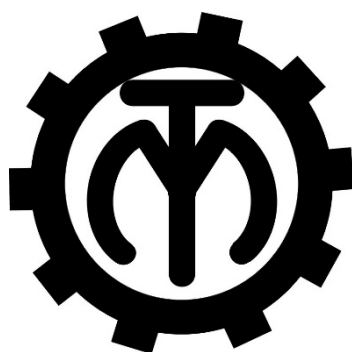


МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Технология машиностроения»

# ТЕХНОЛОГИЯ ОБРАБОТКИ НА СТАНКАХ С ЧПУ

*Методические рекомендации к практическим занятиям  
для студентов специальности 1-53 01 01 «Автоматизация  
технологических процессов и производств (по направлениям)»  
очной формы обучения*



Могилев 2022

УДК 621.9.06  
ББК 34.63  
Т38

Рекомендовано к изданию  
учебно-методическим отделом  
Белорусско-Российского университета

Одобрено кафедрой «Технология машиностроения» «22» февраля 2022 г.,  
протокол № 9

Составители: канд. техн. наук, доц. А. М. Федоренко;  
Е. Ю. Демиденко

Рецензент Д. М. Свирепа

Изложены методические рекомендации по выполнению практических занятий по дисциплине «Технология обработки на станках с ЧПУ», а также теоретические положения в области программирования обработки деталей на металлорежущих станках.

Учебно-методическое издание

## ТЕХНОЛОГИЯ ОБРАБОТКИ НА СТАНКАХ С ЧПУ

Ответственный за выпуск	В. М. Шеменков
Корректор	А. А. Подошевка
Компьютерная верстка	М. М. Дударева

Подписано в печать 14.04.2022. Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.  
Печать трафаретная. Усл. печ. л. 2,79. Уч.-изд. л. 3,06. Тираж 36 экз. Заказ № 162.

Издатель и полиграфическое исполнение:  
Межгосударственное образовательное учреждение высшего образования  
«Белорусско-Российский университет».  
Свидетельство о государственной регистрации издателя,  
изготовителя, распространителя печатных изданий  
№ 1/156 от 07.03.2019.  
Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский  
университет, 2022

## Содержание

1 Практическое занятие № 1. Программирование токарной обработки. Язык GTL.....	4
2 Практическое занятие № 2. Использование трансформации систем координат.....	13
3 Практическое занятие № 3. Программирование с использованием переменных и подпрограмм.....	23
4 Практическое занятие № 4. Программирование токарной многоцелевой обработки.....	29
5 Практическое занятие № 5. Программирование контроля.....	33
6 Практическое занятие № 6. Нормирование операций, выполняемых на станках с ЧПУ.....	39
Список литературы.....	48

# 1 Практическое занятие № 1. Программирование токарной обработки. Язык GTL

**Цель занятия:** приобретение практических навыков разработки управляющих программ токарной обработки на СЧПУ NC-201 на основе применения языка GTL.

## 1.1 Общие сведения о программировании

### *Векторная геометрия.*

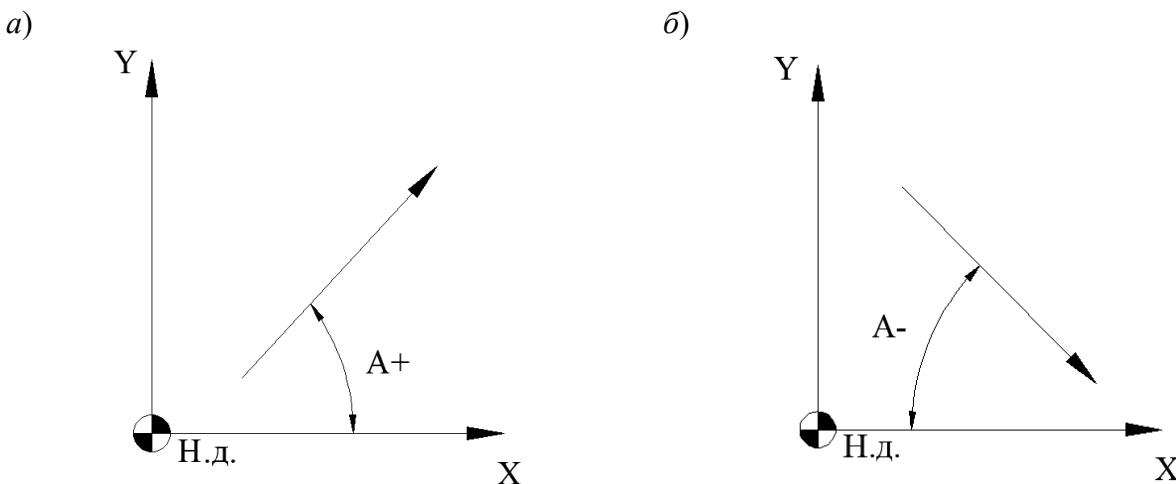
Определение профиля с использованием GTL основано на применении четырёх типов геометрических элементов: точки начала отсчета, точки, прямые и окружности.

Программирование при помощи GTL, базирующееся на векторной геометрии, требует для каждой прямой линии назначения направления движения. Направление движения прямой определяется углом, который она образует с положительной осью X. Угол будет иметь положительный знак, если ось X необходимо повернуть против часовой стрелки до совпадения с направлением линии, и отрицательный – в обратном случае, как показано на рисунке 1.1.

Направление должно быть придано также и окружностям. Условно принимается за положительное направление движение против часовой стрелки и за отрицательное – по часовой.

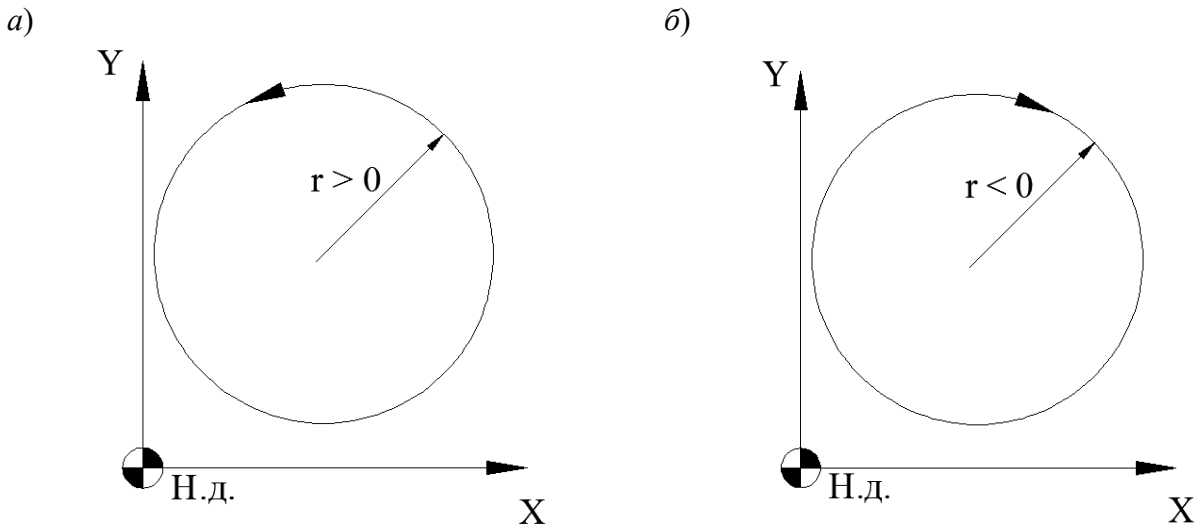
По договоренности придается положительное значение радиусам окружностей с направлением движения против часовой стрелки и отрицательное – в обратном случае (рисунок 1.2).

Направление, данное элементу, обычно соответствует направлению движения инструмента по профилю.



*а* – положительное значение угла; *б* – отрицательное значение угла

Рисунок 1.1 – Определение прямой в полярной системе координат



*a* – положительное; *б* – отрицательное

Рисунок 1.2 – Схема определения значений радиуса

*Программирование информации о геометрических элементах.*

Описание в программе геометрических элементов предусматривает использование следующих строчных символов:

- a – угол;
- l – прямая линия;
- c – окружность;
- d – расстояние;
- m – модуль;
- o – точка начала отсчета;
- r – радиус;
- p – точка;
- s – номер пересечения (дискриминатор);
- b – скос.

Формат геометрических определений предусматривает использование символа «,» (запятая) для разделения геометрического элемента (прямая – точка – окружность) от последующего геометрического элемента или информации, такой как радиус «r» или угол «a».

**Примеры**

1 Разделитель не требуется:

p1 = Z30 X30, где Z30 X30 – точка;

c1 = I10 J20 r30, где I10 J20 r30 – окружность.

2 Разделитель требуется:

L1 = Z20 X20, Z100 X-10, где Z20 X20 – точка, Z100 X-10 – точка;

L2 = I30 J20 r10, Z80 X80, где I30 J20 r10 – окружность, Z80 X80 – точка;

L3 = Z100 X100, a45, где Z100 X100 – точка, a45 – угол;

c3 = L1, L2, r18, где L1 – прямая, L2 – прямая, r18 – радиус.

### Определение точек.

Функция определения точек позволяет определить точки в прямой (явной) или косвенной (неявной) форме. Определение может быть дано как в декартовых, так и в полярных координатах (рисунок 1.3).

