

Министерство образования Республики Беларусь

Учреждение образования  
«Гомельский государственный технический  
университет имени П. О. Сухого»

Кафедра «Технология машиностроения»

**Н. А. Старовойтов**

## **РАЗРАБОТКА УПРАВЛЯЮЩИХ ПРОГРАММ ДЛЯ ТОКАРНЫХ СТАНКОВ С ЧПУ**

**ПРАКТИКУМ**

**по выполнению лабораторных работ  
для студентов специальности 1-36 01 01  
«Технология машиностроения»  
дневной и заочной форм обучения**

**Электронный аналог печатного издания**

**Гомель 20017**

УДК 621.9.06-529(075.8)  
ББК 34.630.2я73  
С77

*Рекомендовано к изданию научно-методическим советом  
машиностроительного факультета ГГТУ им. П. О. Сухого  
(протокол № 1 от 12.09.2016 г.)*

Рецензент: доцент кафедры «Сельскохозяйственные машины» ГГТУ им. П. О. Сухого  
канд. техн. наук *А. В. Голопятин*

**Старовойтов, Н. А.**

С77

Разработка управляющих программ для токарных станков с ЧПУ : практикум по выполнению лаборатор. работ для студентов специальности 1-36 01 01 «Технология машиностроения» днев. и заоч. форм обучения / Н. А. Старовойтов. – Гомель : ГГТУ им. П. О. Сухого, 2017. – 112 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <https://elib.gstu.by>. – Загл. с титул. экрана.

ISBN 978-985-535-350-9.

Представлены лабораторные работы по подготовке управляющих программ для станков с ЧПУ серии NC. Изложенный материал предназначен для более глубокого изучения на практике лекционного материала и приобретения практических навыков подготовки управляющих программ для токарных, сверлильно-фрезерно-расточных и других станков.

Для студентов специальности 1-36 01 01 «Технология машиностроения» дневной и заочной форм обучения.

УДК 621.9.06-529(075.8)  
ББК 34.630.2я73

ISBN 978-985-535-350-9

© Старовойтов Н. А., 2017  
© Учреждение образования «Гомельский  
государственный технический университет  
имени П. О. Сухого», 2017

## ПРЕДИСЛОВИЕ

Внедрение ЧПУ в технологию машиностроения обусловило необходимость построения числовых моделей технологического процесса, что в свою очередь привело к широкому использованию математических методов и числовой вычислительной техники. Технология машиностроения из науки, носившей преимущественно качественный характер, стала превращаться в науку точную. Программирование обработки на станках с ЧПУ, возникшее на стыке ряда дисциплин: технологии машиностроения, математики и кибернетики, отличающееся трудоемкостью и сложностью, требует от технолога высокой профессиональной подготовки, знания не только ряда технологических дисциплин, но и основ программирования и некоторых разделов математики и кибернетики. В результате появилась новая специальность «технолог-программист». На предприятиях сформировались бюро программного управления для технологического обеспечения станков с ЧПУ.

Дисциплина «Технология обработки на станках с ЧПУ» является одной из заключительных технологических дисциплин, а поэтому ее задача – систематизированное изложение методов расчета и проектирования технологических процессов обработки деталей на современных станках с ЧПУ.

Основная задача дисциплины заключается в том, чтобы студенты приобрели умение самостоятельно решать комплекс задач и вопросов, связанных с разработкой технологических процессов для станков с ЧПУ. После усвоения материала лекций они должны уметь:

- разрабатывать, контролировать и редактировать управляющие программы для токарных и сверлильно-фрезерно-расточных станков в ручном режиме на основе приобретенных знаний;
- вводить управляющую программу обработки детали в устройство числового программного управления, осуществлять ее контроль и редактирование.

Для более прочного закрепления теоретических знаний программой предусмотрены лабораторные занятия.

## *Лабораторная работа № 1*

### **ИЗУЧЕНИЕ КОНСТРУКЦИИ И ТЕХНИЧЕСКИХ ХАРАКТЕРИСТИК СТАНКА мод. 16А20Ф3 С ЧПУ**

**Цель работы:** изучение конструкции и технических характеристик станка модели 16А20Ф3 с ЧПУ.

В качестве учебного пособия используются руководство по эксплуатации (РЭ) станка мод. 16А20Ф3 с ЧПУ и практическое ознакомление с конструкцией станка в лаборатории университета.

При самостоятельной работе РЭ можно скачать бесплатно на сайтах: [www.chipmarket.ru](http://www.chipmarket.ru), [www.studmed.ru](http://www.studmed.ru). При выполнении работы в лаборатории университета РЭ входит в состав работы № 1.

При изучении конструкции станка необходимо обратить особое внимание на изучение разделов РЭ: 1; 2 (табл. 1); 5; 7 (п.п. 11–25); 8 и 9.

В конце занятия необходимо ответить на ряд тестов для подтверждения усвоения материала и зачета работы:

1. Какие виды обработки можно производить на токарном патронно-центровом станке мод. 16А20Ф3?
2. Наибольший диаметр обработки над суппортом.
3. Наибольший диаметр обработки над станиной.
4. Наибольшая длина обрабатываемого изделия с 8-позиционной головкой.
5. Диаметр цилиндрического отверстия в шпинделе.
6. Область применения станка мод. 16А20Ф3.
7. Для чего в шпиндельной бабке установлен датчик, беззазорно связанный со шпинделем через зубчатую передачу?
8. Наибольший диаметр изделия, устанавливаемого над станиной.
9. Номер конуса Морзе центра в пиноли.
10. Наибольшая высота резца, устанавливаемого в резцедержателе.
11. Как регулируется усилие зажима детали?
12. Как обозначаются оси в токарном патронно-центровом станке мод. 16А20Ф3?
13. Наибольший ход суппорта по оси X.
14. Наибольший ход суппорта по оси Z.
15. Пределы частот вращения шпинделя, об/мин.
16. Номер конуса Морзе центра в шпинделе.

Для сдачи тестов отводится 10 мин.

Результаты тестов заносятся в ведомость оценки результатов сдачи тестов.

## Лабораторная работа № 2

### ИЗУЧЕНИЕ ПУЛЬТА ОПЕРАТОРА УЧПУ NC201M СТАНКА мод. 16A20Ф3

**Цель работы:** изучение пульта оператора УЧПУ NC201M, инсталлированного на станке с мод. 16A20Ф3, и приобретение навыков управления станком.

В качестве учебного пособия используются:

1. Руководство по эксплуатации (РЭ) станка мод. 16A20Ф3 с ЧПУ NC201M и практическое ознакомление с пультом управления ЧПУ в лаборатории университета. При самостоятельной работе РЭ можно скачать бесплатно на сайтах: [www.chipmarket.ru](http://www.chipmarket.ru), [www.studmed.ru](http://www.studmed.ru). При выполнении работы в лаборатории университета РЭ входит в состав работы № 1.

2. Руководство оператора (РО УЧПУ NC-201, NC-201M, NC-202, ООО «Балт-Систем», Санкт-Петербург) можно скачать на бесплатном доступном сайте [www.bsystem.ru](http://www.bsystem.ru). При выполнении работы в лаборатории университета РО входит в состав работы № 2.

3. Руководство программиста (РП УЧПУ NC-110, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230, ООО «Балт-Систем», Санкт-Петербург, [www.bsystem.ru](http://www.bsystem.ru)). При выполнении работы в лаборатории университета РП входит в состав работы № 2.

В конце занятия необходимо ответить на ряд тестов для подтверждения усвоения материала и зачета работы:

1. Для чего служит переключатель JOG на станочной консоли?
2. Для чего служит переключатель режимов на станочной консоли?
3. Какие команды выполняются при задании режима работы станка MDI «Ручной ввод кадра»?
4. Какие команды выполняются при задании режима работы станка AUTO «Автоматический»?
5. Какие команды выполняются при задании режима работы станка STEP «Кадр»?
6. Какие команды выполняются при задании режима работы станка MANU «Безразмерные ручные перемещения»?
7. Какие команды выполняются при задании режима работы станка MANJ «Фиксированные ручные перемещения»?
8. Какие команды выполняются при задании режима работы станка PROF «Автоматический возврат на профиль»?
9. Какие команды выполняются при задании режима работы станка HOME «Выход в ноль»?

10. Какие операции позволяет выполнять режим «Команда»?
11. Какие операции позволяет выполнять режим «Управление станком»?
12. Каким кодом обозначаются нулевые точки локальных систем?
13. Какой формат применяется для записи начальной нулевой точки?
14. Какая команда предусмотрена для вызова нулевой точки?
15. Для чего служит переключатель F на станочной консоли?
16. Для чего служит переключатель S на станочной консоли?

Для сдачи тестов отводится 10 мин.

Результаты тестов заносятся в ведомость оценки результатов сдачи тестов.

### *Лабораторная работа № 3*

## **ПРИОБРЕТЕНИЕ НАВЫКОВ НАПИСАНИЯ ПРОГРАММ ДЛЯ ТОКАРНОЙ ОБРАБОТКИ ДВУХСТУПЕНЧАТОГО ВАЛИКА ДВУМЯ РЕЗЦАМИ**

### **Цель работы:**

1. Изучить «Характеристики программирования» УЧПУ N201M.
2. Изучить подготовительные G-функции и вспомогательные M-функции, некоторые трехбуквенные коды.
3. Научиться составлять программы для токарной обработки двухступенчатого валика двумя резцами.

В качестве учебного пособия используются:

1. Методические указания по выполнению данной работы.
2. Руководство оператора (РО УЧПУ NC-201, NC-201M, NC-202, ООО «Балт-Систем», Санкт-Петербург, [www.bsistem.ru](http://www.bsistem.ru)). При выполнении работы в лаборатории университета РО входит в состав работы № 3.
3. Руководство программиста (РП УЧПУ NC-110, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230, ООО «Балт-Систем», Санкт-Петербург, [www.bsistem.ru](http://www.bsistem.ru)). При выполнении работы в лаборатории университета РП входит в состав работы № 3.

В конце занятия необходимо:

- предоставить для защиты написанную управляющую программу (УП) двухступенчатого валика;
- ответить письменно на ряд нижеперечисленных вопросов для подтверждения усвоения материала и зачета работы:

1. Какое количество подготовительных функций M можно программировать в одном кадре?
2. Какой формат нарезания метрической резьбы с постоянным шагом?
3. Что определяет адресное слово R?
4. Что определяет функция T4.4?
5. Какой формат написания управляющих программ (УП) применяется для УЧПУ NC 201M?
6. Что означают в формате написания УП цифры после функции T4.4?
7. Что определяют адресные слова I и J?
8. Что означают в формате написания УП цифры после функции F5.2?

9. Что определяет функция S?
10. Какими подготовительными функциями приводится в действие корректировка длины и диаметра инструмента?
11. Какую плоскость круговой интерполяции определяет подготовительная функция G17?
12. Какую плоскость круговой интерполяции определяет подготовительная функция G19?
13. Какое количество подготовительных функций G можно программировать в одном кадре?
14. Какой трехбуквенный код используется для повторения программ, например, 5 раз?
15. Какую плоскость круговой интерполяции определяет подготовительная функция G18?
16. Какой трехбуквенный код закрывает часть программы, которую необходимо повторить?
- Для сдачи тестов отводится 10 мин.
- Результаты тестов заносятся в ведомость оценки результатов сдачи тестов.

### **МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ № 3**

Для выполнения работы необходимо изучить следующий материал:

#### **1. Характеристики программирования УЧПУ NC-201M**

##### **1.1. Система измерения**

Миллиметры или дюймы, выбираемые посредством функции, соответственно G71/G70.

##### **1.2. Программирование абсолютных размеров или по приращениям**

Подготовительная функция: G90 – абсолютное программирование, G91 – программирование по приращениям.

##### **1.3. Программирование относительно нуля станка**

Перемещения, запрограммированные в кадре, могут быть отнесены к нулю станка заданием функции G79.

##### **1.4. Программирование с десятичной точкой**

Размеры программируются так, как читаются (без нулей в начале или в конце), с указанием точки разделения целой части от десятичной. Пример: X-20.275.

### **1.5. Код ленты**

EIA RS244, ISO 840 с автоматическим распознаванием.

### **1.6. Формат программирования**

N4,G2,X/Y/Z/A/B/C/U/W/V/P/Q/D/5.4,R5.4,I/J/K5.4,F5.2,S5.2,T4.4,  
M2,H2\*

\*) Временно не программируется, в стадии разработки.

### **1.7. Координаты осей**

Координаты программируются в миллиметрах или дюймах от +/-0.0001 до +/-99999.9999.

### **1.8. Координаты I, J**

Определяют координаты центра окружности в круговой интерполяции I по оси абсцисс J и по оси ординат. Программируемое значение: от +(-) 0.0001 до +(-) 99999.9999 миллиметров или дюймов.

### **1.9. Вращательные движения**

Во время характеристики системы любая ось может быть определена как ось вращения. Программируемое значение: от +/-0.0001 до +/-99999.9999 градусов.

### **1.10. Функция F**

Программируется от 0.01 до 99999.99.

**G94** – определяет скорость подачи осей в мм/мин или дюйм/мин; с помощью символа «t» можно запрограммировать время в секундах, необходимое для отработки элемента, определенного в кадре («F» для кадра является отношением между длиной элемента и запрограммированным «t»);

**G93** – определяет обратное время, т. е. отношение скорость подачи/расстояние;

**G95** определяет скорость осей в мм/оборот.

### **1.11. Функция S**

Программируется от 0.01 до 99999.99. Может выражать:

– число оборотов/мин шпинделя (**G97**);

– скорость резания в м/мин (**G96**).

### **1.12. Функция T**

Определяет требуемый для обработки инструмент и номер коррекции для данного инструмента. Программируемая величина: от 1.0 до 9999.9999. Цифры перед десятичной точкой определяют инструмент, после – номер корректора.

### **1.13. Подготовительные функции G**

G00 – быстрое позиционирование;

G01 – линейная интерполяция;

- G02 – интерполяция круговая по часовой стрелке;
- G03 – интерполяция круговая против часовой стрелки;
- G04 – выдержка времени, заданная в кадре;
- G06 – сплайновая интерполяция;
- G09 – замедление в конце кадра;
- G17 – выбирает плоскость интерполяции, определенную конфигурируемыми осями 1 и 2;
- G18 – выбирает плоскость интерполяции, определенную конфигурируемыми осями 3 и 1;
- G19 – выбирает плоскость интерполяции, определенную конфигурируемыми осями 2 и 3;
- G20 – закрывает среду программирования языка GTL;
- G21 – открывает среду программирования языка GTL;
- G27 – непрерывная обработка с автоматическим уменьшением скорости на углах;
- G28 – непрерывная обработка без автоматического уменьшения скорости на углах;
- G29 – позиционирование «от точки к точке»;
- G33 – нарезание резьбы с постоянным или изменяющимся шагом;
- G34 – нарезание резьбы с постоянным или изменяющимся шагом;
- G35 – синхронизированное начало движения со шпинделем;
- G40 – отмена корректировки на профиле;
- G41 – приводит в действие корректировку на профиле (инструмент слева);
- G42 – приводит в действие корректировку на профиле (инструмент справа);
- G70 – программирование в дюймах;
- G71 – программирование в миллиметрах;
- G72 – измерение точки с компенсацией радиуса инструмента;
- G73 – измерение параметров отверстия;
- G74 – измерение отклонения от теоретической точки без компенсации радиуса инструмента;
- G79 – программирование относительно нуля станка (действительно только в данном кадре);
- G80 – отмена постоянных циклов;
- G81 – цикл сверления;
- G82 – цикл растачивания;
- G83 – цикл глубокого сверления;
- G84 – цикл нарезания резьбы метчиком;

G85 – цикл рассверливания; цикл развертывания; цикл развертывания с остановкой;

G90 – абсолютное программирование;

G91 – программирование по приращениям;

G93 – скорость подачи, выраженная в виде обратного времени выполнения;

G94 – скорость подачи осей, мм/мин или дюйм/мин;

G95 – скорость подачи осей, мм/оборот;

G96 – скорость вращения шпинделя, м/мин;

G97 – скорость вращения шпинделя, оборот/мин.

#### **1.14. Вспомогательные функции M**

M00 – остановка программы;

M01 – условная остановка программы;

M02 – конец программы;

M03 – вращение шпинделя по часовой стрелке;

M04 – вращение шпинделя против часовой стрелки;

M05 – остановка вращения шпинделя; замена инструмента;

M06 – замена инструмента;

M07 – включение вспомогательного охлаждения;

M08 – включение основного охлаждения;

M09 – выключение охлаждения;

M10 – блокировка осей;

M11 – разблокировка осей;

M12 – блокировка вращающихся осей;

M13 – вращение шпинделя по часовой стрелке и охлаждение;

M14 – вращение шпинделя против часовой стрелки и охлаждение;

M19 – остановка вращения шпинделя с угловой ориентацией;

M30 – конец программы и возврат к первому кадру;

M41–M44 – выбирает диапазон вращения шпинделя 1–4;

M40 – аннулирует диапазон вращения шпинделя;

M45 – автоматическая замена диапазона;

M60 – замена детали.

#### **1.15. Постоянные циклы**

С использованием подготовительных функций G81–G89 определения подготовительного цикла можно запрограммировать ряд операций (сверление, нарезание резьбы метчиком, растачивание и т. д.) без повторения для каждой из них параметров отверстия, запрограммированную обработку которого надо осуществить. Последовательность движений циклов может быть установлена в следующем порядке:

- быстрое позиционирование к оси отверстия;
- быстрый подход к плоскости обработки (размер R);
- рабочая скорость подачи до запрограммированного размера Z;
- фиксации цикла на дне отверстия;
- ускоренное или со скоростью обработки возвращение к точке R.

Можно программировать размер возвращения R2, отличный от R (тогда два размера R в кадре).

Характеристики постоянных циклов приведены в табл. 3.1.

Таблица 3.1

### Характеристики постоянных циклов

Постоянный цикл	Подход	Функции на дне отверстия		Возврат
		выдержка времени	вращение шпинделя	
G81 – сверление	рабочая подача	нет	рабочая скорость	быстрый ход
G82 – растачивание	рабочая подача	да	рабочая скорость	быстрый ход
G83 – глубокое сверление (с разгрузкой стружки)	в прерывистой работе	нет	рабочая скорость	быстрый ход
G84 – нарезание резьбы метчиком	рабочая подача, начало вращения шпинделя	нет	изменение направления	рабочая подача
G85 – рассверливание или tapermatic	рабочая подача	нет	рабочая скорость	рабочая подача
G86 – развертывание	рабочая подача, начало вращения шпинделя	нет	останов	быстрый ход
G89 – развертывание с растачиванием	рабочая подача	да	рабочая скорость	рабочая подача

#### **1.16. Постоянный цикл нарезания резьбы метчиком с датчиком на шпинделе**

В этом цикле скорость подачи F не программируется, так как вычисляется автоматически в соответствии с числом оборотов шпинделя и шага K метчика нарезания резьбы.

#### **1.17. Изменение скорости возвращения при нарезании резьбы метчиком**

Определяя процентное содержание изменения посредством кода RMS, введенного в программу или накопленного в памяти с клавиатуры.

туры, можно модифицировать скорость возврата в цикле нарезания резьбы метчиком.

*Пример*

RMS=110 (+10 % запрограммированного F)

RMS=10 (–90 % запрограммированного P)

### **1.18. Сообщения программы**

На втором видеокadre в зоне сообщений могут быть воспроизведены сообщения, переменные, константы, которые программируются посредством трехбуквенного кода DIS.

*Примеры*

(DIS, «ИНСТРУМЕНТ=12»);

(DIS, E37);

(DIS, UOV)

### **1.19. Коэффициент масштабирования**

Коэффициент масштабирования применяется для масштабирования заданного перемещения для определенных осей. Программируют трехбуквенный код SCF и коэффициент масштабирования, который необходимо применить.

*Примеры*

(SCF, 2) для всех осей,

(SCF, 2, X) для оси X.

### **1.20. Нарезание резьбы**

С функцией G33 программируется цикл цилиндрического или конического нарезания резьбы, с постоянным или переменным шагом. Параметры, запрограммированные в кадре, определяют тип нарезания резьбы.

Формат:

**G33 Z..K..** – цилиндрическое нарезание резьбы с постоянным шагом;

**G33 Z..U..K..** – коническое нарезание резьбы с постоянным шагом;

**G33 Z..K..I+..** – нарезание резьбы с увеличивающимся шагом;

**G33 Z..K..I-..** – нарезание резьбы с уменьшающимся шагом,

где **G33** – подготовительная функция; **Z, U** – координаты конечной точки; **K** – шаг нарезания резьбы; **I+/-** – изменение шага.

### **1.21. Векторная компенсация радиуса инструмента**

Векторная компенсация радиуса инструмента позволяет осуществить программирование контуров профиля без учета радиуса инструмента. Корректировка радиуса действует в перпендикулярном направлении к запрограммированному профилю и приводится в действие при помощи функций:

- G41** – корректировка слева от профиля;
- G42** – корректировка справа от профиля.

Параметры корректировки, которую надо применить к паре осей для их коррекции, вычисляются автоматически. Корректировка отменяется функцией G40.

### **1.22. Определение припуска**

Кодом UOV можно определить припуск в операциях контурной обработки. Заданный в программе или введенный с клавиатуры код UOV временно модифицирует значение корректировки на величину, равную установленному значению. Отмена припуска программируется установкой кода UOV=0.

*Пример*  
(UOV=1.5)

### **1.23. Зеркальная обработка**

Трехбуквенный код MIR позволяет зеркальную обработку для всех скоординированных осей.

*Пример*  
(MIR,X)  
.....  
(MIR,Y)  
.....  
(MIR,X,Y)

### **1.24. Вращение в плоскости**

Программируя трехбуквенный код URT, можно вращать в плоскости часть или всю запрограммированную деталь. Вращение происходит вокруг начальной точки, активной в этот момент.

*Пример:*  
(URT,45)

### **1.25. Повторение программ**

Используя трехбуквенный код RPT, можно повторять *n* раз программу или часть программы для создания специальных циклов. Максимальное количество повторений – 99. Внутри повторяющегося цикла можно создать другой цикл, а в нем – еще один (до трех уровней). Часть программы, которую необходимо повторить, закрывается трехбуквенным кодом ERP.

*Пример:*  
(RPT,99)  
.....  
(ERP)

## 1.26. Выполнение частей программы

Программируя трехбуквенный код EPP, можно выполнить часть программы, заключенной между двумя метками.

*Пример*

"НАЧАЛО"

.....

"КОНЕЦ"

(EPP, НАЧАЛО, КОНЕЦ)

После выполнения части программы программа продолжается от кадра, следующего после команды EPP.

## 2. Функции программирования

### 2.1. Ввод программ

Программа, которую необходимо выполнить, должна быть занесена в память системы. Ввод программы в память может осуществляться с клавиатуры, с дискеты или с ПК по последовательному каналу. В последнем случае в ПК должна быть загружена программа `comp.c.exe`.

### 2.2. Информация управляющих программ

#### 2.2.1. Символ

Символ – это число, буква или знак, используемые для выражения информации.

*Пример:*

I, G, %, 3, X, LF..

Используемые в УЧПУ символы должны соответствовать символам, которые описаны в табл. 3.2.

Таблица 3.2

Символы, используемые в УЧПУ

Описание	Символы
Заглавные буквы	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Строчные буквы	a b c d l m o p r s t u v w
Десятичные цифры	от 0 до 9
Математические действия 1	+ –
Математическое действие 2	*
Математическое действие 3	/
Десятичная точка	.
Сепаратор	"

Описание	Символы
Открытая скобка	(
Закрытая скобка	)
Пояснительный знак	;
Разделительный знак	,
Знак	=
Конец или начало ленты	% (ISO)
Терминатор	L.F. (ISO)
Особые символы	:
Символы-приставки	# (запрос синхронизации) & (аннулирует синхронизацию)

### 2.2.2. Адрес

Адрес представлен буквой, которая определяет тип инструкции.

*Пример:*

G, Z, X, F

### 2.2.3. Слово

Слово состоит из адреса, за которым следует цифровое значение.

*Пример:*

G1X50.5Y-3.15F200T1.1M6

Все цифровые значения, которые записаны за адресным словом, выражены своей системой измерения. В общем случае нули в начале и в конце могут быть опущены. Если величины имеют десятичную часть, она должна быть записана после десятичной точки.

### 2.2.4. Кадр

Программа состоит из последовательности кадров, которые позволяют описать цикл обработки. Каждый кадр – это последовательность слов, определяющих операции, которые необходимо выполнить. Каждый кадр должен заканчиваться символом LF (ISO). Максимальная длина кадра – 128 символов. Все кадры, кроме комментирующего, который будет описан далее, могут иметь в начале три дополнительных поля, независимо от класса, к которому принадлежит кадр. А именно:

- 1) поле подтверждения кадра и выведения его из рабочего состояния (символ 1);
- 2) поле метки;
- 3) поле номера кадра.

Они могут присутствовать в кадре поодиночке или одновременно. В случае если они присутствуют одновременно, последовательность расположения одиночных полей должна быть в обязательном порядке следующей: 1), 2), 3). Поле подтверждения кадра и выведения его из рабочего состояния позволяет включить в программу кадры, выполнение которых зависит от параметра системы, названного USB (см. кадры назначения в п. 2.5). Если параметр является активным (USB=1), кадр выполняется, в противном случае (USB=0) кадр рассматривается как комментирующий. Формат устанавливается знаком «/» в первой позиции кадра.

*Пример:*

/N100G00X100

Поле метки позволяет дать символическое название кадру, которому оно принадлежит. Метка служит для возможности вызова кадра из различных точек программы при помощи инструкций перехода. Метка – это алфавитно-цифровая последовательность символов, заключенная в знак « » (кавычки), максимальная длина которой шесть символов. Метка должна быть запрограммирована сразу же после поля «/», если оно присутствует.

*Пример:*

"START"

/"END"

Поле номера кадра служит для нумерации одиночных кадров программы. Номер кадра устанавливается символом «N», за которым следует число, и должен быть запрограммирован в начале каждого кадра, но после символа «/» и метки.

*Пример:*

N125

"INIZIO" N 125

/"FINE" N 125

### **2.3. Типы кадров**

В языке можно определить четыре типа кадров:

- 1) комментирующие кадры;
- 2) кадры ISO;
- 3) кадры назначения;
- 4) кадры с трехбуквенными кодами.

Комментирующий кадр дает возможность программисту вводить в программу фразы, описывающие функции, которые он должен выполнить, делая, таким образом, программу более легко читаемой.

Такой кадр не выдает посылки оператору и не учитывается в стадии выполнения программы. Формат состоит из последовательности алфавитно-цифровых символов, из которых **первым элементом в обязательном порядке должен быть символ «;»**.

*Пример:*

;ЭТО - ПРИМЕР

Кадры ISO – это кадры, операторы которых предусмотрены стандартом ISO.

*Пример:*

G1X500Y20F200

Кадры назначения непосредственно из программы пользователя позволяют определить величину нескольких глобальных параметров системы. Впоследствии эти параметры могут быть использованы в других кадрах того же или другого класса. В зависимости от типа переменных кадры назначения могут быть подразделены на три класса:

1) кадры назначения с переменными вычисления; например: E30=28.5;

2) кадры назначения с геометрическими переменными; например: p2=X10 Y25;

3) кадры назначения с глобальными переменными системы; например: UOV=1.5 (Определение припуска).

Кадры с трехбуквенными кодами – это кадры, в которых тип операции, выполнение которой предусмотрено, определен трехбуквенной командой (кодом), **согласованной со стандартом EIA 1177B**.

*Пример:*

(URT,45) (Поворот плоскости)

Оператор G может быть запрограммирован либо неявным способом при помощи параметров E, либо явным. В неявном программировании используемый тип параметра – байт. Все элементы, заключенные в квадратные скобки [ ], должны рассматриваться как необязательные. Все элементы, заключенные в фигурные скобки {...}, должны рассматриваться как альтернативные.

#### **2.4. Начало и конец программы**

В первом кадре обычно программируется информация о замене инструмента (T..M06). В конце обработки необходимо установить оси в позиции, удобной для демонтажа детали. Затем следует остановить вращение шпинделя и охлаждающий поток и осуществить управление автоматической установкой («СБРОС») программы при помощи функции M30.

N1(DIS,".....")  
 N2 G97S800T1.1M6  
 N3G0Z80X80M13

.....

.....

N236G0Z250X50M5  
 N237M30

Можно вставить во внутрь программы сообщение, заключенное в кавычки и предназначенное для оператора станка. Это сообщение программируется трехбуквенным кодом следующим образом: (DIS,"**текст сообщения**"). Текст сообщения не должен превышать 32 символа.

### 2.5. Адресные слова управляющей программы

Подготовительные функции (G), допустимые для программирования в УЧПУ, представлены в табл. 3.3.

Таблица 3.3

Подготовительные функции G

Код	Группы модальных функций	Действительна только в кадре	Функция	Присутствует при включении
G00	a	нет	Быстрое позиционирование осей	да
G01	a	нет	Линейная интерполяция	нет
G06	a	нет	Сплайновая интерполяция	нет
G02	a	нет	Круговая интерполяция по часовой стрелке	нет
G03	a	нет	Круговая интерполяция против часовой стрелки	нет
G33	a	нет	Нарезание резьбы с постоянным или переменным шагом	нет
G34	a	нет	Нарезание резьбы с постоянным или переменным шагом	нет
G17	b	нет	Функция задания плоскости XY (1–2 оси)	да
G18	b	нет	Функция задания плоскости ZX (3–1 оси)	нет
G19	b	нет	Функция задания плоскости YZ (2–3 оси)	нет

Код	Группы модальных функций	Действительна только в кадре	Функция	Присутствует при включении
G27	c	нет	Непрерывный режим обработки с автоматическим замедлением скорости на углах	да
G28	c	нет	Непрерывный режим обработки без замедления скорости на углах	нет
G29	c	нет	Перемещение от точки к точке	нет
G21	d	нет	Вход в программу GTL	нет
G20	d	нет	Выход из программы GTL	да
G40	e	нет	Отмена компенсации радиуса инструмента	да
G41	e	нет	Компенсация радиуса инструмента (инструмент слева)	нет
G42	e	нет	Компенсация радиуса инструмента (инструмент справа)	нет
G70	f	нет	Программа в дюймах	нет
G71	f	нет	Программа в мм	да
G80	g	нет	Отмена постоянных циклов	да
G81	g	нет	Постоянный цикл сверления	нет
G82	g	нет	Постоянный цикл растачивания	нет
G83	g	нет	Цикл глубокого сверления (с разгрузкой стружки)	нет
G84	g	нет	Постоянный цикл нарезания резьбы метчиком	нет
G85	g	нет	Постоянный цикл рассверливания	нет
G86	g	нет	Постоянный цикл развертывания	нет
G89	g	нет	Постоянный цикл развертывания с остановкой	нет
G90	h	нет	Абсолютное программирование	да
G91	h	нет	Программирование в приращениях	нет
G79	k	да	Программирование относительно нуля станка	нет
G04	i	да	Выдержка времени в конце кадра	нет
G09	i	да	Замедление в конце кадра	нет
G72	j	да	Измерение точки с компенсацией радиуса	нет

Код	Группы модальных функций	Действительна только в кадре	Функция	Присутствует при включении
G73	j	да	Измерение параметров отверстия	нет
G74	j	да	Измерение теоретического смещения от точки без компенсации радиуса	нет
G93	l	нет	Скорость подачи выражена как обратное время выполнения элемента	нет
G94	l	нет	Скорость подачи в мм/мин или дюйм/мин	нет
G95	l	нет	Скорость подачи в мм/об или дюйм/об	да
G96	m	нет	Скорость резания в м/мин или фут/мин	да
G97	m	нет	Скорость вращения шпинделя выражена в об/мин	нет
G35	n	да	Синхронизация начала движения со шпинделем	нет

*Примечания:* 1. Выдержка времени программируется трехбуквенным кодом TMR: TMR = n; n – выражено в секундах при G94 и в количестве оборотов шпинделя при G95. 2. Представляется возможным программировать несколько функций G в одном и том же кадре, с учетом их совместимости.

