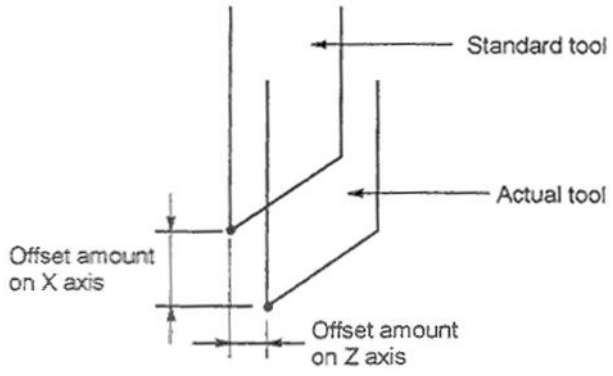
2.14 Функция компенсации

Эта глава содержит следующие функции компенсации:

2.14.1 Офсет инструмента
2.14.2 Обзор компенсации радиуса кромки инструмента
2.14.3 Подробное описание компенсации радиуса кромки инструмента
2.14.4 Значения компенсации инструмента, число значений компенсации и ввод значений из программы (G10)
2.14.5 Автоматический офсет инструмента (G36, G37)

2.14.1 Офсет инструмента

Офсет используется для компенсации разницы инструментов если действительная длина отличается от запрограммированной (обычный, стандартный инструмент).



Offset amount on X axis – Значение офсета по оси X
Standard tool – стандартный инструмент
Actual tool – действительный инструмент
Рис. 14.1 Офсет инструмента

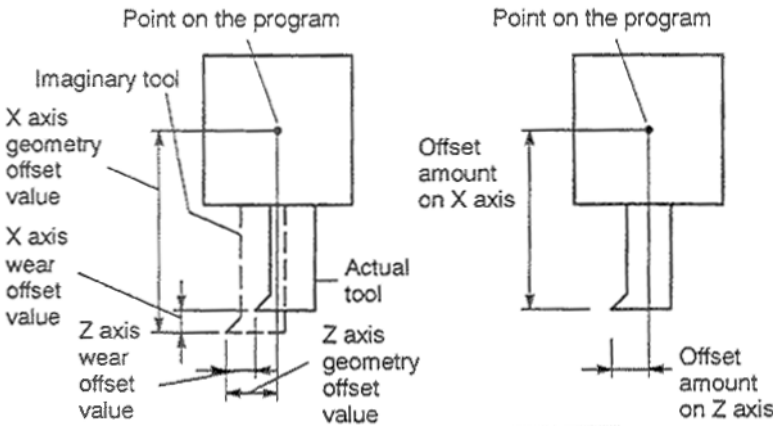
Для указания офсета инструмента не нужен G код, офсет указывается в T коде.

12.4.1.1 Офсет геометрии и офсет износа инструмента

Офсет геометрии инструмента и офсет износа можно разделить на две составляющие: значения компенсации геометрии и значение компенсации износа кромки инструмента. Общее значение офсета геометрии инструмента и офсета износа устанавливается в значения офсета износа инструмента.

ПРИМЕЧАНИЕ

Офсет геометрии и офсет износа инструмента опциональны.



Point on the program – Точка для программы
Imaginary tool – Воображаемый инструмент
Actual tool – действительный инструмент
X axis geometry offset value – Значение офсета геометрии по оси X
X axis wear offset value – Значение офсета износа по оси X
Рис. 14.1.1 Инструмент имеющий и не имеющий значения офсета

2.14.1.2 Т код офсета инструмента

Существует два способа указания Т кода, они приведены в таблице 14.1.2(a) и 14.1.2(b).

Формат

Последняя цифра Т кода обозначающая офсет геометрии и износа

Таблица 14.1.2(a)

Тип Т кода	Значение Т кода	Параметр номера офсета	
Команда с 2 цифрами	Тху х – Номер инструмента у – Номер офсета	Если бит 0 (LD1) параметра 5002 установлен в 1, офсет износа определяется последней цифрой Т кода.	Если бит 1 LGN параметра 5002 установлен в 0, номер офсета геометрии и износа инструмента для обозначения офсета используются две цифры.
Команда с 4 цифрами	Тххуу хх – Номер инструмента уу – Номер офсета	Если бит 0 (LD1) параметра 5002 установлен в 0, офсет износа определяется последними двумя цифрами Т кода.	

Последняя цифра Т кода обозначающая офсет износа и первые цифры выбора инструмента с кодом офсета геометрии

Таблица 14.1.2(b)

Тип Т кода	Значение Т кода	Параметр номера офсета	
Команда с 2 цифрами	Тху х – Номер инструмента и номер офсета геометрии у – Номер офсета износа	Если бит 0 (LD1) параметра 5002 установлен в 1, офсет износа определяется последней цифрой Т кода.	Если бит 1 LGN параметра 5002 установлен в 0, номер офсета геометрии и износа инструмента для обозначения офсета используются две цифры.
Команда с 4 цифрами	Тххуу хх – Номер инструмента и номер офсета геометрии уу – Номер офсета износа	Если бит 0 (LD1) параметра 5002 установлен в 0, офсет износа определяется последними двумя цифрами Т кода.	

2.14.1.3 Выбор инструмента

Выбор инструмента производится указанием Т кода с соответствующим номером инструмента. См. документацию производителя станка для определения отношения между номером инструмента и его позицией.

2.14.1.4 Номер офсета

Номер офсета инструмента имеет два значения.

Он указывается на офсет расстояния соответствующему номеру выбранного инструмент для включения функции офсета. Номер офсета 0 или 00 указывает что офсет инструмента отключается.

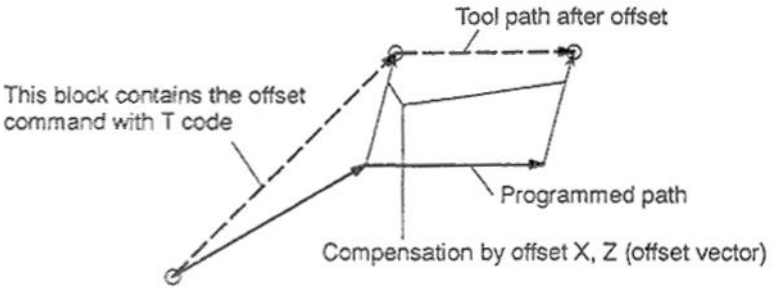
2.14.1.5 Офсет

Существует два типа офсета: офсет износа и офсет геометрии инструмента

Пояснения

Офсет износа инструмента

Траектория инструмента сдвигается на значения офсета X, Y, Z от запрограммированного пути. Расстояние офсета соответствует номеру офсета в T коде добавляемому или вычитаемому к позиции каждого запрограммированного блока.



Tool path after offset – Траектория инструмента после офсета
This block contains the offset command with T code – Этот блок содержит команду офсета с T кодом
Programmed path – Запрограммированный путь
Compensation by offset X, Z – Сдвиг на значение X, Z (вектор офсета)
Рис. 14.1.5(a) Движение офсета (1)

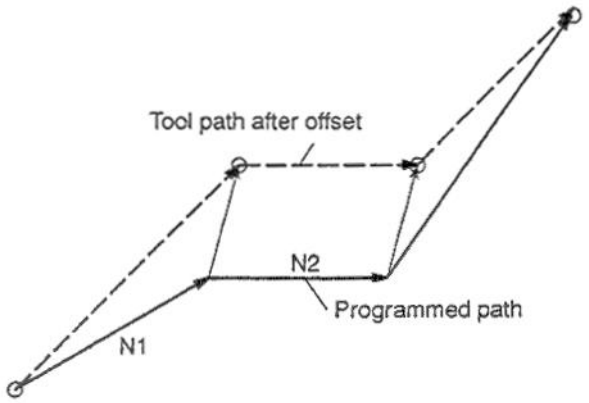
Вектор офсета

На рисунке 14.1.5(a) вектор X, Y, и Z называется фактором офсета. Значение компенсации равно величине этого вектора

Отмена офсета

Офсет отменяется T кодом с номером офсет 0 или 00. В конце блока отмены офсета, вектор офсета приравнивается к 0.

N1 X50.0 Z100.0 T0202; Создает вектор офсета соотв. офсету номер 02.
N2 X200.0;
N3 X100.0 Z250.0 T0200; Отмена вектора офсета



Tool path after offset – Траектория инструмента после офсета
Programmed path – Запрограммированный путь
Рис. 14.1.5(b) Движение офсета (2)

После первого включения питания и сброса нажатием на панели MDI ли сброса сигналом ЧПУ, офсет отменяется.
Параметр LVK (5003#6) может установлен в положение при котором значение офсета не будет сбрасываться.

Только Т код

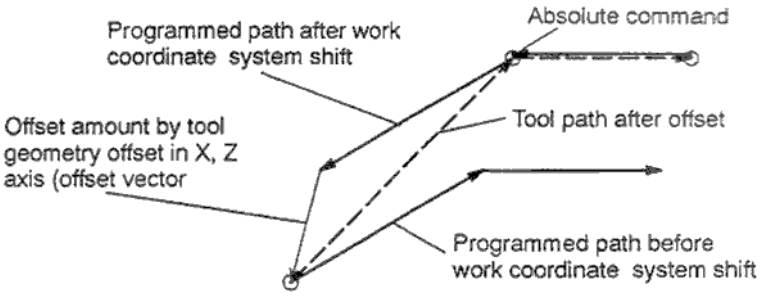
Если в блоке указан только Т код, инструмент перемещается на значение офсета без команды перемещения. Движение выполняется быстрой подачей в режиме G00. В других режимах используется подача обработки.

ВНИМАНИЕ

Если указано G50 X_ Z_ T_;, инструмент не перемещается.
Используется система координат, в которой установлена позиция инструмента X, Z. Позиция инструмента получается вычитание значения офсета из запрограммированной позиции.

Офсет геометрии инструмента

Система координат сдвигается на значение офсета геометрии по X, Y, Z. Значение офсета соответствует номеру указанному в коде и добавляется или вычитается из текущей позиции.



Absolute command – Абсолютная команда
Tool path after offset – Траектория инструмента после офсета
Programmed path before work coordinate system shift – Запрограммированный путь до сдвига координат
Programmed path after work coordinate system shift – Запрограммированный путь после сдвига координат
Offset amount by tool geometry offset in X, Z – Сдвиг на величину офсета по осям X, Z

Рис. 14.1.5(c) Движение офсета геометрии инструмента

ПРИМЕЧАНИЕ

Так же как и офсет износа, инструмент может быть компенсирован в соответствии с параметром LGT (5002#4) для добавления или вычитания запрограммированной точки к каждому блоку.

Отмена офсета

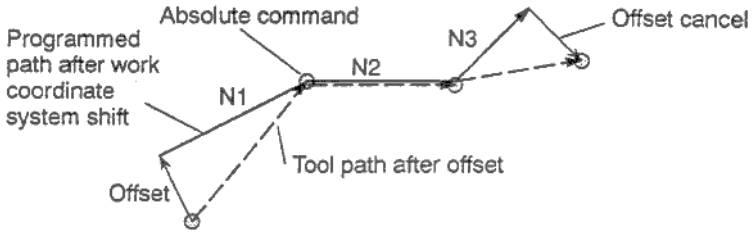
Офсет отменяется программированием офсет 0, 00 или 0000.

ПРИМЕЧАНИЕ

Если бит 5 (LGC) параметра 5002 установлен в 0, указание офсета 0 или 00 не отменяет офсет.

Примеры

1. Указан офсет геометрии и износа инструмента с помощью двух цифр T кода инструмента (бит 1 LGN параметра 5002 установлен в 0).
   N1 X50.0 Z100.0 T0202; Создает вектор офсета соотв. офсету номер 02.
   N2 X200.0;
   N3 X100.0 Z250.0 T0200; Отмена вектора офсета

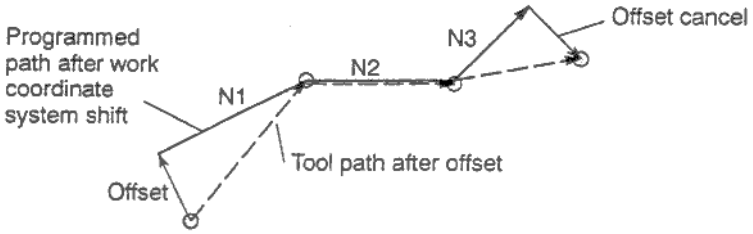


Tool path after offset – Траектория инструмента после офсета
Programmed path after work coordinate system shift – Запрограммированный путь после сдвига координат
Absolute command – Абсолютная команда
Offset cancel – Отмена офсета

ПРИМЕЧАНИЕ

Если бит 5 (LGC) параметра 5002 установлен в 0, указание офсета 0 или 00 не отменяет офсет.

2. Геометрический офсет не отменен офсетом номер 0 (параметр 5002#1).
   N1 X50.0 Z100.0 T0202; Выбор инструмента
   N2 X200.0;
   N3 X100.0 Z250.0 T0200; Отмена вектора офсета



2.14.1.6 Команды G53, G28, G30, G0.1 с активным офсетом

Приводится описание этих операций с включенным офсетом при позиционировании инструмента.

Описание

Возврат в точку ссылки (G28) и G53 при активном офсете инструмента

Выполнение возврата в точку ссылки G28 или G53 при активном офсете позиции инструмента не отменяет офсета вектора позиции инструмента. Дисплей абсолютной позиции работает в соответствии с значением бита 4 (LGT) параметра 5002.

LGT = 0 (Компенсация геометрии инструмента основана на сдвиге системы координат)

		Офсет позиции инструмента (без опций)	Компенсация геометрии инструмента	Компенсация износа геометрии
Дисплей Абсолютной позиции	Блок возврата в точку ссылки или команда G53	Вектор не изменяется. Координата отображается как и при отмене офсета	Осуществляется сдвиг. Система координат сдвигается в соотв. с компенсацией геометрии инструмента.	Вектор не изменяется. Координата отображается как и при отмене офсета
	Следующий блок	Вектор изменяется	Координаты сдвигаются соответственно значению компенсации геометрии инструмента	Вектор изменяется.

LGT = 1 (Компенсация геометрии инструмента основана на движении инструмента)

		Офсет позиции инструмента (без опций)	Компенсация геометрии инструмента	Компенсация износа геометрии
Дисплей Абсолютной позиции	Блок возврата в точку ссылки или команда G53	Вектор не изменяется. Координата отображается как и при отмене офсета	Вектор не изменяется. Координата отображается как и при отмене офсета	Вектор не изменяется. Координата отображается как и при отмене офсета
	Следующий блок	Вектор изменяется	Вектор изменяется.	Вектор изменяется.

ПРИМЕЧАНИЕ

Бит 6 (DAL) параметра 3104 установлен в 0 (текущая позиция к которой применяется офсет отображается в дисплее абсолютной позиции.

Ручной переход в точку ссылки с активным офсетом инструмента

Выполнение ручного перехода в точку ссылки при активном офсете позиции инструмента не отменяет офсета вектора позиции инструмента. Дисплей абсолютной позиции работает в соответствии с значением бита 4 (LGT) параметра 5002.

LGT = 0 (Компенсация геометрии инструмента основана на сдвиге системы координат)

		Офсет позиции инструмента (без опций)	Компенсация геометрии инструмента	Компенсация износа геометрии
Дисплей Абсолютной позиции	До ручного перехода в точку ссылки	Вектор не изменяется. Координата отображается как и при отмене офсета	Осуществляется сдвиг. Система координат сдвигается в соотв. с компенсацией геометрии инструмента.	Вектор не изменяется. Координата отображается как и при отмене офсета
	Следующий блок	Вектор изменяется	Координаты сдвигаются соответственно значению компенсации геометрии инструмента	Вектор изменяется.

LGT = 1 (Компенсация геометрии инструмента основана на движении инструмента)

		Офсет позиции инструмента (без опций)	Компенсация геометрии инструмента	Компенсация износа геометрии
Дисплей Абсолютной позиции	До ручного перехода в точку ссылки	Вектор не изменяется. Координата отображается как и при отмене офсета	Вектор не изменяется. Координата отображается как и при отмене офсета	Вектор не изменяется. Координата отображается как и при отмене офсета
	Следующий блок	Вектор изменяется	Вектор изменяется.	Вектор изменяется.

ПРИМЕЧАНИЕ

Бит 6 (DAL) параметра 3104 установлен в 0 (текущая позиция к которой применяется офсет отображается в дисплее абсолютной позиции.

Отмена офсета позиции инструмента с помощью T00

При указании T00 при активном офсете позиции инструмента, отмена офсета зависит от значений следующих параметров.

Опция коррекции геометрии/износа выбрана.

LGN=0

LGN (5002#1)	LGT (5002#4)	LGC (5002#5)	
Номер офсета геометрии: 0: Равен номеру износа 1: Равен номеру инстр.	Компенсация геометрии 0: Основана на сдвиге системы координат 1: Основана на движении инструмента		Результат
LGT=0	LGT=0	LGC=0 LGC=1	Не отменяется Отменяется
		LWM (5002#6) Позиция инстр. включена: 0: С помощью T кода 1: Движением по оси	
	LGT=1	LWM=0 LWM=1	Отменяется Не отменяется

ПРИМЕЧАНИЕ

1. Если LGT=0, LWM не влияет
2. Если LGT=1, LWM не влияет даже если LGN = 0.

LGN=1

LGN (5002#1)	LGT (5002#4)	LGC (5002#5)	
Номер офсета геометрии: 0: Равен номеру износа 1: Равен номеру инстр.	Компенсация геометрии 0: Основана на сдвиге системы координат 1: Основана на движении инструмента		Результат
LGT=0	LGT=0	LGC не влияет	Отменяется
		LWM (5002#6) Позиция инстр. включена: 0: С помощью T кода 1: Движением по оси	
	LGT=1	LWM=0 LWM=1	Отменяется Не отменяется

ПРИМЕЧАНИЕ

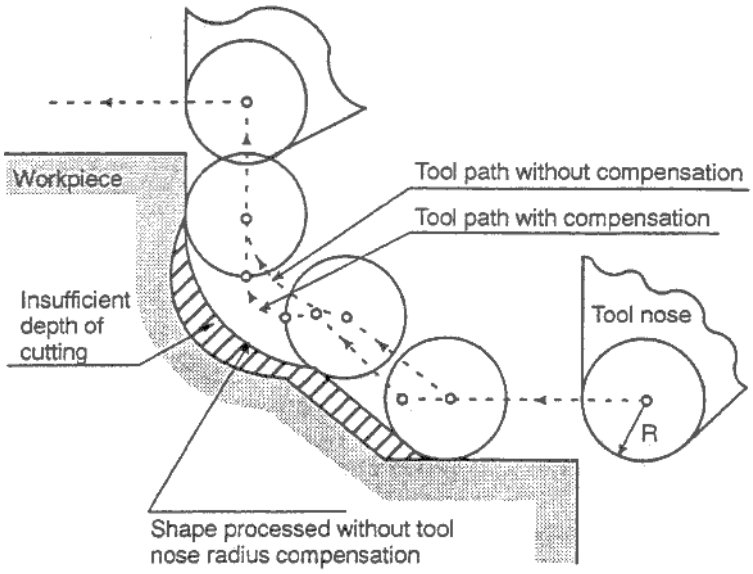
1. Если LGT=0, LWM не влияет
2. Если LGT=1, LWM не влияет даже если LGN = 0.

Если опция компенсации геометрии/офсета инструмента не выбрана

LGN (5002#1)	LGT (5002#4)	LGC (5002#5)	
Номер офсета геометрии: 0: Равен номеру износа 1: Равен номеру инстр.	Компенсация геометрии 0: Основана на сдвиге системы координат 1: Основана на движении инструмента	Офсет геометрии: 0: Не отменяется по T00 1: Отменяется по T00	Результат
LGT=Не влияет Номер офсета инструмента всегда использует последние цифры.	LGT=Не влияет Позиция инструмента всегда применяется на основании движения инструмента	LGC не влияет LWM (5002#6) Позиция инстр. включена: 0: С помощью T кода 1: Движением по оси	
		LWM=0 LWM=1	Отменяется Не отменяется

2.14.2 Обзор компенсации радиуса кромки инструмента

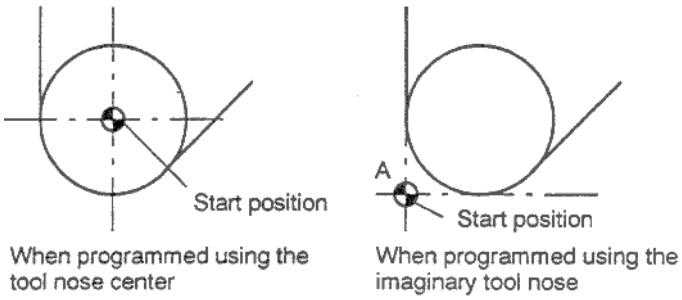
При использовании только функции офсета без компенсации радиуса кромки инструмента сложно произвести компенсацию точной формы при нарезании резьбы или круговой обработки. Компенсация кромки инструмента автоматически компенсирует следующие неточности:



Workpiece – деталь
Isufficient depth of cutting – Недостаточная глубина обработки
Tool path without compensation – Траектория инструмента без компенсации
Tool path with compensation – Траектория инструмента с компенсацией
Shape processed without tool nose compensation – Деталь обработанная с компенсацией радиуса кромки
Рис. 14.2 Траектория инструмента с компенсацией радиуса кромки инструмента

2.14.2.1 Воображаемая кромка инструмента

Кромка инструмента в точке на следующем рисунке в действительности не существует. Воображаемая кромка инструмента требуется потому что обычно более сложно установить действительный центр радиуса кромки инструмента для начала обработки чем воображаемую кромку инструмента.
Также если используется воображаемая кромка инструмента, радиус кромки инструмента не нужно учитывать.
Связь между позициями при установки инструмента в начальную точку приведена на следующем рисунке:

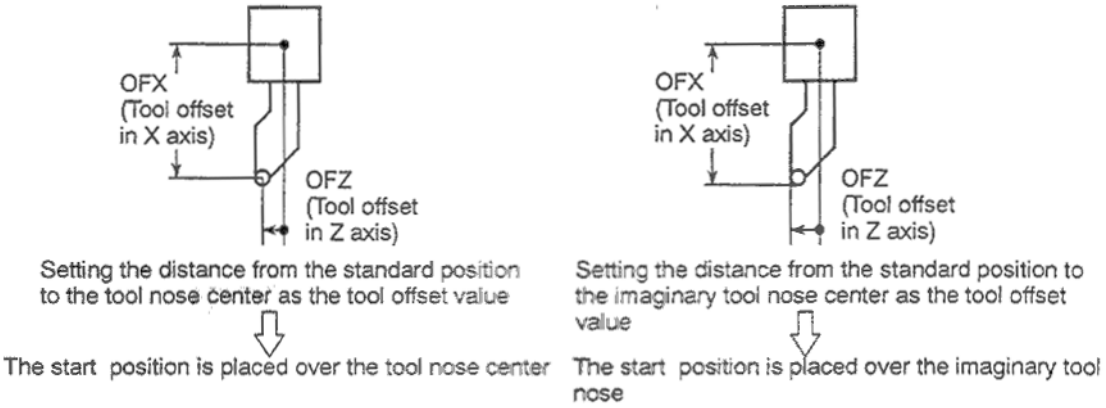


When programmed using tool nose center – При программировании с использованием центра радиуса кромки
When programmed using imaginary center – При программировании с использованием воображаемого центра
Рис. 14.2.1 (a) Центр и воображаемая кромка радиуса инструмента

ВНИМАНИЕ

В станках с позицией ссылки, стандартная позиция как центр револьверной головки может быть установлен как начальная позиция. Расстояние от стандартной позиции до центра радиуса кромки инструмента или воображаемой кромки инструмента устанавливается как значение офсета инструмента.

Установка расстояния от стандартной позиции в центр радиуса кромки инструмента как значение офсета равнозначно установки центра кромки инструмента как стандартной позиции. Для установки стандартного значения, обычно проще измерять расстояние от стандартной точки до воображаемой кромки инструмента.



Tool offset in X axis – Офсет инструмента по оси X
Setting the distance to the tool nose center – Установка расстояния от стандартной точки до центра кромки
Setting the distance to the imaginary tool nose – Установка расстояния от стандартной точки до воображаемой кромки инструмента
The start position is placed over the tool nose center – Начальная точка в центре кромки радиуса
The start position is placed over the imaginary tool nose – Начальная точка в воображаемой кромке инструмента

Рис. 14.2.1 (b) Перенос стандартной точки к кромке инструмента

Пока компенсация радиуса не выполнена, инструмент перемещается по запрограммированной траектории.

Если компенсация радиуса используется, выполняется точная обработка.

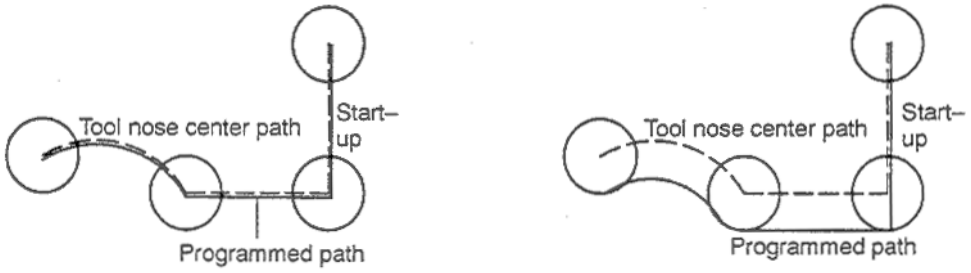
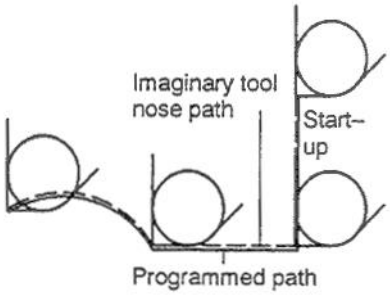


Рис. 14.2.1(c) Траектория инструмента с офсетом радиуса инструмента.

Без компенсации радиуса кромки инструмента, с воображаемым краем инструмента, запрограммированный путь не изменяется.

Если компенсация радиуса используется, выполняется точная обработка.



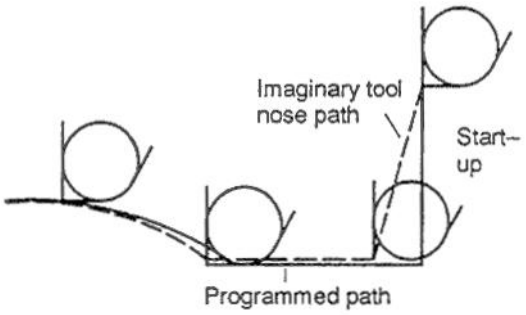


Рис. 14.2.1(d) Траектория инструмента с воображаемой кромкой инструмента

2.14.2.2 Направление воображаемой кромки инструмента

Направление воображаемой кромки инструмента определяется если смотреть со стороны инструмента во время обработки.

Направление воображаемой кромки инструмента может быть выбрано из восьми вариантов указанных на рис. 14.2.2.2. На этом рисунке показано отношение между инструментом и начальной позицией. Это используется при выборе офсета геометрии и износа инструмента.

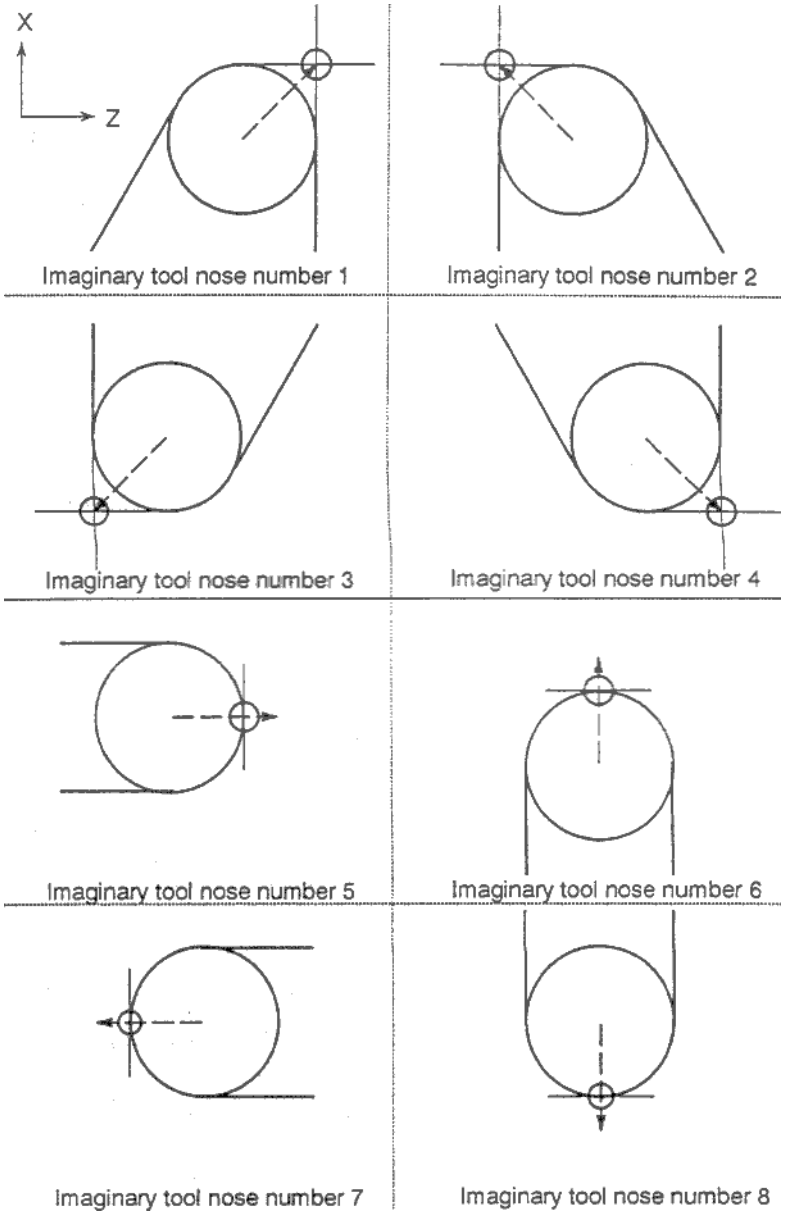
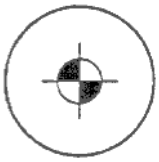


Рис. 14.2.2 Направление воображаемой кромки инструмента

Воображаемые кромки 0 и 9 используются если кромка инструмента совпадает с начальной точкой. Установите номер воображаемой кромки в адрес OFT для каждого номера офсета.

Бит 7 (WNP) параметра 5002 используется для определения направления номера офсета геометрии или номера офсета износа виртуальной кромки инструмента для компенсации радиуса кромки инструмента.

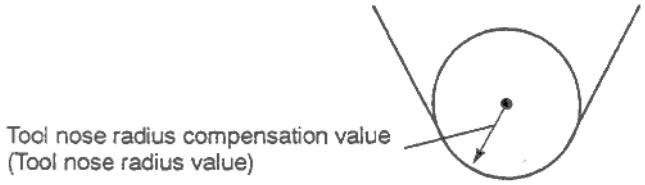


Номер воображаемой кромки инструмента от 0 до 9.

Ограничения

Плоскость
Направления виртуальной кромки инструмента может быть использовано только в G18 (Z-Z) плоскости. Для виртуальной кромки инструмента 0 или 9 возможны плоскости G17 и G19.

2.14.2.3 Номер офсета и величина офсета



Tool nose radius compensation value – Величина компенсации радиуса кромки инструмента

Это значение устанавливается на MDI в соответствии с номером офсета. Если используются опции компенсации геометрии и износа, значения офсета становится следующими:

Таблица 14.2.3(a) Номер и величина офсета

Номер офсета	OFX (величина офсета по X)	OFX (величина офсета по X)	OFR Величина компенсации радиуса	OFT Направл. Воображаемой кромки	OFY Величина офсета по Y
01	0.040	0.020	0.20	1	0.030
02	0.060	0.030	0.25	2	0.040
..	..	..	..	..	..
98	0.050	0.015	0.12	6	0.025
99	0.030	0.025	0.24	3	0.035

Если используются опции компенсации геометрии и износа, значения офсета становится следующими:

Таблица 14.2.3.(b) Офсет геометрии инструмента

Номер офсета геометрии инструмента	OFGX Величина офсета геометрии по X	OFGZ Величина офсета геометрии по Y	OFGR Величина геометрии офсета радиуса	OFT Направл. Воображаемой кромки	OFGY Величина офсета геометрии по Y
G01	10.040	50.020	0	1	70.020
G02	20.060	30.030	0	2	90.030
G03	0	0	0.20	6	0
G04	..	..	..	..	..
G05	..	..	..	..	..
..	..	..	..	..	..

Таблица 14.2.3.(c) Офсет износа инструмента

Номер офсета износа инструмента	OFGX Величина офсета износа по X	OFGZ Величина офсета износа по Y	OFGR Величина офсета износа радиуса	OFT Направл. Воображаемой кромки	OFGY Величина офсета износа по Y
W01	0.040	0.020	0	1	0.020
W02	0.060	0.030	0	2	0.030
W03	0	0	0.20	6	0
W04	..	..	..	..	..
W05	..	..	..	..	..
..	..	..	..	..	..

Компенсация радиуса кромки инструмента
В этом случае величина компенсации является суммой величины геометрии или износа инструмента. OFR = OFGR + OFWR

Направление воображаемой кромки инструмента
Направление воображаемой кромки инструмента может быть установления либо для офсета геометрии, либо для офсета износа. Эффективно последнее заданное значение.

Команда величины офсета
Номер офсета определяется тем же T кодом, что и при использовании для офсета инструмента. Подробнее см. 2.14.1.2.

ПРИМЕЧАНИЕ

Если номер офсета геометрии сделан общим для выбора инструмента параметром LGT (5002#1) и T код для которого офсет геометрии и износа отличается друг от друга, направление воображаемой кромки инструмента определяется правильным номером офсета геометрии.
Пример:
T0102
$OFR=RFGR_{01}+OFWR_{02}$
$OFT=OFT_{01}$

Установка диапазона значений офсета
Диапазон значений офсета следующий:

Система приращений	Метрическая система	Дюймовая система
IS-B	0 - +-999.999мм	0 - +-99.9999дюйм
IS-C	0 - +-999.9999мм	0 - +-99.99999дюйм

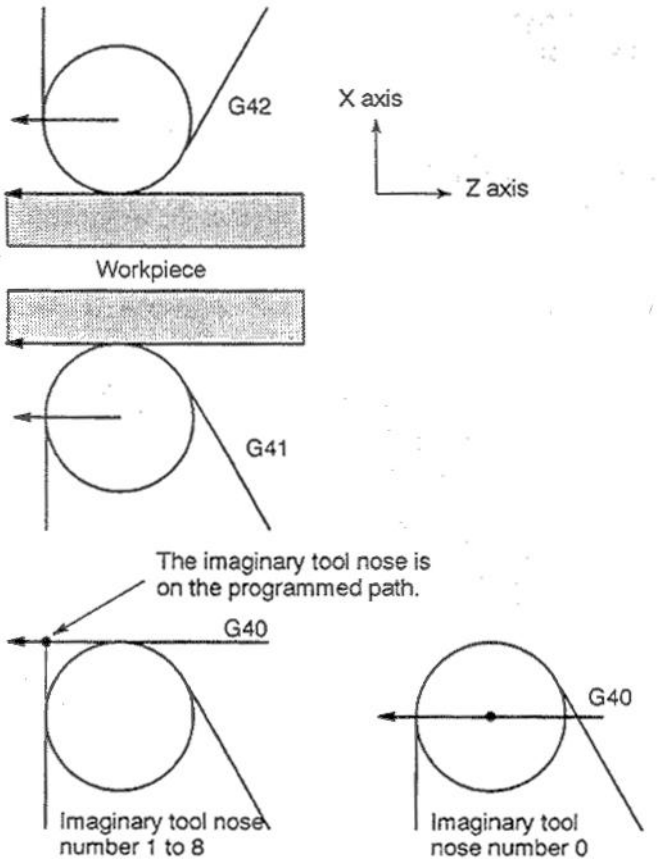
Значение офсета номер ноль всегда равно нулю. Для офсета 0 не может быть установлено значение.

2.14.2.4 Рабочая позиция и команда перемещения

При компенсации радиуса кромки инструмента, должна быть учтена позиция заготовки.

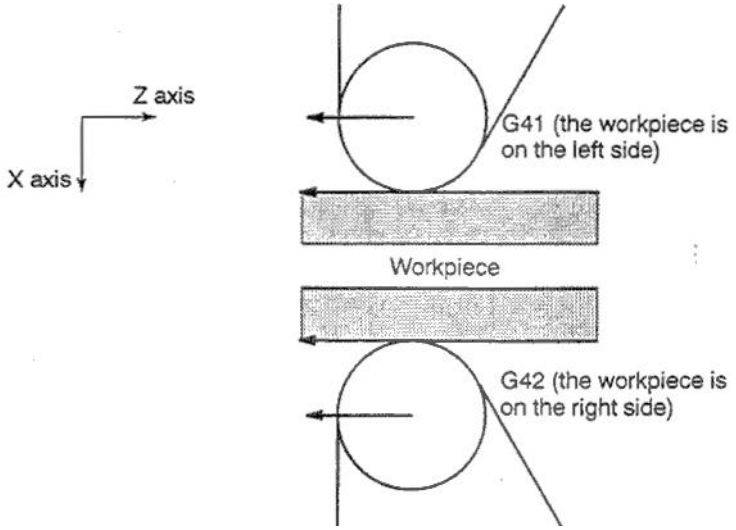
G код	Позиция заготовки	Траектория инструмента
G40	Отмена	Перемещение по запрограммированной траектории
G41	Правая сторона	Перемещение слева от запрограммированной траектории
G42	Левая сторона	Перемещение справа от запрограммированной траектории

Инструмент сдвигается на обратную сторону детали.



Workpiece – Деталь
The imaginary tool nose is on the programmed path – Воображаемая кромка инструмента на Запрограммированной траектории
Imaginary tool nose number 1 to 8 – Воображаемая кромка 1 – 8

Позиция детали может быть изменена установкой системы координат как показано на следующем рисунке:

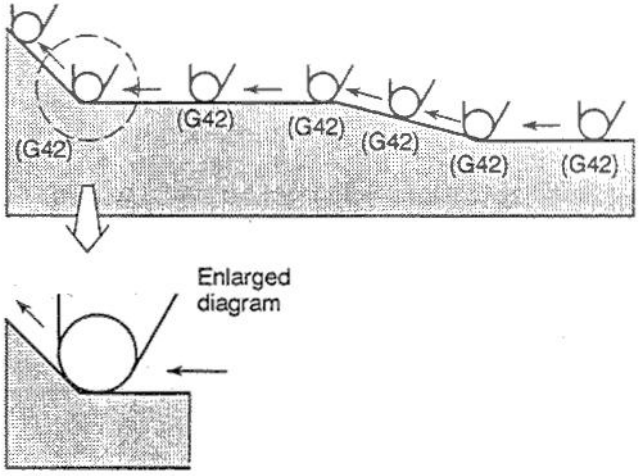


Workpiece – Деталь
G41 – Деталь слева
G42 – Деталь справа

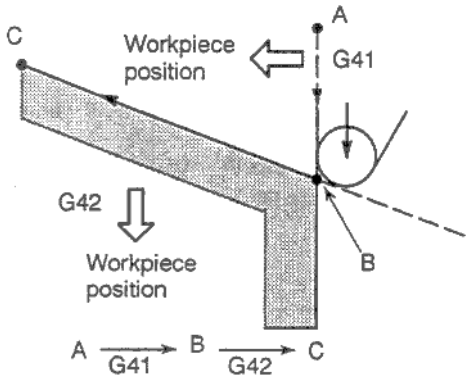
Примечание: Если значение компенсации радиуса кромки инструмента отрицательно, позиция детали изменяется.

G40, G41, G42 модальные коды. Если код G41 активен, его не нужно повторно программировать. Если это сделать, компенсация будет работать неправильно. Также не следует повторно программировать G42.

Перемещение инструмента во время не перемещающейся детали
Если инструмент перемещается, кромка инструмента остается в контакте с деталью.

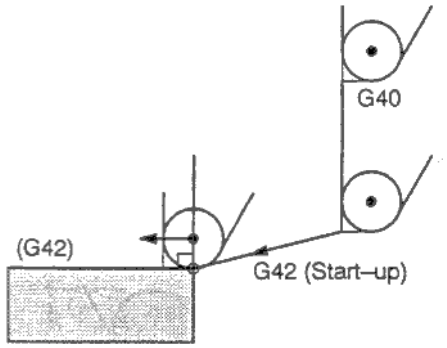


Перемещение инструмента при перемещающейся детали
Позиция детали у запрограммированного угла показана на следующем рисунке:

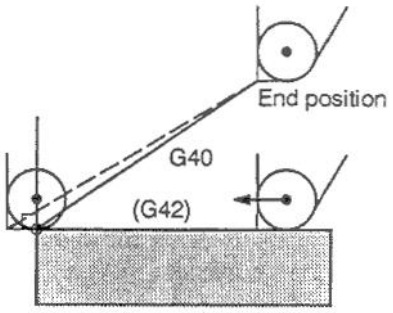


Хотя деталь не существует с правой стороны в данном случае, предполагается движением от A до B. Позиция детали должна быть изменена в следующем блоке после начального блока. В этом примере если блок описывает перемещение от A до B является начальным блоком, траектория инструмента не будет такой как показана.

Начало
Блок в котором изменяется режим G41, G42, из G40 называется начальным блоком.
Выполняется промежуточное движение инструмента в начальном блоке. В блоке после начального, кромка инструмента позиционируется вертикально в блок позицию начального блока.