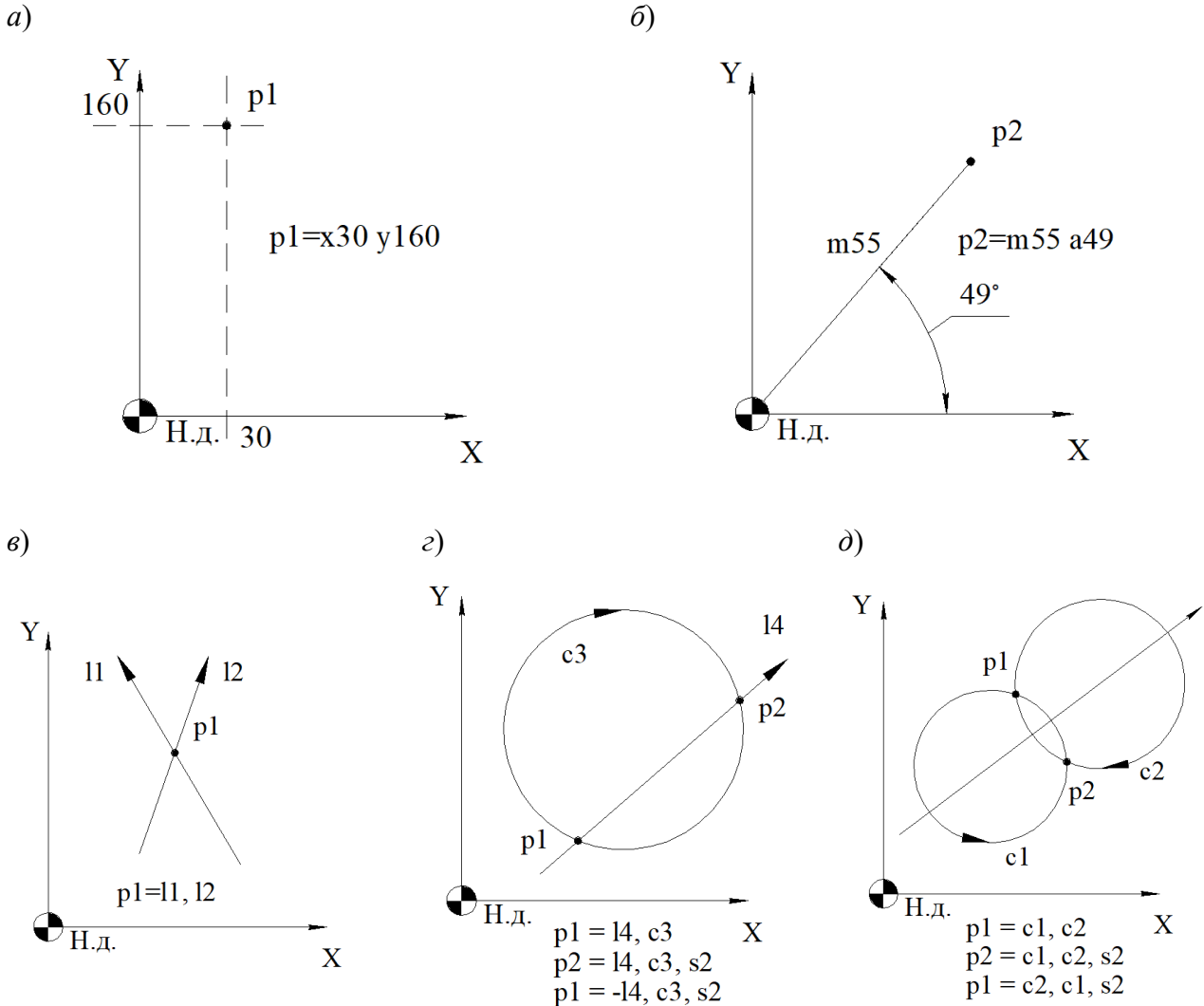


Рисунок 1.3 – Схемы описания точки

В случае пересечения прямой с окружностью или наоборот (см. рисунок 1.3, e) существуют два возможных решения: окружность  $c3$  и прямая  $l4$  пересекаются в точках  $p1$  и  $p2$ . Проходя прямую  $l4$ , следуя ее направлению, сначала встречаем точку  $p1$  (первое пересечение), а затем – точку  $p2$  (второе пересечение). Для выбора второго пересечения  $p2$  необходимо использовать индикатор  $s2$ . Если он опущен, то выбирается первое пересечение  $p1$ .

В случае пересечения окружности с окружностью существуют два возможных решения: окружности  $c1$  и  $c2$  пересекаются в точках  $p1$  и  $p2$  (см. рисунок 1.3, d). Рассматривается ориентированная прямая, соединяющая центр первой окружности с центром второй. Она делит плоскость на две полуплоскости. Для выбора точки в правой полуплоскости (если смотреть в

направлении сориентированной прямой – p2) следует использовать индикатор s2. Если он опущен, то автоматически выбирается точка в левой полуплоскости p1.

*Определение прямой линии.*

Функция определения прямой линии позволяет определить прямую линию в прямой (явной) или косвенной (неявной) форме (рисунок 1.4).

При описании геометрических элементов всегда должна быть обеспечена совместимость направлений: линия l3 совпадает с направлением окружностей и направлена от первой окружности ко второй.

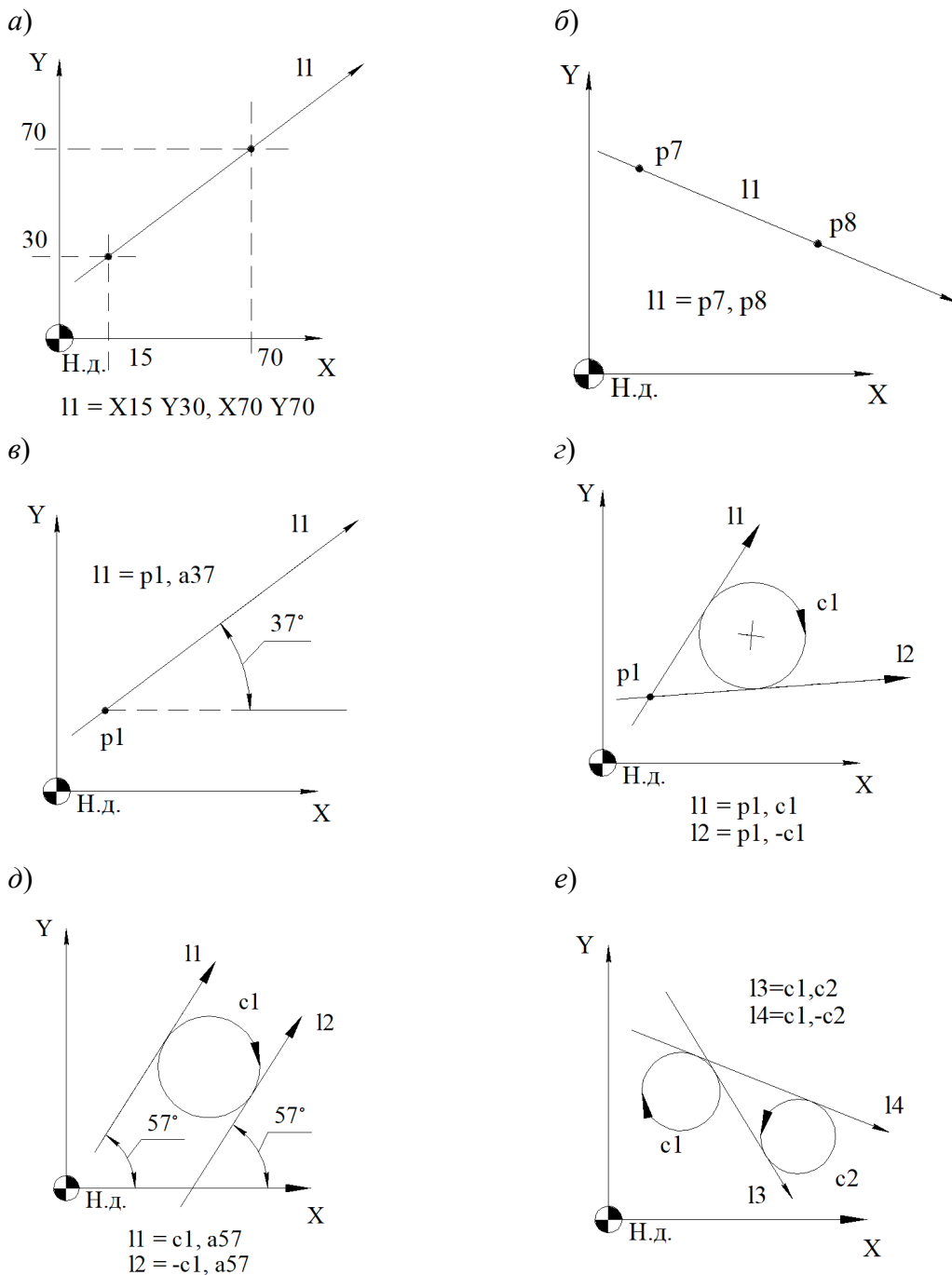


Рисунок 1.4 – Схемы описания линии

### Определение окружностей.

Язык GTL позволяет определить окружности в прямой (явной) или косвенной (неявной) форме (рисунки 1.5 и 1.6).