#### 2.5.1. Адресные слова координатных осей A, B, C, U, V, W, X, Y, Z, P, Q, D

Координаты программируются в миллиметрах или дюймах от  $\pm 0.0001$  до  $\pm 99999.9999$ .

Любая ось в фазе характеристики системы может быть объявлена осью вращения. Программируемая величина от  $\pm 0.0001$  до  $\pm 99999.9999$  градусов.

#### 2.5.2. Адресное слово R

Определяет в постоянном цикле величину перемещения до точки начала обработки отверстия или величину возврата к этой точке. Программируемая величина от  $\pm 0.0001$  до  $\pm 99999.9999$  миллиметров или дюймов. В кадре нарезания резьбы R представляет сдвиг фаз относительно угловой позиции нуля шпинделя (для многозаходной резьбы).

### 2.5.3. Адресные слова I J

Выражают координаты центра окружности в круговой интерполяции, соответственно **I** – абсцисса и **J** – ордината. Программируемая величина от  $\pm 0.0001$  до  $\pm 99999.9999$  миллиметров или дюймов. Используемыми символами всегда являются **I** и **J**, независимо от плоскости интерполяции. Символы **I** и **J** используются также в постоянном цикле сверления (G83). Символ **I** в кадре нарезания резьбы определяет изменение шага нарезания резьбы с изменяющимся шагом: (**I+**) – для увеличивающихся шагов, (**I-**) – для уменьшающихся шагов.

### 2.5.4. Адресное слово K

Определяет коэффициент умножения для обработки глубины отверстия **I** в G83 (постоянный цикл глубокого сверления с разгрузкой стружки). Определяет шаг резьбы, который необходимо выполнить в G33 (нарезание резьбы) и в G84 (нарезание резьбы метчиком). Определяет в винтовой интерполяции шаг винта. Определяет величину корректировки диаметра инструмента. Программируемая величина от  $\pm 0.0001$  до  $\pm 99999.9999$  миллиметров или дюймов.

### 2.5.5. Функция F

Программируется от 0.01 до 99999.99.

Функция **G94** – определяет скорость подачи осей в мм/мин (если в G71) или в дюйм/мин (если в G70). Имеется возможность программирования посредством символа «t» времени в секундах, необходимого для прохождения участка, определенного в кадре (**F** кадра является отношением между длиной участка и запрограммированным (**t**)). Функция (**t**) действительна только в кадре, в котором она запрограммирована.

Функция **G95** – определяет скорость подачи осей в мм/оборот (G71) или в дюйм/об (G70), если это предусмотрено в характеристике.

Функция **G93** – определяет в минутах обратное время выполнения участка, определенного из отношения: скорость подачи / расстояние. Функция **F** в **G93** действительна только в одном кадре.

### 2.5.6. Функция S

Программируется от 0.01 до 99999.99.

Определяет скорость вращения шпинделя в об/мин при **G97** или скорость резания в м/мин при **G96** (когда это предусмотрено при характеристике).

### 2.5.7. Функция T

Определяет инструмент, необходимый для обработки, и номер соответствующей коррекции. Программируемая величина от 1.0 до

9999.9999. Цифры до десятичной точки определяют инструмент, после – номер коррекции. Число коррекций определяется в фазе установки. Коррекция приводится в действие при помощи функции **M06**. Величины коррекции относятся к длине и диаметру (**K**) инструмента. Корректировка длины инструмента может быть применена к любой оси станка. Выбор зависит от названия оси, к которой присоединена корректировка длины.

*Пример:*

X55, Y20

Корректировка длины приводится в действие без использования других подготовительных функций. **Корректировка диаметра инструмента**, вызванная одновременно с корректировкой длины, приводится в действие при помощи функций компенсации радиуса инструмента **G41/G42** (см. функции программирования **G**).

2.5.8. Обычно используемые вспомогательные функции **M**

Описание функций **M** представлено в табл. 3.4.

Таблица 3.4

#### Функции **M**

Функция	Активная функция		Функция или операции, которые ее отменяют	Значение
	начало движения	конец движения		
M00	–	х	ПУСК	Остановка программы
M01	–	х	–	Условная остановка программы
M02	–	х	–	Конец программы
M03	х		M4-M5-M14-M19	Вращение шпинделя по часовой стрелке
M04	х		M3-M5-M13-M19	Вращение шпинделя против часовой стрелки
M05	–	х	M13-M4-M13-M14	Остановка вращающегося шпинделя
M06	–	х	–	Замена инструмента
M07	х		M9	Включение дополнительного охлаждения
M08	х		M9	Включение основного охлаждения
M09	–	х	M7-M8	Выключение охлаждения
M10	х		M11	Блокировка осей

Функция	Активная функция		Функция или операции, которые ее отменяют	Значение
	начало движения	конец движения		
M11	x	–	M11	Разблокировка осей
M12	x	–	–	Блокировка вращающихся осей
M13	x	–	M4-M5-M14-M19	Вращение шпинделя по часовой стрелке и охлаждение
M14	x	–	M3-M5-M13-M19	Вращение шпинделя против часовой стрелки и охлаждение
M19	x	–	M3-M4-M5-M13-M14	Остановка вращения шпинделя и угловая ориентация
M30	–	x	–	Конец программы и установка на 1-м кадре
M41	x	–	M42-M43-M44-M40	Форсирует 1-й диапазон вращения шпинделя
M42	x	–	M41-M43-M44-M40	Форсирует 2-й диапазон вращения шпинделя
M43	x	–	M41-M42-M44-M40	Форсирует 3-й диапазон вращения шпинделя
M44	x	–	M41-M42-M43-M40	Форсирует 3-й диапазон вращения шпинделя
M40	–	x	M41-M42-M43-M44	Отменяет форсированный диапазон вращения шпинделя
M45	x	–	M41-M42-M43-M44	Автоматическая замена диапазона вращения шпинделя
M60	–	x	–	Замена детали

M00 – останавливает выполнение программы после выполнения операций, содержащихся в кадре. Останавливает вращение шпинделя и охлаждающий поток. Сохраняет всю информацию, накопленную в памяти.

M01 – условная остановка программы: если трехбуквенный код USO = 1 занесен с клавиатуры, функция M01 интерпретируется управлением как M00; если трехбуквенный код USO = 0 подтвержден, функция M01 не учитывается.

M02 – определяет конец программы без перемотки ленты на начало.

M03 – вращение шпинделя по часовой стрелке.

M04 – вращение шпинделя против часовой стрелки.

M05 – остановка шпинделя и подачи охлаждения. Осуществляется после выполнения операций, содержащихся в кадре.

M06 – замена инструмента. Останавливает вращение шпинделя, подачу охлаждения и выполнение программы. Подтверждает корректировки, выбранные функцией T. Осуществление становится возможным после выполнения информации, содержащейся в кадре. Не стирает M03, M04, M08, M13, M14.

M07 – подача вспомогательного охлаждения.

M08 – подача основного охлаждения.

M09 – остановка охлаждения. Осуществляется после выполнения операций, содержащихся в кадре.

M10 – блокировка линейных и вращающихся осей. При помощи этой функции осуществляется блокировка осей, не участвующих в процессе обработки.

M11 – отмена M10.

M12 – блокировка вращающихся осей. При помощи этой функции осуществляется блокировка осей, не участвующих в процессе обработки.

M13 – вращение шпинделя по часовой стрелке и подача охлаждения.

M14 – вращение шпинделя против часовой стрелки и подача охлаждения.

M19 – остановка вращения шпинделя с угловой ориентацией осуществима после операций, содержащихся в кадре. Отменяется функциями M03, M04, M13, M14.

M30 – автоматический СБРОС в конце программы. При помощи функции M30 стирается вся информация, находящаяся в динамическом буфере системы. Подтверждаются автоматически: начальная точка 0 и возобновление выбранной программы. Корректировка инструмента в шпинделе не стирается.

M40 – отмена диапазона вращения шпинделя.

M41-M42-M43-M44 – активизирует диапазон вращения шпинделя 1-4.

M45 – автоматическая смена диапазона вращения шпинделя.

M60 – замена детали.

Путем программирования интерфейса PLS представляется возможным запрограммировать эти функции другим образом, добавляя или сокращая их. В каждом кадре можно запрограммировать до четырех функций М.

Все функции М стираются при помощи выполнения режима «СБРОС».

### 2.6. Кадры программирования с функциями G

Эти кадры определены подготовительными функциями G. Оператор G определяется символом «G», за которым следуют две цифры. Этот оператор должен быть запрограммирован после номера кадра (если таковой имеется) и до какого-либо операнда в кадре. Теоретически существуют 100 операторов типа G (G00-G99), но только часть из них декодируется системой УЧПУ. В одном кадре можно запрограммировать несколько операторов G в случае, если они конгруэнтны. Табл. 3.5 демонстрирует разделения операторов G на классы с точки зрения конгруэнтности и совместимости внутри одного и того же кадра. В этой таблице величина «1» означает «**НЕСОВМЕСТИМОСТЬ**». С функциональной точки зрения функции G подразделены на классы и занесены в табл. 3.5.

Оператор G может быть запрограммирован либо неявным способом при помощи параметров E, либо явным. Параметр, используемый в неявном программировании, является типа «байт». При описании формата кадра будут встречаться следующие знаки:

1) все элементы, заключенные в квадратные скобки [ ], должны рассматриваться как необязательные;

2) все элементы, заключенные в фигурные скобки {..}, должны рассматриваться как альтернативные.

Таблица 3.5

Конгруэнтность операторов G в кадре

G	00	01	02 03	33	81 86 89	80	72 73 74	21	20	41 42	40	27 28	29	04	09	90 91	79	70 71	17 18 19	
G00	1	1	1	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1
G01	1	1	1	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1
G02	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1
G03	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1
G04	0	0	0	1	1	0	1	0	0	0	0	1	0	1	1	0	0	0	0	1
G06	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1
G09	0	0	0	1	1	0	1	0	0	0	0	1	0	1	1	0	0	0	0	1

G	00	01	02 03	33	81 86 89	80	72 73 74	21	20	41 42	40	27 28	29	04	09	90 91	79	70 71	17 18 19
G17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G19	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
G20	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	1	0	1
G21	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	1	0	1
G27	0	0	0	0	1	0	1	0	0	0	0	1	1	1	0	0	0	0	1
G28	0	0	0	1	1	0	1	0	0	0	0	1	1	1	0	0	0	0	1
G29	0	0	0	0	1	0	1	0	0	0	0	1	1	0	0	0	0	0	1
G33	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	1
G34	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	1
G35	1	0	0	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	1
G40	0	0	0	1	1	1	1	0	0	1	1	0	0	0	0	0	1	0	1
G41	0	0	0	1	1	1	1	0	1	1	1	0	0	0	0	0	1	0	1
G42	0	0	0	1	1	1	1	0	1	1	1	0	0	0	0	0	1	0	1
G70	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
G71	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
G72	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G73	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G74	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G79	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	1	1	0	1
G80	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0	1
G81	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G82	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G83	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G84	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G85	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G86	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G89	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1
G90	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
G91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
G93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
G94	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
G97	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Примечание. «1» означает несовместимость.

## Деление функций G на функциональные классы

Класс	Функции	Описание
<b>a</b>	G00-G01-G02-G03- -G06-G33-G34	Определение типа движения
<b>b</b>	G17-G18-G19	Определение плоскости интерполяции
<b>c</b>	G27-G28-G29	Определение динамического режима («от точки к точке» или непрерывный)
<b>d</b>	G21-G20	Открыть и закрыть среду программирования GTL
<b>e</b>	G40-G41-G42	Активизация компенсации радиуса инструмента и ее отмена
<b>f</b>	G70-G71	Программирование в альтернативной системе измерения
<b>g</b>	G81...G86-G89-G80	Постоянные циклы обработки отверстия
<b>h</b>	G90-G91	Программирование абсолютное/в приращениях
<b>i</b>	G79	Программирование относительно нуля станка
<b>j</b>	G04-G09	Свойства динамического типа
<b>k</b>	G72-G73-G74	Циклы измерения
<b>l</b>	G93-G94-G95	Скорость подачи
<b>m</b>	G96-G97	Скорость вращения шпинделя

**2.7. Тип движения**

Тип движения определяется функциями:

**G00** – быстрое позиционирование осей;

**G01** – линейная интерполяция;

**G02** – интерполяция круговая по часовой стрелке;

**G03** – интерполяция круговая против часовой стрелки;

**G06** – сплайновая интерполяция;

**G33** – нарезание резьбы с постоянным или переменным шагом;

**G34** – нарезание резьбы с постоянным или переменным шагом.

**2.7.1. Быстрое позиционирование осей (G00)**

Быстрое позиционирование осей (G00) определяет линейный тип движения, скоординированный по всем осям, запрограммированным в кадре с быстрым ходом.

**Формат:**

**G00 [ДРУГИЕ G] [ОСИ] [ОПЕРАНДЫ КОРРЕКТИРОВКИ] [СКОРОСТЬ ПОДАЧИ] [ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ]**,  
где [ДРУГИЕ G] – все другие функции G, совместимые с G00 (см. табл. 2.3); [ОСИ] – представлены символом оси, за которым сле-

дует числовое значение в явной или неявной форме (параметр E). Могут присутствовать 8 осей (максимально), они не должны быть заменимыми между собой. Для неявного определения осей необходимо вначале определить точку согласно текущей абсциссе и ординате; **[ОПЕРАНДЫ КОРРЕКТИРОВКИ]** – коэффициенты коррекций на плоскости (u,v,w); **[СКОРОСТЬ ПОДАЧИ]** – рабочая подача для скоординированных перемещений. Она запоминается, но не определяет движение осей, определенных в кадре с функцией G00. Скорость подачи в кадре с функцией G00 для программируемых осей определяется на базе скоростей быстрого хода. Скорости быстрого хода определяются в файлах характеристики УЧПУ; **[ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ]** – вспомогательные функции M, S и T. В одном кадре можно программировать: до 4 функций M, по 1 – функции S и T.

### 2.7.2. Линейная интерполяция (G01)

Линейная интерполяция (G01) определяет линейное одновременное движение, скоординированное по всем осям, которые запрограммированы в кадре, с заданной скоростью обработки.

*Формат:*

**G01 [ДРУГИЕ G] [ОСИ] [ОПЕРАНД КОРРЕКТИРОВКИ] [СКОРОСТЬ ПОДАЧИ] [ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ]**, где **[СКОРОСТЬ ПОДАЧИ]** – выражает рабочую скорость (F), с которой выполняется движение. В случае отсутствия используется ранее запрограммированная скорость. Это означает, что в предшествующих кадрах должна быть запрограммирована скорость. В противном случае подается сигнал ошибки.

Описание остальных полей изложено в предыдущем пункте. Пример линейной интерполяции приведен на рис. 3.1.

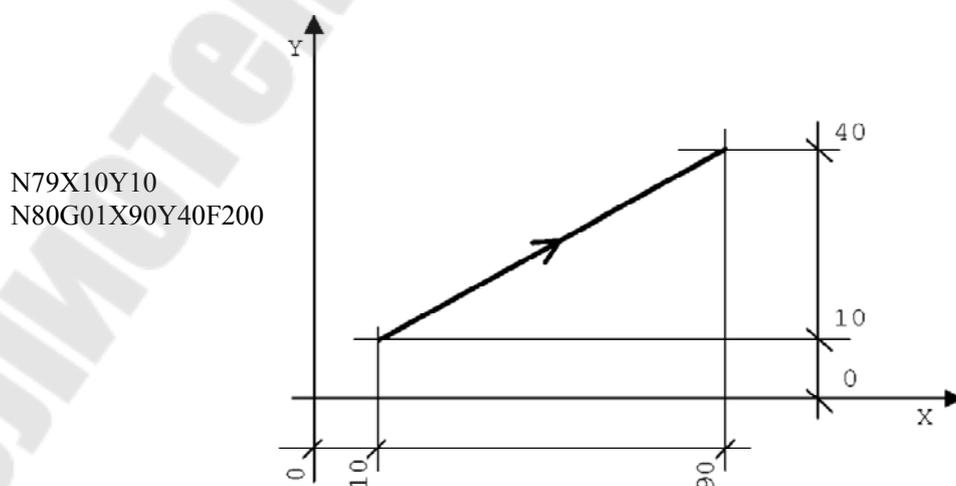


Рис. 3.1. Линейная интерполяция

### 2.7.3. Круговая интерполяция (G02-G03)

Круговая интерполяция (**G02-G03**) определяет круговое движение по часовой стрелке (G02) или против часовой стрелки (G03). Это движение является скоординированным и одновременным по всем осям, запрограммированным в кадре с заданной скоростью обработки.

Формат:

**{G02,G03} [ДРУГИЕ G] [ОСИ] I J [СКОРОСТЬ ПОДАЧИ] [ОПЕРАНДЫ КОРРЕКТИРОВКИ] [ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ],**

где **[G]** – смешанные операторы и вспомогательные функции; имеют те же значения, что и в предыдущих случаях; **[СКОРОСТЬ ПОДАЧИ]** – скорость подачи; **[ОСИ]** – представлены символом оси и цифровым значением в явной или неявной форме (параметр E). Если ни одна ось не запрограммирована, то выполняемым движением будет полное круговое движение в плоскости интерполяции. Оси могут быть определены неявным образом посредством геометрического элемента – точки. Если координата прибытия равна координате отправления, она может быть опущена; **I** и **J** – являются адресными словами, выражающими координаты центра окружности, цифровая часть которых может быть выражена в явной или неявной форме (параметр E). Используемыми символами всегда являются I и J независимо от плоскости интерполяции и всегда присутствуют. Пример круговой интерполяции приведен на рис. 3.2. Максимальная программируемая дуга – 360 градусов.

```
N10 G1X-20Y60F200  
N20G3G91X-20Y20I-20J0  
N30G1X-5  
N40G2X-10Y10I0J10  
N50 G01Y..
```

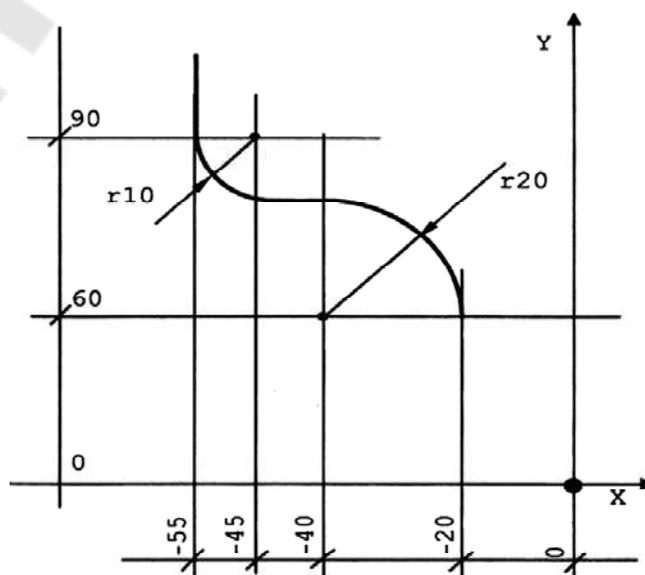


Рис. 3.2. Круговая интерполяция

Координаты начальной точки, запрограммированные в предшествующем кадре, конечной точки и центра окружности должны быть вычислены таким образом, **чтобы разница между начальным и конечным радиусом не превышала 0,01 мм**. Если разница превышает это значение, воспроизводится запись: «Профиль не конгруэнтен», – и окружность не выполняется.

Круговая интерполяция может быть также запрограммирована в приращениях (G91), т. е. с координатами конечной точки и точки центра окружности относительно начальной точки, запрограммированной в предшествующем кадре.

Направление кругового движения (по часовой или против часовой стрелки) определяется, глядя на плоскость интерполяции со стороны положительной полуоси, перпендикулярной к плоскости, в соответствии с рис. 3.3.

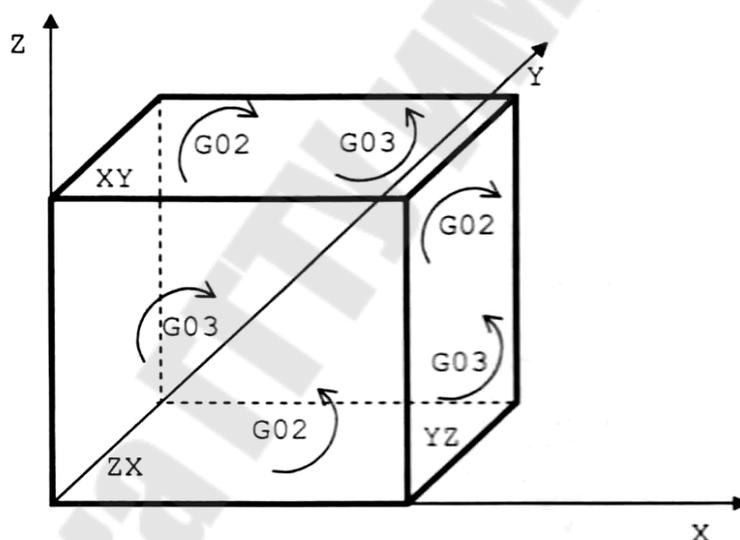


Рис. 3.3. Направление кругового движения

#### 2.7.4. Плоскость интерполяции

Плоскость интерполяции устанавливается при характеристике системы и может быть переопределена при помощи функций G17-G18-G19 или же посредством определения плоскости, образованной парой осей, установленных трехбуквенным кодом DPI (G17 в любом случае присутствует при включении).

Формат:

{G17}

{G18}

{G19},

где **G17** – определяет плоскость круговой интерполяции и компенсации радиуса инструмента, образованную осями 1–2 (XY); **G18** – определяет плоскость круговой интерполяции и компенсации радиуса инструмента, образованную осями 1–3 (XZ); **G19** – определяет плоскость круговой интерполяции и компенсации радиуса инструмента, образованную осями 2–3 (YZ).

*Примечание.* Направление круговой интерполяции (по часовой стрелке или против) определяется при взгляде на плоскость с позитивной стороны перпендикулярной к ней полуоси.

## **2.8. Подготовка УП для токарного станка**

На токарных станках, как известно, обрабатывают детали типа «тел вращения» – цилиндры, конуса, сферы и их сочетания. Во время обработки резец перемещается относительно заготовки по образующим этих поверхностей, а также по радиусу оснований соответствующих поверхностей. Следовательно, траектория инструмента представляет собой сочетание отдельных переходящих друг в друга участков – отрезков прямых и дуг окружностей.

Управляющая программа описывает эту траекторию движения инструмента. Точки перехода участков являются опорными или узловыми точками, координаты которых отображаются в управляющих программах. Для создания программы обработки необходимо определить координаты всех опорных точек. Для знакомства со структурой управляющей программы в целом рассмотрим программу обработки ступенчатого валика (рис. 3.4) из материала – сталь 45.

Из предварительно обработанной ступенчатой заготовки, имеющей припуск на диаметр 1 мм, надо изготовить ступенчатый валик. При таком припуске обработку можно выполнить за один проход. Перемещения на участках 1–2, 2–3, 3–4 выполним на рабочей подаче, а при подводе резца в точку 1, на участках 4–5 и 5–6 – на ускоренных ходах.

Программу обработки назовем STUP3. Она будет выглядеть следующим образом:

;STUP1 (Название программы, не более 6-ти знаков, не должна начинаться с цифр)

N5G79X0 (Перемещение по оси X в абсолютной G79 системе координат в «Нуль» станка)

N10G79Z0 (Перемещение по оси Z в абсолютной G79 системе координат в «Нуль» станка)

N15T8.8M06 (Смена инструмента на T8.8)

N20(UAO1) (Вызов локальной системы координат детали)  
 N25(USG,2,Z-30Z10,X40X0) (Кадр визуализации)  
 N30S200M04 (Включение вращения шпинделя против часовой стрелки)

N35G0Z3 (Быстрое перемещение в точку начала обработки (программы) по оси Z)

N40G90G95X31.5F0.3 (Быстрое перемещение в точку начала обработки (программы) по оси X, программирование подачи)

N45G01Z-10 (Обработка поверхности диаметром 31,5)

N50X35.5 (Перемещение на рабочей подаче на диаметр 35,5)

N55Z-25 (Обработка поверхности диаметром 35,5)

N55G0X38 (Выход на ускоренной подаче на диаметр 38)

N60Z5 (Выход на ускоренной подаче в точку Z5)

В тексте программы можно не указывать нумерацию кадров. Но на начальном этапе рекомендуется указывать номера кадров, так как они используются для облегчения поиска требуемой информации в УП. Также, если в кадре по отношению к предыдущему кадру не менялись команды (тип траектории, значения подач, способ отсчета координат и т. д.), их в этом кадре тоже можно опустить.

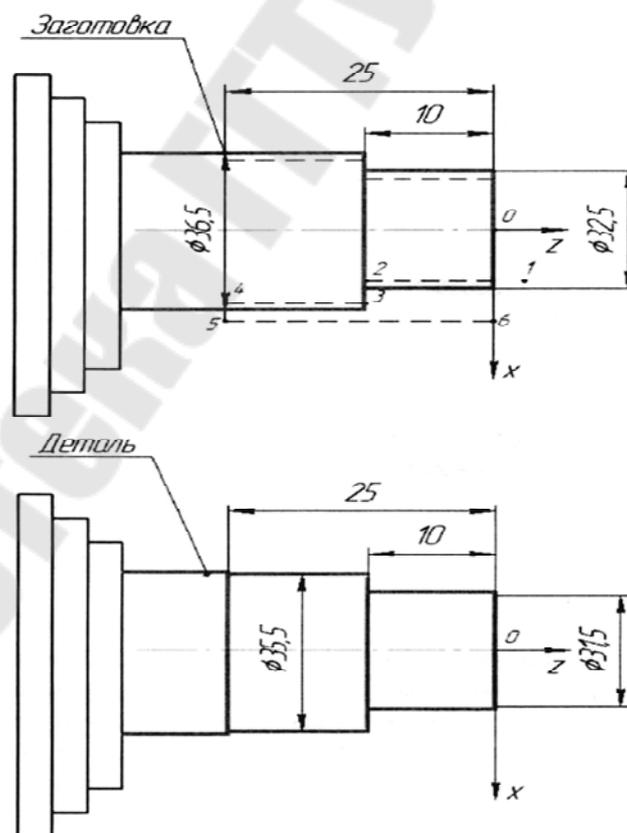


Рис. 3.4. Ступенчатый валик

## 2.9. Визуализация графического отображения обработки детали

Для использования режима графического отображения траектории обработки необходимо в управляющую программу сразу после команды UAO, осуществляющей вызов начальной точки, в которую была помещена локальная система координат («нуль заготовки»), добавить строку, содержащую трехбуквенный код UCG, который воспроизводит этот режим. Команда UCG определяет пределы графического поля с учетом «нуля заготовки». Размеры поля определяются размерами отображения обрабатываемой детали.

На рис. 3.5 изображена заготовка обрабатываемого валика. Поле графической информации о процессе обработки, перемещениях инструментов выделено пунктирными линиями. Ниже приведен пример программы с командой ISO двухмерного отображения графического поля обработки детали (рис. 3.5).

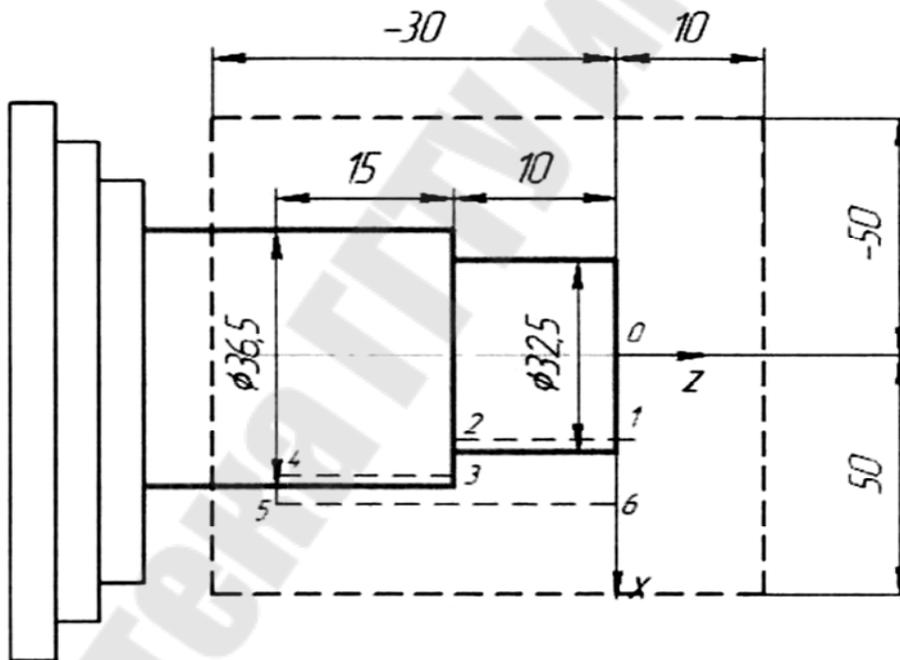


Рис. 3.5. Графическое отображение поля визуализации

```
;STUP2  
N5T8.8M6  
N10(UAO,1)  
N15(UCG,1,X-50X50,Z-30Z10)  
N20G0X31.3Z3S200M4  
N25G01G90G95Z-10F0.5  
N30X35.5
```

N40Z-25  
N45G0X38  
N50Z0  
M5  
M30

**Формат видеокadra: (UCG,1,X-50X50,Z-30Z10),**

где **1** – воспроизведение, не скоординированное с осями; может принимать значение **2** для воспроизведения изображения, скоординированного с осями.

При использовании режима воспроизведения, не скоординированного с осями, сначала появляется контур траектории, потом происходит обработка; при использовании режима воспроизведения, скоординированного с осями, обрисовка контура траектории происходит одновременно с обработкой; **X-50X50** – нижний и верхний пределы отображения графического поля по оси X; **Z-30Z10** – нижний и верхний пределы отображения графического поля по оси Z.

Контурные рабочих ходов обозначаются сплошной линией, контуры быстрых перемещений – пунктирной. Пример программы STUP3D с командой UCG трехмерного отображения графического поля обработки детали представлен на рис. 3.6.