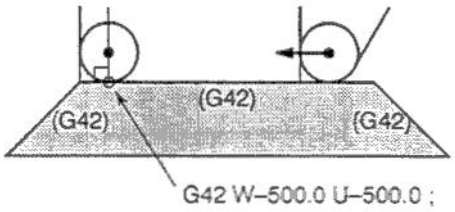


Отмена офсета
Блок в котором производится переключение в режим G40, называется блоком отмены офсета. Центр кромки инструмента перемещается в запрограммированную позицию в блоке отмены офсета. Перемещение показано на следующем рисунке:



Указание G41/G42 в режиме G41/G42

Если повторно запрограммирован режим G41/G42, центр кромки инструмента позиционируется по вертикали в запрограммированную позицию предыдущего блока.

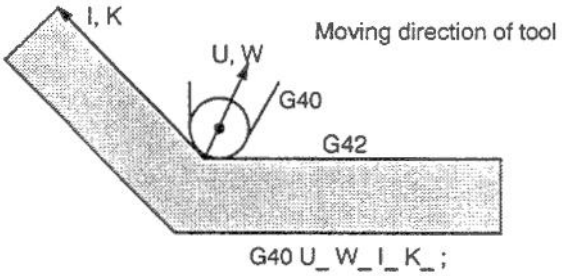


В блоке в котором впервые запрограммирован G41/G42, позиционирование кромки инструмента не производится.

Перемещение инструмента если направление перемещения инструмента в блоке с G40 отличается от направления детали

Если вы хотите совершить отвод инструмент в направлении X(U) и Z(W) при этом отменяя компенсацию радиуса кромки инструмента в конце обработки, укажите следующее:

G40 X(U)_ Z(W)_ I_ K_;



Moving direction of tool - Направление перемещения инструмента

Позиция детали указанная в адресе I и K такая же как и в предыдущем блоке. Если I и/или K указано с G40, I и K игнорируются.

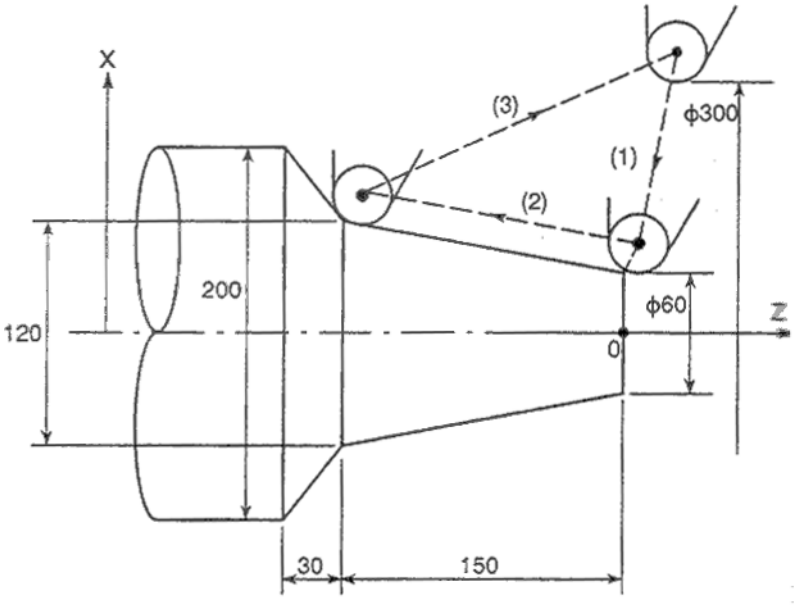
G40 X_ Z_ I_ K_;	Компенсация радиуса кромки инструмента
G40 G02 X_ Z_ I_ K_;	Круговая интерполяция

G40 G01 X_ Z_;

G40 G02 X_ Z_ I_ K_; Отмена режима офсета (I и K не эффективны)

Значения после I и K всегда должны указываться как радиус.

Пример



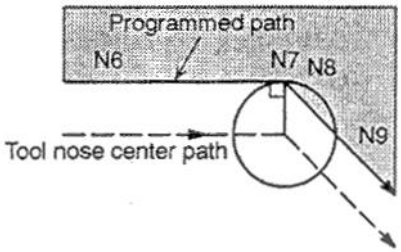
Режим G40
1. G42 G00 X60.0;
2. G01 X120.0 W-150.0 F10;
3. G40 G00 X300.0 W150.0 I40.0 K-30.0;

2.14.2.5 Замечания к компенсации радиуса кромки инструмента

Перемещение инструмента если два и более блоков без команды перемещения не должны программироваться последовательно

1. M05;
2. S210;
3. G04 X1000; Задержка
4. G01 U0; Нулевое перемещение
5. G98; Только G код
6. G10 P01 X10.0 Z20.0 R0.5 Q2; Изменение офсета

Если два или более блоков указаны последовательно, инструмент переходи в позицию по вертикали по запрограммированной траектории в конце предыдущего блока. Если команда перемещения не запрограммирована, это движение инструмента выполняется только в одном блоке.



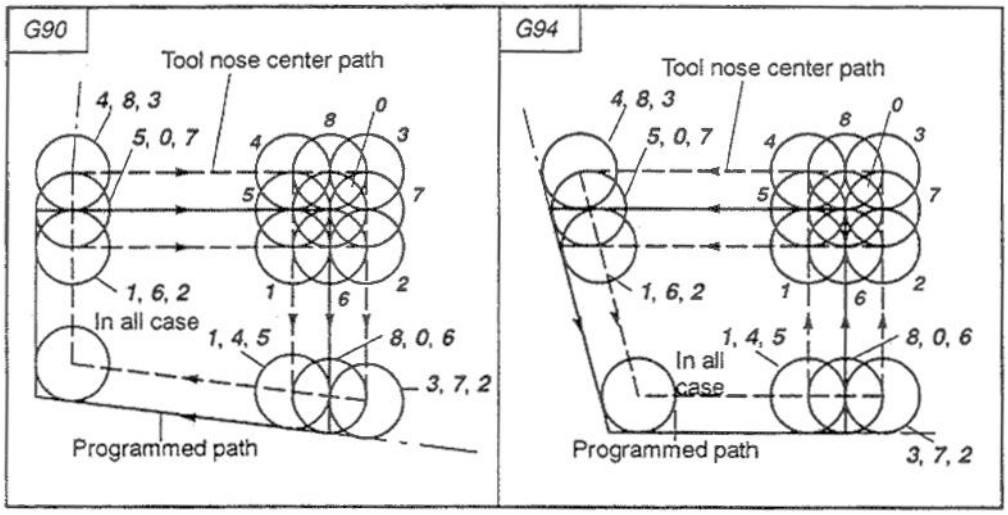
Programmed path – Запрограммированная траектория
Tool nose center path – Центр кромки инструмента

Компенсация радиуса кромки инструмента с G90 или G94

Компенсация радиуса кромки инструмента с G90 (цикл обработки внутреннего/внешнего диаметра) или G94 (цикл обработки торца) работает следующим образом:

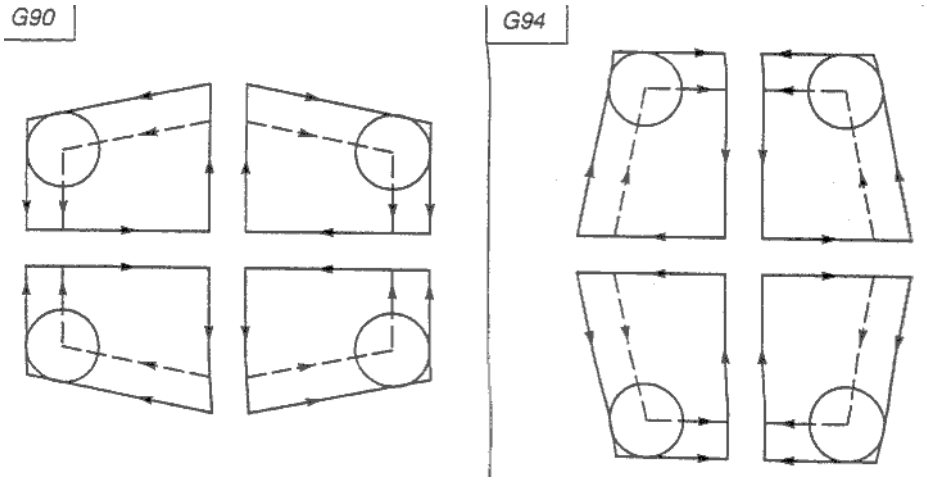
1. Движения для номера воображаемой кромки инструмента.

Для каждого прохода в цикле, кромка инструмента в основном параллельна запрограммированному пути.



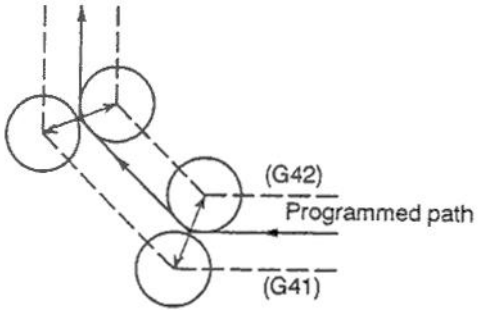
Programmed path – Запрограммированная траектория
Tool nose center path – Центр кромки инструмента

2. Направление офсета
Направление офсет показано на следующем рисунке в соответствии с G41/G42.



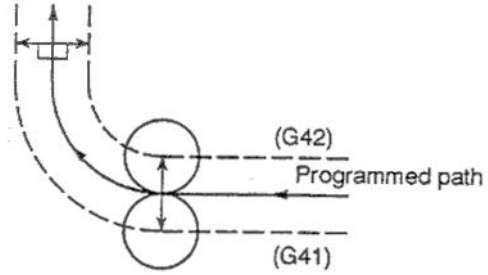
Компенсация радиуса кромки инструмента с G71 – G76 или G78
Если запрограммирован один из этих циклов, цикл отклоняется на величину коррекции радиуса кромки инструмента. В этом цикле не производится вычисление пересечения.
G71 (снятие материала на диаметре)
G72 (снятие материала на торце)
G73 (Повторение шаблона)
G74 (Торцевое многопроходное сверление)
G75 (Сверление на внутреннем / внешнем диаметре)
G76 (Многопроходное нарезание резьбы)
G78 (Цикл резьбы)

Компенсация радиуса кромки инструмента при снятии фаски
Движение производится следующим образом:



Programmed path – Запрограммированная траектория

Компенсация радиуса кромки инструмента при вставке дуги в угол
Движение производится следующим образом:



Programmed path – Запрограммированная траектория

Компенсация радиуса кромки инструмента при программировании блока с MDI
В этом случае компенсация радиуса кромки не производится

2.14.3 Подробное описание компенсации радиуса кромки инструмента

В этой главе подробно описывается перемещение при использовании компенсации радиуса кромки инструмента. В этой главе содержатся следующие темы:

2.14.3.1 Общие сведения
2.14.3.2. Начальное перемещение инструмента
2.14.3.3 Перемещение инструмента в режиме офсета
2.14.3.4 Перемещение инструмента в режиме отмены офсета
2.14.3.5 Проверка интерференции
2.14.3.6 Излишняя обработка при компенсации радиуса кромки
2.14.3.7 Коррекция при снятии фаски и закруглении
2.14.3.8 Ввод команд с MDI
2.14.3.9 Общие предостережения при операциях с офсетом
2.14.3.10 G53, G28, G30 и G30.1 команды при компенсации радиуса кромки

2.14.3.1 Общие сведения

Вектор офсета центра радиуса кромки инструмента является двумерным, равен величине офсет указанной в Т коде и вычисляется в ЧПУ. Изменения перемещения производятся от блока к блоку.

Вектор офсета (здесь и далее просто вектор) создается автоматически внутри ЧПУ для правильной обработки запрограммированного контура данным инструментом с точным офсетом.

Вектор удаляется после сброса.

Вектор всегда представляет собой изменение размера инструмента. Правильное понимание действия вектора необходимо для аккуратного программирования. Прочитайте все описания далее для правильного создания вектора.

G40, G41, G42

G40, G41, G42 используются для удаления или создания вектора. Эти коды используются с G00, G01, G02, G03 или G33 для указания перемещения инструмента (сдвиг).

G40 Отмена компенсации радиуса кромки инструмента
G41 Левая компенсация траектории инструмента
G42 Правая компенсация траектории инструмента

G41 и G42 должны указываться в режиме отключенного офсета G40.

Отмена коррекции

Система переходит в режим отмены коррекции сразу после включения питания, сброса ЧПУ или завершения программы командой M02 или M30 (в зависимости от настроек производителя станка, станок может не отключить коррекцию). В режиме отмены офсета, вектор становится нулевым и инструмент перемещается в точку запрограммированной траектории. Программа должна заканчиваться в режиме отмены офсета. Если она заканчивается в режиме офсета, инструмент не может быть позиционирован в конечную точку и инструмент останавливается на некотором расстоянии до конечной точки.

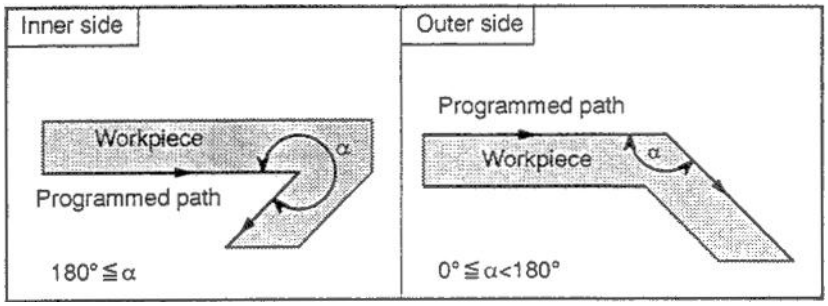
Запуск

Если блок удовлетворяющий всем следующим условиям выполняется в режиме отмены офсета инструмента, система входит в режим офсета. Момент включения офсета называется запуском.

- G41 или G42 содержаться в блоке или указываются для перехода системы в режим офсета. Момент включения офсета называется запуском.
- Номер офсета для коррекции радиуса кромки инструмента не может равняться 00.
- Перемещения X или Z указываемые в блоке и расстояние перемещения не должно быть нулевым.

Команды движения по дуге (G02 или G03) не допускаются при запуске офсета. Если они будут указаны, появится сигнал тревоги 34. При запуске офсета считывается два блока. Первый блок выполняется и второй вносится в буфер для вычисления движения компенсации. В поблочном режиме считывание и выполнение производится по 2 блока, после чего производится остановка.
В последовательных операциях два блока считываются наперед так, чтобы ЧПУ могло правильно выполнить текущий блок.

Внутренняя и внешняя сторона
Если создается угол более 180 градусов, считается что производится обработка внутренней стороны поверхности. Если угол от 0 до 180 градусов, угол считается внешним.



Programmed path – Запрограммированная траектория
Inner side – Внутренняя сторона
Outer side – Внешняя сторона
Workpiece - Деталь

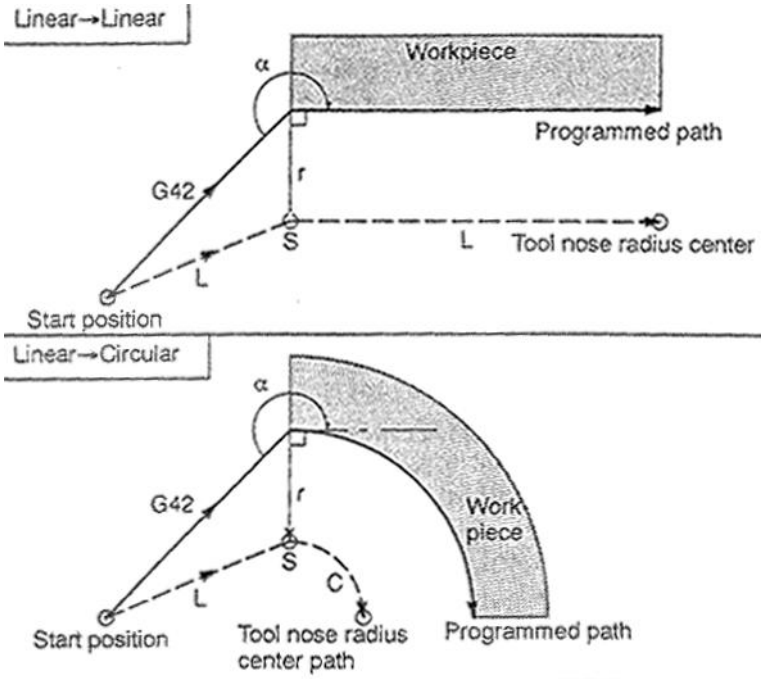
Значения символов
На следующих рисунках используются эти символы:
- **S** обозначает позицию в которой блок выполняется один раз
- **SS** обозначает позицию в которой блок выполняется дважды
- **SSS** обозначает позицию в которой блок выполняется трижды
- **L** обозначает что инструмент перемещается по прямой линии
- **C** обозначает что инструмент перемещается по дуге
- **r** обозначает величину компенсацию радиуса кромки инструмента
- Пересечение является позицией в которой запрограммированная траектория из двух блоков пересекается друг с другом после сдвига на **r**.
- **O** обозначает центр радиуса кромки инструмента.

2.14.3.2 Начальное перемещение инструмента

При переключении из режима отмены офсета в режим офсета, инструмент перемещается как показано на следующих рисунках.

Перемещение инструмента во внутренней стороне угла

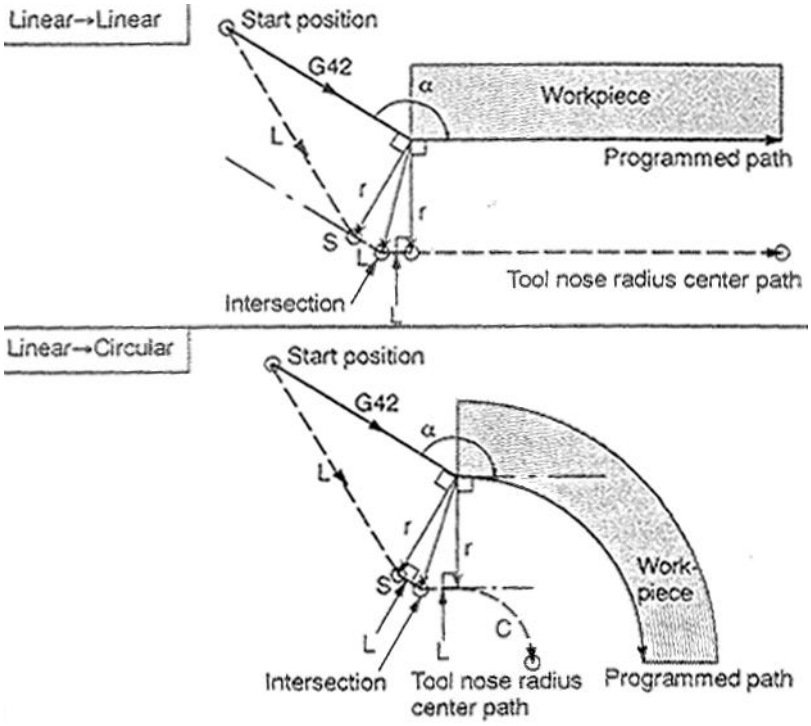
(180° ≤ α)



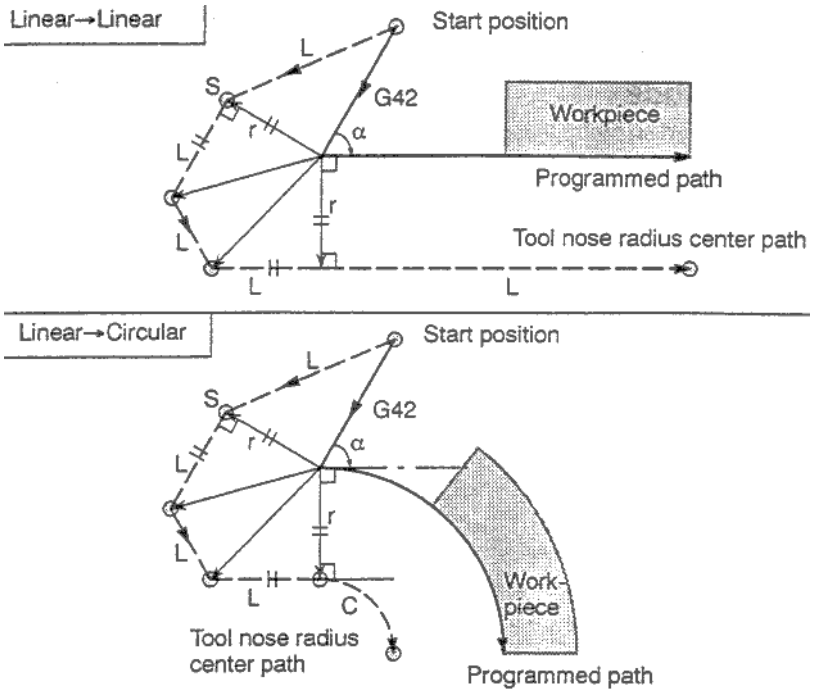
Programmed path – Запрограммированная траектория
Workpiece – Деталь
Tool nose radius center path – Траектория центра кромки радиуса инструмента

Перемещение инструмента по внешней стороне угла под тупым углом

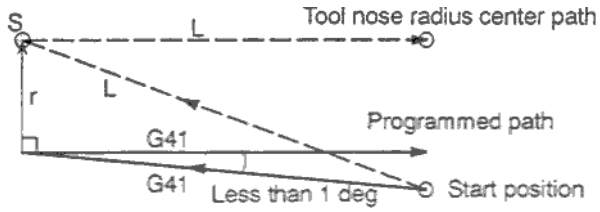
(90° ≤ α ≤ 180°)



Перемещение инструмента по внешней стороне угла под острым углом
(α < 90°)

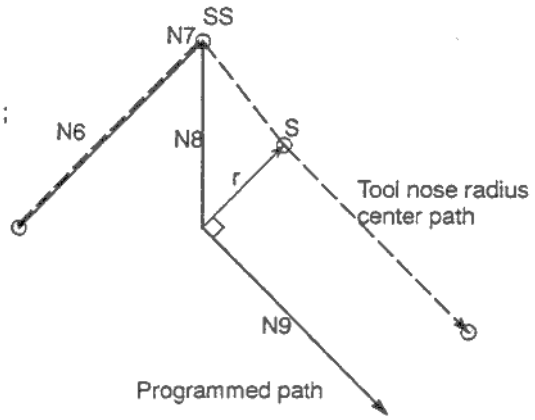


Перемещение инструмента из линии в линию с углом меньше 1 градуса
(α < 1°)



Блок без движения указанном при запуске офсета

```
G91 G40 ... ;
      :
N6 U100.0 W100.0 ;
N7 G41 U0 ;
N8 U–100.0 ;
N9 U–100.0 W100.0 ;
```

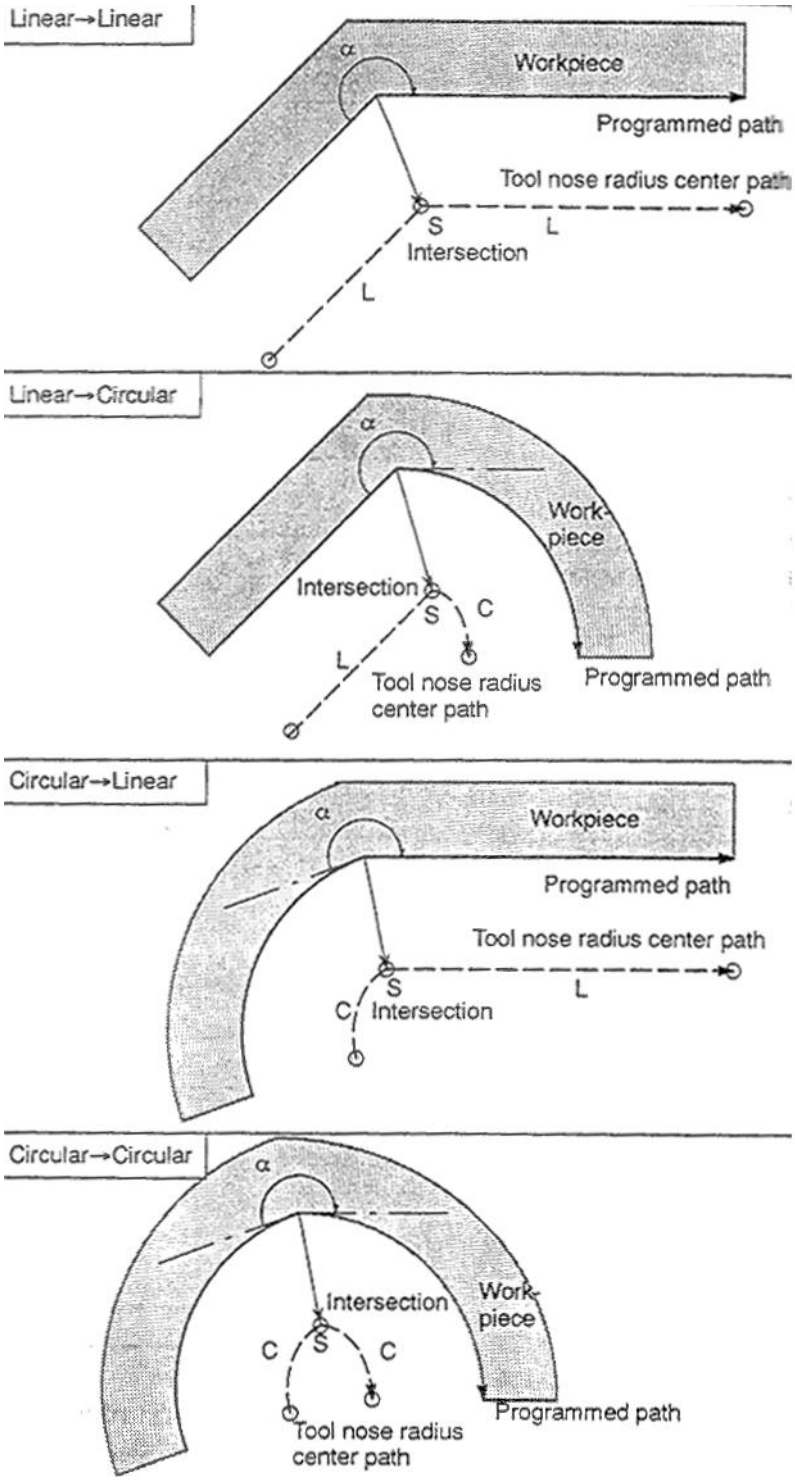


ПРИМЕЧАНИЕ
Описание блоков не производящих движение см. в 2.14.3.3.

2.14.3.3 Перемещение инструмента в режиме офсета

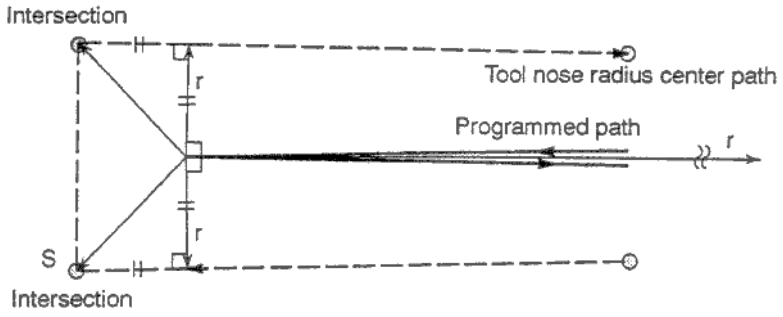
В режиме офсета инструмент перемещается следующим образом:

Перемещение инструмента по внутреннему контуру
(180° ≤ α)

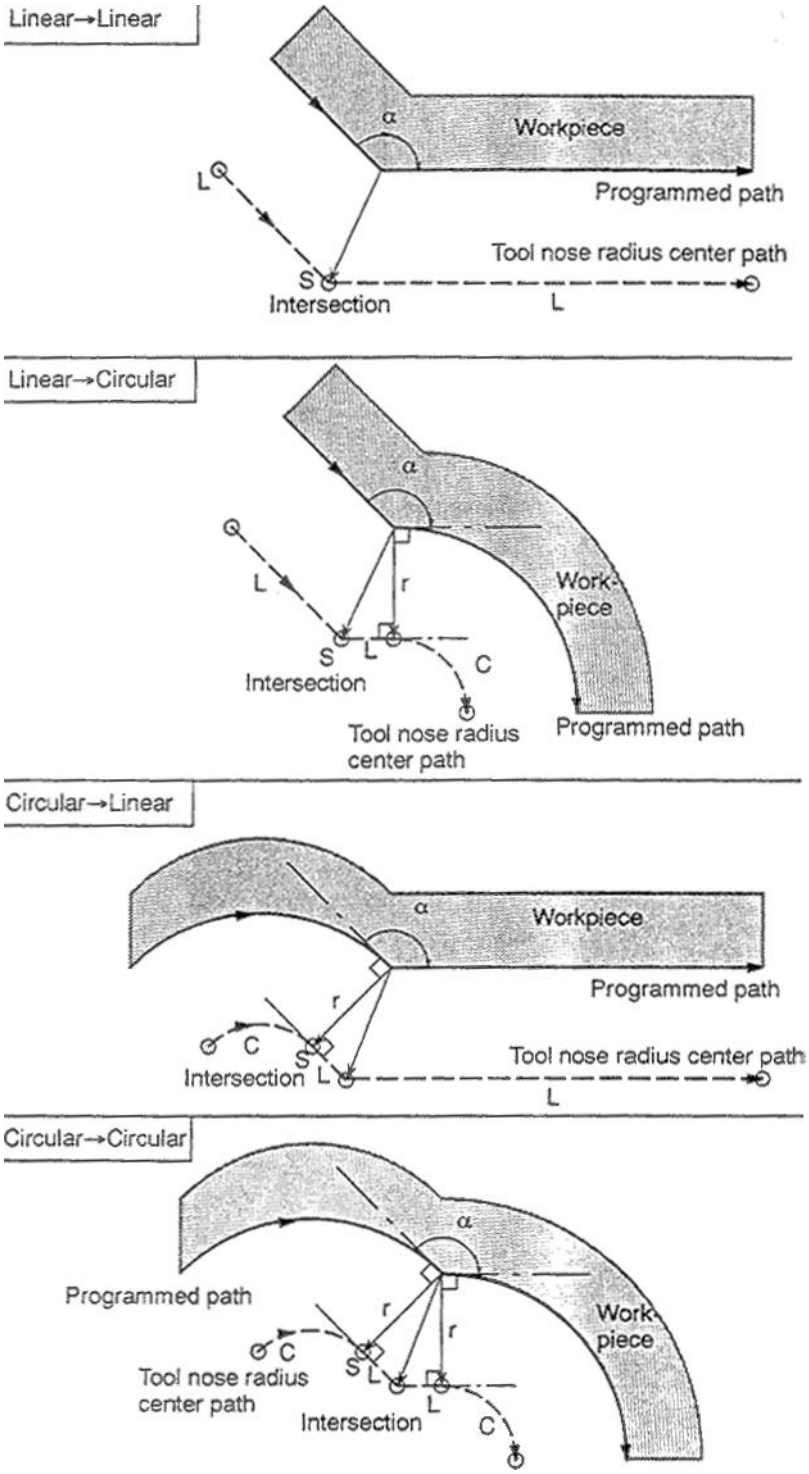


Programmed path – Запрограммированная траектория
Workpiece – Деталь
Tool nose radius center path – Траектория центра кромки радиуса инструмента
Intersection - Пересечение

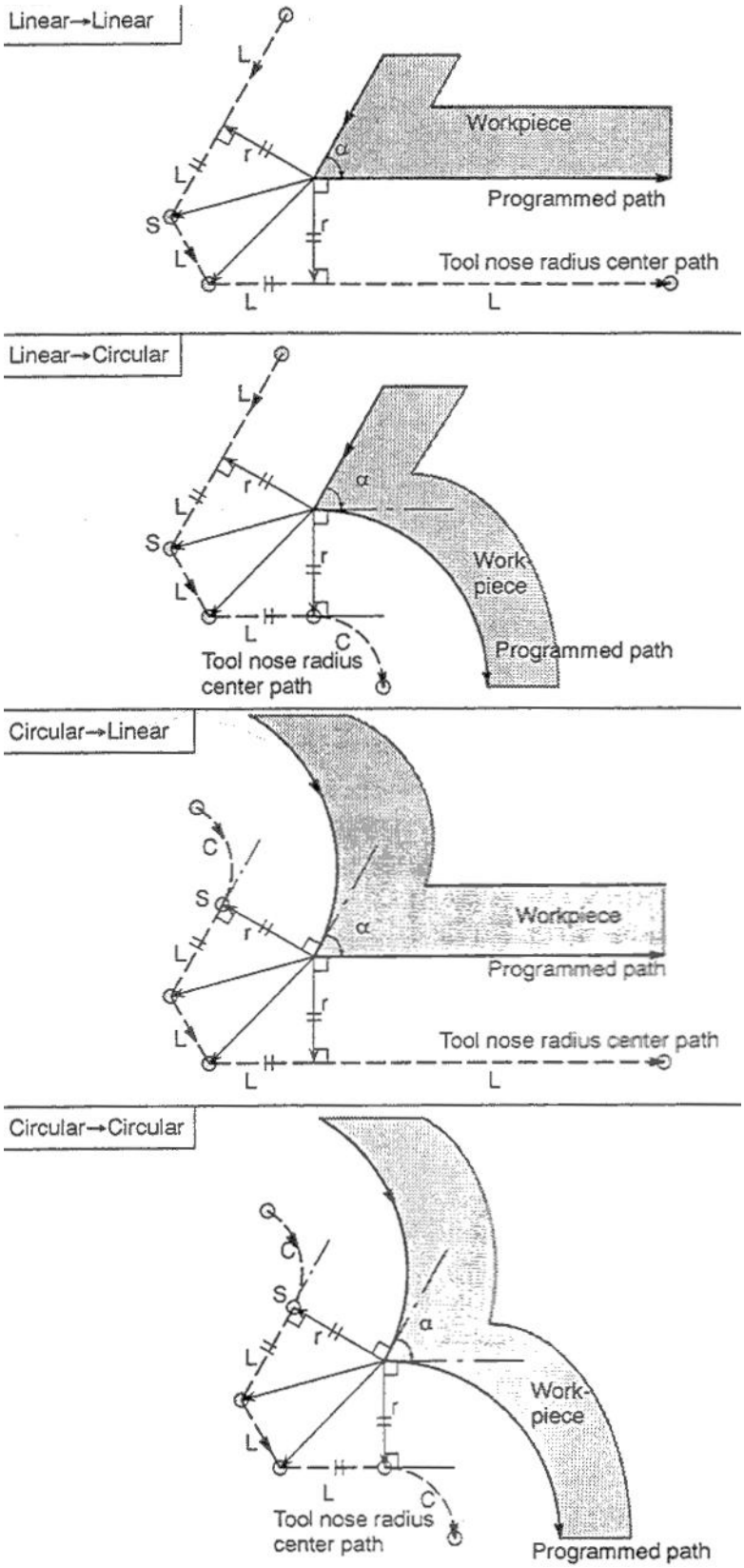
Перемещение инструмента от линии к линии с углом меньше 1 градуса с ненормально длинным вектором



Перемещение инструмента по внешней стороне угла под тупым углом
(90° ≤ α ≤ 180°)



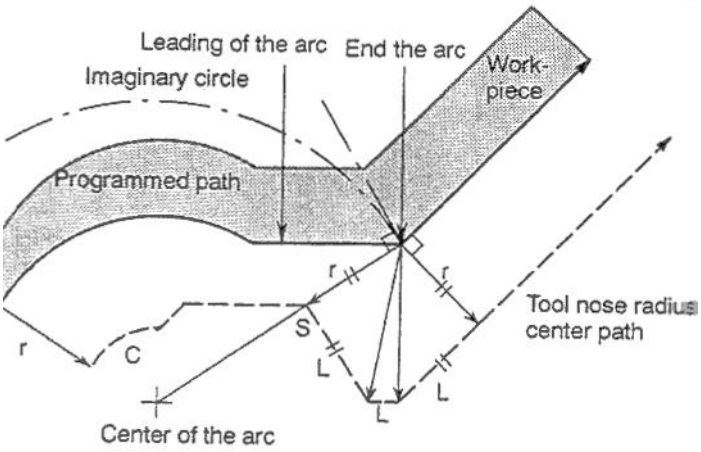
Перемещение инструмента по внешней стороне угла под острым углом
(α < 90°)



Исключительный случай

Конечная точка дуги не на дуге

Если конец линии перед дугой запрограммирован как конец дуги по ошибке как на следующем рисунке, система считает что компенсация радиуса кромки инструмента выполняется так что воображаемый центр равен центру дуги и проход производится в конечную позицию. На основании этого предположения система создает вектор и выполняет компенсацию. Результирующий путь центра радиуса кромки инструмента отличается от запрограммированного за счет предыдущей линии перед дугой.

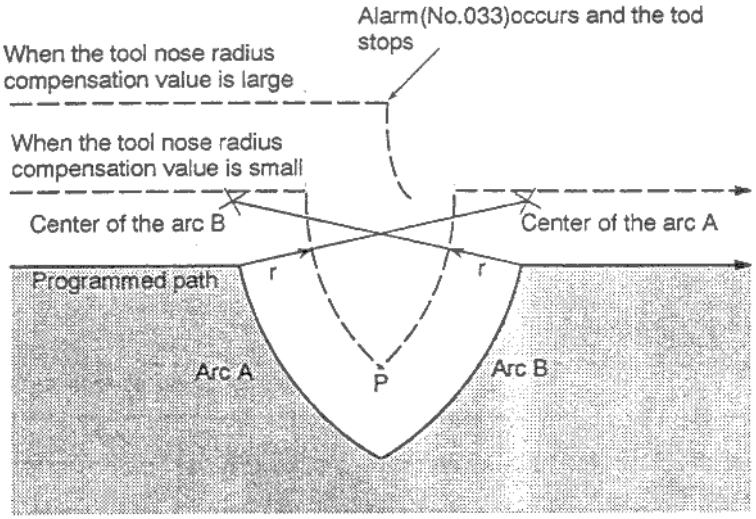


Programmed path – Запрограммированная траектория
Workpiece – Деталь
Tool nose radius center path – Траектория центра кромки радиуса инструмента
Imaginary circle – Воображаемый круг
Center of the arc – Центр дуги
End the arc – Конец дуги
Leading of the arc – Начало дуги

То же описание применяется к перемещению инструмент между двумя окружностями.

Отсутствие внутреннего пересечения

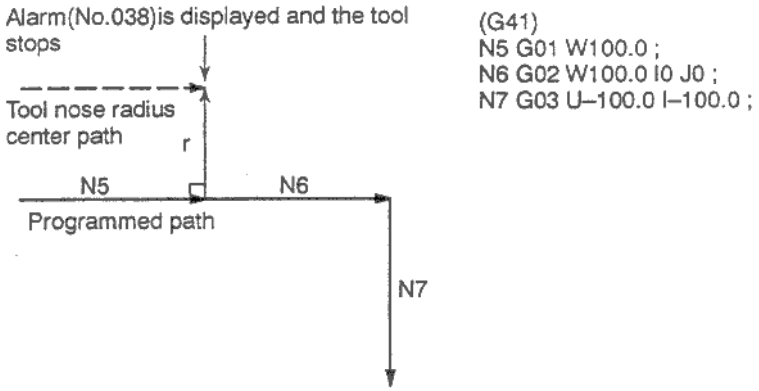
Если значение компенсации радиуса кромки инструмента достаточно мало, два контура окружности выполненных после компенсации пересекаются в позиции P. Пересечение P может не произойти если указано слишком большое значение компенсации радиуса кромки. В этом случае появится сигнал тревоги 33 в конце предыдущего блока и инструмент остановится. На следующем примере инструмент в результате обработки A и B пересекается в P. При указании слишком большого значения компенсации, пересечение не произойдет.



Programmed path – Запрограммированная траектория
Center of the arc – Центр дуги

Центр дуги в начальной или конечной точке дуги

Если центр дуги равен начальной или конечной точке, появится сигнал тревоги 38, инструмент остановится в предыдущем блоке.