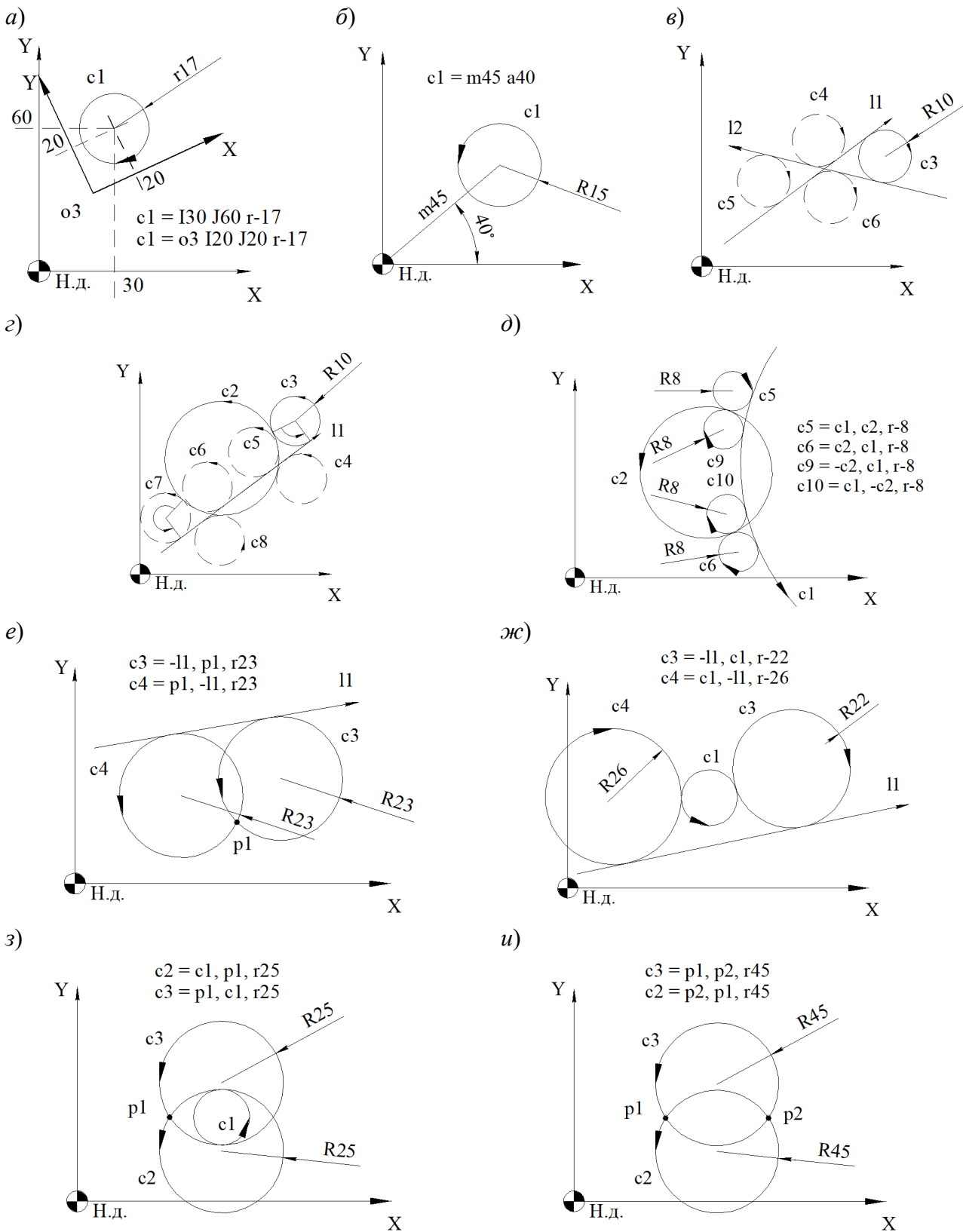


Рисунок 1.5 – Схемы определения окружностей



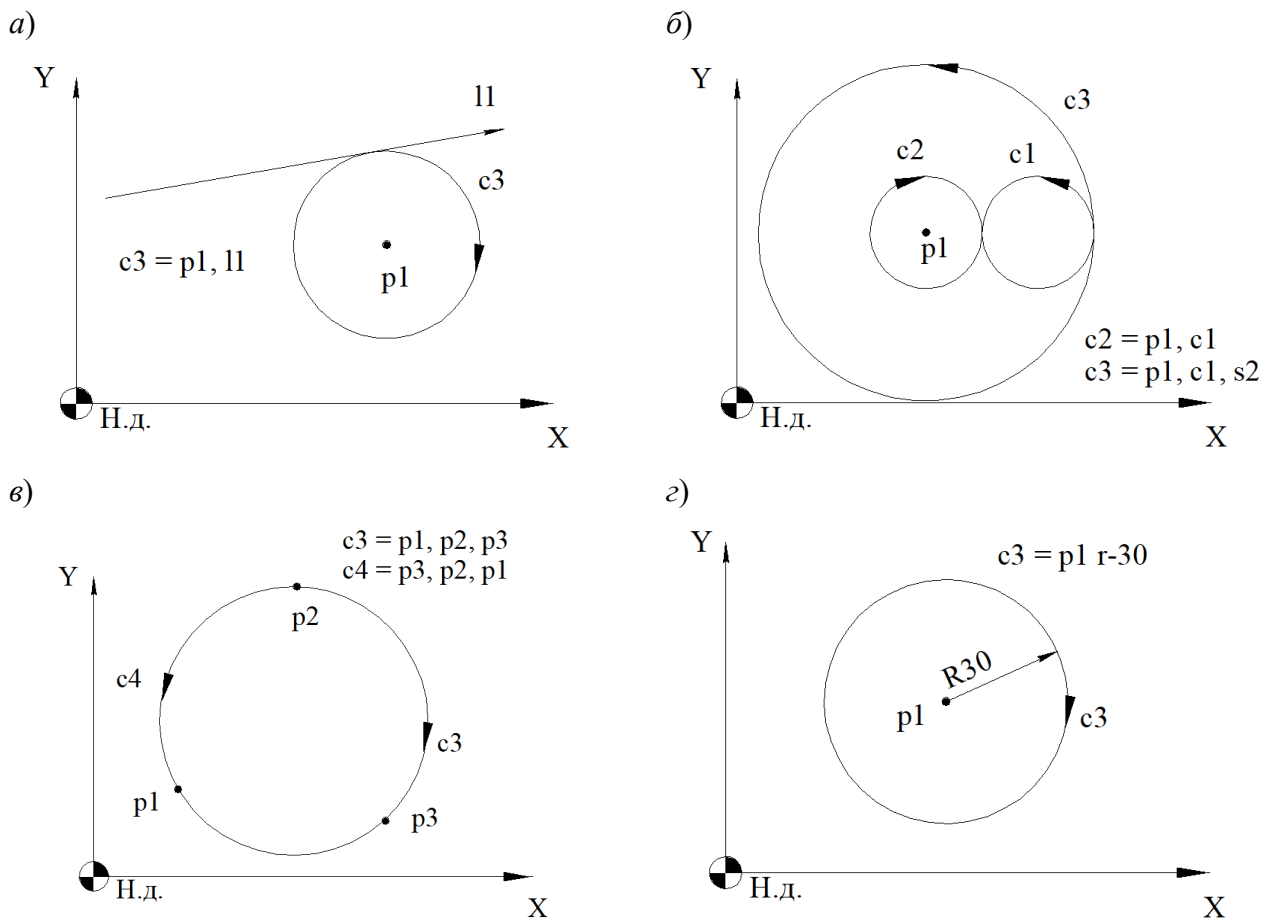


Рисунок 1.6 – Схема определения окружности точкой центра и линией

При определении окружностей следует учитывать правило: *GTL всегда создает окружность с направлением от первого ко второму элементу и дугой, имеющей меньший центральный угол.*

*Определение профиля.*

Под профилем подразумевается ряд геометрических элементов, записанных в последовательности обхода инструментом и описывающих геометрию обрабатываемой поверхности детали.

Профиль, запрограммированный в геометрии GTL, определяется через функции G21 и G20:

- G21 устанавливает начало профиля;
- G20 устанавливает конец профиля.

Профиль может быть открытым и закрытым. Открытый профиль всегда начинается с точки и заканчивается другой точкой.

Компенсация радиуса инструмента действует перпендикулярно к первому элементу на точке начала профиля и перпендикулярно к последнему элементу на точке конца профиля. Компенсация радиуса должна быть открыта на первой точке профиля программированием в кадре функций G21 G41/G42 и закрыта на последней точке с функциями G20 G40, как представлено на рисунке 1.7.

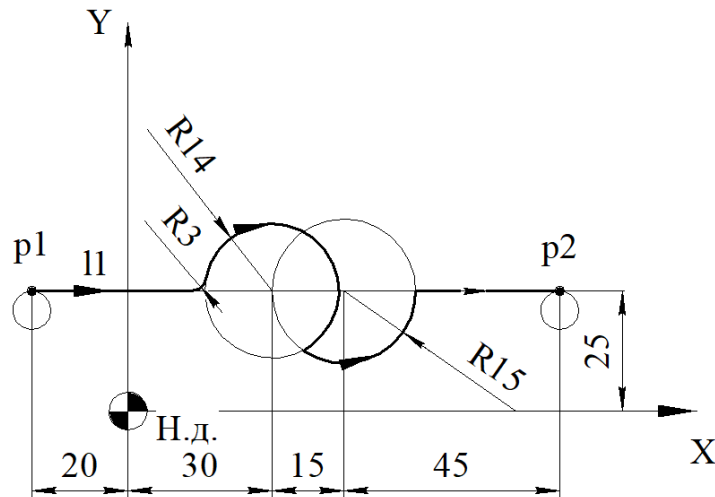


Рисунок 1.7 – Пример открытого профиля

### 1.2 Порядок выполнения практического занятия

1 Изучить состав и назначение адресов при разработке управляющей программы на основе языка GTL.

2 Для детали, изображенной на рисунке 1.8, из таблицы 1.1 по заданному варианту выбрать необходимые размеры.