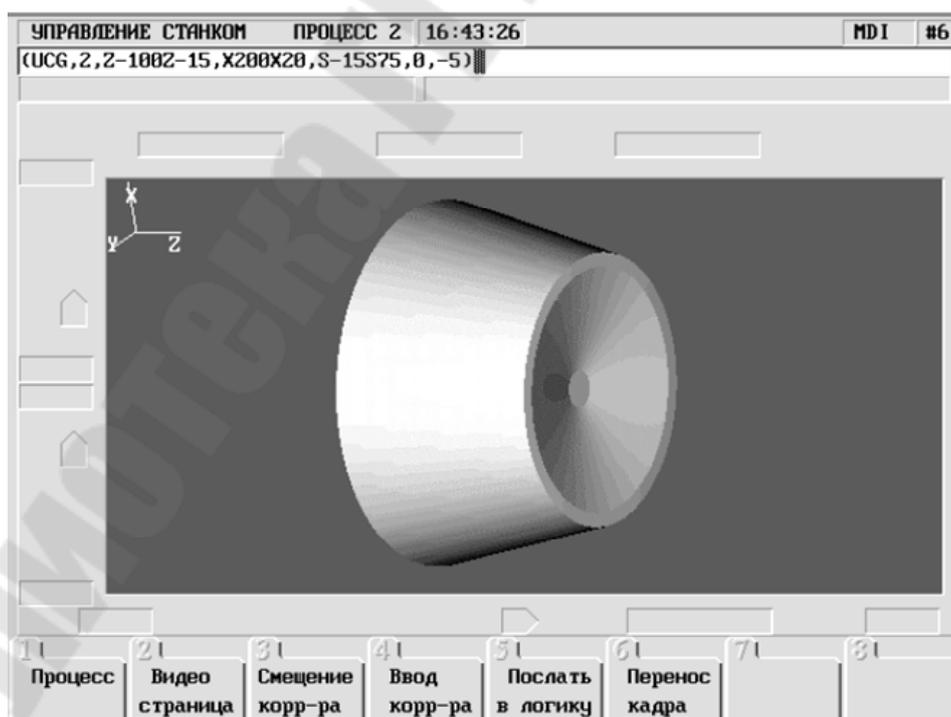


Рис. 3.6. Трехмерное отображение графического поля обработки детали

**Формат видеокадра: (UCG,2,Z-25Z0,X35.5X0,S0S0,0,-5),**  
где **2** – воспроизведение, условно скоординированное с осями. Может принимать значение **1** для воспроизведения, не скоординированного с осями; **Z-25** – определяет название оси абсциссы и максимальную длину вдоль нее для наружного диаметра заготовки. Знак длины указывает направление, в котором должна быть нарисована заготовка, от нуля детали; **Z0** – определяет название оси абсциссы и длину вдоль нее для отверстия, изображаемого влево от торца заготовки. Знак длины для внутреннего диаметра заготовки должен совпадать со знаком длины для наружного диаметра заготовки; **X35.5** – определяет название оси ординаты и значение, которое сформирует максимальный наружный диаметр заготовки; **X0** – определяет название оси ординаты и значение, которое сформирует внутренний диаметр заготовки; **S0** – имя данного параметра всегда должно быть записано буквой «S», а значение определяет угол конуса для наружной образующей тела вращения; **S0** – имя данного параметра всегда должно быть записано буквой «S», а значение определяет угол конуса для внутреннего отверстия заготовки; **0** – определяет горизонтальное направление **оси абсциссы и инструмента** в видеокадре; при вертикальном положении оси абсциссы параметр принимает значения – **1**; **5** – коэффициент, определяющий качество трехмерного изображения. Данный параметр детализации может принимать значения от **минус 1** (минимальное качество) до **минус 5** (максимальное качество).

*2.10. Разработка управляющей программы для обработки ступенчатого валика*

Используя шаблон программы управления VAL2 для обработки ступенчатого валика (рис. 3.7), записать программу с размерами детали согласно табл. 3.7. Номер варианта – по указанию преподавателя.

При защите дать словесные пояснения кадров программы. Набрать на ПК программу обработки в текстовом редакторе «Блокнот». Сохранить ее на съемном носителе USB. Распечатать и предъявить программу преподавателю для защиты и прогона на графическом эмуляторе.

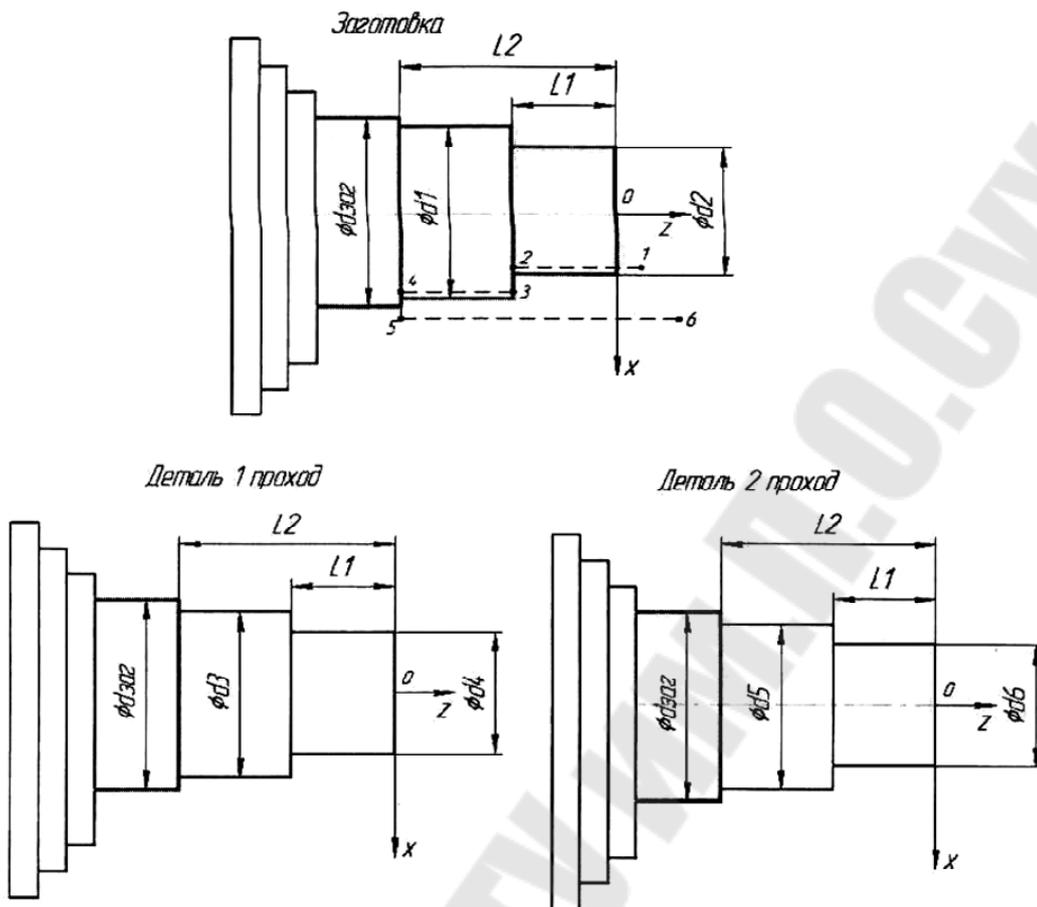


Рис. 3.7. Шаблон программы VAL2 обработки ступенчатого валика

```

;VAL2
N10 T1.1M06
N20 (UAO,1)
N30 (UCG,2,Z-L2 Z0,X(d3ар) X0)
N40 G97S200M04
N50 G01G90G95 Xd4Z5F0.1
N60 Z-L1
N70Xd3
N80Z-L2
N90G0X(d3ар + 3)
N100Z30
N110M05
N120T2.2M06
N130(UAO,1)
N140G97S200M04
N150G01G90G95 Xd6Z5F0.1
N160Z-L1

```

N170Xd5  
 N180Z-L2  
 N190G0X ( $d_{\text{заг.}} +3$ )  
 N200Z0  
 N210M05  
 N220M30

Таблица 3.7

**Размеры детали VAL2**

Вариант	$d_{\text{заг.}}$ , мм	d1, мм	d2, мм	L1, мм	L2, мм	d3, мм	d4, мм	d5, мм	d6, мм
1	48	46	42	30	58	44	40	43,5	39,5
2	50	48	46	32	60	46	42	45,5	41,5
3	52	50	48	34	62	48	44	47,5	43,5
4	54	52	50	36	64	50	46	49,5	45,5
5	56	54	52	38	66	52	48	51,5	47,5
6	58	56	54	40	68	54	50	53,5	49,5
7	60	58	56	42	70	56	52	55,5	51,5
8	62	60	58	44	72	58	54	57,5	53,5
9	64	62	60	46	74	60	56	59,5	55,5
10	66	64	62	48	76	62	58	61,5	57,5
11	68	66	64	50	78	64	60	63,5	59,5
12	70	68	66	52	80	66	62	65,5	61,5
13	72	70	68	54	82	68	64	67,5	63,5
14	74	72	70	56	84	70	66	69,5	65,5
15	76	74	72	58	86	72	68	71,5	67,5

## *Лабораторная работа № 4*

# **ПРИОБРЕТЕНИЕ НАВЫКОВ НАПИСАНИЯ ПРОГРАММ ДЛЯ ТОКАРНОЙ ОБРАБОТКИ ДЕТАЛЕЙ, СОДЕРЖАЩИХ СФЕРИЧЕСКИЕ ПОВЕРХНОСТИ И РЕЗЬБЫ**

### **Цель работы:**

1. Изучить формат управляющих команд при составлении программ с использованием функций линейной, круговой интерполяции при использовании УЧПУ N201M.

2. Изучить формат управляющих команд при составлении программ нарезания резьб при использовании УЧПУ N201M.

3. Изучить трехбуквенные коды, применяемые при составлении программ для токарного станка с УЧПУ N201M.

4. Научиться составлять простейшие программы для токарной обработки сферических поверхностей и для нарезания резьб.

В качестве учебного пособия используются:

1. Методические указания по выполнению данной работы.

2. Руководство оператора (РО УЧПУ NC-201, NC-201M, NC-202, ООО «Балт-Систем») можно скачать на бесплатном доступном сайте [www.bsystem.ru](http://www.bsystem.ru).

При выполнении работы в лаборатории университета РО входит в состав работы № 4.

3. Руководство программиста (РП УЧПУ NC-110, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230, ООО «Балт-Систем») можно скачать на бесплатном доступном сайте [www.bsystem.ru](http://www.bsystem.ru).

При выполнении работы в лаборатории университета РП входит в состав лабораторной работы № 4.

В конце занятия необходимо:

– предоставить для защиты написанные управляющие программы (УП) обработки сферы и резьбы на токарном станке в соответствии с заданием, выданным преподавателем;

– ответить письменно на ряд нижеперечисленных вопросов для подтверждения усвоения материала и зачета работы:

1. Что определяют адресные слова I и J?

2. Как выглядит формат программирования нарезания резьб при токарной обработке?

3. Какой трехбуквенный код используют, чтобы изменить существующую программу или записать новую программу с клавиатуры?

4. Какой трехбуквенный код используют, чтобы изменить имя программы?
5. Какой трехбуквенный код используют, чтобы вызвать список программ из памяти МРх?
6. Какой трехбуквенный код используют, чтобы вызывать указанную программу?
7. Какой трехбуквенный код используют, чтобы активизировать штурвал или его отключить?
8. Какой трехбуквенный код используют, чтобы стереть начальную точку локальной системы координат?
9. Какой трехбуквенный код используют, чтобы удалить корректор инструмента?
10. Какой трехбуквенный код используют, чтобы выбрать УП?
11. Какой трехбуквенный код используют, чтобы отменить выбранную УП?
12. Какой трехбуквенный код используют, чтобы вызывать подпрограмму?
13. Какой трехбуквенный код используют, чтобы повторить часть УП  $N$  раз?
14. Какой трехбуквенный код используют, чтобы закрыть повторение части УП?
15. Какой трехбуквенный код используют, чтобы объявить начальную точку в приращениях относительно текущей начальной точки?
16. Какой трехбуквенный код используют, чтобы определить плоскость интерполяции?

После усвоения материала нижеприведенных методических указаний необходимо сдать тесты и разработать три УП по заданиям № 1–3.

Для сдачи тестов отводится 10 мин.

Результаты тестов заносятся в ведомость оценки результатов сдачи тестов.

#### **МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ № 4**

##### **1. Программирование дуг и окружностей с помощью круговой интерполяции**

**Формат программирования:**

**{G02,G03}[ДРУГИЕ G][ОСИ] I J [СКОРОСТЬ ПОДАЧИ] [ОПЕРАНДЫ КОРРЕКТИРОВКИ][ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ],**

где [G] – смешанные операторы и вспомогательные функции; имеют те же значения, что и в предыдущих случаях; [СКОРОСТЬ ПОДАЧИ] – скорость подачи; [ОСИ] – представлены символом оси и цифровым значением в явной или неявной форме (параметр E). Если ни одна ось не запрограммирована, то выполняемым движением будет полное круговое движение в плоскости интерполяции. Оси могут быть определены неявным образом посредством геометрического элемента – точки. Если координата прибытия равна координате отправления, она может быть опущена; I и J – являются адресными словами, выражающими координаты центра окружности, цифровая часть которых может быть выражена в явной или неявной форме (параметр E). Используемыми символами всегда являются I и J независимо от плоскости интерполяции и всегда присутствуют. Пример круговой интерполяции приведен на рис. 4.1.

```

N10 G1X-20Y60F200
N20G3G91X-20Y20I-20J0
N30G1X-5
N40G2X-10Y10I0J10
N50 G01Y..
    
```

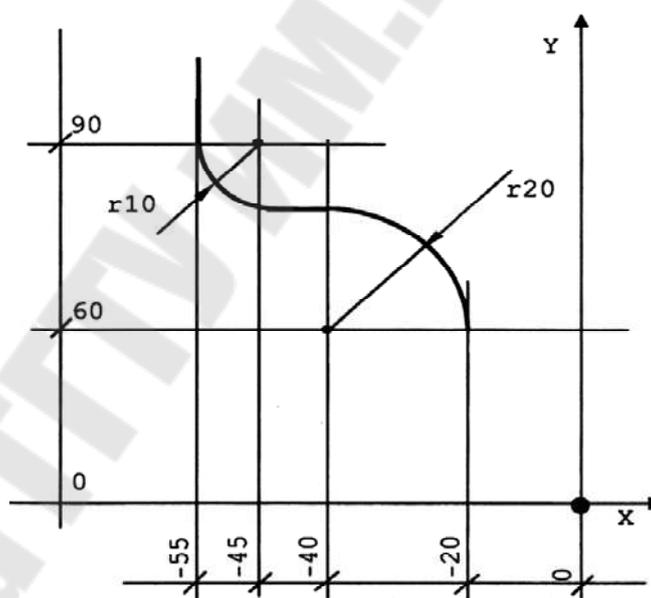


Рис. 4.1. Круговая интерполяция. Максимальная программируемая дуга 360 градусов

Координаты начальной точки, запрограммированные в предшествующем кадре, конечной точки и центра окружности должны быть вычислены таким образом, что бы разница между начальным и конечным радиусом не превышала бы 0,01 мм. Если разница превышает это значение, воспроизводится запись: «Профиль не конгруэнтен», и окружность не выполняется.

Круговая интерполяция может быть также запрограммирована в приращениях, т. е. с координатами конечной точки и точки центра окружности относительно начальной точки, запрограммированной в предшествующем кадре.

Направление кругового движения (по часовой или против часовой стрелки) определяется, глядя на плоскость интерполяции со стороны положительной полуоси, перпендикулярной к плоскости, в соответствии с рис. 4.2.

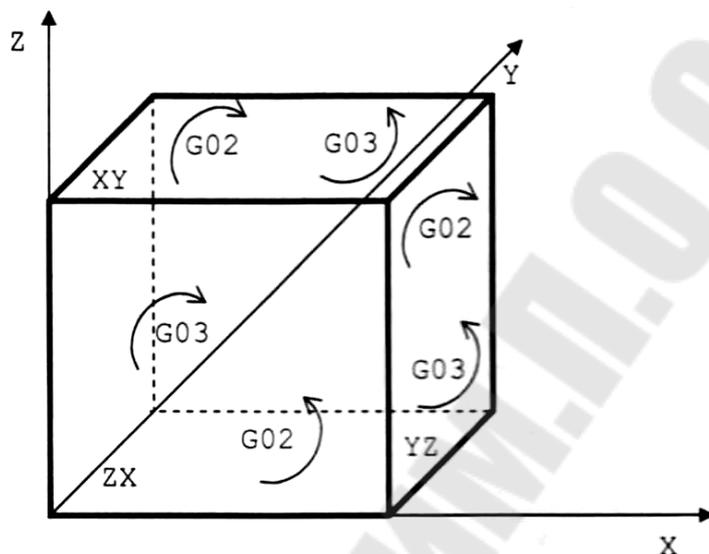


Рис. 4.2. Определение направления кругового движения

## 2. Программирование дуги менее 360 градусов через задание координат конечной точки и радиуса

Формат:

**{G02,G03}[ДРУГИЕ G][ОСИ] R± [СКОРОСТЬ ПОДАЧИ][ОПЕРАНДЫ КОРРЕКТИРОВКИ][ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ],**

где [G] – смешанные операторы и вспомогательные функции; имеют те же значения, что и в предыдущих случаях; [СКОРОСТЬ ПОДАЧИ] – скорость подачи; [ОСИ] – представлены символом оси и цифровым значением в явной или неявной форме (параметр E); оси могут быть определены неявным образом посредством геометрического элемента – точки; R – адресное слово, выражающее радиус дуги окружности, цифровая часть которой может быть выражена в явной или неявной форме (параметр E); знак «+» или «-» перед адресным словом выбирает одно из двух возможных решений: «+» – для дуги до 179.999°; «-» – для дуги от 180° до 359.999°.

*Пример* программирования дуги менее 360 градусов через задание координат конечной точки и радиуса приведен на рис. 4.2.

G02X20Y20R+20F100  
G02X20Y20R-20F100

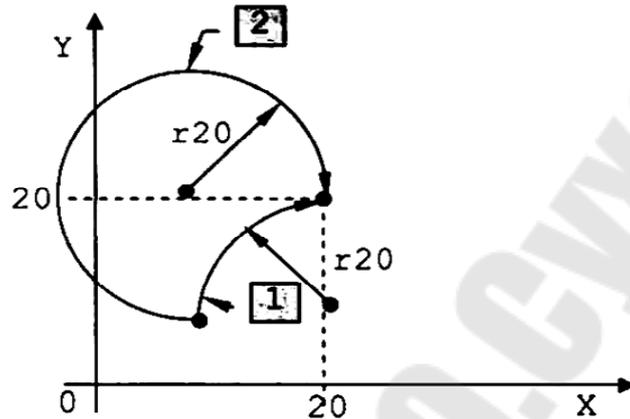


Рис. 4.2. Программирование дуг

### 3. Нарезание резьбы с постоянным или переменным шагом (G33)

Нарезание резьбы с постоянным или переменным шагом (G33) определяет цикл цилиндрического или конического нарезания резьбы с постоянным или переменным шагом. Это движение координируется с вращением шпинделя. Запрограммированные в кадре параметры определяют тип резьбы, которую следует осуществить.

**Формат:**

**G33[ОСИ]K[I][R],**

где [ОСИ] – представлены символом оси и цифровым значением в явной или неявной форме (параметр E); K – представляет шаг резьбы; в случае переменного шага, представляет начальный шаг; должен присутствовать всегда; I – представляет изменение шага; для нарезания резьбы с возрастающим шагом I должна быть положительной, для нарезания резьбы с уменьшающимся шагом должна быть отрицательной; R – представляет отклонение по отношению к угловой позиции нуля шпинделя (в градусах); используется при многозаходной резьбе для того, чтобы не сдвинуть начальную точку.

Во время нарезания резьбы выведены из состояния работы команда «СТОП» и коррекции подачи и скорости вращения шпинделя.

Функция G33 программируется только с датчиком в шпинделе.

Примеры нарезания резьбы с постоянным шагом приведены на рис. 4.3:

- фронтальной (торцевой) резьбы;
- цилиндрическое нарезание резьбы;
- коническое нарезание резьбы.

*Примечание.* Все параметры могут быть выражены цифровым значением в явной или неявной форме через параметры E.

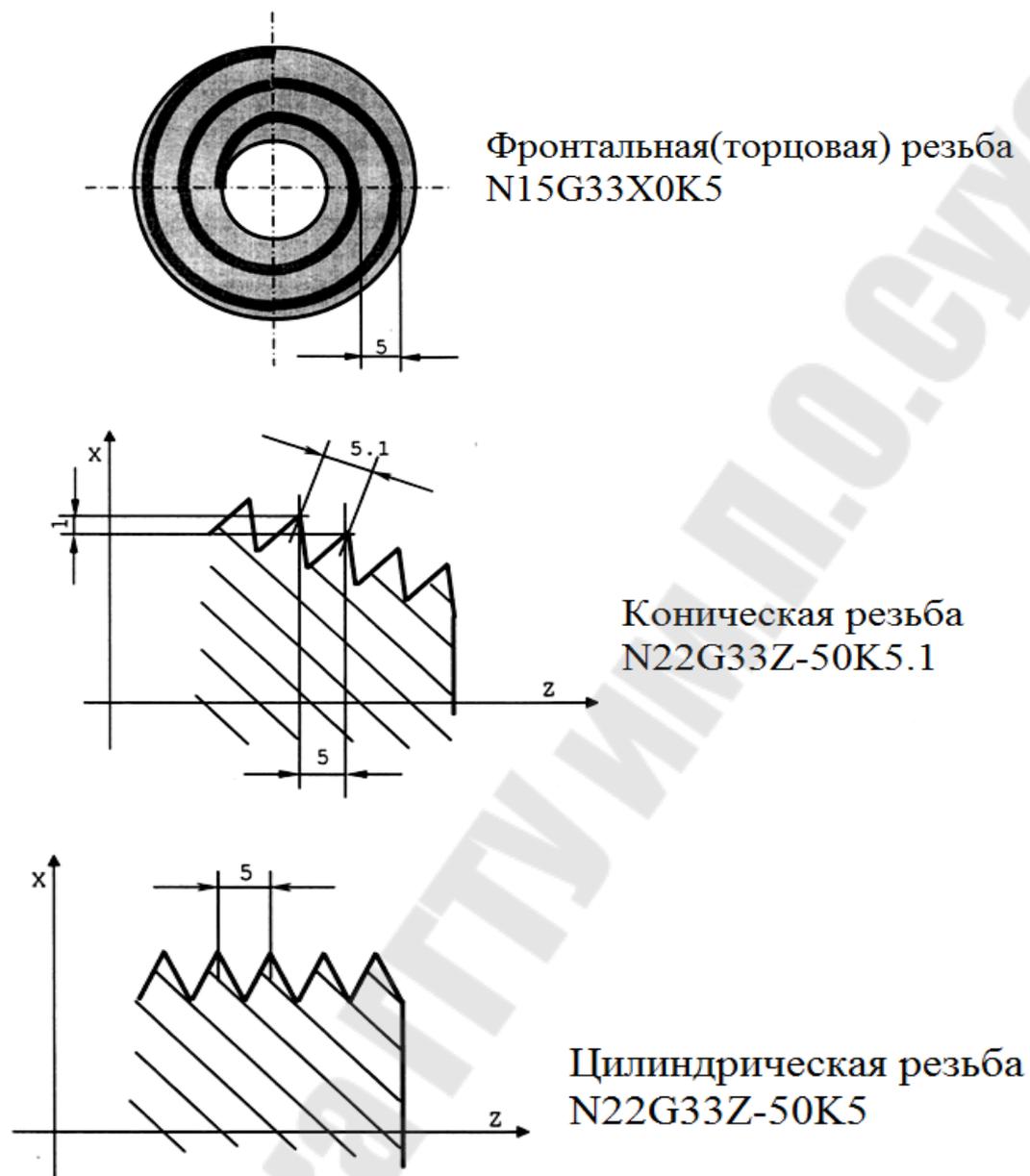
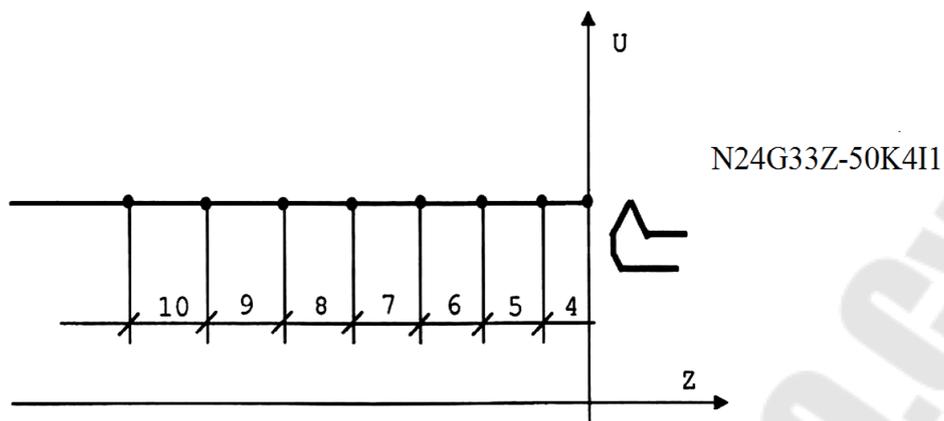


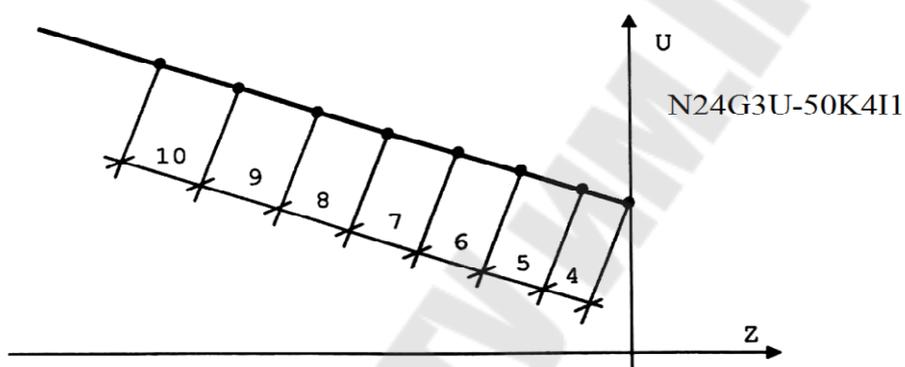
Рис. 4.3. Нарезание резьбы с постоянным шагом

Примеры нарезания резьбы с переменным шагом приведены на рис. 4.4:

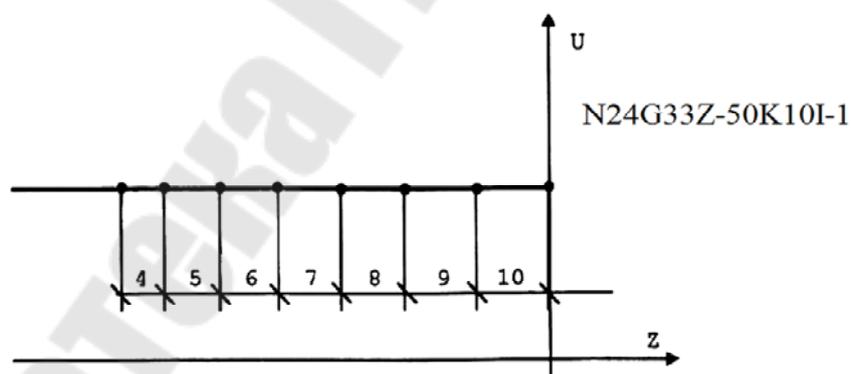
- цилиндрическое нарезание резьбы с возрастающим шагом;
- коническое нарезание резьбы с возрастающим шагом;
- цилиндрическое нарезание резьбы с уменьшающимся шагом.



Цилиндрическая резьба с увеличивающимся шагом



Коническая резьба с увеличивающимся шагом



Цилиндрическая резьба с уменьшающимся шагом

Рис. 4.4. Нарезание резьбы с переменным шагом

Во время нарезания резьбы с уменьшающимся шагом начальный шаг, изменения шага и длина нарезания резьбы должны быть такими, чтобы шаг не становился равным нулю до достижения конечного размера. Для проверки применяется формула

$$I \leftarrow \frac{K2}{(ZK - ZN)}, \quad (4.1)$$

где **I** – максимальное изменение шага; **K** – начальный шаг; **ZK** – координата конечной точки; **ZN** – координата начальной точки; **(ZK – ZN)** – длина нарезания резьбы.

*Пример* нарезания резьбы с 3-мя заходами:

.....  
N37G33Z3K6            первый заход

.....  
N41G33Z3K6R120    второй заход

.....  
N45G33Z3K6 R240    третий заход

Функция R дает команду системе для размещения осей в угловой позиции, которая меняется в зависимости от запрограммированной величины R. Таким образом, представляется возможным программировать одну начальную точку для различной нарезки в отличие от других систем, в которых для осуществления многозаходной резьбы необходимо сместить начальную точку каждой нарезки на величину, равную шагу, разделенному на количество заходов.

#### 4. Цикл нарезания резьбы с помощью файла FIL

Цикл нарезания резьбы FIL позволяет вам программировать в одном кадре цилиндрическую или коническую многопроходную резьбу. Как правило, перед циклом FIL рекомендуется указывать плоскость, в которой нарезается резьба с помощью трехбуквенного кода (DPI, Z, X), при этом первой указывается координата оси, вдоль которой нарезается резьба.

Допустимый формат цикла следующий:

**(FIL,Z...,X...,K...,L...,R...,T...,P...,a...,b...),**

где Z – конечный размер Z; X – конечный размер X; K – шаг; L – число проходов черновой и чистовой обработки, например, L5.2; R – расстояние между инструментом и деталью (по умолчанию, R=1); T – 3-цифровой код, определяющий тип нарезания резьбы (по умолчанию, T000).

Цифра 1:

- 0 – нарезание с конечным пазом (канавкой);
- 1 – нарезание без конечного паза (канавки).

Цифра 2:

- 0 – внешнее нарезание резьбы;
- 1 – внутреннее нарезание резьбы.

Цифра 3:

- 0 – метрическое нарезание резьбы;
- 1 – дюймовая резьба;
- 2 = нестандартное нарезание резьбы с глубиной и углом, определяемыми параметрами **a** и **b**;
- P – число заходов резьбы (по умолчанию, P = 1);
- a – угол резьбы (только для нестандартной, отличный от 60 град.);
- b – глубина резьбы (только для нестандартной).

*Примечания.* **1.** Устройство управления автоматически вычисляет позиции, скользя вдоль края резьбы, так что часть результирующей стружки остается постоянной. Для резьбы с несколькими заходами вы должны только определить шаг каждого витка. Устройство управления выполняет каждый проход для каждого захода перед выполнением последующего прохода. Каждый заход выполняется без перемещения начальной точки каждой резьбы, но на расстоянии от углового нуля шпинделя. **2.** Для резьбы с конечным пазом вы должны запрограммировать теоретический конечный Z, так как фиксированный цикл обеспечивает увеличение хода, равное половине шага. **3.** В резьбе без конечного паза инструмент достигает программируемого размера и затем перемещается обратно с конической резьбой вдоль обратного диаметра.

Резьба без конечного паза не может быть получена в кадровом режиме.

*Пример* цикла нарезания резьбы приводится на рис. 4.5.

