольких отверстий.

На рис. 28b и рис. 28d приведена последовательность движений с возвратом на опорный уровень (G99) при обработке одного или нескольких отверстий.



**Рис. 28. Режим выбора точки возврата**

Пример использования G98 и

```

...
G0 X0 Y0 Z20
    Z10
G98
G81 X10 Y10 R5 Z-20 ; Возврат на начальный уровень Z10
    X40 ; Возврат на начальный уровень Z10
...
G99
    Y50 ; Возврат на опорный уровень R5
G80 ; Отмена постоянного цикла
    
```

#### 8.4. Сверление с выводом инструмента на быстром ходу (G81)

Цикл **G81** предназначен для сверления отверстий. При работе цикла выполняется следующая последовательность движений (см. Рис. 29):

1. Инструмент на быстром ходу перемещается в центр отверстия, определенный параметрами **X** и **Y**;
2. Выполняется быстрое перемещение по оси **Z** на опорный уровень **R**;
3. Выполняется сверление (расточивание) на рабочей подаче на заданную глубину (уровень глубины обработки) по оси **Z**;
4. Выполняется возврат на быстрой подаче на опорный уровень **R**;

#### Примечание:

*Если активна подготовительная функция G98, то возврат инструмента осуществляется в начальный уровень Z. В противном случае, когда активна подготовительная функция G99, возврат осуществляется на опорный уровень R.*

*Цикл будет выполняться в каждом кадре, содержащем движения по осям X и/или Y, пока постоянный цикл не будет отменен функцией отмены постоянного файла G80.*

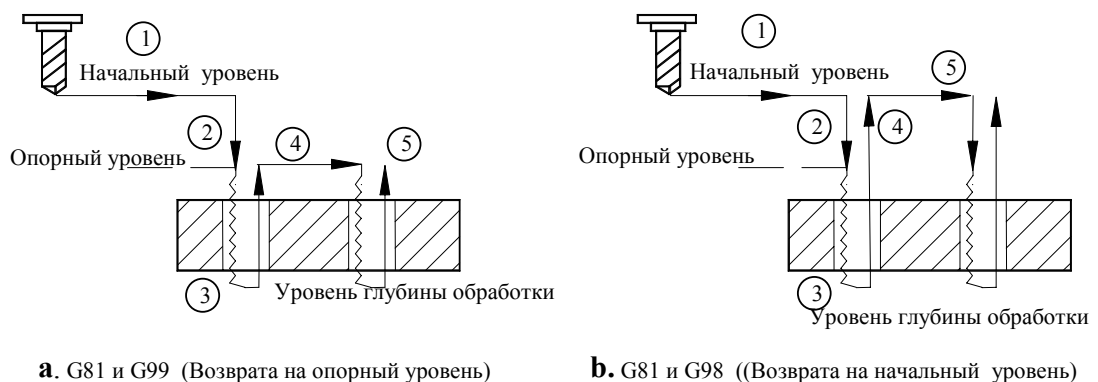


Рис. 29. Последовательность операций при G81

Подготовительная функция G81 имеет следующий формат:

**G81 X\_ Y\_ Z\_ R\_ F\_ L\_**



Где:

**X** – координата центра отверстия по оси **X**;

**Y** – координата центра отверстия по оси **Y**;

**Z** – глубина сверления;

**R** – координата опорного уровня по оси **Z**;

**F** – скорость подачи при обработке;

**L** – число повторов.

Параметры **X,Y,R,F,L** являются необязательными.

При отсутствии параметра **L** - число повторов равно 1.

При отсутствии параметра **R** - значение опорного уровня равняется значению начального уровня.

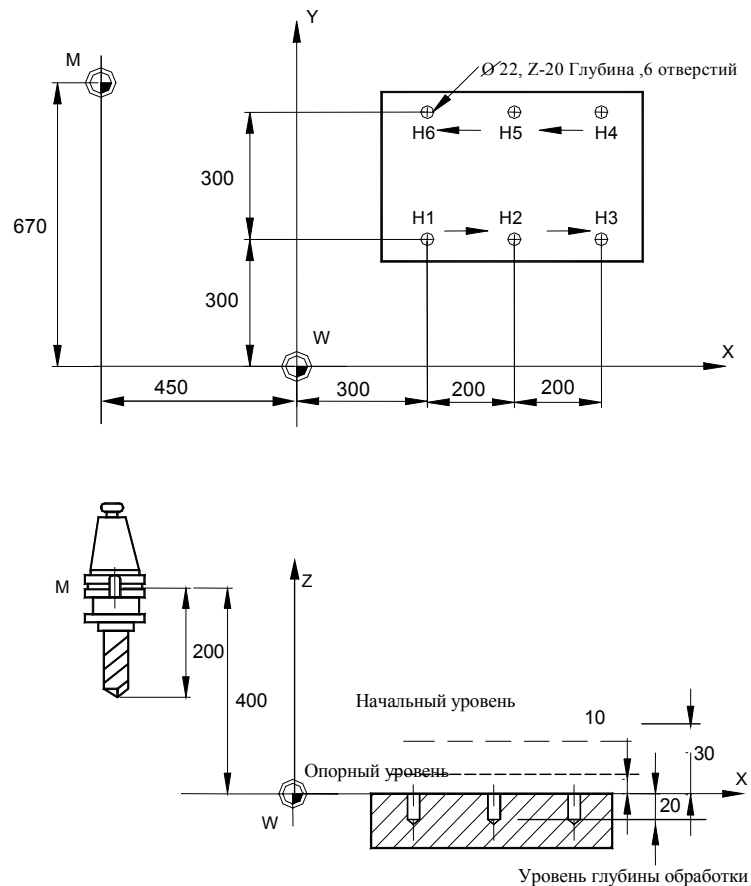
При отсутствии параметров **X,Y** цикл сверления выполняется в текущей точке.

Параметр **Z** обязателен. При отсутствии параметра **Z** выводится сообщение о синтаксической ошибке.

**Пример:**

программы с применением цикла G81 для детали на **Рис. 30.** приведен ниже:

```
G90 G54 G80 G40
T7 M06
S750 M03
G43
G0 X0 Y0
G0 Z100
G99 G81 X300 Y300 Z-20 R10 F100
    X500
    X700
    Y600
    X500
    X300
G80 G90 Z100 M05
    X0 Y0
M30
```



**Рис. 30** Пример использования цикла G81

### 8.5. Растачивание (сверление) с паузой перед выводом (G82)

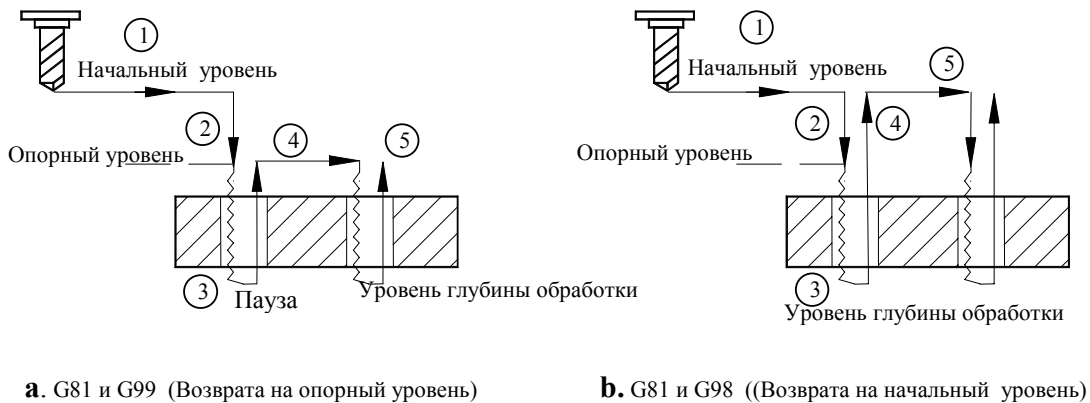
Подготовительную G82 функцию часто называют циклом встречного растачивания. Этот цикл также используется при сверлении отверстий. Он функционирует аналогично циклу G81, но включает функцию паузы. При работе цикла выполняется следующая последовательность движений (см Рис. 31):

1. Инструмент на быстром ходу перемещается в центр отверстия, определенный параметрами X и Y;
2. Выполняется быстрое перемещение по оси Z на опорный уровень R;
3. Выполняется сверление (расточивание) на рабочей подаче на заданную глубину (уровень глубины обработки) по оси Z;
4. Выполняется пауза в конце отверстия после обработки;
5. Выполняется возврат на быстрой подаче на опорный уровень R;

**Примечание:**

*Если активна подготовительная функция G98, то возврат инструмента осуществляется в начальный уровень Z. В противном случае, когда активна подготовительная функция G99, возврат осуществляется на опорный уровень R.*

*Цикл будет выполняться в каждом кадре, содержащем движения по осям X и/или Y, пока постоянный цикл не будет отменен функцией отмены постоянного файла G80.*



**Рис.31. Последовательность операций при G82**

Подготовительная функция **G82** имеет следующий формат:

**G82 X\_ Y\_ Z\_ R\_ F\_ L\_ P\_**,

Где:

**X** – координата центра отверстия по оси **X**;

**Y** – координата центра отверстия по оси **Y**;

**Z** – глубина сверления;

**R** – координата опорного уровня по оси **Z**;

**F** – скорость подачи при обработке;

**L** – число повторов.

**P** – длительность паузы в секундах в нижней позиции отверстия.

Параметры **X,Y,R,F,L,P** являются необязательными.

При отсутствии параметра **L** - число повторов равно 1.

При отсутствии параметра **R** - значение опорного уровня равняется значению начального уровня.

При отсутствии параметров **X,Y** цикл сверления выполняется в текущей точке.

При отсутствии параметра **P** - пауза равна нулю.

Параметр **Z** обязателен. При отсутствии параметра **Z** выводится сообщение о синтаксической ошибке.

### 8.6. Сверление с периодическим выводом инструмента (G83)

При сверлении глубоких отверстий используется постоянный цикл **G83** с периодическим выводом инструмента из просверленного отверстия.

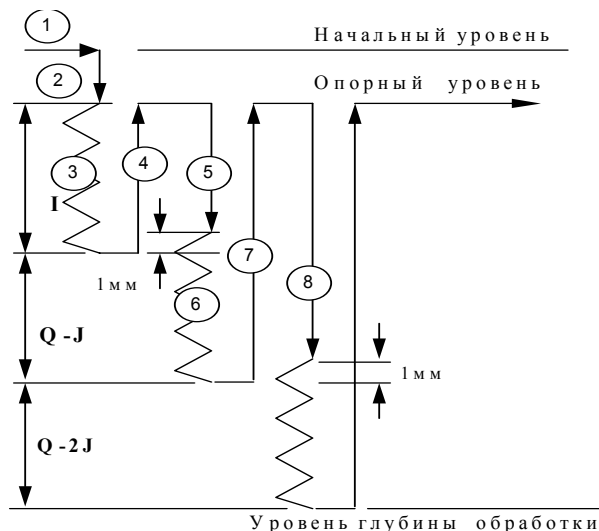
При работе цикла **G83** выполняется следующая последовательность движений (рис. 32):

1. Инструмент на быстром ходу перемещается в центр отверстия, определенный параметрами **X** и **Y**;
2. Выполняется быстрое перемещение по оси **Z** на опорный уровень **R**;
3. Выполняется сверление по заданной подаче на глубину первого уровня, определенного переменной **I**;
4. Выполняется пауза в **P** секунд на дне отверстия и возврат на быстрой подаче на опорный уровень **R**;
5. Выполняется быстрое перемещение по оси **Z** в позицию предыдущий уровень сверления минус 1мм, чтобы не произошел удар о дно отверстия.
6. Выполняется сверление по заданной скорости подачи по оси **Z** на глубину **Q-nJ**, где **n** принимает значение от **1** до **n**;
7. Выполняется пауза в **P** секунд на дне отверстия и возврат на быстрой подаче на опорный уровень **R**;
8. Повторение операций **5-7** до достижения глубины отверстия.

**Примечание:**

Если активна подготовительная функция **G98**, то возврат инструмента осуществляется в начальный уровень **Z**. В противном случае, когда активна подготовительная функция **G99**, возврат осуществляется на опорный уровень **R**.

Цикл будет выполняться в каждом кадре, содержащем движения по осям **X** и/или **Y**, пока постоянный цикл не будет отменен функцией отмены постоянного файла **G80**



**Рис.32. Последовательность операций при G83**

Подготовительная функция **G83** имеет следующий формат:

**G83 X\_ Y\_ Z\_ R\_ F\_ L\_ P\_ Q\_ I\_ J\_ K\_**

Где:

- X** – координата центра отверстия по оси **X**;
- Y** – координата центра отверстия по оси **Y**;
- Z** – глубина сверления;
- R** – координата опорного уровня;

- F** – скорость подачи при обработке;  
**L** – число повторов;  
**P** – длительность паузы в секундах в нижней позиции отверстия;  
**Q** – величина шага сверления;  
**I** – величина первого шага сверления;  
**J** – величина, на которую уменьшается шаг **Q** для каждого следующего рабочего прохода;  
**K** – минимальное значение, до которого может уменьшиться **Q**.