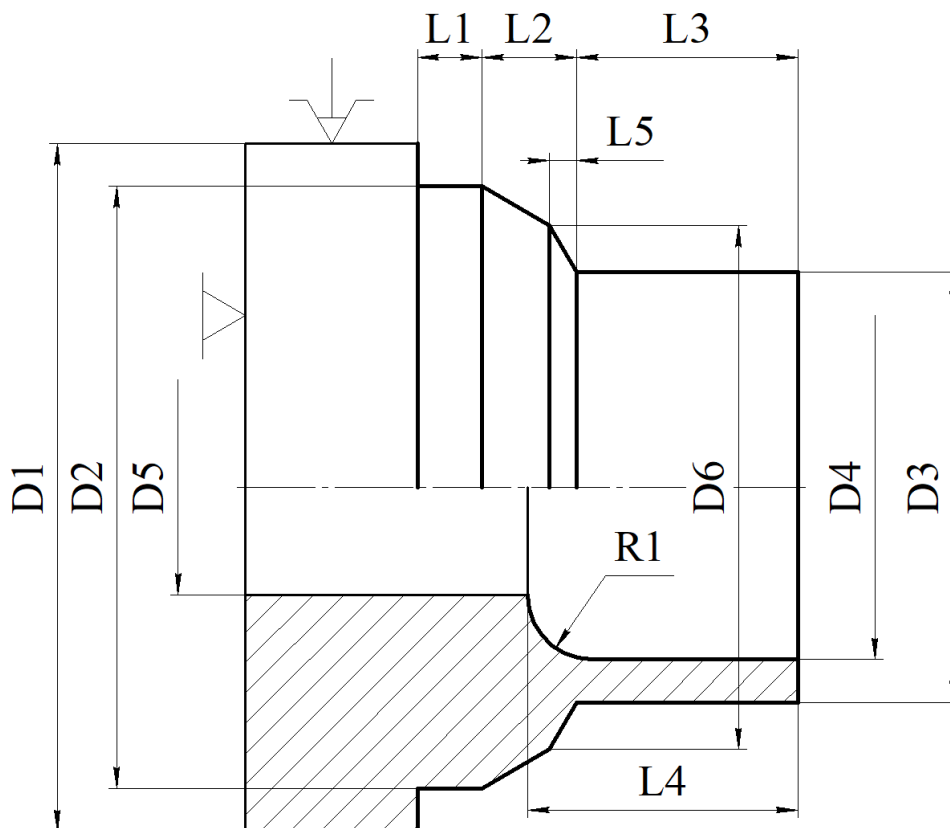


Рисунок 1.8 – Чертеж детали

Таблица 1.1 – Варианты заданий к рисунку 1.1

Вариант	D1	D2	D3	D4	D5	D6	L1	L2	L3	L4	L5	R1
1	100	90	70	50	40	80	10	10	16	24	5	20
2	120	100	76	52	48	90	12	16	18	26	9	24
3	110	100	72	54	46	84	14	14	20	28	8	22
4	150	130	100	56	44	110	16	20	22	30	10	40
5	140	120	80	58	42	100	18	28	24	32	12	25
6	130	110	74	60	40	90	20	18	26	34	10	15
7	128	122	88	62	48	100	22	16	28	18	9	14
8	144	130	96	64	46	110	24	30	30	40	15	20
9	90	84	60	50	44	70	26	12	32	50	6	10
10	160	130	90	66	42	110	28	26	34	30	13	18
11	158	140	106	68	40	120	30	22	18	32	11	24
12	98	92	70	52	46	80	32	12	20	34	6	14
13	104	88	70	54	44	60	34	8	22	36	4	14
14	124	112	80	70	42	90	36	20	24	38	10	10
15	114	100	74	60	40	90	40	18	20	42	9	12

3 С учетом особенностей станков ЧПУ установить состав переходов при обработке детали, определить количественные и качественные параметры необходимого инструмента.

4 Определить способ установки детали на станке, установить нуль детали, положение исходной точки. Построить траектории перемещений каждого инструмента.

5 Определить координаты опорных точек, заполнить таблицу.

6 Определить режимы резания для каждого инструмента.

7 Для заданной детали разработать управляющую программу для обработки на языке GTL. С целью получения траектории перемещений инструмента на графической видеостранице дополнить текст управляющей программы дополнительным кадром настройки видеостраницы UCG, в котором указать предельные размеры зоны обработки по осям Z и X.

8 Составить отчет по практическому занятию и предоставить его преподавателю для проверки и защиты.

### ***1.3 Типовое содержание отчета по практическому занятию***

- 1 Цель практического занятия.
- 2 Используемое оборудование.
- 3 Используемый инструмент.
- 4 Эскиз обработки детали.
- 5 Координаты опорных точек.
- 6 Текст управляющей программы на языке GTL:

(UCG, 1, Z\_\_Z\_\_,X\_\_X\_\_)

l1=\_\_

l2=\_\_

...

p1=\_\_

p2=\_\_

...

(DFP, 1)

G21 p1

l1

...

p2

G20

(EPF)

T\_\_ M6

S\_\_ F\_\_ M3

G0 X\_\_ Z\_\_

(SPP, 1, L2, X\_\_X\_\_, Z\_\_Z\_\_)

S\_\_ F\_\_

(CLP, 1)

G\_\_X\_\_Z\_\_

M5

M30

- 7 Ответы на контрольные вопросы.

- 8 Выводы.

#### ***Контрольные вопросы***

- 1 Какие подготовительные функции необходимо использовать для открытия и закрытия программы на языке GTL?
- 2 Как задаются окружности?
- 3 Что означает адрес «p»?

## 2 Практическое занятие № 2. Использование трансформации систем координат

**Цель занятия:** приобретение практических навыков трансформации систем координат при разработке управляющих программ обработки на СЧПУ NC-201 и СЧПУ семейства SINUMERIK.

### 2.1 Модификация системы координат

Операторы этого класса позволяют изменять декартовую систему отсчета, по отношению к которой был запрограммирован профиль. К этому классу принадлежат следующие операторы: сдвига, поворота, зеркало, масштабирование.

В связи с тем, что подходы к реализации трансформации системы координат в рассматриваемых СЧПУ разнятся, рассмотрим их индивидуально.

### 2.2 Трансформация системы координат в СЧПУ NC-201

*Определение и использование начальных точек.*

UАО – использование абсолютных начальных точек.

Оператор UАО выбирает одну из абсолютных начальных точек, ранее определенных командой ORA (таблица 2.1):

Формат: (UАО, 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.

Таблица 2.1 – Примеры задания начальных точек

Пример	Описание
(UАО,1)	Абсолютная начальная точка 1 активна для всех осей
(UАО,2,X,Y)	Абсолютная начальная точка 2 активизируется для осей X и Y
(UАО,3,B)	Абсолютная начальная точка 3 активизируется только для оси B
(UАО,0)	Активизирование нулевой начальной точки для всех осей

При включении СЧПУ 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.

*Определение и использование временных начальных точек.*

Оператор UOT выбирает абсолютную начальную точку, объявленную в кадре, меняя ее временно на величину, равную запрограммированной (сдвиг) (таблица 2.2):

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 оси. Для необъявленных осей остается в силе текущая начальная точка.

Таблица 2.2 – Примеры модификации начальных точек (рисунок 2.1)

Пример	Описание
(UAO,0)	Активируется абсолютная начальная точка 0
(UOT,0,X100,Y100)	Применяется временная начальная точка к начальной точке 0 с корректировками X100 и Y100
(UOT,1,X-250,Y-50)	Применяется временная начальная точка к абсолютной начальной точке 1 с корректировками X-250 и Y-50

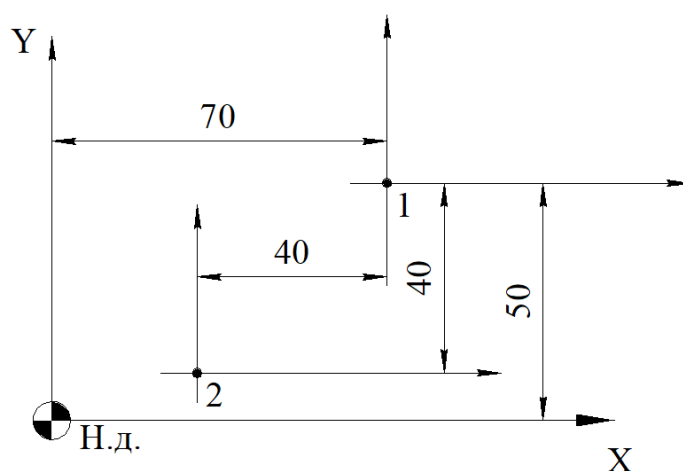


Рисунок 2.1 – Примеры начальных точек

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 UOT необходимо программировать в текущей размерности (G70, G71).

*Зеркальная обработка.*

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*

...  
N24 (MIR,X)

...  
N42 (MIR,X,Y)

...  
N84 (MIR,X)

...  
N99 (MIR)

Пример использования оператора MIR приведен на рисунке 2.2.

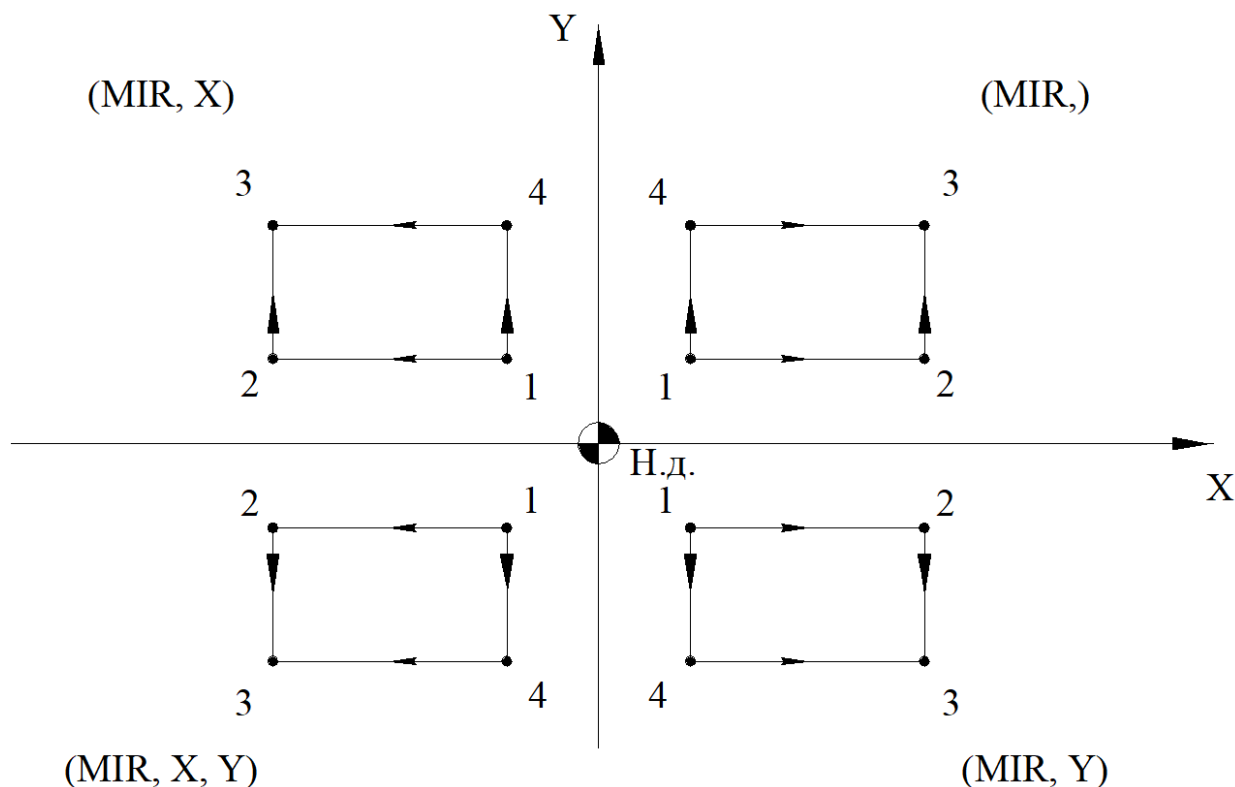


Рисунок 2.2 – Пример использования оператора MIR

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 два раза одну и ту же ось.