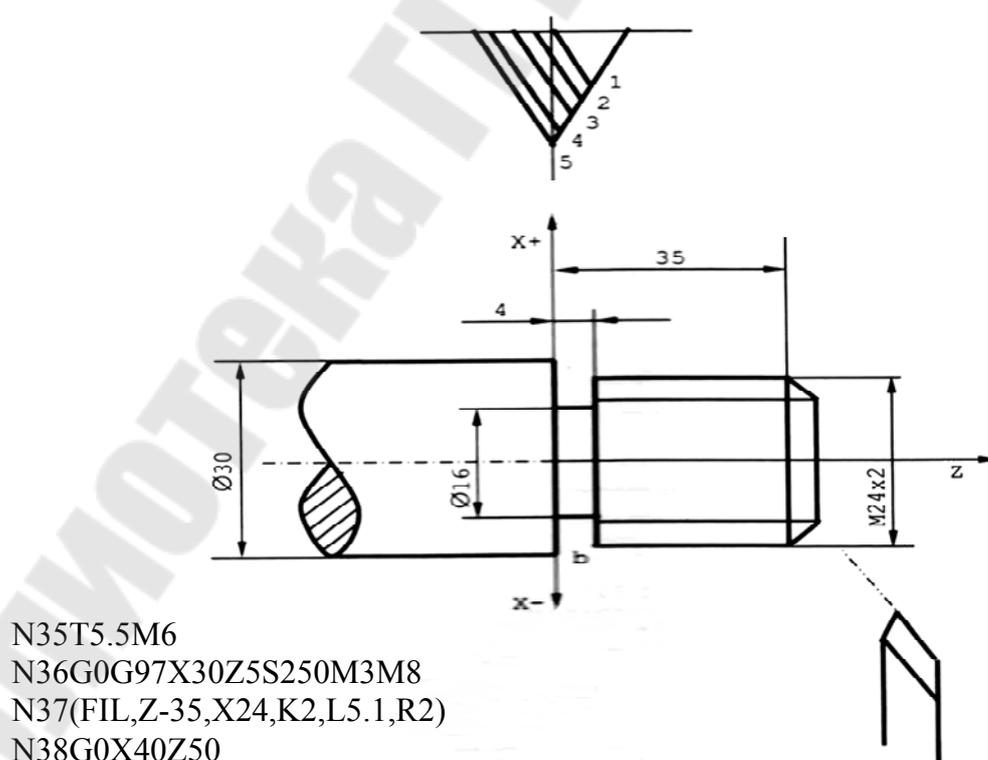


Рис. 4.5. Нарезание резьбы с помощью файла FIL

## 5. Трехбуквенные коды, используемые при работе с УЧПУ NC201M

Трехбуквенные коды УЧПУ в зависимости от их функций могут быть разделены на четыре группы:

1) трехбуквенные коды, используемые в режиме «КОМАНДА» и задаваемые с клавиатуры;

2) трехбуквенные коды, которые могут быть использованы в кадрах УП, присвоены из УП или введены с клавиатуры;

3) трехбуквенные коды, используемые для управления оборудованием;

4) трехбуквенные коды, которые используются внутри УП или вводятся с клавиатуры при испытаниях УП.

### 5.1. Трехбуквенные коды режима «КОМАНДА»

Трехбуквенные коды, используемые в режиме «КОМАНДА» и задаваемые с клавиатуры, представлены в табл. 4.1.

Таблица 4.1

Трехбуквенные коды режима «КОМАНДА»

Код	Формат	Функции
EDI	EDI,имя/MEM	Вызов редактора для того, чтобы изменить существующую программу или записать новую программу с клавиатуры
DEL	DEL,имя/MEM	Удаляет программу из устройства памяти
COP	COP,имя/MEM, имя/MEM	Копирует указанную программу из одной памяти МРх в другую память МРх под любым именем
	COP,имя/MEM, /TY	Копирует указанную программу из памяти МРх на периферийное устройство
	COP,/TY,имя/MEM	Копирует программу с периферийного устройства в память МРх
REN	REN,имя/MEM, имя1(имя1/MEM)	Изменяет имя программы
DIR	DIR/MEM	Показывает список программ в памяти МРх
FOR	Имя/MEM,кол-во строк	Создает файл фиксированной длины и формирует поля файлов корректоров, продолжительности срока службы инструмента, начальных точек
ATT	ATT,имя,100	Защищает программу от записи. Убирает защиту. Делает программу нечитаемой в 01К
	ATT,имя,0	
	ATT,имя,0001	

Код	Формат	Функции
VLT	VLT	Визуализирует список файлов секции 2 файла FCRSYS
RUN	RUN,имя	Вызывает указанную программу
DIF	(DIF,имя/MPx, /MPx)	Проверяет разницу между программами

### 5.2. Трехбуквенные коды в кадрах УП

Трехбуквенные коды, которые могут быть использованы в кадрах УП, присвоены из УП или введены с клавиатуры, представлены в табл. 4.2.

Таблица 4.2

#### Трехбуквенные коды в кадрах УП

Код	Формат	Функция
E	EN[.тип]=значение	Определяет числовые переменные с одним из следующих типов: BY=байт; IN=целое число; LI=длинное целое число; RE=действительное; LR=длинное действительное; N – номер параметра
O	oN=значения координат или переменных	Определяет геометрический элемент как точку начала отсчета; N – номер элемента
P	PN=значения координат или переменных	Определяет геометрический элемент как точку; N – номер элемента
l	LN=значения координат или переменных	Определяет геометрический элемент как прямую; N – номер элемента
c	CN=значения координат или переменных	Определяет геометрический элемент как окружность; N – номер элемента
TMR	TMR=значение	Определяет время, затрачиваемое на движение при G04 или в фиксированных циклах (выражается в секундах)
SSL	SSL=величина	Определяет предельную скорость шпинделя
RTR	RTR=1 RTR=0	Дробление стружки разрешено Дробление стружки запрещено
SRT	SRT=значение	Определяет шаг дробления стружки
PRT	PRT=значение	Определяет время останова при дроблении стружки
VRT	VRT=значение	Определяет скорость дробления стружки
UOV	UOV=1 UOV=0	Определяет значение припуска. Отмена припуска

Код	Формат	Функция
VOG	VOG=значение	Определяет величину перемещения, выполняемого в режиме ручных фиксированных перемещений
RTA	RTA=значение	Определяет изменение величины щупа для оси X (аттестация щупа)
RTO	RTO=значение	Определяет изменение величины щупа для оси Y (аттестация щупа)
ERF	ERF=значение	Определяет допустимую ошибку формы
MCD	MCD=значение	Определяет максимальное отклонение направляющих косинусов в движении
USB	USB=1 USB=0	Выполнение кадров с символом "/" (пропуск). Пропуск кадров с символом "/"
UVR	UVR=1 UVR=0	Выполнение УП в режиме быстрого хода. Отмена вышеназванного режима
URL	URL=1 URL=0	Разрешение работы переключателя <b>КОРРЕКТОР ПОДАЧ («ЖОС»)</b> для управления быстрым ходом (G00). Отмена вышеназванного режима
USO	USO=1 USO=0	Подтверждение M01. Отмена M01
UCV	UCV=N	Определяет тип вывода на экран осевых значений для видеостраниц #1 и #7: UCV=0 – рассчитанные величины осей; UCV=1 – значения датчиков; UCV=2 – ошибки позиционирования; UCV=3 – остаток пути в кадре
RAP	RAP=0 RAP=1	Автоматический возврат на профиль после перемещения вручную, последовавшего после «Стопа» с выбором оси. Автоматический возврат на профиль после перемещения вручную, последовавшего после «Стопа» по пути ручного перемещения
UAS	UAS=1 UAS=0	Отключение осей (блокировка привода). Отмена вышеназванного режима
RMS	RMS=значение	Определяет процент изменения скорости при выводе инструмента из отверстия в цикле G84
UEP	UEP=1 UEP=0	Отменяет скоростную компенсацию. Разрешает скоростную компенсацию

Код	Формат	Функция
SA	SAN=значение	Определяет из УП значение сигнала пакета «А»; N – номер параметра
SK	SKN=значение	Определяет из УП значение сигнала пакета «К»; N – номер параметра
SYVAR	SYVARN=значение	Определяет значение переменных; N – номер параметра
TIM	TIMN=значение	Определяет из УП системное время. TIM=0 сбрасывает часы; N – номер параметра
TOT	TOTN=значение	Определяет из УП суммарное время; N – номер параметра
VOL	VOL=1 VOL=0	Активизация штурвала. Отключение штурвала
MBR	MBR=1 MBR=0	Активизация обратного прослеживания профиля. Отмена обратного прослеживания профиля
ORA	ORA,N,X...,Y...,Z...	Определяет абсолютную начальную точку по осям; N – номер начальной точки. Для определения начальных точек в альтернативных единицах измерения номер должен быть взят с отрицательным знаком (-N)
CAO	CAO,N	Стирает начальную точку; N – номер начальной точки. Если N отсутствует, удаляются все записи файла начальных точек

### 5.3. Трехбуквенные коды для управления оборудованием

Трехбуквенные коды, используемые для управления оборудованием, представлены в табл. 4.3.

Таблица 4.3

#### Трехбуквенные коды для управления оборудованием

Код	Формат	Функция
VOA	VOA,N	Воспроизводит начальную точку; N – номер начальной точки
URP	URP,N	Определяет угол вращения детали; N – плоский угол вращения вокруг оси в градусах
VOL	VOL=1 VOL=0	Активизация штурвала. Отключение штурвала

Код	Формат	Функция
UCG	UCG,N,AXIS1I,AXIS1S, AXIS2I,AXIS2S[AXIS3]	Определяет параметры инициализации для графического экрана: N=1 – просмотр осей, не входящих в систему координат; N=2 – просмотр осей, входящих в систему координат; AXIS1I – нижний предел оси X; AXIS1S – верхний предел оси X; AXIS2I – нижний предел оси Y; AXIS2S – верхний предел оси Y; AXIS3 – ось, перпендикулярная рабочей плоскости
CLG	CLG	Очищает графический экран
DCG	DCG	Запрещает графический экран (всегда после CLG)
CAC	CAC,N	Удаляет корректор инструмента; N – номер корректора. Если N не определен, команда удаляет весь файл
SPG	SPG,имя	Выбирает УП
REL	REL	Сбрасывает выбор УП
DPT	DPT,QG,QS,VM	Определяет параметры шупа: QG – величина приближения (расстояние от условной точки шупа); QS – величина безопасности (максимальное перемещение от точки касания шупа); VM – скорость, выраженная в мм/мин
RCM	RCM	Разрешает запомненный поиск
ERM	ERM	Запрещает запомненный поиск
PTM	PTM,час:мин.: [сек.]	Загружает системные часы требуемым временем
VIC	VIC,N	Визуализирует содержание таймерной переменной (TIMX); N – номер переменной. На дисплее визуализируется: VIC, имя переменной, часы, минуты, секунды
ESE	ESE,N	Выполнение УП до кадра с номером N, например: ESE,24
DIS	DIS,переменная	Воспроизведение переменной
EVA	EVA,выражение	Вычисляет выражение и воспроизводит его на экране

Код	Формат	Функция
UCA	UCA,n,Z,X	Модифицирует инкрементально величину диаметра (X) и/или длины (Z) корректора n
MBR	MBR=1 MBR=0	Активизация обратного прослеживания профиля. Отмена обратного прослеживания профиля
DBT	(DBT,T,тип)	Активизация функции осциллографирования (задается с клавиатуры): T – временной интервал между осциллографированием; «тип» – тип осциллографирования: 1 – однократный; 2 – непрерывный
GSE	(GSE,имя переменной, номер интерполятора, адрес, формат, нижний предел, верхний предел) (GSE,параметр,имя оси, нижний предел, верхний предел)	Определение параметров осциллографирования: «адрес» – адрес переменной; «формат» – формат переменной (например: LI, LR, BU и т. п.); «нижний предел» – нижний предел значения переменной; «верхний предел» – верхний предел значения переменной «параметр» – V или E: V – скорость; E – ошибка

#### 5.4. Трехбуквенные ходы для испытания УП

Трехбуквенные коды, которые используются внутри УП или вводятся с клавиатуры при испытаниях УП, представлены в табл. 4.4.

Таблица 4.4

#### Трехбуквенные коды для испытания УП

Код	Формат	Функция
CLS	(CLS,имя подпрограммы)	Вызывает подпрограмму
BNC	(BNC,метка)	Выполняет безусловный переход к метке
BGT	(BGT,VAR1,VAR2,метка)	Переходит, если VAR1 > VAR2
BLT	(BLT,VAR1,VAR2,метка)	Переходит, если VAR1 < VAR2
BEQ	(BEQ,VAR1,VAR2,метка)	Переходит, если VAR1 = VAR2
BNE	(BNE,VAR1,VAR2,метка)	Переходит, если VAR1 ≠ VAR2
BGE	(BGE,VAR1,VAR2,метка)	Переходит, если VAR1 ≥ VAR2
BLE	(BLE,VAR1,VAR2,метка)	Переходит, если VAR1 ≤ VAR2
EPP	(EPP,метка1,метка2)	Выполняет часть УП между меткой «1» и меткой «2»

Код	Формат	Функция
RPT	(RPT,N)	Повторяет часть УП N раз ( $K < 99$ ). Описание части УП начинается после блока, содержащего RPT, и заканчивается блоком, содержащим код ERP
ERP	(ERP)	Закрывает повторение части УП
UAO	(UAO,N)	Выбор абсолютной начальной точки. N – номер абсолютной начальной точки, задается с клавиатуры
UOT	(UOT,N,X...,Y...,Z...)	Определяет временную начальную точку. N – номер абсолютной начальной точки, задается с клавиатуры
UIO	(UIO,X...,Y...)	Объявляет начальную точку в приращениях относительно текущей начальной точки
MIR	MIR,X,Y) (MIR)	Определяет зеркальное отображение для объявленных осей. Отмена зеркального отображения
URT	(URT,угол)	Поворачивает плоскость на угол, относительно текущей начальной точки. Отмена поворота плоскости
SCF	(SCF,N[,ось])	Масштабный коэффициент для объявленных осей, N – масштабный коэффициент. <i>Примечание.</i> Если оси не определены, масштабный коэффициент устанавливается для всех осей
RQO	(RQO,N,ось...)	Переквалификация начальной точки для осей, определенных в УП; N – номер начальной точки
RQU	(RQU,NUT,NCOR...,Z...,K...)	Переквалификация инструмента: NUT – номер инструмента; NCOR – номер корректора. Изменяет текущие корректоры и файл корректоров
RQP	(RQP,NUT,NCOR,Z...,K...)	Изменяет корректоры Z и/или K, определенные в объявлении. Файл корректоров не изменяется
DPI	(DPI,AXIS1,AXIS2)	Определяет плоскость интерполяции; ось1, ось2 – оси, имена которых определяют плоскость
DTL	(DTL,ось1,ось2)	Определяет при позиционировании величину допуска для программированных осей (отличную от величин, объявленных в файле характеристики)

Код	Формат	Функция
DLO	(DLO,AXIS+AXIS-)	Определяет рабочее поле программируемых осей (максимальный и минимальный предел)
DIS	(DIS,переменная)	Воспроизводит на экране переменную
TOF	(TOF,>0)	Объявляет инструмент «вне использования»; N – номер инструмента
UCG	(UCG,N,AXIS1I AXIS1S, AXIS2I AXIS2S,[AXIS3])	Определяет параметры графического экрана: N-1 – воспроизведение с отключенными осями; N-2 – воспроизведение с подключенными осями; N-3 – одновременное воспроизведение движения, не скоординированного с осями (зеленый цвет следа) и скоординированного с осями (розовый цвет следа)
CLG	(CLG)	Очищает область графического экрана дисплея
DCG	(DCO)	Запрещает графический экран (должен быть запрограммирован после CLG)
DSA	(DSA,N,X-X+,Y-Y+)	Определяет пределы защищенной области: N – номер области; X- – нижний предел оси X; X+ – верхний предел оси X; Y- – нижний предел оси Y; Y+ – верхний предел оси Y
ASC	(ASC,N)	Разрешает защищенную область; N – номер области
DSC	(DSC,K)	Запрещает защищенную область; N – номер области
DPT	(DPT,QA,QS,VM)	Определяет параметры щупа: QA – величина подхода; QS – величина безопасности; VM – скорость измерения
DLY	(DLY,N)	Определяет выдержку на указанный промежуток времени; N – выдержка времени в секундах (максимум 32 с)
REL	(REL)	Сброс выбора УП

Код	Формат	Функция
UAV	(UAV,1,XC,PV,r)	Определяет виртуальные оси P и V, r – минимальный радиус
	(UAV,2,B,W,r)	Определяет виртуальный оси B и W, r – радиус цилиндра
	(UAV,0)	Запрещает виртуальные оси
CTL	(CTL,F)	Активизирует конфигурацию фрезерного варианта ПрО
	(CTL,T)	Активизирует конфигурацию токарного варианта ПрО
	(CTL,0)	Возвращает управление к исходному варианту ПрО
DAM	(DAM,имя оси)	Определяет ось шпинделя
TGL	(TGL,Z–величина,X–величина,K–ширина инструмента)	Цикл нарезания пазов: Z – конечный размер паза; X – внутренний диаметр; K – ширина инструмента
FIL	(FIL,Z...,X...,K...,L...,R...,T...,P...,a...,b...)	Цикл нарезания резьбы: Z... – конечный размер Z; X... – конечный размер X; K... – шаг; L... – число проходов черновой и чистовой обработки, например: L11.2; R... – расстояние между инструментом и деталью; T... – 3-цифровой код, определяющий тип нарезания резьбы
DFP	(DFP,n)	Определяет номер профиля (1–8), который вызывается во время циклов черновой и чистовой обработки
EPF	(EPF)	Закрывает определение профиля
SPA	(SPA,a,n,1,x,z)	Цикл черновой обработки, параллельной к оси «а»: а – ось x или z; n – номер профиля; 1 – число проходов; x – припуск по оси x; z – припуск по оси z. SPA не может быть применена к немоно-тонным профилям

Код	Формат	Функция
SPF	(SPF,a,n,1,x,z)	Цикл черновой обработки, параллельной к оси «а» с предварительной чистовой обработкой: а – ось x или z; n – номер профиля; 1 – число проходов; x – припуск по оси x; z – припуск по оси z. SPF не может быть применена к немонотонным профилям
SPP	(SPP,n,1,z1,z2,x1,x2)	Цикл черновой обработки, параллельной к профилю: z1 – припуск по z; z2 – первоначальный припуск по z; x1 – припуск по x; x2 – первоначальный припуск по x
CLP	(CLP,n)	Вызов цикла чистовой обработки; n – номер профиля

## 6. Токарная обработка сферической поверхности

Токарный станок с ЧПУ позволяет получать геометрически правильные элементы сферической поверхности на обрабатываемых деталях. Формообразование сферических поверхностей (впрочем, как и конусных) требует согласованного перемещения резца по двум координатам X и Z. Согласованная работа поперечного и продольного приводов подач обеспечивается УЧПУ. Движение резца по дуге, являющейся образующей сферической поверхности, определяется командой **G02** (по часовой стрелке) или командой **G03** (против часовой стрелки). Как и в команде перемещения по прямой указываются координаты конечной точки перемещения (X,Z), однако дополнительно нужна информация о координатах расположения центра дуги (I,J). Кроме этого требуется применение команды **G18**, дающей информацию о плоскости обработки (X,Z). Поскольку на изучаемом токарном станке только одна плоскость обработки, в которой может лежать дуга окружности, эту команду можно не применять, так как она определяется при загрузке исходных данных УЧПУ.

Итак, формат команды при обработке дуги окружности:

**G02(G03)X...,Z...,I...,J...,**

где **X, Z** – координаты конечной точки дуги; **I, J** – координаты центра дуги (при задании размеров в абсолютной системе координат – относительно нулевой точки заготовки). Направление оси **I** совпадает с направлением оси **Z**, а направление оси **J** совпадает с направлением оси **X**.

*Пример.* Рассмотрим пример написания управляющей программы (УП) SFERA для обработки сферической поверхности за один проход (рис. 4.6).

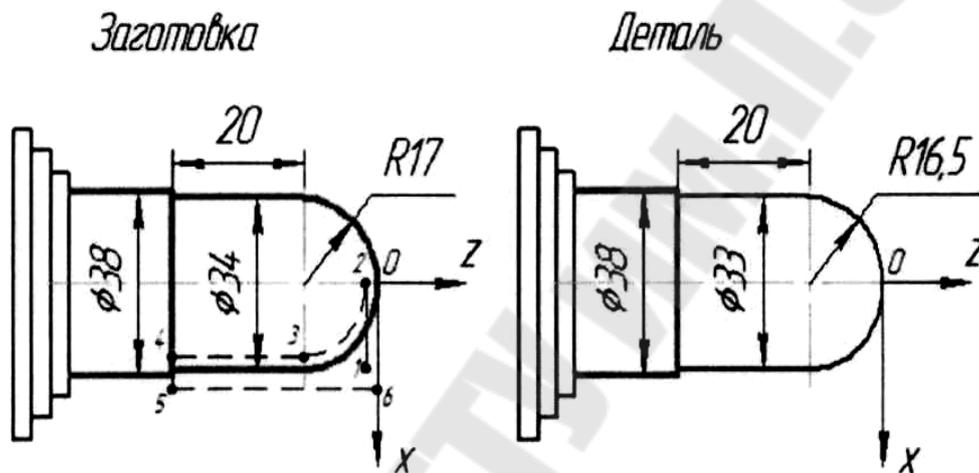


Рис. 4.6. Токарная обработка сферической поверхности

;SF1 (Наименование программы «Сфера»)  
N5T1.1M6 (Смена инструмента)  
N10(UAO,1) (Вызов локальной системы координат детали)  
N15G97S200M4M08 (Включение вращения шпинделя и эмульсии)  
N20G0G90G95X34Z-0.5F0.5 (Ускоренное перемещение в точку 1  
программирование подачи 0.5 мм/об)  
N25G01X0 (Линейное перемещение с рабочей подачей из точки 1  
в точку 2)  
N30G02X33Z-17I-17J0 (Перемещение по дуге по часовой стрелке  
из точки 2 в точку 3)  
N35G01 Z-37 (Линейное перемещение с рабочей подачей из точки 3  
в точку 4)  
N40G00 X40 (Быстрое перемещение из точки 4 в точку 5)  
N45Z0 (Быстрое перемещение из точки 5 в точку 6)  
N50M05 (Выключение вращения шпинделя, эмульсии)  
N55M30 (Сброс программы, возвращение на ее начало).

### Задание 1

Используя шаблон управляющей программы сфера (SFn) для обработки валика со сферической поверхностью (рис. 4.7), записать программу с размерами согласно табл. 4.5. Номер варианта  $n$  – по указанию преподавателя. Быть готовым дать словесные пояснения кадров программы.

Набрать на ПК программу обработки в текстовом редакторе «Блокнот». Записать на флэш-память, распечатать и предъявить преподавателю на проверку программы на эмуляторе.

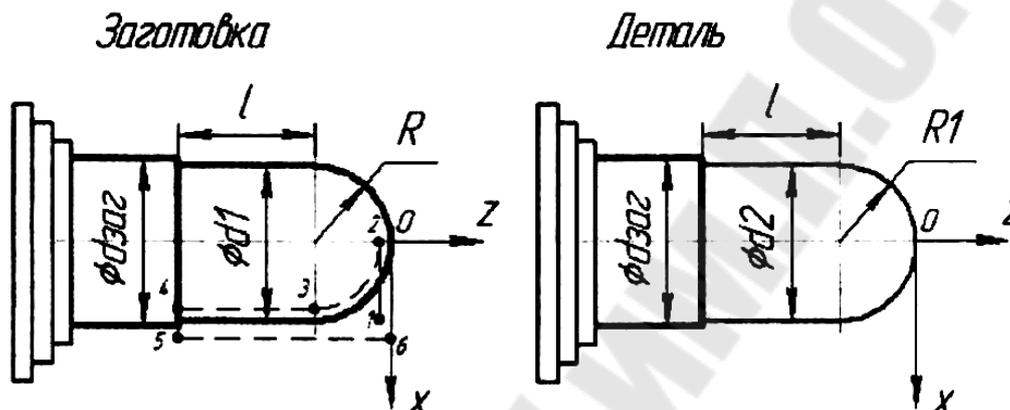


Рис. 4.7. Токарная обработка сферической поверхности

Таблица 4.5

#### Исходные данные для УП SFERA

Вариант	$d_{\text{заг}}$	$d1$	$d2$	$R$	$R1$	$l$
1	36	34	33	17	16,5	28
2	38	36	35	18	17,5	29
3	40	38	37	19	18,5	30
4	42	40	39	20	19,5	31
5	44	42	41	21	20,5	32
6	46	44	43	22	21,5	33
7	48	46	45	23	22,5	34
8	50	48	47	24	23,5	35
9	52	50	49	25	24,5	36
10	54	52	51	26	25,5	37
11	56	54	53	27	26,5	38
12	58	56	55	28	27,5	39
13	60	58	57	29	28,5	40
14	62	60	59	30	29,5	41
15	64	62	61	31	30,5	42

Примечание. 1. Все размеры указаны в мм. 2. « $n$ » – равно номеру варианта.

## 7. Нарезание резьбы

Нарезание резьбы резцом можно выполнять на цилиндрических и конических поверхностях. На токарных станках с ЧПУ формообразование винтовой поверхности резьбы обеспечивается согласованием перемещения резца по оси  $Z$  с вращением шпинделя с заготовкой. Эти согласованные движения контролируются датчиком обратной связи в приводе главного движения. Переход к нарезанию резьбы осуществляется с помощью команды **G33**.

**Формат** управляющей команды для нарезания резьб с постоянным шагом: **G33Z..K..R..**,

где **Z** – длина нарезаемой резьбы, например, Z-10 при длине резьбы 10 мм; **K** – шаг резьбы, например, K2 для резьбы с шагом 2 мм; **R** – параметр многозаходных резьб, представляет собой смещение угловой позиции шпинделя (в градусах), например, R 120 для второго захода и R 240 для третьего захода, трехзаходной резьбы.

*Пример.* Рассмотрим пример управляющей программы **REZVn** для нарезания метрической резьбы M42\*1,5 на цилиндрической поверхности рис. 4.8. Заготовка имеет технологический диаметр ( $\varnothing 41,92$  мм) под нарезание резьбы резцом, заходную фаску и канавку для выхода резца. Внутренний диаметр резьбы – 40,376 мм, **n** – порядковый номер варианта.

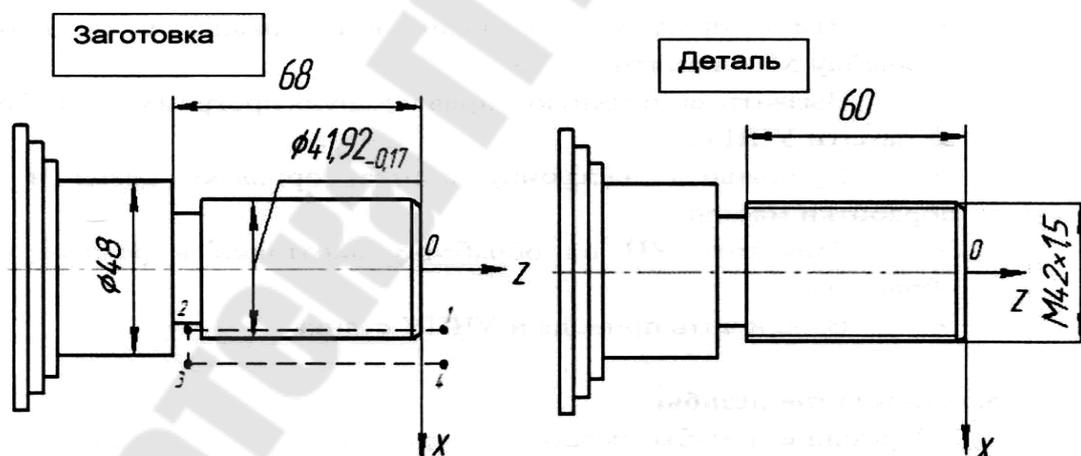


Рис. 4.8. Нарезание метрической резьбы

Для обеспечения качества обработанной поверхности резьбу желательно нарезать в несколько проходов. Снимаемый припуск (по номиналу) определяется как  $d_{нар} - d_{вн}$ . Примем технологию обработки за два прохода, разделив припуск на три равные части. Координаты инструмента по оси  $Z$  выбираем, исходя из соображений, что в начале резьбы необ-

ходимо добавлять технологический запас хода для разгона (торможения) приводов. Поэтому примем: для точек 1 и 4 –  $Z = 5$  мм, для точек 2 и 3 –  $Z = -60$  мм, где 60 – длина резьбы.

;REZB1

N5T2.2 M06

N10(UAO,1)

N15(UCG,2,Z-62Z0,X40.9X0,S0S0, 0, -5)

N20G97S100M04

N25G00G90Z4X41.00F1.5 – линейное перемещение в точку 1 по оси X для нарезания при первом проходе.

N30G33Z-60K1.5 – нарезание резьбы с шагом 1,5 мм до выхода резца в канавку.

N35G0X44 – быстрое перемещение в точку 3.

N40Z5 – быстрое перемещение в точку 4.

N45X39.86 – перемещение в начальную точку 1 нарезания при втором проходе.

N50G33Z-60K1.5 – нарезание резьбы с шагом 1,5 мм при втором проходе.

N55G0X44 – быстрое перемещение в точку 3.

N60Z5 – быстрое перемещение в точку 4.

M05 – выключение вращения шпинделя, эмульсии.

M30 – сброс программы, возвращение на ее начало.

### **Задание 2**

Используя шаблон программы управления REZB для нарезания резьбы (рис. 4.8), записать программу обработки детали (рис. 4.9) с размерами согласно приведенной ниже табл. 4.6. Номер варианта – по указанию преподавателя. Написать словесные комментарии кадров программы.

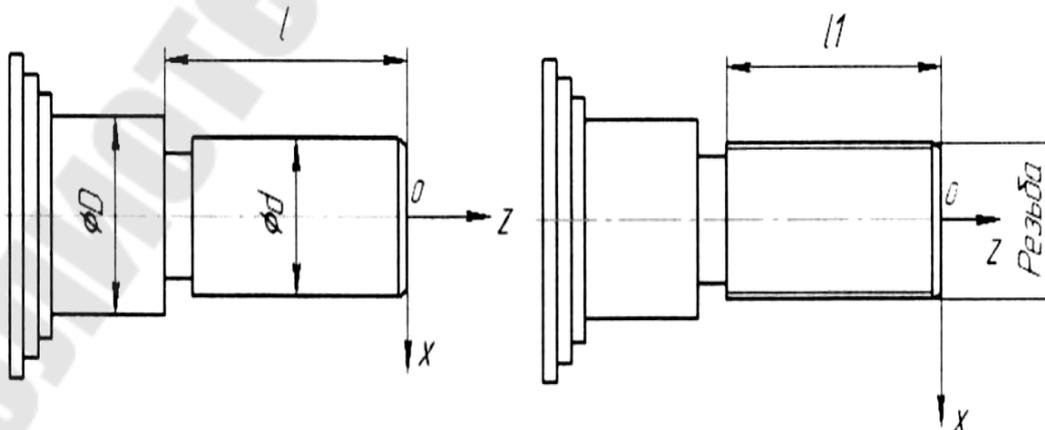


Рис. 4.9. Нарезание метрической резьбы

Исходные данные для нарезания резьбы

Вариант	$d_{нар}$ , мм	$d_{вн}$ , мм	Резьба с шагом К, мм	L, мм	L1, мм
1	14	11,835	M14x2	18	14
2	16	13,835	M16x2	19	15
3	18	15,294	M18x2,5	20	16
4	20	17,294	M20x2,5	21	17
5	22	19,294	M22x2,5	22	18
6	24	20,752	M24x3	23	19
7	30	26,211	M30x3,5	24	20
8	36	31,670	M36x4	38	36
9	42	37,129	M42x4,5	40	35
10	48	42,587	M48x5	44	38
11	52	46,587	M52x5	48	42
12	56	50,046	M56x5,5	56	50
13	60	54,046	M60x5,5	60	54
14	64	57,505	M64x6	66	60
15	68	61,505	M68x6	70	64

Набрать на ПК программу обработки в текстовом редакторе «Блокнот». Записать на флэш-память, распечатать и *предъявить преподавателю на проверку программы на эмуляторе.*

### **Задание 3**

Используя стандартный цикл **G1L** для нарезания метрической резьбы (см. п. 4 «Цикл нарезания резьбы с помощью файла G1L»), записать программу нарезания резьбы на детали с размерами согласно приведенной выше таблице. Номер варианта – по указанию преподавателя. Быть готовым дать словесные комментарии кадров программы.