Параметры **X,Y,R,F,L,I,J,K** являются необязательными.

При отсутствии параметра **L** - число повторов равно 1.

При отсутствии параметра **R** - значение опорного уровня равняется значению начального уровня.

При отсутствии параметров **X,Y** цикл сверления выполняется в текущей точке.

При отсутствии параметра **P** - пауза равна нулю.

При отсутствии параметра **I** глубина первого шага сверления равна **Q**.

При отсутствии параметра **J** величина уменьшения равна нулю.

При отсутствии параметра **K** минимальное значение шага сверления равно **Q**.

Параметры **Z,Q** обязательны. При отсутствии параметров **Z,Q** выводится сообщение о синтаксической ошибке.

**Пример:**

Рассмотрим деталь изображенную на **Рис. 33** с двумя отверстиями на глубину 30мм и двумя отверстиями на глубину 120 мм. С помощью цикла **G81** и инструмента **T1** осуществляется сверление двух неглубоких отверстий, а с помощью цикла **G83** и инструмента **T2** двух глубоких.

Ниже приведена программа сверления детали.

```
G90 G80 G40
T1 M6
G54 G0 X0 Y0 S500 M3
G0 G43 Z10
G99 G81 X100 Y150 Z30 R5 F50
Y450
G80
M5

T2 M6
G0 X0 Y0
G43 Z10 S500 M3
G99 G83 X700 Y450 Z-120 Q30 J3 R5 F50
      X400 Y150
G80 M5
G0 X0 Y0
M30
```

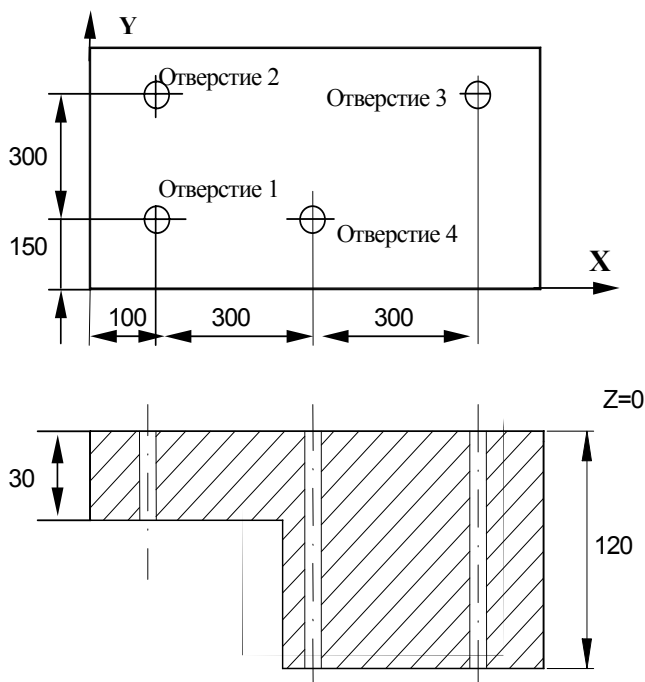


Рис. 33. Пример использования постоянных циклов G81 и G83

### 8.7. Сверление с периодическим отскоком инструмента (G183)

Для сверления глубоких отверстий можно использовать постоянный цикл **G183** с периодическим отскоком инструмента от дна отверстия, без выхода сверла на начальный или опорный уровень.

Цикл **G183** похож на цикл **G83**, но в отличие от него после каждого этапа обработки сверло не выводится на опорный уровень, а выполняется отскок от дна отверстия на 1 мм.

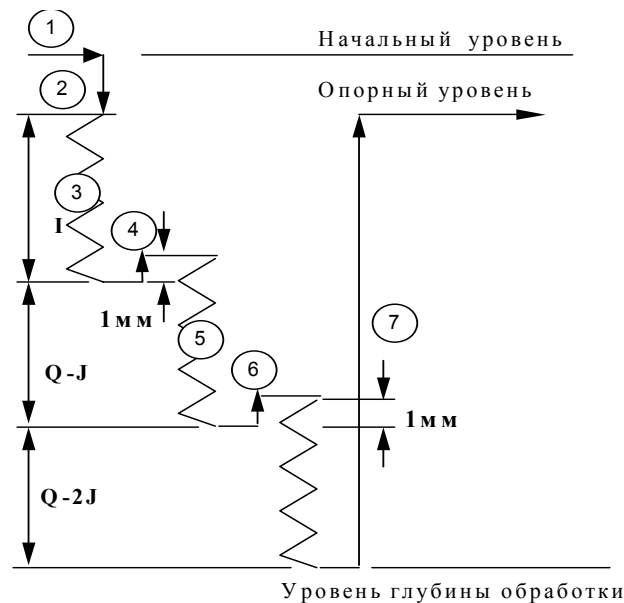
При работе цикла **G183** выполняется следующая последовательность движений (рис. 34):

1. Инструмент на быстром ходу перемещается в центр отверстия, определенный параметрами **X** и **Y**;
2. Выполняется быстрое перемещение по оси **Z** на опорный уровень **R**;
3. Выполняется сверление по заданной подаче на глубину первого уровня, определенного адресом **I**;
4. Выполняется пауза в **P** секунд на дне отверстия и отвод по оси **Z** на **1мм**.
5. Выполняется сверление по заданной скорости подачи по оси **Z** на глубину **Q-nJ**, где **n** принимает значение от **1** до **n**.;
6. Пауза в **P** секунд на дне отверстия и быстрый отвод по оси **Z** на **1мм**.
7. Повторение операций 5-6 до достижения глубины отверстия и быстрый вывод по оси **Z** на опорный уровень, или начальный уровень.

**Примечание:**

*Если активна подготовительная функция G98, то возврат инструмента осуществляется в начальный уровень Z. В противном случае, когда активна подготовительная функция G99, возврат осуществляется на опорный уровень R.*

*Цикл будет выполняться в каждом кадре, содержащем движения по осям X и/или Y, пока постоянный цикл не будет отменен функцией отмены постоянного файла G80.*



**Рис. 34** Последовательность операций при G183

Подготовительная функция **G183** имеет следующий формат:

**G183 X\_ Y\_ Z\_ R\_ F\_ L\_ P\_ Q\_ I\_ J\_ K\_**

Где:

- X** – координата центра отверстия по оси X;
- Y** – координата центра отверстия по оси Y;
- Z** – глубина сверления;
- R** – координата опорного уровня;
- F** – скорость подачи при обработке;
- L** – число повторов;
- P** – длительность паузы в секундах в нижней позиции отверстия;
- Q** – величина шага сверления;
- I** – величина первого шага сверления;
- J** – величина, на которую уменьшается шаг **Q** для каждого следующего рабочего прохода;
- K** – минимальное значение, до которого может уменьшиться **Q**.

Параметры **X,Y,R,F,L,I,J,K** являются необязательными.

При отсутствии параметра **L** - число повторов равно 1.

При отсутствии параметра **R** - значение опорного уровня равняется значению начального уровня.

При отсутствии параметров **X,Y** цикл сверления выполняется в текущей точке.

При отсутствии параметра **P** - пауза равна нулю.

При отсутствии параметра **I** глубина первого шага сверления равна **Q**.  
При отсутствии параметра **J** величина уменьшения равна нулю.  
При отсутствии параметра **K** минимальное значение шага сверления равно **Q**.  
Параметры **Z, Q** обязательны. При отсутствии параметров **Z, Q** выводится сообщение о синтаксической ошибке.

### **8.8. Нарезание резьбы с отводом на рабочей подаче и реверсе (G84)**

Постоянный цикл **G84** предназначен для нарезания резьбы в отверстиях, которые были просверлены с заданным диаметром.

При работе цикла **G84** выполняется следующая последовательность движений (см. **Рис.35**):

1. Инструмент на быстром ходу перемещается в центр отверстия, определенный параметрами **X** и **Y**;
2. Выполняется быстрое перемещение по оси **Z** на опорный уровень **R**;
3. Выполняется нарезание резьбы со скоростью подачи **F** по оси **Z** до уровня глубины обработки;
4. Останов вращения шпинделя на дне отверстия;
5. Выполняется пауза в **P** секунд на дне отверстия;
6. Реверс шпинделя;
7. Возврат на рабочую подачу по оси **Z** на опорный уровень;
8. Выполняется пауза в **Q** секунд на опорном уровне;
9. Останов шпинделя.

#### **Примечание:**

*Если активна подготовительная функция **G98**, то возврат инструмента осуществляется в начальный уровень **Z**. В противном случае, когда активна подготовительная функция **G99**, возврат осуществляется на опорный уровень **R**.*

*Цикл будет выполняться в каждом кадре, содержащем движения по осям **X** и/или **Y**, пока постоянный цикл не будет отменен функцией отмены постоянного файла **G80***

Подготовительная функция **G84** имеет следующий формат:

**G84 X\_ Y\_ Z\_ R\_ F\_ L\_ P\_ Q\_**

Где:

**X** – координата центра отверстия по оси **X**;

**Y** – координата центра отверстия по оси **Y**;

**Z** – глубина нарезания резьбы;

**R** – координата опорного уровня;

**F** – скорость подачи при резьбе (в единицах мм/мин или оборот/мин);

**L** – число повторов;

– длительность паузы в секундах в нижней позиции отверстия;

**Q** – длительность паузы в секундах на опорном уровне, после выхода из отверстия.

Параметры **X, Y, R, F, L, P, Q** являются необязательными.

При отсутствии параметра **L** - число повторов равно 1.



При отсутствии параметра **R** - значение опорного уровня равняется значению начального уровня.

При отсутствии параметров **X,Y** цикл сверления выполняется в текущей точке.

При отсутствии параметра **P** - пауза равна нулю.

При отсутствии параметра **Q** - пауза равна нулю.

Параметр **Z** обязателен, При отсутствии параметра **Z** выводится сообщение о синтаксической ошибке.

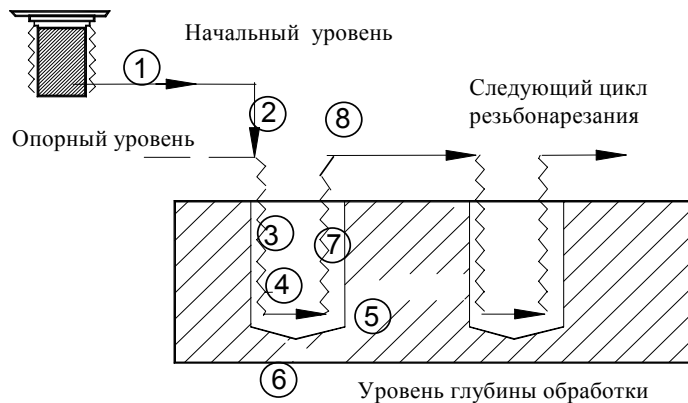


Рис. 35 Постоянный цикл нарезания резьбы G84.

**Пример:**

Рассмотрим пример резьбонарезания с метчиком диаметром 12 мм и с шагом 1.5 мм на глубину 15мм со скоростью 150 об/мин.

Скорость подачи метчика определяется следующим образом:

**$P = 1 \text{ мм} - \text{ шаг резьбы}$**

**$f_r = P = 1.5 \text{ мм/об}$**

**$f_m = P \times N = 1.5 \times 150 = 225 \text{ мм/мин}$**

Таким образом, фактическая скорость подачи будет F1.5мм/об для G95 или F225 для G94.

В станках с ЧПУ обычно применяются метчики с плавающими держателями. Эти держатели обладают следующими свойствами:

- компенсируют различия между скоростью подачи метчика и шагом резьбы до  $\pm 15 \text{ мм}$ ;
- позволяют повысить скорость резания;
- сокращают время замены инструмента.

Ниже дана полная программа обработки детали.

G90 G54 G80 G40

T7 M6

S150 M3

G43

G00 X0 Y0

G00 Z20 G95

G98 G84 X300 Y300 Z-15 R10 P2 Q2 F1.5

X500

X700

Y600

X500  
 X300  
 G80 G90 M5  
 G0 Z100  
 X0 Y0  
 M30

### 8.9. Растачивание с отводом на рабочей подаче без реверса (G85)

Подготовительная функция **G85** активизирует двунаправленный цикл расточки. Постоянный цикл **G85** предусмотрен для растачивания при контролируемой подаче как на входе в отверстие, так и на выходе из него. Он используется при прецизионных расточных операциях. Поэтому его называют прецизионным расточным циклом.