*Поворот плоскости.*

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.

Пример поворота плоскости представлен на рисунке 2.3.

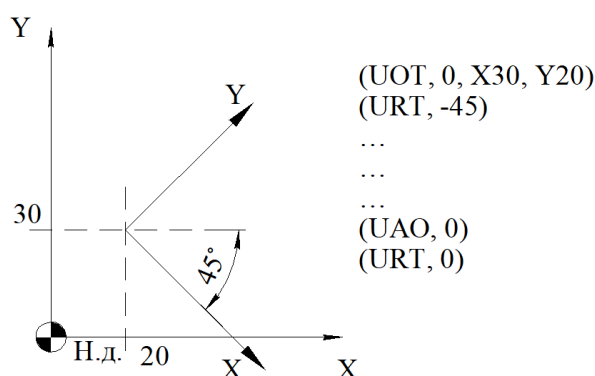


Рисунок 2.3 – Пример использования URT

*Масштабирование.*

Масштабирование используется для объявленных в операторе SCF осей (таблица 2.3).

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 типа RE (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.



Таблица 2.3 – Примеры масштабирования системы координат

Пример	Описание
(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

Может быть запрограммировано максимально шесть названий осей.

### 2.3 Трансформация системы координат СЧПУ семейства SINUMERIK

В СЧПУ семейства SINUMERIK существует понятие «фрейм». Фрейм – это автономное правило вычисления, которое переводит одну декартовую систему координат в другую декартовую систему координат.

Фрейм может состоять из следующих компонентов: сдвиг, поворот, масштабирование, отражение.

Все фреймы делятся на (рисунок 2.4):

- базовый фрейм – описывает трансформацию координат из базовой кинематической системы (BKS) в базовую систему нулевой точки (BNS);
- устанавливаемые фреймы (G54...G57, G505...G599) – предварительно устанавливаются оператором и сохраняются в памяти СЧПУ. С их помощью устанавливается система координат детали (WCS) и система нулевой точки (ENS);
- программируемые фреймы.

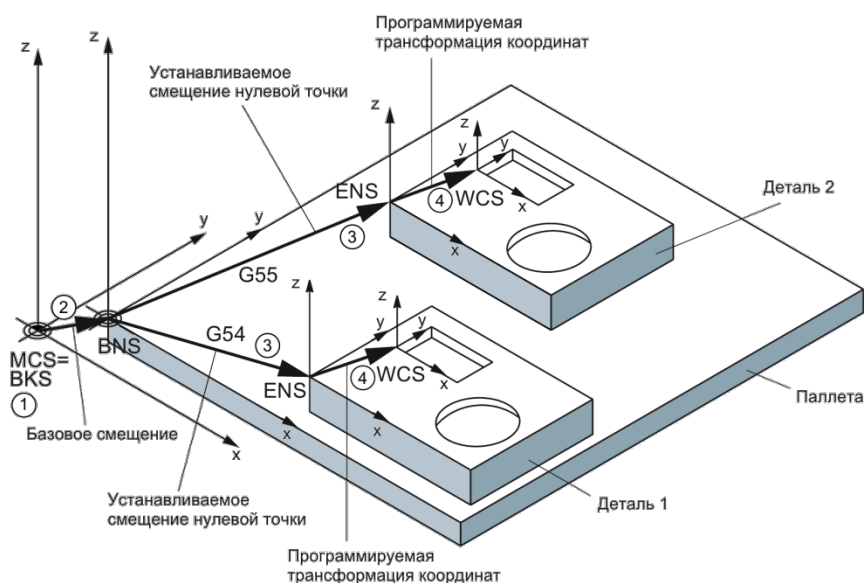


Рисунок 2.4 – Типы фреймов

Все операторы, трансформирующие систему координат, делятся на две группы:

1) замещающие операторы: TRANS, ROT, SCALE и MIRROR. Они стирают все запрограммированные до этого фрейм-операторы;

2) аддитивные операторы: ATRANS, AROT, ASCALE, AMIRROR. Они надстраиваются на уже существующие фреймы. Базой служит последняя запрограммированная нулевая точка детали.

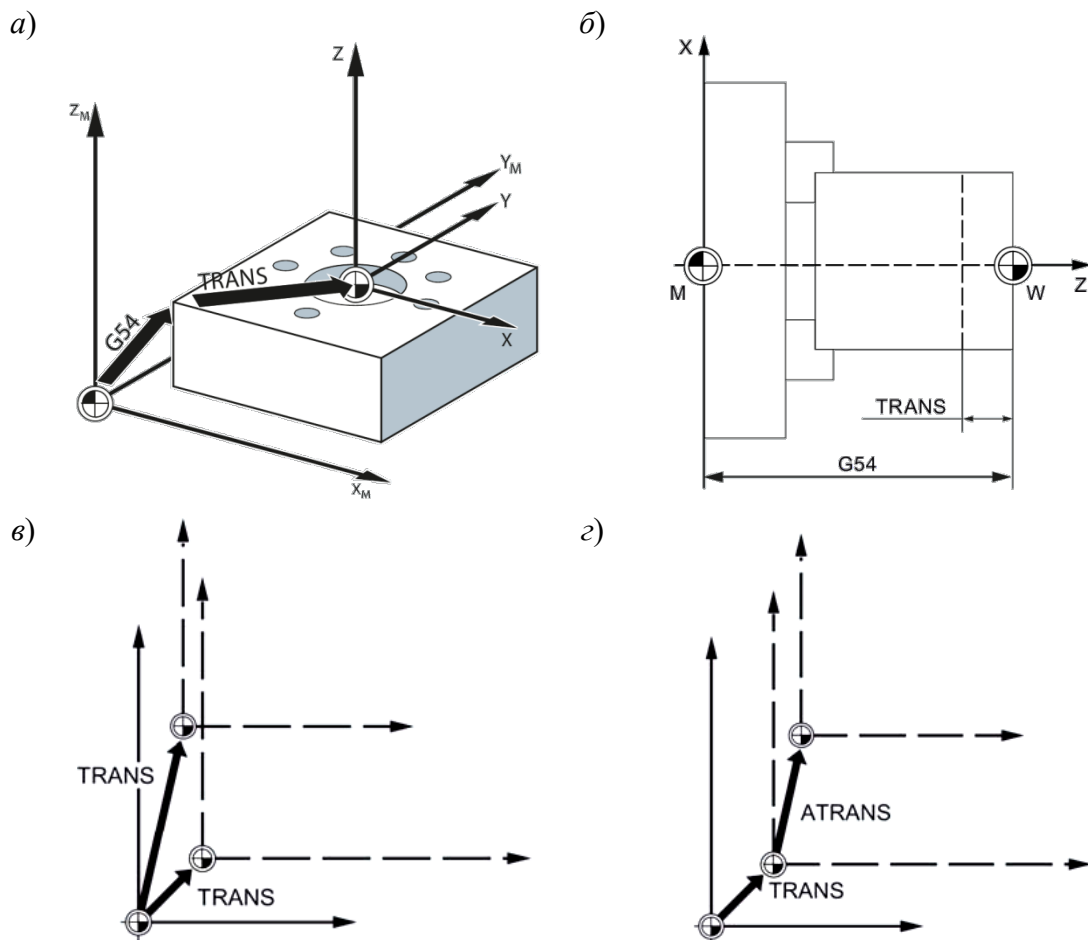
*Фрейм сдвига.*

Синтаксис фрейма сдвига (рисунок 2.5):

TRANS Xx Yy Zz (абсолютный сдвиг);

ATRANS Xx Yy Zz (инкрементальный сдвиг),

где  $x, y, z$  – значение смещения в направлении указанной геометрической оси.