Набрать на ПК программу обработки в текстовом редакторе «Блокнот». Записать на флэш-память, распечатать и предъявить преподавателю на проверку программы на эмуляторе.

## Лабораторная работа № 5

# ПРИОБРЕТЕНИЕ НАВЫКОВ НАПИСАНИЯ ПРОГРАММ ДЛЯ ТОКАРНОЙ ОБРАБОТКИ ДЕТАЛЕЙ С ИСПОЛЬЗОВАНИЕМ ПАРАМЕТРИЧЕСКОГО ПРОГРАММИРОВАНИЯ И ТРЕХБУКВЕННЫХ ОПЕРАТОРОВ

### Цель работы:

1. Изучить программирование с помощью кода **E** параметрических, геометрических и технологических данных цикла обработки. Ознакомиться с различными индексами параметров **E** для переменных различного формата УЧПУ N201M. Изучить рекомендации по использованию параметров **E**.

2. **Изучить** трехбуквенные операторы, модифицирующие систему отсчета осей.

3. Научиться составлять программы для токарной обработки с использованием параметрического программирования и трехбуквенных операторов, модифицирующие систему отсчета осей.

В качестве учебного пособия используются:

1. Методические указания по выполнению данной работы.

2. Руководство оператора (РО УЧПУ NC-201, NC-201M, NC-202, ООО «Балт-Систем», [www.bsystm.ru](http://www.bsystm.ru)). При выполнении работы в лаборатории университета РО входит в состав работы № 5.

3. Руководство программиста (РП УЧПУ NC-110, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230, ООО «Балт-Систем», [www.bsystm.ru](http://www.bsystm.ru)). При выполнении работы в лаборатории университета РП входит в состав работы № 5.

В конце занятия необходимо:

– предоставить для защиты написанную с комментариями управляющую программу (УП) «Сетка» по заданию преподавателя;

– ответить письменно на ряд нижеперечисленных вопросов для подтверждения усвоения материала и зачета работы:

1. Для обозначения числовых значений каких функций и кодов используются параметры E0–E9?

2. Для обозначения числовых значений каких функций и кодов используются параметры E10–E19?

3. Какая из функций меняет знак параметра E на противоположный?

4. Для обозначения числовых значений каких функций и кодов используются параметры E25–E29?

5. Для обозначения числовых значений каких функций и кодов используются параметры E30...(\*)?
6. Каким трехбуквенным кодом вызываются подпрограммы?
7. Какие трехбуквенные операторы модифицируют систему отсчета осей?
8. Какой из операторов выбирает абсолютную начальную точку, объявленную в кадре, изменяя ее временно на величину, равную запрограммированной?
9. Какой из операторов позволяет в приращениях перемещать текущую начальную точку для всех осей, заданных в ней?
10. Какой из операторов активирует зеркальную обработку относительно выбранных осей?
11. Какой из операторов вращает плоскость интерполяции на угол, значение которого задано операндом?
12. Какой из операторов осуществляет масштабирование?
13. Какой из операторов модифицирует начальную точку?
14. Каким числом ограничено количество параметров E?
15. Какие арифметические операции можно выполнять с помощью операторов E?
16. Какие действия допускаются производить с параметрами E?

После усвоения материала нижеприведенных методических указаний необходимо сдать тесты и разработать и защитить УП по заданию № 4 (рис. 5.1).

Для сдачи тестов отводится 10 мин.

Результаты тестов заносятся в ведомость оценки результатов сдачи тестов.

## **МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ № 5**

### **1. Параметрическое программирование**

Используя коды E, можно через параметры программировать геометрические и технологические данные цикла обработки. С параметрами допускаются математические и тригонометрические действия, а также вычисление выражений. Максимальное число параметров E не ограничено и определяется во время конфигурации системы. Параметры E имеют различные индексы для переменных различного формата. Описание параметров E для различных форматов представлено в табл. 5.1.

Параметры **E** получают значения в кадрах назначения.

Таблица 5.1

### **E-параметры и их форматы**

<b>Формат</b>	<b>Параметры</b>	<b>Мин/макс величина</b>
BY(байт)	E0...E9	От 0 до 255
IN(целое)	E10...E19	От -32768 до +32767
LI(целое с двойной точностью)	E20...E24	От -2.147.483.647 до +2.147.483.647
RE(действительное)	E25...E29	+(-)7 – целые или десятичные цифры
LR(действительное с двойной точностью)	E30...(*)	+(-)16 – целые и десятичные цифры, +(-)13 – целые числа

(\*) – максимальное число E-параметров, определенное во время конфигурации

**Формат** кадра назначения:

**E<sub>n</sub>** = <выражение> ,

где <выражение> – может быть цифровой величиной или математическим выражением, результат которого будет запомнен под параметром **E** индекса **n**.

«**Выражение**» – это математическое выражение, составленное из арифметических операторов, функций и операндов (параметры **E**, числовые константы).

Допустимы следующие арифметические операции:

- + сложение;
- вычитание;
- \* умножение;
- / деление.

Возможные функции:

**SIN(A)** – вычисляет синус A;

**COS(A)** – вычисляет косинус A;

**TAN(A)** – вычисляет тангенс A;

**ARS(A)** – вычисляет арксинус A;

**ARC(A)** – вычисляет арккосинус A;

**ART(A)** – вычисляет арктангенс A;

**SQR(A)** – вычисляет квадратный корень A;

**ABS(A)** – вычисляет абсолютное значение A;

**INT(A)** – вычисляет целое число A;

**NEG(A)** – инвертирует знак A;

**MOD(A,B)** – вычисляет остаток отношения между A и B;

**FEL(A,B)** – извлекает элемент с индексом **B(1,2,3)** для геометрического элемента «линия» (прямая), которая определена в GTL с индексом **A** ( $l_n = p_1, p_2$ , где  $n=A$ ); для индекса **B**: 1 – синус угла; 2 – косинус; 3 – расстояние от прямой линии до начальной точки;

**FEP(A,B)** – извлекает элемент с индексом **B(1,2)** для геометрического элемента «точка», которая определена в GTL с индексом **A** ( $p_n = X_{10}Y_{15}$ , где  $n=A$ ); для индекса **B**: 1 = абсцисса точки; 2 = ордината;

**FEC(A,B)** – извлекает элемент с индексом **B(1,2,3)** для геометрического элемента «окружность», которая определена в GTL с индексом **A** ( $c_n = X_{10}Y_{15}r_{24}$ , где  $n=A$ ); для индекса **B**: 1 – абсцисса центра; 2 – ордината; 3 – радиус окружности.

Значения для **(A)** и **(A, B)** могут быть параметрами E или числовыми константами. Выражение вычисляется с учетом приоритета скобок и знаков. Результат, если совместим, преобразуется в формат параметра E, указанный слева от знака «=».

*Примеры:*

$E_{30} = FEL(5,1)$  – придает E30 значение синуса угла, который образует прямая l5 с абсциссой.

$E_{34} = FEP(4,2)$  – придает E34 значение ординаты точки p4.

$E_{42} = FEC(8,3)$  – придает E42 значение радиуса окружности c8.

Значения вычисления параметров:

$N1E_{37} = (E_{31} * SIN(E_{30}) + 123.4567) / SQR(16)$  – выполняет математическое решение выражения и придает результат параметру E37.

"LAB1"E<sub>51</sub> = -0.00000124+5 – выполняет вычисление выражения и придает результат параметру E51.

$E_{40} = TAN(35)$  – извлекает тангенс 35 градусов и придает результат параметру E40.

$E_{35} = FEP(37,1)$  – извлекает абсциссу точки p37, ранее занесенной в память, и придает значение параметру E35.

$E_{31} = NEG(E_{31})$  – меняет знак параметра E31.

$E_7 = 81$  – придает значение параметру E7.

$E_{25} = E_{25} + 30$  – новым значением параметра E25 будет сумма константы 30 и текущего значения параметра E25.

$E_2 = SK396$  – придает E2 содержимое байта 396 пакета «K».

$E_8 = SYVAR1$  – придает E8 значение переменной SYVAR1.

Операнды тригонометрических функций должны быть выражены в градусах. Результат функций ARS, ARC, ART также выражается в градусах.

Параметры **E** могут быть использованы как внутри программы, так и внутри подпрограммы и могут быть воспроизведены.

*Пример:*

(DIS,E54) – воспроизводит на экране величину E54.

Параметры стираются при выключении УЧПУ и могут быть приведены к нулю, если это предусмотрено в фазе характеристики. Использование параметров E сведено в табл. 5.2.

Таблица 5.2

### Использование параметров E

Параметры (формат)	Данные (геометрические, технологические)	Пример программирования
E0...E9 (BY)	Функция G функция M коды RPT	GE1 ME3 (RPT, E9)
E10...E19 (IN)	Номер абсолютной начальной точки: функция T функция S	(UAO,E10) (UOT, E11,X...,Y..) TE14.E15 SE15
E20...E24 (LI)	—	—
E25...E29 (RE)	Функция F коды URT коды SCF индексные оси код UGF	FE27 (URT,E25) (SCF,E26) RE29 UGF=E28
E30...(*) (LR)	Координаты осей A B C X Y Z U координаты R составные операции IJK факторы корректировки u,v,w глобальные переменные системы: TMR UOV	XE32 RE33 KE34 VE35  TMR=E38 UOV=E40

## 2. Параметрические подпрограммы

Под подпрограммой понимается последовательность кадров, определяющая пользовательский цикл обработки, которая может быть вызвана из основной программы. Подпрограмма может вызывать только одну подпрограмму (2 уровня вложенности). Подпрограммы хранятся в памяти пользователя, их количество зависит только от их длины и от объема используемой памяти. Подпрограмма вызывается трехбуквенным кодом CLS.

Пример:

N35(CLS,PROG1)

### 2.1. Параметрические подпрограммы

Подпрограммы, т. е. последовательность кадров, которые определяют цикл обработки, вызываемые из основной программы, являются параметрическими в случае, если геометрические и технологические данные цикла обработки, т. е. величины, которые должны быть приданы различным функциям **G**, **S**, **X**, **Y**, **Z** и т. д., заданы через параметры, значения которых определены в основной программе.

Пример программирования круговой сетки с использованием подпрограммы приведен на рис. 5.1.

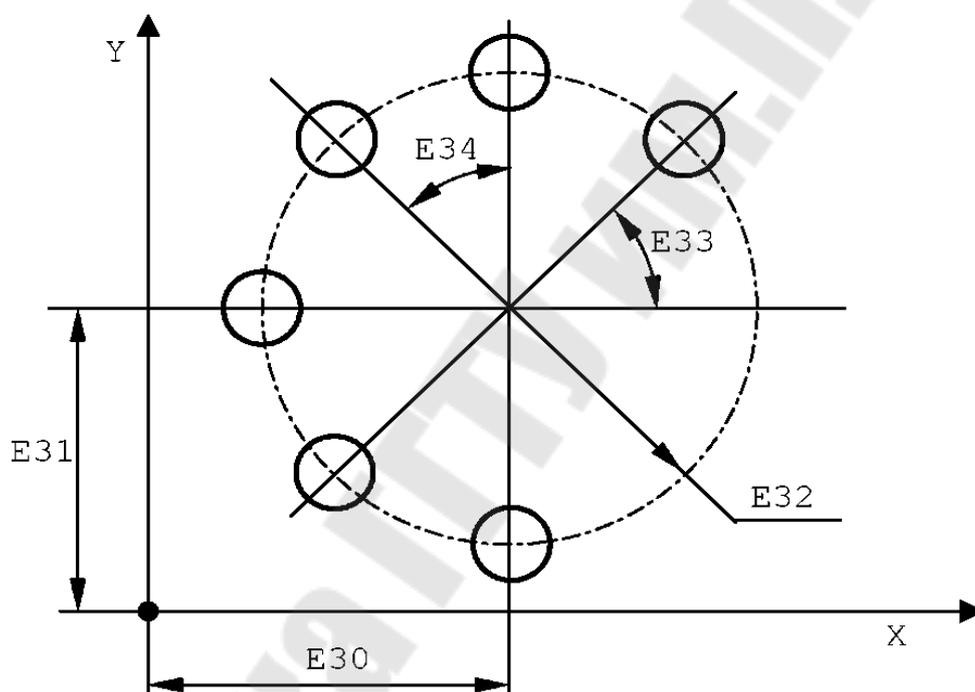


Рис. 5.1. Пример программирования круговой сетки отверстий

E1 – количество отверстий

E30 – X абсцисса центра окружности

E31 – Y ордината центра окружности

E32 – радиус окружности

E33 – начальный угол

E34 – угловой шаг

E36 – размер **R**

E37 – размер **Z**

Основная программа  
;SET1  
N10(DIS,"SETKA")  
N20G97S900T2.2 F70M6M3  
N30(UAO,1)  
N40E30=50  
N50E31=50  
N60E32=30  
N70E33=45  
N80E34=45  
N90E1=6  
N100E36=2  
N110E37=10  
N120(CLS,PP800)  
N130G0Z20M5M30

Подпрограмма  
PP800  
799(DIS," SETKA")  
N800(UOT,1,XE30,YE31)  
N801E25=E33  
N802(RPT,E1)  
N803(URT,E25)  
N804 G81RE36ZE37  
N805XE32Y0  
N806E25=E25+E34  
N807G80  
N808(ERP)  
N809(UAO,0)  
N810(URT,0)

### 3. Кадры с трехбуквенными операторами

Этот раздел описывает функциональность и синтаксис кадров, имеющих в качестве операторов трехбуквенные коды. Можно установить семь классов трехбуквенных операторов:

- операторы, изменяющие систему начала отсчета осей;
- операторы, изменяющие последовательность выполнения программы;
- смешанные операторы;
- операторы ввода/вывода;
- операторы контроля инструмента;
- операторы видеографического управления;
- операторы управления коррекциями.

#### **3.1. Трехбуквенные операторы, модифицирующие систему отсчета осей**

Операторы этого класса позволяют изменять декартовую систему отсчета, по отношению к которой был запрограммирован профиль. К этому классу принадлежат следующие операторы: **UAO**, **UOT**, **UIO**, **MIR**, **URT**, **SCF**, **RQO**.

##### *3.1.1. Использование абсолютных начальных точек – UAO*

Оператор **UAO** выбирает одну из абсолютных начальных точек, ранее определенных командой **ORA**.

**Формат:****(UAO,n[,VAR-1,VAR-2...VAR-n]),**

где **n** – определяет номер начальной точки, которую надо выбрать.

Может быть цифровой постоянной или параметром E типа целый (от **E10–E19**); **VAR-1** – символ, представляющий название оси, для которой определяется начальная точка «**n**»; для необъявленных осей остается в силе текущая начальная точка. Если «название оси» не присутствует, начальная точка «**n**» приводится в действие для всех осей, для которых была объявлена эта начальная точка.

*Пример:***(UAO,1)** – абсолютная начальная точка 1 активна для всех осей;**(UAO,2,X,Y)** – абсолютная начальная точка 2, активизируется для осей X и Y;**(UAO,3,B)** – абсолютная начальная точка 3; активизируется только для оси B;**(UAO,0)** – активизирование нулевой абсолютной начальной точки станка для всех осей.

При включении УЧПУ и после команды «СБРОС» автоматически активизируется нулевая начальная точка для всех осей. Максимально могут присутствовать **шесть** «названий осей». Не могут быть определены одинаковые «названия осей». Если требуется привести в действие различные начальные точки для различных осей, необходимо программировать столько кадров с этими операторами, сколько имеется начальных точек. Если выбранная начальная точка (**n**) загружена в файл альтернативной системы измерения, она автоматически переводится в текущую систему измерения.

### 3.1.2. *Определение и использование временных начальных точек UOT*

Оператор UOT выбирает абсолютную начальную точку, объявленную в кадре, изменяя ее временно на величину, равную запрограммированной.

**Формат:****(UOT,n,VAR-1[,VAR-2...VAR-n]),**

где **n** – имеет то же значение, что и для оператора UAO; **VAR-1** – операнд типа «ось–размер»; значение, приданное ему, рассматривается как корректировка, к которой надо прибавить значение, содержащееся в абсолютной начальной точке для той же оси. Для необъявленных осей остается в силе текущая начальная точка.

(UАО,0) – активизируется абсолютная начальная точка 0. Программа, отнесенная к абсолютной начальной точке 0 для всех осей.

1) (UОТ,0,X100,Y100) – применяется временная начальная точка к абсолютной начальной точке 0 с корректировками X100 и Y100 (временная начальная точка).

Пример приведен на рис. 5.2.

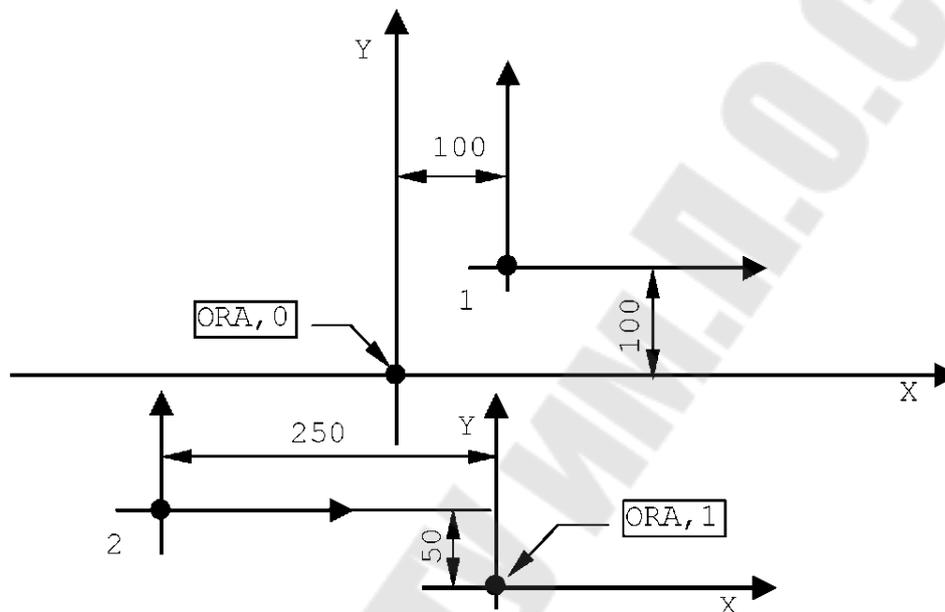


Рис. 5.2. Использование временных начальных точек UOT

2) (UОТ,1,X-250,Y-50) – применяется временная начальная точка к абсолютной начальной точке 1 с корректировками X-250 и Y-50.

(UАО,0) – активизируется абсолютная начальная точка 0 для всех осей.

По крайней мере, должен присутствовать один операнд оси. Максимально могут присутствовать шесть осей. Не могут быть определены операнды осей с одним и тем же названием. Временная начальная точка остается активной до того, как определяется новая временная начальная точка, или до вызова абсолютной начальной точки, или до команды «СБРОС». Размер в операторе UОТ необходимо программировать в текущей размерности мм/дюйм (G70/G71).

### 3.1.3. Определение и использование начальных точек по приращениям UЮ

Эта команда позволяет приращением переместить текущую начальную точку для всех осей, заданных в ней (рис. 5.3).

N65(UIO,X20,Y20) – точка 1  
 N121(UIO,Y-40) – точка 2  
 N180(UIO,X-45) – точка 3  
 N230(UIO,Y-35) – точка 4  
 N300(UIO,0) – точка ORA,0

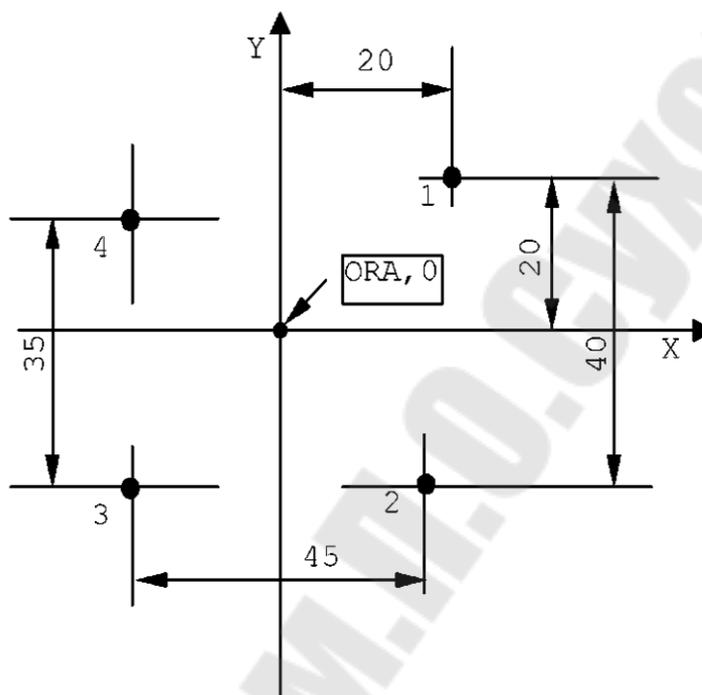


Рис. 5.3. Определение и использование начальных точек по приращениям UIO

**Формат:**

**(UIO,VAR-1[,VAR-2,...VAR-n]),**

где **VAR-n** – представляет ось и размер. Система берет размер как абсолютное смещение и прибавляет его к абсолютной начальной точке для данной оси. Для необъявленных осей текущая начальная точка остается в силе. Количество значений VAR-n должно быть не больше шести (одно значение на одну ось).

Начальная точка, заданная по приращениям, остается в силе до ее переопределения с новой командой UIO, или **восстанавливается абсолютная начальная точка при помощи (UAO,0) или операции [«СБРОС»]**.

Значение приращений в команде UIO необходимо программировать в текущей размерности мм/дюйм (G70, G71).

**3.1.4. Использование подпрограммы CLS**

Оператор CLS позволяет вызвать и выполнить программу (подпрограмму), находящуюся в памяти. Под подпрограммой понимаем последовательность кадров, которые определяют цикл обработки. Подпрограмма может быть вызвана из основной программы.

**Формат:**

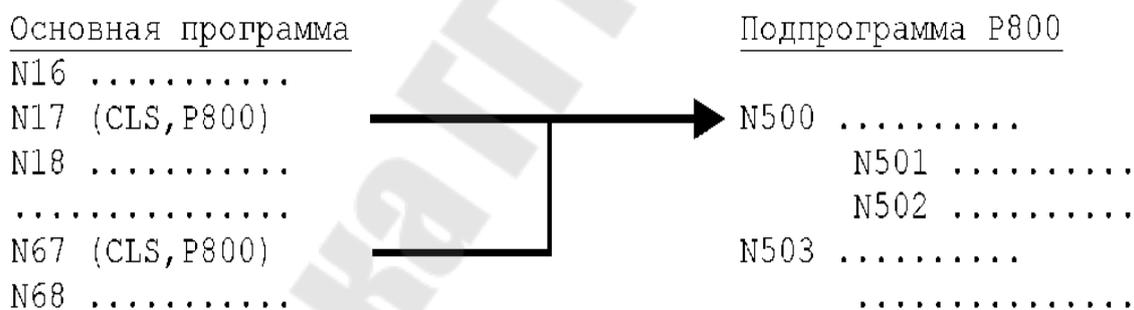
**(CLS,ФАЙЛ[ДИРЕКТОРИЙ]),**

где **ФАЙЛ** – название программы для вызова; **ДИРЕКТОРИЙ** – название устройства памяти, которое содержит программу. Если определение устройства памяти в команде отсутствует, применяется то, которое объявлено для данного процесса по умолчанию в секции 4 файла PGCFIL инструкции NDD.

**Синтаксические обязательства** – название программы состоит из алфавитно-цифровой последовательности и может иметь максимум шесть символов. Название программы отделяется от названия устройства символом «/». Устройство заменяется последовательностью двух или трех алфавитно-цифровых символов. Алфавитные символы, которые используются для параметров **ФАЙЛ** и **УСТРОЙСТВО**, могут быть только заглавными; первым символом должна быть буква.

*Пример:*

N1(CLS,P800/MP2) – передает управление подпрограмме P800, расположенной в директории MP2 (если MP2 характеризуется по умолчанию, название не записывается):



Допустимы два уровня вызова. Программа, вызванная с CLS, может, в свою очередь, вызывать другие программы, в то время как эти программы уже не могут вызывать другие программы.

Подпрограммы могут быть параметрическими. Цифровые значения параметров определяются в основной программе в момент вызова. Пример использования оператора UOT и подпрограмм для сверления группы отверстий приведен на рис. 5.4.

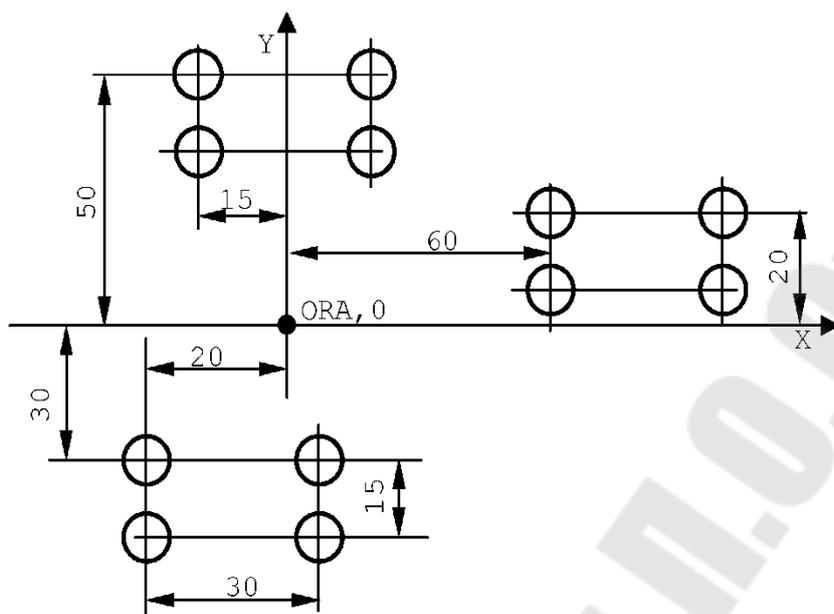


Рис. 5.4. Использование подпрограмм

Основная программа

```

;OTV4
N20G95G97S2000F180T2.2M6M3
N21(UOT,0,X-20,Y-30)
N2(CLS,S600)
N23(UOT,0,X-15,Y-50)
N24(CLS,PP600)
N25(UOT,0,X60,Y20)
N26(CLS,PP600)
N27Z30
N28M05
N29M30

```

Подпрограмма

```

;PP600
N501 G81 R3Z-12
N502 X0Y0
N503 Y-15
N504 X30
N505Y0
N506G80

```

### 3.1.5. Зеркальная обработка MIR

Оператор MIR инвертирует запрограммированные направления перемещений, объявленных в операторе. Для необъявленных осей предыдущая функция MIR остается в силе. Если не запрограммирован никакой операнд, функция MIR выводится из действия для всех конфигурируемых осей.

**Формат:**

(MIR[,VAR-1,...,VAR-n]),

где VAR-n – должен быть буквой, соответствующей одному из возможных названий конфигурируемых осей системы.

Пример приведен на рис. 5.5:

.....  
N24(MIR,X)

.....  
N42(MIR,X,Y)

.....  
N84(MIR,X)

.....  
N99(MIR)

Зеркальная обработка активизируется для запрограммированной оси, начиная с первого движения данной оси после команды MIR. Зеркальное отображение осуществляется относительно текущей начальной точки. Максимально может быть запрограммировано шесть осей. Нельзя программировать 2 раза одну и ту же ось. Если присутствуют команды вращения (URT) и зеркальной обработки (MIR), они устанавливаются в следующем порядке: MIR и URT.

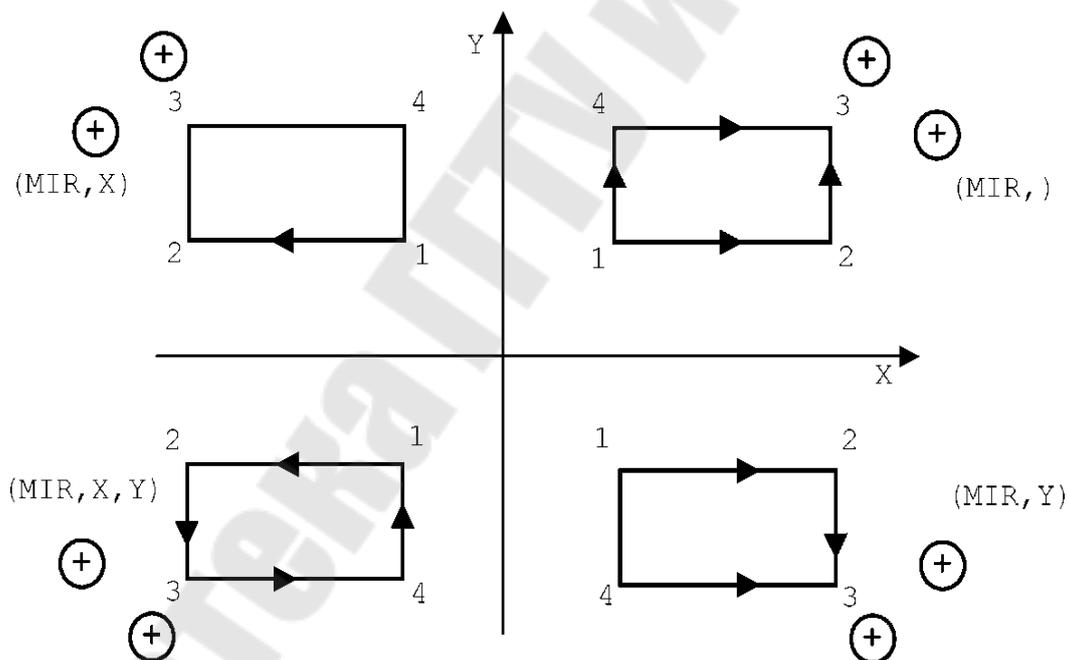


Рис. 5.5. Зеркальная обработка MIR

### 3.1.6. Поворот плоскости URT

Оператор URT вращает плоскость интерполяции на угол, значение которого дано операндом. Центром вращения является текущая начальная точка.