Tool nose radius center path – Траектория центра кромки радиуса инструмента
Programmed path – Запрограммированная траектория

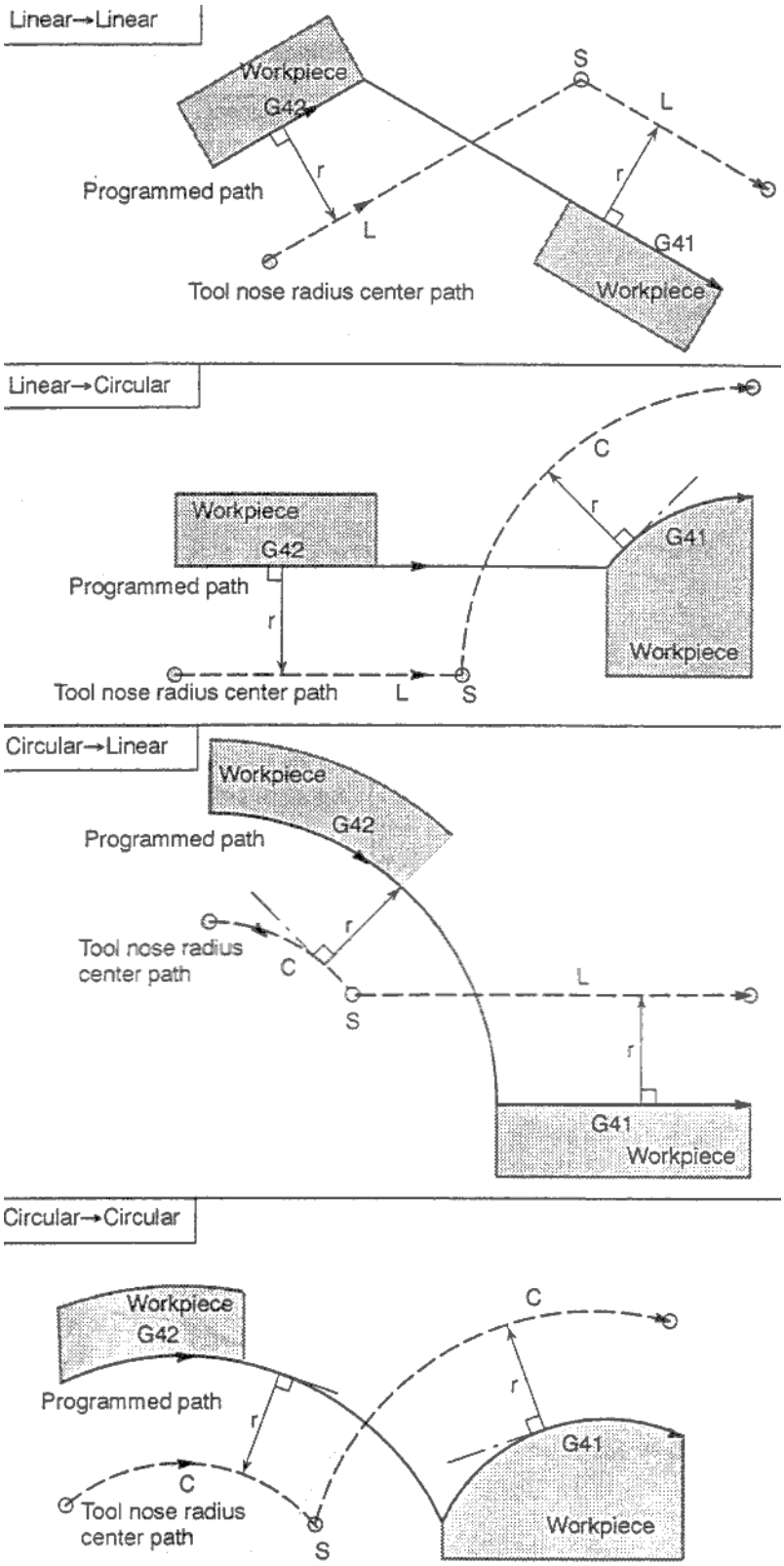
Изменение направления офсета

Направление офсета определяется G кодом G41 и G42 и знаком офсета для значения компенсации радиуса кромки инструмента следующим образом:

Знак значения офсета G код	+	-
G41	Офсет с левой стороны	Офсет с правой стороны
G42	Офсет с правой стороны	Офсет с левой стороны

Направление офсета может быть изменено в режиме офсета если направление офсета изменяется в блоке, вектор генерируется в пересечении данного и предыдущего блока. Изменение офсета невозможно при включении офсета и блоке следующем за ним.

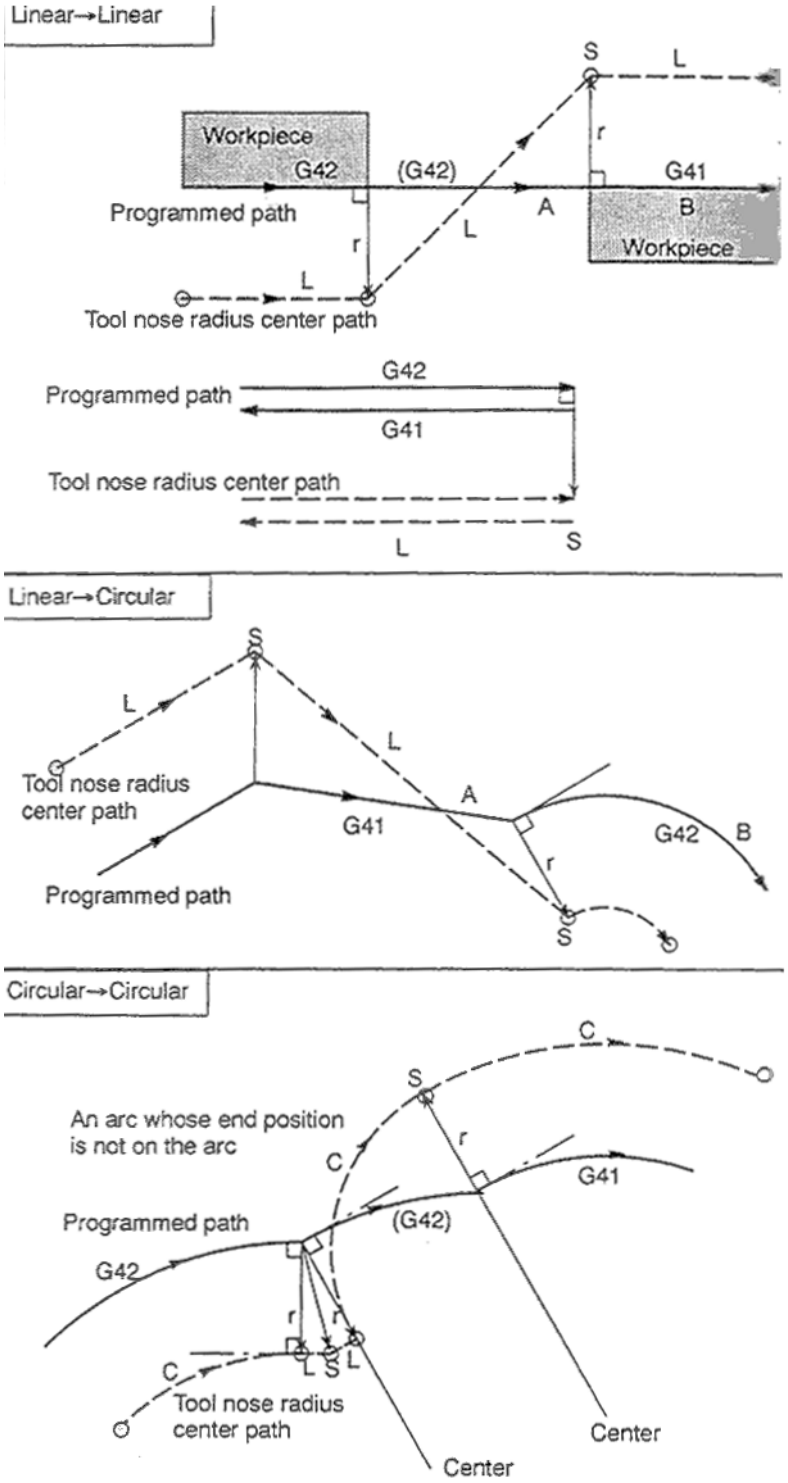
Траектория центра кромки инструмента при пересечении



Tool nose radius center path – Траектория центра кромки радиуса инструмента
Programmed path – Запрограммированная траектория

Траектория центра кромки инструмента без пересечения

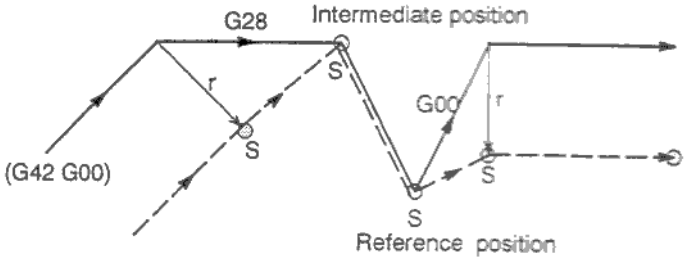
При изменении направления офсета в блоке A в блок B использующих режим G41 и G42, если не требуется пересечение с траекторией офсета, вектор нормали блока B создается в начальной позиции блока B.



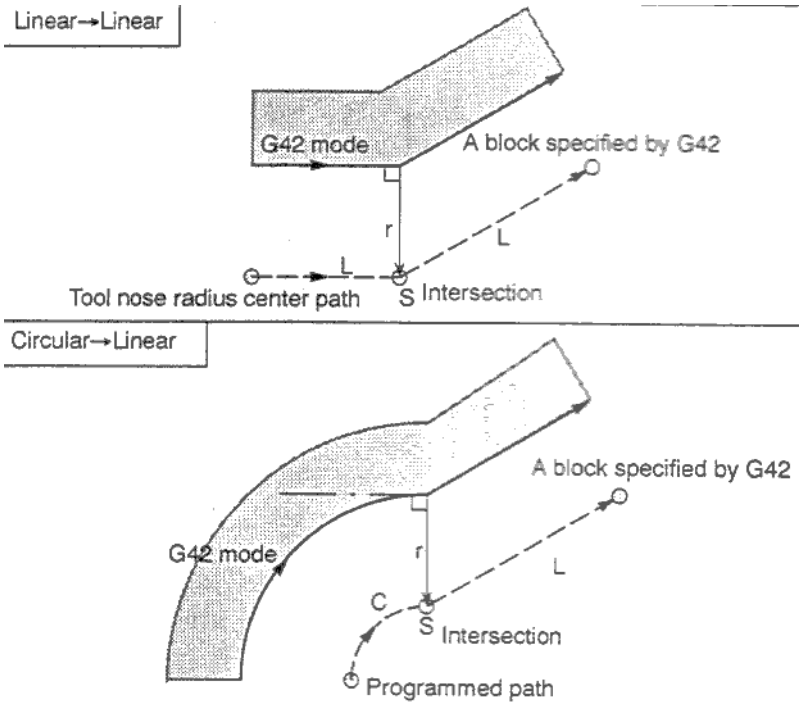
Tool nose radius center path – Траектория центра кромки радиуса инструмента
Programmed path – Запрограммированная траектория
An arc whose end position is not on the arc – Дуга чья конечная позиция не на дуге

Временная отмена компенсации радиуса кромки инструмента
Если следующая команда указана в режиме офсета, режим офсета временно отменяется и затем автоматически восстанавливается. Режим офсета может быть отменен и включен как описано в 2.14.3.2 и 2.14.3.4.

Указание G28 автоматический возврат в точку ссылки в режиме офсета
Если запрограммирован G28 в режиме офсета, режим офсета временно отменяется в промежуточной точке. Если вектор все еще остается после возврата в точку ссылки, компоненты вектора сбрасываются в ноль для каждой оси для которой был совершен переход в точку ссылки.



G код компенсации радиуса кромки инструмента в режиме офсета
Вектор офсета может быть установлен для создания прямого угла относительно перемещения предыдущего блока вне зависимости от внешнего или внутреннего угла, для этого укажите G41 или G42 в режиме офсета. Если этот код указан в круговом движении, правильное движение по окружности не будет получено.
Если направление офсета изменяется командой компенсации кромки инструмента G41, G42, см. 14.3.3.

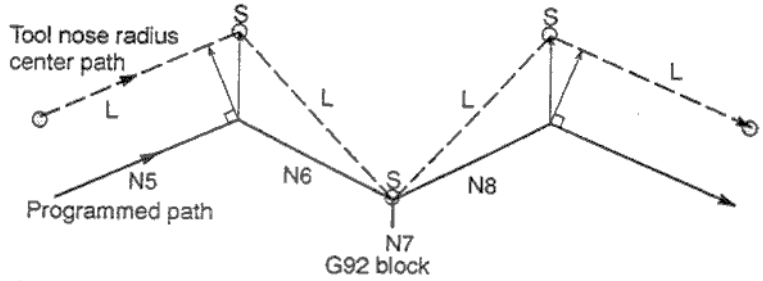


Команда временной отмены вектора офсета

В режиме офсета, если запрограммировать команду G50, вектор офсета будет временно отменен, и затем автоматически восстановлен.

В этом случае, офсет отменяется без движения, инструмента переходит прям в запрограммированную точку в которой совершена отмена офсета. Также, если офсет восстановлен, инструмент прямо переходит в точку пересечения.

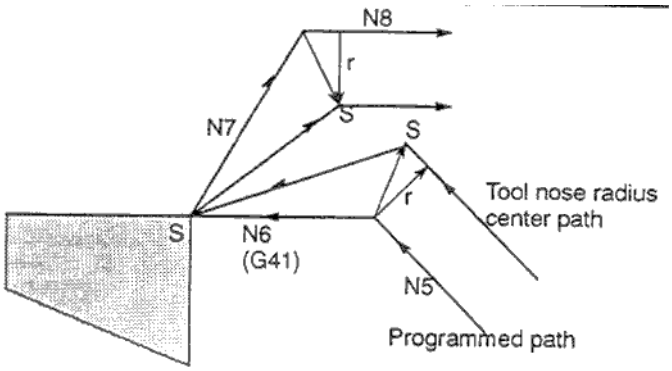
Установка системы координат детали G50




```
(G41)
N5 G91 G01 U700.0 W300.0 ;
N6 U600.0 W–300.0 ;
N7 G50 X200.0 Z100.0 ;
N8 G01 X800.0 Z400.0 ;
```

Фиксированные циклы G90, G92, G94 и многопроходные циклы G71 – G76

См. 2.14.1 и 2.14.2. для описания функции компенсации кромки инструмента в этих циклах.




```
(G42)
N5 G01 U500.0 W600.0 ;
N6 W–800.0 ;
N7 G90 U–600.0 Z–800.0 I–300.0 ;
N8 U1200.0 W500.0 ;
```

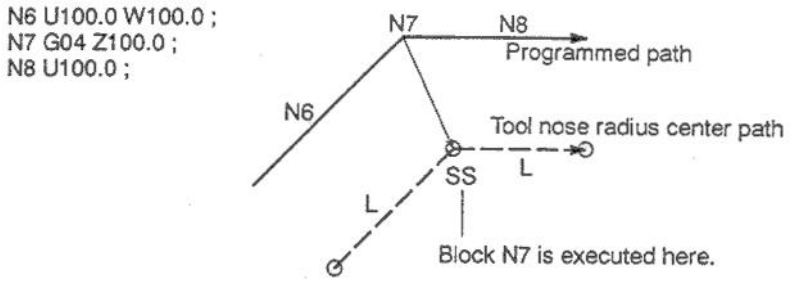
Блок без перемещения инструмента

Следующие блоки не содержат перемещения инструмента. В этих блоках инструмента не будет перемещаться даже если будет включена компенсация кромки инструмента.

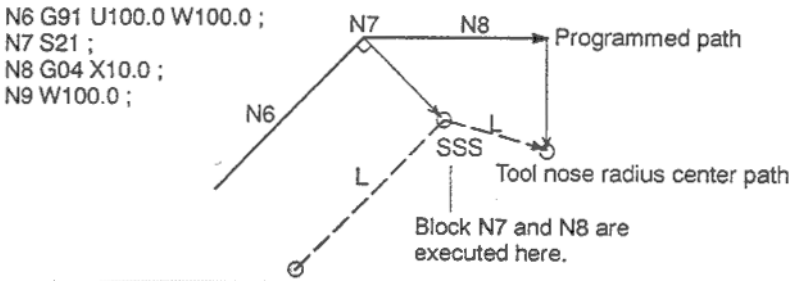
1. M05;	M код
2. S21;	S код
3. G04 X10.0;	Задержка
4. G10 P01 X10 Z20 R10.0;	Значение компенсации
5. (G17) Z200.0;	Команда перемещения не в текущей плоскости
6. G98;	Только G код
7. X0;	Нулевое перемещение

Блок без перемещения инструмента в режиме офсета

Если отдельный блок не содержит перемещения инструмента в режиме офсета, вектор и траектория центра кромки инструмента совпадают как и в блоке без команд. Блок исполняется одним шагом.



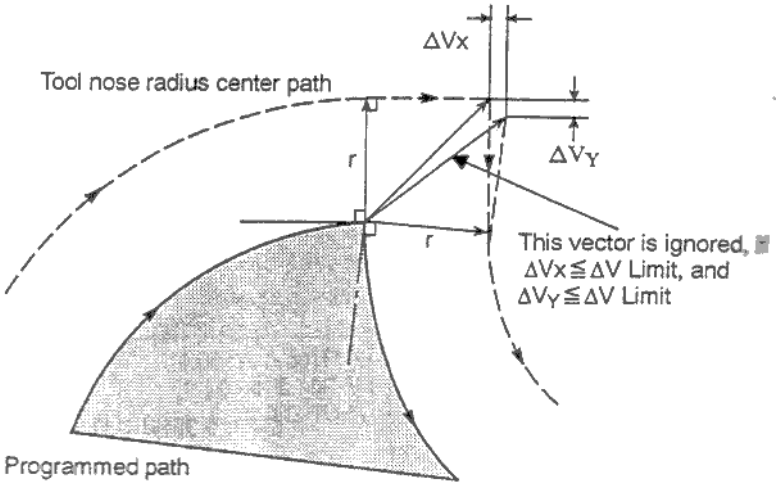
Однако, если задано нулевое расстояние перемещения, или если блок задан отдельно, движение инструмента такое же, как и в случае нескольких блоков перемещения заданных последовательно.



Движение у угла

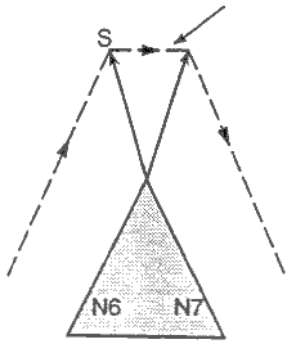
Если в конце блока генерируется два или более векторов, инструмент перемещается линейно от одного вектора к другому.

Если эти векторы полностью совпадают друг с другом, движение не производится и последний вектор игнорируется.



Если Δ Vx ≤ ΔVlimit и Δ Vy ≤ ΔVlimit, последний вектор игнорируется. Vlimit устанавливается параметром 50101.

Если векторы не совпадают, совершается переход от одного вектора к другому.



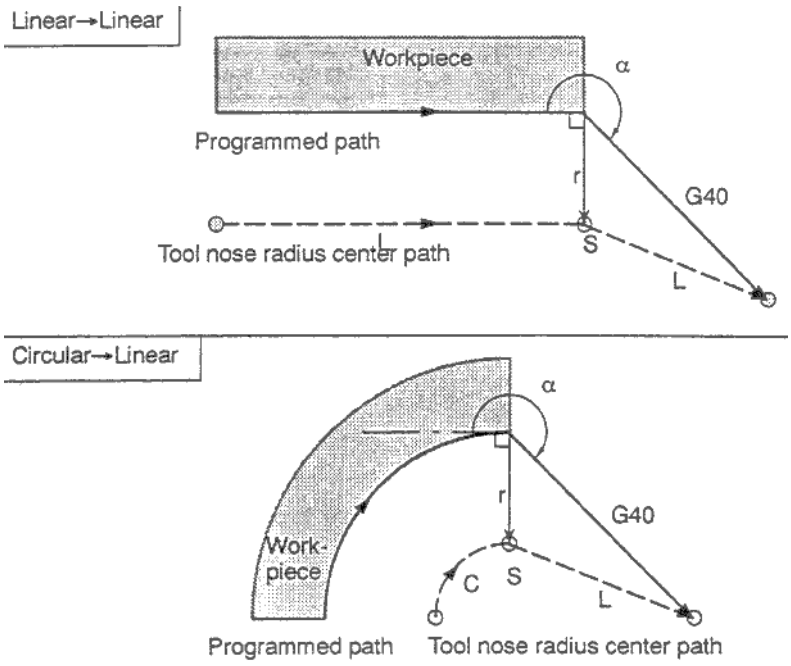
Движение перехода принадлежит блоку N7, скорость подачи как и в блоке N7. Если блок N7 в режиме G00, инструмент перемещается с быстрой подачей. Если G01, G02, G03, инструмента перемещается в режиме G01.

Прерывание ручного управления

Описание операций коррекции в режиме ручного управления см в 3.3.5.

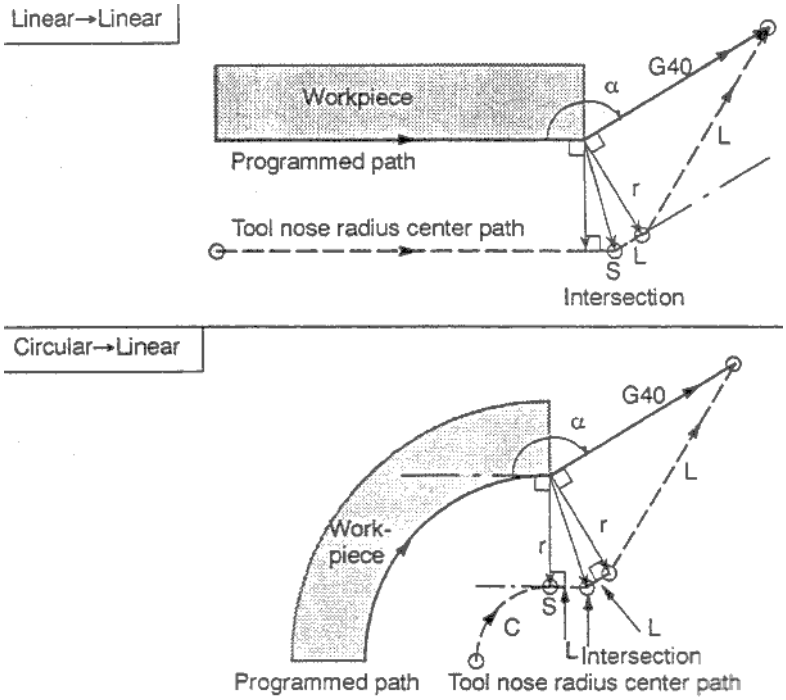
2.14.3.4 Перемещение инструмента в режиме отмены офсета

Перемещение инструмента по внутреннему контуру
(180° ≤ α)

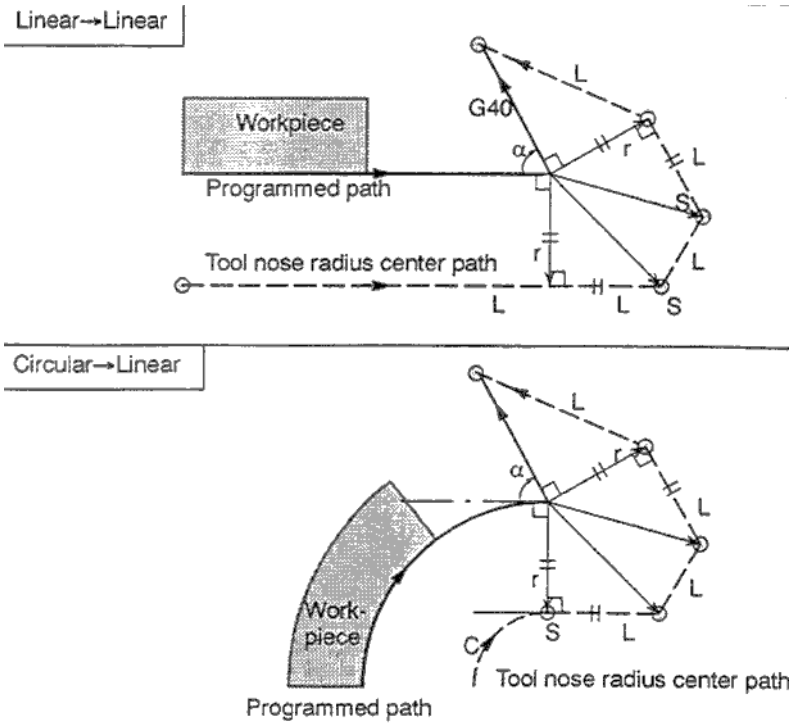


Programmed path – Запрограммированная траектория
Workpiece – Деталь
Tool nose radius center path – Траектория центра кромки радиуса инструмента
Intersection - Пересечение

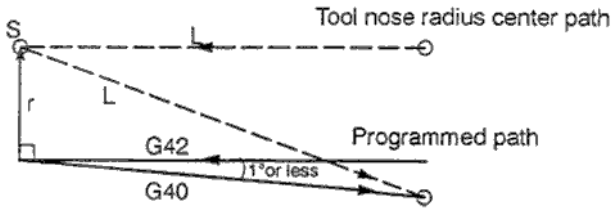
Перемещение инструмента по внешней стороне угла под тупым углом
(90° ≤ α ≤ 180°)



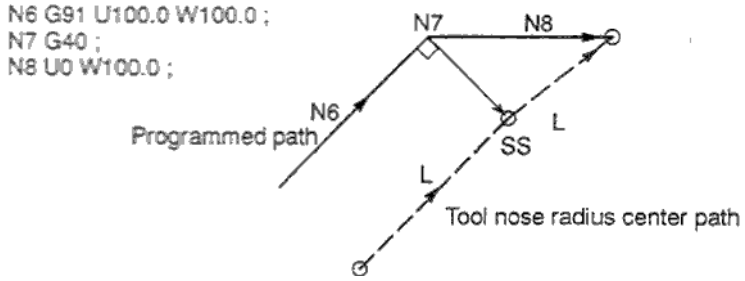
Перемещение инструмента по внешней стороне угла под острым углом
(α < 90°)



Перемещение инструмента из линии в линию с углом меньше 1 градуса
(α < 1°)



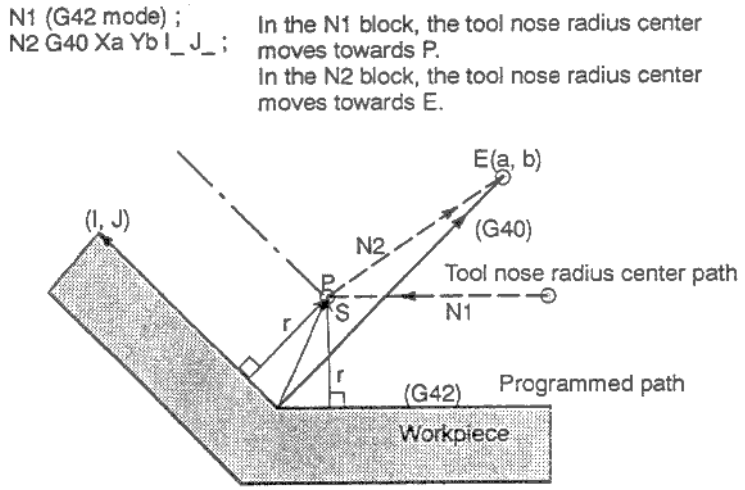
Блок без движения при отмене офсета
Если блок не содержащий движения инструмента запрограммирован с отменой офсета, вектор, чья длина равна значению офсета в направлении нормали не создает движения относительно предыдущего блока, вектор отменяется при следующей команде перемещения.



Блок содержащий G40, и I, J, K

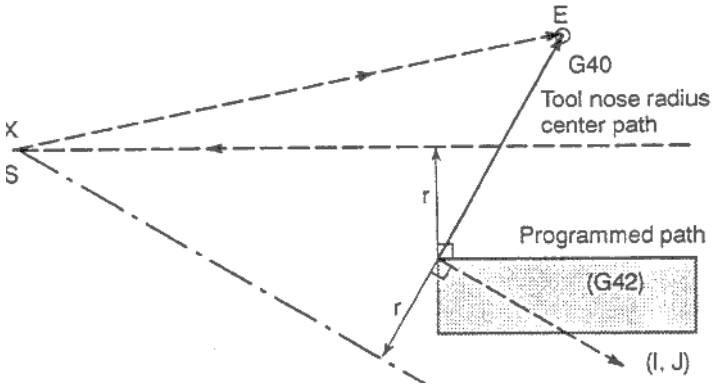
Предыдущий блок содержит G41 или G42

Если G41 или G42 находятся в предыдущем блоке относительно блока с G40 и I, J, K, система предполагает что траектория запрограммирована как путь от конечной точки определяемой начальным блоком в вектор определяемый (I, J0, (I, K), (J, K). Направление компенсации в начальном блоке наследуется.

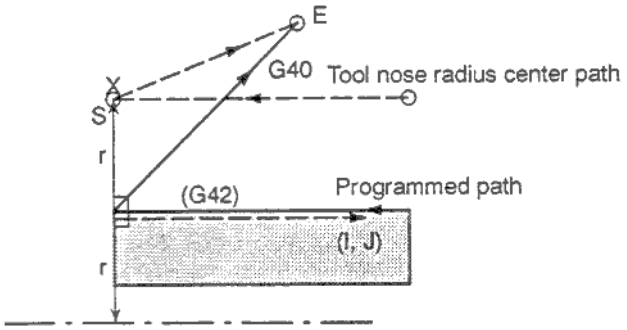


В блоке N1 центр кромки перемещается к P.
В блоке N2 инструмент перемещается к E.

В этом случае, помните, что ЧПУ получает точку пересечения пути инструмента независимо от внешней или внутренней стороны обрабатываемой поверхности.



Если точка пересечения не вычислена, инструмент перемещается в нормальную позицию предыдущего блока.



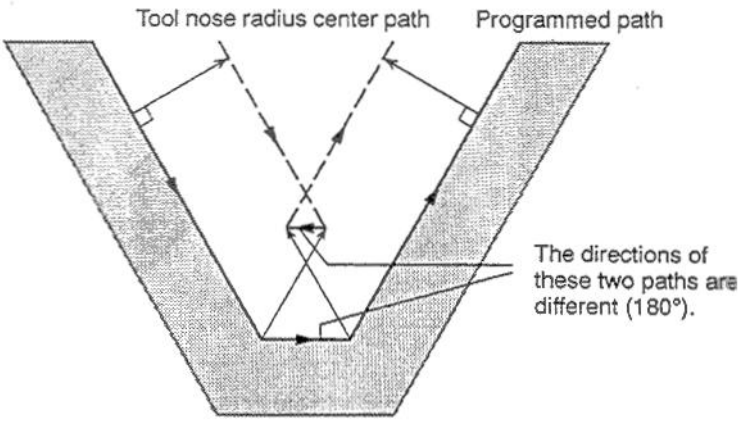
2.14.3.5 Проверка интерференции

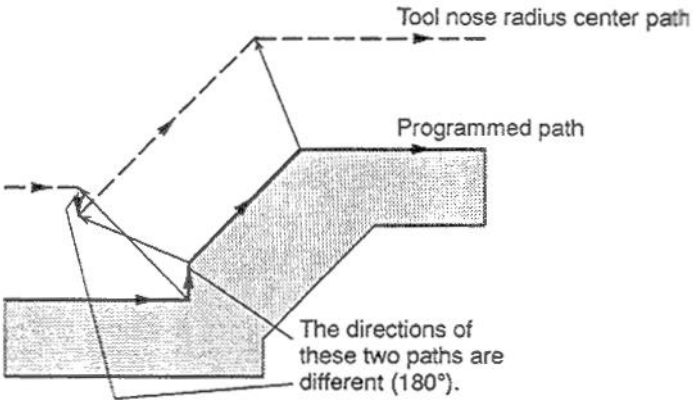
Наложение перемещения инструмента при лишней обработке называется интерференцией. Функция проверки интерференции (взаимовлияния) проверяет перемещение инструмента наперед. Интерференция используется даже если не производится наложение перемещения.

Пояснения

Критерий определения интерференции

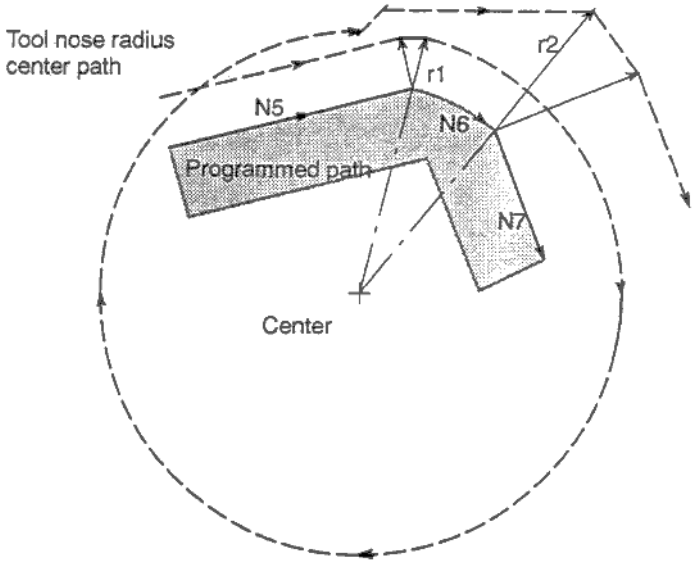
1) Направление радиуса кромки инструмента отличается от направления запрограммированного пути (от 90 до 270 градусов между блоками перемещения).





Tool nose radius center path – Траектория центра кромки радиуса инструмента
Programmed path – Запрограммированная траектория
Directions of two paths are different – Направление блоков отличается

2. В дополнение к условию 1, угол между начальной точкой и конечной у кромки инструмента отличается от начальной и конечной точки запрограммированной траектории при обработке окружности более чем на 180 градусов.




```
(G41)
N5 G01 U200.0 W800.0 T1 ;
N6 G02 U–160.0 W320.0 I–800.0 K–200.0 T2 ;
N7 G01 U–500.0 W200.0 ;
(Tool compensation value corresponding to T1 : r1 = 200.0)
(Tool compensation value corresponding to T2 : r2 = 600.0)
```

Tool nose radius center path – Траектория центра кромки радиуса инструмента
Programmed path – Запрограммированная траектория

Значение компенсации для T1: $r_1 = 200.0$
Значение компенсации для T2: $r_2 = 600.0$

В этом примере, дуга в блоке N6 находится в одном квадранте. Но после компенсации радиуса, дуга перемещается в четвертый квадрант.

Коррекция интерференции с упреждением

1) Удаление вектора создающего интерференцию

Если компенсация кромки инструмента выполняется для блоков A, B и C и векторы V1, V2, V3 и V4 находятся между блоками A и B, и V5, V6, V7 между блоками B и C, сначала проверяются ближайшие вектора. Если происходит интерференция, они игнорируются. Но если векторы должны быть проигнорированы из-за интерференции последнего вектора у угла, они не могут быть проигнорированы.

Проверка между векторами V4 и V5

Интерференция V4 и V5 игнорируется.

Проверка между векторами V3 и V6

Интерференция V3 и V6 игнорируется.

Проверка между векторами V2 и V7

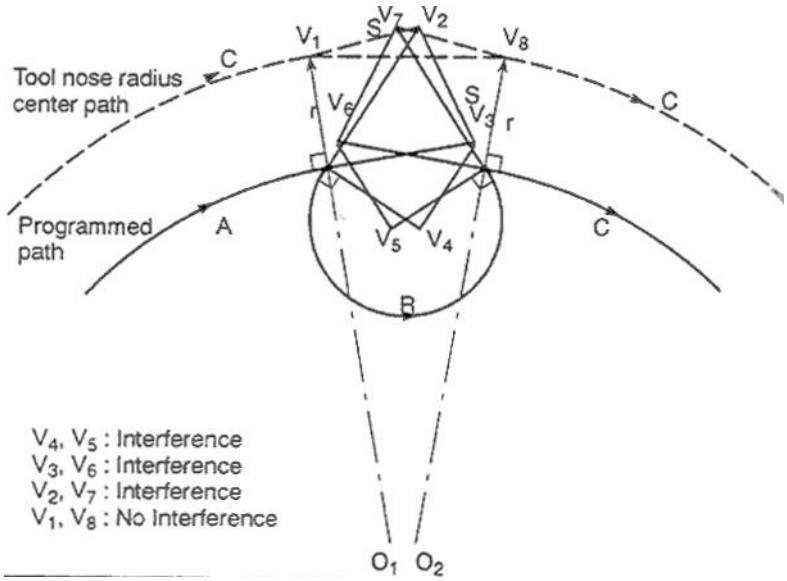
Интерференция V2 и V7 игнорируется.

Проверка между векторами V1 и V8

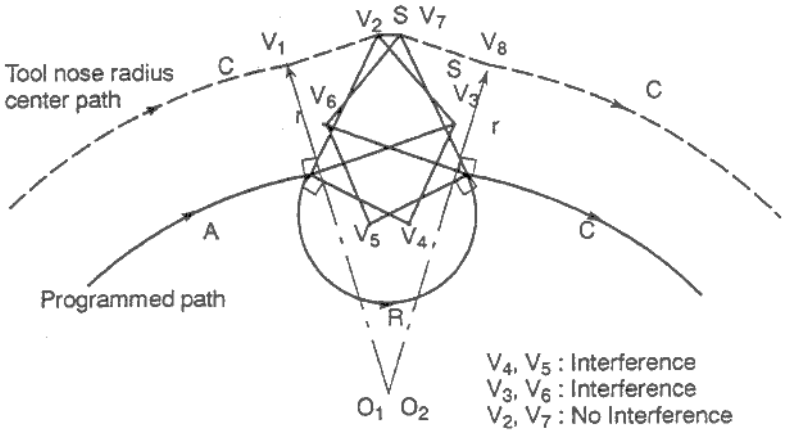
Интерференция V1 и V8 игнорируется.

Если во время проверки вектор без интерференции определен, последующие вектора не проверяются. Если блок B создает круговое движение, линейное движение создается при интерференции векторов.

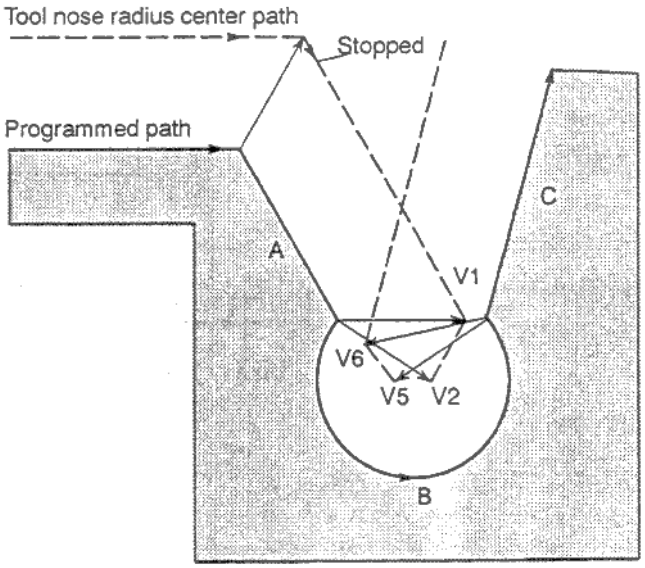
Пример 1. Инструмент линейно перемещается от V1 до V8



Пример 2. Инструмент линейно перемещается от V1, V2, V7 до V8



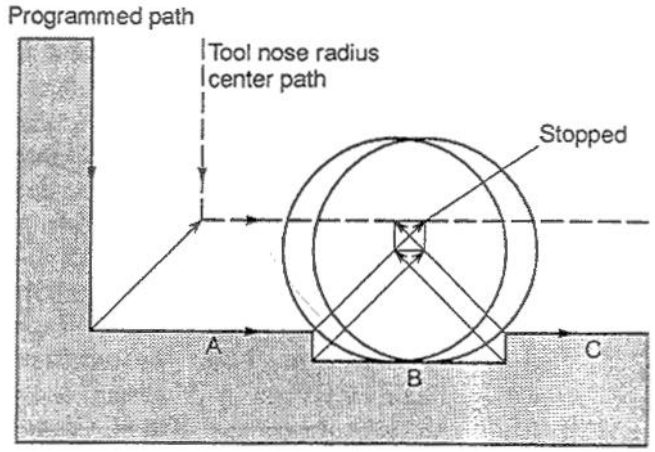
Если интерференция создается после коррекции 1, инструмент останавливается с сигналом тревоги.
Если интерференция создается после коррекции 1 или если существует только одна пара векторов с начала проверки и векторами с интерференцией, появится сигнал тревоги 41, инструмент остановится сразу после выполнения предыдущего блока. Если работа производится в поблочном режиме, инструмент останавливается в конце блока.



После игнорирования векторов V2 и V5 из-за интерференции, интерференция также создается между векторами V1 и V6. Сигнал тревоги появится, инструмент остановится.

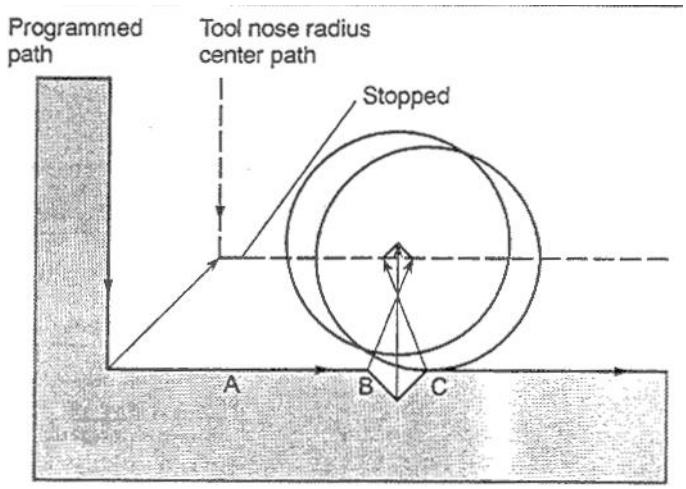
Определение интерференции тогда как действительной интерференции не существует

1) Впадина меньшая величины компенсации



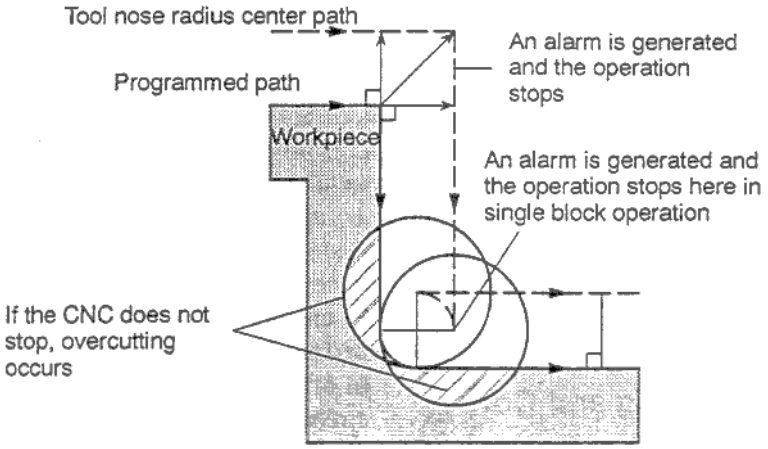
В данном случае не существует реальной интерференции, но запрограммированное направление в блоке B обратно направления после применения компенсации инструмента. Инструмент остановится, появится сигнал тревоги 041.