При работе цикла **G85** выполняется следующая последовательность движений (см **Рис.36**):

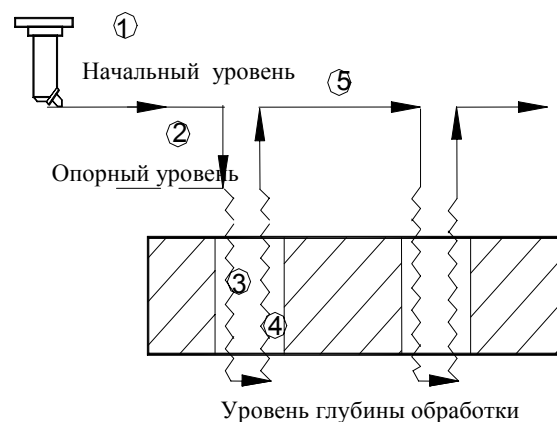
1. Инструмент на быстром ходу перемещается в центр отверстия, определенный параметрами **X** и **Y**;
2. Выполняется быстрое перемещение по оси **Z** на опорный уровень **R**;
3. Рабочая подача со скоростью **F** по оси **Z** на глубину расточки отверстия;
4. Пауза в **P** секунд на дне отверстия;
5. Рабочая подача со скоростью **F** по оси **Z** до позиции опорного уровня;
6. Возврат на быстром ходу на опорный или на начальный уровень;

#### Примечание

*Если активна подготовительная функция **G98**, то возврат инструмента осуществляется в начальный уровень **Z**. В противном случае, когда активна подготовительная функция **G99**, возврат осуществляется на опорный уровень **R**.*

*Цикл будет выполняться в каждом кадре, содержащем движения по осям **X** и/или **Y**, пока постоянный цикл не будет отменен функцией отмены постоянного файла **G80**.*

*В процессе выполнения этого цикла ручные переназначения скорости подачи игнорируются.*



**Рис. 36. Постоянный цикл прецизионной расточки G85**

Подготовительная функция **G85** имеет следующий формат:

**G85 X\_Y\_Z\_R\_F\_L\_P\_**

Где:

**X** – координата центра отверстия по оси **X**;

**Y** – координата центра отверстия по оси **Y**;

**Z** – глубина расточки;

**R** – координата по оси **Z** плоскости возврата;

**F** – скорость подачи при обработке;

**L** – число повторов;

**P** – пауза в нижней части отверстия на уровне **Z** в секундах;

Параметры **X,Y,R,F,L,P** являются необязательными.

При отсутствии параметра **L** - число повторов равно 1.

При отсутствии параметра **R** - значение опорного уровня равняется значению начального уровня.

При отсутствии параметров **X,Y** цикл расточки выполняется в текущей точке.

При отсутствии параметра **P** - пауза равна нулю.

Параметр **Z** обязателен. При отсутствии параметра **Z** выводится сообщение о синтаксической ошибке.

В качестве примера цикла прецизионной расточки рассмотрим деталь, изображенную на **рис. 30**. Предположим, что перед активацией цикла **G85** отверстия были просверлены под диаметр с припуском для прецизионной расточки. Программа будет отличаться от вышеприведенной заменой строки цикла на следующую:

```
G99 G85 X300 Y300 Z-15 R5 F0.5
```

### **8.10. Растачивание с быстрым отводом и остановом шпинделя (G86)**

Подготовительная функция **G86** активизирует цикл расточки с быстрым отводом. При работе цикла **G86** выполняется следующая последовательность движений (**см. Рис.37**):

1. Инструмент на быстром ходу перемещается в центр отверстия, определенный параметрами **X** и **Y**;
2. Выполняется быстрое перемещение по оси **Z** на опорный уровень **R**;
3. Рабочая подача со скоростью **F** по оси **Z** на глубину расточки отверстия;
4. Пауза в **P** секунд на дне отверстия;
5. Возврат на быстром ходу на опорный или на начальный уровень;

#### **Примечание:**

*Если активна подготовительная функция **G98**, то возврат инструмента осуществляется в начальный уровень **Z**. В противном случае, когда активна подготовительная функция **G99**, возврат осуществляется на опорный уровень **R**.*

*Цикл будет выполняться в каждом кадре, содержащем движения по осям **X** и/или **Y**, пока постоянный цикл не будет отменен функцией отмены постоянного файла **G80***

*В процессе выполнения этого цикла ручные переназначения скорости подачи игнорируются.*

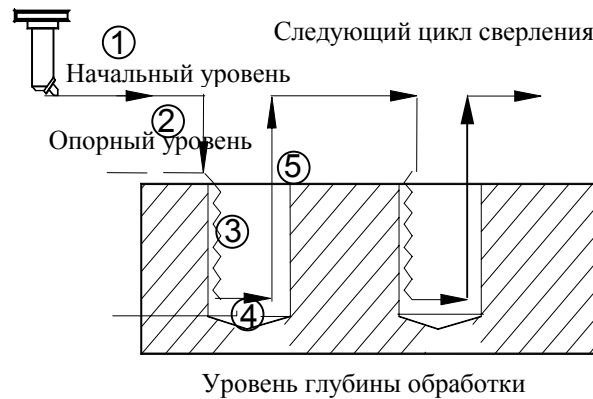


Рис. 37. Цикл растачивания с быстрым отводом G86.

Подготовительная функция G86 имеет следующий формат:

**G86 X\_ Y\_ Z\_ R\_ F\_ L\_ P\_**

Где:

**X** – координата центра отверстия по оси **X**;

**Y** – координата центра отверстия по оси **Y**;

**Z** – глубина расточки;

**R** – координата по оси **Z** плоскости возврата;

**F** – скорость подачи при обработке;

**L** – число повторов;

**P** – пауза в нижней части отверстия на уровне **Z** в секундах;

Параметры **X,Y,R,F,L,P** являются необязательными.

При отсутствии параметра **L** - число повторов равно 1.

При отсутствии параметра **R** - значение опорного уровня равняется значению начального уровня.

При отсутствии параметров **X,Y** цикл расточки выполняется в текущей точке.

При отсутствии параметра **P** - пауза равна нулю.

**Примечание:**

**Параметр Z обязателен. При отсутствии параметра Z выводится сообщение о синтаксической ошибке.**

В качестве примера рассмотрим деталь, изображенную на рис. 30. Предположим, что перед активацией цикла **G86** отверстия были просверлены под диаметр с припуском для расточки. Программа будет отличаться от вышеприведенной заменой строки цикла на следующую:

G99 G86 X300 Y300 Z-15 R5 F0.5

## 9. МНОГООПЕРАЦИОННЫЕ ЦИКЛЫ

Многооперационные циклы, **G700, G701, G702** предназначены для обработки отверстий равномерно расположенных на окружности (**G700**) на дуге окружности (**G701**) или на отрезке прямой линии (**G702**) .

### 9.1 Обработка отверстий по контуру окружности (G700)

Многооперационные циклы **G700** предназначены для обработки отверстий равномерно расположенных на окружности (см. Рис. 38). Этой подготовительной функции должна предшествовать функция постоянного цикла из списка **G81-G86, G183**. Подготовительная функция постоянного цикла, которая предшествует функции **G700**, устанавливает способ обработки отверстий.

Параметры **X\_** и **Y\_** , описанные в кадре с функциями **G81- G86, 183**, определяют центр окружности.

#### Примечание:

*Описание постоянного цикла **G81-G86, G183** не может размещаться в том же кадре, где запрограммирован цикл **G700**.*

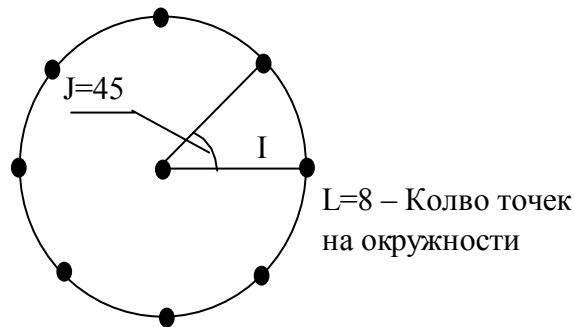


Рис. 38. Многооперационный цикл G700

Подготовительная функция **G700** имеет следующий формат:  
**G700 I\_ J\_ L\_**

Где:

**I** – радиус окружности должен быть больше 0;

**J** – угол между осью **X** и вектором центра первой точки на окружности;

**L** – количество точек на окружности.

Параметр **J** является необязательным.

При отсутствии параметра **J** - угол равен нулю.

При отсутствии параметров **I, L** выводится сообщение о синтаксической ошибке

#### Пример:

В приведенном примере сначала производится сверление отверстий с помощью цикла сверления **G83**, а затем в тех же точках производится нарезание резьбы.

```
G83 X_ Y_ Z_ R_ L_
```

```
G700 I3 J45 L8
```

```
G80
```

```
G84 X_ Y_ Z_ R_ L_ F_ P_ Q_
```

```
G700 I3 J45 L8
```

```
G80
```

## 9.2 Обработка отверстий по контуру дуги окружности (G701)

Многооперационные циклы **G701** предназначены для обработки отверстий равномерно расположенных на дуге окружности (см. рис. 39). Этой подготовительной функции должна предшествовать функция постоянного цикла, например: **G81**, **G82**, **G83**, **G84**, **G85** или **G86**. Подготовительная функция постоянного цикла, которая предшествует функции **G701**, устанавливает способ обработки отверстий.

Параметры  $X_$  и  $Y_$ , описанные в кадре с функциями **G81- G88**, определяют центр, где размещается окружность.

### Примечание:

Описание постоянного цикла **G81-G86,G183** не может размещаться в том же кадре, где запрограммирован цикл **G701**.

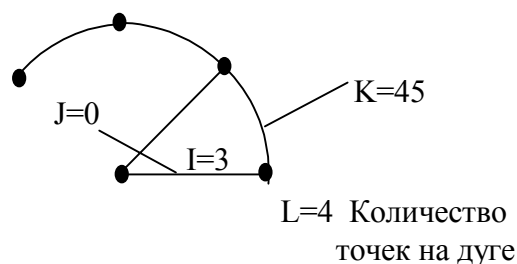


Рис. 39. Многооперационный цикл G701

Подготовительная функция **G701** имеет следующий формат:

**G701 I\_ J\_ K\_ L\_**

Где,

**I** – радиус дуги окружности должен быть больше 0;

**J** – угол между осью **X** и вектором центра первой точки на дуге окружности;

**L** – количество точек на дуге окружности;

**K** – угол между точками на дуге окружности.

Параметры **J,K** являются необязательными.

При отсутствии параметра **J** - угол равен нулю.

При отсутствии параметра **K** - угол между точками равен нулю.

При отсутствии параметров **I,L** выводится сообщение о синтаксической ошибке.

### Пример:

В вышеприведенном примере сначала производится сверление отверстий с помощью цикла сверления отверстий, а затем в тех же точках производится нарезание резьбы.

...

G180

G83  $X_ Y_ Z_ R_ L_$

G701 I3 J0 L8

G80

G180

G84  $X_ Y_ Z_ R_ L_ F_ P_ Q_$

G701 I3 J0 L8

G80

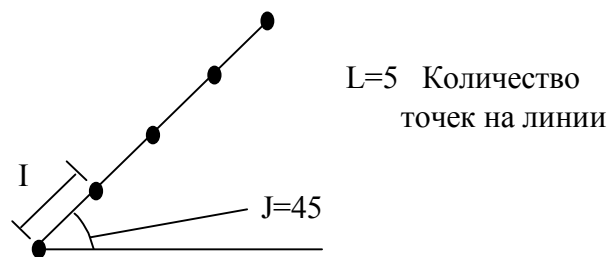
### 9.3 Обработка отверстий по прямолинейному контуру (G702)

Многооперационные циклы **G702** предназначены для обработки отверстий равномерно расположенных на отрезке прямой линии (см. рис. 40). Этой подготовительной функции должна предшествовать функция постоянного цикла, например: **G81**, **G82**, **G83**, **G84**, **G85** или **G86**. Подготовительная функция постоянного цикла, которая предшествует функции **G702**, устанавливает способ обработки отверстий.

Параметры **X\_** и **Y\_** описанные в кадре с функциями **G81- G88** определяют центр, где размещается окружность.

**Примечание:**

*Описание постоянного цикла **G81-G86,G183** не может размещаться в том же кадре, где запрограммирован цикл **G702***



**Рис. 40. Многооперационный цикл G702**

Подготовительная функция **G702** имеет следующий формат:

**G702 I\_ J\_ L\_**

Где:

**I** – расстояние между точками сверления, должно быть больше 0;

**J** – угол между осью **X** и вектором прямой линии;

**L** – количество точек на отрезке прямой линии.

Параметр **J** является необязательным.

При отсутствии параметра **J** - угол равен нулю.

При отсутствии параметров **I,L** выводится сообщение о синтаксической ошибке.

**Пример:**

В вышеприведенном примере сначала производится сверление отверстий с помощью цикла сверления отверстий, а затем в тех же точках производится нарезание резьбы.