**Формат:**  
**(URT,ЗНАЧЕНИЕ),**

где **ЗНАЧЕНИЕ** – представляет величину угла, выраженную в градусах и десятых градуса; может быть выражен явно или неявно (параметр **E** типа от E25 до E29). Если операндом является «0», функция отменяется.

Операнд должен присутствовать обязательно. После кадра с URT вращение применяется к запрограммированным координатам. Координаты, относящиеся к нулю станка (G79), не вращаются. Если присутствуют команды вращения (URT) и зеркальной обработки (MIR), они устанавливаются в следующем порядке: MIR и URT.

Угол поворота, заданный оператором URT, автоматически обнуляется при выполнении команды SPG или операции [«СБРОС»].

Пример поворота плоскости изображен на рис. 5.6.

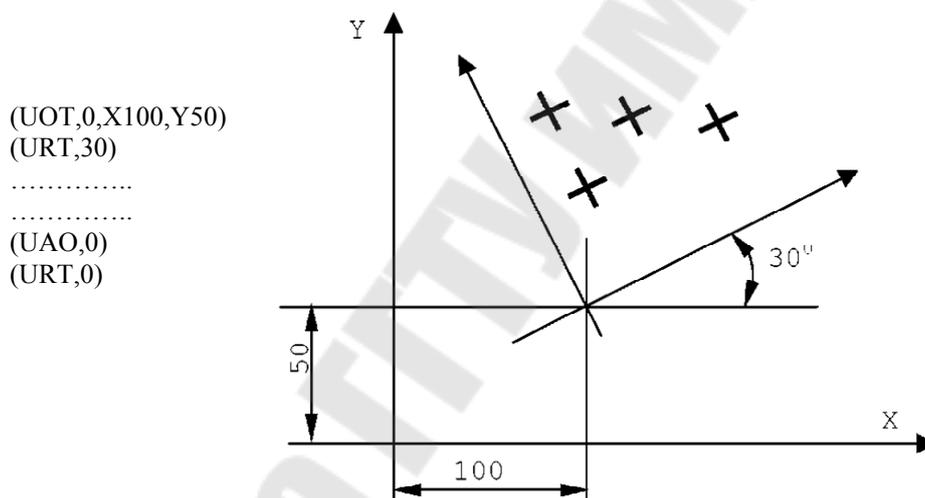


Рис. 5.6. Поворот плоскости с помощью кода URT

### 3.1.7. Масштабирование SCF

Масштабирование применяется для объявленных в операторе SCF осей.

**Формат:**  
**(SCF[n[,VAR-1,...,VAR-m]]),**

где **N** – определяет коэффициент масштабирования, который должен быть применен; может быть запрограммирован как явно, так и неявно при помощи параметра **E** (от E25 до E29); **VAR-1** – символ, представляющий одну из осей, для которой приведен в действие коэффициент масштабирования; для необъявленных осей масштабирование отменяется. Если к SCF не присоединен какой-либо операнд, масштабирование отменяется для всех осей.

Может быть запрограммировано максимально шесть названий осей.

*Пример:*

.....  
(SCF,3) – применяет коэффициент 3 ко всем конфигурируемым осям

.....  
(SCF,2,X) – применяет коэффициент 2 для оси X и отменяет коэффициент 3 для других осей

.....  
(SCF) – отменяется коэффициент масштабирования для всех осей

### 3.1.8. Модификация начальной точки RQO

Оператор RQO изменяет начальную точку для осей, объявленных в операторе, на запрограммированную величину.

**Формат:**

**(RQO,n,VAR-1[,VAR-2,...,VAR-N]),**

где **n** – определяет номер модифицируемой начальной точки, его величина заключена между 0 и 99 и тесно связана с числом записей, определенных во время создания файла начальных точек; **n** – может быть выражено явно или неявно при помощи параметра E типа целый (от E10 до E19); **VAR-1** – операнд типа «ось-размер», приданное ему значение является коррекцией запрограммированной начальной точки данной оси.

*Пример:*

(RQO,3,XE31) – изменяет начальную точку 3 оси X на величину, объявленную в E31.

Должен присутствовать, по крайней мере, один операнд (ось). Может быть запрограммировано максимально шесть операндов (осей). Не может быть определено больше одного операнда с одним и тем же названием оси. Начальная точка изменяется как в файле начальных точек (следовательно, результат модификации постоянный), так и в памяти пользователя, если эта начальная точка активна для оси в момент модификации.

В файле начальных точек модификации выполняются в той системе измерения, в которой выражена выбранная начальная точка. Однако необходимо определить программируемую величину модификации в текущих единицах измерения (G70/G71), и к ним не применяется масштабирование.

#### 4. Задание

Написать параметрические программу и подпрограмму «SETn» сверлильной обработки отверстий согласно варианту, выданного преподавателем (табл. 5.3) в редакторе «Блокнот» и скопировать на флэш-память для возможности проверки на эмуляторе УЧПУ. Быть готовым дать комментарии по каждому кадру программы.

*Примечание:* n – номер варианта.

Таблица 5.3

Исходные данные к программе «Сетка»

Номер варианта	E1	E30	E31	E32	E33	E34	E36	E37
1	6	50	45	60	60	50	2	-10
2	6	55	50	65	40	45	3	-20
3	6	60	55	70	30	50	4	-15
4	6	65	60	75	40	40	2	-14
5	6	70	65	80	30	45	5	-16
6	6	75	70	85	60	40	4	-20
7	6	80	75	90	50	40	3	-15
8	6	85	80	95	30	50	4	-20
9	6	90	85	100	45	45	2	-14
10	6	95	90	105	40	60	3	-15
11	6	100	95	110	30	50	2	-17
12	6	105	100	115	50	40	4	-20
13	6	110	105	120	30	45	3	-15
14	6	115	110	125	60	40	2	-20
15	6	120	115	130	45	50	4	-15

## Лабораторная работа № 6

### ПРИОБРЕТЕНИЕ НАВЫКОВ НАПИСАНИЯ ПРОГРАММ ДЛЯ СВЕРЛИЛЬНО-ФРЕЗЕРНОЙ ОБРАБОТКИ С ИСПОЛЬЗОВАНИЕМ ПОСТОЯННЫХ ЦИКЛОВ

#### Цель работы:

1. Изучить формат управляющих команд при составлении программ с использованием постоянных циклов УЧПУ N201M.

2. Научиться составлять программы для сверлильно-фрезерной обработки с использованием постоянных циклов УЧПУ N201M.

В качестве учебного пособия используются:

1. Методические указания по выполнению данной работы.

2. Руководство оператора (РО УЧПУ NC-201, NC-201M, NC-202, ООО «Балт-Систем», [www.bsystem.ru](http://www.bsystem.ru)). При выполнении работы в лаборатории университета РО входит в состав работы № 6.

3. Руководство программиста (РП УЧПУ NC-110, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230, ООО «Балт-Систем», [www.bsystem.ru](http://www.bsystem.ru)). При выполнении работы в лаборатории университета РЭ входит в состав работы № 6.

В конце занятия необходимо:

– предоставить для защиты написанную с комментариями управляющую программу «Плита» по заданию преподавателя (см. п. 1.6 методических указаний);

– ответить письменно на ряд нижеперечисленных вопросов для подтверждения усвоения материала и зачета работы:

1. Какие операции позволяют программировать функции постоянных циклов?

2. Каков формат программирования постоянных циклов?

3. К какой оси относятся координаты [R1[R2]]?

4. Как программируются координаты R 1 и R2?

5. Какая из функций используется в качестве постоянного цикла растачивания?

6. Какая из функций используется в качестве постоянного цикла нарезания резьбы метчиком?

7. Какая из функций используется в качестве постоянного цикла сверления?

8. Какая из функций используется в качестве постоянного цикла глубокого сверления?

9. Какая из функций используется в качестве постоянного цикла развертывания?

10. Какая из функций используется в качестве постоянного цикла развертывания с растачиванием?

11. Какая из функций используется в качестве постоянного цикла рассверливания?

12. В каком кадре программируется координата цикла?

13. Что необходимо сделать, чтобы выполнить последующие циклы, идентичные первому?

14. Какие подготовительные функции определяют постоянные циклы?

Для сдачи тестов отводится 10 мин.

Результаты тестов заносятся в ведомость оценки результатов сдачи тестов.

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ № 6

### 1. Программирование с использованием постоянных циклов

#### 1.1. Постоянные циклы (G80-G89)

Функции постоянных циклов G81–G89 позволяют программировать ряд операций, таких как сверление, нарезание резьбы метчиком, растачивание и т. д., без повторения для каждой из них размеров отверстия. Характеристики постоянных циклов приведены в табл. 6.1.

**Формат** кадра постоянного цикла:

**G8..[ДРУГИЕ G][R1[R2]]КООРДИНАТА ЦИКЛА [ДОПОЛНИТЕЛЬНЫЕ ОПЕРАНДЫ] [СКОРОСТЬ ПОДАЧИ] [ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ],**

где [ДРУГИЕ G] – это подготовительные функции, которые разрешаются программировать в кадре постоянного цикла; [R1[R2]] – это координаты, определенные в явном или неявном виде (параметр E), относящиеся к оси шпинделя; они определяют координату точки быстрого позиционирования начала обработки и координату точки возврата в конце обработки; если R2 отсутствует, то R1 считается конечной координатой при возврате в точку начала обработки; **КООРДИНАТА ЦИКЛА** – определяет координату глубины отверстия, значение которой выражено в явном или неявном виде (параметр E), и ось, вдоль которой выполняется цикл; **[СКОРОСТЬ ПОДАЧИ]** – определяется символом «F»; выражает скорость подачи, с которой выполняется обработка отверстия; если отсутствует, то скоростью подачи

будет последняя запрограммированная «F»; **[ДОПОЛНИТЕЛЬНЫЕ ОПЕРАНДЫ]** – являются операндами, определяющими параметры частных операций (например, I, J, K для глубокого сверления); **[ВСПОМОГАТЕЛЬНЫЕ ФУНКЦИИ]** – определяют функции S, M, T, H.

Последовательность движений при постоянных циклах можно представить следующим образом:

- быстрое позиционирование к оси отверстия;
- быстрый подход к плоскости обработки (размер R1);
- перемещение со скоростью рабочей подачи до запрограммированного размера (Z);
- функции цикла на дне отверстия;
- возвращение на быстром ходу или со скоростью рабочей подачи к координате R1 или R2, если координата возврата отличается от координаты подхода R1.

Если R2 отлично от R1, то необходимо программировать R1 и R2 в одном и том же кадре.

Фаза ускоренного возврата производится как движение с рабочей скоростью (G01), так и с быстрым ускорением. Характеристика постоянных циклов приведена в табл. 6.1.

Таблица 6.1

**Характеристики постоянных циклов**

Постоянный цикл	Подход	Функция на дне отверстия		Возврат
		выдержка времени	вращение шпинделя	
<b>G81</b> – сверление	Рабочая подача	нет	нормальное	Ускоренное перемещение к R1 или R2, если программируется
<b>G82</b> – растачивание	Рабочая подача	да	нормальное	Ускоренное перемещение к R1 или R2, если программируется
<b>G83</b> – глубокое сверление (с дроблением и разгрузкой стружки)	С прерывистой подачей (подход с рабочей скоростью с промежутком во время быстрого возврата или остановки)	да/нет	нормальное	Ускоренное перемещение к R1 ускоренное перемещение к R2, если программируется

Постоянный цикл	Подход	Функция на дне отверстия		Возврат
		выдержка времени	вращение шпинделя	
<b>G84</b> – нарезание резьбы метчиком	Рабочая подача; начало вращения	нет	реверсное	Рабочая подача к R1, ускоренное перемещение к R2, если программируется
<b>G85</b> – рассверливание	Рабочая подача	нет	нормальное	Рабочая подача к R1, ускоренное перемещение к R2, если программируется
<b>G86</b> – развертывание	Рабочая подача; начало вращения шпинделя	нет	остановка в конце цикла	Ускоренное перемещение к R1, ускоренное перемещение к R2, если программируется
<b>G89</b> – развертывание с растачиванием	Рабочая подача	да/нет	нормальное	Рабочая подача к R1, ускоренное перемещение к R2, если программируется
<b>G80</b> – отмена постоянных циклов	–	–	–	–

*Примечания* общего характера, имеющие отношение ко всем постоянным циклам:

1. В кадре, содержащем функцию G постоянного цикла, не программируется никакое дополнительное движение осей, кроме самого цикла; цикл не приводится в действие, а кадр заносится в память системы. Цикл стартует координатами, запрограммированными сразу после кадра, содержащего постоянный цикл (после выполнения первого цикла для того, чтобы выполнить последующие циклы, идентичные первому, достаточно запрограммировать координаты точек отверстия).

2. Продолжительность выдержки времени программируется трехбуквенным кодом TMR.

3. Не представляется возможным программировать постоянные циклы G8., если профиль запрограммирован на языке GTL и/или внутри кадров с G41/G42-G40.

4. Функции GTL являются модалными. Невозможно программировать новый постоянный цикл без закрытия предыдущего постоянного цикла с G80.

Пример: Постоянный цикл с  $R2=R1$  и  $R2$  не равно  $R1$  приведен на рис. 6.1.

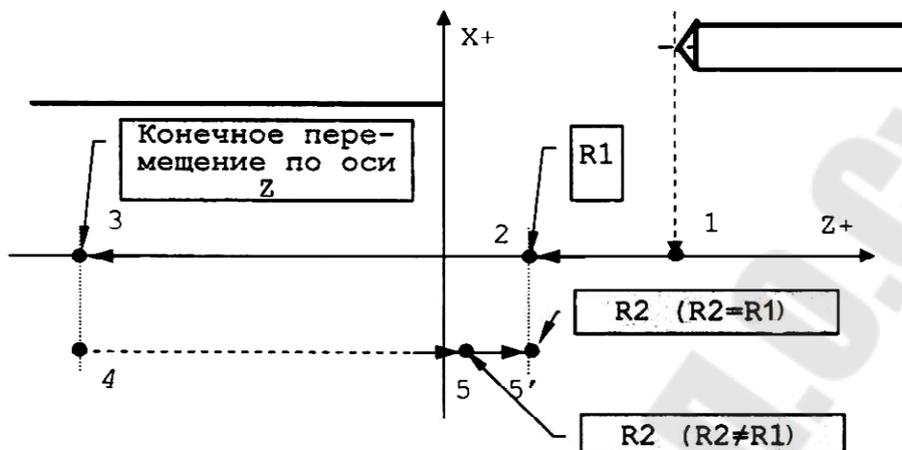


Рис. 6.1. Программирование постоянных циклов

### 1.2. Постоянный цикл сверления (G81)

Кадр программирования: **G81[R1..[R2..]]Z..**

Постоянный цикл G81 может быть также использован для операций растачивания, развертывания и центровочного сверления. Программирование постоянных циклов G82, G85, G86, G89 идентично программированию G81. В кадры, предшествующие постоянным циклам G82 и G89, вводится при необходимости выдержка времени через команду TMR.

Пример:

N33TMR=2

N34 G82 R3 Z-15 T6.6M6M13

N35X35Y150

N36 G80

В данном примере выдержка времени равна 2 с.

Постоянный цикл сверления G81 приведен на рис. 6.2.

N32 G95 G97S1000 F95 T3.3 M3M6M8

N33G81 R3 Z-15 – задание параметров постоянного цикла сверления

N34X15Y15 – движение к точке 1 и выполнение цикла

N35Y60 – движение к точке 2 и выполнение цикла

N36X80 – движение к точке 3 и выполнение цикла

N37Y15 – движение к точке 4 и выполнение цикла

N38G80Z50M5 – отмена действия цикла

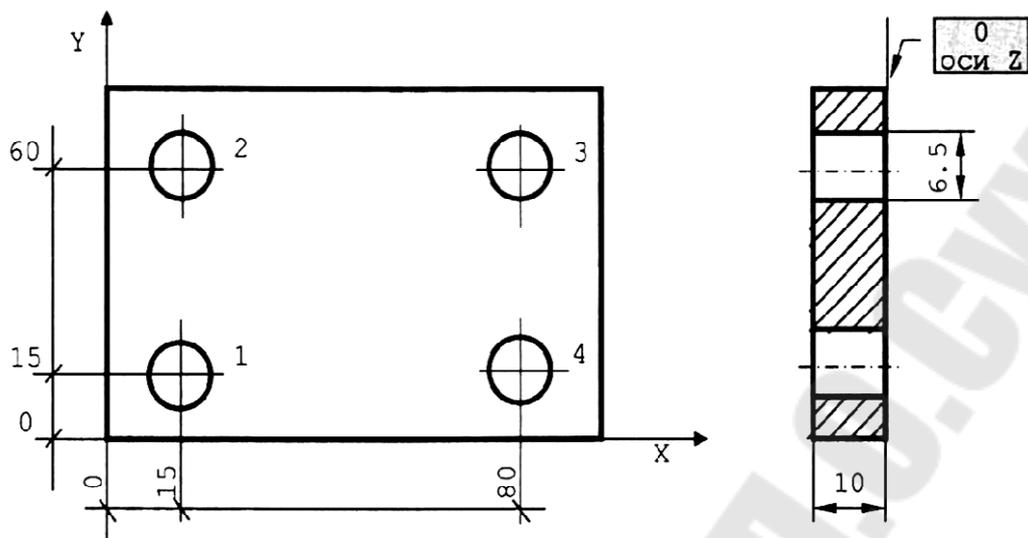


Рис. 6.2. Постоянный цикл сверления

Пример программирования постоянного цикла сверления G81 приведен на рис. 6.3.

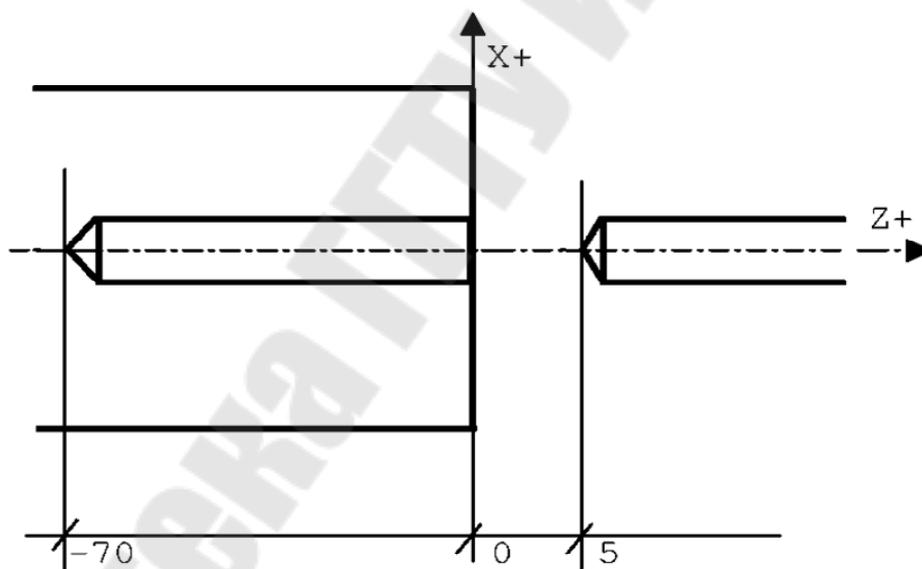


Рис. 6.3. Пример программирования постоянного цикла G81

```

N31(DIS, "TWIST DRILL D=6.5)
N32G97S1000T4.4M06M3M7
N33G81R5Z70F45
N34X0
N35G80
N36G..X..Z..

```

Пример программирования постоянного цикла с двумя параметрами R приведен на рис. 6.4.

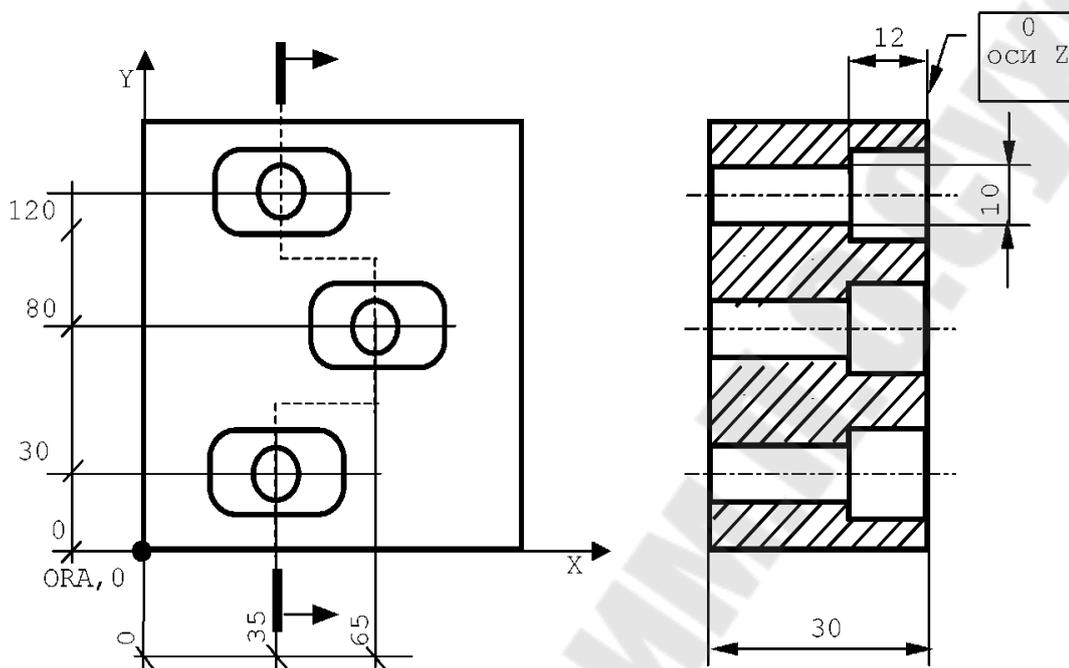


Рис. 6.4. Пример программирования постоянного цикла с двумя параметрами R

```
N43 G97G95S850 F100 T4.4 M6
N44 G81 R -10 R 2 Z36 M3
N45 X35 Y30
N46 X65 Y80
N47 X35 Y120
N48 G80 Z50 M5
```

### 1.3. Постоянный цикл глубокого сверления (G83)

Формат кадра:

**G83**[R1..[R2..]]Z..I..[K..][J..],

где [R1] – начальная координата отверстия (как для G81); [R2] – координата точки возврата (как для G81); Z – координата дна отверстия (как для G81); I – приращение размера Z после каждого цикла разгрузки стружки; J – минимальное приращение цикла разгрузки стружки; после достижения запрограммированного значения следуют постоянные приращения; K – коэффициент уменьшения параметра I (до достижения величины J).

Присутствие или отсутствие этих параметров определяет два разных цикла:

1) в случае, при котором были запрограммированы **I**, **K**, **J**, цикл имеет следующие шаги:

- быстрый подход к оси отверстия для обработки;
- быстрый подход к точке R1;
- подход с рабочей подачей к точке R1+I;
- быстрый возврат к точке R1 (разгрузка стружки);
- вычисление нового значения R1:R1=R1+I-1;
- вычисление нового значения I:

$I=J*K$ , если  $I*K \geq J$ .

$I=J$ , если  $I*K < J$ .

Шаги, начиная со второго, выполняются один за другим до получения запрограммированного размера глубины сверления (рис. 6.5).

*Примечание.* Для сохранения параметра **I** неизменным (постоянное приращение) запрограммировать **K=1** в отсутствии параметра **J**.

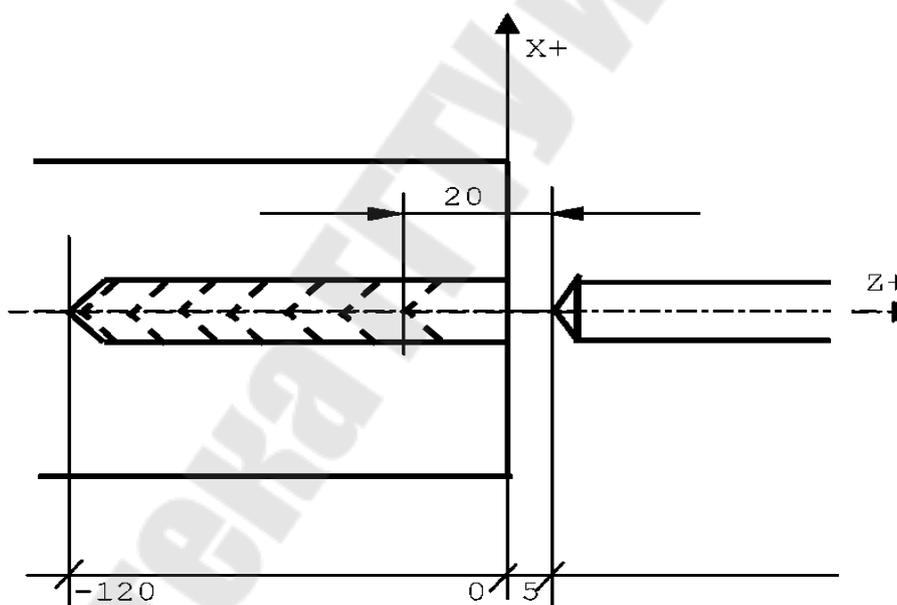


Рис. 6.5. Постоянный цикл глубокого сверления G83 с разгрузкой стружки

N61(DIS, "TWIST DRILL D=6)

N62G97S1000T3.3M6M3M7

N63G83R5Z120I20K0.8J6

N64X0Y0

N65G80

N66G..X..Z..

2) случай, при котором не были запрограммированы **K** и **J** (дробление стружки без разгрузки) – подача с постоянным приращением и выдержка времени при любом приращении, обеспечивается следующими шагами:

- быстрый подход к центру отверстия для обработки;
- быстрый подход к размеру R1;
- рабочая подача к точке R1+I;
- выдержка времени, запрограммированная с TMR;
- подход по другой величине I;
- три последних шага следуют один за другим до достижения запрограммированного размера глубины;
- быстрый выход из отверстия к точкам R1 или R2, если R2 запрограммирован.

*Пример глубокого сверления* приведен на рис. 6.6.

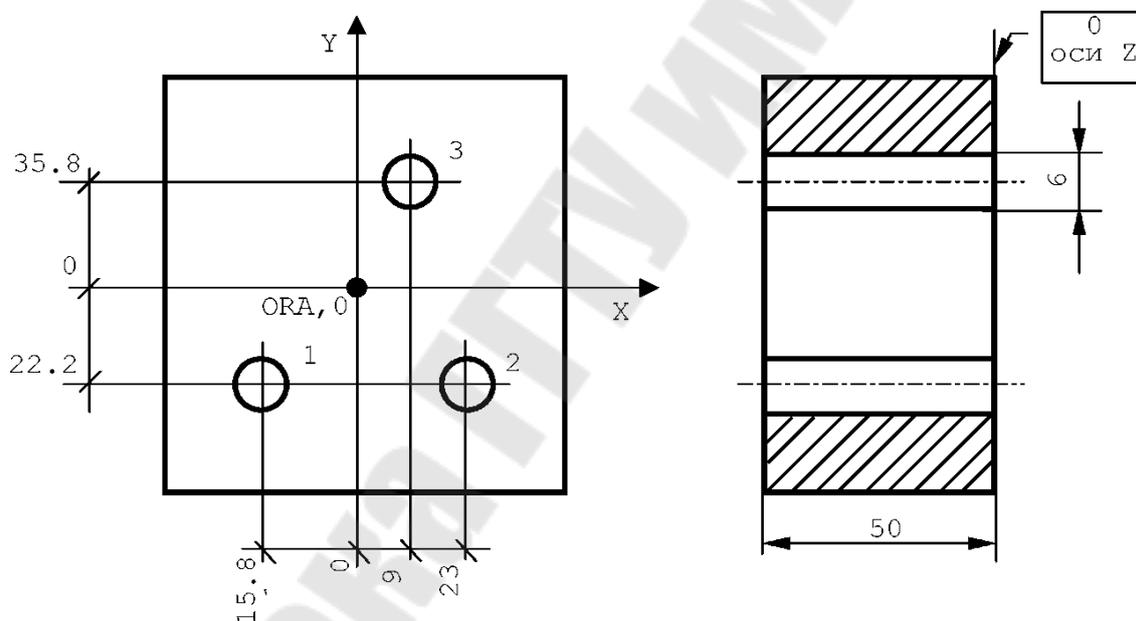


Рис. 6.6. Постоянный цикл глубокого сверления G83 без разгрузки стружки с ее дроблением

N66 G95G97S930 F65 T6.6 M6

N67 G83 R3 Z-55 I20 K0.8 J6 M13 – задание цикла

N68 X-15.8 Y-22.2 – сверление в точке 1

N69 X23 – сверление в точке 2

N70 X9 Y35.8 – сверление в точке 3

N71 G80 Z50 M – подъем инструмента и отмена цикла.

#### 1.4. Постоянный цикл нарезания резьбы метчиком (G84)

Постоянный цикл нарезания резьбы метчиком (G84) может быть выполнен двумя способами.

- Шпиндель без датчика

Формат кадра цикла G84:

**G84[R1..][R2..]Z..,**

где **G84** – код цикла нарезания резьбы метчиком; **[R1]** – координата точки начала обработки и конца обработки, если  $R1=R2$  (размер быстрого подхода и возврата на рабочей скорости); **[R2]** – координата точки конца обработки, если  $R1 \neq R2$ ; **Z** – конечная координата нарезания резьбы.

При программировании необходимо учитывать следующее:

- размер перемещения быстрого хода инструмента к детали в операциях нарезания резьбы метчиком должен всегда заканчиваться на расстоянии от детали, равном пяти шагам резьбы, если глубина  $\leq 3$  диаметров, или семи шагам, если глубина  $>3$  диаметров;

- скорость подачи  $F$ , которую следует запрограммировать, вычисляется следующим образом:

$$F = S * p * 0.9,$$

где  $S$  – скорость вращения шпинделя;  $P$  – шаг резьбы;  $0,9$  – коэффициент уменьшения скорости для сохранения упругости пружинного компенсатора резцедержателя.

При этом посредством кода RMS, задаваемого в программе или введенного с клавиатуры, можно изменять скорость возврата инструмента, определяя это изменение в процентах. Окончательный размер  $Z$  должен быть уменьшен на величину, равную 10 % от фактической глубины глухого отверстия под резьбу.

*Пример* приведен на рис. 6.7.

В рассматриваемом примере с помощью функции G84 в кадре N92 объявляется цикл нарезания резьбы метчиком, который будет выполняться в точках 1, 2, 3 (рис. 6.7), заданных следующими кадрами УП:

N91 S280 T8.8 M6 M3

N92 G84 R7 Z-15

N93 X-51.96 Y-30 – точка 1

N94 X51.96 – точка 2

N95 X0 Y60 – точка 3

N96 G80 Z50 M5

Кадр N92 обеспечивает быстрый подвод инструмента в точку ( $R1 = 7$  мм), нарезание резьбы метчиком на глубину, определяемую координатой «Z» (Z-15), и возврат к точке R1 на рабочей скорости. Цикл отменяется функцией G80 в кадре N96.

Эта программа используется для нарезания правосторонней резьбы метчиком (вращение вправо), что обеспечивается функцией M13, запрограммированной в кадре N91. Если необходимо запрограммировать нарезание резьбы влево, достаточно запрограммировать функцию M14 (M04) вместо M13 (M03).