2) Канавка меньшая величины компенсации

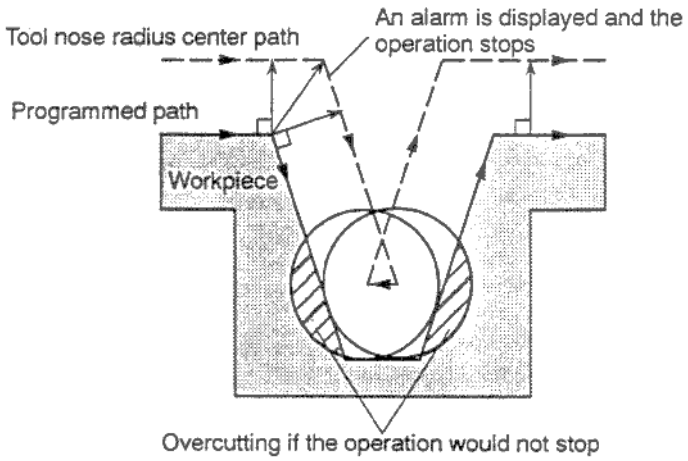


2.14.3.6 Излишняя обработка при компенсации радиуса кромки

Обработка внутреннего угла с радиусом меньше радиуса кромки инструмента
Если радиус угла меньше величины радиуса кромки инструмента, т.к. будет совершена излишняя обработка, появится сигнал тревоги и инструмент остановится в начале блока. В поблочном режиме будет совершена излишняя обработка т.к. инструмент останавливается только после выполнения блока.

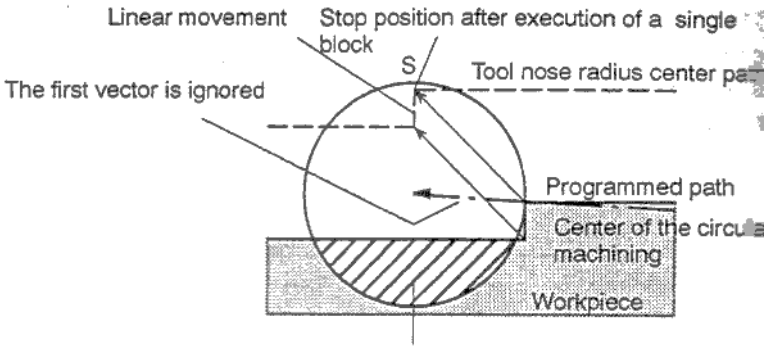


Обработка канавки меньшей радиуса кромки инструмента
Так как компенсация кромки инструмента перемещает центр инструмента в обратную сторону от запрограммированного направления, в результате будет излишняя обработка. В этом случае появится сигнал тревоги и инструмент остановится в начале блока.



Обработка шага меньшего коррекции кромки инструмента

При обработке кругового движения величиной меньше значения коррекции кромки инструмента, инструмент должен будет совершить движение в обратном направлении. В этом случае вектор игнорируется и инструмент перемещается в позицию второго вектора. В поблочном режиме операция останавливается в этой точке. Если обработка производится не в поблочном режиме, операция будет продолжена. Если шаг линейный, сигнал тревоги не появится и контур будет обработан правильно, однако останется необработанная часть.



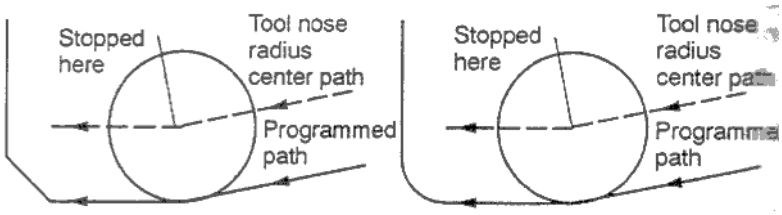
An overcutting will result if the first vector is not ignored.
However, tool moves linearly.

2.14.3.7 Коррекция при снятии фаски и закруглении

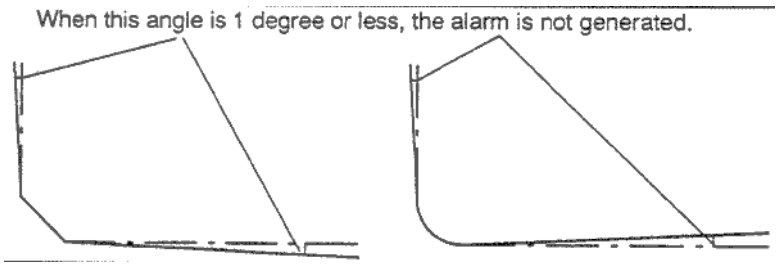
При снятии фаски или закруглении угла, компенсация радиуса кромки выполняется только при обычном пересечении ус углом.

В режиме отмены офсета, блоке включения офсета или переключения режима офсета, компенсация не производится, появится сигнал тревоги 39 и инструмент остановится.

При снятии фаски на внутренней поверхности или закруглении внутреннего угла, если значение фаски или закругления меньше значения коррекции радиуса, инструмент остановится с сигналом тревоги 39.

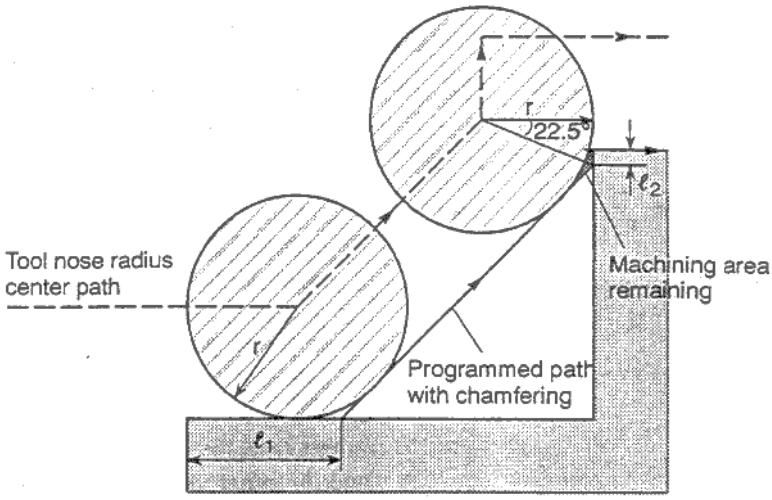


Допустимый угол врезания запрограммированного пути в предыдущем блоке и после блока угла является 1 градус или меньше, в противном случае появится сигнал тревоги 52, 54.



Если остается область обработки или появляется сигнал тревоги

На следующем примере показана область которая не может быть полностью обработана.

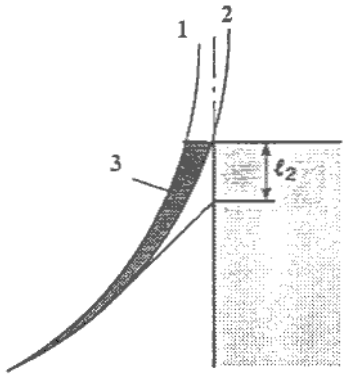


Programmed path with chamfering – Запрограммированный путь с фаской

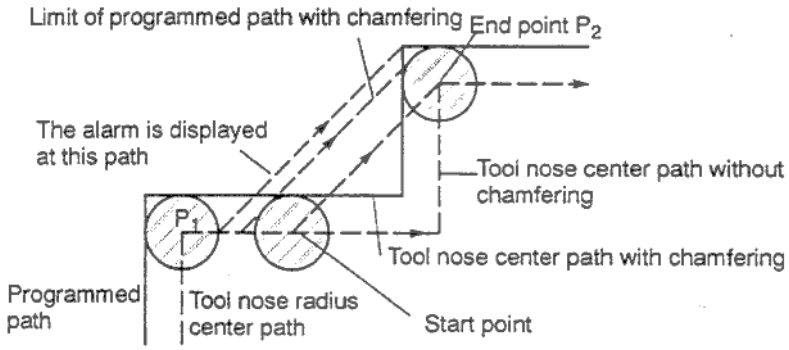
При внутренней фаске, если часть запрограммированной траектории срезается фаской, будет совершена неполная обработка.

$0 \le l_1$ или $l_2 < r * \tan 22.5°$ (r – радиус кромки инструмента)

Увеличенный вид необработанной части



Хотя инструмент должен быть позиционирован в 2, инструмент позиционируется в 1 (кромка инструмента по касательной к линии L).
Поэтому часть 3 не обрабатывается.
Сигнал тревоги 52 или 55 появится в следующих случаях:



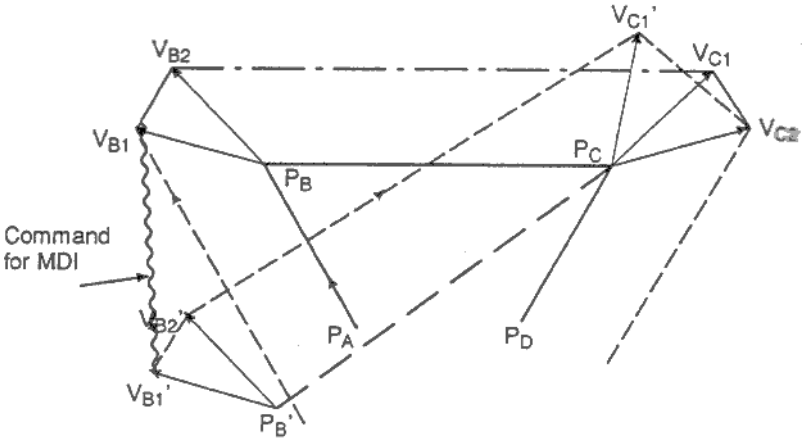
Limit of programmed path with chamfering – Предел запрограммированного пути с фаской
The alarm is displayed at this path – Сигнал тревоги появится здесь
Programmed path – Запрограммированная траектория
Tool nose radius center path – Траектория центра кромки радиуса инструмента
Tool nose radius center path with chamfering – Траектория центра кромки радиуса инструмента с фаской
Tool nose radius center path without chamfering – Траектория центра кромки радиуса инструмента без фаски

Если обрабатывается внешняя фаска с офсетом, используется ограничение запрограммированного пути. Путь при снятии фаски совпадает с точкой пересечения P1 или P2 без фаски, и поэтому внешняя фаска ограничивается. На предыдущем рисунке конечная точка фаски совпадает с точкой пересечения P2 следующего без снятии фаски. Если фаска больше значения ограничения, появится сигнал тревоги 52.

2.14.3.8 Ввод команд с MDI

Компенсация радиуса кромки инструмента не производится для команд вводимых с MDI. Тем не менее, если автоматическое управление производится с использованием абсолютных координат, и временно останавливается функцией поблочного режима, операции MDI выполняются, после чего автоматическое управление включается снова. В этом случае траектория инструмента будет следующей:

Вектор в начальной позиции следующего блока транслируется и другой вектор создается следующими двумя блоками. Поэтому, со следующего блока компенсация радиуса кромки выполняется корректно:



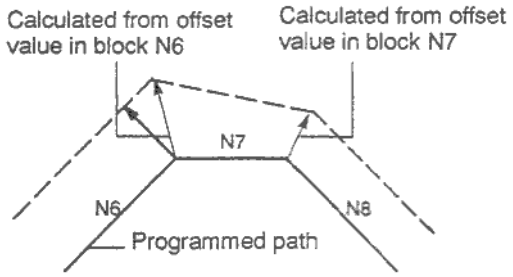
Если позиция PA, PB и PC запрограммированы в абсолютных координатах, инструмент останавливается в поблочном режиме после выполнения блока от Pa в PB и станок переключается в режим MDI. Векторы VB1 и VB2 транслируются в VB1’ и VB2’ и вектор офсета перевычисляется для векторов VC1 и VC2 между блоками PB-PC и PC-PD.

Однако, если вектор VB2 повторно не вычислен, компенсация корректно выполняется из позиции PC.

2.14.3.9 Общие предостережения при операциях с офсетом

Изменение значения офсета

Обычно, значение офсет изменяется при отключенном офсете или при смене инструмента. Если значение офсета изменяется в режиме офсета, вектор в точке конца блока вычисляется для нового значения офсета.



Calculated from offset value in block N6 – Вычислено из значения офсета блока N6

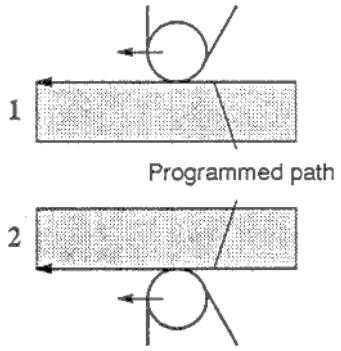
Если несколько векторов создается между блоками N6 и N7, вектор конечной точки текущего блока вычисляется с использованием значения офсета блока N6.

Полярность значения офсета и путь точки центра инструмента

Если указано отрицательное значение офсета, программа выполняется с заменой G41 на G42 и наоборот.

Инструмент для внутреннего профиля будет обрабатывать внешний профиль и наоборот.

Обычно, обработка с ЧПУ программируется предполагая положительное значение офсета. Если программа описывает путь как в случае 1 на следующем рисунке, при отрицательном значении офсета, инструмент перемещается как показано в 2.



ВНИМАНИЕ

Если знак офсета меняется на противоположный, вектор офсета меняет направление, но направление воображаемой кромки не изменяется.

Поэтому, не изменяется знак офсета во время обработки.

2.14.3.10 G53, G28, G30 и G30.1 команды при компенсации радиуса кромки

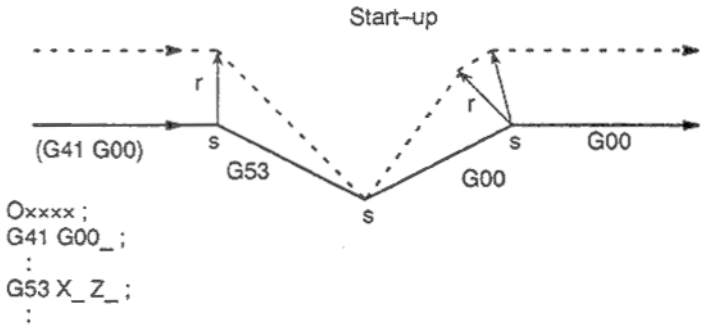
- Если команда G53 выполнена в режиме компенсации кромки инструмента, вектор компенсации автоматически отменяется перед позиционированием, после выполнения перемещения вектор автоматически будет восстановлен. Формат восстановления вектора FS16 если бит 2 (CCN) параметра 5003 установлен в 0, или FS15 если в 1.
- Если команда G28, G30 или G30.1 выполнена в режиме компенсации кромки инструмента, вектор компенсации автоматически отменяется перед позиционированием, после выполнения перемещения вектор автоматически будет восстановлен. Время и формат восстановления отмены и восстановления вектора FS16 если бит 2 (CCN) параметра 5003 установлен в 0, или FS15 если в 1.

Пояснения

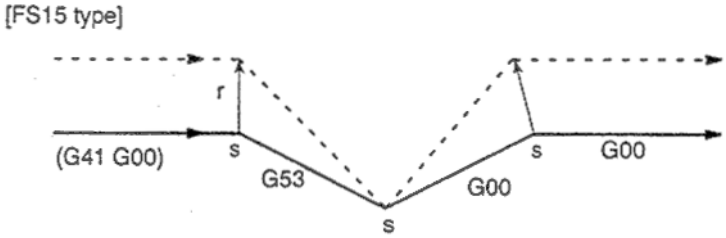
Команда G53 в режиме компенсации радиуса кромки инструмента
Если команда G53 выполняется в режиме компенсации радиуса кромки инструмента, в конце предыдущего блока создается вектор имеющий длину офсета, и расположен перпендикулярно направлению перемещения инструмента. Когда инструмент перемещается в указанную позицию в соответствии с командой G53, вектор офсета отменяется. Когда инструмент перемещается по следующей команде, вектор офсета автоматически восстанавливается.
Формат восстановления вектора определяется битом 2 (CCN) параметра 5003.

Команда G53 в режиме офсета

Если бит 2 (CCN) параметра 5003 установлен в 0.

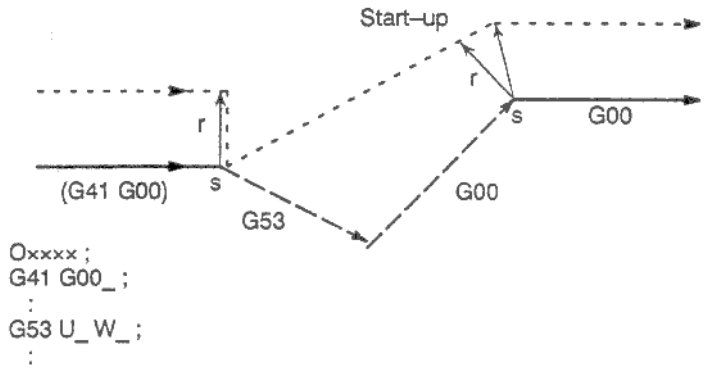


Если бит 2 (CCN) параметра 5003 установлен в 1.

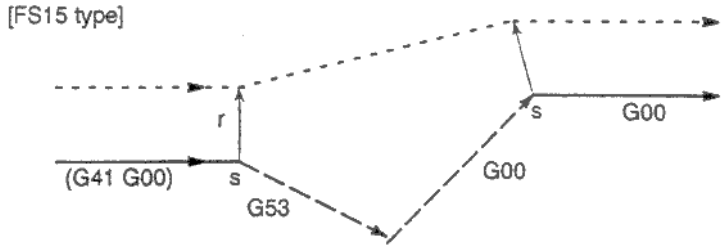


Относительная команда G53 в режиме офсета

Если бит 2 (CCN) параметра 5003 установлен в 0.

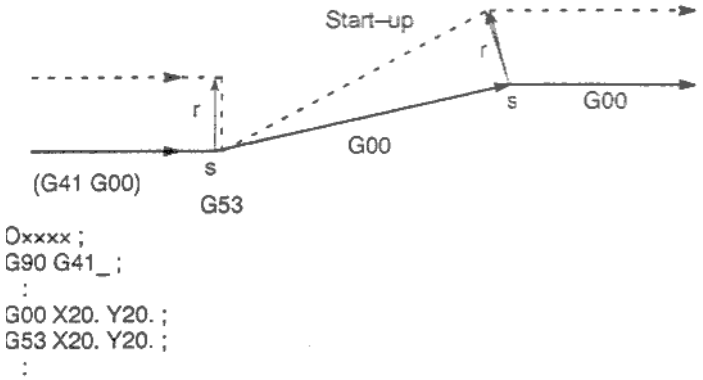


Если бит 2 (CCN) параметра 5003 установлен в 1.

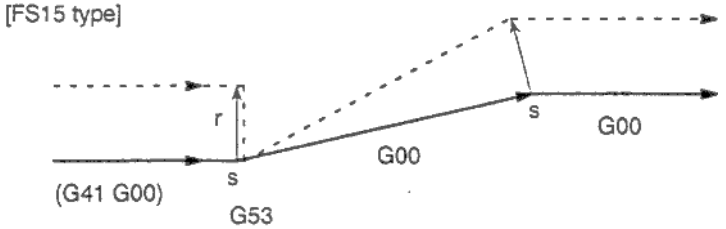


Команда G53 не описывающая движения в режиме офсета

Если бит 2 (CCN) параметра 5003 установлен в 0.



Если бит 2 (CCN) параметра 5003 установлен в 1.

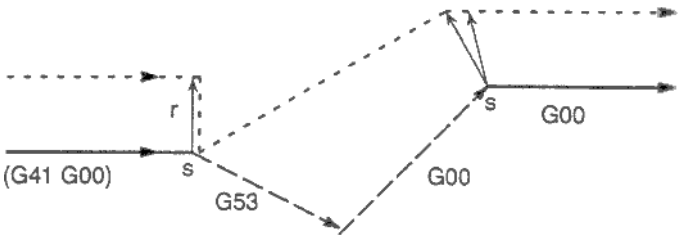


ВНИМАНИЕ

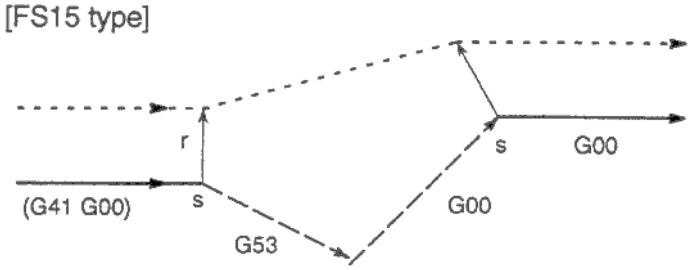
1 Если команда G53 выполняется в режиме компенсации радиуса кромки инструмента при блокировании всех осей станка, позиционирование не производится для тех осей, которые не заблокированы, вектор офсета не отменяется. Если бит 2 параметра 5003 установлен в 0 или включена функция блокировки каждой оси, вектор офсета отменяется.

Пример 1.

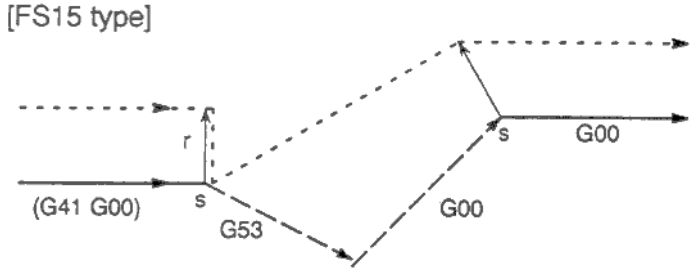
Если бит 2 (CCN) параметра 5003 установлен в 0 и включена функция блокировки всех осей



Пример 2.
Если бит 2 (CCN) параметра 5003 установлен в 1 и включена функция блокировки всех осей



Пример 3.
Если бит 2 (CCN) параметра 5003 установлен в 1 и включена функция блокировки каждой оси

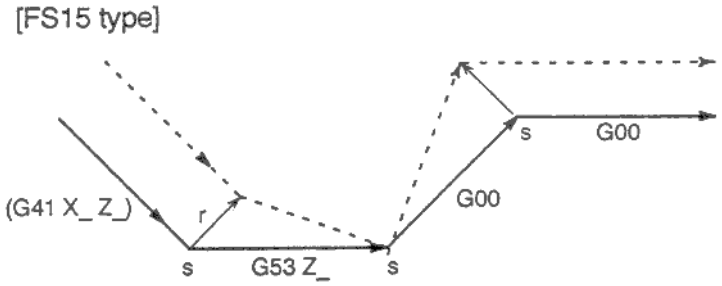


ВНИМАНИЕ

Если компенсационная ось указана в команде G53 в режиме компенсации, вектор для других компенсационных осей также отменяется. Эта также применяется если бит 2 параметра 5003 установлен в 1.

Пример

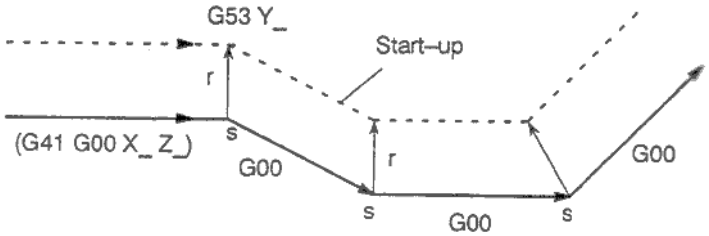
Если бит 2 (CCN) параметра 5003 установлен в 0



ПРИМЕЧАНИЕ

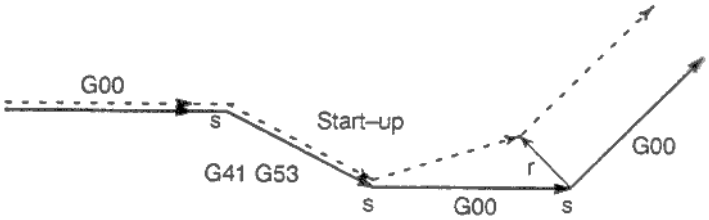
1. Если ось не включена в плоскость компенсации радиуса кромки инструмента в команде G53, вектор перпендикулярен направлению перемещения инструмента созданному в конце предыдущего блока и инструмент не перемещается. Режим офсета автоматически продолжается со следующего блока.

Пример
Если бит 2 (CCN) параметра 5003 установлен в 0



2. Если команда G53 указана в блоке включения офсета, следующий блок становится действительным блоком включения офсета. Если бит 2 параметра 5003 установлен в 1, следующий блок создает вектор пересечения.

Пример
Если бит 2 (CCN) параметра 5003 установлен в 0

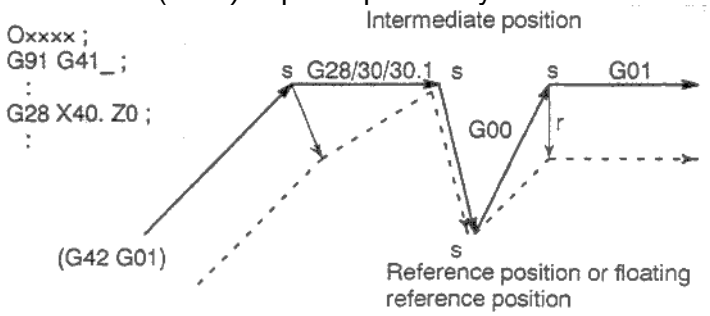


Команда G28, G30, G30.1 в режиме компенсации радиуса кромки

Если команда G28, G30, G30.1 в режиме компенсации, операция указанная в команде выполняется в соответствии с форматом FS15 если бит 2 параметра 5003 установлен в 1. Вектор пересечения создается в конце предыдущего блока и перпендикулярный вектор создается в промежуточной позиции. Вектор офсета отменяется если инструмент перемещается из промежуточной точки в точку ссылки. Вектор офсета восстанавливается в следующем блоке.

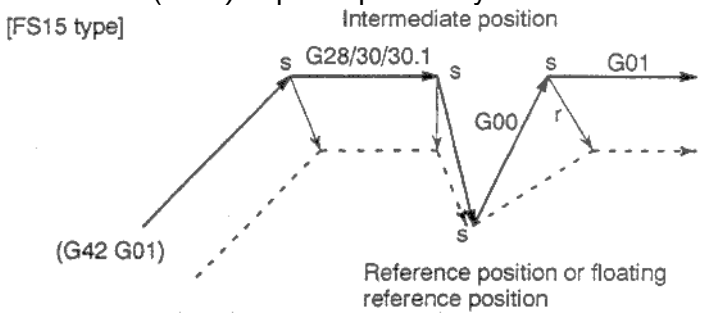
Команда G28, G30, G30.1 в режиме компенсации радиуса кромки (с движением в промежуточную точку и точку ссылки)

Если бит 2 (CCN) параметра 5003 установлен в 0.



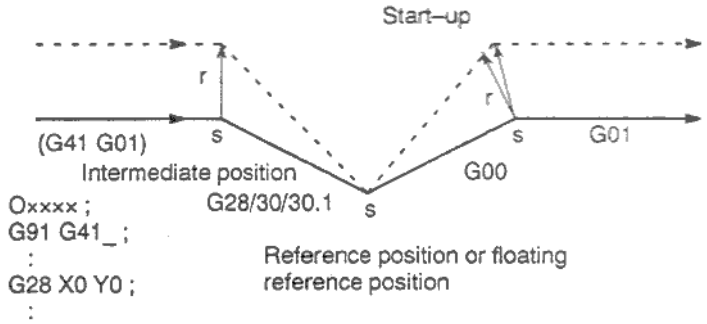
Intermediate position – Промежуточная точка
Reference position or floating reference position – Точка ссылки или плавающая точка ссылки

Если бит 2 (CCN) параметра 5003 установлен в 1.

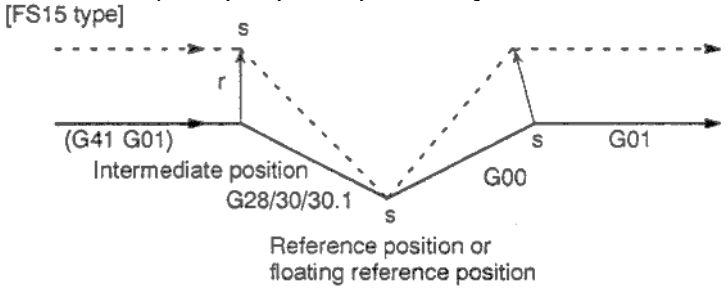


Команда G28, G30, G30.1 в режиме компенсации радиуса кромки (без движения в промежуточную точку)

Если бит 2 (CCN) параметра 5003 установлен в 0.

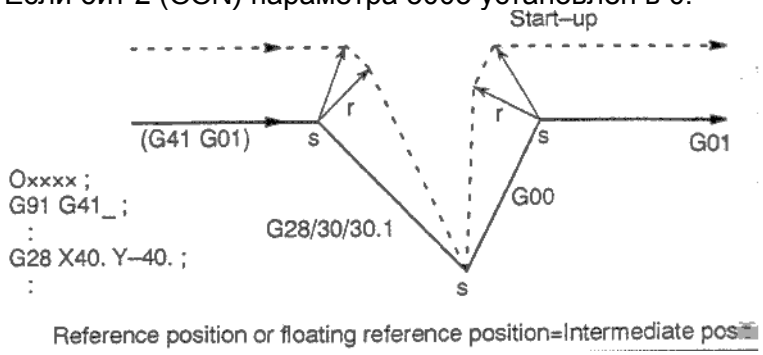


Если бит 2 (CCN) параметра 5003 установлен в 1.



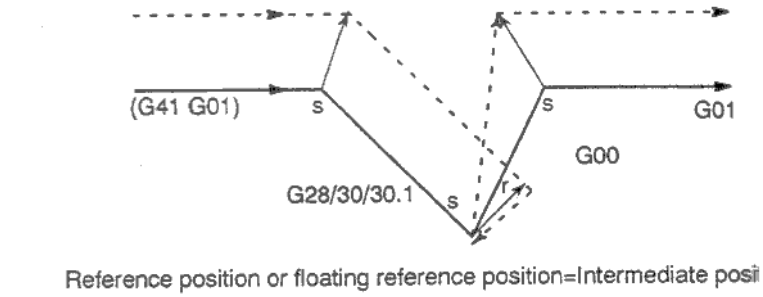
Команда G28, G30, G30.1 в режиме компенсации радиуса кромки (без движения в точку ссылки)

Если бит 2 (CCN) параметра 5003 установлен в 0.



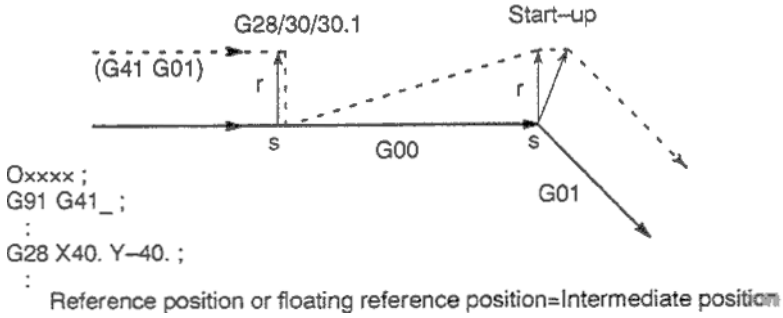
Если бит 2 (CCN) параметра 5003 установлен в 1.

[FS15 type]



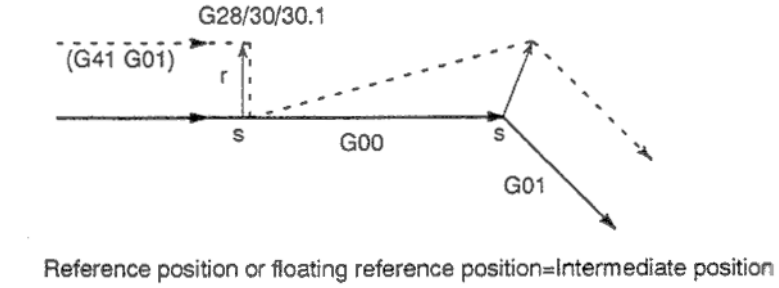
Команда G28, G30, G30.1 в режиме компенсации радиуса кромки (без перемещения)

Если бит 2 (CCN) параметра 5003 установлен в 0.



Если бит 2 (CCN) параметра 5003 установлен в 1.

[FS15 type]

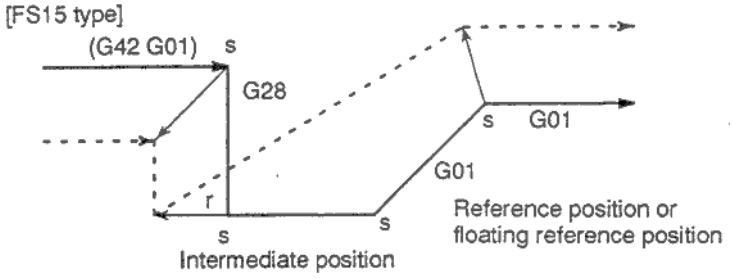


ВНИМАНИЕ

1 Если команда G28, G30 или G30.1 выполняется при включенной функции блокировки всех осей станка, вектор перпендикулярный направления перемещения инструмента создается в промежуточной позиции. В этом случае, инструмент не перемещается в точку ссылки и вектор офсета не отменяется. Если бит 2 параметра 5003 установлен в 0 или включена функция блокировки каждой оси, вектор офсета отменяется.

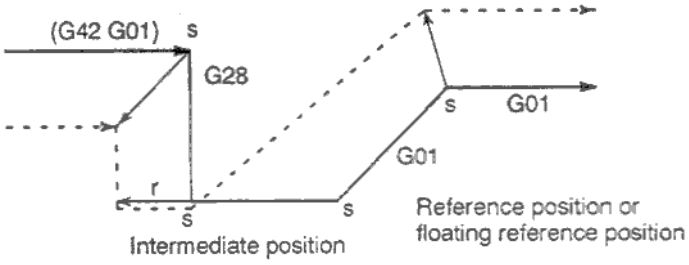
Пример 1.

Если бит 2 (CCN) параметра 5003 установлен в 0

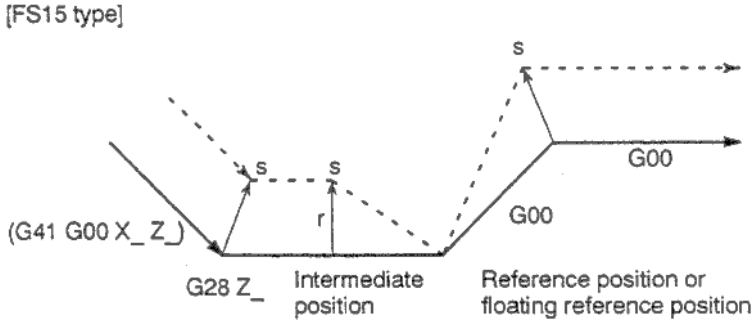


Пример 2.

Если бит 2 (CCN) параметра 5003 установлен в 0 и включена функция блокировки всех осей



2. Если компенсация оси указана в G28, G30 или G30.1 в режиме компенсации радиуса кромки, вектор для компенсации других осей также отменяется. Это также применяется если бит 2 (CCN) параметра 5003 установлен в 1.

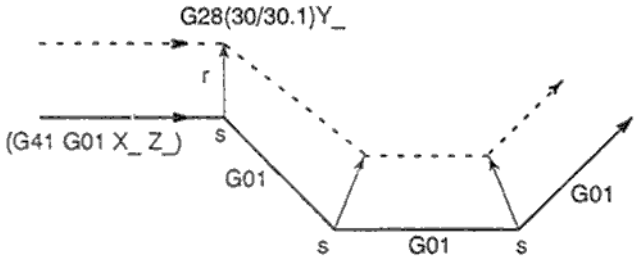


ПРИМЕЧАНИЕ

1. Если ось не включена в плоскость компенсации радиуса кромки инструмента в команде G28, G30, G30.1, вектор перпендикулярен направлению перемещения инструмента созданному в конце предыдущего блока и инструмент не перемещается. Режим офсета автоматически продолжается со следующего блока.

Пример

Если бит 2 (CCN) параметра 5003 установлен в 0

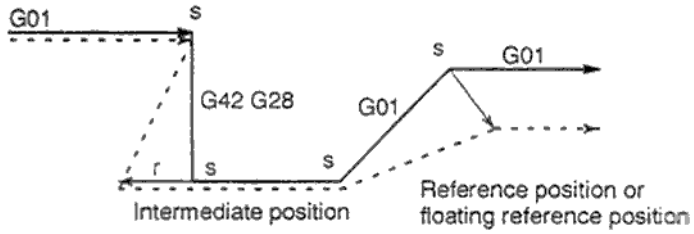


2. Если команда G28, G30, G30.1 вектор перпендикулярен направлению перемещения инструмента созданному в промежуточной точке. Вектор отменяется в точке ссылки. В следующем блоке создается вектор пересечения.

Пример

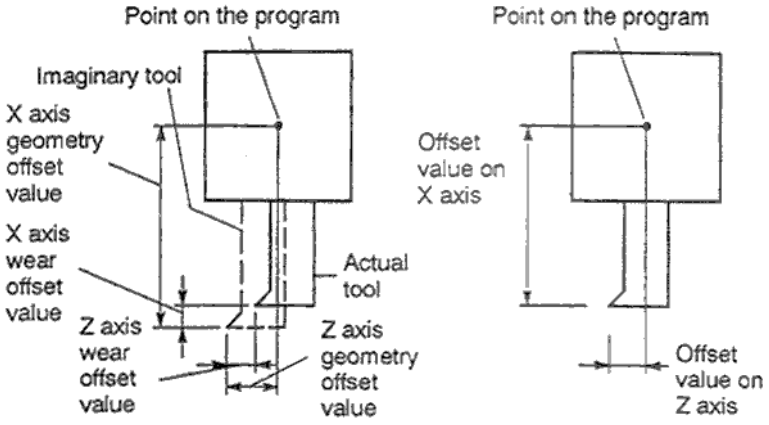
Если бит 2 (CCN) параметра 5003 установлен в 0

[FS15 type]



2.14.4 Значения компенсации инструмента, число значений компенсации и ввод значений из программы (G10)

Значения компенсации инструмента включают компенсацию геометрии и компенсацию износа. Компенсация инструмента может быть указана без дифференциации компенсации геометрии и компенсации износа.



Point of the program – Точка программы
Imaginary tool – Воображаемый инструмент
Рис. 14.4 Различия офсета геометрии и офсета износа

Значения компенсации инструмента могут быть введены в память ЧПУ с панели MDI или из программы. Значения компенсации выбираются из памяти ЧПУ когда соответствующий код указан после адреса T в программе.
Значение используется для офсета инструмента или компенсации радиуса кромки инструмента. См. 2.14.1.2.

2.14.4.1 Компенсация инструмента и номер компенсации

В таблице 14.4.1 показаны допустимые значения величин компенсации.

Таблица 14.4.1 Диапазон значений компенсации

Система приращений	Метрическая система	Дюймовая система
IS-B	0 - +-999.999мм	0 - +-99.9999дюйм
IS-C	0 - +-999.9999мм	0 - +-99.99999дюйм

Семизначная спецификация офсета инструмента

Число цифр определяющих офсет геометрии/износа инструмент может быть увеличено до 7. Если эта опция включена, в системе IS-B может использоваться 7 цифр и в IS-C восемь цифр. В этом случае диапазон значений компенсации будет следующим:

Таблица 14.4.1 (b) Диапазон значений компенсации

Система приращений	Метрическая система	Дюймовая система
IS-B	0 - +-9999.999мм	0 - +-999.9999дюйм
IS-C	0 - +-9999.9999мм (0 - +-4000.0000мм)	0 - +-999.99999дюйм (0 - +-160.00000дюйм)

ПРИМЕЧАНИЕ

1. Диапазон значений в скобках применяется при включенной функции автоматического преобразования дюймовой/метрической системы (бит 0 параметр 5006 установлен в 1).
2. Опция включающая семизначный офсет инструмент не может использоваться для определения офсета оси B.

Номера инструментов

В памяти могут находиться 16, 32, 64 или 99 инструментов.

ПРИМЕЧАНИЕ

При двухстороннем контроле, номер определяющий значения компенсации равен номеру компенсации инструмента для каждого посадочного места.

2.14.4.2 Изменение значения офсета
(Программирование данных G10)

Значение офсета может быть введено из программ с помощью следующей команды:

Формат
G10 P_ X_ Y_ Z_ R_ Q_;
или
G10 P_ U_ V_ W_ C_ Q_;
P: Номер офсета
0: Команда сдвига рабочей системы координат
1-64: Команда значения износа инструмента
10000+(1-64): Команда значения офсета геометрии инструмент
(1-64): Номер офсета
X: Величина офсета по оси X (абсолютное значение)
Y: Величина офсета по оси Y (абсолютное значение)
Z: Величина офсета по оси Z (абсолютное значение)
U: Величина офсета по оси X (относительное значение)
V: Величина офсета по оси Y (относительное значение)
W: Величина офсета по оси Z (относительное значение)
R: Значение офсета радиуса кромки инструмента (абсолютное значение)
C: Значение офсета радиуса кромки инструмента (относительное значение)
Q: Номер воображаемой кромки инструмента

При использовании абсолютных координат, значения адресов X, Y, Z, R устанавливают значения офсетов для соответствующего номера офсета указанного в адресе P. При использовании относительных координат, адреса U, V, W, C добавляются к текущему значению офсета.

ПРИМЕЧАНИЕ
1. Адреса X, Y, Z, U, V, W могут быть указаны в одном блоке.
2. Используйте эту команду в программе для внесения небольших изменений. Эта команда также может использоваться для ввода значений офсета с MDI один раз перед запуском программы.

2.14.5 Автоматический офсет инструмента (G36, G37)

Когда инструмент переходит в позицию измерения по команде ЧПУ, ЧПУ автоматически измеряет разницу между координатами текущей позиции и координатами запрограммированной и использует это значение офсета. Если ЧПУ решило что необходимо использовать измеренное значение, текущий офсет будет заменен.
См. документацию производителя станка для более полной информации.

Пояснения

Система координат
При перемещении инструмент в позицию измерения, система координат должна быть установлена заблаговременно.

Перемещение в позицию измерения
Перемещение в позицию измерения производится указанием следующей команды в режиме MDI или MEM:

G36Xx_a; или G37 Zz_a;

В этом случае, позиция измерения должна быть x_a или z_a. (абсолютная команда).
Выполнение этой команды перемещает инструмент с быстрой подачей в точку измерения с замедлением скорость подачи на полпути, и продолжая движения до получения сигнала от измерительного устройства. Когда кромка инструмента достигнет позиции измерения, система измерения подаст сигнал, ЧПУ остановит инструмента.