G83 X\_ Y\_ Z\_ R\_ L\_

G702 I1 J45 L5

G80

G84 X\_ Y\_ Z\_ R\_ L\_ F\_ P\_ Q\_

G702 I1 J45 L5

G80

## 10. ПАРАМЕТРИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

### 10.1 Назначение

Параметрическое программирование предназначено для расширения возможностей программирования УП для СЧПУ. Применение параметрического программирования повышает производительность и гибкость создания УП. Параметрическое программирование дает технологю-программисту следующие возможности:

- использовать в программах переменные;
- вычислять сложные выражения, использовать тригонометрические, алгебраические и логические функции;
- выполнять условные и безусловные переходы внутри программы;
- создавать подпрограммы с параметрами, и вызывать их;
- создавать библиотеки программ и подпрограмм для технологических и измерительных циклов.

### 10.2 Вызовы подпрограмм

#### 10.2.1 Вызов подпрограмм с параметрами

Параметрическая подпрограмма является внешней подпрограммой (не находится в кадрах текущей УП). Она расширяет возможности технологического программирования и позволяет передать числовые параметры из вызывающей программы в подпрограмму.

Синтаксис вызова параметрической подпрограммы:

**LP**<число> [**A**\_] [**B**\_] [**C**\_] ... [**Z**\_]

где:

**LP**<число> – команда вызова подпрограммы с параметрами.

**P**<число> – представляет собой имя файла с расширением **.NC**, в котором хранятся кадры подпрограммы.

<число> – может иметь значение в диапазоне от 1 до 99999.

#### Примечание:

*Задание незначащих нулей в имени файла подпрограммы в поле <число> не разрешается.*

*Например: Для подпрограммы P0123.nc вызов по команде LP0123 приведет к ошибке. Подпрограмма не будет найдена.*

Параметры передаются при помощи буквенных кодов (адресов), следующих за текстом **LP**<число>.

Для передачи параметров разрешается использовать все буквы латинского алфавита от **A** до **Z**, за исключением букв **E,G**. Строчные и прописные буквы не различаются.

#### Пример:

N10 LP352 A1.5 D3.4 K33.2

В этом кадре вызывается подпрограмма **P352** (файл с именем P352.NC), в которую передаются параметры A,D, K с указанными значениями.



При вызове подпрограммы с параметрами по команде **LP<число>**, программа **MSHAK-CNC** ищет файл-подпрограмму **P<число>.NC** в текущем каталоге (там, где находится основная программа). Если ее нет в текущем каталоге, то поиск производится в каталоге, определяемый системным параметром **SubProgLib** в файле **Profile.CNC**.

Если файл подпрограммы отсутствует в обоих каталогах, то **MSHAK-CNC** выдает ошибку.

Возврат из подпрограммы задается вспомогательной функцией **M17**.

Вызов одной подпрограммы из другой называется вложенным вызовом.

Главная программа имеет уровень вложенности **0**.

При вызове подпрограммы по команде **LP** (вызов внешней подпрограммы) создается набор локальных переменных от **#1** до **#99** для данной подпрограммы (см. Раздел 10.5.). Они иницируются значением “не определено”.

При передаче параметров в подпрограмму, значения параметров запоминаются в локальных переменных.

На рис 51 показано, как происходит вложенный вызов подпрограмм, и как создаются и удаляются локальные переменные.

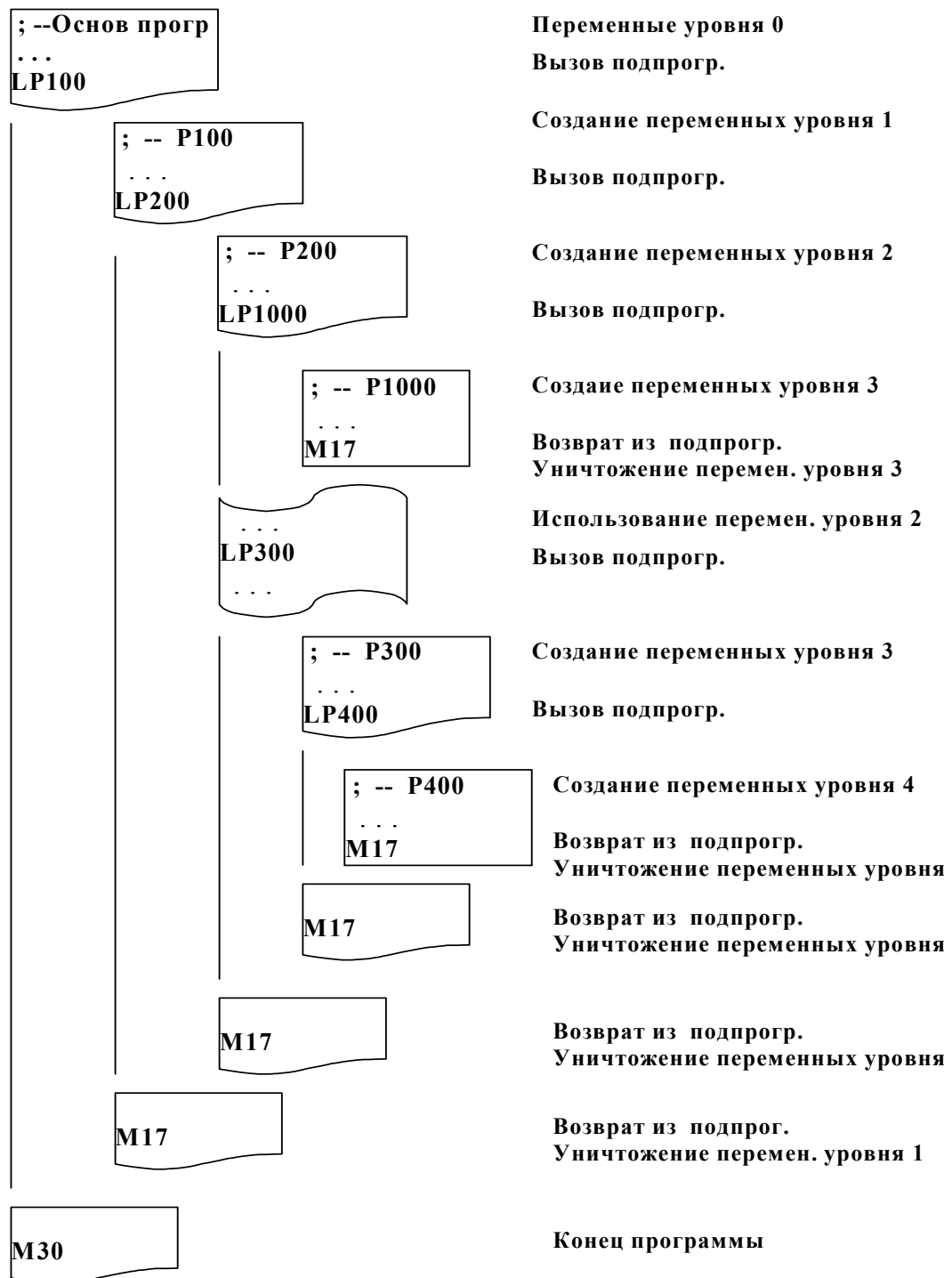


Рис. 52 Вложенный вызов подпрограмм и локальные переменные

### 10.2.2 Простой вызов подпрограмм

Простой вызов подпрограммы программируется с помощью адресного слова **L**. Подпрограмма в этом случае должна находиться в том же файле, что и основная программа.

При использовании простого вызова подпрограммы передача параметров невозможна.

Формат:

**L**<число>

<кадры>

**N**<число>

<кадры>

**M17**

Где:

**L**<число> означает вызов подпрограммы с номером кадра **N**<число>

Число уровней вложенности вызовов подпрограмм не ограничено, но рекомендуется не использовать более чем 100 уровней вложенности.

Конец подпрограммы программируется вспомогательной функцией **M17**, после которой происходит возврат из подпрограммы, и затем выполняется тот кадр, который следует за кадром, содержащим вызов подпрограммы **L**<число>.

#### Примечание:

*В кадре, где запрограммирован вызов подпрограммы, не должны быть указаны следующие величины:*

*начало повторяющего отрезка (**H**),*

*конец повторяющего отрезка (**M20**),*

*программный переход (**E**),*

*возврат из подпрограммы (**M17**) и конец программы (**M02**).*

*В кадре, где запрограммирован возврат из подпрограммы, не должны быть указаны следующие величины:*

*начало повторяющего отрезка (**H**),*

*конец повторяющего отрезка (**M20**),*

*программный переход (**E**) и конец программы (**M02**),*

*вызов подпрограммы (**L**).*

#### Примеры:

; Основная программа

X5Z5

L10 ;Вызов подпрограммы, начинающейся с кадра N10

X2Z8

L20 ;Вызов подпрограммы, начинающейся с кадра N20

X0Z0

M2

N10Z2 ;Подпрограмма с меткой кадра N10

X10

M17 ;Конец подпрограммы и возврат в основную программу

N20Z15 ;Подпрограмма с меткой кадра N20

X20

L10 ;Вложенный вызов подпрограммы N10 из подпрограммы L20

M17 ;Конец подпрограммы и возврат в основную программу

### **10.2.3 Программирование повторяющегося отрезка программы (H)**

Повторяющийся отрезок управляющей программы задается с помощью адресного слова **H** и вспомогательной функцией **M20**.

Формат:

**H<число>**

**<кадры>** ; повторяющийся отрезок программы

**M20**

Где:

**H<число>** - показывает количество повторов отрезка **УП**.

**<число>** - принимает значение в диапазоне **0...99999**.

Конец отрезка задается вспомогательной функцией **M20**.

После выполнения последнего кадра отрезка, если количество повторов не закончилось, выполнение программы продолжается, начиная с первого кадра отрезка.

Разрешается программировать вложение повторяющегося отрезка программы.

Количество вложенных повторов - до 100.

В этом случае первый кадр, который содержит **M20**, считается концом последнего повторяющегося отрезка. Следующий кадр, который содержит **M20**, считается концом предпоследнего повторяющегося кадра и так далее до первого отрезка, который содержит все повторяющиеся отрезки.

Если в УП запрограммировано начало повторяющегося отрезка (**H**), но отсутствует конец отрезка (**M20**), то повтор отрезка игнорируется.

#### **Примечание:**

*В кадре, где задан **H**, не должны быть указаны следующие величины:*

*конец повторяющегося отрезка (**M20**),*

*вызов подпрограммы (**L**),*

*программный переход (**E**),*

*возврат из подпрограммы (**M17**) и конец программы (**M02**).*

*В кадре, где задан конец повторяющегося отрезка (**M20**), не должны быть указаны следующие величины:*

*начало выполняющегося отрезка (**H**),*

*вызов подпрограммы (**L**),*

*программный переход (**E**),*

*возврат из подпрограммы (**M17**) и конец программы (**M02**).*

#### **Пример:**

Z10X5 F1000 G1

N5 H5 ;выполнить отрезок программы N5 - N25 5 раз

G91 X5

Z10

N10 H2 ;выполнить отрезок программы N10 - N15  
;2\*5 = 10 раз  
Z2X1  
X1  
N15 M20 ;конец отрезка программы для повтора,  
;начиная с кадра N10  
Z3X3  
Z1  
N25 M20 ;конец отрезка программы для повтора,  
;начиная с кадра N5

### 10.3 Переменные (#)

Применение переменных позволяет расширить возможность программирования УП.

Переменные можно использовать в выражениях присвоения, в условных выражениях для выполнения условных переходов. Переменные и выражения с переменными можно использовать в качестве значений адресных слов.

#### Примечание:

*Переменные не имеют специальной размерности, но при использовании их в качестве переменных каких-то величин, значения принимают размерность этих величин.*

#### Пример:

G1 X2.5 Y3.6 ; значения заданы явно  
#1= 2.5  
#2= 3.6  
G1 X#1 Y#2 ; значения заданы переменными

Переменные (в качестве значений адресных слов) можно применить для букв от **A** до **Z** за исключением адресных слов **D,E,G,H,M,L,N,O,P,T**  
Во входном языке используется **10000** переменных.

Синтаксис задания переменной:

**#<целое>**

Где:

**<целое>** - номер переменной целое число в диапазоне **0...9999**

Значениями переменных являются числа с плавающей запятой.

Доступ к переменным можно осуществить и при помощи косвенной адресации, когда вместо явного задания номера переменной задается выражение в круглых скобках.

Синтаксис задания косвенной адресации:

**#(<выраж>)**

Где:

**<выраж>** - алгебраическое выражение, вычисленное значение которого выступает как номер переменной.

Диапазон значений алгебраического выражения **<выраж>** - целое число в диапазоне **0...9999**

Примеры косвенной адресации:

#3 =15

#15=10

#17=2

#5 = #(#3) ; Присвоить переменной #5 значение переменной #15, так как выражение

; в скобках возвращает значение переменной #3 равное 15, то есть  
; #5 будет равно 10

#6 = #(#3+2) ; Присвоить переменной #6 значение переменной #17, так как выражение

; в скобках возвращает значение 17, то есть  
; #6 будет равно 2

В следующем примере переменная #10 принимает значение равное значению переменной #1.