Если рабочий путь недостаточен для выполнения разгона/торможения, подается сигнал ошибки.

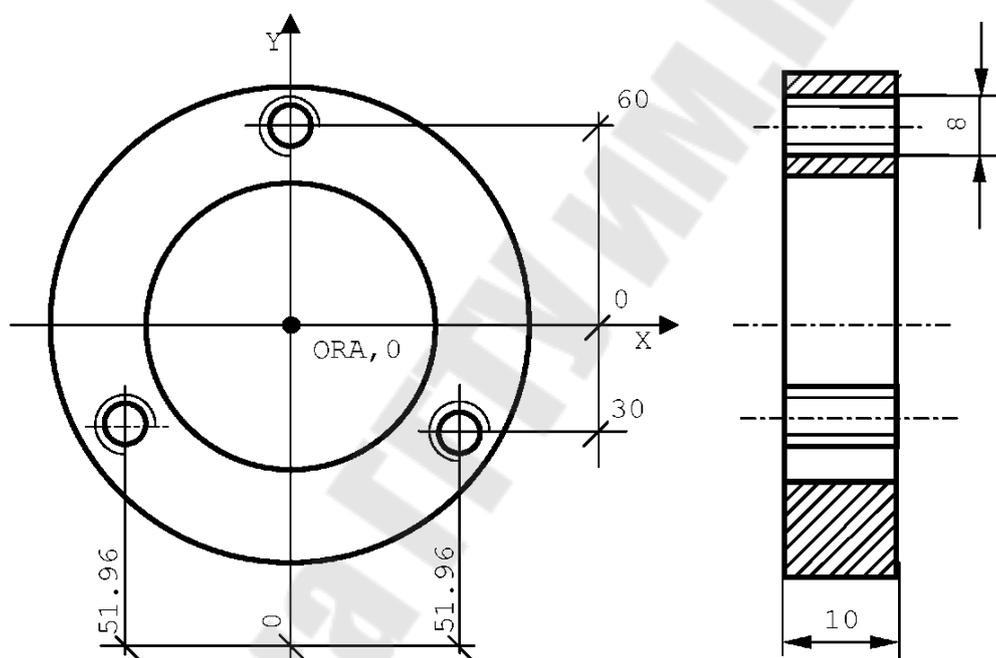


Рис. 6.7. Нарезание резьбы метчиком

#### • Шпиндель с датчиком

В данном случае существует два способа программирования функции G84:

- использование программирования скорости подачи  $F$ , как в случае для шпинделя без датчика;
- использование программирования шага резьбы  $K$ . В этом случае система автоматически вычисляет подачу, умножая шаг  $K$  на число оборотов шпинделя.

**Формат кадра цикла G84:**

**G84[R1..][R2..]Z..K..,**

где **G84** – код цикла нарезания резьбы метчиком; **[R1]** – координата точки начала обработки и конца обработки, если  $R1 = R2$  (размер быстрого подхода и возврата на рабочей скорости); **[R2]** – координата точки конца обработки, если  $R1 \neq R2$ ; **Z** – конечная точка нарезания резьбы метчиком; **[K]** – шаг резьбы.

*Пример* приведен на рис. 6.7:

N91G97S280T8.8M6M3

N92 G84R7Z-15K1

N93X-51.96Y-30

N94X51.96

N95XY60

N96G80Z50M5

Данный фрагмент программы задает обработку трех отверстий. В кадре N92 задан шаг резьбы **K1**. Система автоматически рассчитывает величину подачи на основе информации по адресам **S** и **K**.

### **1.5. Особенности постоянных циклов**

Если внутри постоянного цикла программируется кадр типа **X,Y,R** или же **X,Y,R** и/или **Z**, размеры **R** и/или **Z** постоянного цикла будут изменены, и движения осей будут выполнены в следующем порядке:

X и Y;

R обновленная;

Z обновленная.

Это позволяет изменять глубину отверстия и переходить от обработки на одной плоскости к обработке на плоскости ниже без отмены постоянного цикла функцией G80.

*Пример* приведен на рис. 6.8:

N35(DIS,".....")

N36G97S1000F100T4.4M6

N37G81R3Z-42M3

N38X15Y15

N39X65

N40Y85R-13

N41X15

N42G80Z50M5

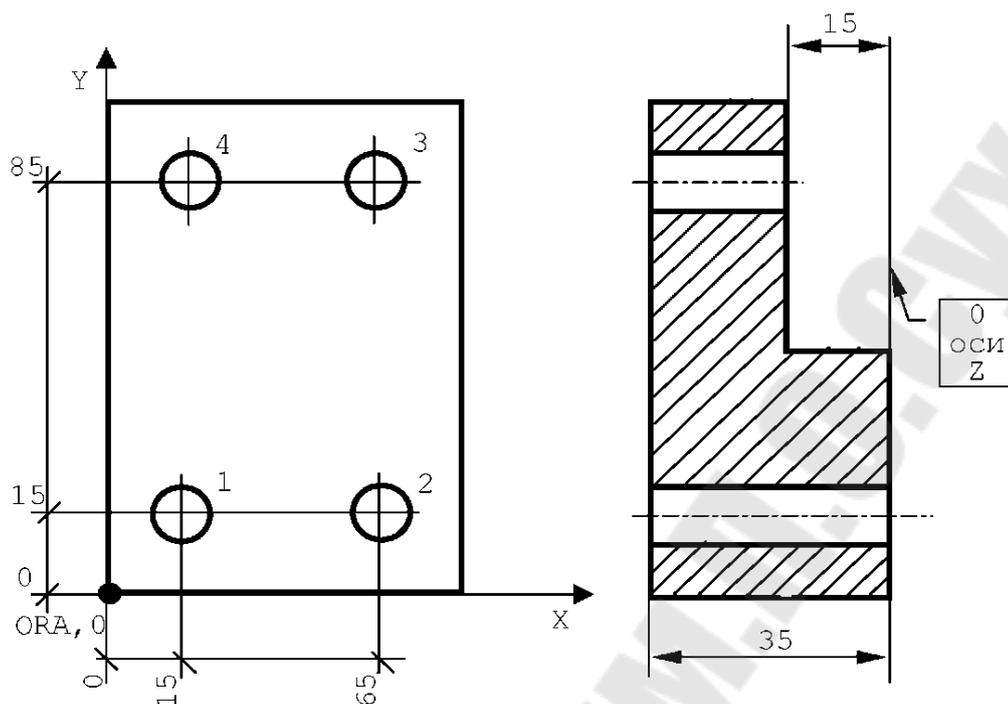


Рис. 6.8. Использование постоянного цикла сверления с применением R параметров

В рассматриваемом примере обработка отверстий в точках 1 и 2 (кадры N38, N39) осуществляется в соответствии с параметрами цикла сверления по функции G81, заданными в кадре N37(R3, Z-42).

Кадр N40, кроме координат точки 3, задает новое значение координаты точки начала обработки и возврата в конце обработки отверстия (R-13).

Таким образом, отверстия в точках 3 и 4 обрабатываются с параметрами цикла R-13, Z-42 при выполнении кадров N40 и N41 соответственно.

Внутри постоянного цикла, если программируется кадр типа **X Y R1 R2** (последнее отверстие нижней плоскости), размеры **R1** и **R2** постоянного цикла будут обновлены, и движения будут выполнены в следующем порядке:

- X и Y перемещения к точке;
- выполнение постоянного цикла с обновленными R1 и R2 (в конце цикла шпиндель подходит с быстрой скоростью к новой точке возврата R2), что дает возможность перейти от обработки нижней плоскости к обработке более высокой плоскости без отмены постоянного цикла. Для обработки первого отверстия на более высокой плоскости следует запрограммировать кадр типа XYR2.

Пример приведен на рис. 6.9.  
 N42(DIS,"PLITA")  
 N43G95G97S1000F100T5.5M6  
 N44G81R-18 Z-46M13  
 N45 X25Y25  
 N46 X60R-18R-8  
 N47Y75R-8R2Z-25  
 N48Y175R-3Z-46  
 N49X95  
 N50G80Z50M5

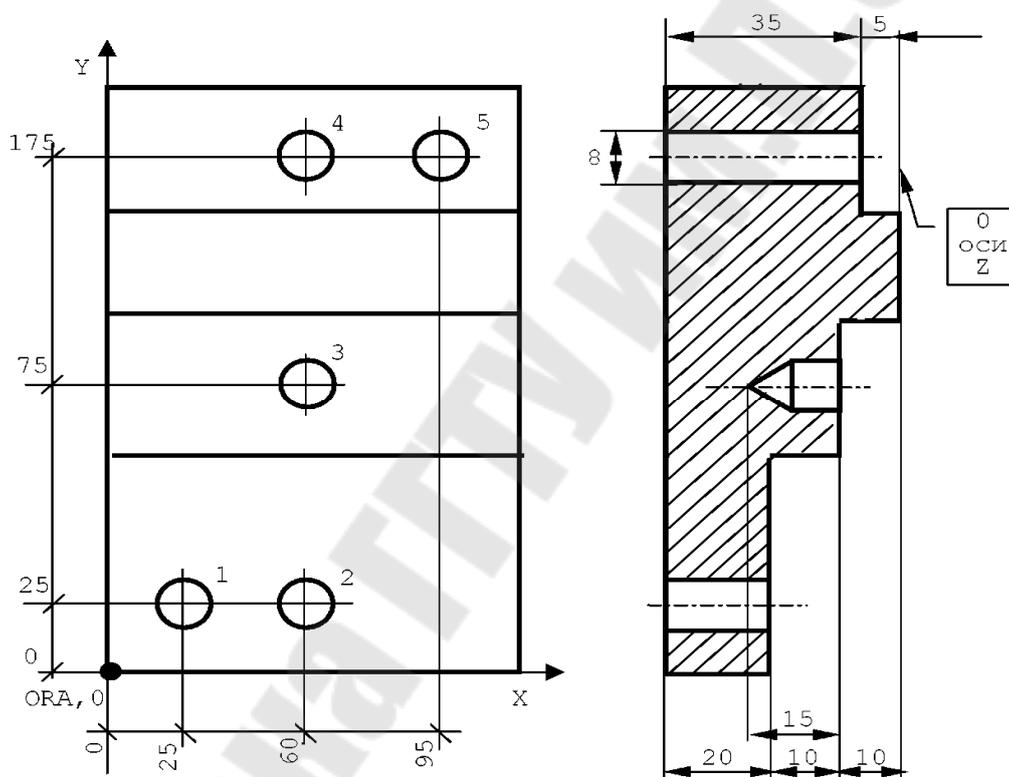


Рис. 6.9. Использование постоянного цикла сверления с применением R1 и R2 параметров

При выполнении этой программы в точке 1 (кадр N45) обеспечивается выполнение постоянного цикла с параметрами, заданными в кадре N44 (R-18, Z-46), где параметр R1 = R2. В кадре N46 указаны новые значения параметров цикла, где R1 не равно R2, а глубина сверления осталась прежней. Параметр R-8 обеспечивает выход инструмента после обработки отверстия в новую позицию для последующего перемещения инструмента в точку 3.

В кадре N47, кроме значений положения инструмента в начальной и конечной точках цикла, указана новая глубина сверления (Z-25) в точке 3. Новое значение параметра R2 обеспечивает дальнейшее перемещение инструмента в позицию точки 4. При обработке кадров N48 и N49 выполняется сверление отверстий в точках 4 и 5 с одинаковыми параметрами цикла на глубину (Z-46). Положение точки начала и конца цикла определяется параметром R=3 (R1=R2).

1.6. *Задание:* Разработать УП для сверловки пяти отверстий в детали «PLITn», используя постоянные циклы сверления и R параметры в редакторе «Блокнот», распечатать и предоставить преподавателю на флэш-памяти для прогона на эмуляторе. Быть готовым дать комментарии по каждому кадру программы.

**Исходные данные:**

Координаты положения инструмента в начале программы: X = -50.0, Y = -50.0, Z = 150.0; координаты L заданы в абсолютной системе исчисления; n – номер варианта, табл. 6.2.

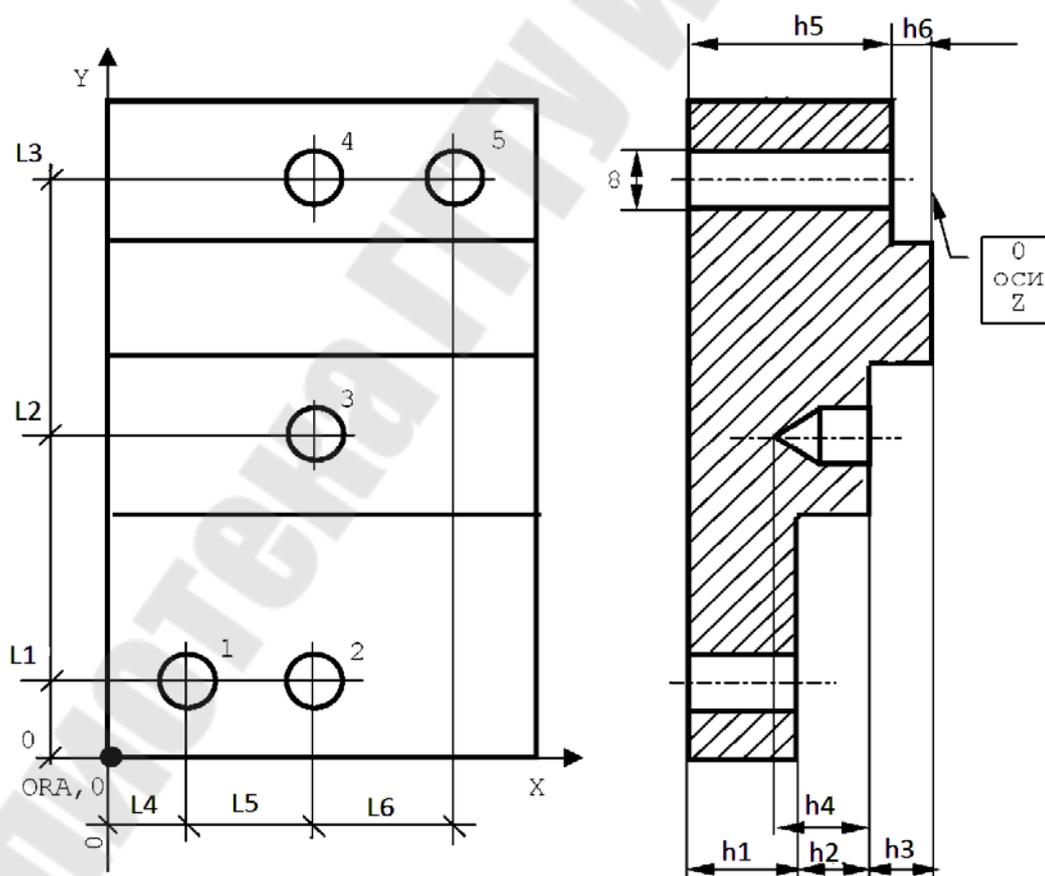


Рис. 6.10. Использование постоянного цикла сверления с применением R1 и R2 параметров

Таблица 6.2

## Исходные данные к заданию

Номер варианта	L1	L2	L3	L4	L5	L6	h1	h2	h3	h4	h5	h6
1	30	80	180	30	65	100	25	15	15	20	40	10
2	32	82	182	32	67	102	27	17	17	22	42	12
3	34	84	184	34	69	104	29	19	19	24	44	14
4	36	86	186	36	71	106	31	21	21	26	46	16
5	38	88	188	38	73	108	33	23	23	28	48	18
6	40	90	190	40	75	110	35	25	25	30	50	20
7	42	92	192	42	77	112	37	27	27	32	52	22
8	44	94	194	44	79	114	39	29	29	34	54	24
9	46	96	196	46	81	116	41	31	31	36	56	26
10	48	98	198	48	83	118	43	33	33	38	58	28
11	50	100	200	50	85	120	45	35	35	40	60	30
12	52	102	202	52	87	122	47	37	37	44	62	32
13	54	104	204	54	89	124	49	39	39	46	64	34
14	56	106	206	56	91	126	51	41	41	48	66	36
15	58	108	208	58	93	128	53	43	43	50	68	40

*Лабораторная работа № 7*  
**РАЗРАБОТКА УПРАВЛЯЮЩЕЙ ПРОГРАММЫ  
ДЕТАЛИ VALn...**

**Цель работы:**

1. Изучить методику привязки инструментов к нулевой точке локальной системы координат детали на токарном станке 16А20Ф3 с УЧПУ N201М, изложенную в настоящей работе.

2. Используя методики составления УП ступенчатого валика с резьбой, обработки цилиндрических поверхностей с применением трехбуквенных кодов путем многократного повторения одинаковых кадров и визуализацией обработки, обработки сферических поверхностей, нарезания резьб путем применения специальных файлов (лаб. работа № 4), применения параметрического программирования (лаб. работа № 5), а на основании знаний, полученных из предыдущих лабораторных работ и учебных пособий, разработать УП VAL с привязкой инструмента к нулевой точке локальной системы координат детали и с внесением корректоров в таблицу корректоров.

В качестве учебного пособия используются:

1. Методические рекомендации по выполнению работ № 1–6, лекционный материал.

2. Руководство оператора (РО УЧПУ NC-201, NC-201М, NC-202, Балт-Систем, Санкт-Петербург, 2008 г., [www.bsystem.ru](http://www.bsystem.ru)).

3. Руководство программиста (РП УЧПУ NC-110, NC-201, NC-201М, NC-202, NC-210, NC-220, NC-230, Балт-Систем, Санкт-Петербург, 2008 г., [www.bsystem.ru](http://www.bsystem.ru)).

4. *Задание:* Разработать УП обработки детали «Valn» в редакторе «Блокнот», распечатать и предоставить преподавателю на флэш-памяти для прогона на эмуляторе. Быть готовым дать комментарии по каждому кадру программы.

Координаты расположения вершины резцов необходимо выбирать в соответствии со своим вариантом.

1. Для нарезания резьбы необходимо использовать стандартный файл F11, с применением не менее двух черновых и одного чистового прохода.

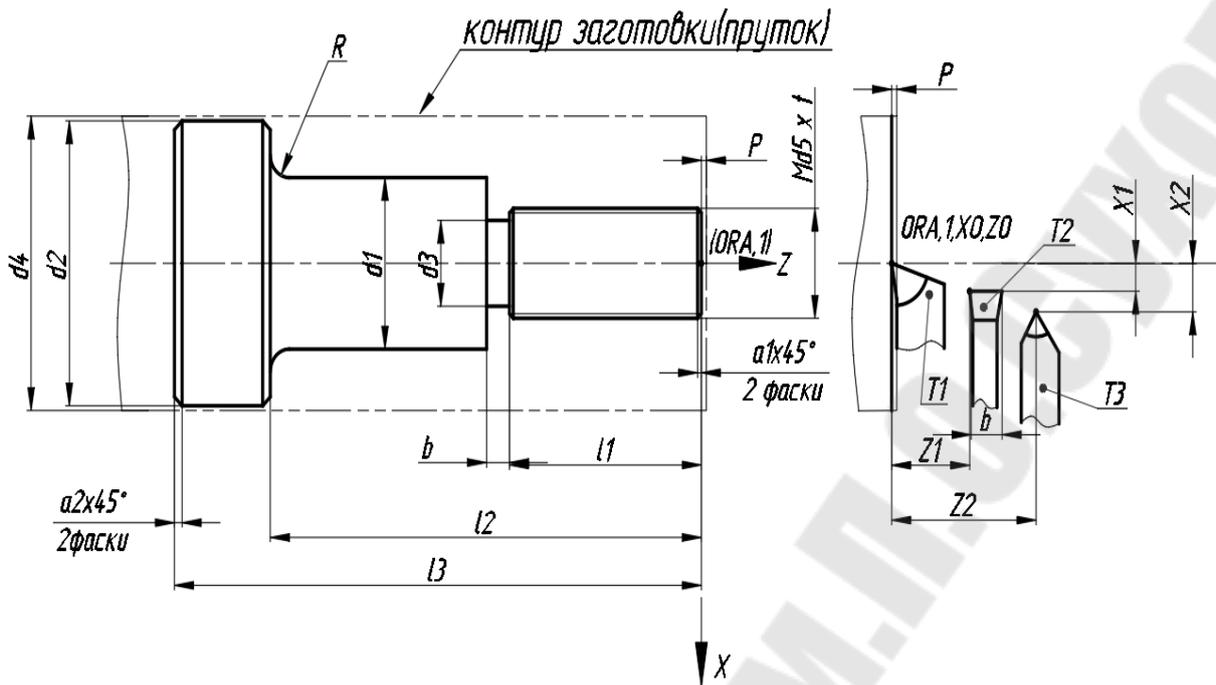


Рис. 7.1. Чертеж детали VAL...

На рис. 7.1  $d_4$  – диаметр заготовки (контур обозначен штрихпунктирной линией). Материал сталь 45; (UAO,1) – точка начала локальной системы координат детали; O – точка начала координат Z, X; P – торцовый припуск; T1 – проходной резец, материал режущей части твердый сплав T15K6; T2 – канавочно-отрезной резец, материал режущей части твердый сплав T15K6; T3 – резьбонарезной резец для метрической резьбы, материал режущей части твердый.

Таблица 7.1

### Корректоры

Координаты инструмента	Варианты															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
T1	Z0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	X0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
T2	Z1	5	10	15	10	20	18	26	30	40	16	34	15	22	30	16
	X1	22	25	34	22	35	17	28	34	25	22	26	14	17	22	30
T3	Z2	15	22	34	39	14	27	40	20	65	25	56	29	55	65	20
	X2	34	35	26	15	32	43	34	55	28	16	28	15	25	33	16

## Исходные данные детали VALn

Обозначение	Варианты														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
d1	14	16	18	20	22	24	26	28	32	34	38	40	44	46	50
d2	26	28	30	32	34	36	38	40	44	46	50	52	56	58	62
d3	8,4	10	12	13	14	16	18	19	22	24	27	30	33	35	38
d4	30	32	34	36	38	40	42	44	48	50	54	56	60	62	66
d5	10	12	14	16	18	20	22	24	27	30	33	36	39	42	45
11	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48
12	44	46	48	50	52	54	56	58	60	62	64	66	68	70	72
13	54	56	58	60	62	64	66	68	70	72	74	76	78	80	82
a2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a1	1	1	1,5	1,5	2	2	2	2,5	2,5	3	3	3	3	3,5	3,5
b	3	3	3	3	3	3	3	5	5	5	5	6	6	6	6
Mxt	10x 1,5	12x 1,5	14x 2	16x 2	18x 2,5	20x 2,5	22x 2,5	24x 3	27x 3	30x 3,5	33x 3,5	36x 4	39x 4	42x 4,5	45x 4,5
R	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4

Примечание. n – номер варианта задания.

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ № 7

### 1. Методика привязки инструмента при токарной обработке к локальной системе координат станка

#### 1.1. Привязки локальной системы координат детали к абсолютной системе координат токарного станка

Для привязки локальной системы координат детали к абсолютной системе координат станка необходимо:

1. Включить станок, выбрать переключателем режимов работы режим «НОМЕ», выбрать сначала ось X, затем Y и вывести оси в 0 (нуль) абсолютной системы станка (рис. 7.1).

2. Установить и закрепить заготовку в трехкулачковом патроне станка.

Зажим заготовки с помощью 3-кулачкового патрона в станке осуществляется вручную или с помощью электрического привода путем нажатия на педали или кнопок «зажим/разжим» детали на станочном пульте. До нажатия на педали управления или кнопки необ-

ходимо одной из клавиш F3 выбрать команду «КОЛЬЦО» или «ВАЛ», а затем осуществить зажим/разжим патрона, вставить заготовку и зажать заготовку в патроне.

Проверить положение пиноли задней бабки. Пиноль задней бабки должна либо поджимать деталь с заданным усилием, либо при короткой заготовке должна находиться в крайнем правом положении.

3. Закрепить инструмент, соответствующий технологическому процессу обработки заготовки, в соответствующие позиции поворотной инструментальной головки.

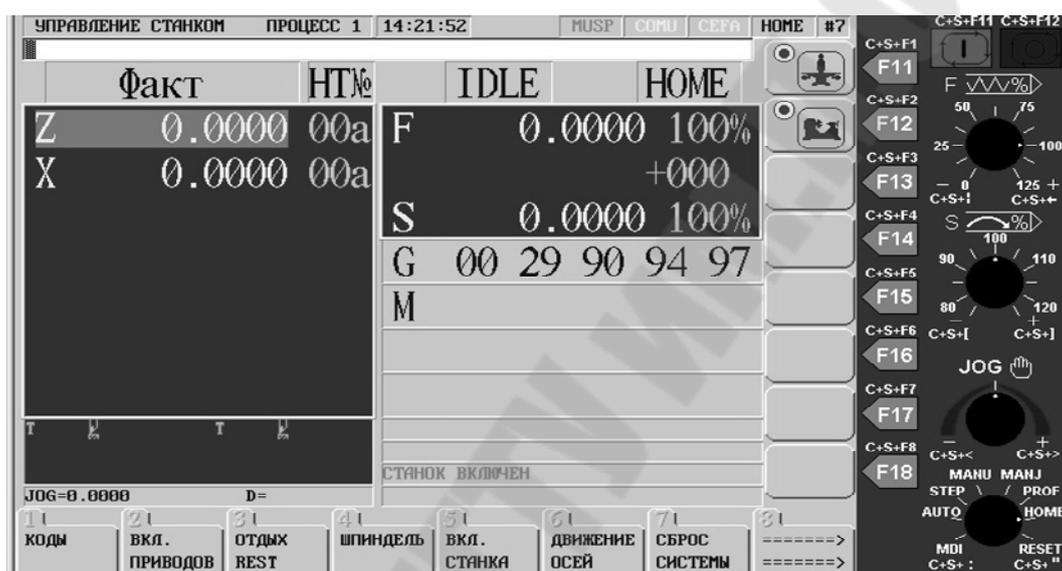


Рис. 7.1. Вывод осей в абсолютную систему станка

Перед поворотом инструментального диска для установки инструмента в соответствующие гнезда необходимо отвести головку в безопасное положение, точку смены инструмента. Если инструментальная головка будет находиться в непосредственной близости от шпинделя станка с заготовкой, при повороте может произойти поломка инструмента или повреждение патрона.

Для перемещения головки в безопасное положение необходимо установить переключатель режимов работы в положение «MANU», с помощью клавишей крестового переключателя выбрать ось, с помощью переключателя JOG выбрать направление, скорость и нажатием кнопки «Пуск» переместить инструментальную головку в безопасную зону.

Для поворота инструментальной головки в нужную позицию необходимо установить переключатель режимов работы в положение «MDI», в командной строке режима «Управление станком» ввести

команду «Т» с номером нужной позиции, например, для работы инструментом, закрепленным, например, в позицию 1 инструментальной головки, вводится команда «Т1.1М6», закрыть рабочую зону станка ограждением и нажать кнопку «ПУСК» (рис. 7.2).

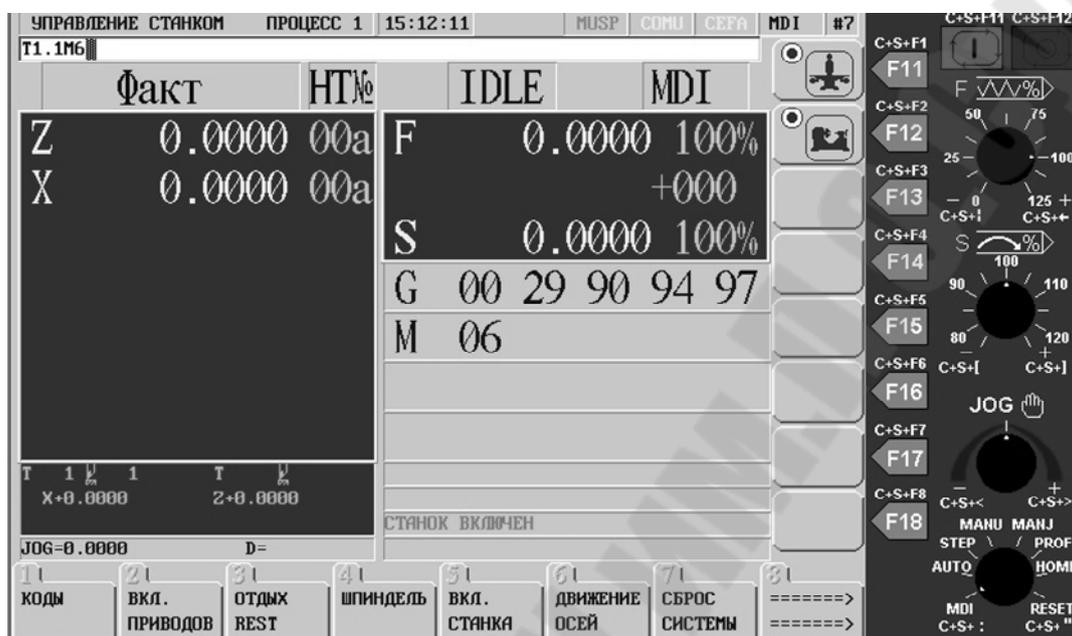


Рис. 7.2. Установка инструмента № 1

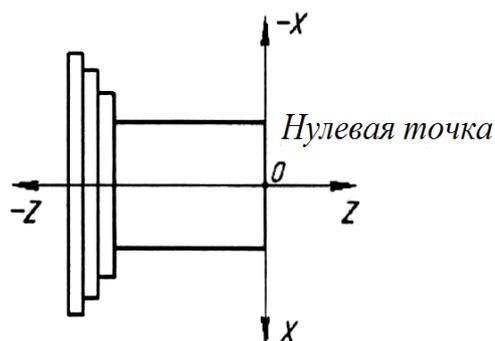
Необходимо помнить, что вращение шпинделя и подача суппорта в автоматическом режиме работы заблокированы, если:

- не зажата заготовка в механическом патроне;
- пиноль задней бабки либо не поджимает деталь с заданным усилием, либо не отведена в крайнее правое положение;
- рабочая зона станка не закрыта ограждением станка.

Для того чтобы обработать заготовку, необходимо установить локальную систему координат детали. Локальную систему координат необходимо «привязать» к «начальной (нулевой) точке» детали. Нулевая точка выбирается и устанавливается относительно конструктивного элемента. Для детали (заготовки) типа «тела вращения» этим элементом, как правило, является центр правого торца (рис. 7.3).

Это облегчает технологу подготовку программы управления, а оператору легче «привязаться» к этой точке и контролировать перемещения инструментов во время работы.

### Система координат детали



### Абсолютная система координат станка



Рис. 7.3. Абсолютная (глобальная) система координат станка и (локальная) система координат детали

Привязка локальной системы координат к оси и торцу детали осуществляется методом пробной стружки с измерением фактически полученного диаметра.

До начала привязки необходимо повернуть инструментальную головку в ту позицию, где установлен резец, с помощью которого будут выполняться подрезка торца и проточка диаметра. Это, как правило, резец № 1.

Для этого необходимо установить переключатель режимов работы в положение «MDI», закрыть ограждение, в командной строке ввести команду, например, «T1.1M6» с номером нужного инструмента и нажать кнопку «ПУСК». После этого можно начать процесс привязки инструмента, находящегося, например, в позиции «1» к «нулевой точке» заготовки и тем самым установить локальную систему координат детали.