Офсет
Текущее значение офсета инструмента будет заменено на значение разницы между значениями координат (α или β) после достижения инструментом позиции измерения и значения Xa или Za указанного в G36Xx_a или G37 Zz_a.

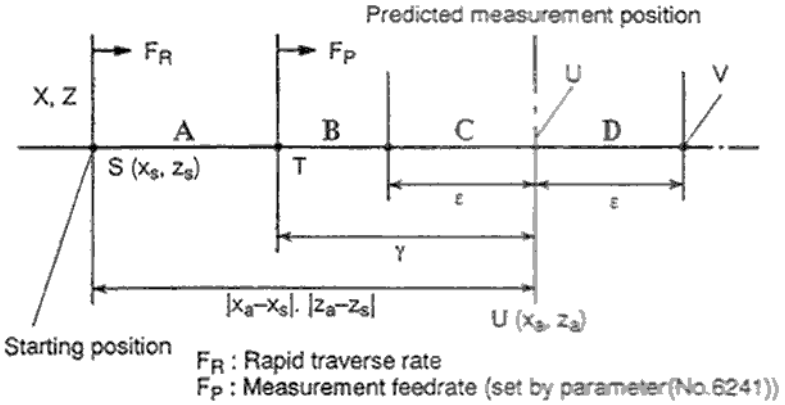
Значение офсета x = Текущее значение офсета x + ($\alpha - x_a$)
Значение офсета z = Текущее значение офсета x + ($\beta - z_a$)

x_a: Запрограммированная позиция измерения оси X
z_a: Запрограммированная позиция измерения оси Z

Эти значения офсета могут быть изменены с панели MDI.

Скорость подачи и сигнал тревоги

Когда инструмент перемещается из начальной точки в точку измерения, подача в области A является быстрой подачей. Затем инструмент останавливается в точке T и перемещается со скоростью измерения установленное параметром 6241 по области B, C и D. Если во время перемещения по области B появится сигнал завершения перемещения, появится сигнал тревоги. Если сигнал не появится до точки V, инструмент остановится в точке V и появится сигнал тревоги 080.



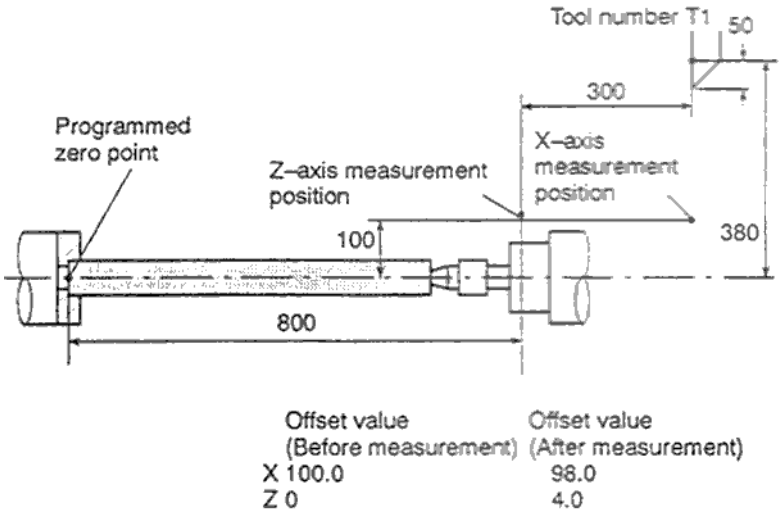
Predicted measurement position – Предполагаемая точка измерения
F_R: быстрая подача
F_P: подача измерения

Рис. 14.5 (a) Скорость подачи и сигнал тревоги

G код

Если бит 3 (G36) параметра 3405 установлен в 1, G37.1 и G37.2 используются G кодом для автоматической компенсации осей X и Z.

Пример

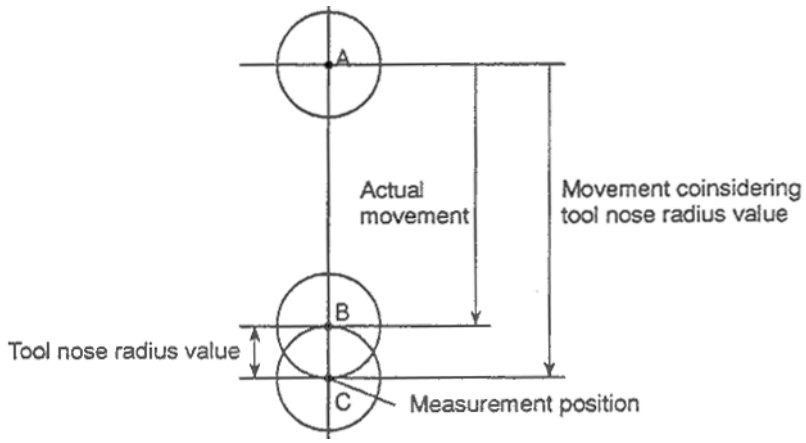


Programmed zero point – Запрограммированная нулевая точка
Measurement position – Позиция измерения
Offset value before measurement – Значение офсета до измерения
Offset value after measurement – Значение офсета после измерения

G50 X760.0 Z1100.0;	Программирование абсолютной точки нуля Установка системы координат
S01 M03 T0101;	Указание инструмента T1, номер офсета 1, вращение шпинделя
G36 X200.0;	Переход в позицию измерения Если инструмент достиг точки измерения в X198.0, тогда как точка измерения X200, значения офсета = -2.0 мм
G00 X204.0;	Переход по оси X
G37 Z800.0;	Перемещения в позицию измерения Z Если инструмент достиг точки измерения в Z804.0, тогда как точка измерения Z800, значения офсета = 4.0 мм
T0101;	Установка разницы офсета. Новое значение будет учтено при новом вызове T кода.

ВНИМАНИЕ

1. Скорость измерения Fp, γ и ε устанавливается параметрами 6241, 6251, 6254 производителем станка. ε должно быть положительным так чтобы $\gamma > \varepsilon$.
2. Перед G36, G37 необходимо отменить компенсацию радиуса кромки инструмента.
3. Если совершено ручное перемещение при движении измерения, переместите инструмент в точку в которой было совершено ручной перемещение.
4. При использовании функции опциональной компенсации радиуса кромки инструмента, величина офсета определяется сопоставлением значения радиуса кромки инструмента R. Убедитесь что значение радиуса кромки инструмента установлено правильно.
   Пример: Центр кромки инструмента совпадает с начальной точкой.



Tool nose radius value – Значение радиуса кромки инструмента
Measurement position – Позиция измерения
Actual value – Действительное перемещение
Movement coinsidering tool nose value – Перемещение совпадающее со значением радиуса кромки

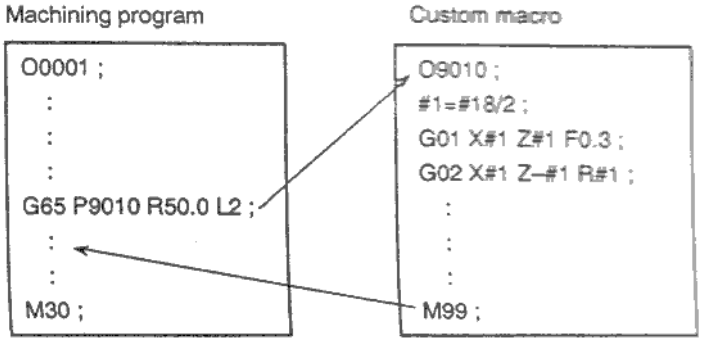
Инструмент в действительности перемещается из точки A в точку B, но значение офсета инструмента определяется исходя из предположения что инструмент перемещается в точку C на расстояние радиуса кромки инструмента.

ПРИМЕЧАНИЕ

1. Если перед G36 или G37 нет T кода, появится сигнал тревоги 81.
2. Если T коду указан в одном блоке с G36, G37, появится сигнал тревоги 82.

2.15 Произвольный макрос

Хотя подпрограммы полезны для повторов одинаковых операций, функция произвольного макроса также позволяет использовать переменные, арифметические и логические операции и условные переходы для легкого создания программ. Программы обработки могут вызвать произвольный макрос простой командой, так же как и подпрограмму.



2.15.1 Переменные

Простая программа обработки имеет G коды и команды перемещения задаются напрямую в числовом выражении, например G100 и X100.0.
С произвольным макросом числовые значения могут быть указаны непосредственно в числовом выражении или с использованием номера переменной. Если используется номер переменной, значение переменной может быть изменено в программе или с использованием операций на панели MDI.

```
#1=#2=100;

G01 X#1 F0.3;
```

Пояснения

Представление переменной
При определении переменной, укажите знак номера (#) и число после него. Языки программирования общего назначения позволяют назначать переменной имя, но это свойство невозможно для произвольного макроса.
Пример: #1

Выражение может использоваться для определения номера переменной. В этом случае, выражение должно быть заключено в скобки.
Пример: #[#1+#2-12]

Типы переменных
Переменные классифицируются четырьмя типами номером переменной.

Таблица 15.1 Типы переменных

Номер переменной	Тип переменной	Функция
#0	Всегда ноль	Эта переменная всегда равна нулю. Значения не могут быть присвоены.
#1 - #33	Локальная переменная	Локальные переменные могут использоваться только внутри макроса для хранения промежуточных результатов операций. После выключения питания, значения переменных устанавливаются в ноль. При вызове макроса, аргументы присваиваются локальным переменным.
#100 - #149 (#199) #500 - #531 (#999)	Общие переменные	Общие переменные могут использоваться совместно разными макро программами. После выключения питания, переменные #100 - #149 инициализируются с нулевым значением. Переменные 500 – 531 сохраняют свое значение. Опционально, переменные 150 – 199 и 532 – 999 также доступны. Однако, если их использовать, длина ленты для хранения данных уменьшается на 8.5м.
#1000 -	Системные переменные	Системные переменные используются для чтения и записи различных данных ЧПУ, таких как текущая позиция и величины компенсации.

ПРИМЕЧАНИЕ
Общие переменные #150 – #199 и #532 – #999 опциональны.

Диапазон значений переменных
Локальные и общие переменные могут иметь значение 0 или значение в следующем пределе:
-10^{47} до -10^{-29}
0
$+10^{-29}$ до $+10^{47}$
Если результат вычислений неправилен, появится сигнал тревоги 111.

Пропуск десятичной точки
Если значение переменной определяется в программе, десятичная точка может быт пропущена.

Пример:
Если #1=123, действительное значение переменной будет равно 123.00.

Ссылка переменной
Для ссылки на значение переменной в программе, укажите слово адреса после номера переменной. Если выражение используется для определения переменной, закройте выражение в скобки.
Пример: G01X[#1+#2]F#3;

Значение переменной ссылки автоматически округляется до наименьшего приращения адреса.

Пример:
Если G00X#1; выполнено на ЧПУ с шагом 1/1000мм со значением переменной #1 12.3456, действительной выполняемой командой будет G00X12.346;

Для изменения знака значения ссылки, поставьте знак минус (-) перед #.
Пример: G00X-#1;

Если производится ссылка на неопределенную переменную, переменная игнорируется.

Пример:
Если значение переменной #1 равно 0, и значение переменной #2 ссылается на переменную “null” (пусто), выполнение G00X#1Z#2 приведет к выполнению G00X0;

Неопределенные переменные
Если переменная не определена, эта переменная ссылается на “null”. Оно не может быть записано, но может быть прочитано.

(a) Цитирование
Если неопределенная переменная цитируется, весь адрес игнорируется.

Если #1 = <пусто>	Если #1 = 0
G90 X100 Y#1 будет означать G90 X100	G90 X100 Y#1 будет означать G90 X100 Y0

(b) Операция
<пусто> равнозначно 0, но заменяется на <пусто>

Если #1 = <пусто>	Если #1 = 0
#2 = #1 будет означать #2 = <пусто>	#2 = #1 будет означать #2 = 0
#2 = #1*5 будет означать #2 = 0	#2 = #1*5 будет означать #2 = 0
#2 = #1+#1 будет означать #2 = 0	#2 = #1+#1 будет означать #2 = 0

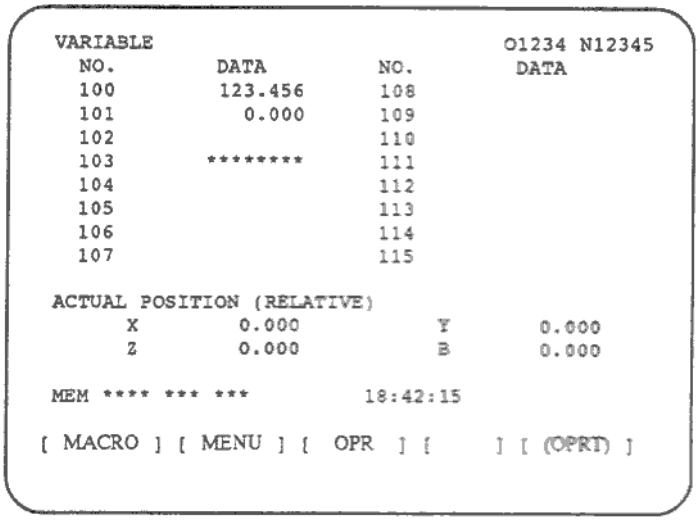
(c) Условные выражения
<пусто> отличается от 0 только для EQ и NE

Если #1 = <пусто>	Если #1 = 0
#1 EQ #0 будет означать Определено	#1 EQ #0 будет означать Не определено
#1 NE 0 будет означать Определено	#1 NE 0 будет означать Не определено
#1 GE #0 будет означать Определено	#1 GE #0 будет означать Определено
#1 GT 0 будет означать Не определено	#1 GT 0 будет означать Не определено

Произвольные макропеременные для позиции инструмента (двусторонний контроль)

При двухстороннем контроле макропеременные используются для каждой позиции инструмента. Указание параметра 6036 и 6037 позволяет некоторым переменным быть использованными для всех позиций инструмента.

Отображение значений переменных



Если значение переменной пусто, оно “null”.
Обозначение ******** обозначает переполнение, если абсолютное значение больше 99999999 или меньше 0.0000001.

Ограничения

Номера программ, номера блоков и номера опционально пропускаемых блоков не могут использовать значение ссылки.

Пример:

O#1;
/#2G00X100.0;
N#3Z200.0;

2.15.2 Системные переменные

Системные переменные могут использоваться для чтения и записи внутренних данных ЧПУ, таких как значения компенсации и данные текущей позиции. Помните, что некоторые системные переменные могут быть только прочитаны. Системные переменные нужны для автоматизации и написания программ общего назначения.

Пояснения

Сигналы интерфейса

Между различными программируемым контроллером станка (PMC) и произвольным макросом возможен обмен сигналами.

Таблица 15.2 (a) Системные переменные для сигналов интерфейса.

Номер переменной	Функция
#1000-#1015 #1035	16-битный сигнал может быть послан из PMC в произвольный макрос. Переменные 1000 – 1015 используются для побитного чтения. Переменная 1032 используется для чтения всех 16 бит сигнала за раз.
#1100-#1115 #1132	16-битный сигнал может быть послан из произвольного макроса в PMC. Переменные 1100 – 1115 используются для побитной передачи. Переменная 1132 используется для чтения всех 16 бит сигнала за раз.
#1133	Переменная 1133 используется для записи всех 32 битов сигнала за раз из произвольного макроса в PMC. Примечание, могут быть использованы значения от –99999999 до +99999999.

Подробное описания смотри в документации соединений B-63003EN-1.

Значения компенсации инструмента

Системные переменные не различают разницу между компенсацией геометрии и компенсацией износа инструмента, используйте эти переменные для компенсации износа.

Таблица 15.2 (b) Системные переменные компенсации инструмента памяти C

Номер компенс.	Значение компенс. оси X		Значение компенс. оси Z		Значение компенс. радиуса кромки		Позиция воображ. кромки	Значение компенс. оси Y	
	Износ	Геометр.	Износ	Геометр.	Износ	Геометр.		Износ	Геометр.
1 .. 49 .. 64	#2001 .. .. .. #2064	#2701 .. #2749	#2101 .. .. .. #2164	#2801 .. #2849	#2201 .. .. .. #2264	#2901 .. #2949	#2301 .. .. .. #2364	#2401 .. .#2449	#2451 .. #2499

Таблица 15.2 (c) Системные переменные для компенсации 99 инструментов

Номер компенс.	Значение компенс. оси X		Значение компенс. оси Z		Значение компенс. радиуса кромки		Позиция воображ. кромки	Значение компенс. оси Y	
	Износ	Геометр.	Износ	Геометр.	Износ	Геометр.		Износ	Геометр.
1 .. 99	#10001 .. #10099	#15001 .. #15099	#11001 .. #11099	#12001 .. #12099	#12001 .. #12099	#17001 .. #17099	#13001 .. #13099	#14001 .. #14099	#19001 .. #19099

ПРИМЕЧАНИЕ

Системные переменные #2001 - #2964 могут быть также использованы для определения износа по оси Y или компенсации геометрии оси Z со значениями 1 – 49 и другими значениями компенсации 1 – 64.

Макро сигналы тревоги

Таблица 15.2 (d) Системные переменные для макро сигналов тревоги

Номер переменной	Функция
#3000	Если значение от 0 до 200 назначается переменной #3000, ЧПУ останавливается с сигналом тревоги. После выражения, сообщение сигнала тревоги не может быть больше 26 символов. На экране монитора отображается сигнал тревоги с номер 3000 с добавлением значения переменной #3000.

Пример:
#3000=1(TOOL NOT FOUND);
→ На экране появится “3001 TOOL NOT FOUND.”

Информация времени
Эта информация может быть прочитана и записана.

Таблица 15.2 (e) Системные переменные для информации времени

Номер переменной	Функция
#3001	Эта переменная функции таймера с шагом 1 миллисекунда. После включения питания, значение переменной устанавливается в 0. При достижении 2147483648 миллисекунд, значение таймера сбрасывается в 0.
#3002	Эта переменная функции таймера с шагом 1 час. После включения питания, значение переменной устанавливается в 0. При достижении 9544.371767 часов, значение таймера сбрасывается в 0.
#3011	Эта переменная может быть использована для чтения текущей даты (год/месяц/день). Информация Год/месяц/день конвертируется в соответствующее числовое значение. Например, 28 марта, 1993 представлено как 19930328.
#3012	Эта переменная может быть использована для чтения текущего времени (час/минута/секунда). Информация час/минута/секунда конвертируется в соответствующее числовое значение. Например, 15 часов, 34 минуты и 56 секунда представлены как 153456.

Контроль автоматического управления
Состояние режима работы ЧПУ может быть изменено.

Таблица 15.2 (f) Системные переменные #3003 для управление автоматическим режимом

#3003	Поблочный режим	Выполнение вспомогательной функции
0	Включено	Ожидается
1	Выключено	Ожидается
2	Включено	Не ожидается
3	Выключено	Не ожидается

- После включения питания значение переменной равно 0.
- Если поблочный режим отключен, выполнение отдельного блока не выполняется даже в случае включения поблочного режима.
- Если ожидании выполнения вспомогательной функции (M, S и T функции) не производится, выполнение программы переходит к следующему блоку до выполнения вспомогательной функции. Также, сигнал DIN не подается на выход.

Таблица 15.2(g) Системные переменные #3004 для управление автоматическим режимом

#3004	Остановка подачи	Потенциометр подачи	Точная остановка
0	Включена	Включен	Выключена
1	Выключена	Включен	Выключена
2	Включена	Выключен	Выключена
3	Выключена	Выключен	Выключена
4	Включена	Включен	Включена
5	Выключена	Включен	Включена
6	Включена	Выключен	Включена
7	Выключена	Выключен	Включена

- После включения питания значение переменной равно 0.
- Если остановка подачи выключена:
  1) Если кнопка остановки подачи удерживается нажатой, станок останавливается в поблочном режиме. Однако, выполнение операции в поблочном режиме не производится если поблочный режим отменен переменной #3003.
  2) Если кнопка остановки подачи нажата и отпущена, лампа остановки подачи загорается, но станок не останавливается; выполнение программы продолжается и станок останавливается в первом блоке в котором разрежена остановка подачи.
- Если потенциометр подачи отключен, всегда используется значение 100%.
- Если отключена функция точной остановки, не производится проверка точной остановки даже в блоке содержащем операцию, которая не может быть выполнена без точной остановки.

Настройки

Настройки могут быть прочитаны и записаны. Двоичные числа конвертируются в десятичные.

#3005

	#15	#14	#13	#13	#11	#10	#9	#8
Настройка							FCV	
	#7	#6	#5	#4	#3	#2	#1	#0
Настройка			SEQ			INI	ISO	TVC

#9 FCV: Использование формата конвертации FS15
#5 SEQ: Автоматическая вставка номера последовательности
#2 INI: Миллиметровый или дюймовый ввод
#1 ISO: Использовать EIA или ISO код вывода
#0 TVC: Проверка TV

Зеркальное отображение

Статус зеркального отображения каждой оси используется внешним переключением или установка операции может быть прочитана через выходной сигнал. Наличие статуса зеркального отображения может быть проверено (см. 3.4.7).
Значение получаемое в двоичном коде конвертируется в десятеричное значение.

#3007

	#7	#6	#5	#4	#3	#2	#1	#0
Настройка	8 ось	7 ось	6 ось	5 ось	4 ось	3 ось	2 ось	1 ось

Для каждого бита при значении 0 функция отключена, при 1 включена.

Пример: Если #3007 установлено в 3, функция зеркального отображения включена для первой и второй оси.

- Если функция зеркального отображения установлена для определенной оси сигналом зеркального отображения и настройкой параметра, к значению сигнала и значению параметра будет применена функция OR (условное ИЛИ) и выдан сигнал.
- Если сигналы зеркального отображения для каждой оси отличаются от режима работы оси, они будут считаны в системную переменную #3007.
- Системная переменная #3007 является переменной защищенной от записи. Если будет совершена попытка записи, появится сигнал тревоги 116 “WRITE PROTECT VARIABLE».

Число изготовленных деталей
Число требуемых и изготовленных деталей может быть прочитано и записано.

Таблица 15.2 (h) Системные переменные числа требуемых и изготовленных деталей

Номер переменной	Функция
#3901	Число изготовленных деталей
#3902	Число требуемых деталей

ПРИМЕЧАНИЕ
Не указывайте отрицательное значение.

Модальная информация
Модальная информация в блоках может быть прочитана.

Таблица 15.2 (i) Системные переменные для модальной информации

Номер переменной	Функция	
#4001	G00, G01, G02, G03, G33, G34	Группа 01
#4002	G96, G97	Группа 02
#4003	..	Группа 03
#4004	G68, G69	Группа 04
#4005	G98, G99	Группа 05
#4006	G20, G21	Группа 06
#4007	G40, G41, G42	Группа 07
#4008	G25, G26	Группа 08
#4009	G22, G23	Группа 09
#4010	G80 – G89	Группа 10
#4011	..	Группа 11
#4012	G66, G67	Группа 12
#4014	G54-G59	Группа 13
#4015	..	Группа 14
#4016	G17 – G19	Группа 15
..		Группа 16
..		
#4022		Группа 22
#4109	F код	
#4113	M код	
#4114	Номер последовательности	
#4115	Номер программы	
#4119	S код	
#4120	T код	

Пример:
Если #1=#4001 будет выполнено, в результате в #1 будет 0, 1, 2, 3 или 33.

Если модальная информация считываемая из системной переменной соответствующей G коду группы которая не может быт определена, появится сигнал тревоги.

Текущая позиция
Информация текущей позиции не может быть записана, но может быть прочитана.

Таблица 15.2 (j) Системные переменные позиции информации

Номер переменной	Информация позиции	Система координат	Значение компенсации	Операция чтения при перемещении
#5001-#5008	Конечная точка блока	Детали	Не включено	Включена
#5021-#5028	Текущая позиция	Станка	Включено	Выключена
#5041-#5048	Текущая позиция	Детали		
#5061-#5068	Сигнал пропуска позиции			Включена
#5081, #5082	Значение офсета инструмента			Выключена
#5101-#5108	Позиция отклонения серво механизма			

- Первая цифра (от 1 до 8) представляет номер оси.
- Текущее используемое значение офсета инструмента находится в переменных #5081 - #5088
- Позиция инструмента в которой сигнал пропуска включается в G31 находится в переменных #5061 - #5068. Если сигнал пропуска не включен в блоке G31, конечная точка находится в этих переменных.
- Если чтение во время перемещения отключено, значения не будут прочитаны для функции буферизации.

Значения системы компенсации координат детали

Таблица 15.2 (к) Системные переменные офсета точки нуля детали

Номер переменной	Функция
5201 .. 5208	Значение офсета первой оси нуля детали .. Значение офсета восьмой оси нуля детали
5221 .. 5228	Значение офсета первой оси G54 .. Значение офсета восьмой оси G54
5241 .. 5248	Значение офсета первой оси G55 .. Значение офсета восьмой оси G55
5261 .. 5268	Значение офсета первой оси G56 .. Значение офсета восьмой оси G56
5281 .. 5288	Значение офсета первой оси G57 .. Значение офсета восьмой оси G57
5301 .. 5308	Значение офсета первой оси G58 .. Значение офсета восьмой оси G58
5321 .. 5328	Значение офсета первой оси G59 .. Значение офсета восьмой оси G59

ПРИМЕЧАНИЕ
Для использования переменных #5201 – 5328, необходима системная опция координат детали.

2.15.3 Арифметические и логические операции

С переменными могут быть произведены операции приведенные в таблице 15.3(a). Выражение справа от оператора может содержать константы и / или переменные с функцией или оператором. Переменные #j и #k в выражении могут быть использованы вместо констант. Переменные слева также могут быть заменены выражением.

Таблица 15.3(a) Арифметически и логические операции

Функция	Формат	Примечания
Определение	#i=#j	
Сумма Разность Произведение Частное	#i=#j + #k #i=#j - #k #i=#j * #k #i=#j / #k	
Синус Арксинус Косинус Арккосинус Тангенс Котангенс	#i = SIN[#k]; #i = ASIN[#k]; #i = COS[#k]; #i = ACOS[#k]; #i = TAN[#j]; #i = ATAN[#j]/[#k];	Угол указывается в градусах. 90 градусов и 30 минут обозначаются как 90.5 градусов.
Квадратный корень Абслютное значение Закругления нет Закругление вниз Закругление вверх Натуральный логарифм Экспонента	#i = SQRT[#k]; #i = ABS[#k]; #i = ROUND[#k]; #i = FIX[#k]; #i = FUP[#k]; #i = LN[#k]; #i = EXP[#k];	
ИЛИ Исключающее или И	#i=#j OR #k #i=#j XOR #k #i=#j AND #k	Логические операции выполняются над бинарными значениями побитно.
Конвертация из BCD в BIN Конвертация из BIN в BCD	#i=BIN[#k]; #i= BCD[#k];	Используется для обмена данными для PMC

Пояснения

Единицы угла

Единицы угла используются в SIN, COS, TAN, ASIN, ACOS, ATAN и указываются в градусах.

ARCSIN #i = ASIN[#j];

- Решение функции находится в следующих пределах:
  Если бит NAT (бит 0 параметра 6004) установлен в 0: 270 – 90 градусов
  Если бит NAT (бит 0 параметра 6004) установлен в 1: -90 – 90 градусов
- Если #j находится вне пределов -1 - +1, появится сигнал тревоги 111.
- Вместо переменной #j может использоваться константа.

ARCCOS #i = ACOS[#j];

- Решение функции находится в пределах 180 – 0 градусов:
- Если #j находится вне пределов -1 - +1, появится сигнал тревоги 111.
- Вместо переменной #j может использоваться константа.

ARCTAN #i = ATAN[#j]/[#k];

- Укажите длины сторон разделенные дробью /.
- Решение функции находится в следующих пределах:
  Если бит NAT (бит 0 параметра 6004) установлен в 0: 0 – 360 градусов
  Если бит NAT (бит 0 параметра 6004) установлен в 1: -180 – 180 градусов
- Вместо переменной #j может использоваться константа.

Натуральный логарифм #i = LN[#j];

- Помните, что погрешность составляет 10^{-8} или больше.
- Если антилогарифм равен 0 или меньше, появится сигнал тревоги 111.
- Вместо переменной #j может использоваться константа.

Экспонента #i = EXP[#j];

- Помните, что погрешность составляет 10^{-8} или больше.
- Если в результате операции значение превысит 3.65 x 10^{47}, появится сигнал тревоги 111.
- Вместо переменной #j может использоваться константа.

Функция ROUND

- Если функция ROUND включена в арифметическую или логическую операцию, с IF, или WHILE, функция округляется сразу после десятичной точки.
  Пример:
  Если выполнено #1=ROUND[#2] и #2 содержит 1.2345, значение #1 будет равно 1.0.

- Если функция ROUND используется в указании адреса ЧПУ, функция округляет указанное значение в соответствии с наименьшим инкрементом адреса.

Округление вверх и вниз

Если при округлении числа до целого значения получается большее число, это называется округлением вверх. Если при округлении числа до целого значения получается меньшее число, это называется округлением вниз. Будьте осторожны при округлении отрицательных чисел.

Пример:
#1 = 1.2 и #2=-1.2
#3=FUP[#1], результат 2.0
#3=FIX[#1], результат 1.0
#3=FUP[#2], результат -2.0
#3=FIX[#2], результат -1.0

Аббревиатура команд логических и арифметических операций

В программе может использоваться только два символа команды. (см. 3.9.7).

Пример:
ROUND → RO
FOX → FI

Приоритет операций

1. Функции
2. Операции такие как умножение и деление (*, /, AND, MOD)
3. Операции такие как сложение и вычитание (+, -, OR, XOR)

Пример: #1=#2+#3*SIN[#4];
Будет выполнено в следующем порядке:
SIN[#4];
#3*
#2+

Вложенность скобок
Скобки используются для изменения порядка выполнения операций. Скобки могут иметь уровень вложенности не более 5. При превышении вложенности, появится сигнал тревоги 118.

Ограничения

Скобки
Скобки ([,]) используются для закрытия выражения. Помните, что круглые скобки используются для выделения комментария.

Погрешности операции
При выполнении операции может возникнуть погрешность.

Таблица 15.3 (b) Погрешности при выполнении операции

Операция	Средняя погрешность	Максимальная погрешность	Тип погрешности
a=b*c	1.55×10^{-10}	4.66×10^{-10}	Относительная ошибка *1
a=b/c	4.66×10^{-10}	1.88×10^{-9}	
a= √b	1.24×10^{-9}	3.73×10^{-9}	
a=b+c a=b-c	2.33×10^{-10}	5.32×10^{-10}	Произведение *2
a=SIN[b] a=COS[b]	5.0×10^{-9}	1.0×10^{-8}	Абсолютная ошибка *3
a=ATAN[b]/[c] (*4)	1.8×10^{-6}	3.6×10^{-6}	

ПРИМЕЧАНИЕ
1. Относительная ошибка зависит от результата операции
2. Используется наименьшее значение.
3. Абсолютная ошибка постоянна.
4. Функция тангенса выполняется как SIN/COS.

Пример
Если совершается попытка назначить следующие значения переменным #1 и #2:
#1 = 9876543210123.456
#2 = 9876543277777.777
значения переменных будут следующими:
#1=9876543200000.000
#1=9876543200000.000
В этом случае, если #3=#2-#1 вычисляется #3=100000.000.

- Также, опасайтесь погрешности в следующих функциях: EQ, NE, GE, GT, LE, LT.

Пример
IF [#1 EQ #2] приводит к погрешности как в #1, так и в #2, что может привести к неправильному решению условия. Поэтому, вместо определения разницы между двумя переменными, используйте IF[ABS[#1-#2]LT0.001].
После этого предположите что две переменные равны если разница не превышается допустимого предела (в этом случае 0.001).

- Также, будьте осторожны при округлении вниз.

Пример
Если #2=#1*1000; вычисляется при #1=0.002, в результате значение переменной не точно 2.0, а 1.99999997. Необходимо использовать округление.

Делитель
Если в делителе указывается 0 или тангенс 90 градусов, появится сигнал тревоги 112.

2.15.4 Макро определения и определения ЧПУ

Следующие блоки являются макроопределениями:

- Блоки содержащие арифметические и логические операции (=)
- Блоки содержащие вызовы управляющих элементов (таких как GOTO, DO, END)
- Блоки содержащие команды вызова макроса.

Любые другие блоки являются блоками ЧПУ.

Пояснения

Отличия от определений ЧПУ

- Даже при включенном поблочном режиме, станок не останавливается. Помните, что станок останавливается если бит 5 параметра 6000 равен 1.
- Блоки макроса не являются блоками создающими перемещение в режиме компенсации радиуса кромки инструмента (см. 2.15.7).

Определения ЧПУ имеющие те же свойства, что и макро определения

- Если блок содержит команду вызова подпрограммы, и не содержит других адресов таких как O, N, P, L, этот блок эквивалентен макро определению.
- Если блок содержит M99 и не содержит других адресов таких как O, N, P, L, этот блок эквивалентен макро определению.

2.15.5 Переходы и повторы

В программе возможен контроль и изменение выполнения программы с использованием IF и GOTO. Используются три типа ветвления:

GOTO: безусловный переход
IF: переход по условию
WHILE: повторение

2.15.5.1 Безусловный переход (GOTO)

Совершается переход в блок номер n. Если указан номер блока вне пределов от 1 до 99999, появится сигнал тревоги 128. Номер блока может быть указан в виде выражения.

GOTO n;
GOTO#10;

2.15.5.2 Переход по условию (IF)

После IF укажите условие в случае удовлетворения которого, будет выполнена операция. Если условие не выполняется, команда не будет выполнена.
IF [#1 GT 10] GOTO 2; Если #! больше 10, переход в блок 2.

IF[<условие>]THEN
Если выполняется условие, выполняется указанное макро определение. Может быть выполнено только одно макро определение.
IF [#1 EQ #2] THEN #3=0; Если #1 = #2 то #3 приравнять к нулю.

Пояснения

Выражение условия
Выражение условия должно быть заключено в квадратные скобки. Возможно использование знака неравенства и выражения.

Таблица 15.5.2 Операторы

Оператор	Значение
EQ	Равно
NE	Не равно
GT	Больше
GE	Больше или равно
LT	Меньше
LE	Меньше или равно

Пример программы

```
O9500
#1=0;                    Инициализации переменной
#2=1;
N1 IF[#2 GT 10] GOTO 2;  Если #2 больше 10, переход в блок N2
10
#1=#1+#2;                Вычисление суммы
#2=#2+1;                 Сложение
GOTO 1;                  Переход в N1
N2 M30;                  Конец программы
```

2.15.5.3 Повторение (WHILE)

После WHILE необходимо указать определение условия при выполнении которого, будет выполняться программа от DO до END. Если условие не выполняется, будет произведено выполнение блока после END.

```
WHILE [условие] DO m;
        ..
        ..
        END m;
..
```

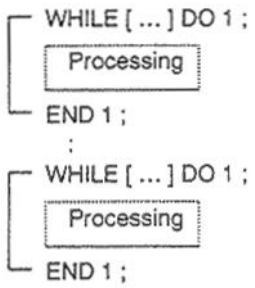
Пояснения

Если указанное условие удовлетворено, программа от DO до END будет выполняться. Если условие перестало быть истинным, произойдет переход в блок после END. Тот же формат используется для IF. Число после DO и число после END является идентификационным номером для определения границ выполнения. Могут быть использованы числа 1, 2 и 3. Если используются числа отличные от 1, 2, и 3, появится сигнал тревоги 126.

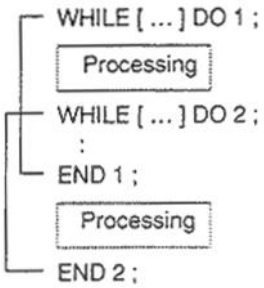
Вложенность

Идентификационные номер 1 – 3 в DO – END могут использоваться столько раз, сколько необходимо. Помните, что если программа содержит перекрестную вложенность, появится сигнал тревоги 124.

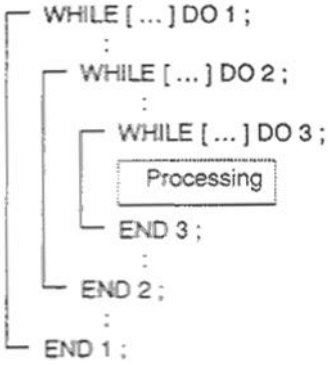
1. Идентификационные номер могут использоваться любое число раз.



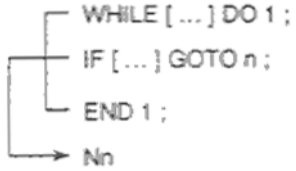
2. Пределы действия DO не могут пересекаться



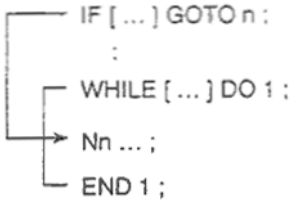
3. Вложенность может достигать трех уровней



4. Управление может быть передано из цикла



5. Ветвление не может производиться в цикл



Ограничения

Бесконечный цикл

Если указано DO m без указания WHILE, производится бесконечный цикл от DO до END.

Время обработки

Если указано ветвление (переход) в блок указанный в GOTO, производится поиск блока. Поиск в обратном направлении занимает больше времени, чем поиск вперед. Использование WHILE экономит время.

Неопределенные переменные

В условном переходе с использованием EQ или NE, значение null или нулевое значение имеют разный эффект. Во всех других условных выражениях, значение null считается равным нулю.

Пример

```
O0001;
#1=0;
#2=1;
WHILE[#2 LE 10] DO
1;
#1=#1+#2;
#2=#2+1;
END 1;
M30;
```

2.15.6 Вызов макроса

Программа макроса может быть вызвана следующими способами:
Простой вызов – G65
Модальный вызов – G67
Вызов макроса G кодом
Вызов макроса M кодом
Вызов подпрограммы M кодом
Вызов подпрограммы T кодом

Ограничения

Вызов макроса G65 отличается от вызова подпрограммы M98 следующим образом:

- В G65 возможно указание аргументов. M98 не имеет такой возможности.
- Если M98 блок содержит другие команды ЧПУ, подпрограмма вызывается после выполнения этих команд. G65 делает безусловный вызов макроса.
- Если M98 блок содержит другие команды ЧПУ, подпрограмма останавливается в поблочном режиме. G65 не останавливает станок.
- В G65 изменяется уровень локальных переменных. В M98 уровень локальных переменных не изменяется.

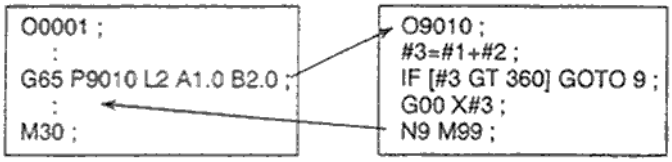
2.15.6.1 Простой вызов (G65)

Если запрограммирован G65, вызывается произвольный макрос указанный в адресе P. Данные (аргументы) могут быть пропущены.

Формат

G65 P p L l <передаваемые-аргументы>;

P	Номер вызываемой программы
l	Число повторов (1 по умолчанию)
Аргумент	Данные передаваемые макросу



Пояснения

Вызов

- После G65 укажите номер адреса программы с произвольным макросом.
- Если требуется повторить макрос несколько раз, в адресе L укажите требуемое число повторов от 1 до 9999. Если L пропущено, считается l = 1.
- Если указать передаваемые аргументы, значения будут присвоены соответствующим локальным переменным.

Передаваемые аргументы

Существует два типа передаваемых аргументов. Аргумент типа 1 использует буквы отличные от G, L, O, N, и P один раз каждую. Аргумент типа 2 использует A, B, C один раз каждую, а также I, J, K до десяти раз. Тип передаваемого аргумента определяется автоматически в соответствии с используемой буквой.

Спецификация аргумента 1

Адрес	Номер переменной	Адрес	Номер переменной	Адрес	Номер переменной
A	#1	I	#4	T	#20
B	#2	J	#5	U	#21
C	#3	K	#6	V	#22
D	#7	M	#13	W	#23
E	#8	Q	#17	X	#24
F	#9	R	#18	Y	#25
H	#11	S	#19	Z	#26

- Адреса G, L, N, O, P не могут использоваться в аргументах.
- Адреса которые не нужны, могут быть пропущены. Локальные переменные соответствующие пропущенным адресам устанавливаются в null.

Спецификация аргумента 2
Спецификация аргумента 2 использует A, B и C один раз и I, J, K до десяти раз. Спецификация аргумента 2 используется для передачи трехмерных координат.

Адрес	Номер переменной	Адрес	Номер переменной	Адрес	Номер переменной
A	#1	K_3	#12	J_7	#23
B	#2	I_4	#13	K_7	#24
C	#3	J_4	#14	I_8	#25
I_1	#4	K_4	#15	J_8	#26
J_1	#5	I_5	#16	K_8	#27
K_1	#6	J_5	#17	I_9	#28
I_2	#7	K_5	#18	J_9	#29
J_2	#8	I_6	#19	K_9	#30
K_2	#9	J_6	#19	I_{10}	#31
I_3	#10	K_6	#20	J_{10}	#32
J_3	#11	I_7	#22	K_{10}	#33

- Индексы I, J, K обозначают порядок указания аргументов и не записываются в действительной программе.

Ограничения

Формат
G56 должен быть указан до аргументов.

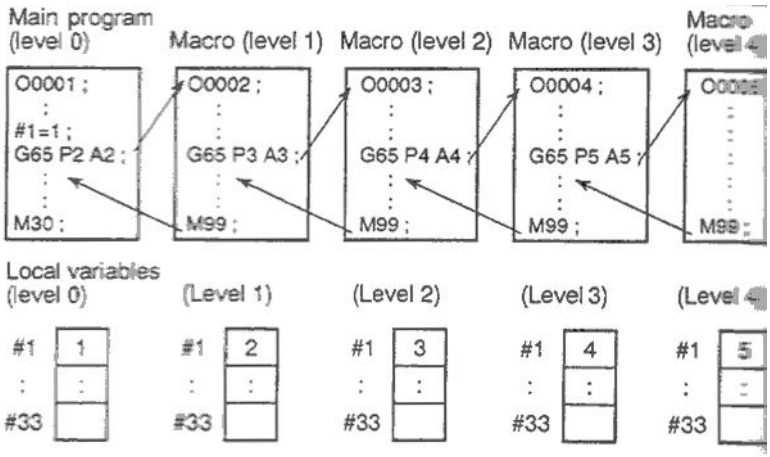
Смесь аргументов спецификации 1 и 2
ЧПУ внутренне различает аргументы спецификации 1 и спецификации 2. Если смешать аргументы типа 1 и 2, тип указанных аргументов определенный позднее будет учтен.