#10 = #1

#10 = #(1)

#10 = #((1+2+3)/6)

#2 =1

#10 = #(#2)

#10 = #(SIN(90))

Для использования переменных в УП, необходимо заранее им присвоить значения. Значения переменным могут присваиваться как в программе обработки, так и с помощью редактора переменных программы **MSHAK-CNC**. (см. документ **MSHAK-CNC Руководство оператора**). Значения переменных могут изменяться в процессе выполнения программы.

Присваивания выполняются перед выполнением команд перемещений, в заданном в кадре порядке.

Переменным можно присвоить, как числовое значение, так и сложное алгебраическое и тригонометрическое выражение, состоящее из функций и арифметических действий над переменными и константами.

**Пример:**

#3= #3+2/(#6 \* #5)

Различаются следующие типы переменных:

- Неопределенно - #0
- Локальные - #1- #99
- Общие - #100-#999
- Системные - #2000 - #7400

В программе MSHAK-CNC значения общих и локальных переменных можно наблюдать на экране. **Переменные** (см. документ **MSHAK-CNC Руководство оператора**)

Во время выполнения УП программа MSHAK-CNC показывает текущий уровень вложенного вызова подпрограммы, при помощи выделения синим цветом строки текущего уровня вызова.

Значение переменной или выражения можно задавать вместе с адресным словом следующими двумя вариантами:

### 10.3.1 Задание значения адреса совместно с выражением

Переменную или выражение можно задавать после числового значения слова совместно со знаком арифметического действия: "+", "-", "\*", "/". При наличии этого знака, числовое значение адресного слова используется в выражении как константа при вычислении выражения. В формировании выражений, для задания приоритетности выполнения вычислений, можно использовать круглые скобки.

#### Пример:

#3=25 ; присвоение переменной #3 значения 25  
 #4=#3\*2 ; присвоение переменной #4 значения 50  
 X5+#4 ; перемещение по X в координату  $5+\#4 = 55$   
 X35-#3 ; перемещение по X в координату  $35-\#3 = 10$   
 #3=10  
 #2=5  
 X4\*#2+#3 ; перемещение по X в координату  $4*\#2+\#3 = 30$   
 X4\*(#2+#3) ; перемещение по X в координату  $4*(\#2+\#3) = 60$   
 X4\*#3\*(#2+#3); перемещение по X в координату  $4*\#3*(\#2+\#3) = 600$

### 10.3.2 Задание значения адресного слова как значения выражения

Значение переменной или выражения можно присвоить адресному слову, задав его в круглых скобках. В формировании выражений можно использовать вложенные круглые скобки, для задания приоритетности выполнения вычислений.

#### Пример:

#3=25 ;присвоение переменной #3 значения 25  
 #4=#3\*2 ;присвоение переменной #4 значения 50  
 #5=40 ;присвоение переменной #5 значения 40  
 X(#3) ;перемещение по X в координату 25  
 Y(#3+#4) ;перемещение. по Y в координату 75:  $25+50=75$   
 Z(#3\*2+#4/5) ;перемещение по Z в координату 60:  $25*2+50/5=60$   
 Y((#3+#4)/3 +#5) ;перемещение по Y в координату 65:  $75/3+40=65$

## 10.4 Переменная #0 - “неопределенно”

Неопределенные значения переменных позволяют технологу-программисту определять: был ли правилен вызов подпрограммы, - все ли параметры заданы при вызове, или правильна ли какая-то логическая цепочка вызовов.

Чтобы переменная имела значение “неопределенно”, необходимо присвоить ему значение #0 или значение другой «неопределенной» переменной.

Алгебраическое выражение преобразует значение “неопределенно” в число 0.0.

Но если выражение простое присвоение, – то возвращаемое значение тоже “неопределенно”.

**Пример:**

#12 = #0 ; #12 имеет значение “неопределенно”  
 #3 = #12 ; #3 имеет значение “неопределенно”  
 #4 = -#12 ; #4 имеет значение “неопределенно”  
 #5 = #12 + #3; #5 имеет значение 0.0  
 #6 = #12 \* 3 ; #6 имеет значение 0.0

Если переменная, используемая в адресном слове, имеет значение “неопределенно”, то это адресное слово в текущем кадре игнорируется.

**Пример:**

#5 = #0  
 #6 = 35.6  
 G1 X#5 Y#6 ; то же, что и G1 Y35.6  
 X(#5 + #0) ; то же, что и X0.0  
 Z-(#5) ; Z игнорируется

**Логическое выражение** преобразует значение “неопределенно” в число 0.0. В логических выражениях, параметры со значением “неопределенно” рассматриваются как число 0.0, за исключением отношений “=” и “<”

Условие	#1 равно #0	#1 равно 0
#1 = #0	Истина	Ложь
#1 <> 0	Истина	Ложь
#1 = 0	Ложь	Истина
#1 <= #0	Истина	Истина
#1 >= #0	Истина	Истина
#1 > 0	Ложь	Ложь
#1 < 0	Ложь	Ложь

**Примечание:**

*Важно понимать отличия между понятиями “неопределенно” и равно нулю. Если в условии сравнения переменных необходимо проверить равенство нулю, то следует использовать 0.0. Если условие сравнения используется для определения - было ли присвоено значение переменной, то следует использовать #0.*

## 10.5 Локальные переменные

Локальные переменные создаются и инициализируются при каждом вызове подпрограммы по **LP** команде. Локальные переменные определяются в диапазоне от #1 до #99. Они инициализируются значением “неопределенно”, которое эквивалентно #0.

Вызов подпрограммы по **L** команде, отличается от вызова по **LP**, так как в этом случае не создаются локальные переменные, но существующие при последнем вызове по команде **LP** локальные переменные остаются доступными подпрограмме.

При возврате из подпрограммы **LP** по **M17**, созданные при вызове на этом уровне локальные переменные удаляются. При возврате из подпрограммы **L** по **M17**, локальные переменные не удаляются.



Локальные переменные #1...#26 соответствуют адресным словам **A...Z** и используются для передачи значений параметров команды LP в подпрограмму. Соответствие адресного слова параметра и номера локальной переменной приведено в таблице ниже.

Адресное слово	Локальная переменная
A	#1
B	#2
C	#3
D	#7
F	#9
H	#11
I	#4
J	#5
K	#6
L	#12
M	#13
N	#14
O	#15
P	#16
Q	#17
R	#18
S	#19
T	#20
U	#21
V	#22
W	#23
X	#24
Y	#25
Z	#26

**Примечание:**

*Адресные слова E,G не могут использоваться для передачи параметров.*

Если в команде LP задан параметр (адресное слово), то соответствующая переменная принимает значение, равное значению адресного слова. В противном случае, данная переменная принимает значение “неопределенно”.

**Пример:**

При двух вариантах вызова подпрограммы:

LP100 A20 - вызов подпрограммы с передачей параметра  
 LP100 - вызов подпрограммы без передачи параметра

Необходимо в подпрограмме выполнить проверку на передачу параметра.

IF (#1 = #0) E1 ; переход на кадр N1 , если параметр A- неопределен  
 X #1 ; перемещение по X на величину параметра #1  
 E2 ; безусловный переход на кадр N2  
 N1 X 100 ; перемещение по X на величину 100  
 N2

Таким образом, если условное выражение  $\#1 = \#0$  определяется как ИСТИНА, это значит, что переменной  $\#1$  не было присвоено значение и, либо аргумент **A** при вложенном вызове подпрограммы отсутствует, либо имеет значение “неопределенно”.

В подпрограмме с параметрами локальные переменные  $\#27\...\#99$  можно использовать в качестве обычных переменных. Необходимо учесть, что при возврате из подпрограммы по M17 созданные на этом уровне локальные переменные удаляются.

**Пример:**

$\#31 = \#3 * 2$  ; присвоить переменной  $\#31$  удвоенное значение параметра **C**  
**G1 G91 X#24** ; перемещение в приращениях на величину параметра **X**

### 10.6 Общие переменные

Общие переменные доступны в любой части программы. Значения этих переменных сохраняются при выключении СЧПУ и восстанавливаются при включении СЧПУ.

Во входном языке **MSHAK-CNC** можно использовать общие переменные, которые определены в диапазоне  $\#100\...\#999$ .

**Примечание:**

*Значения локальных переменных основной программы (нулевой уровень) также сохраняются при выключении СЧПУ.*

Общие переменные могут использоваться для передачи значений из одной подпрограммы в другую. Присвоенное значение переменной остается неизменной, независимо от уровня вложенности подпрограммы, до выполнения ей другого присвоения.

### 10.7 Системные переменные

Системные переменные предоставляют доступ к системным параметрам СЧПУ. Это необходимо для изменений станочных системных параметров из управляющей программы.

Перечень системных переменных приведен в таблице ниже:

<b><u>Переменная #</u></b>	<b>Описание</b>
$\#2000 - \#2800$ $\#3000$	Корректоры инструментов по длине Z и радиусу D Аварийное прерывание выполнения УП с сообщением
$\#3001 - \#3002$ $\#3006$	Системные таймеры Стоп выполнения УП с сообщением
$\#6001 - \#7400$	Корректоры инструментов по X, Y, U, V, W
$\#4001 - \#4020$	Модальная информация о G функциях
$\#4101 - \#4126$	Модальная информация об адресных словах
$\#5001 - \#500n$	Значение командной координаты, с учетом корректора инструмента в рабочей КС
$\#5021 - \#502n$	Значения машинных координат

#5041-#504n	В рабочей КС - значение программных координат, без учета корректора инструмента
#5061-#506n	Значение координат контакта с измерительным щупом в рабочей КС без учета корректора инструмента
#5070	Признак события – контакт с измерительным щупом
#5071	Признак, что активен измерительный щуп в шпинделе
#5072	Признак, что активен измерительный щуп на столе
#5080	Значения корректора радиуса текущего инструмента
#5081-#508n	Значения корректоров текущего инструмента
#5090	Значения корректора износа радиуса текущего инструмента
#5091-#509n	Значения корректоров износа текущего инструмента
#5201-#520n	Смещения текущей координатной системы
#5221-#522n	Смещения координатной системы G54
#5241-#524n	Смещения координатной системы G55
#5261-#526n	Смещения координатной системы G56
#5281-#528n	Смещения координатной системы G57
#5301-#530n	Смещения координатной системы G58
#5321-#532n	Смещения координатной системы G59

### 10.7.1 Корректоры инструментов #2000 - #2800, #6000-#7400

Системные переменные, **#2000 - #2800, #6000-#7400** предназначены для хранения значений корректоров инструментов (смещение на радиус) для 200 инструментов. При программировании используются корректоры инструментов по координатам X,Y,Z, U,V,W и корректор радиуса инструмента. Эти корректоры активизируются при задании H и D адресных слов в УП.

#2000	Всегда возвращает ноль в выражении
#2001-#2200	Корректор длины инструмента по оси Z для инструментов с 1 по 200
#2201-#2400	Корректор износа длины инструмента по оси Z для инструментов с 1 по 200
#2401-#2600	Корректор радиуса инструмента с 1 по 200
#2601-#2800	Корректор износа радиуса инструмента с 1 по 200
#6001-#6200	Корректор по оси X для инструментов с 1 по 200
#6201-#6400	Корректор износа по оси X для инструментов с 1 по 200
#6401-#6600	Корректор по оси Y для инструментов с 1 по 200
#6601-#6800	Корректор износа по оси Y для инструментов с 1 по 200
#6801-#7000	Корректор по оси U для инструментов с 1 по 200
#7001-#7200	Корректор по оси V для инструментов с 1 по 200
#7201-#7400	Корректор по оси W для инструментов с 1 по 200

Например, для инструмента **4** системные переменные **#2004, #2204, #2404, #2604**

являются соответственно корректорами:

- длины инструмента по оси Z;
- износа инструмента по оси Z;
- радиуса инструмента;
- износа радиуса инструмента.

Используя эти переменные в УП, можно изменить или прочитать значения корректоров инструментов.

Присвоение этим переменным значений в УП, является аналогичным действием, что и задание значений корректорам инструментов на экране ИНСТР (TOOLS) в программе MSHAK-CNC.

**Примечание:**

*Когда выполняется операция присвоения системным переменным, СЧПУ прекращает выполнение кадров УП, пересчитывает значения координат с учетом новых значений и затем продолжает выполнение УП.*

### **10.7.2 Аварийное прерывание выполнения УП с сообщением #3000**

Аварийное прерывание выполнения УП с выдачей сообщения может быть выполнено из параметрической программы с помощью присвоения значения переменной #3000.

Номер сообщения будет соответствовать присвоенному значению переменной #3000, а выводимым текстовым сообщением будет строка в кадре присвоения, заданная после символа комментария “;”.