*a* – пример для фрезерного станка; *б* – пример для токарного станка; *в* – пример замещающего оператора; *г* – пример аддитивного оператора

Рисунок 2.5 – Фрейм сдвига

*Фрейм поворота.*

Синтаксис фрейма поворота (рисунок 2.6):

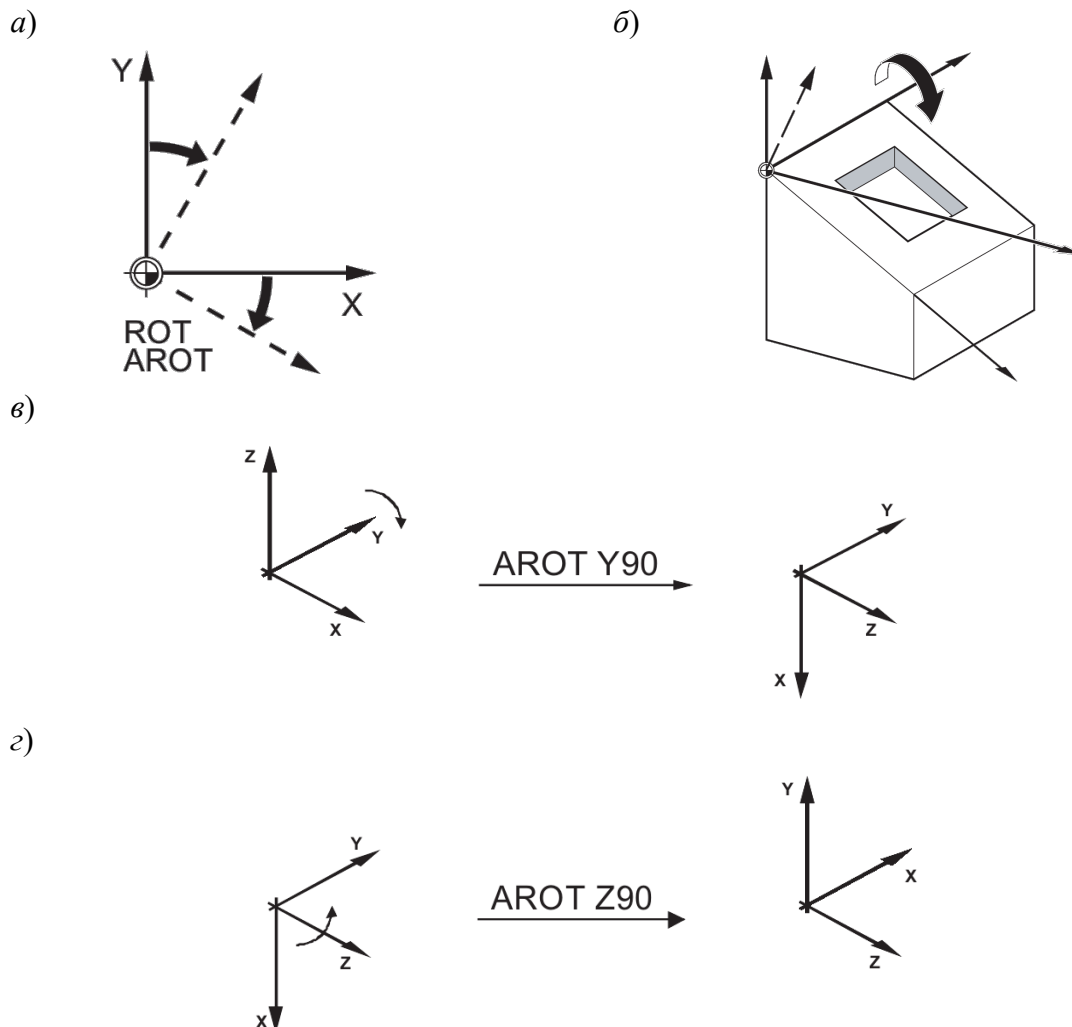
ROT Xx Yy Zz (абсолютный поворот);

AROT Xx Yy Zz (инкрементальный поворот);

ROT RPL = a (абсолютный поворот в плоскости);

AROT RPL = a (инкрементальный поворот в плоскости),

где  $x, y, z$  – значение поворота вокруг указанной геометрической оси, может принимать значения от -180 до 180;  $a$  – угол поворота в плоскости интерполяции (G17, G18, G19).



$a$  – поворот в плоскости;  $б$  – пример применения на многокоординатном станке;  
 $в, г$  – примеры вращения вокруг осей

Рисунок 2.6 – Фрейм поворота

*Программируемый коэффициент масштабирования.*

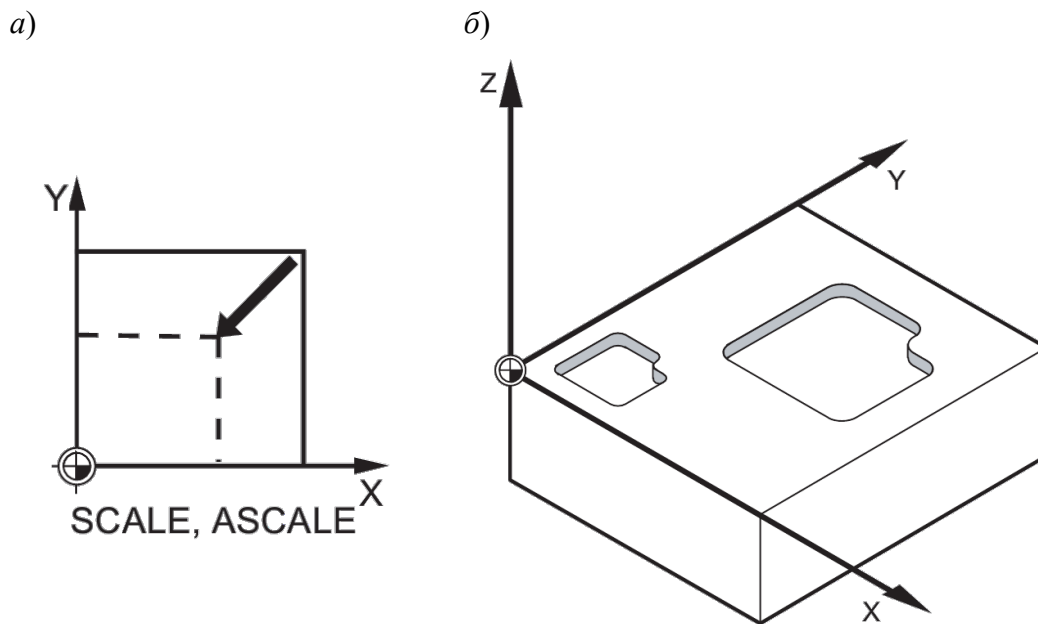
Синтаксис фрейма масштабирования (рисунок 2.7):

SCALE Xx Yy Zz;

ASCALE Xx Yy Zz,

где  $x, y, z$  – коэффициенты масштабирования для соответствующей оси.

При использовании *ASCALE* – последний действующий коэффициент масштабирования умножается на новый.



*a* – масштабирование в плоскости; *b* – пример применения фрейма масштабирования

Рисунок 2.7 – Фрейм масштабирования

*Программируемое отражение.*

Синтаксис фрейма программируемого отражения (рисунок 2.8):

MIRROR Xx Yy Zz;

AMIRROR Xx Yy Zz,

где  $X, Y, Z$  – геометрическая ось, направление которой должно быть изменено;  $x, y, z$  – значения, выбираются свободно, например  $X0$  или  $Y0$  или  $Z0$ .

Команда *MIRROR* сбрасывает все фрейм-компоненты установленного до этого программируемого фрейма.

*Отключение фрейма.*

Стирание программируемых фреймов осуществляется через указание компонента *TRANS, ROT, SCALE, MIRROR* без указания оси.

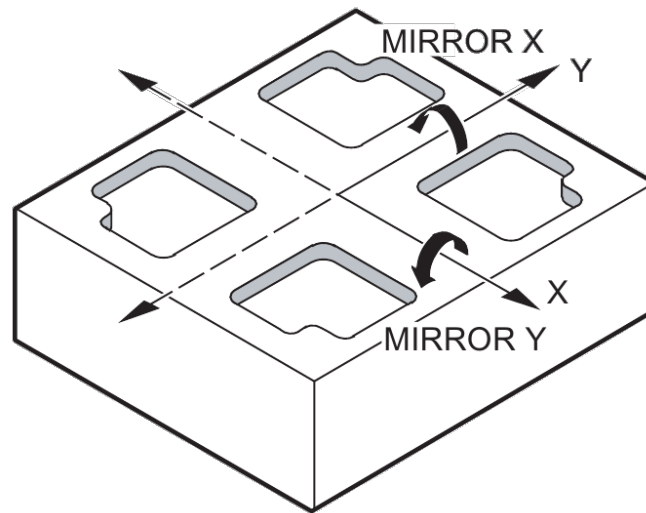


Рисунок 2.8 – Пример программируемого отражения по оси X и Y

#### 2.4 Пример использования трансформации системы координат

В качестве примера рассмотрим обработку группы отверстий расположенных внутри кармана (рисунок 2.9), программы обработки приведены в таблице 2.4. В связи с тем, что положение центров отверстий задано размерами от левого нижнего угла кармана, с целью упрощения расчета координат выполним следующие действия по трансформации системы координат:

- а) переместим ноль детали в левый нижний угол кармана;
- б) повернем систему координат вокруг оси Z на  $20^\circ$  так, чтобы ось X совпала с нижней гранью кармана.

И в таком случае система координат совпадет с измерительной базой. Координаты отверстий легко определятся из чертежа (см. рисунок 2.9).

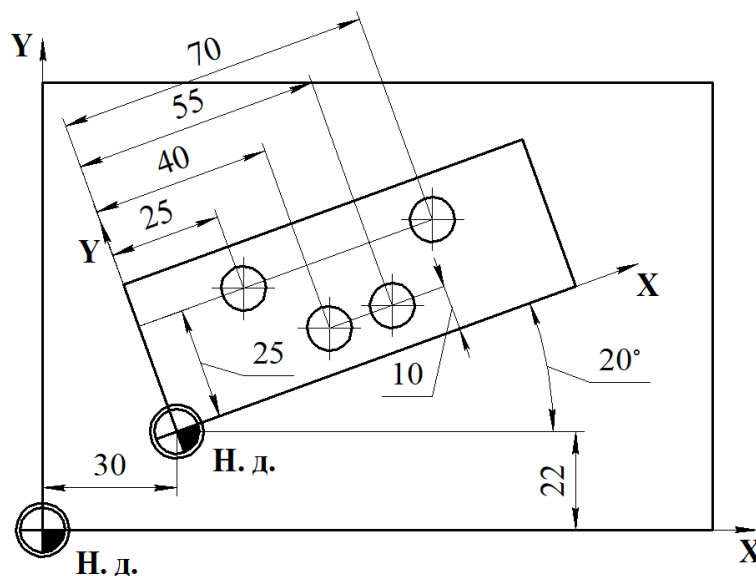


Рисунок 2.9 – Эскиз детали

Таблица 2.4 – Пример обработки группы отверстий

Программа		Комментарий
NC-201	SINUMERIK	
T1.1 M6	T1 D1 M6	Установить на обработку первый инструмент
S2000 F200	S2000 F200	Установить режимы резания
(UOT,0,X30,Y22)	TRANS X30 Y22	Переместить ноль системы координат в левый нижний угол кармана
(URT,20)	AROT RPL=20	Развернуть систему координат на 20 град
G0 X25 Y25	G0 X25 Y25	
G81 R-7 Z-25 M3	MCALL CYCLE81 (-7, -10, 3, -25,)	Вызываем цикл сверления с параметрами
X25 Y25	X25 Y25	Сверлить четыре отверстия
X40 Y10	X40 Y10	
X55	X55	
X70 Y25	X70 Y25	
G80	MCALL	Завершить цикл сверления
Z20	Z20	Поднять сверло на безопасное расстояние
(URT,0)	TRANS	Отменить трансформацию системы координат
(UAO,0)		
G0 X0 Y200 Z100	G0 X0 Y200 Z100	Вернуть инструмент в исходную точку
M30	M30	Конец программы

### ***2.5 Порядок выполнения практического занятия***

- 1 Изучить особенности трансформации систем координат.
- 2 Получить у преподавателя чертеж детали, разработать эскиз обработки.
- 3 Разработать управляющую программу для СЧПУ NC-201 и SINUMERIK.
- 4 Составить отчет и предоставить его преподавателю для проверки.