Перед привязкой инструмента 1 необходимо обнулить корректор 1 в файле корректоров FILCOR. Для этого необходимо нажать клавишу «F4» в меню «Ввод корректора». При этом в командной строке появится символ  $\square$ . Далее необходимо ввести с клавиатуры номер корректора, имя осей, нулевые координаты по осям и нажать клавишу «ENTER».

Пример:  $\square$  1, X0, Z0, где: 1 – номер корректора для инструмента 1, X0, Z0 – имена осей и нулевые координаты по осям. При этом внизу слева на дисплее, где высветился символ T1, ниже высветятся значения корректора 1: X+0.0000 Z+0.0000 (рис. 7.4).

Максимальное количество локальных нулевых точек может быть не более 100 (0–99).

Нулевые локальные точки сохраняются в памяти УЧПУ в файле начальных точек FILEOR, имея свой номер, и создаются с помощью команды управления **ORA**.

Формат команды **ORA,n,Z...,X...**, где n – номер начальной точки. (Примечание: В формате команды после букв вместо двух точек должны быть вставлены числовые значения. Если точки отсутствуют, то вместо буквы n должно быть вставлено числовое значение, например: 0,1,2,... и т. д.)

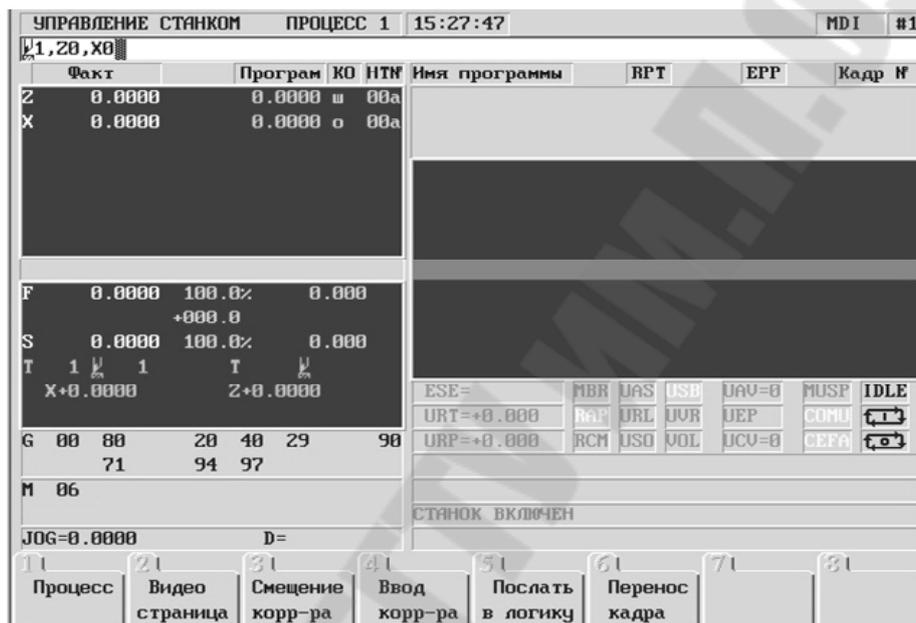


Рис. 7.4. Привязка инструмента T1.1

### 1.1.1. Привязка инструмента по оси Z

В режиме «Управление станком», видеостраница #7, перевести переключатель режимов работы станка в режим «MDI», закрыть ограждение, в командной строке набрать частоту вращения шпинделя, например, G97S300M03, и нажать кнопку «Пуск», при этом включится вращение шпинделя. После этого необходимо перевести переключатель режимов работы в режим «MANU», открыть ограждение, при этом вращение шпинделя должно продолжиться. С помощью корректора подачи «JOG» выберите скорость и направление перемещения, а с помощью крестового переключателя – ось, вдоль которой собираетесь двигаться и с помощью кнопки «ПУСК» переместитесь к торцу заготовки для его подрезки. Далее, если штурвал активируется автоматически (светится в режиме «JOG» «штурвал готов к работе»), с его помощью необходимо подрезать торец детали (рис. 7.5). Если штурвал не активен, активируйте его командой VOL=1.

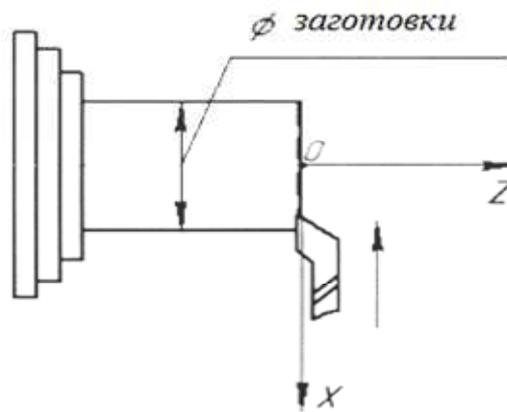


Рис. 7.5. Привязка инструмента по оси Z

Не перемещая суппорт по оси Z, необходимо ввести с клавиатуры пульта управления в командной строке команду **ORA,1,Z0** (рис. 7.6) и нажать клавишу «ENTER».

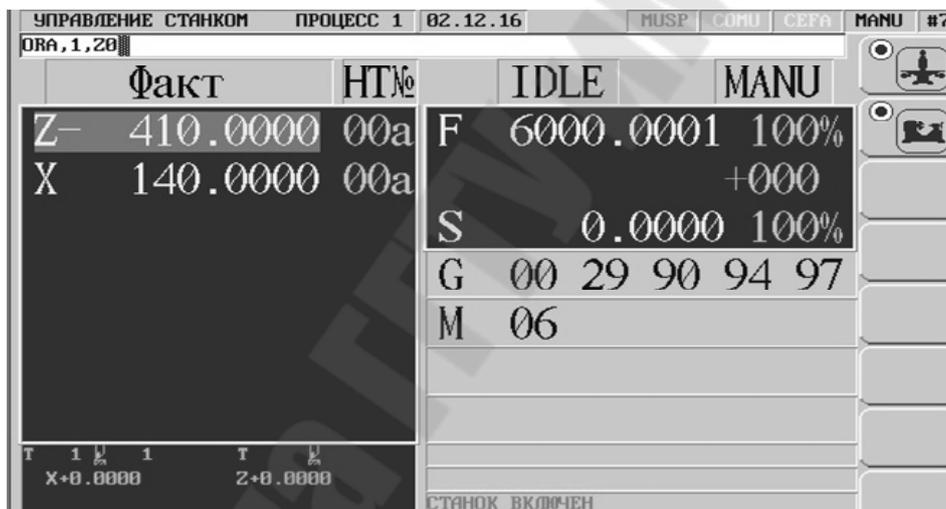


Рис. 7.6. Привязка инструмента T1.1 по оси Z

Здесь ORA – код ввода начальной точки; 1 – номер начальной точки; Z0 – положение точки начала локальной системы координат на оси Z.

Номера начальных точек детали должны начинаться с ORA,1, Z...,X... .

Номер начальной точки ORA,0,Z...,X... принадлежит абсолютной системе координат станка и при внесении в нее значений смещается абсолютная система координат станка, что делать не рекомендуется.

#### 1.1.2. Привязка инструмента по оси X

Для выполнения привязки необходимо произвести пробное снятие припуска с заготовки по диаметру (рис. 7.7) до состояния поверх-

ности как «чисто», руководствуясь вышерасположенным алгоритмом действий, при этом не перемещайте инструмент по оси X до окончания привязки, т. е. до ввода команды  $ORA,n,Z..X...$

После обработки необходимо измерить обработанный наружный диаметр заготовки и, не перемещая суппорт по оси X, ввести с клавиатуры в командной строке в режиме «Управление станком» команду:  $ORA,1,X(-d)$ , где d – значение измеренного диаметра со знаком минус.

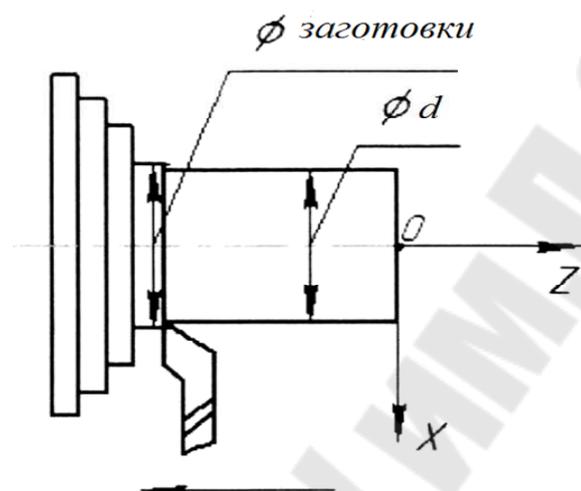


Рис. 7.7. Привязка инструмента T1.1 по оси X

Например, если измеренный диаметр 45 мм, команда записывается следующим образом (рис. 7.8):  $ORA,1,X-45$ . Величина «-45» означает, что начальная точка переместится по оси X в отрицательном направлении (к оси детали) на величину радиуса, т. е. на 22,5 мм. Следовательно, она окажется на оси детали, что нам и требовалось. После ввода команды необходимо нажать клавишу «ENTER».

Необходимо помнить, что все координаты по оси X вводятся как диаметральные размеры, что устанавливается при характеристике станка.

Выполнить проверку правильности привязки инструмента нужно следующим образом: в режиме «Управление станком» MANU, используя клавиши крестового переключателя и штурвал, расположенные в секции функциональной клавиатуры и станочной консоли и штурвал, переместить инструмент на некоторое расстояние от точки привязки и вызвать нулевую точку локальной системы координат заготовки.

Для вызова нулевой точки необходимо установить режим «MDI», закрыть ограждение, ввести в командную строку (UАО,1) и нажать клавишу «Пуск».

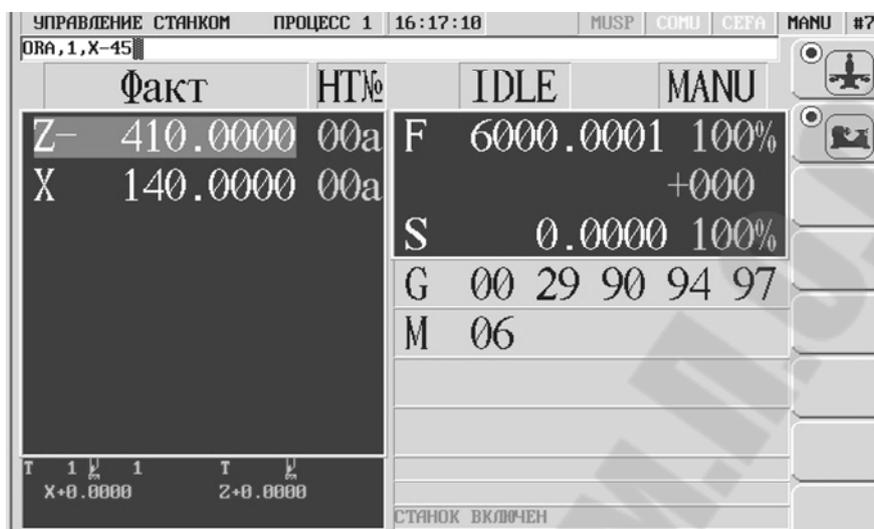


Рис. 7.8. Привязка инструмента T1.1 по оси X

Установить корректором подач «JOG» минимально возможное значение и направление подачи и с помощью кнопки «Пуск» и штурвала переместить инструментальную головку до касания вершиной режущей кромки резца торца и наружного диаметра заготовки, на дисплее должны высвечиваться Z=0, X=45 (рис. 7.9).

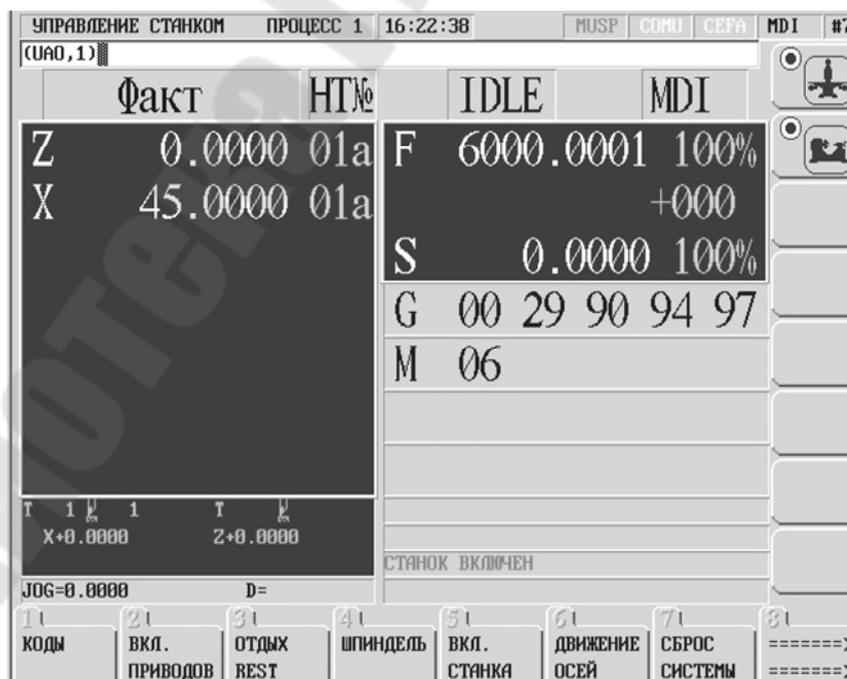


Рис. 7.9. Вызов системы координат детали

Если числа совпали с числами, вводимыми при привязке, то привязка инструмента осуществлена правильно, если не совпали, необходимо повторить процедуру привязки. После привязки локальной системы координат к нулевой точке заготовки первого инструмента необходимо привязать остальные инструменты.

### **1.2. Привязка инструмента при обработке детали двумя и более инструментами**

Технологический процесс обработки любой детали на токарном станке предполагает применение хотя бы минимум двух резцов – проходного и отрезного. При обработке сложных деталей количество применяемых инструментов ограничивается количеством позиций в поворотной инструментальной головке.

По этой причине возникает проблема привязки инструментов, связанная с их конструктивными и размерными отличиями, поскольку меняется положение режущей кромки относительно базовых поверхностей державки различных видов резцов. Решение этой проблемы заключается в применении метода корректировки положения инструментов.

Рассмотрим порядок привязки двух инструментов в позициях 1 и 8 инструментальной головки. Привязка инструмента «1» к нулевой точке локальной системы заготовки производится по описанному выше алгоритму и имеет нулевые корректора.

#### **1.2.1. Привязка инструмента «позиция 8»**

Установить нужный резец в позицию 8 инструментальной головки, повернуть головку в рабочее положение для инструмента «8», для чего необходимо установить переключатель режимов работы в положение «MDI», в командной строке режима «**Управление станком**» ввести команду «T8.8M6» (где 8 – номер инструмента; 8 – номер корректора; M6 – функция смены инструмента), закрыть ограждение и нажать кнопку «**ПУСК**». Номер корректора может быть равным от 1 до 9999 и не обязательно должен быть равным номеру инструмента. Перед привязкой инструмента 8 необходимо обнулить корректор 8 в файле корректоров FILCOR. Для этого необходимо нажать клавишу «F4» в меню «**Ввод корректора**» для обнуления корректоров инструмента 8.

При этом в командной строке появится символ . Далее необходимо ввести с клавиатуры: номер корректора, имя осей, нулевые координаты по осям и нажать клавишу «ENTER».

Пример:  $T8, X0, Z0$ , где 8 – номер корректора для инструмента;  $8, X0, Z0$  – имена осей и нулевые координаты по осям.

При этом внизу слева на дисплее, где высветился символ T8, ниже высветятся значения корректора 8:  $X+0.0000, Z+0.0000$ .

Далее необходимо ввести в командной строке режима «Управление станком» команду (UАО,1) (писать в скобках) для вызова начальной точки 1 локальной системы координат детали. Координаты перемещения инструмента 8 будут отсчитываться от точки, находящейся на оси вращения и плоскости торца детали, так как в эту точку помещена локальная система координат при привязке инструмента № 1.

Далее перевести переключатель режимов станка в режим «MANU», в режиме «Управление станком» коснуться торца заготовки. Но, как видно на рис. 7.10, из-за разных конструктивных размеров инструментов «1» и «8» положение инструмента «8» (если он короче по оси Z) в нулевом положении по оси Z смещено вправо от нулевой точки заготовки. «Жирным» показано положение инструмента № 8 в нулевой точке инструмента «1».

Для того чтобы при обработке инструментом «8» координаты перемещений отсчитывались тоже от нулевой точки заготовки, в которой размещена локальная система координат, необходимо определить величину смещения инструмента 8 от этой нулевой точки и ввести эти значения в корректор инструмента «8».

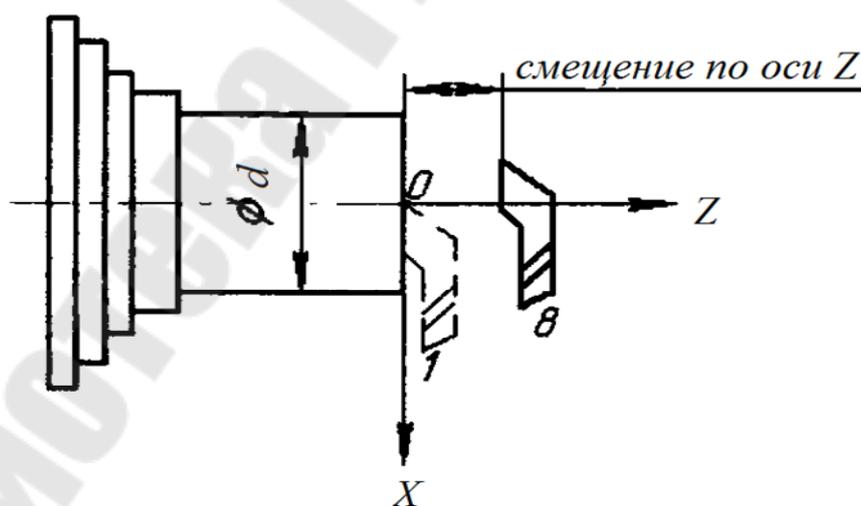


Рис. 7.10. Привязка инструмента T8 по оси Z



Для этого надо перевести переключатель режимов работы станка в режим «MANU», затем, используя «Пуск» и штурвал, переместить инструментальную головку и коснуться резцом ранее обработанного наружного диаметра заготовки.

При этом на дисплее будет высвечено значение координаты по оси X равное, например, X=46.3244 (рис. 7.13), не равное диаметру детали 45 мм.

ФАКТ		ИТ№	IDLE	MDI
Z-	8.0304	01a	F 6000.0001	100%
X	46.3244	01a	S 0.0000	100%
			G 00 29 90 95 97	
			M	

СТАНОК ВКЛЮЧЕН

Рис. 7.13. Привязка инструмента T8 по оси X

Это значит, что в нашем случае резец 8 по оси X длиннее резца 1 и требует коррекции. Чтобы определить величину коррекции, необходимо произвести следующее вычисление с учетом, что значения координат по оси X являются диаметральными, а корректоры величинами радиусными:  $(46,3244 - 45) / 2 = 0,6622$ .

То есть координата расположения режущей кромки инструмента относительно нуля детали по оси X составляет +0,6622 мм. Таким образом, так как резец 8 *длиннее* резца 1, и необходимо на +0,6622 мм откорректировать положение инструмента 8 по оси X в положительном направлении, чтобы его режущая кромка совпала с нулевой точкой заготовки в локальной системе координат детали по оси X.

Теперь можно ввести значения смещений положения инструмента 8 по осям X и Z в его корректор. Выберем для этого корректор 8.

Для ввода смещений корректора необходимо перейти из видеостраницы #7, изображенной на рис. 7.13, на видеостраницу #1 с помощью клавиши  «Прокрутка» на панели оператора.

Внести корректоры можно двумя способами:

**Первый способ.** Можно это сделать, нажав клавишу F4 меню «Ввод корректора» в режиме «Управление станком», коснувшись резцом диаметра «45 мм», внести значение  $\mu$  8,X0.6622 и нажать кнопку «ENTER». Затем коснуться резцом торца детали, внести значение  $\mu$  8,Z-8.0304 и нажать кнопку «ENTER», соблюдая процедуру, описанную выше, при этом внизу слева на дисплее, где высветился символ T8, ниже высветятся значения корректора 8,X+0.662, Z-8.0304. Эти значения будут записаны и в файл «FILCOR» (рис. 7.14).

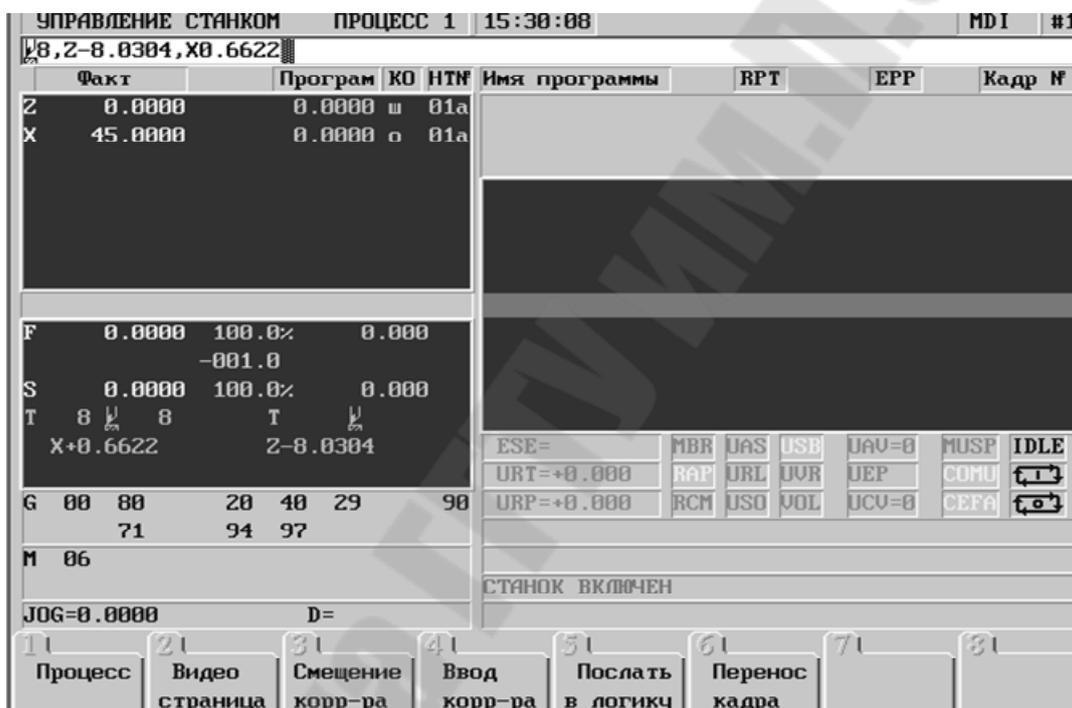


Рис. 7.14. Ввод корректоров клавишей «Ввод корректора»

**Второй способ.** Нажать клавишу «F3» опция меню «Смещение корректора», в режиме «Управление станком», и в командной строке появится значок  $\mu$ .

Затем необходимо коснуться резцом торца детали и внести значение  $\mu$  8,Z0 нажатием кнопки «ENTER», затем диаметра  $\mu$  8,X45, соблюдая процедуру, описанную выше, при этом внизу слева на дисплее, где высветился символ T8, ниже высветятся значения корректора 8: Z-8.0304 X+0.6623. Эти значения будут записаны и в файл «FILCOR»(рис. 7.15).

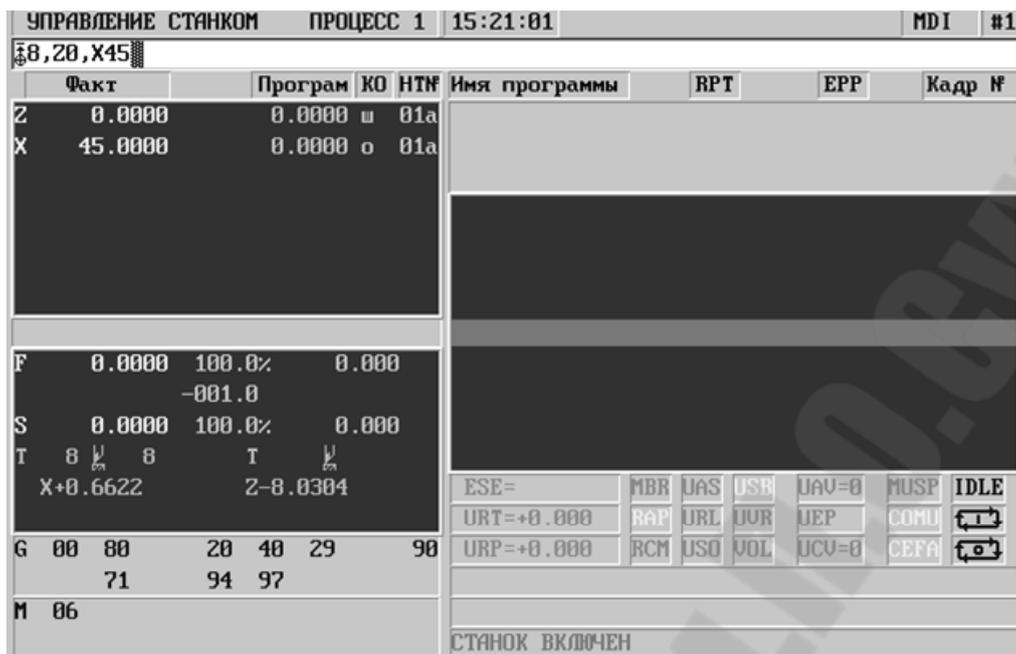


Рис. 7.15. Ввод корректоров клавишей «Смещение корректора»

Далее необходимо проверить правильность ввода смещений корректора 8 при привязке инструмента 8.

Для этого необходимо:

- ввести в командной строке режима «Управление станком» MDI команду T8.M6 для вызова инструмента 8 и (UАО,1), для вызова начальной точки 1. После смены инструмента всегда необходимо указывать эту команду, для того чтобы дать понять инструменту, введенному в обработку, в какой системе координат будет происходить отсчет по осям координат;

- установить переключатель режимов работы в положение «MANU»;

- установить переключателем корректором подач «JOG» необходимое значение подачи и направление перемещений и с помощью кнопок крестового переключателя выбрать ось;

- с помощью кнопки «Пуск» и штурвала переместить инструментальную головку до касания вершиной режущей кромки резца 8 торца и наружного диаметра заготовки.

На дисплее при этом должны высвечиваться по оси Z числа  $Z = 0$ , по оси X –  $X = 45$  (рис. 7.15). Если числа совпали, привязка инструмента 8 осуществлена правильно, если не совпали – необходимо повторить процедуру привязки.

Для других инструментов, используемых в обработке этой детали, повторить процедуру привязки аналогично резцу 8.

## РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

### *Основная*

1. Гжиров, Р. И. Программирование обработки на станках с ЧПУ : справочник / Р. И. Гжиров, П. П. Серебrenицкий. – Л. : Машиностроение, Ленингр. отд-ние, 1990. – 558 с.
2. Схиртладзе, А. Г. Технологическое оборудование машиностроительных производств : учеб. пособие для машиностр. специальностей вузов / А. Г. Схиртладзе, В. Ю. Новиков ; под ред. Ю. М. Соломенцева. – 2-е изд., перераб. и доп. – М. : Высш. шк., 2001. – 407 с.
3. Автоматизация подготовки управляющих программ для станков с ЧПУ : в 2 ч. / В. И. Аверченков [и др.]. – Брянск : БГТУ, 2010. – Ч. 2. – 211 с.
4. Фельдштейн, Е. Э. Обработка деталей на станках с ЧПУ / Е. Э. Фельдштейн, М. А. Корниевич. – Минск : Новое знание, 2005. – 287 с.

### *Дополнительная*

5. Автоматизация выбора режущего инструмента для станков с ЧПУ : монография / В. И. Аверченков [и др.]. – Брянск : БГТУ, 2010. – 147 с.
6. Инструментальная оснастка станков с ЧПУ / С. Н. Григорьев [и др.]. – М. : Машиностроение, 2006. – 544 с.
7. Руководство оператора (УЧПУ NC-201, NC-201M, NC-202), Балт-Систем. – СПб., 2008. – Режим доступа: [www.bsystem.ru](http://www.bsystem.ru).
8. Руководство программиста (УЧПУ NC-110, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230), Балт-Систем. – СПб., 2008. – Режим доступа: [www.bsystem.ru](http://www.bsystem.ru).
9. Дулько, О. Л. В помощь оператору ГПС : справоч. кн. / О. Л. Дулько. – Л. : Лениздат, 1990. – 235 с.
10. Проектирование технологии : учеб. для студентов машиностр. специальностей вузов / М. И. Бранчукова [и др.] ; под общ. ред. Ю. М. Соломенцева. – М. : Машиностроение, 1990. – 416 с.

### *Электронные учебно-методические комплексы*

11. Старовойтов, Н. А. Технология обработки на станках с числовым программным управлением : электрон. учеб.-метод. комплекс дисциплины / Н. А. Старовойтов. – Гомель : ГГТУ им. П. О. Сухого, 2015. – Режим доступа: [elib.gstu.by](http://elib.gstu.by).

## Содержание

Предисловие.....	3
<i>Лабораторная работа № 1. Изучение конструкции и технических характеристик станка мод. 16А20Ф3 с ЧПУ.....</i>	4
<i>Лабораторная работа № 2. Изучение пульта оператора УЧПУ NC201М станка мод. 16А20Ф3 .....</i>	5
<i>Лабораторная работа № 3. Приобретение навыков написания программ для токарной обработки двухступенчатого валика двумя резцами.....</i>	7
<i>Лабораторная работа № 4. Приобретение навыков написания программ для токарной обработки деталей, содержащих сферические поверхности и резьбы.....</i>	39
<i>Лабораторная работа № 5. Приобретение навыков написания программ для токарной обработки деталей с использованием параметрического программирования и трехбуквенных операторов.....</i>	63
<i>Лабораторная работа № 6. Приобретение навыков написания программ для сверлильно-фрезерной обработки с использованием постоянных циклов .....</i>	79
<i>Лабораторная работа № 7. Разработка управляющей программы детали VALn.....</i>	95
Рекомендуемая литература.....	111

Учебное электронное издание комбинированного распространения

Учебное издание

**Старовойтов Николай Андреевич**

# **РАЗРАБОТКА УПРАВЛЯЮЩИХ ПРОГРАММ ДЛЯ ТОКАРНЫХ СТАНКОВ С ЧПУ**

**ПРАКТИКУМ  
по выполнению лабораторных работ  
для студентов специальности 1-36 01 01  
«Технология машиностроения»  
дневной и заочной форм обучения**

**Электронный аналог печатного издания**

Редактор  
Компьютерная верстка

*Н. В. Гладкова  
Н. Б. Козловская*

Подписано в печать 03.11.17.

Формат 60x84/16. Бумага офсетная. Гарнитура «Таймс».

Ризография. Усл. печ. л. 6,74. Уч.-изд. л. 6,12.

Изд. № 107.

<http://www.gstu.by>

Издатель и полиграфическое исполнение  
Гомельский государственный  
технический университет имени П. О. Сухого.  
Свидетельство о гос. регистрации в качестве издателя  
печатных изданий за № 1/273 от 04.04.2014 г.  
пр. Октября, 48, 246746, г. Гомель