**Пример:**

```
IF (#24 <> #0 ) E5  
#3000 = 1024 ; Не задан параметр X  
N5 ; условие было верно, параметр X был передан
```

Выполнение программы будет прервано и в окне «Сообщения» выводится сообщение:

Не задан параметр X

### **10.7.3 Системные таймеры #3001 - #3002**

В управляющей программе можно использовать таймеры: #3001 и #3002. Дискрета этих таймеров - миллисекунда.

Таймер #3001, после загрузки ПО MSHAK-CNC, инициализируется значением ноль, и начинает счет.

Таймер #3002 работает только тогда, когда выполняется УП.

Значение таймера #3002 сохраняется при выключении СЧПУ и восстанавливается по его включению. Оба таймера можно инициализировать в УП, операцией присвоения.

**Пример:**

```
#5 = #3001  
N2 IF (#3001 < (#5 + 3500)) E2 ; Организация задержки на 3.5 сек. в УП
```

#### 10.7.4 Останов выполнения УП с сообщением #3006

Программный останов (M00) выполнения УП с выдачей сообщения может быть выполнен из параметрической программы при присвоении значения переменной #3006.

Номер сообщения будет соответствовать значению, присвоенному переменной #3006, а выводимым текстовым сообщением будет строка, заданная в кадре присвоения после символа комментария “;”.

#### Пример:

#3006 = 2001 ; Сообщение для оператора.

#### 10.7.5 Модальная информация о G функциях в группе #4001 – #4020

Системные переменные #4001 - #4026 могут использоваться в УП для определения номера активной G функции для любой группы G-функций. Переменные от #4001 до #4020 хранят значения активного кода для групп от 1 до 20, соответственно.

Принадлежность G функции номеру группы приведено в таблице ниже

Номер группы	Наименование группы	Перечень G- функций
0	Действующие в одном кадре	G4,G9, G28,G29,G30,G31, G33,G53,G92, G74,G75,G76, G77,G78,G79, G171,G172, G180, G700,G701,G702
1	Выбор разновидности движения	<b>G0</b> ,G1,G2,G3
2	Выбор плоскости	<b>G17</b> , G18,G19
3	Размерность задания перемещений	G20, <b>G21</b>
4	Масштабирование	G22, <b>G23</b> ,G24
5	Контроль вращения шпинделя	G25,G26
6	Коррекция на радиус инструмента	G40,G41,G42
7	Коррекция на положение (длину) инструмента	G43,G44,G49
8	Локальная координатная система	G52, <b>G152</b>
9	Рабочая координатная система	<b>G54</b> ,G55,G56,G57,G58,G59
10	Движение со слиянием	G61, <b>G64</b>

11	Задание X по диаметру или радиусу	<b>G62,G63</b>
12	Поворот системы координат	<b>G68,G69</b>
13	Многопроходные токарные циклы	<b>G70,G170,G71,G72,G73</b>
14	Сверлильно-расточные фрезерные циклы	<b>G80,G81,G82,G83,G84,G86,G88,G183</b>
15	Режим задания перемещений	<b>G90,G91</b>
16	Режим задания скорости подач	<b>G93,G94,G95</b>
17	Режим задания вращения шпинделя	<b>G96,G97</b>
18	Точка возврата в сверлильных циклах	<b>G98,G99</b>
19	Зеркальная обработка	<b>G150,G151</b>
20	Режим стружкодробления	<b>G160,G161,G162</b>

Жирным шрифтом заданы G-функции, устанавливаемые по умолчанию при включении СЧПУ или при начальном запуске УП.

**Пример:**

Для проверки выключена ли компенсация радиуса инструмента **G40**, необходимо проверить, что значение группы 6 равно 40

IF (#4006 = 40) E200 ; переход на кадр N200 , если компенсация радиуса инструмента выключена.

**10.7.6 Информация об адресных словах #4101 – #4126**

Системные переменные #4101 - #4126 могут использоваться в УП для определения значений адресных слов от A до Z. Они содержат значения последних выполненных в УП адресных слов.

Переменные от #4101 до #4126 соответствуют адресным словам в соответствии с таблицей приведенной ниже.

**Примечание:**

*При вызове подпрограмм с параметрами, когда адресные слова используются для передачи параметров, значения переменных #4101-#4126 не меняются.*

Адресное слово	Переменная
A	#4101
B	#4102
C	#4103
D	#4107
F	#4109
H	#4111
I	#4104
J	#4105
K	#4106
M	#4113
Q	#4117
R	#4118
S	#4119
T	#4120
U	#4121
V	#4122
W	#4123
X	#4124
Y	#4125
Z	#4126

**Пример:**

Для определения значения последнего задания по адресному слову S, необходимо проверить переменную #4119.

IF (#4119 > 1500) E200 ; переход на кадр N200, если S был больше 1500 об/мин

***10.7.7 Командная координата, с учетом корректора инструмента  
#5001 – #500n***

Системные переменные **#5001 – #500n** содержат значения командных координат в последнем выполняемом кадре (координаты конечной точки перемещения), с учетом смещений рабочей координатной системы и корректоров инструмента. Фактически эти переменные содержат конечную программную координату в кадре.

Соответствие между осью и номером переменной приведено в таблице ниже.

Ось	Переменная
X	#5001
Y	#5002
Z	#5003
U	#5004
V	#5005
W	#5006
A	#5007
B	#5008
C	#5009

**Пример:**

N10 G54 G43 H1 F1000  
 N20 X20 Y30 Z40  
 N30 #51=#5001 #52=#5002 #53=#5003  
 M30

После выполнения кадра с номером N30, локальные переменные #51, #52, #53 будут иметь значения 20, 30, 40.

**10.7.8 Текущая машинная координата #5021-#502n**

Системные переменные #5021- #502n содержат значения текущих машинных координат.

Соответствие между осью и номером переменной приведено в таблице ниже.

Ось	Переменная
X	#5021
Y	#5022
Z	#5023
U	#5024
V	#5025
W	#5026
A	#5027
B	#5028
C	#5029

**10.7.9 Текущая координата в рабочей координатной системе #5041-#504n**

Системные переменные #5041-#504n принимают значения текущих координат в рабочей координатной системе, без учета корректоров инструментов.

Значения этих переменных учитывают наличие операций зеркалирования, масштабирования и установки локальных координат по G52.

Рабочая координатная система определяется заданием одной из функций G54-G59 из 9-ой группы G-функций.

Значение этих переменных вычисляются по формуле:

$$\#504n = K_{\text{маш}} - K_{\text{смещ}}$$

Где:

$K_{\text{маш}}$  - машинная координата по оси

$K_{\text{смещ}}$  - заданное смещение по оси в текущей координатной системе (G54-G59)

Соответствие между осью и номером переменной приведено в таблице ниже.

Ось	Переменная
X	#5041
Y	#5042
Z	#5043
U	#5044



V	#5045
W	#5046
A	#5047
B	#5048
C	#5049

#### ***10.7.10 Координата контакта с измерительным щупом #5061-#506n***

Системные переменные #5061-#506n принимают значения координат, в которых сработал измерительный щуп, при движении по функции **G31**. Координаты передаются с учетом смещений рабочей координатной системы, но без учета корректоров инструментов.

Соответствие между осью и номером переменной приведено в таблице ниже

Ось	Переменная
X	#5061
Y	#5062
Z	#5063
U	#5064
V	#5065
W	#5066
A	#5067
B	#5068
C	#5069

#### ***10.7.11 Признак события - контакт с измерительным щупом #5070***

Системная переменная **#5070** принимает значение 1, если во время движения по функции **G31** произошло событие срабатывания измерительного щупа.

#### ***10.7.12 Признак активности измерительного щупа в шпинделе #5071***

Системная переменная **#5071** принимает значение 1, если в УП была задана вспомогательная функция **M51** (активизировать измерительный щуп в шпинделе) и измерительный щуп в рабочем состоянии. Эта переменная принимает значение 0, если была задана функция **M50** (деактивизировать измерительный щуп), или при потере работоспособности щупа.

#### ***10.7.13 Признак активности измерительного щупа на столе #5072***

Системная переменная **#5072** принимает значение 1, если в УП была задана вспомогательная функция **M52** (активизировать измерительный щуп на столе) и измерительный щуп в рабочем состоянии. Эта переменная принимает значение 0, если была задана функция **M50** (деактивизировать измерительный щуп) или при потере работоспособности щупа.

#### ***10.7.14 Значения корректора радиуса текущего инструмента #5080***

Системная переменная **#5080** принимает значение корректора радиуса текущего инструмента.

**10.7.15 Значения корректоров текущего инструмента #5081-#508n**

Системные переменные **#5081- #508n** принимают значения величин корректоров текущего инструмента.

Соответствие между осью и номером переменной приведено в таблице ниже.

Ось	Переменная
X	#5081
Y	#5082
Z	#5083
U	#5084
V	#5085
W	#5086

**10.7.16 Значения корректора износа радиуса текущего инструмента #5090**

Системная переменная **#5090** принимает значение корректора износа радиуса текущего инструмента.

**10.7.17 Значения корректоров износа текущего инструмента #5091-#509n**

Системные переменные **#5091- #509n** принимают значения величин корректоров текущего инструмента.

Соответствие между осью и номером переменной приведено в таблице ниже.

Ось	Переменная
X	#5091
Y	#5092
Z	#5093

**10.7.18 Смещения текущей координатной системы #5201-#520n**

Системные переменные **#5201-#520n** принимают значения смещений текущей рабочей координатной системы (G54-G59).

В УП можно изменить значения рабочих координатных систем с помощью присвоения переменным **#5201-#520n** новых значений.

**Примечание**

*Когда выполняется операция присвоения этим переменным, СЧПУ прекращает выполнение УП, пересчитывает значения передаваемых координат с учетом новых значений и затем продолжает выполнение кадров уже с новыми значениями координат.*

Соответствие между осью и номером переменной приведено в таблице ниже.

Ось	Переменная
X	#5201
Y	#5202

Z	#5203
U	#5204
V	#5205
W	#5206
A	#5207
B	#5208
C	#5209

**10.7.19 Смещения координатной системы G54 #5221-#522n**

Имеют те же функции, что и в разделе 4.7.18, но для координатной системы G54.

**10.7.20 Смещения координатной системы G55 #5241-#524n**

Имеют те же функции, что и в разделе 4.7.18, но для координатной системы G55.

**10.7.21 Смещения координатной системы G56 #5261-#526n**

Имеют те же функции, что и в разделе 4.7.18, но для координатной системы G56.

**10.7.22 Смещения координатной системы G57 #5281-#528n**

Имеют те же функции, что и в разделе 4.7.18, но для координатной системы G57.

**10.7.23 Смещения координатной системы G58 #5301-#530n**

Имеют те же функции, что и в разделе 4.7.18, но для координатной системы G58.

**10.7.24 Смещения координатной системы G59 #5321-#532n**

Имеют те же функции, что и в разделе 4.7.18, но для координатной системы G59.

**10.8 Форматированный вывод**

Форматированный вывод предоставляет возможность записи информации в текстовый файл из УП.

При помощи форматированного вывода можно из УП ( во время ее выполнения) создавать отчетный файл, в который можно записать информацию о ходе выполнения УП (координаты точек измерений и разную вспомогательную информацию).

Для форматированного вывода используются команды PRINT(), POPEN(),PCLEAR

Команда POPEN открывает или создает файл для записи.

Команда PRINT выполняет запись информации в файл, открытый командой POPEN.

Каждая команда PRINT записывает (добавляет) информацию в конец открытого файла, не удаляя его содержимого

Команды PRINT должны быть заданы в УП после задания команды POPEN.

Команда PCLEAR выполняет удаление содержимого файла (но не самого файла), открытого командой POPEN.

PCLEAR должна быть задана в УП после задания команды POPEN.

### **Команда POPEN:**

Синтаксис команды:

```
POPEN (<имя файла>)  
POPEN ()
```

Где:

<имя файла> - маршрут и имя файла, в который будет выводиться информация.

Если файл с таким именем не существует, то в заданном каталоге создается файл с этим именем.

Если параметр <имя файла> не задан, то и в каталоге открывается файл с именем, определяемый системным параметром **ProbeFileName** системного файла **Profile.Cnc**.

POPEN должен быть задан в УП до задания команды PRINT

Если POPEN задан несколько раз с разными именами файлов, то активным остается последний заданный файл.

### **Пример:**

```
; Подпрограмма печати  
POPEN ()  
.  
.  
PRINT ()  
PRINT (Какой-то текст и переменные)  
PRINT (Какой-то текст и переменные)  
.  
.  
M17
```

### **Команда PCLEAR**

Синтаксис команды:

```
PCLEAR
```

Команда PCLEAR задается без круглых скобок, в отличие от команд POPEN и PRINT.

Его применяют, если необходимо удалить содержимое, открытого командой POPEN, файла.