### ***2.6 Типовое содержание отчета по практическому занятию***

- 1 Цель практического занятия.
- 2 Используемое оборудование и инструмент.
- 3 Эскиз обработки детали.
- 4 Текст управляющей программы для СЧПУ NC-201 и SINUMERIK.
- 5 Ответы на контрольные вопросы.
- 6 Выводы.

### ***Контрольные вопросы***

- 1 Какие виды трансформаций Вы знаете?
- 2 Поясните отличия между аддитивными и замещающими операторами?
- 3 Задачи, для которых следует применять трансформацию системы координат?

### 3 Практическое занятие № 3. Программирование с использованием переменных и подпрограмм

**Цель занятия:** приобретение практических навыков программирования обработки деталей с использованием переменных и подпрограмм.

#### 3.1 Программирование с использованием переменных

Используя переменные, можно через параметры программировать геометрические и технологические данные цикла обработки. С параметрами допускаются математические и тригонометрические действия, а также вычисление выражений. Также с помощью переменных можно разрабатывать «умные» программы, в которых технология обработки или траектория перемещения инструмента меняется в зависимости от текущего состояния заготовки, станка и т. п. Грамотное применение переменных позволяет создавать собственные циклы обработки.

##### 3.1.1 Определение переменных.

*Особенности применения переменных для СЧПУ NC-201.*

Формат обозначения переменных в СЧПУ NC-201:  $E_e$ , где  $e$  – порядковый номер переменной.

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 таблице 3.1.

Таблица 3.1 – Описание параметров «E» для переменных различного формата

Формат	Параметры	Диапазон значений
BY (байт)	E0...E9	От 0 до 255
IN (целое)	E10...E19	От -32768 до +32768
LI (целое с двойной точностью)	E20...E24	От -2147483647 до +2147483647
RE (действительное)	E25...E29	7 знаков
LR (действительное с двойной точностью)	E30...E $n$	16 знаков

*Особенности применения переменных для СЧПУ семейства SINUMERIK.*

Все переменные в СЧПУ семейства SINUMERIK условно можно разделить на три группы:

1) системная переменная – переменные, задействованные программным обеспечением станка, они могут обрабатываться в пользовательской программе (запись, чтение);

2) пользовательская переменная – переменная, имя и тип которой определяются пользователем;

3) R-параметр – операционная переменная, для которой предусмотрен адрес R с последующим номером.

По адресу «R» находятся переменные типа REAL без дополнительного описания: **Re**, где *e* – порядковый номер переменной, стандартно 100, максимально 1000 (зависит от версии программного обеспечения станка).

Важной особенностью R-параметров является их глобальный характер, т. е. их можно использовать внутри не только одной программы, но и передавать между программами, также значения R-параметров сохраняются после включения (выключения) станка.

Переменные получают значения в кадрах назначения. Формат кадра назначения:

СЧПУ: NC-201  
Формат: **Ee=** <выражение>  
*Пример: E10=5*

СЧПУ: SINUMERIK  
Формат: **Re=** <выражение>  
*Пример: R10=5*

Здесь <выражение> может быть цифровой величиной или математическим выражением, результат которого будет запомнен под параметром с индексом «e»; «выражение» – это математическое выражение, составленное из арифметических операторов, функций и операндов (параметры, числовые константы).

В дополнение к R-параметрам в СЧПУ семейства SINUMERIK можно определять пользовательские переменные – переменные, обозначение которых и тип данных устанавливает пользователь.

Для обозначения пользовательской переменной используются идентификатор – буквенно-цифровая аббревиатура. Например: «USERVAR5».

В общем случае идентификаторы могут быть использованы также для обозначения подпрограмм, ключевых слов, меток.

Несмотря на то, что начертание идентификатора устанавливает пользователь, существуют определенные ограничения:

- длина – до 32 знаков;
- в идентификаторе допускаются: буквы, знаки подчеркивания, цифры;
- первые два знака обязательно буква или знак подчеркивания;
- идентификаторы не должны повторяться.

Пользовательскую переменную необходимо объявить в начале программы перед использованием с помощью функции

**DEF** <тип> <идентификатор>=<значение>;

*Пример: DEF INT USERVAR5=7,*

где <тип> – тип переменной; <идентификатор> – идентификатор переменной; <значение> – начальное значение переменной.



Если переменные определены в главной программе – действуют и в подпрограмме. Если при определении не присвоено значение, то система присваивает ей ноль.

Пользовательские переменные могут содержать следующие типы данных (таблица 3.2).

Таблица 3.2 – Типы данных для переменных

Обозначение	Описание	Диапазон значений
INT	Целые величины	-2147483648 ... +21474836
REAL	Действительное число	$\pm(\sim 2,2 \cdot 10^{-308} \dots \sim 1,8 \cdot 10^{+308})$
BOOL	Значения истинности: TRUE (1) и FALSE (0)	1, 0
CHAR	Символ ASCII	Код 0...255
STRING	Цепочка символов	Максимально 200 символов
AXIS	Идентификатор оси/шпинделя	
FRAME	Геометрические данные для смещения, вращения, масштабирования, отражения	

### 3.1.2 Вычисления в управляющей программе.

В таблице 3.3 приведены арифметические операции, допустимые с переменными.

Таблица 3.3 – Математические операторы

NC-201	SINUMERIK	Содержание операции
+	+	Сложение
-	-	Вычитание
*	*	Умножение
/	/	Деление
<b>SIN(A)</b>	<b>SIN(A)</b>	Синус <i>A</i>
<b>COS(A)</b>	<b>COS(A)</b>	Косинус <i>A</i>
<b>TAN(A)</b>	<b>TAN(A)</b>	Тангенс <i>A</i>
<b>ARS(A)</b>	<b>ASIN(A)</b>	Арксинус <i>A</i>
<b>ARC(A)</b>	<b>ACOS(A)</b>	Арккосинус <i>A</i>
<b>ART(A)</b>	<b>ATAN2(A)</b>	Арктангенс <i>A</i>
<b>SQR(A)</b>	<b>SQRT(A)</b>	Квадратный корень <i>A</i>
<b>ABS(A)</b>	<b>ABS(A)</b>	Абсолютное значение <i>A</i>
<b>INT(A)</b>	<b>TRUNC(A)</b>	Целое число <i>A</i>
<b>NEG(A)</b>	-	Инвертирует знак <i>A</i>
<b>MOD(A,B)</b>	<b>A MOD B</b>	Вычисляет остаток отношения между <i>A</i> и <i>B</i>
-	<b>POT(A)</b>	2-я степень (квадрат)
-	<b>ROUND(A)</b>	Округление до целого
-	<b>LN(A)</b>	Натуральный логарифм
-	<b>EXP(A)</b>	Роказательная функция

### 3.2 Программирование с использованием подпрограмм

#### 3.2.1 Повторение части программы.

При помощи операторов **REPEAT** часть программы, которая следует за меткой «стартовая метка перехода» и которая заканчивается меткой «конечная метка перехода», повторяется определенное количество раз.

Формат: **REPEAT** «стартовая метка перехода» «конечная метка перехода»  $P = n$ ,  
где  $n$  – число повторений.

#### 3.2.2 Использование подпрограмм.

Указав имя программы, можно вызвать и выполнить программу (подпрограмму), находящуюся в памяти. Под подпрограммой понимают последовательность кадров, которые определяют цикл обработки. Подпрограмма может быть вызвана из основной программы.

#### *Пример*

N10 FALE,

где FALE – название подпрограммы для вызова.

### 3.3 Порядок выполнения практического занятия

1 Изучить состав и назначение адресов при разработке управляющей программы с использованием переменных и подпрограмм.

2 Для детали, изображенной на рисунке 3.1, из таблицы 3.4 по заданному варианту выбрать необходимые размеры.