Позиция десятичной точки
Единицы передаваемые в аргументах не могут иметь десятичную точку. Передаваемые значения передаются в единицах наименьшего приращения для каждого адреса. Результирующее значение переданное как аргумент без десятичной точки может отличаться в зависимости от конфигурации станка. Хорошим тоном считается использование десятичной точки в вызовах аргументов макроса для поддержания совместимости программы.

Вложенность вызовов
Вызовы могут иметь вложенность до четырех уровней включая простые вызовы G65 и модальные вызовы G66. Это не включает вызовы подпрограмм M98.

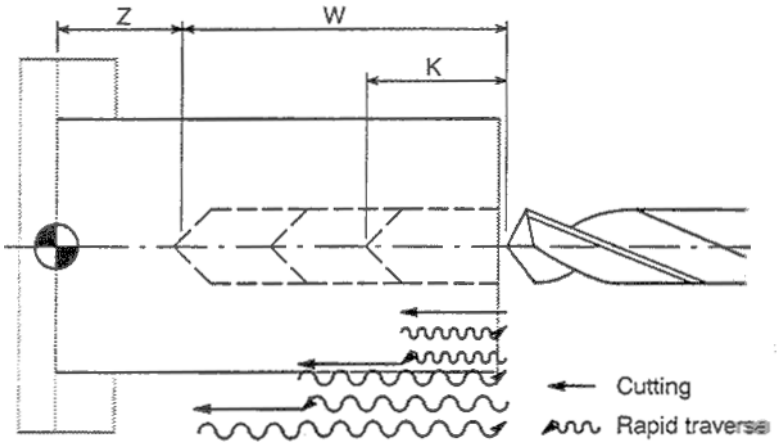
Уровни локальных переменных

- Локальные переменные имеют уровни вложенности от 0 до 4.
- Уровень основной программы 0.
- При каждом вызове макроса G65 или G66, уровень локальных переменных увеличивается на единицу. Значения локальных переменных на предыдущем уровне сохраняется в ЧПУ.
- Если M99 выполняется в макро программе, ЧПУ совершает возврат в основную программу. В это время, уровень локальных переменных уменьшается на единицу. Ранее сохраненные значения локальных переменных восстанавливаются.



Пример программы (цикл сверления)

Предварительно позиционируйте инструмент в позицию X, Z. Укажите Z или W глубину сверления, K глубину прохода, и F скорость подачи сверла.



Cutting – Обработка
Rapid traverse – Быстрое перемещение

Формат вызова

G65 P9100 (Zz или Ww) Kk Ff;
Z: Глубина отверстия (абсолютная величина)
U: Глубина отверстия (Относительная величина)
K: Величина одного прохода
F: Скорость подачи обработки

Программа вызывающая макропрограмму

```
O0002;
G50 X100.0 Z200.0;
G00 X0 Z102.0 S1000 M03;
G65 P9100 Z50.0 K20.0 F0.3;
G00 X100.0 Z200.0 M05;
M30;
```

Программа макроса

```
  O9100;
  #1=0;                      Удаление данных глубины данного отверстия
  #2=0;                      Удаление данных предыдущего отверстия
  IF [#23 NE #0] GOTO 1;     При относительном программировании переход в N1
  IF [#26 EQ #0] GOTO 8;     Если не указана ни Z, ни W, ошибка
  #23=#502-#26               Вычисление глубины отверстия
N1 #1=#1+#6;                 Вычисление глубины данного отверстия
  IF [#1 LE #23] GOTO 2;     Нужно ли сверление
  #1=#23;                    Остановка на глубине отверстия
N2 G00 W-#2;                 Сверление
  G01 W-[#1-#2] F#9;         Цикл сверления
  G00 W#1;                   Переход в начало отверстия
  IF [#1 GE #23] GOTO 9;     Проверка совершено ли сверление
  #2=#1;                     Сохранение глубины этого прохода
  GOTO 1;
N9 M99;
N8 #3000=1 (NOT Z OR U COMMANT);
```

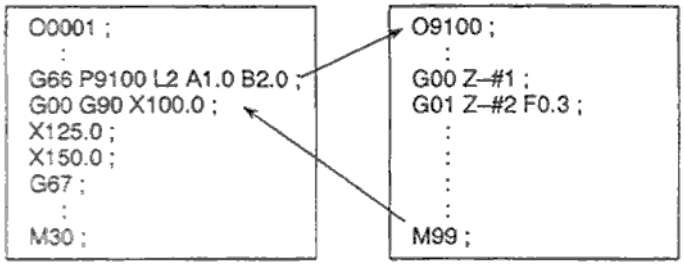
2.15.6.2 Модальный вызов

G66 используется для программирования модального вызова макроса. После каждого перемещения инструмента будет вызываться макрос определенный в G66. Это продолжается пока не будет запрограммирована отмена модального макроса с помощью G67.

Формат

G66 P p L l <передаваемые-аргументы>;

P	Номер вызываемой программы
l	Число повторов (1 по умолчанию)
Аргумент	Данные передаваемые макросу



Пояснения

Вызов

- После G66 укажите в адресе P номер адреса программы с модальным макросом.
- Если требуется повторить макрос несколько раз, в адресе L укажите требуемое число повторов от 1 до 9999. Если L пропущено, считается l = 1.
- Также как и в G65, возможна передача аргументов. Если указать передаваемые аргументы, значения будут присвоены соответствующим локальным переменным.

Отмена

Если указан код G67, модальный вызов макроса больше не будет производиться.

Вложенность вызовов

Вызовы могут иметь вложенность до четырех уровней включая простые вызовы G65 и модальные вызовы G66. Это не включает вызовы подпрограмм M98.

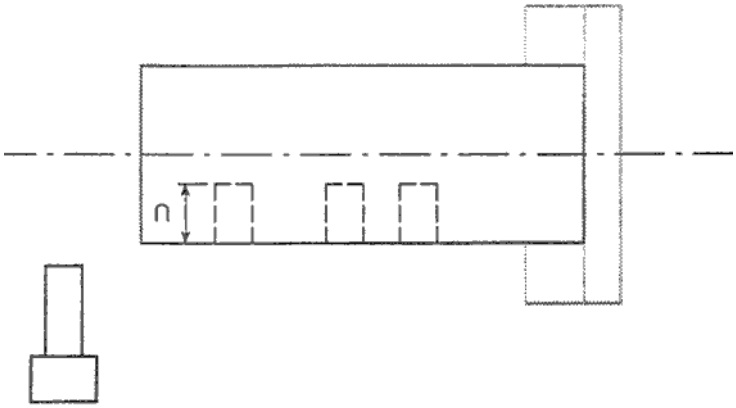
Вложенность модальных вызовов

Модальные вызовы могут иметь вложенность благодаря другому коду G66 во время модального вызова.

Ограничения

- В блоке G66 не может производиться вызов макроса.
- G66 должно быть определено перед аргументами
- Вызов макроса невозможен в блоках содержащих код такой как дополнительные функции не производящие перемещение осей.
- Локальные переменные (аргументы) могут быть установлены в блоках G66. Помните, что локальные переменные не устанавливаются при каждом модальном вызове.

Простая программа
Эта программа делает канавку в указанной позиции.



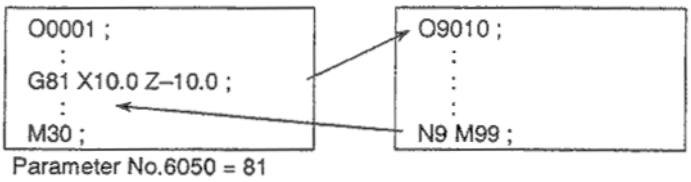
Формат вызова
G66 P9110 Uu Fu;
U Глубина канавки (относительное обозначение)
F Скорость подачи обработки

Программа вызывающая макрос
```
O0003;
G50 X100.0 Z200.0;
S1000 M03;
G66 P9110 U5.0 F0.5;
G00 X60.0 Z80.0;
Z50.0;
Z30.0;
G67;
G00 X00.0 Z200.0 M05;
M30;
```

Программа макроса
```
O9110;
G01 U-#21 F#9;
G00 U#21;
M99;
```

2.15.6.3 Вызов макроса используя G код

Установив G коду номер макроса в параметре, в дальнейшем возможен простой вызов этого макроса.



Пояснения

Установив G код номер от 1 до 9999 для вызова произвольного макроса (9010 – 9019) в соответствующем параметре (6050 – 6069), вызов макроса может быть осуществлен так же, как и программированием G65. Например, если параметр установлен так, что программа макроса O9010 может быть вызвана с помощью G81, для этого не нужно будет изменять программу.

Соответствие между номерами параметров и номерами программы

Номер программы	Номер параметра
O9010	6050
O9011	6051
O9012	6052
O9013	6053
O9014	6054
O9015	6055
O9016	6056
O9017	6057
O9018	6058
O9019	6059

Повторы

Как и при простом вызове макроса, в адресе L можно указать число повторов от 1 до 9999.

Спецификация аргументов

Как и при простом вызове, возможно указание двух типов аргументов. Тип аргумента определяется автоматически в соответствии с используемым адресом.

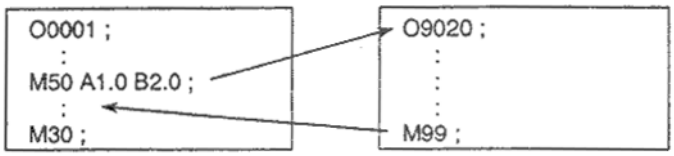
Ограничения

Вложенность вызовов G кода

В программе вызываемой G кодом, невозможно вызывать макрос с помощью G кода. Такой G код будет интерпретирован как обычный G код. В программе вызываемой как подпрограмма с помощью M или T кода, макросы не могут быть вызваны с помощью G кода.

2.15.6.4 Вызов макроса с помощью M кода

Установив M коду номер макроса в параметре, в дальнейшем возможен простой вызов этого макроса.



Parameter 6080 = 50

Пояснения

Установив M код номер от 1 до 9999 для вызова произвольного макроса (9020 – 9029) в соответствующем параметре (6080 – 6089), вызов макроса может быть осуществлен так же, как и программированием G65.

Соответствие между номерами параметров и номерами программы

Номер программы	Номер параметра
O9020	6080
O9021	6081
O9022	6082
O9023	6083
O9024	6084
O9025	6085
O9026	6086
O9027	6087
O9028	6088
O9029	6089

Повторы

Как и при простом вызове макроса, в адресе L можно указать число повторов от 1 до 9999.

Спецификация аргументов

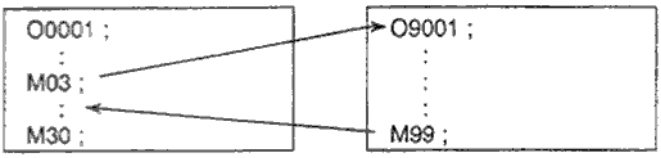
Как и при простом вызове, возможно указание двух типов аргументов. Тип аргумента определяется автоматически в соответствии с используемым адресом.

Ограничения

- M код используемый для вызова макроса должен быть указан в начале блока.
- При вызове макроса с помощью G кода, или в подпрограмме с помощью T или M кода, невозможен вызов макроса с помощью M кода. Такой код будет расценен как обычный M код.

2.15.6.5 Вызов подпрограммы с помощью M кода

Установив M коду номер подпрограммы в параметре, в дальнейшем возможен простой вызов этой подпрограммы, как и с помощью M98.



Parameter 6071 = 03

Пояснения

Установив M код номер от 1 до 99999999 для вызова произвольного макроса (9001 – 9006) в соответствующем параметре (6071 – 6076), вызов подпрограммы может быть осуществлен так же, как и программированием M98.

Соответствие между номерами параметров и номерами программы

Номер программы	Номер параметра
O9000	6070
O9001	6071
O9002	6072
O9003	6073
O9004	6074
O9005	6075
O9006	6076
O9007	6077
O9008	6078
O9009	6079

Повторы

Как и при простом вызове, в адресе L можно указать число повторов от 1 до 9999.

Спецификация аргументов

Передача аргументов невозможна

M код

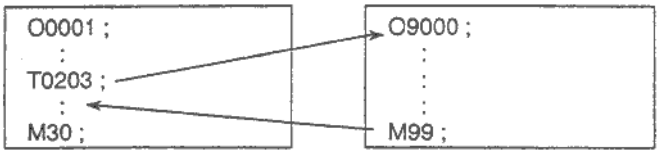
M код в программе макроса, который был вызван расценивается как обычный M код.

Ограничения

При вызове макроса с помощью G кода, или в подпрограмме с помощью T или M кода, невозможен вызов макроса с помощью M кода. Такой код будет расценен как обычный M код.

2.15.6.6 Вызовы подпрограммы с помощью T кода

Включив возможность вызова подпрограмм (программ макроса) с помощью T кода в параметре, программ макроса может быть вызвана простым программированием T кода в программе обработки.



Bit 5(TCS) of parameter No.6001 = 1

Пояснения

Установив бит 5 параметра 6001 в 1, программа макроса O9000 может быть вызвана T кодом указанным в программе обработки. T код указываемый в программе обработки назначается общей переменной #149.

Ограничения

При вызове макроса с помощью G кода, или в подпрограмме с помощью T или M кода, невозможен вызов макроса с помощью T кода. Такой код будет расценен как обычный T код.

2.15.6.7 Пример программы

Функция вызова подпрограммы с помощью М кода при каждом вызове будет осуществлять измерение времени работы каждого инструмента.

Условия

- Совокупное время использования инструмента подсчитывается для инструментов 1 – 5. Для инструментов 6 и выше время не подсчитывается.
- Следующие переменные используются для сохранения номеров инструмента и измеренного времени:

#501	Время использования инструмента 1
#502	Время использования инструмента 2
#503	Время использования инструмента 3
#504	Время использования инструмента 4
#505	Время использования инструмента 5

- Начало подсчета времени использования начинается при подаче команды М03 и останавливается по команде М05. Системная переменная #3002 используется для измерения времени работы лампы старта цикла. Время остановки станка при остановке подачи и поблочном режиме не учитывается, но учитывается время смены инструмента и паллеты.

Проверка операции

Установка параметра

Установите 3 в параметр 6071, и установите 05 в параметр 6072.

Установка значений переменных

Установите 0 в переменные #501 – #505.

Программа осуществляющая вызов макроса

```
O0001;
T0100 M06;
M03;
 ..
M05;                    Изменение #501
T0200 M06;
M03;
 ..
M05;                    Изменение #502
T0300 M06;
M03;
 ..
M05;                    Изменение #503
T0400 M06;
M03;
 ..
M05;                    Изменение #504
T0500 M06;
M03;
 ..
M05;                    Изменение #505
M30;
```

Программа макроса

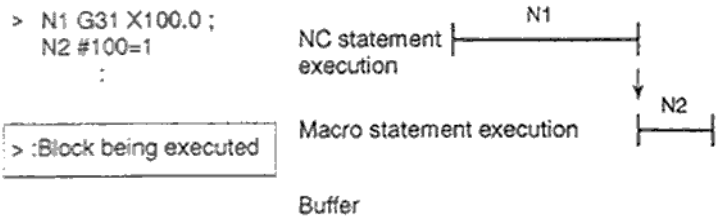
```
O9001(M03)                          Макроса начала расчета
  M01;
  IF[FIX[$4120/100] EQ 0] GOTO 9;   Инструмент не назначен
  IF[FIX[$4120/100] GT 5] GOTO 9;   Номер инструмент не подходит
  #3003=0;                          Очистка таймера
N9 M03;                             Вращение шпинделя вперед
  M99;

O9002(M05);                         Макрос завершения подсчета
  M01;
  IF[FIX[$4120/100] EQ 0] GOTO 9;   Инструмент не назначен
  IF[FIX[$4120/100] GT 5] GOTO 9;   Номер инструмент не подходит
  #[500+FIX[#4120/100]]=#3002+#[500+FIX[#4120/100]];   Расчет времени
N9 M05;                             Остановка шпинделя
  M99;
```

2.15.7 Обработка макро определения

Для точной обработки ЧПУ предварительно считывается определения ЧПУ для последующего выполнения. Эта операция называется буферизацией. В режиме компенсации радиуса инструмента G41, G42, ЧПУ считывает наперед определения ЧПУ на два или три блока вперед для определения пересечения. Макро определения для арифметических выражений и логически операций ветвления обрабатываются сразу после поступления в буфер. Блоки содержащие M00, M01, M02 или M30, блоки содержащие M коды для которых буферизация отключена установкой параметра (3411 – 3420) и блоки содержащие G31 не буферизируются.

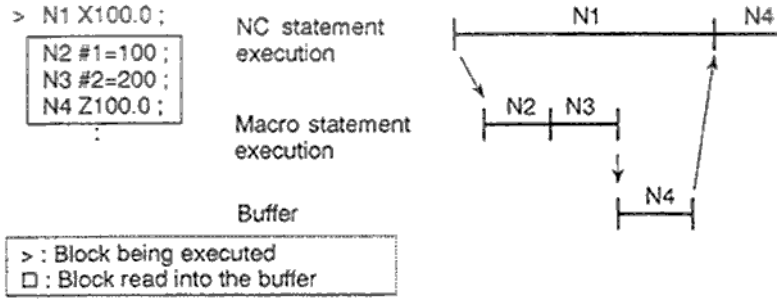
Пояснения

Если следующий блок не буферизирован (M коды не буферизируемые, G32 и т.д.)



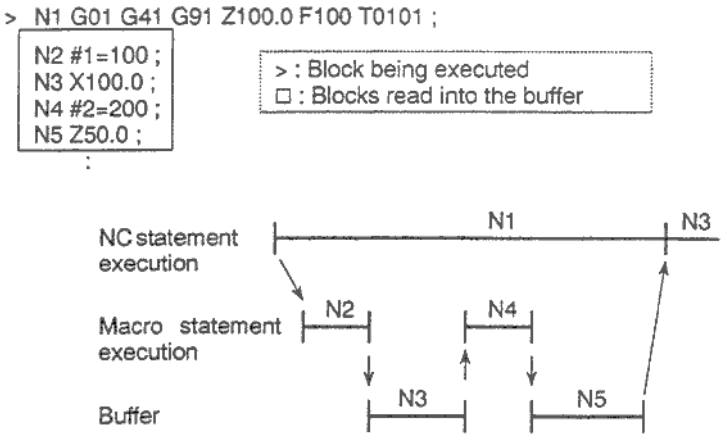
NC statement execution – Выполнение определения ЧПУ
Macro statement execution - Выполнение определения макроса

Буферизация следующего блока не в режиме компенсации радиуса кромки (нормальное предварительное чтение одного блока)



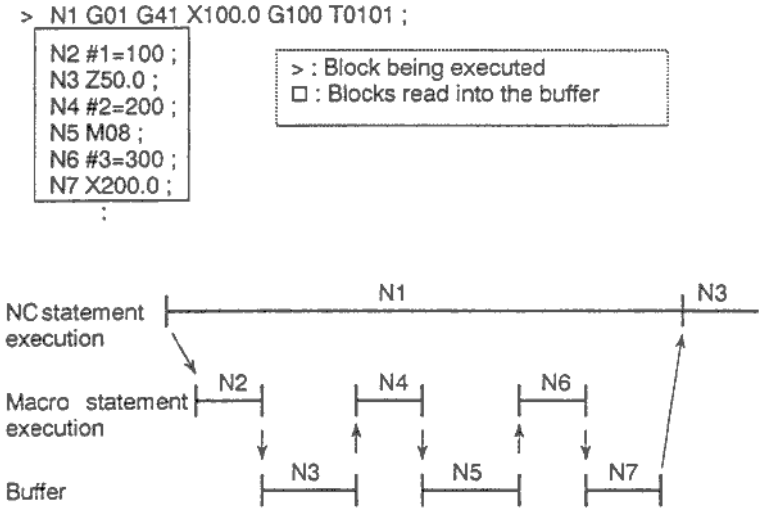
Если N1 выполнен, следующее определение ЧПУ (N4) считывается в буфер. Макро определение N2, N3 между N1 и N4 обрабатывается во время выполнения N1.

Буферизация следующего блока в режиме компенсации радиуса кромки (G41, G42)



Если N1 выполнен, ЧПУ считывается в буфер два следующих блока (до N5). Макро определение N2, N4 между N1 и N5 обрабатывается во время выполнения N1.

Если следующий блок не содержит перемещения в режиме компенсации радиуса кромки C (G41, G42)



Если блок N1 выполнен, ЧПУ считывается в буфер два следующих блока (до N5). Так как блок N5 не создает перемещения, пересечение не рассчитывается. В этом случае определение ЧПУ в следующих трех блока (до N7) считываются. Макро определение N2, N4 и N6 между N1 и N1 обрабатывается во время выполнения N1.

2.15.8 Регистрация произвольного макроса

Произвольная макро программа похожа на подпрограмму. Она может быть зарегистрирована и отредактирована как и подпрограмма. При подсчете хранимых программ объем рассчитывается общей длиной ленты используемой для хранения как макросов, так и подпрограмм.

2.15.9 Ограничения

MDI
Вызов макроса возможен с панели MDI. Однако, во время автоматического управления невозможно переключить режим MDI для выполнения макроса.

Поиск номера блока
Невозможен поиск произвольного блока для программы макроса.

Поблочный режим
Во время выполнения программы макроса, выполнение для отдельных блоков может быть прервано (за исключением блоков содержащих вызовы макроса, команды арифметических операций, и команды управления).
Блок содержащий вызов макроса (G65, G66, G67) не останавливает выполнение в поблочном режиме. Блоки содержащие арифметические операции и команды управления могут останавливать станок в поблочном режиме если установить бит 5 параметра 6000 в 1.
Поблочный режим используется для тестирования программ. Помните, что если выполнение приостановлено в поблочном режиме с функцией компенсации радиуса кромки инструмента, предполагается что движение и компенсация не могут выполняться в некоторых случаях.

Пропуск опционального блока
Если в середине блока указан символ /, считается что он является символом деления если находится в скобках и в правой части выражения. В этом случае этот символ не считается символом пропуска блока.

Работа в режиме редактирования EDIT
Если установить бит 0 параметра 3202 и бит 4 параметра 3202 в 1, удаление и редактирование отключается для произвольных программ макроса и подпрограмм с номерами 8000 – 8999 и 9000 – 9999. Зарегистрированные программы макроса и подпрограммы должны быть защищены от случайного изменения. Если вся память очищена (нажатием клавиш RESET и DELETE одновременно во время включения питания), содержание всей памяти включая макросы будет удалено.

Сброс
Операцией сброса локальные переменные и общие переменные #100 - #149 удаляются и устанавливаются в значение null. Они могут быть защищены от удаления настройки CLV и CCV (биты 7 и 6 параметра 6001). Системные переменные #1000 - #1133 не удаляются.

Отображение дисплея PROGRAM RESTART
Как и с M98, M коды и T коды вызовов подпрограмм не отображаются.

Остановка подачи.
Если во время выполнения макроса будет осуществлена остановка подачи, станок остановится после выполнения макроса. Станок может быть остановлен в результате сброса или аварийной остановки.

Значения констант, которые могут использоваться в <выражении>
+0.0000001 - +99999999
-99999999 - -0.0000001
Число цифр не должно превышать восьми. В противном случае появится сигнал тревоги 003.

2.15.10 Команды внешнего вывода

В дополнение к стандартным командам произвольного макроса, возможны следующие макро команды. Они называются командами внешнего вывода

- **BPRNT**
- **DPRNT**
- **POPEN**
- **PCLOS**

Эти команды используются для вывода переменных и символов через интерфейс считывающего перфоратора.

Пояснения

Укажите эти команды в следующем порядке:

Команда открытия: POPEN

Перед программированием последовательности команд вывода данных, запрограммируйте команду для установления соединения с внешним устройством ввода/вывода.

Команда вывода данных: BPRNT или DPRNT

Укажите необходимые данные для вывода.

Команда закрытия: PCLOS

Если все данные переданы, запрограммируйте PCLOS для завершения соединения с внешним устройством ввода/вывода.

Команда открытия POPEN

Команда POPEN устанавливает соединение с внешним устройством ввода/вывода. Это должно быть сделано до начала передачи данных. ЧПУ подает управляющий сигнал DC2.

Команда передачи данных BPRNT

BPRNT [a #b [c] …]

где

a символ

#b переменная

[c] Число значащих десятичных мест

Команда BPRNT производит передачу символов и переменных в двоичном коде.

(i) Указанные символы конвертируются согласно ISO кодам данных установленных для передачи. Возможно задать следующие символы:
- Буквы (A – Z)
- Числа
- Специальные символы (*, /, +, - и т.д.)

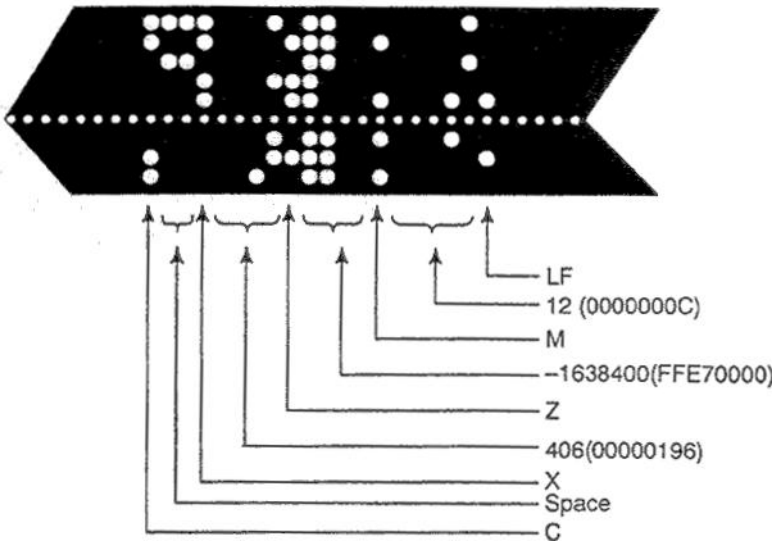
(ii) Все переменные сохраняются с десятичной точкой. Укажите переменную и в скобках число десятичных знаков. Значение переменной ограничено двумя словами (32 бита), включая десятичную точку. Оно передается как двоичный код начиная от старшего байта.

(iii) Когда передаются указанные данные. код EOB передается в соответствии с настройками ISO.

(iv) Переменные равные null, передаются как 0.

Пример:

```
BPRINT [ C** X#100 [3] Z#101 [3] M#10 [0] ]
Variable value
  #100=0.40596
  #101=–1638.4
  #10=12.34
```



Команда передачи данных DPRNT
DPRNT [a #b [c d] …]

где

a	символ
#b	переменная
[c]	Число значащих десятичных мест в целой части
[d]	Число значащих десятичных мест

Команда DPRNT производит передачу символов и переменных в коде определенном в настройках ISO.

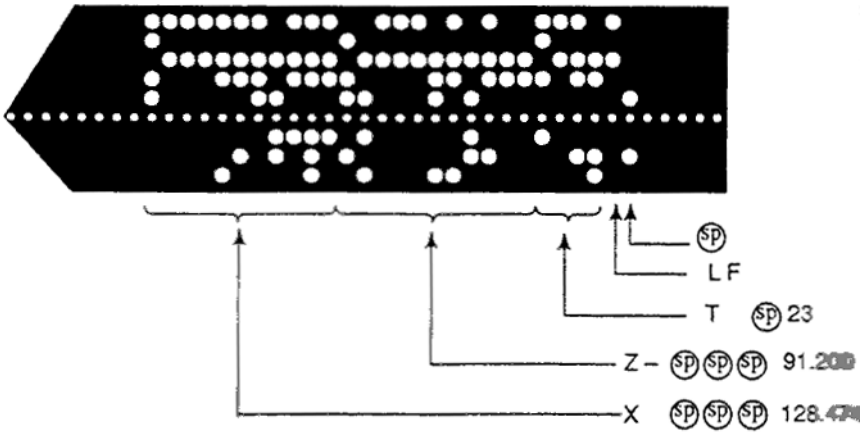
(i) См. описание (i) – (iii) для команды BPRNT.

(ii) При передаче переменных, укажите # и номер переменной, затем укажите число цифр целой части и число десятичных знаков в скобках.
Один код указывается для передачи одного значения, передача начинается со старшего байта. Для каждой цифры код соответствует настройкам ISO. Десятичная точка также соответствует настройкам ISO.
Каждая переменная должна содержать не более 8 цифр.
Если наивысшая цифра равна нулю, нулевое значение не передается если бит 1 параметра 6001 установлен в 1.
Если число десятичных знаков не равно нулю, цифры в десятичной части всегда передаются. Если число десятичных знаков равно нулю, десятичная часть не передается. Если бит 1 параметра 6001 установлен в 0, код пропуска передается для обозначения положительного числа вместо +, в противном случае код пропуска не передается.

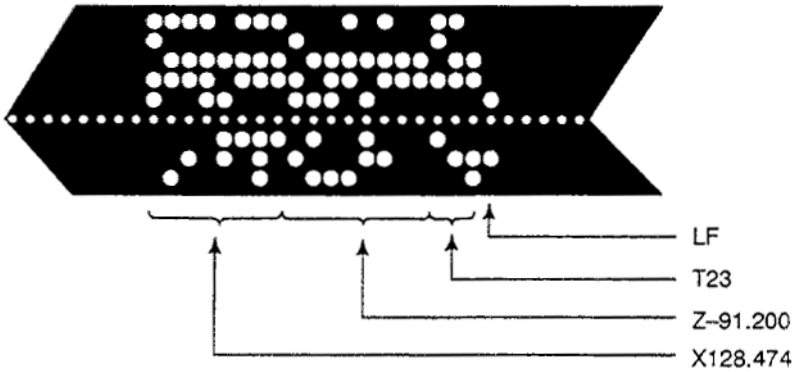
Пример:

```
DPRNT [ X#2 [53] Z#5 [53] T#30 [20] ]
Variable value
  #2=128.47398
  #5=-91.2
  #30=123.456
```

(1) Parameter PRT(No.6001#1)=0



(2) Parameter PRT(No.6001#1)=1



Команда закрытия PCLOS

Команда PCLOS завершает соединение с внешним устройством ввода/вывода. Укажите эту команду после того как все данные были переданы. ЧПУ передаст код DC4.

Требуемые настройки

Укажите используемый канал в параметре 020. В соответствии со значением этого параметра, данные интерфейса будут переданы с данными настройками (например, скорость передачи данных.

I/O канал 0: параметры 101, 102 и 103

I/O канал 1: параметры 111, 112 и 113

I/O канал 2: параметры 121, 122 и 123

Никогда не указывайте вывод на кассету Fanuc или гибкий диск.

Если указана команда DPRNT для вывода данных, укажите предварительные нули для определения пробелов (установка бита 1 параметра 6001 в 1 или 0). Для определения конца строки данных ISO кода, укажите использование LF (бит 3 NCR параметра 0103 в 0) или LF и CR (NCR в 1).

ПРИМЕЧАНИЕ

1. Необходимо всегда указывать команду POPEN перед командой BPRNT, DPRNT и команду PCLOS вместе. Если команда POPEN один раз указана в программе, ее не нужно повторять пока не будет дана команда закрытия.
2. Убедитесь, что команды открытия и закрытия указаны. Укажите команду закрытия в конце программы. Однако, не обязательно программировать команду закрытия если на указана программа открытия.
3. Если во время выполнения команды вывода данных совершен сброс, вывод прекращается и данные удаляются. Поэтому, если сброс осуществляется командой M30 в конце программы осуществляющей вывод данных, укажите команду закрытия до того как команда M30 будет выполнена.
4. Аббревиатура макросов в скобках [ ] остается неизменной. Однако, помните что эти символы в скобках делятся и вводятся несколько раз, следующие аббревиатуры конвертируются и вводятся.
5. O может быть указано в скобках [ ]. Помните, что если символы указаны в скобках, они делятся и вводятся несколько раз, в следующей передаче O пропускается.

2.15.11 Функция прерывания

При выполнении программы, другая программа может быть вызвана с помощью сигнала прерывания (UINT). Эта функция называется функция прерывания произвольного макроса. Формат программирования прерывания следующий:

M96 Pxxxx; Включения прерывания произвольного макроса
M97; Выключение прерывания произвольного макроса

Пояснения

Использование функции прерывания позволяет вызвать программу во время выполнения любого другого блока. Это позволяет обрабатывать ситуации возникающие время от времени.

1) Если обнаружен ненормальный инструмент, сигнал подан с внешнего устройства.

2) Последовательность операций обработки прерывается другой последовательностью отменяющей текущую операцию.

3) Через регулярные интервалы считывается информация текущей обработки. Приведенные примеры позволяет учитывать изменившиеся условия.

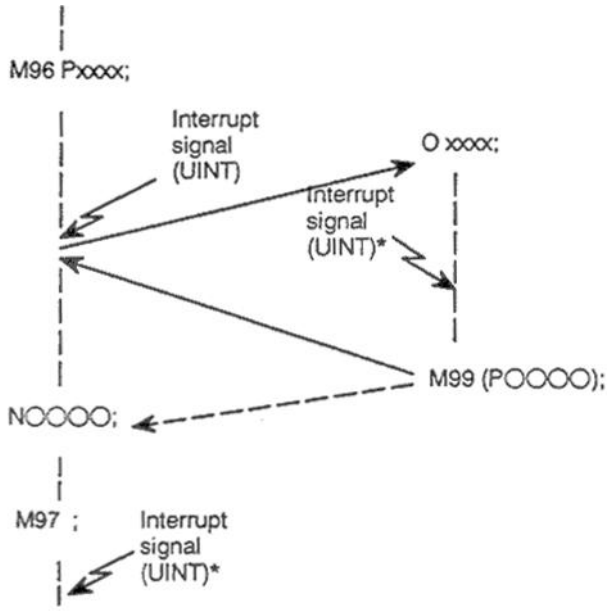


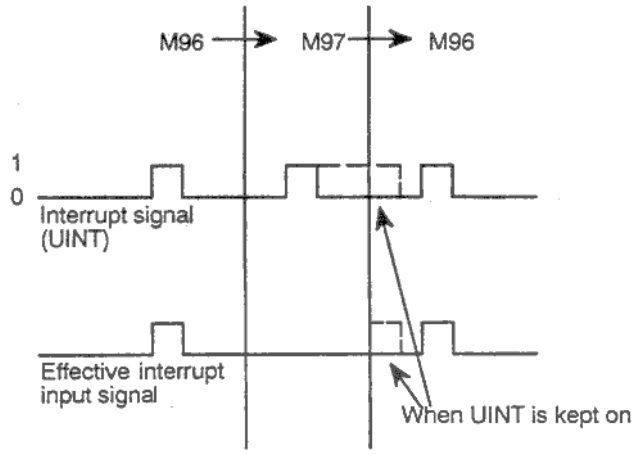
Рис. 15.11 Функция прерывания

Если в программе запрограммировано M96Pxxxx, последовательное выполнение программы может быть прервано сигналом UINT для выполнения программы указанной в Pxxxx.
Если появился сигнал UINT во время выполнения программы прерывания или после M97, он игнорируется.

2.15.11.1 Метод спецификации

Условия прерывания
Исполнение макроса прерывания возможно при выполнении следующих условий:
- **Выбран режим работы в памяти или MDI**
- **Включено STL (лампа запуска)**
- **В данный момент макрос прерывания не выполняется**

Спецификация
В основном, функция прерывания используется включением с помощью M96 и M97 для отключения.
Запрограммировав один раз M96, программа макроса будет вызвана при появлении сигнала M96 до команды M97. После команды M97 или сброса ЧПУ, макрос не будет вызываться сигналом UINT. Сигнал прерывания UINT игнорируется пока не будет запрограммировано M96.



Сигнал прерывания контролируется после программирования M96.

ПРИМЕЧАНИЕ
Описание схемы срабатывания по состоянию или по изменению, см. далее.

2.15.11.2 Описание функционирования

Прерывание подпрограммы и прерывание макроса
Существует два типа прерывания: тип прерывания подпрограммы и макроса. Тип прерывание выбирается битом 5 параметра 6003.

a) Прерывание типа подпрограммы
При подаче сигнала прерывания вызывается подпрограмма. Это означает, что уровень локальных переменных не изменяется. Не изменяется уровень вложенности.

b) Прерывание типа макроса
При подаче сигнала прерывания вызывается макрос. Это означает, что уровень локальных переменных изменяется до и после прерывания. Не изменяется уровень вложенности макроса. Если при выполнении программы прерывания производится вызов подпрограммы или макроса, уровень вложенности изменяется. Аргументы не могут передаваться из текущей программы.

M коды управления прерываниями
В основном, для использования макросов используются M96 и M97. Однако, эти M коды могут быть заняты для других целей некоторыми производителями станков. Для таких случаев существует бит 4 параметра 6003 использующийся для определения M кода управления прерыванием.
При установке этого параметра, с помощью параметров 6033 и 6034 могут быть определены M коды прерывания.
В параметре 6033 устанавливаемся M код включения прерывания, и в параметре 6034 код отключения прерывания.
Если эти параметры не используются, для включения и выключения прерывания используются M коды M96 и M97.

Прерывания макроса и определения ЧПУ

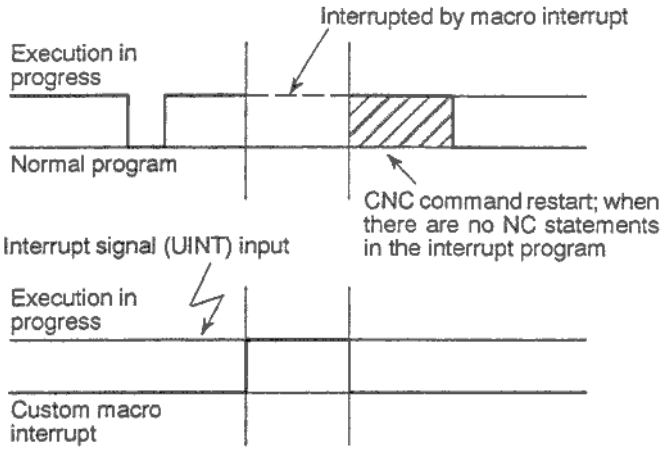
При выполнении макроса прерывания, пользователь может захотеть прервать выполняемое определение ЧПУ, или прерывать выполнение текущего блока. Бит 2 MIN параметра 6003 используется для определения места выполнения прерывания посреди блока, или в конце.

Тип 1 (Прерывание производится посреди блока)

(i) При появления сигнала UINT все движения, или функция задержки немедленно останавливаются и выполняется программа прерывания.

(ii) Если в момент запуска программы прерывания в интерпретируемом блоке находится определение ЧПУ, команда теряется и выполняется программа прерывания. После возвращения управления в программу, управление передается в следующий блок после прерванного блока.

(iii) Если в прерванной программе нет определения ЧПУ, управление передается в прерванную программу с помощью M99, после этого программа перезапускается с команды в интерпретированном блоке.

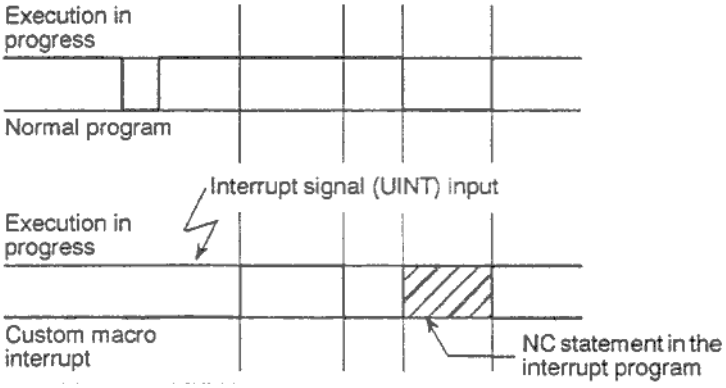


Execution in progress – Производится выполнение
Interrupted by macro interrupt – Прерывание
CNC command restart – Перезапускаемая команда ЧПУ если в прерванной программе нет определения ЧПУ

Тип 2 (Прерывание производится в конце блока)

(i) Если выполняемый блок не содержит операций в несколько циклов таких как сверление и возврата в точку ссылки, прерывание производится следующим образом:
При появлении сигнала UINT, макрос выполняется сразу. Определение ЧПУ не выполняется пока текущий блок не будет завершен.

(ii) Если блок содержит операции из нескольких циклов, прерывание производится следующим образом:
После завершения последнего движения, выполняется макрос. Определение ЧПУ выполняется после завершения макроса.



Execution in progress – Производится выполнение
Interrupt – Прерывание

Условия включения и выключения контроля сигнала прерывания

Контроль сигнала прерывания включается после блока содержащего M96. Отключение контроля сигнала прерывания производится командой M97.

Во время выполнения программы прерывания, контроль сигнала прерывания не производится. Контроль сигнала включается сразу после завершения программы прерывания и возврата в основную программу.

Прерывание во время выполнения блока из нескольких циклов

Для типа 1

Даже если выполняется операция, движение прерывается и выполняется программа прерывания. Если программа прерывания не содержит определения ЧПУ, операция цикла после возврата будет перезапущена. Если программа содержит определения ЧПУ, оставшаяся часть цикла не будет выполнена, программа будет продолжена со следующего блока.

Для типа 2

После завершения операции цикла, начнется выполнения макроса прерывания до определения ЧПУ. После выполнения прерывания, программа ЧПУ будет продолжена.