### **Пример:**

```
POPEN()  
PCLEAR
```

## **Команда PRINT**

Синтаксис команды:

PRINT ( <текст>| <форматированная переменная> ...| @DATE | @TIME)

Где:

- <текст> - текст, состоящий из любых символов кроме символа @;
- Знак “|” в синтаксисе означает, что можно задать, или <текст>, или <форматированная переменная> ;
- Знак “...” означает, что параметры <текст> и <форматированная переменная> могут повторяться много раз;
- Параметр @DATE выводит в файл текущую дату в одном из двух числовых форматов:  
либо DD.MM.YY – дата, месяц, год – европейский стиль,  
либо MM.DD.YY – месяц, дата, год – американский стиль;  
Тип формата вывода даты определяется системными установками Windows;
- Параметр @TIME выводит в файл текущее время в формате HH:MM:SS, где HH:MM:SS – время в часах, минутах, секундах.

Форматированная переменная имеет следующий формат:

#<число> ( <ширина>[<точка><точность>])

Где:

- <число> - произвольный номер разрешенных локальных или глобальных переменных;
- <ширина> - общее количество символов для вывода значения переменной, включая знак минуса числа, десятичную точку и знаки после десятичной точки;  
Если число подлежащих выводу символов меньше, чем указано, то слева или справа добавляются пробелы для достижения указанного значения. Если перед числом стоит нуль, то вместо пробелов добавляются нули. Если заданная ширина недостаточна, чтобы вместить все символы для вывода целого числа, то, для нормального отображения числа, ширина числа увеличивается;
- <точка> - символ десятичной точки “.”
- <точность> - количество символов после десятичной точки. Если параметр <точность> пропущен или задан 0 , то десятичная точка и дробная часть числа не выводятся.

Квадратные скобки показывают, что лексемы <точка> и <точность>, являются необязательными для задания.

### **Примечание:**

*Если в команде PRINT не задан ни один параметр, то в файл записывается символ перевода строки.*

*Если выводимая переменная имеет значение “неопределенно”, то выводится значение “0”.*

**Примеры:**

Задано:

#15 = 6.79  
#25 = 7.9  
#256 = -5.78

После команды:

PRINT(A=#15(4.2), X=#25(08.3) и переменная #256=#256(6.2) -- дата: @DATE)

В файл будет выведено:

A=6.79, X= 0007.900 и переменная #256= -5.78 -- дата: 21.09.01

Задано:

#9 = 50  
#24 = -0.125  
#25 = 1

После команды:

PRINT (G1 X#24(08.4) Y-#25(08.4) F#9(3))

В файл будет выведено:

G1 X-00.1250 Y-01.0000 F 50

Задано:

#500 = 100  
#501 = #0

После команды:

PRINT (Это часть #501 из #500(3))

В файл будет выведено:

Это часть 0 из 100.

## 10.9 Выражения

В параметрических программах, вычисление выражений используется в операциях присвоения и операциях переходов.

В этой главе объясняется, что такое выражение, как оно формируется и как используется в программе.

Выражение состоит из трех элементов: **операнды, операторы и круглые скобки.**

**Операнд** – это **переменная** или **литеральное число.**

**Переменная** – локальная, общая или системная переменная.

**Литеральные числа** - константы.

**Оператор** – это функция, которая использует операнды, и возвращает результирующее значение.

Операторами являются простые символы, такие как “+” и “/” или функции типа SIN() или LOG().

Операторами могут быть условные операторы, такие как “=” или “>”.

**Примечание:**

*Выражения вычисляются слева направо. При вычислении выражения, операторы выполняются, либо сразу, либо в порядке уменьшения приоритета операций.*

**Пример:**

В следующем выражении,

#5 = 4+5\*9

переменной #5 будет присвоено значение 49, потому что операция умножения имеет больший приоритет, чем сложение, и сначала выполняется эта операция.

Круглые скобки, символы “(“ и “)”, могут менять порядок выполнения операций, так как сначала выполняются операторы, заданные в скобках, затем остальные операторы.

Следующая таблица определяет приоритетность выполнения операторов в выражении. Чем больше значение “**Приоритет**”, тем больше приоритет операции.

Символ	Значение	Приоритет
=	Равно (оператор условия)	1
<>	Не равно (оператор условия)	1
>	Больше (оператор условия)	1
>=	Больше или равно (оператор условия)	1
<	Меньше (оператор условия)	1
<=	Меньше или равно (оператор условия)	1
+	Сложение	2
-	Вычитание	2
	Логическое ИЛИ	2
&	Логическое И	2
*	Умножение	3
/	Деление	3
%	Остаток от деления	3
+	Унарный плюс	6
-	Унарный минус	6
POPEN	Открытие файла вывода	7
PRINT	Вывод в файл	7
#(	Косвенная адресация	7
ABS	Абсолютное значение	7
ACOS	Арккосинус	7
ASIN	Арсинус	7
ATAN	Арктангенс	7
COS	Косинус	7
EXP	Экспонента	7
FIX	Выделение целой части числа	7

FUP	Округление до целого в большую сторону	7
LN	Натуральный логарифм	7
ROUND	Округление числа	7
SIN	Синус	7
SQRT	Квадратный корень	7
TAN	Тангенс	7

В тригонометрических функциях SIN, COS, TAN аргументы задаются в градусах. Для функций ATAN, ASIN, ACOS результаты возвращаются в градусах

**Примеры выражений:**

#4	Простое выражение;
3.14159	Литеральная константа;
#4/2.5	Сложное выражение;
(#1+#5)/3	Сложное выражение с измененным порядком выполнения операций;
(#6<>#8)	Логическое условие;
SIN(#5) / COS(#7)	Применение функций в выражении.

При обсуждении синтаксиса параметрического программирования, понятие “выражение”, подобно приведенным выше, в тексте будет задаваться в виде - <выраж.>.

В параметрическом программировании выражение используется в следующих четырех синтаксических формах:

- Предложение присвоения;
- Адресное слово;
- Условное выражение;
- Переход.

**10.9.1 Предложение присвоения**

Предложение присвоения используется для изменения значения переменной, присвоения ей нового значения. Оно может иметь следующие формы:

- #<целое> = <выраж> ; прямое присвоение;
- #(<выраж>) = <выраж> ; косвенное присвоение.

Прямое присвоение меняет значение переменной, которое явно задано.

**Пример:**

```
#5 =12.1
#3000=200
```

Косвенное присвоение меняет значение переменной, которое вычисляется в результате вычисления выражения.

**Пример:**

```
#5 =12
#(#5) = 10 ; Присвоить переменной #12 значение равное 10
#(#5+1)= #0 ; Присвоить переменной #13 значение “неопределенно”
```



### 10.9.2 Адресное слово

Адресное слово с применением выражения, можно задавать в следующих формах:

- x<литерал>
- x(<выраж>)
- x-(<выраж>)

Где:

<литерал> - это литеральные числа - константы, например 1.0, 5, 3.567;  
 x - это адресное слово, задаваемое буквой латинского алфавита от A до Z.

#### Примечание

*Если после x задан <литерал>, x может быть любой буквой алфавита A-Z.*

Переменные (в качестве значений адресных слов)

*Если после x задано <выраж>, можно применить буквы от A до Z, за исключением адресных слов D,E,G,H,M,L,N,O,P,T*

#### Пример:

#1=#0

#2=3.2

G0 X#1 Y#2 ; то же, что и G0 Y3.2

### 10.9.3 Условное выражение

Условное выражение задается при помощи следующих операторов условия:

=	Равно
<>	Не равно
>	Больше
>=	Больше или равно
<	Меньше
<=	Меньше или равно

Условное выражение имеет следующий формат:

(<выраж>)

или

(<выраж> <условие> <выраж>)

Где:

<условие> - это один из операторов условия приведенных выше.

Эти операторы работают с двумя операндами, и в результате возвращают либо значение 1.0, либо 0.0.

Если выражение условия ИСТИННО, то возвращается 1.0, если выражение условия ЛОЖНО, возвращается значение 0.0.

### 10.9.4 Переход

Параметрическое программирование позволяет управлять последовательностью выполнения кадров программы. Для управления этой последовательностью используются следующие программные конструкции:

Безусловный переход	<b>E</b>
Условный переход	<b>IF</b>
Останов или прерывание	#3006= n ; останов выполнение УП #3000= n ; аварийное прерывание выполнения УП

При составлении УП, кадры можно отмечать с помощью адресного слова **N**. Отмеченные кадры можно использовать при выполнении переходов на них с помощью **E** или **IF** конструкций. Номера отмеченных кадров не должны повторяться.

Переходы могут быть двух типов – **условные и безусловные**.

#### 10.9.4.1 Безусловный переход

Безусловный переход задается при помощи адресного слова **E**. При выполнении УП в кадре, где запрограммирована **Exxx**, выполняется переход на кадр с номером **Nxxx**.

Формат:

**Exxx**

Где:

**xxx** – любое положительное число указывающее номер кадра, который должен выполняться после текущего кадра

#### Пример:

N1 G54G90

...

E5 ; выполняется переход вперед на кадр N5  
X0Y0

...

N5 G1 X40 F100

...

E5 ; выполняется переход назад на кадр N5.

#### Примечание:

Если **E** задан не отдельным кадром, а задан после какого-то адресного слова, то следует после числа обязательно задать пробел, иначе СЧПУ воспримет **E** как показатель числа **10** для этого слова и использует его для умножения.

При загрузке УП в буфер отработки, если в УП не найден какой-либо кадр **N** для перехода **E**, СЧПУ выдаст аварийное сообщение.

Для удобства программирования УП, можно в файле макроопределений задать определение (см. Раздел 10.10 Макроопределения), и использовать его в тексте УП.

### **#define GOTO E**

Например, учитывая вышеприведенное макроопределение, можно написать:

```
GOTO5      ; выполняется переход вперед на кадр N5
X0Y0
...
N5 G1 X40 F100
...
GOTO5      ; выполняется переход назад на кадр N5.
```

#### **10.9.4.2 Условный переход**

Кадр с условием позволяет выполнять в УП, или условный переход, или кадр УП. Кадр с условием задается при помощи слова IF, и имеет следующие форматы:

```
IF (<выраж><условие><выраж>) <кадр УП>
Exxx (<выраж><условие><выраж>)
```

Где:

- <выраж><условие><выраж> - условное выражение (см. Раздел 8.9.3);
- Exxx - указывает на кадр xxx, на который должен быть выполнен переход;
- <кадр УП> - кадр УП, где могут быть любые команды в кадре - например движения, присвоения, безусловного перехода - Exxx.

Если значение условного выражения ЛОЖНО, то выполняется следующий кадр УП.

Если значение условного выражения ИСТИННО то:

1 для формата задания со словом IF - выполняется предложение, заданное в этом кадре, <кадр УП>, и затем следующий кадр. Если <кадр УП> является командой безусловного перехода, то выполняется команда перехода, и следующий кадр не выполняется;

2 для формата задания с Exxx - выполняется переход на кадр Nxxx.

#### **Пример- переход по условию:**

```
IF (#24 = #0 ) E99 ; переход на кадр N99, так как не задан аргумент для X
E12 (#5>3) ; переход на кадр N12, если переменная #5 больше трех
```

#### **Пример - создание цикла в УП:**

```
N10 #5 = 0
N20 #5 = #5+1
...
...
N100 IF (#5 < 10) E20 ; переход на кадр N20, пока переменная #5 меньше 10
```

**Пример – присвоение:**

```
IF (#500 <> 0) #500=0
```

Для удобства составления УП, можно задать определение в файле макроопределений:

```
#define GOTO E
```

и в тексте УП использовать **GOTO** вместо **E**.

**Пример- переход по условию с использованием GOTO:**

```
IF (#24 = #0 ) GOTO 99 ; переход на кадр N99, так как не задан аргумент для X.
```

**Пример - создание цикла в УП с использованием GOTO:**

```
N10 #5 = 0
```

```
N20 #5 = #5+1
```

```
...
```

```
...
```

```
N100 IF (#5 < 10) GOTO 20 ; переход на кадр N20, пока переменная #5 меньше 10.
```

### 10.10 Макроопределения

При составлении УП, в тексте программ можно использовать макроопределения, которые заданы в специальном файле определений.

Макроопределения позволяют заменить одну последовательность символов на другую, которая задана в этом файле.

Применение макроопределений создает более удобный для чтения и понимания текст УП.

Формат задания макроопределения:

```
#define <слово1> <слово2>
```

Где:

<слово1> - набор латинских букв, цифр и символов без пробелов и не больше 32;

<слово2> - любое допустимое адресное слово или переменная.

Программа **MSHAK-CNC** при загрузке УП, обнаружит символы <слово1>, выполнит поиск в файле макроопределений соответствующего определения, и заменит его на <слово2>.