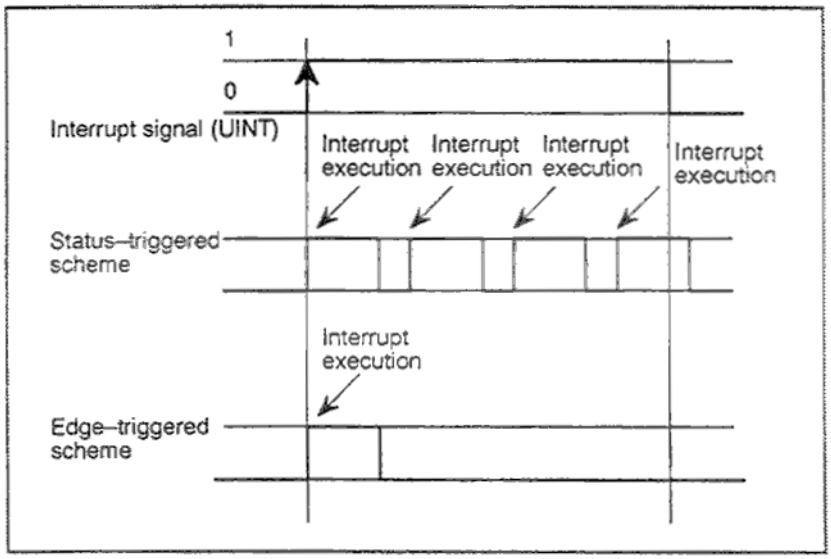
Сигнал прерывания UINT

Существует две схемы сигнала прерывания UINT: схема статуса сигнала и схеме срабатывания при изменении. Схема статуса считает сигнал подаваемым, если на входе есть сигнал. Схема срабатывания при изменении считает что сигнал подается в момент перехода из режима выкл, в режим вкл.

Одна из двух схем выбирается битом 3 параметра 6003. Если выбрана схема статуса, сигнал прерывания UINT считается присутствующим пока он подается на ход. Удержание сигнала приведет к повторению выполнения программы прерывания.

При выборе схемы срабатывания при изменении, сигнал UINT считается присутствующим только в момент его включения, поэтому программа выполняется только один раз. Если схема статуса не подходит или если программа макроса должна выполняться только один раз, следует использовать схему срабатывания при изменении.

За исключением особых случаев, как правило обе схемы приводит к одному результату. Время от подачи сигнала то выполнения прерывания не зависит от схемы сигнала.



Status triggered scheme – Схема срабатывания по статусу
Edge-triggered scheme – Схема срабатывания при изменении

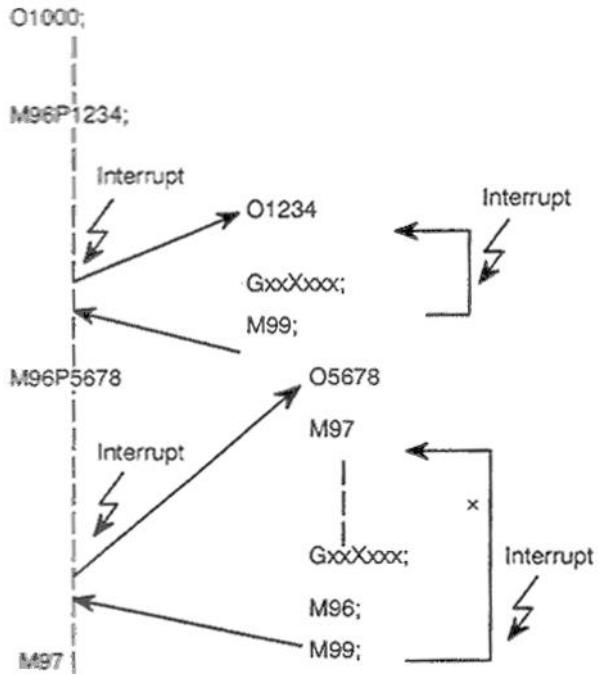
В этом примере при схеме статуса программа прерывания выполняется четыре раза, тогда как при схеме срабатывания при изменении только один раз.

Возврат после выполнения прерывания

Для возврата из программы прерывания, запрограммируйте M99. В программе прерывания также может быть указан адрес P, в этом случае управление будет передано в указанный блок.

При выполнении программы макроса не создается прерываний. Для возможности выполнения прерывания, запрограммируйте M99. Если M99 указано отдельно, но будет выполнено до завершения предыдущей операции. Поэтому, при выполнении последней команды прерывания, возможно срабатывание прерывания. Если это нежелательно, необходимо использовать команды M96 и M97 в программе.

Во время выполнения макроса прерывания исполнение других макросов прерывания невозможно, они автоматически запрещаются. M99 указанное отдельно будет выполнено до завершения предыдущего блока. В следующем примере прерывание возможно в блоке O01234. Если сигнал подан, O1234 будет выполнено снова. O5678 контролируется M96 и M97, и поэтому не может быть выполнено прерыванием.



ПРИМЕЧАНИЕ

Если M99 содержит только адреса O, N, P, L или M, этот блок считается продолжением предыдущего блока программы. Поэтому, поблочная остановка не производится в этом блоке. В терминах программировани1 (1) и (2) являются одним и тем же. (Разница где будет выполнено Gxx до M99).

(1) Gxx Xxxx;
M99;

(2) Gxx Xxxx M99;

Прерывание и модальная информация

Выполнение макроса прерывания отличается от нормального вызова программы. Оно инициируется сигналом UINT. В основном, любые изменения модальной информации не должны влиять на программу прерывания. Поэтому, даже в случае изменения модальной информации в программе прерывания, модальная информация восстанавливается в состояние до вызова программы прерывания.

После возвращения управления из программы прерывания с помощью M99 Pxxxx, модальная информация снова контролируется программой. В этом случае модальная информация измененная программой прерывания доступна основной программе. Восстановление старой модальной информации до прерывания нежелательно. Это происходит потому что после возврата управления, некоторые программы могут работать по-разному в зависимости от модальной информации. В этом случае действуйте следующим образом:

1) Программа прерывания передающая модальную информацию используемую после возврата в прерванную программу.

2) После возврата в прерванную программу, модальная информация при необходимости определяется снова.

Модальная информация при возврате с помощью M99

Модальная информация восстанавливается в состояние до вызова программы прерывания.

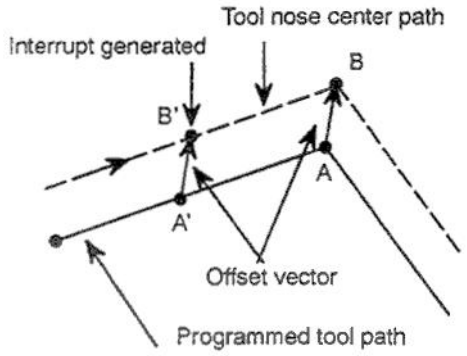
Модальная информация при возврате с помощью M99 Pxxxx

Новая информация измененная в программе прерывания остается действительной. Старая модальная информация до прерывания может быть прочитана в системных переменных #4001 - #4120.

Помните, что если модальная информация изменена в программе прерывания, системные переменные 4001 – 4120 не изменяются.

Системные переменные (информация позиции) для программы прерывания

- Точка координат A может быть прочитана из системной переменной #5001
- Точка координат A’ может быть прочитана после определения ЧПУ без указания движения
- Координаты станка и координаты детали точки B’ могут быть прочитаны из системной переменной #5021 и далее и #5041 и далее.



Tool nose center path – Пут перемещения кромки инструмента
Offset vector – Вектор офсета
Programmed tool path – Запрограммированный путь инструмента
Interrupt generated – Генерируемое прерывание

Прерывание и модальный вызов макроса

Если осуществляется прерывание UINT и вызвана программа прерывания, модальный вызов макроса отменяется. Однако, если G66 запрограммировано в макросе прерывания, модальный макрос будет включен. После выхода из программы будет восстановлено предыдущее значение модального макроса. После возврата с помощью M99Pxxx, остается значение установленное в программе макроса.

Прерывание и перезапуск программы

Сигнал прерывания в тестовом режиме после операции поиска приводит к перезапуска, программа прерывания вызывается после завершения операции перемещения для всех осей. Это означается что используется тип прерывания 2 вне зависимости от настроек параметра.

2.16 Программируемый ввод параметра (G10)

Введение

Значения параметров может быть введено в программе. Эта функция используется для установки значений компенсации при использовании подключаемого модуля или для максимальной скорости подачи обработки или изменения констант времени обработки для соответствия условиям обработки.

Формат

G10L50; Режим установки параметров
N_R_; Для параметров отличных от осей
N_P_R_; Для параметров осей
..
..
G11; Отмена режима установки параметров

Значение команд

N_; Параметр (4 цифры) или позиции компенсации (0 –1023) шага +10,000 (5 цифр)
R_; Значение устанавливаемое параметру (начальный ноль может быть пропущен
P_; Ось номер 1 – 8 (используется ввода параметров осей)

Пояснения

Значение устанавливаемое параметру (R_)

Не используйте десятичную точку в параметре R_. Десятичная точка может быть использована в макросе R_.

Номер оси P_

Укажите номер оси P_ от 1 до 8 для параметра оси. Управляемые оси нумеруется в порядке в котором они отображены на экране ЧПУ.
Например, укажите P2 для второй отображаемой оси.

ВНИМАНИЕ

1. Обязательно выполните ручной переход в точку ссылки после изменения компенсации. В противном случае возможно неправильное отображение позиции.
2. Перед вводом параметров необходимо отменить режим фиксированных циклом. В противном случае при выполнении движения будет вызван цикл сверления.

ПРИМЕЧАНИЕ

Во время ввода параметров не могут быть заданы другие определения ЧПУ.

Примеры

1. Установка бита 2 параметра 3404.

```
G10L50;
N3404 R00000100;
G11;
```

2. Изменение значений осей Z, C в параметре 1322 (координаты ограничения перемещения в положительном направлении для каждой оси)

```
G10L50;
N1322P3R4500;
N1322P4R12000;
G11;
```

2.17 Операции с памятью ленточного формата Series 10/11

Программы в ленточном формате в Series 10/11 могут быть зарегистрированы для работы в памяти установкой бита 1 параметра 0001. Регистрация в памяти и работа с памятью возможна для функций в формате Series 10/11 также как и для следующих функций имеющих другой формат ленты:

- Нарезание резьбы с постоянным шагом
- Вызов подпрограммы
- Фиксированные циклы
- Многократно выполняемые фиксированные циклы
- Фиксированные циклы сверления

Примечание
Регистрация в памяти и операции в памяти возможны только для функций поддерживаемых ЧПУ.

2.17.1 Адреса и возможные значения для формата Series 10/11

Некоторые адреса, которые не могут быть использованы в данном ЧПУ, могут быть определены в ленточном формате Series 10/11. Возможные значения для Series 10/11 такие же как и для ЧПУ, см. 2.17.2 – 2.17.5. Если указано недопустимое значение, появится сигнал тревоги.

2.17.2 Нарезание резьбы с постоянным шагом

Формат
G32IP_F_Q_;
или
G32IP_E_Q;
IP Комбинация адресов осей
F Шаг по продольной оси
E Шаг по продольной оси
Q Место начала нарезания резьбы (игнорируется если указано)

Пояснения

Хотя Series 15 позволяет оператору указать число шагов на дюйм в адресе E, ленточный формат Series 15 не содержит адреса E и поэтому F используется также как и E. Адрес Q указывает сдвиг начального угла нарезания резьбы. Может быть указан, но будет проигнорирован.

Возможные значения для шага резьбы

Адрес		Мм ввод	Дюймовый ввод
E		0.0001 – 500.0000 мм	0.000001 дюйма – 9.999999 дюйма
F	Команда с десятичной точкой	0.0001 – 500.0000 мм	0.000001 дюйма – 9.999999 дюйма
	Команда без десятичной точки	0.01 – 500.00 мм	0.0001 – 9.9999 дюйма

Возможные значения для скорости подачи

Адрес			Мм ввод	Дюймовый ввод
F	Подача на оборот		0.01 – 500 мм/об	0.0001 – 9.9999 дюйм/об
	Подача в минуту	IS-B	1 – 24000 мм/мин	0.01 – 9600.00 дюйм/мин
		IS-C	1 – 100000 мм/мин	0.01 – 9600.00 дюйм/мин

ВНИМАНИЕ
Укажите скорость подачи один или более раз при переключении между подачей в минуту и подачей на оборот

2.17.3 Вызов подпрограммы

Формат
M98PxxxxLxxxx;
P Номер подпрограммы
L Число повторений

Пояснения

Адрес
Адрес L не может использоваться в данном ленточном формате ЧПУ, но может использоваться в формате Series 15.

Номер подпрограммы
Диапазон значений адреса номера подпрограммы от 1 до 9999. Если указано значение содержащее больше 4 цифр, используются последние четыре цифры.

Число повторов
Число повторов может иметь значение от 1 до 9999. Если оно не указано, используется значение 1.

2.17.4 Фиксированные циклы

Формат
Прямая обработка
G90X_Z_F_;

Конусная обработка
G90X_Z_I_F_;
I: Длина секции конуса по оси X (радиус)

Нарезание резьбы
G92X_Z_F_Q_;
F: Шаг резьбы
Q: Сдвиг начального угла (игнорируется)

Нарезание резьбы на конусе
G92X_Z_I_F_;
I: Длина секции конуса по оси X (радиус)

Обработка торца
G94X_Z_F_;
G94X_Z_K_F_;
K: Длина секции конуса по оси Z

Адрес
Адреса I и K не могут быть использованы для фиксированных циклов в данном ленточном формате ЧПУ, но могут использоваться в Series 15. Адрес Q может быть указан в Series 15, но будет проигнорирован.

Допустимые значения подачи
Ограничения такие же, как и в 2.17.2.

2.17.5 Многократно повторяемые циклы

Формат

Обработка поверхности

G71P_Q_U_W_I_K_D_F_S_T_;

I: Длина и направление припуска для чистового прохода по X (игнорируется)
K: Длина и направление припуска для чистового прохода по Z (игнорируется)
D: Глубина прохода

Обработка торца

G72P_Q_U_W_I_K_D_F_S_T_;

I: Длина и направление припуска для чистового прохода по X (игнорируется)
K: Длина и направление припуска для чистового прохода по Z (игнорируется)
D: Глубина прохода

Циклическая обработка

G73P_Q_U_W_I_K_D_F_S_T_;

I: Длина и направление припуска для чистового прохода по X
K: Длина и направление припуска для чистового прохода по Z
D: Число делений

Снятие материала (отрезание)

G74X_Z_I_K_F_D_;

или

G74U_W_I_K_F_D_;

I: Расстояние перемещения по оси X
K: Глубина врезания по оси Z
D: Припуск в конце обработки

Снятие материала на диаметре

G75X_Z_I_K_F_D_;

или

G75U_W_I_K_F_D_;

I: Расстояние перемещения по оси X
K: Глубина врезания по оси Z
D: Припуск в конце обработки

Многопроходное нарезание резьбы

G76X_Z_I_K_F_D_A_P_Q_;

I: Расстояние перемещения по оси X
K: Глубина врезания по оси Z
D: Припуск в конце обработки
A: Угол кромки инструмента
P: Метод обработки

Адреса и допустимые значения

Если следующие адреса указаны в ленточном формате Series 15, они будут проигнорированы.

- I и K для чернового цикла обработки диаметра (G71)
- I и K для чернового цикла обработки торца

Адрес P определяет тип обработки для многопроходного цикла G76 и всегда P1 вне зависимости от определения значения адреса P. Для A значение между 0 и 120 градусами, при других значениях появится сигнал тревоги 062.

Адрес D может быть указан в пределах от –99999999 до 999999999 с минимальным приращением указанным в соответствии с битом 0 параметра 3401. Если адрес D содержит десятичную точку, появится сигнал тревоги 007.

Допустимые значения скорости подачи указаны в 2.17.2.

2.17.6 Фиксированные циклы сверления

Цикл сверления
G81X_C_Z_F_L_; или G82X_C_Z_R_F_L_;
R Расстояние от исходного уровня до позиции R
P Время задержки на дне отверстия
F Скорость обработки
L Число повторов

Цикл многопроходного сверления
G81X_C_Z_R_Q_P_F_L_;
R Расстояние от исходного уровня до позиции R
Q Длина каждого прохода
P Время задержки на дне отверстия
F Скорость обработки
L Число повторов

Цикл высокоскоростного многопроходного сверления
G83X_C_Z_R_Q_P_F_L_;
R Расстояние от исходного уровня до позиции R
Q Длина каждого прохода
P Время задержки на дне отверстия
F Скорость обработки
L Число повторов

Нарезание резьбы
G84X_C_Z_R_P_F_L_;
R Расстояние от исходного уровня до позиции R
P Время задержки на дне отверстия
F Скорость обработки
L Число повторов

Жесткое нарезание резьбы
G84.2X_C_Z_R_P_F_L_S_;
R Расстояние от исходного уровня до позиции R
P Время задержки на дне отверстия
F Скорость обработки
L Число повторов
S Скорость шпинделя

Цикл расточки
G84.2X_C_Z_R_F_L_; или G89X_C_Z_R_P_F_L_;
R Расстояние от исходного уровня до позиции R
P Время задержки на дне отверстия
F Скорость обработки
L Число повторов

Отмена
G80;

Пояснения

Адреса
Для данного ленточного формата, адрес К используются для определения числа повторов. В формате Series 15 этот адрес L.

G код

Некоторые G коды пригодны только для ленточного формата ЧПУ или Series 15. Указание неправильного G кода приведет к появлению сигнала тревоги 10.

G коды применяемы только в Series 15	G81, G82, G83.1, G84.2
G коды применяемы только в Series 16/18/160/180	G87, G88

Плоскость позиционирования и ось сверления

Для данного ленточного формата, ось позиционирования и ось сверления определяются G кодом для фиксированного цикла.

Для ленточного формата Series 15, плоскость позиционирования и ось сверления определяются в соответствии с G17/G19.

Ось сверления в основной плоскости (Z, X) не находится в плоскости позиционирования.

G код	Плоскость позиционирования	Ось сверления
G17	XY	Z
G19	YZ	X

Сброс бита 1 (FXY) параметра 5101 включает фиксированную ось сверления Z.

Подробности данных определяющих обработку

Данные фиксированного цикла определяются следующим образом:

Gxx X C Z R Q P F L ;

где

Gxx	Режим сверления
X C	Данные позиции
Z R Q P F	Данные сверления
L ;	Число повторов

Адрес	Описание
Gxx	Код цикла сверления
X/U (Z/W) C/H	Относительные или абсолютные значения позиции отверстия
Z/W (X/U)	Относительные или абсолютные значения расстояния позиции R до дна отверстия
R	Относительные или абсолютные значения используемые для описания расстояния исходного уровня. Используется в соответствии со значением бита 6 параметра 5102 и системы G кода.
Q	Относительное значения глубины прохода
P	Время задержки на дне отверстия
F	Скорость подачи обработки
L	Число повторов

Определение позиции R

Позиция К указывается в относительных координатах расстояния до исходного уровня. Для ленточного формата Series 15, параметр и система G кода используется для определения относительного или абсолютного значения.

Если бит 6 параметра 5102 установлен в 0, всегда используется относительное значения. Если 1, значение зависит от используемой системы G кода. Если используется G код системы A, указывается абсолютное значение. Если B и C, в режиме G90 указывается абсолютное значение, и относительное в режиме G91.

<table>
<tr><td colspan="4">Ленточный формат Series 15</td><td>Series 16/18/ 160/180</td></tr>
<tr><td colspan="3">Бит 6 параметра 5102 = 1</td><td>Бит 6 = 0</td><td rowspan="5">Относительное значение</td></tr>
<tr><td colspan="3">Система G код</td><td rowspan="4">Относительное значение</td></tr>
<tr><td>A</td><td colspan="2">B, C</td></tr>
<tr><td rowspan="2">Абсолютное значение</td><td>G90</td><td>G91</td></tr>
<tr><td>Абсолютное значение</td><td>Относительное значение</td></tr>
</table>

Описание фиксированных циклов

Соответствие между G кодами и ленточным форматом Series 15 приведено ниже.

1. G81 (Цикл сверления) G83 (G87) P0 <Q не указывается>
   Нет задержки
2. G82 (Цикл сверления) G83 (G87) P <Q не указывается>
   Инструмент всегда задерживается на дне отверстия
3. G83 (многопроходное сверление) G83 (G87) P <Тип B>
   Если блок содержит P команду, инструмент задерживается на дне отверстия.
4. G83.1 (многопроходное сверление) G83 (G87) P <Тип A>
   Если блок содержит P команду, инструмент задерживается на дне отверстия.
   Помните, что тип A или B выбирается в соответствии с битом 2 параметра 5101.
5. G84 (Нарезание резьбы метчиком) G84 (G88) I
   Если блок содержит P команду, инструмент задерживается на дне отверстия и после возвращения в R.
6. G84.2 (Жесткое нарезание резьбы метчиком) M29 S_ G84 (G88)
   Если блок содержит P команду, инструмент задерживается перед включением шпинделя в обратном направлении на дне отверстия и перед включением в R.
7. G85 (Расточка) G85 (G89) P0
   Нет задержки.
8. G89 (Расточка) G85 (G89) P_
   Инструмент всегда задерживается на дне отверстия.

Расстояние припуска d для G83

Параметр 5114 определяет величину припуска d для G83 и G83.1.

Задержка в G83 и G83.1

Для Series 15-T, G83 или G83.1 не создаются задержку инструмента. Для ленточного формата Series 15, инструмент задерживается на дне отверстия только если указан адрес P.

Задержка в G84 и G84.2

Для Series 15-T, G84 или G84.2 создают задержку инструмента перед включением шпинделя или изменением направления вращения. Для ленточного формата Series 15, инструмент задерживается на дне отверстия и в позиции R только если указан адрес P.

Жесткое нарезание резьбы

Для ленточного формата Series 15 жесткое нарезание резьбы может быть задано следующим образом:

Формат	Значение
G84.2 X_Z_R_…S****;	Установка (F10/F11) = 1
S****; G84.2X_Z_R_…;	
M29 S****; G84X_Z_R_…;	* Общее для Series 16
M29 S**** G84X_Z_R_…;	
G84X_Z_R_…S****;	G84 создает G код жесткого нарезания резьбы. Бит 0 (G84) параметра 5200 = 1 * Общее для Series 16
S**** G84X_Z_R_…;	

Программирование диаметра или радиуса

Установка бита 7 параметра 5102 в 1 приведет к указанию R в фиксированных циклах с обозначением радиуса или диаметра в ленточном формате Series 15.

Отключение формата Series 15

Указание бита 3 параметра 5102 отключает формат Series 15. Это касается фиксированных циклов сверления. Однако, число повторов может быть указано в адресе L.

ВНИМАНИЕ

Установка бита 3 параметра 5102 в 1 отключает биты 6 и 7 параметр 5102, оба значения считаются равными 0.

Ограничения

Ось C как ось сверления

Ось C невозможно использовать как ось сверления. Указание плоскости G18 приведет к сигналу тревоги 28.

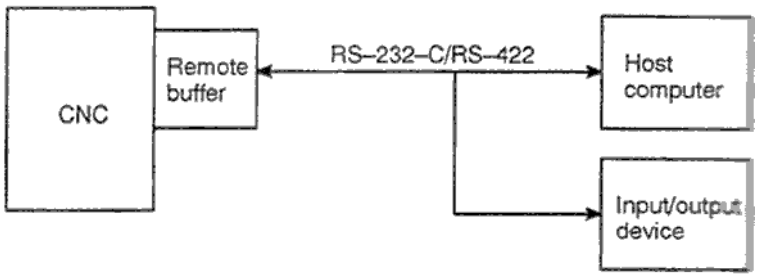
Зажатие оси C

Для формата Series 15 невозможно указание M кода зажатия оси C.

2.18 Функции высокоскоростной обработки

2.18.1 Буфер управления

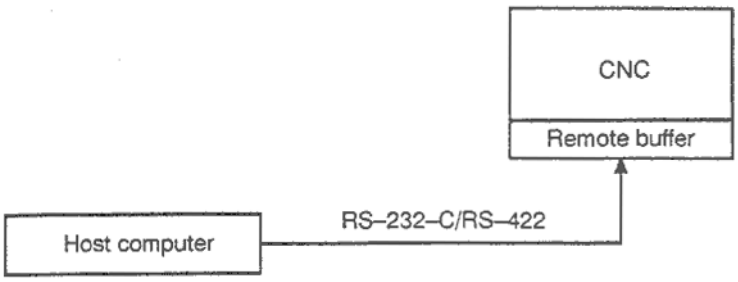
Для возможности передачи больших объемов данных при высокоскоростной обработке, используется буфер управления. Буфер находится между ЧПУ и ведущим компьютером (хост) на последовательном интерфейсе.



Использование буфера при онлайн подключении к ведущему компьютеру позволяет производить высокоскоростную обработку DNC с большой степенью гибкости.
Буфер управления включает высокоскоростной буфер А, который позволяет производить высокоскоростную обработку с использованием двоичных данных.
Подробное описание смотри в описании Буфера управления (B-61802E-1).

2.18.2 Высокоскоростной буфер A (G05)

После программирования блока содержащего только G05, операция может быть выполнения с помощью указания данных в формате описанном далее. После указания нулевого перемещения для каждой оси, восстанавливается режим команд ЧПУ.

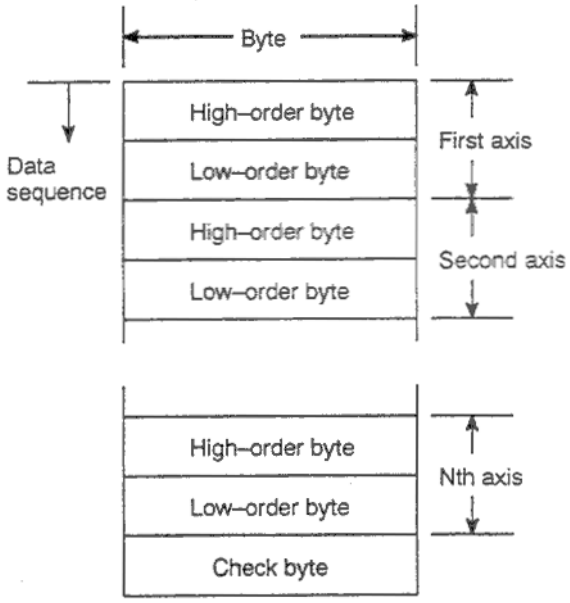


Формат

Включение режима двоичных данных: G05

Выключение режима двоичных данных: нулевое перемещение каждой оси

Формат двоичных данных обработки:



Data sequence – Последовательность данных
High-order byte – Старший порядок байта
First axis – Первая ось
Second axis – Вторая ось
Low-order byte – Низший порядок байта
Check byte – байт проверки

Данные двоичного формата описывают перемещения на единицу времени (2 байта) для каждой оси указываемой последовательно от первой, до N-й, после чего указывается байт проверки (длина данных отдельного блока [2 x N +1] байт).

Все данные в двоичном формате.

Пояснения

Выбор единицы времени
Единица времени может быть указана с помощью битов 4, 5 и 6 (IT0, IT1 и IT2) параметра 7501.

Величина перемещения
Для каждой оси может быть задана величина перемещения в следующих единицах (укажите две составляющие для отрицательного значения).

	IS-B	IS-C	
Миллиметровый станок	0.001	0.0001	мм
Дюймовый станок	0.0001	0.00001	дюйм

Формат данных следующий. Используйте биты указанные * для определения перемещения за единицу времени.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	0	*	*	*	*	*	*	*	0

Пример: Если величина перемещения 700 мкм, (миллиметровый станок, IS-B).

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	1	0	1	0	0	1	1	1	1	0	0	0

Байт проверки
Байт проверки должен содержать сумму [2 x N] байт, не включая байт проверки, в единицах байта, без учета переполнения восьми бит.

Скорость передачи
ЧПУ передает данные [2 x N + 1] байт (N – число осей) из буфера управления каждую единицу времени указанную в параметрах. Для проверки, что ЧПУ продолжает обработку без остановки, минимальная скорость передачи данных между ведущим хост компьютером и буфером следующая:
(2 x N + 1) x 11/T x 1000 (T: единица времени)

Компенсация радиуса кромки инструмента
Если G05 указано в режиме компенсации кромки инструмента, появится сигнал тревоги 178.

Остановка подачи и блокировка
Остановка подачи и блокировка могут использоваться в режиме двоичного управления.

Зеркальное отображение
Зеркальное отображение не может быть включено или выключено в режиме G05.

Тип ускорения/торможения
В режиме обработки, при ускорении и торможении используется экспоненциальное ускорение/торможение (с временной константой установленной в параметре 1622).

Ограничения

Модальные команды
В режиме двоичного ввода данных, выполняется только линейная интерполяция.

Некорректные функции
Поблочный режим, потенциометр подачи, и ограничение максимальной скорости подачи не могут выполняться в режиме двоичного ввода данных. Функция перезапуска программы не может выполняться. Дополнительные функции также не выполняются.

Регистрация в памяти
Программа не может быть зарегистрирована в памяти в режиме двоичного ввода данных.

2.18.3 Распределенный мониторинг высокоскоростной обработки

Во время высокоскоростной обработки, производится распределенный мониторинг состояния обработки. При завершении распределенного мониторинга в конце передачи команд высокоскоростной обработки, появится сигнал тревоги 000 и 179 (в соответствии с настройками бита 7 параметра 7501).

Сигнал тревоги может быть удален только выключением ЧПУ.

Пояснения

Команда высокоскоростной обработки

Высокоскоростная обработка требует использования функции высокоскоростного буфера A, буфера B и функции высокоскоростной обработки G05.

Завершение распределенной обработки

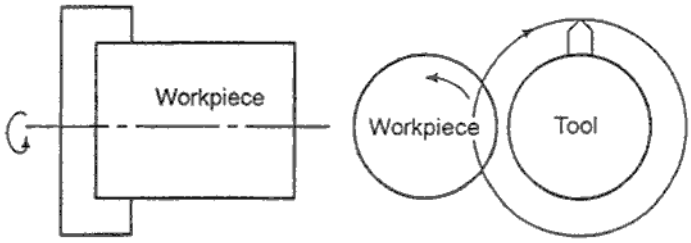
Если высокоскоростная обработка не получает вовремя данные обработки, появятся сигналы тревоги:

Номер	Сообщение	
000	PLEASE TURN OFF POWER	Во время выполнения высокоскоростной обработки, процесс распределенной обработки был завершен. Параметры: Скорость передачи данный (параметр 133) Число контролируемых осей при высокоскоростной обработке (параметр 7150) Выбор оси при высокоскоростной обработке (бит 0 параметра 7510).
179	PARAM. (NO. 7510) SETTING ERROR	

2.19 Функция контроля оси

2.19.1 Вращение многоугольника

Вращение многоугольника обозначает обработку фигуры многоугольника вращением детали и инструмента.



Workpiece – Деталь
Tool - Инструмента
Рис.19.1(a) Вращение многоугольника

Изменяя значение вращения детали и инструмента и числа углов, обрабатываемая фигура может быть квадратом или пятиугольником. Время обработки может быть уменьшено в зависимости от того по какой оси X или Z обрабатывается фигура многоугольника. В основном обработка фигуры многоугольника используются для создания шестигранных болтов или гаек.

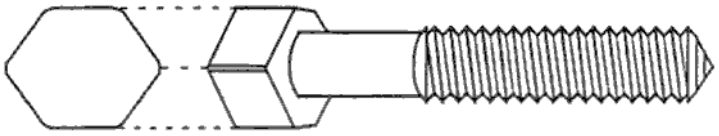


Рис. 19.1(b) Шестигранный болт

Формат
G51.2 (G251) P_Q_;
P, Q: Значение вращения шпинделя и оси Y
Диапазон значений: целое число от 1 до 9 для P и Q.
Если Q положительное число, ось Y делает вращение в положительном направлении, и наоборот.

Пояснения

Вращения многоугольника контролируется ЧПУ. Ось вращения инструмента называется осью Y. Ось Y контролируется командой G51.2 так, что скорость вращения детали и инструмента имеют связь.
Пример: Отношение вращения детали к оси Y равно 1:2, и ось Y имеет положительное вращение.
G51.2P1Q2;
Если указан одновременный запуск G51.2, посылается сигнал одного вращения, ось Y вращается в соответствии с отношением вращения P:Q с синхронизацией скорости шпинделя. Ось Y вращается так, что шпиндель и ось Y находятся в отношении P:Q. Это отношение будет сохраняться пока не будет дана команда отмены вращения многоугольника (G50.2 или сброс). Направление вращение оси Y определяется кодом.
Q не влияет на направление вращения.
Синхронизация шпинделя и оси Y отменяется следующей командой:
G50.2 (G250);

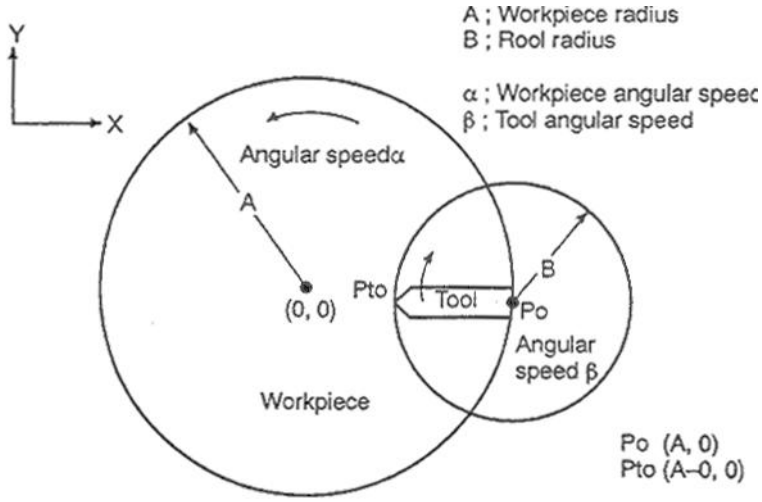
Если указано G50.2, синхронизация шпинделя и оси Y отменяется, и ось Y останавливается
Синхронизация также отменяется следующими случаями:
1) Выключение питания
2) Аварийная остановка
3) Сигнал тревоги серво механизма
4) Сброс
5 Появления сигнала тревоги 217 или 221.

Пример

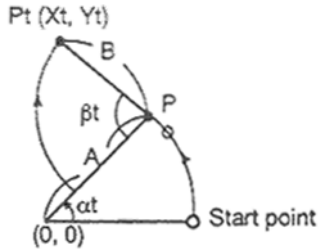
G00 X100.0 Z20.0 S1000.0 M03;	Скорость вращения детали 100 об/мин
G51.2 P1 Q2;	Начало вращения инструмента (2000 об/мин)
G01 X80.0 F10.0;	Подачи оси X
G04 X2;	
G00 X100.0;	Отвод оси X
G50.2;	Остановка инструмента (всегда отдельно)
M05;	Остановка шпинделя

Принципы вращения многоугольника

Принцип вращения многоугольника описан далее: на следующем рисунке радиус инструмента и детали A и B, и угловая скорость инструмента и детали a и b. Начало координат XY находится в центре детали.
Если просто говорить, считайте что центр инструмента находится в точке Po (A, 0) на краю детали, и кромка инструмента начинается обработку в Pto(A-B, 0).



В этом случае, кромка инструмента Pt через время t выражается уравнением 1:



$Xt=A\cos\alpha t-B\cos(\beta-\alpha)t$
(Уравнение 1)
$Yt=A\sin\alpha t+B\sin(\beta-\alpha)t$

Предполагая что отношение вращение 1:2, $\beta=2\alpha$, уравнение 1 получается следующим:

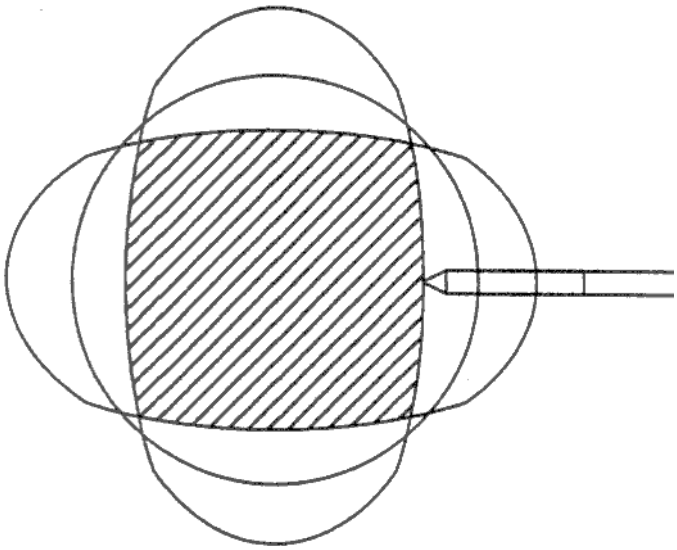
$Xt=A\cos\alpha t-B\cos\alpha t=(A-B)\cos\alpha t$

Уравнение 2

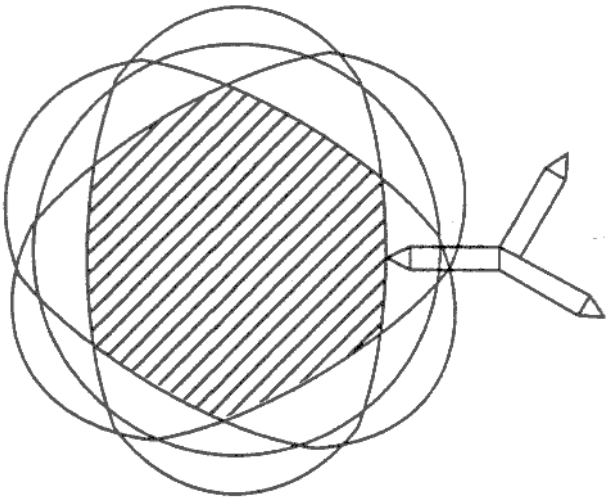
$Xt=A\sin\alpha t+B\sin\alpha t=(A+B)\sin\alpha t$

Уравнение 2 обозначает что кромка инструмент перемещается по эллипсу с большим диаметром A+B и меньшим диаметром A-B.

Квадрат может быть обработан так как показано далее:



Если три инструменты установлены каждый в 120 градусов, будет обработан шестиугольник.



ВНИМАНИЕ

1. Начальная точка резьбы находится вне зависимости от операции синхронизации. Отмените синхронизацию с помощью G50.2 при нарезании резьбы.
2. Следующие сигналы могут быть применимы или неприменимы относительно оси Y при синхронной операции.
   Применимые сигналы:
   - блокировка станка
   - отключение серво подачи

   Неприменимые сигналы
   - остановка подачи
   - блокировка (interlock)
   - потенциометры
   - тестовый запуск
   - (при тестовом запуске, нет сигнала ожидания за оборот в блоке G51.2).

ПРИМЕЧАНИЕ

1. Ось Y в отличие от других контролируемых осей, не может быть указана командой перемещения как Y----. Это команда перемещения необязательна для оси Y Поэтому, если запрограммировано G51.2, необходим только контроль отношения к вращению шпинделя.
   Для оси Y может быть указана только команда перехода в точку ссылки (G28V0;) во время остановки в нестабильной позиции G50.2. Если вращение инструмента запущено из нестабильного положения, возможно возникновение проблемы.
   Указание G28V0; для оси Y равнозначно команде ориентации шпинделя. Для других осей команда ручного перехода в точку ссылки G28 обычно означает перемещения без определения предела торможения. Для оси Y возврат в точку ссылки осуществляется с определением предела торможения, как и при ручном переходе.
   Для обработки такой же фигуры, как и предыдущая, инструмент и шпиндель должны находиться в той же позиции при запуске обработки предыдущей фигуры. Инструмент начинает вращение после получения сигнала оборота шпинделя.
2. Ось Y используется для управления вращения инструмента для вращения многоугольника с использованием 4-й оси. Однако, настройкой параметра 7610, также может использоваться 3-я ось. В этом случае, эта ось должна называться осью C.
3. Помимо отображения позиции оси Y, при перемещении оси Y на экране будет отображаться перемещение оси в пределах от 0 до значения параметра.
   Относительные и абсолютные значения координат не обновляются.
4. Детектор абсолютной позиции не может быть установлен для оси Y.
5. Ручная продолжительная подача или подача от маховика невозможна при синхронном перемещении оси Y.
6. Синхронная операция оси Y не включена в число одновременно управляемых осей.

2.19.2 Вращение поворотной оси

Функция обращения предотвращает переполнение адреса поворотной оси. Эта функция включается битом 0 параметра 1008 в 1.

Пояснение

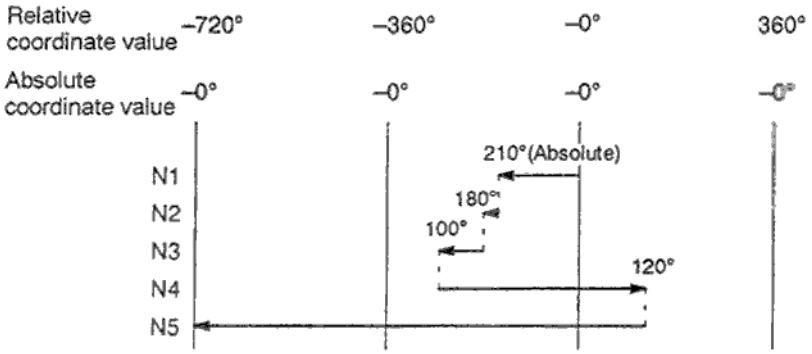
Для относительной команды, инструмент перемещается на указанный угол. Для абсолютной команды, координата в которую перемещается инструмента определяется параметром 1260 и округляется до значения одного оборота. Если бит 1 параметра 1008 установлен в 0, инструмент перемещается в кратчайшем направлении. Отображаемые значения относительных координат также округляются в зависимости от бита 2 параметра 1008.

Примеры

Допусти ось C является осью вращения и значение перемещения на один оборот равно 360.00 (параметр 1260 = 360000). При выполнении следующей программы ось будет перемещаться следующим образом:

C0;

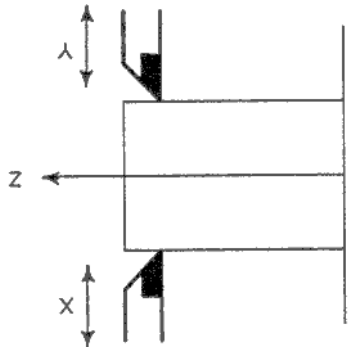
N1 C-150.0;	Значение перемещения: -150	Абсолютная координата: 210
N2 C540.0;	Значение перемещения: -30	Абсолютная координата: 180
N3 C-620.0;	Значение перемещения: -80	Абсолютная координата: 100
N4 H380.0;	Значение перемещения: +380	Абсолютная координата: 120
N5 H-840.0;	Значение перемещения: -840	Абсолютная координата: 0



Relative coordinate – Относительная координата

2.19.3 Простое управление синхронизации

Функция простого управления синхронизации позволяет синхронизировать нормальную операцию двух осей в соответствии с поступающими сигналами станка.
Для станков с двумя креплениями инструмента, которые могут управляться раздельно, эта функция позволяет использовать функцию описываемую далее.
Эта глава описывает работу станка с двумя креплениями инструмента, которые приводятся вы действие независимо и работают в осях X и Y. Если ваш станок использует другие оси для тех же целей, замените эти оси на X и Y.



Пояснения

Функция синхронизации позволяет использовать станок с двумя независимыми креплениями инструмент. В режиме синхронного выполнения, движения каждой оси могут быть синхронизированы с движением другой оси. Команда перемещения может быть указан для одной или для двух осей, который считаются основными осями. Другие оси синхронизируемые с основной осью называются ведомыми. Если основная ось X и ведомая ось Y, синхронная работа осей X и Y выполняется в соответствии с командой Xxxxx подаваемой для основной оси.
В режиме синхронной работы, команда перемещения указанная для основной оси приводит к одновременному перемещения сервомотора основной оси и ведомой.
В этом режиме компенсация ошибки синхронизации не производится. Мониторинг несоответствия позиции между двумя осями отсутствует. Сигнала тревоги ошибки синхронизации нет. Автоматическая работа может быть синхронизирована, ручная нет.

Нормальный режим

Нормальным режимом является работа когда разные детали обрабатываются на разных столах. ЧПУ дает команды перемещения для основной и ведомой оси в соответствии с запрограммированными значениями. Команды перемещения могут быть заданы в идентичных блоках.
1. В нормальном режиме команда Xxxxx приводит к перемещению оси X.
2. Команда Yyyyy приводит к перемещению оси Y.
3. Командой Xxxxx и Yyyyy выполняется одновременное перемещение осей X и Y.

Переключение между нормальным и синхронным режимом

Подробное описание приведено в документации производителя станка.

Автоматический переход в точку ссылки

Если дана команда автоматического перехода в точку ссылки G28 или G30 в синхронном режиме, переход в точку ссылки производится по оси X с идентичным движением оси Y. После достижения осью Y точки ссылки, загорается лампа сигнализирующая о переходе в точку ссылки.
Рекомендуется давать команду G28 и G30 в нормальном режиме.

Проверка перехода в точку ссылки

Если в синхронном режиме запрограммирована команда проверки перехода в точку ссылки G27, производится идентичное движение осей X и Y.
Если обе оси достигли точки ссылки, загорается лампа сигнализирующая о выполнении операции. В противном случае появится сигнал тревоги.
Рекомендуется давать команду G27 в нормальном режиме.

Команда ведомой оси

Если в синхронном режиме дать команду перемещения ведомой оси, появится сигнал тревоги 213.

Основная и ведомая ось
Основная ось определяется параметром 8311. Ведомая ось определяется внешним сигналом.

Ограничения

Установка системы координат и компенсации инструмента
Если синхронный режим включен при существующем сдвиге системы координат в результате установки системы координат или компенсации инструмента, появится сигнал тревоги 214.

Внешнее торможение, блокировка interlock, блокировка станка
В синхронном режиме сигнал внешнего торможения, блокировка interlock, или блокировка станка только для основной оси. Соответствующий сигнал для ведомой оси игнорируется.

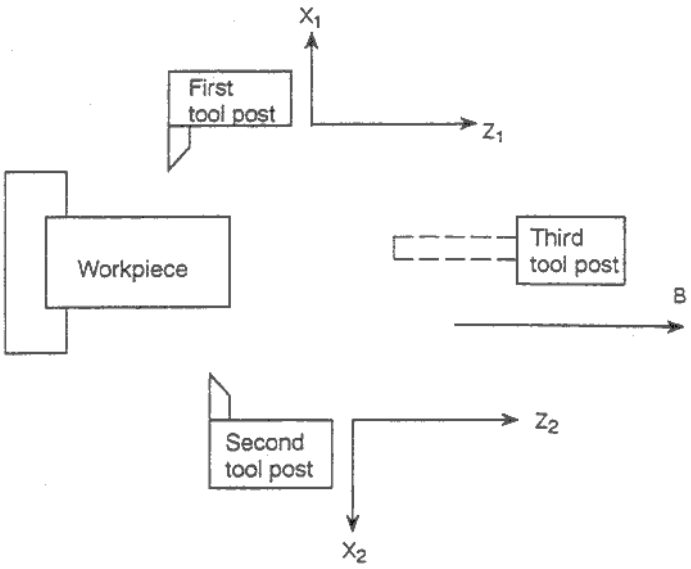
Компенсация ошибки шага
Компенсация ошибки шага или обратная компенсация выполняются отдельно для основной и ведомой оси.

Абсолютный датчик
В синхронном режиме ручной абсолютный датчик должен быть включен (ABS установлено в 1). Если датчик выключен, возможно некорректное перемещение ведомой оси.

Ручной режим
Ручная работа не может быть синхронизирована.

2.19.4 Управление оси B (G100, G101, G102, G103, G110)

Эта функция позволяет устанавливать ось B независимо от основных осей X_1, Z_1, X_2 и Z_2 и позволяет производить сверление, расточку и другую обработку вдоль оси B параллельно основной оси. X_2 и Z_2 могут быть использованы для двухстороннего режима управления.



Tool post – Крепление инструмента

Формат

Регистрация программ

G101-G100; Начало регистрации первой программы
G102-G100; Начало регистрации второй программы
G103-G100; Начало регистрации третьей программы
G100; Завершение регистрации

Для оси могут быть зарегистрированы три операции (программы (в двухстороннем режиме возможно три программы для каждого инструмента).
Программа оси B должна быть указана в блоках между G101, G102, G103 и G100 позволяя отделять основную программу ЧПУ.
Зарегистрированная операция начинается сразу после выполнения соответствующего кода, описанного ниже.

```
O1234;
..          Обычная программа ЧПУ
G101;       Начало программы оси B
..          Программа оси B
G100;
..
M30;
```

Помните, между блоками G101, G102, G103, G100 не должно быть посторонних кодов.

Команда используемая для старта операции
Для запуска операции, в параметрах 8251 – 8253 указывается дополнительная М-функция. Параметр 8251 используется для определения М кода первой программы, 8253 третьей программы.

```
O1234;
..
M**;            Начало выполнения зарегистрированной операции
..
M30;
```

Пример:

```
O1234;
G50 X100. Z200.;
G101;                   Регистрация первой программы
G00 B10.;               \
M03;                    |
G04 P2500;              > Блоки программы
G81 B20. R15. F500;     |
G28;                    /
G100;                   Завершение регистрации программы
G00 X80. Z50.;
G01 X45. F1000;
..
G00 X10.;
M**;                    Команда для запуска операции
G01 Z30. F300;
..
M30;
```

При выполнении зарегистрированных программ блоки программы и нормальная программа ЧПУ выполняются одновременно.

Работа с одним движением
G110[команда операции];
Этой командой может быть определена работа с одним движением для оси В. Такую операцию не нужно регистрировать.

Пояснения

Определение двухстороннего режима управления
Один из следующих трех режим может быть выбран для двухстороннего управления:
1. Управление оси В выполняется либо для крепления инструмента 1, либо 2.
2. Управление оси В выполняется раздельно для крепления инструмента 1 или 2.
3. Идентичное управления оси В для крепления инструмента 1 и 2.

Режим выбирается значением параметра 8250 для каждого крепления инструмента.

Коды используемые в программе управления оси B

Следующие 13 G кодов и M, S, T коды могут быть использованы для управления программой оси B:

Код	Описание
G00	Быстрое позиционирование
G01	Линейная интерполяция
G04	Задержка
G28	Переход в точку ссылки
G80	Отмена фиксированного цикла
G81	Сверление
G82	Расточка
G83	Многопроходное сверление
G84	Нарезание резьбы метчиком
G85	Расточка
G86	Расточка
G98	Подача в минуту
G99	Подача на оборот
M**	Вспомогательная функция
S**	Вспомогательная функция
T**	Вспомогательная функция, офсет инструмента

G28 (переход в точку ссылки)

В отличие от нормального цикла G28, цикл G28 для оси B не совершает немедленный переход. Например, следующая команда не может быть выполнена.

G28 B99.9;

G80 – G86 (фиксированные циклы сверления)

Для фиксированных циклов поддерживаемых FANUC Series 16 или Series 18, могут быть выполнены эквиваленты команд G80 – G86.

Данные указываются так же, как и для Series 16, 18, за исключением следующих отличий:

1. Позиция сверления не указывается в X и Y.
2. Расстояние от точки R до дна отверстия указывается в B.
3. Все операции выполняются в режима возврата в исходную плоскость.
4. Число повторов K не может быть указано.
5. В режиме фиксированного цикла, точка R должна быть указана. (в противном случае сигнал тревоги 5036).
6. Точка начала сверления d для G83 указывается с параметром 8258.

G98, G99 (Подача в минуту, подача на оборот)

Бит 2 параметра 8241 определяет исходное состояние G кода для G110, или G код начала регистрации программы (G101, G102, G103).

Если бит установлен в 0, исходное состояние G98.

Если бит установлен в 1, исходное состояние G90.

Пример

Если бит установлен в 0

G110 B100. F1000.; 1000 мм/мин

G110 G99 B100. F1; 1 мм/об

ПРИМЕЧАНИЕ

В режиме двухстороннего управления, система использует текущую скорость шпинделя, вычисляемую из сигнала позиции кодировщика установленного на креплении инструмента.

M, S, T коды

В соответствии с передаваемым кодом в адресах M, S, T передается двоичный код и прямой сигнал в станок. Коды и сигналы для адресов M, S, T все передаются по одному интерфейсу и могут быть использованы для включения и выключения станка. Для этого используется интерфейс контроля осей PMC, который различает дополнительные функции от программы ЧПУ. Следующие M коды используются для контроля шпинделя, и автоматически передаются при G84 или G86: M03, M04, M05.

T** - T(** + 9), где ** номер указанный в параметре 8257 используется для кода вспомогательной функции для регулировки офсета инструмента.

Пример:

T50 – T59 если 8257 установлен в 50

1. M, S, T код не должен быть указан в блоке содержащем другие команды перемещения. Эти коды не должны быть указаны вы идентичных блоках.

2. Обычно, нормальное управление программой ЧПУ и операция оси B работают независимо друг от друга. Синхронизация между операциями может быть установлена с помощью дополнительных функций нормальной программы ЧПУ и программы операции оси B.

Нормальная операция ЧПУ; Программа оси B

```
M11;          G00 B111;
G01 X999;     G01 B222;
G28 Z777;     G28;
M50;          M50;
G00 X666;     G81 B444 R111 F222;
..
```

При получении M50 для нормальной программы ЧПУ и программы оси B, PMC дает сигнал FIN завершения обоих вспомогательных функций. G00 X666 нормальной программы ЧПУ и G81 B444 R111 F222 оси B выполняются одновременно.

Произвольный макрос

Переменные произвольного макроса (локальные, общие, системные переменные) могут быть использованы в операция между G101, G102, G103, G100.

1. Значения макропеременных вычисляются не из существующих данных при работе оси B, а из существующих данных в момент регистрации программы.
2. Инструкция вызывающая переход в место вне программы G101, G102, G103, G100 будет обработана без проверки.
3. В режиме двухстороннего управления, крепления инструмента 1 и 2 имеют различные макропеременные.

Программа управления

Если зарегистрирована новая программа, предыдущая программа автоматически удаляется.

Если определена ошибка в программы, программа будет инициализирована, но не зарегистрирована.

Модальность

Также как и обычная программа ЧПУ, программа си B может использовать модальные данные: модальные G коды, F коды, и P, Q, F коды в фиксированном цикле. Эти коды не влияют на модальную информацию программы ЧПУ. Если программа оси B запущена, для программы устанавливаются исходные модальные данные. На них влияет предыдущая модальная информация.

Пример

```
G01 X10. F1000;          1
G101 (G102, G103);       2
B10.;                    3
G01 B-10. F500;          4
G100;                    5
X-10.;                   6
..
```

Независимо от модальной информации нормальной программы, блок 3 определяет G00 если бит 1 параметра 8241 установлен в 0, или G01 если бит установлен в 1.

Блок 6 вызывает перемещение F1000, указанное в блоке 1.

Команда запуска операции

Бит 7 параметра 8240 описывает метод используемый для запуска операции оси B следующим образом:

Если бит установлен в 1, операция оси B начинается M кодом.

Если бит установлен в 0, операция оси B начинается если выполнен M код операции оси и PMC выдал сигнал завершения операции FIN.

Для запуска программы может быть сохранено до пяти M кодов (в режиме двухстороннего управления до пяти кодов для каждого крепления инструмента).

Пример

Если первая, вторая и третья программы запускаются кодами M40, M41 и M42

```
O1234;
..
..
M40; M коды запуска программ
M41;
M42;
M40;
M41;
..
..
M30;
```

M41 начинает выполнение после завершения программы M40.

M42, M40 и M41 указанные во время выполнения первой программы сохраняются так, что соответствующая программы выполняется в том порядке, в котором они определены M кодом.

Если указано 6 или более M кодов программы управления осью B во время выполнения первой программы, появится сигнал тревоги 5038.

В двухстороннем режиме управления M код запуска программы для инструмента 1 запускает зарегистрированную программу 1. M код запуска программы для инструмента 2 запускает зарегистрированную программу 2.

Указание абсолютного или относительного режима

Величина перемещения оси B может быть указана в абсолютных или относительных координатах. В абсолютных координатах конечная точка указывает координату перемещения оси. В относительных координатах расстояние перемещения программируется напрямую.

Бит 6 (ABS) параметра 8240 используется для установки абсолютного или относительного режима. Бит установленный 1 задает абсолютный режим, в 0 относительный режим. Режим указанный вы этом параметре определяется в момент регистрации программы.

Определение офсета инструмента

Команда T** сдвигает конечную точку инструмента по оси B в положительном или отрицательном направлении на величину офсета оси B. Если эта функция используется для определения разницы между запрограммированной позицией инструмента и действительной позицией обработки, программу не нужно изменять для исправления позиции инструмента.

Значение параметра 8257 может отменять офсет. Последняя цифра может определять девять различных офсетов инструмента. Эти значения офсета отображаются на дисплее офсетов оси B. Подробнее см. в разделе «Эксплуатация».

Режим отдельного движения (single-motion)

Если запрограммирован блок G110, ось B переходит в режим отдельного движения оси B. В режиме работы отдельного движения операция перемещения выполняется сразу же если это определено для операции оси B. Если операция указана во время выполнения зарегистрированной программы, операция будет выполнена после завершения программы.

После выполнения операции отдельного движения, выполняется следующий блок.

```
..
G110 G01 B100. F200;        Блок отдельного движения оси B
..
G00 X100. Z20.;
..
```

Память программ

Программа регистрируется в памяти программ как последовательность различных блоков перемещения, задержки, вспомогательных и других функций. Память программ может содержать любое число блоков, тогда как в одной программе может находиться до 65535 блоков. Если программа содержит пустые места, в момент регистрации программы оси B появится сигнал тревоги 5033. Шесть блоков требуют 80 символов памяти программы. Фиксированный цикл (G81 – G86) также регистрируется как последовательность блоков, таких как перемещение и задержка.

Вся память программы имеет резервное питание от батарей. Программы зарегистрированные в памяти программ остаются после выключения питания. После включения питания, работа может быть возобновлена простым указанием M кода запуска программы.

Пример:

```
..
G101;
G00 B10.;                   Один блок
G04 P1500;                  Один блок
G81 B20. T50. F600;         Три блока
G28;                        Один блок
M15;                        Один блок
G100;
..
```

Сброс

Если сброс производится нажатием клавиши сброс на панели MDI или подачей внешнего сигнала сброса, сигнала сброса и перемотки, или аварийной остановки, управление осью B также сбрасывается. Сигнал сброса интерфейса PMC может сбросить только управление оси B. Подробное описание см. в документации производителя станка.

Оси контролируемые PMC

Ось B может работать только со включенным управлением PMC осью B. Подробное описание см. в документации производителя станка.

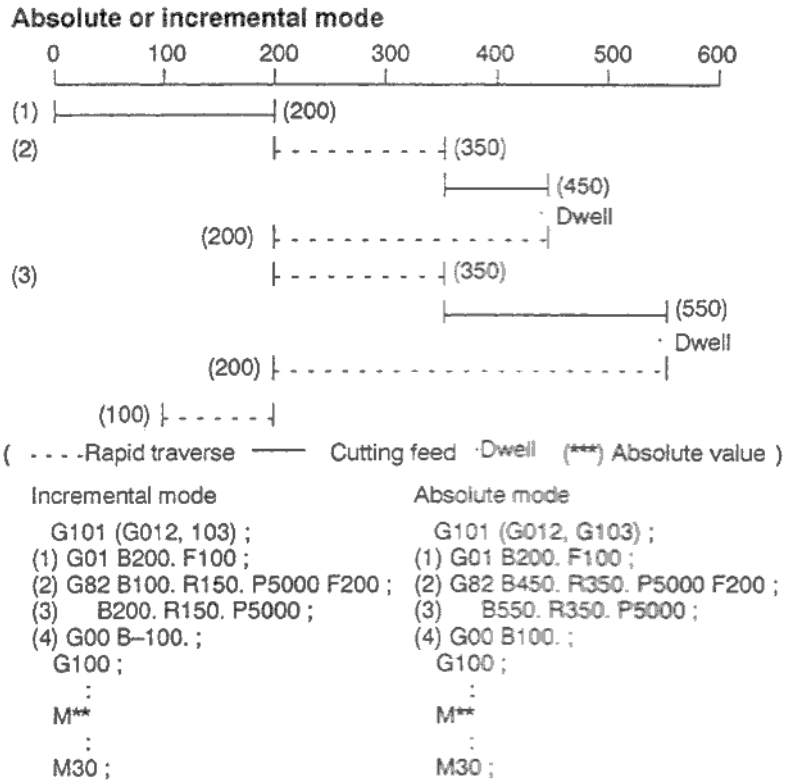
Ограничения

Операция отдельного движения

1. В G110 может быть определена только операция отдельного движения.
   G110 G00 B100.;
   G110 G28; ; Нормальные операции
   G110 G81 B100. R150.0 F100; ; Сигнал тревоги 5034
2. Фиксированный цикл (G81 – G86) и другие операции содержащие многократные движения не могут быть в G110.
3. Модальная информация указанная в G110 не влияет на следующие блоки. В блоке G110 исходные модальные значения указанные в начале, остаются действительными независимо от модальной информации указанной в предыдущих блоках.
   Пример
   Если бит 1 параметра 8241 установлен в 1 и бит 2 параметра 8241 в 1
   G98 G00 X100. F1000; 1
   G110 B200. F2; 2
   X200.; 3
   G01 X200; 4
   Блок 2 устанавливает скорость подачи G01 в 2 мм/ом
   Блок 3 устанавливает быструю подача
   Блок 4 устанавливает скорость подачи G01 в 1000 мм/мин.
4. При компенсации радиуса кромки инструмента два или больше последовательных блока G110 не могут быть определены, появится сигнал тревоги 504. Для этого необходимо зарегистрировать программу для оси B.

Примеры

Абсолютные и относительные координаты



Крепления инструмента 1 и 2

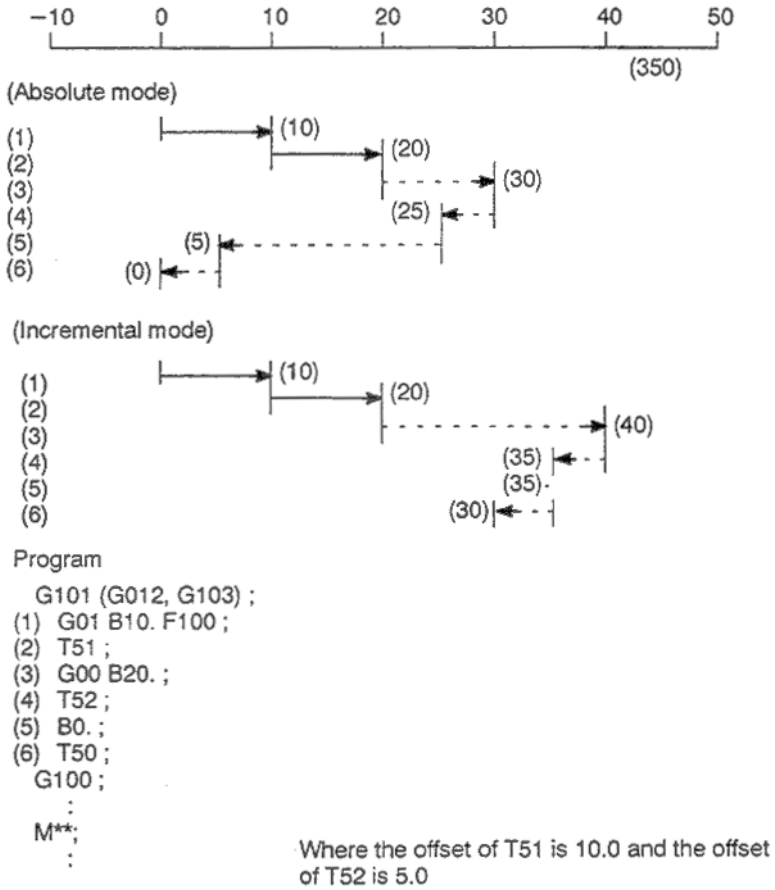
Если одна ось используется для двух креплений инструмента в режиме двустороннего управления, крепления инструмента 1 и 2 разделяют координаты оси B.

Например, после программы 1 для крепления 1 и программы для крепления 2 выполняются в следующем порядке. Общая величина перемещения оси B составляет +100.

Программа 1
G101;
..
G00 B200.; (абсолютный режим)
G100;
..
M30;

G101;
..
G00 B300.; (абсолютный режим)
G100;
..
M30;

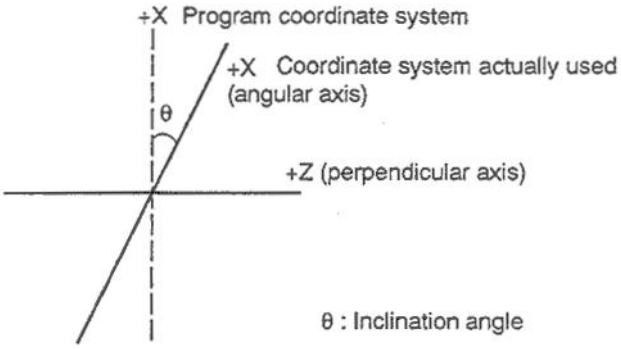
Офсет инструмента
Пример:
Параметр 8257 установлен в 50
Вспомогательная функция используемая для отмены офсета: T50
Вспомогательная функция используемая для установки офсета: T51 – T59



Офсет T51 установлен в 10.0 и офсет T52 в 5.0.

2.19.5 Угловой контроль оси / произвольный угол оси

Если угловая ось создает угол отличный от 90 градусов с перпендикулярной осью, функция контроля угловой оси контролирует расстояние перемещения оси. Ось X всегда используется как угловая ось и Z как перпендикулярная ось. Для контроля угловой оси B могут быть указаны произвольные оси в соответствии со значениями параметров. При создании программы, предполагается что угловая и перпендикулярная ось пересекаются под прямым углом, однако действительное расстояние перемещения контролируется с учетом наклона оси.



Пояснения

Если угловая оси является осью X и перпендикулярная ось – Z, величина перемещения по каждой из осей вычисляет по формулам приведенным ниже.

Расстояние перемещения по оси X определяется следующей формулой:

$Xa = Xp / \cos \theta$

Расстояние перемещения оси Z корректируется наклоном оси X, и определяется следующей формулой:

$Za = Zp – (1/2) * Xp * \tan \theta$

Компонента скорости X определяется следующей формулой:

$Fa = Fp / \cos \theta$

Xa, Za, Fa: Действительные значения

Xp, Zp, Fp: Запрограммированные расстояния и скорость

Метод использования

Угловая и перпендикулярная оси предварительно определяются параметрами 8211 и 8212.

Параметр AAC (8200#0) включает или выключается функцию наклона оси. Если функция включена, расстояние перемещения контролируется в соответствии с углом наклона (8210).

Параметр AZR (8200#2) включает ручной переход в точку ссылки угловой оси.

Если сигнал отключения контроля перпендикулярной/угловой оси NOZAGC установлен в 1, функция контроля применяется только для угловой оси. В этом случае, команда перемещения угловой оси конвертирует угловые координаты. Перпендикулярная ось не влияет на перемещение угловой оси.

Дисплей относительных и абсолютных координат

Относительные и абсолютные координаты отображаются в декартовой прямоугольной системе координат.

Дисплей позиции станка

Отображение позиции станка производится с отображением действительных перемещений инструмента с учетом угла наклона. При использовании дюйм / метрического преобразования, эти расчеты учитываются при отображении результата.

ВНИМАНИЕ

1. После установки значения наклона оси, убедитесь в том что совершено перемещение в точку ссылки.
2. Если бит 2 параметра 8200 установлен в 0, после перехода в точку ссылки угловой оси, также выполните ручной переход в точку ссылки перпендикулярной оси.
3. После перемещения инструмента по угловой и перпендикулярной оси с сигналом NOZAGC установленным в 1, необходим ручной переход в точку ссылки.
4. Перед попыткой ручного перемещения инструмента по угловой и перпендикулярной оси одновременно, установите сигнал контроля осей NOZAGC в 1.

ПРИМЕЧАНИЕ

1. Если угол наклона 0 или 90 градусов, появится сигнал тревоги. Следует использовать угол в пределах от +-20 до +-60 градусов.
2. Перед проверкой перехода перпендикулярной оси в точку ссылки G37, необходимо выполнить переход в точку ссылки угловой оси.

2.20 Функция ввода данных шаблона

Эта функция позволяет пользователю производить простое программирование вводом числовых данных (данных шаблона) с чертежа и указывать числовые значения на панели MDI.
Это позволяет программировать без использования языка ЧПУ.

С помощью этой функции производитель станка может подготовить программы всего производственного цикла (включая расточку или нарезание резьбы) с использованием функции произвольного макроса, и сохранить программу в памяти.
Этот цикл назначается именам шаблона таким как BOR1, TAP3, и DRL2.

Оператор может выбрать шаблон из меню имен шаблонов на экране. Данные шаблона задаваемые пользователем должны соответствовать переменным цикла сверления.
Оператор может идентифицировать эти переменные с помощью имен таких как DEPTH, RETURN RELIEF, FEED, MATERIAL. Оператор назначает значения этим переменным.

2.20.1 Отображение меню шаблонов

Нажатие клавиши OFFSET SETTING и [MENU] отображается на следующем экране:

```
MENU : HOLE PATTERN              O0000 N00000

     1.  BOLT HOLE
     2.  GRID
     3.  LINE ANGLE
     4.  TAPPING
     5.  DRILLING
     6.  BORING
     7.  POCKET
     8.  PECK
     9.  TEST PATRN
    10.  BACK

>_
MDI **** *** ***        16:05:59
[ MACRO ] [ MENU ] [  OPR  ] [      ] [(OPRT)]
```

HOLE PATTERN:

Это заголовок меню. Строка может содержать 12 произвольных символов.

BOLT HOLE

Это название шаблона. Строка может содержать 10 произвольных символов.

Производитель станка должен определить строку заголовка и имя шаблона макроса, и загрузить в память программ как подпрограмму номер 9500.

Макро команда определяющая заголовок меню

Заголовок меню: C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8 C_9 C_{10} C_{11} C_{12}

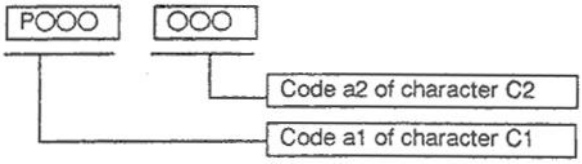
C_1 C_2 - C_{12}: символы меню (12 символов)

Макро команды

G65 H90 Pp Qq Rr Ii Jj Kk

H90 Определяет заголовок меню

p: Считает a1 и a2 кодами символов C1 и C2, тогда



q: Считает a3 и a4 кодами символов C3 и C4, тогда $q = a_3 10^3 + a_4$

r: Считает a5 и a6 кодами символов C5 и C6, тогда $r = a_5 10^3 + a_6$

i: Считает a7 и a8 кодами символов C7 и C8, тогда $i = a_7 10^3 + a_8$

j: Считает a9 и a10 кодами символов C9 и C10, тогда $j = a_9 10^3 + a_{10}$

k: Считает a11 и a12 кодами символов C11 и C12, тогда $k = a_{11} 10^3 + a_{12}$

Пример:

Если заголовок меню “HOLE PATTERN”макро команда будет следующей:

G65 H90 P072079 Q076069 R032080 I065084 J084069 K082078

HO LE _P AT TE RN

Таблицу кодов этих символов см. в 2.20.3.

Макро команда определяющая имя шаблона

Имя шаблона: C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8 C_9 C_{10}

C_1 C_2 - C_{10}: символы шаблона (10 символов)

Макро команды

G65 H91 Pn Qq Rr Ii Jj Kk

H90 Определяет название шаблона

n: Номер шаблона в меню (от 1 до 10)

q: Считает a3 и a4 кодами символов C1 и C2, тогда $q = a_1 10^3 + a_2$

r: Считает a5 и a6 кодами символов C3 и C4, тогда $r = a_3 10^3 + a_4$

i: Считает a7 и a8 кодами символов C5 и C6, тогда $i = a_5 10^3 + a_6$

j: Считает a9 и a10 кодами символов C7 и C8, тогда $j = a_7 10^3 + a_8$

k: Считает a11 и a12 кодами символов C9 и C10, тогда $k = a_9 10^3 + a_{10}$

Пример:

Если шаблон меню 1 “BOLT HOLE” макро команда будет следующей:

G65 H91 P1 Q066079 R076084 I032072 J079076 K069032

BO LT _H OL E_

Выбор номера шаблона

Для выбора номера шаблона, введите соответствующий номер шаблона и нажмите INPUT.

Выбранный шаблон назначается системной переменной #5900. Произвольный макрос выбранного шаблона может быть запущен фиксированной программой (поиск внешней программы) по внешнему сигналу в соответствии с системной переменной #5900.

ПРИМЕЧАНИЕ

Если каждый символ P, Q, R, I, J и K не указаны в макро команде, каждому пропущенному символу назначается пробел.

Пример

Произвольный макрос заголовка меню и имен шаблона.

```
MENU : HOLE PATTERN              O0000 N00000
     1.  BOLT HOLE
     2.  GRID
     3.  LINE ANGLE
     4.  TAPPING
     5.  DRILLING
     6.  BORING
     7.  POCKET
     8.  PECK
     9.  TEST PATRN
    10.  BACK

>_
MDI **** *** ***        16:05:59
[ MACRO ] [ MENU ] [ OPR ] [      ] [ (OPRT) ]
```

```
O9500 ;
N1G65 H90 P072 079 Q076 069 R032 080 I 065 084 J 084 069 K082 078 ;   HOLE PATTERN
N2G65 H91 P1 Q066 079 R076 084 I 032 072 J 079 076 K069 032 ;         1.BOLT HOLE
N3G65 H91 P2 Q071 082 R073 068 ;                                      2.GRID
N4G65 H91 P3 Q076 073 R078 069 I 032 065 J 078071 K076069 ;           3.LINE ANGLE
N5G65 H91 P4 Q084 065 R080 080 I 073 078 J 071 032 ;                  4.TAPPING
N6G65 H91 P5 Q068 082 R073 076 I 076 073 J 078 071 ;                  5.DRILLING
N7G65 H91 P6 Q066079 R082073 I 078 071 ;                              6.BORING
N8G65 H91 P7 Q080 079 R067 075 I 069 084 ;                            7.POCKET
N9G65 H91 P8 Q080069 R067075 ;                                        8.PECK
N10G65 H91 P9 Q084 069 R083 084 I032 080 J065 084 K082 078 ;          9.TEST PATRN
N11G65 H91 P10 Q066 065 R067 0750 ;                                   10.BACK
N12M99 ;
```

2.20.2 Дисплей данных шаблона

После выбора шаблона отображаются данные шаблона.

```
VAR. : BOLT HOLE                    O0001 N00000
NO.  NAME          DATA   COMMENT
500  TOOL          0.000
501  STANDARD X    0.000  *BOLT HOLE
502  STANDARD Y    0.000  CIRCLE*
503  RADIUS        0.000  SET PATTERN
504  S. ANGL       0.000  DATA TO VAR.
505  HOLES NO      0.000  NO.500-505.
506                0.000
507                0.000

ACTUAL POSITION (RELATIVE)
   X    0.000      Y   0.000
   Z    0.000
> _
MDI **** *** ***      16:05:59
[ MACRO ] [ MENU ] [  OPR  ] [     ] [(OPRT)]
```

BOLT HOLE

Это заголовок данных шаблона. Строка символов может содержать до 12 символов.

TOOL

Имя переменной. Строка символов может содержать до 10 символов.

BOLT HOLE CIRCLE

Комментарий. До 8 строк, до 12 символов в строке.

Производитель станка должен определить строку заголовка и имя шаблона макроса, и загрузить в память программ как подпрограмму номер 9500.

Макро команда определяющая заголовок данных шаблона
Заголовок меню: C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8 C_9 C_{10} C_{11} C_{12}
C_1 C_2 - C_{12}: символы меню (12 символов)
Макро команда
G65 H92 Pp Qq Rr Ii Jj Kk
H92 Определяет имя шаблона
p: Считает a1 и a2 кодами символов C1 и C2, тогда $p = a_1 10^3 + a_2$
q: Считает a3 и a4 кодами символов C3 и C4, тогда $q = a_3 10^3 + a_4$
r: Считает a5 и a6 кодами символов C5 и C6, тогда $r = a_5 10^3 + a_6$
i: Считает a7 и a8 кодами символов C7 и C8, тогда $i = a_7 10^3 + a_8$
j: Считает a9 и a10 кодами символов C9 и C10, тогда $j = a_9 10^3 + a_{10}$
k: Считает a11 и a12 кодами символов C11 и C12, тогда $k = a_{11} 10^3 + a_{12}$

Пример:
Если заголовок шаблона “HOLE PATTERN” макро команда будет следующей:
G65 H92 P072079 Q076069 R032080 I065084 J084069 K082078
HO LE _P AT TE RN

Макро команда определяющая имя переменной
Имя шаблона: C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8 C_9 C_{10}
C_1 C_2 - C_{10}: символы названия переменной (10 символов)
Макро команды
G65 H92 Pn Qq Rr Ii Jj Kk
H93 Определяет название переменной
n: Номер шаблона в меню (от 1 до 10)
q: Считает a3 и a4 кодами символов C1 и C2, тогда $q = a_1 10^3 + a_2$
r: Считает a5 и a6 кодами символов C3 и C4, тогда $r = a_3 10^3 + a_4$
i: Считает a7 и a8 кодами символов C5 и C6, тогда $i = a_5 10^3 + a_6$
j: Считает a9 и a10 кодами символов C7 и C8, тогда $j = a_7 10^3 + a_8$
k: Считает a11 и a12 кодами символов C9 и C10, тогда $k = a_9 10^3 + a_{10}$

Пример:
Допустим что имя переменной 503 “BOLT HOLE” макро команда будет следующей:
G65 H91 P503 Q066079 R076084 I032072 J079076 K069032
BO LT _H OL E_

ПРИМЕЧАНИЕ
Имена переменных могут быть назначены 32 общим переменным #500 - #531, которые не удаляются при выключении питания.

Макро команда определяющая комментарий

Комментарий: C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8 C_9 C_{10} C_{11} C_{12}

C_1 C_2 - C_{12}: символы комментария (12 символов)

Макро команда

G65 H94 Pp Qq Rr Ii Jj Kk

H92 Комментарий

p: Считает a1 и a2 кодами символов C1 и C2, тогда $p = a_1 10^3 + a_2$

q: Считает a3 и a4 кодами символов C3 и C4, тогда $q = a_3 10^3 + a_4$

r: Считает a5 и a6 кодами символов C5 и C6, тогда $r = a_5 10^3 + a_6$

i: Считает a7 и a8 кодами символов C7 и C8, тогда $i = a_7 10^3 + a_8$

j: Считает a9 и a10 кодами символов C9 и C10, тогда $j = a_9 10^3 + a_{10}$

k: Считает a11 и a12 кодами символов C11 и C12, тогда $k = a_{11} 10^3 + a_{12}$

Комментарий может быть до 8 строк и 12 символов в каждой строке. Последовательность строк определяется последовательностью макросов для каждой строки.

Пример:

Допустим что нужно вывести комментарий “HOLE PATTERN” макро команда будет следующей:

G65 H94 P072079 Q076069 R032080 I065084 J084069 K082078

HO LE _P AT TE RN

Пример

Макрос описывающий заголовок, имена переменных и комментарий.

```
VAR. : BOLT HOLE                    O0001 N00000
NO.   NAME            DATA   COMMENT
500   TOOL            0.000
501   STANDARD X      0.000  *BOLT HOLE
502   STANDARD Y      0.000  CIRCLE*
503   RADIUS          0.000  SET PATTERN
504   S. ANGL         0.000  DATA TO VAR.
505   HOLES NO        0.000  NO.500-505.
506                   0.000
507                   0.000

ACTUAL POSITION (RELATIVE)
   X    0.000      Y   0.000
   Z    0.000
> _
MDI **** *** ***       16:05:59
[ MACRO ] [ MENU ] [  OPR  ] [     ] [(OPRT)]
```

```
O9501 ;
N1G65 H92 P066 079 Q076 084 R032 072 I 079 076 J069 032 ;          VAR : BOLT HOLE
N2G65 H93 P500 Q084 079 R079076 ;                                  #500 TOOL
N3G65 H93 P501 Q075 073 R074 085 I078 032 J088 032 ;               #501 KIJUN X
N4G65 H93 P502 Q075 073 R074 085 I 078 032 J089 032 ;              #502 KIJUN Y
N5G65 H93 P503 Q082 065 R068 073 I 085 083 ;                       #503 RADIUS
N6G65 H93 P504 Q083 046 R032 065 I 078 071 J 076 032 ;             #504 S.ANGL
N7G65 H93 P505 Q072 079 R076 069 I 083 032 J078 079 K046 032 ;     #505 HOLES NO
N8G65 H94 ;                                                        Comment
N9G65 H94 P042 066 Q079 076 R084 032 I072 079 J076 069 ;           "BOLT HOLE
N10G65 H94 R032 067 I073 082 J067 076 K069 042 ;                   CIRCLE"
N11G65 H94 P083 069 Q084 032 080 065 I084 084 J069 082 K078 032 ;  SET PATTERN
N12G65 H94 P068 065 Q084 065 R032 084 I079 032 J086 065 K082046 ;  DATA NO VAR.
N13G65 H94 P078 079 Q046 053 R048 048 I045 053 J048 053 K046 032 ; No.500-505
N14M99 ;
```

2.20.3 Таблица символов и их кодов

Cha rac- ter	Code	Comment	Cha rac- ter	Code	Comment
A	065		6	054	
B	066		7	055	
C	067		8	056	
D	068		9	057	
E	069			032	Space
F	070		!	033	Exclama– tion mark
G	071		"	034	Quotation mark
H	072		#	035	Hash sign
I	073		$	036	Dollar sign
J	074		%	037	Percent
K	075		&	038	Ampersand
L	076		'	039	Apostrophe
M	077		(	040	Left parenthesis
N	078		)	041	Right parenthesis
O	079		*	042	Asterisk
P	080		+	043	Plus sign
Q	081		,	044	Comma
R	082		–	045	Minus sign
S	083		.	046	Period
T	084		/	047	Slash
U	085		:	058	Colon
V	086		;	059	Semicolon
W	087		<	060	Left angle bracket
X	088		=	061	Equal sign
Y	089		>	062	Right angle bracket
Z	090		?	063	Question mark
0	048		@	064	HAt"mark
1	049		[	091	Left square bracket
2	050		^	092	
3	051		¥	093	Yen sign
4	052		]	094	Right squar bracket
5	053		_	095	Underscore

ПРИМЕЧАНИЕ

Правые и левые скобки не могут использоваться в шаблонах

Номера подпрограмм используемые в функциях шаблона

Подпрограмма	Функция
O9500	Определяет строку символов названия меню
O9501	Определяет строку символов названия шаблона номер 1
O9502	Определяет строку символов названия шаблона номер 2
O9503	Определяет строку символов названия шаблона номер 3
O9504	Определяет строку символов названия шаблона номер 4
O9505	Определяет строку символов названия шаблона номер 5
O9506	Определяет строку символов названия шаблона номер 6
O9507	Определяет строку символов названия шаблона номер 7
O9508	Определяет строку символов названия шаблона номер 8
O9509	Определяет строку символов названия шаблона номер 9
O9510	Определяет строку символов названия шаблона номер 10

Команды макроса используемые для ввода данных

Код	Функция
G 65 H90	Определяет строку символов названия меню
G 65 H91	Определяет строку символов названия шаблона
G 65 H92	Определяет строку символов названия данных шаблона
G 65 H93	Определяет строку символов названия переменной
G 65 H94	Определяет строку символов комментария

Системные переменные используемые в функции ввода данных шаблона

Переменная	Функция
#9500	Номер шаблона выбранный пользователем