**Пример- предположим, что есть программа**

```
#1=5
```

```
#2=10
```

```
G1 F1000 X#1 Y#2
```

```
IF (#3 = #0) E 45
```

Если мы запишем в файл макроопределений следующие строки,

```
#define A_VAR #1  
#define B_VAR #2  
#define C_VAR #3  
#define GOTO E
```

то ту же программу можно написать следующим образом.

```
A_VAR=5  
B_VAR=10  
G1 F1000 X A_VAR Y B_VAR  
IF (C_VAR = #0) GOTO 45
```

В вышеприведенном примере, программа MSHAK-CNC, обнаружив слово A\_VAR в тексте УП, выполнит его поиск в файле макроопределений, и заменит это слово на текст #1.

Точно также, для слова GOTO, обнаружив его, СЧПУ заменит его на слово E.

### 10.11 Организация библиотеки внешних подпрограмм (циклов).

Механизм библиотеки внешних подпрограмм использует подпрограммы с параметрами для организации режима интерактивного программирования СЧПУ, что позволяет:

- выполнять вызов внешних подпрограмм, вместо команды LP, с помощью подготовительной функции G;
- создавать библиотеку подпрограмм: выполнять группировку подпрограмм в виде иерархического списка для быстрого поиска и нахождения нужной подпрограммы;
- выполнять визуализацию подпрограммы с помощью графического рисунка на экране, и выполнять интерактивный ввод параметров подпрограмм, с одновременным контролем правильности ввода данных.

Все внешние подпрограммы должны находится в каталоге, определяемом параметром **SubProgLib** файла **Profile.cnc**.

Для обеспечения функциональности интерактивного программирования циклов в текстовом редакторе программы MSHAK-CNC, организованы два режима работы:

- просмотр списка реализованных циклов в библиотеке и выбора нужного цикла;
- ввод значений параметров для выбранного цикла.

#### 10.11.1 Вызов внешних подпрограмм с помощью подготовительной функции G.

Для вызова внешней подпрограммы с помощью функции GXXX необходимо в файле **Profile.CNC** в секции **GCode** добавить запись GXXX=2,XXX, которая означает, что реализация функции GXXX происходит с помощью внешней подпрограммы.

Номеру G функции должен соответствовать файл с именем PXXX.NC, в котором в виде УП реализована данная функция.

### 10.11.2 Создание библиотеки подпрограмм

Библиотеки подпрограмм создаются путем объединения подпрограмм в функциональные группы, с целью увеличения удобств поиска и доступа к ним. Эти группы в виде двухуровневого списка появляются на экране программы **MSHAK-CNC**, в режиме интерактивного программирования. Внешние подпрограммы поддерживаются с помощью двух системных файлов **Cycles.dat** и **CYC\_RUS.Dat (CYC\_ENG.Dat)**, которые находятся в каталоге, определяемом параметром **SubProgLib**.

#### Структура файла Cycles.dat:

[COMMON]

**SectionNumber** = <количество функциональных групп>

[sec1] ; Первая функциональная группа

**CyclesNumber** = <Количество циклов в первой функциональной группе>

[sec1\_cycle1] ; Описание первого цикла

**Cycle** = <Идентификатор цикла>

**ParNumber** = <Количество параметров в первом цикле>

**par1** = <Описание 1-го параметра >

**par2** = <Описание 2-го параметра >

.....

**parN** = <Описание N-го параметра >

.....

.....

[sec1\_cycleN] ; Описание N-ого цикла

**Cycle** = <Идентификатор цикла>

**ParNumber** = <Количество параметров в N-ом цикле>

**par1** = <Описание 1-го параметра >

**par2** = <Описание 2-го параметра >

....

**parM** = <Описание M-го параметра >

Где:

**SectionNumber** - Количество функциональных групп

**Sec1, Sec2, ...SecN** - Функциональные группы

**CyclesNumber** - Количество циклов в функциональной группе

**Cycle** - Идентификатор цикла - имеет следующий формат:

**G**< номер подпрограммы >

Где:

< номер подпрограммы> - Это трехзначное целое число, которое является номером **G** функции при вызове внешней подпрограммы из **УП**, а также частью имени файла внешней подпрограммы **P**< номер подпрограммы>.**NC**, который хранится в каталоге, определяемый параметром **SubProgLib** файла **Profile.cnc**.

**ParNumber** - Количество параметров в цикле

**Par1** - Описание параметра имеет следующий формат:

<Символ (Адресное слово)>, [P1], [P2], [P3], [P4], [P5], [P6]

Где:

<Символ (Адресное слово)> - идентификатор (название) параметра

[P1] – значение параметра по умолчанию.

[P2] – минимальное значение параметра.

[P3] – максимальное значение параметра.

[P4] – если значение =1, то ввод данного параметра обязателен,  
если значение =0, то ввод данного параметра необязателен.

[P5] – значение позиции названия параметра на графическом рисунке по горизонтали.

[P6] - значение позиции названия параметра на графическом рисунке по вертикали.

**Примечание:**

*Файлы P<номер подпрограммы>.NC должны находиться в директории, определяемом параметром SubProgLib.*

**Структура файла Cycles.dat:**

[sec1]

**SectionName** = < Назначение первой функциональной группы >

[sec1\_cycle1]

**CycleName** = < Назначение первого цикла первой функциональной группы >

**Адресное слово первого параметра** = Назначение первого параметра

.....

**Адресное слово N - го параметра** = Назначение N-ого параметра

[sec1\_cycleM]

**CycleName** - Назначение M-го цикла первой функциональной группы

**Адресное слово первого параметра** = Назначение первого параметра

.....

**Адресное слово N - го параметра** = Назначение N-ого параметра

.....

.....

[secN]

**SectionName** - Назначение N-ой функциональной группы

[secN\_cycle1]

**CycleName** - Назначение первого цикла N-ой функциональной группы

**Адресное слово первого параметра** = Назначение первого параметра

.....

**Адресное слово N - го параметра** = Назначение N-ого параметра

[secN\_cycleM]

**CycleName** - Назначение M-го цикла N-ой функциональной группы

Адресное слово первого параметра = Назначение первого параметра

.....  
Адресное слово N - го параметра = Назначение N-ого параметра

Фактически, файл **CYC\_RUS.Dat (CYC\_ENG.Dat)** представляет собой назначение (имена) функциональных групп, назначение отдельных циклов и назначение параметров этих циклов.

### ***10.11.3 Визуализация цикла и интерактивный ввод параметров подпрограмм***

В поддержку режима интерактивного программирования при разработке внешней подпрограммы создается файл графического рисунка подпрограммы. Этот файл имеет название

**P<номер подпрограммы>.jpg**

Где: < номер подпрограммы> - Это трехзначное целое число, которое является номером **G** функции при вызове внешней подпрограммы из **УП** и определяется в файле **Cycles.dat** в разделе описание цикла в переменном **Cycle = <Идентификатор цикла>**.

Файл **P<номер подпрограммы>.jpg** должен находиться в директории, определяемый параметром **SubProgLib** файла **Profile.cnc**.

На графическом рисунке, для каждого параметра цикла создаются метки. Метка представляет собой символ соответствующей названию параметра.

Местоположение метки на рисунке (координаты по горизонтали и по вертикали) задается в описании параметра в файле **Cycles.dat**.

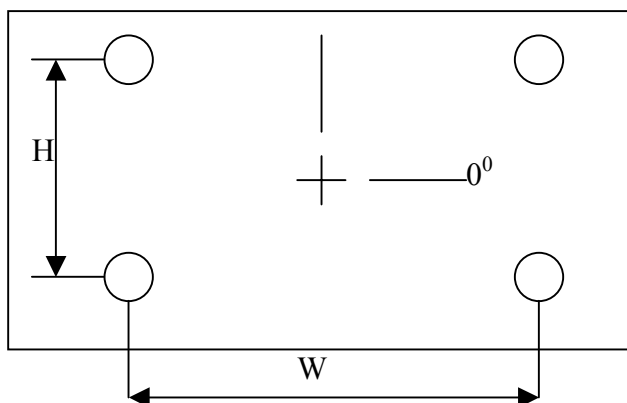
Для изменения положения метки, прямо на рисунке подпрограммы необходимо: курсор мыши переместить на нужную метку, нажимая клавишу **“Shift”**, нажать левую кнопку мыши, и оставляя ее нажатой (**“Shift”** можно отпустить), переместить метку на экране в нужное положение.



## 11. ПРИЛОЖЕНИЕ 1 - ПРИМЕР ПРИМЕНЕНИЯ ПАРАМЕТРИЧЕСКОЙ ПОДПРОГРАММЫ

*Создание пользовательской подпрограммы для сверления отверстий.*

*Требуется разработать параметрическую подпрограмму сверления 4 отверстий в детали, расположенных в углах прямоугольника. Это подпрограмма может быть сохранена на диске и использована в дальнейшем для сверления подобных деталей.*



Входные параметры:

X,Y - Абсолютная позиция центра прямоугольника;

H,W - Значения высоты и ширины между отверстиями;

A - Угол поворота, относительно которого сверлятся отверстия;

R - Координата, с которой начинается выполнение сверления (начальный уровень);

Z - Глубина сверления, заданная в приращении относительно R

### Примечание:

*Четыре отверстия сверлятся относительно центра прямоугольника.*

*Предположим, что сверлится резьбовое отверстие M20 и для цикла необходимо, чтобы сверло было первым инструментом, а метчик вторым.*

Можно в дальнейшем в подпрограмму добавить возможность сверления центрального отверстия или добавить передачу значения подачи при обработке разных материалов.

; Пользовательский цикл сверления и резбонарезания по углам

; прямоугольника

; Файл с именем P9600.nc

; X= абсолютная координата по X центра прямоугольника

; Y= абсолютная координата по Y центра прямоугольника

; H= расстояние по Y между отверстиями при 0 градусе поворота

; W= расстояние по X между отверстиями при 0 градусе поворота

; A= угол поворота прямоугольника в градусах

; R = опорный уровень сверления (R-уровень), Z начальный уровень

; Z = глубина сверления в приращениях

; Сверлятся 4 резьбовых отверстия M20.

; Если X и Y не переданы, текущая координата является центром

; Если A не передан, нет вращения

; Если R не передан, R имеет значение текущей координаты Z

IF (#11 = #0) GOTO9610 ;не передан H  
 IF (#23 = #0) GOTO9620 ;не передан W  
 IF (#26 = #0) GOTO9630 ; не передан Z

;Инициализация R, X, и Y, если они не переданы  
 IF (#24 = #0) #24=#5041 ;текущая позиция по X  
 IF (#25 = #0) #25=#5042 ;текущая позиция по Y  
 IF (#18 = #0) #18=#5043 ;текущая позиция по Z

; вычислить величины смещений от центра по X и Y, используя H и W  
 #31=#11/2  
 #32=#23/2

;переместиться в точку смены инструмента и сменить инструмент - сверло T1  
 T1 M06

S1000 M3 ;включить шпиндель  
 M8 ; включить СОЖ  
 G0 X#24 Y#25 ; переместиться в координаты XY  
 Z#18 ;подойти на R-уровень

; Выполнить вращение, если необходимо

IF (#1 = #0) GOTO9602

G68 R#1

N9602

; Выполнить сверление четырех отверстий относительно центра

G81 X(#24+#31) Y(#25+#32) Z#26 F100 ; Правый верхний  
 X(#24 - #31) Y(#25 + #32) ; Левый верхний  
 X(#24 - #31) Y(#25 - #32) ; Левый нижний  
 X(#24 + #31) Y(#25 - #32) ; Правый нижний

G80

G69 ;отмена поворота

M5 ;выключить шпиндель

M9 ;выключить СОЖ

;переместиться в точку смены инструмента и сменить инструмент – метчик ;M20 T2

T2 M06

S100 M3 ;включить шпиндель

M8 ;включить СОЖ

G0 X#24 Y#25 ; переместиться в координаты XY

Z#18 ;подойти на R-уровень

; Выполнить вращение, если необходимо

IF (#1 = #0) GOTO9604

G68 R#1

N9604

; Выполнить сверление четырех отверстий относительно центра

G84 X(#24+#31) Y(#25+#32) Z(#26-.2) F2 ; Правый верхний  
 X(#24 - #31) Y(#25 + #32) ; Левый верхний  
 X(#24 - #31) Y(#25 - #32) ; Левый нижний  
 X(#24 + #31) Y(#25 - #32) ; Правый нижний

G80

G69 ;отмена поворота

M5 ;выключить вращение

M9 ;выключить СОЖ

GOTO 9699 ; Нормальное завершение

; Аварийный останов

N9610 #3000=961 ; Нет параметра H

N9620 #3000=962 ; Нет параметра W

N9630 #3000=963 ; Нет параметра Z

N9699 M17

; Основная программа, из которой вызывается

; пользовательский цикл сверления

LP9600 X100 Y100 H125 W175 Z50 ; Первая заготовка

LP9600 X200 Y300 H125 W175 Z50 A-135 ; Вторая заготовка

M30