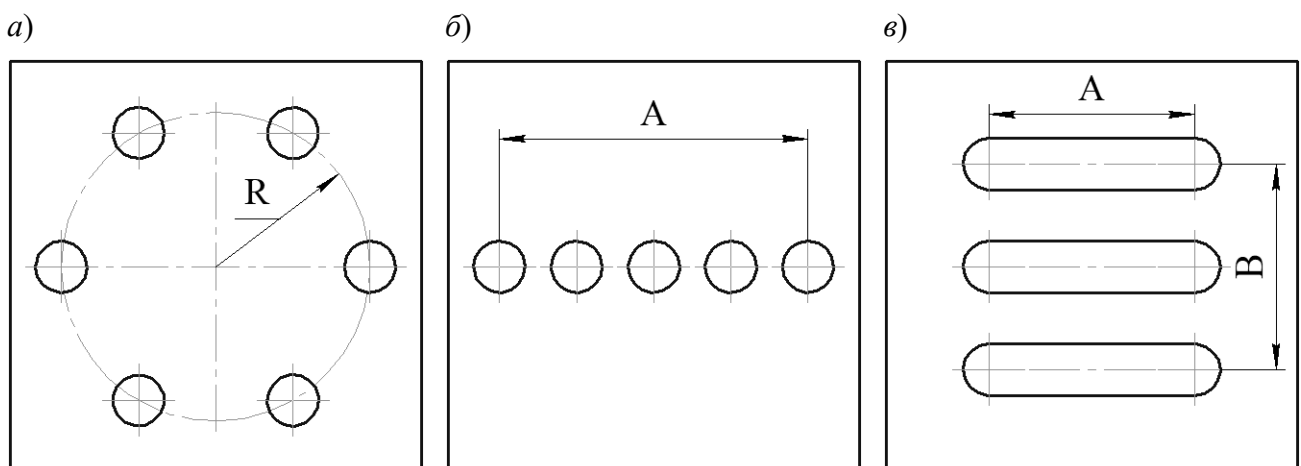


Рисунок 3.1 – Эскизы деталей

Таблица 3.4 – Варианты заданий

Вариант	Деталь по рисунку 3.1	Количество элементов $n$	R	A	B
1	<i>a</i>	3	60	–	–
2	<i>б</i>	6	–	150	–
3	<i>в</i>	6	–	50	90
4	<i>a</i>	9	50	–	–
5	<i>б</i>	7	–	105	–
6	<i>в</i>	8	–	45	160
7	<i>a</i>	5	50	–	–
8	<i>б</i>	7	–	140	–
9	<i>в</i>	9	–	25	45
10	<i>a</i>	6	50	–	–
11	<i>б</i>	3	–	60	–
12	<i>в</i>	3	–	20	15
13	<i>a</i>	6	70	–	–
14	<i>б</i>	3	–	60	–
15	<i>в</i>	6	–	30	90
16	<i>a</i>	9	70	–	–
17	<i>б</i>	9	–	225	–
18	<i>в</i>	6	–	40	30
19	<i>a</i>	10	40	–	–
20	<i>б</i>	4	–	40	–

3 С учетом особенностей станков с ЧПУ установить состав переходов при обработке детали, определить количественные и качественные параметры необходимого инструмента.

4 Определить способ установки детали на станке, установить ноль детали, положение исходной точки. Построить траектории перемещений каждого инструмента.

5 Определить координаты опорных точек, заполнить таблицу.

6 Определить режимы резания для каждого инструмента, определить необходимый диапазон вращения шпинделя.

7 Разработать текст управляющей программы по обработке детали.

8 Составить отчет по лабораторной работе и представить его преподавателю для проверки и защиты.

### ***3.4 Типовое содержание отчета по практическому занятию***

1 Цель практического занятия.

2 Чертеж детали с указанием способа базирования, положения нуля детали и направления осей (рисунок 3.2).

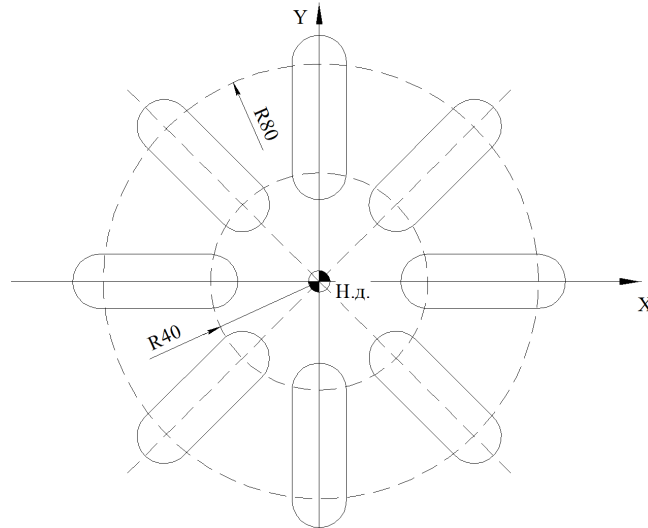


Рисунок 3.2 – Эскиз обработки

- 3 Эскизы обработки и траектории перемещения инструмента.
- 4 Таблица опорных точек.
- 5 Текст управляющей программы (таблица 3.5).
- 6 Ответы на контрольные вопросы и выводы.

Таблица 3.5 – Примеры управляющих программ

С использованием повторов внутри программы	С использованием цикла
T5 D5 M6	DEF INT VARI VARA
G17 S1500 F150 M3	T5 D5 M6
R10=0	G17 S1500 F150 M3
BEGIN1: R11=R10*45	FOR VARI=0 TO 7
ROT RPL=R11	VARA=VARI*45
G0 X40 Y0	ROT RPL=VARA
Z2	G0 X40 Y0
G1 Z-6 X80	Z2
Z-12 X40	G1 Z-6 X80
X80	Z-12 X40
G0 Z20	X80
R10=R10+1	G0 Z20
REPEAT BEGIN1 P=7	ENDFOR
ROT	ROT
G0 X0 Y120 Z50	G0 X0 Y120 Z50
M30	M30

### ***Контрольные вопросы***

- 1 Какие типы переменных Вы знаете?
- 2 Какие операции можно выполнять с переменными?
- 3 Назначение подпрограмм?

## 4 Практическое занятие № 4. Программирование токарной многоцелевой обработки

**Цель занятия:** приобретение практических навыков проектирования и выполнения операций обработки на токарных многоцелевых станках.

### 4.1 Общие сведения о программировании обработки на токарных многоцелевых станках

Для обработки профилей на плоскости или на цилиндре, при помощи оси вращения и линейной оси, вводится понятие виртуальных осей.

Для увеличения жесткости при позиционной обработке станок оснащен зажимами поворотных осей, которые управляются M-кодами: M10/M12 – включить зажимы для осей A и C; M11/M13 – выключить зажимы.

TRANSMIT – трансформация торца детали (рисунок 4.1):

- торцовая обработка в патроне (сверление, фрезерование контура);
- движения по траектории выполняются линейными осями X, Z и круговой осью C.

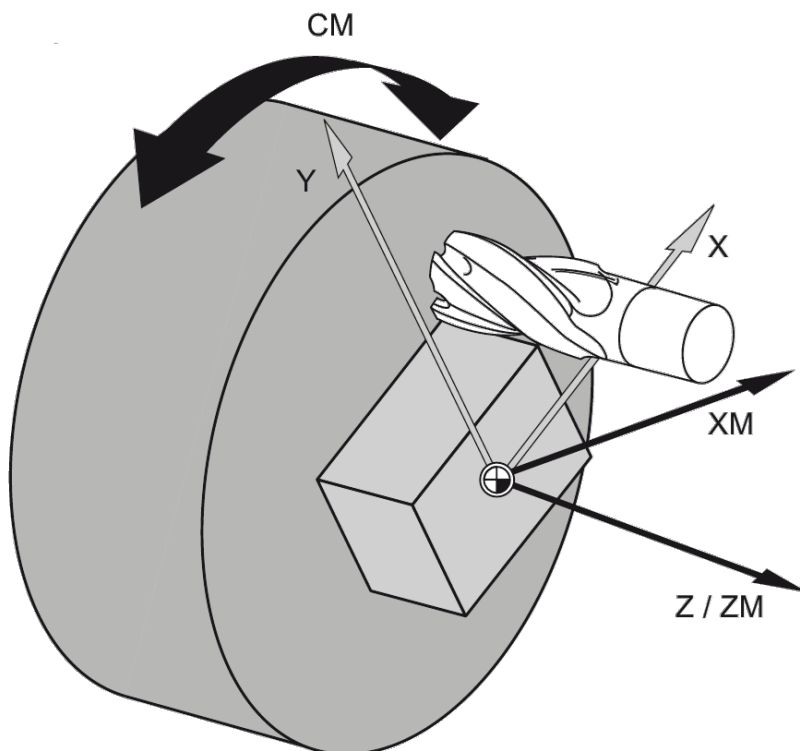


Рисунок 4.1 – Трансформация торца детали

TRACYL(d) – криволинейная трансформация боковой поверхности цилиндра, где d – диаметр обрабатываемого цилиндра (рисунок 4.2).

TRAFOOF – выключает активную трансформацию TRANSMIT, TRACYL.

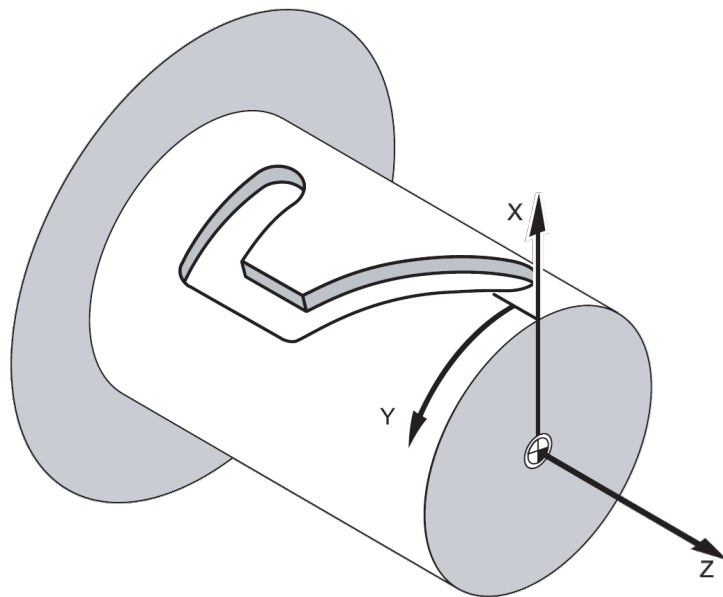


Рисунок 4.2 – Криволинейная трансформация боковой поверхности цилиндра

Пример программирования обработки на многоцелевых токарных станках представлен на рисунке 4.3.

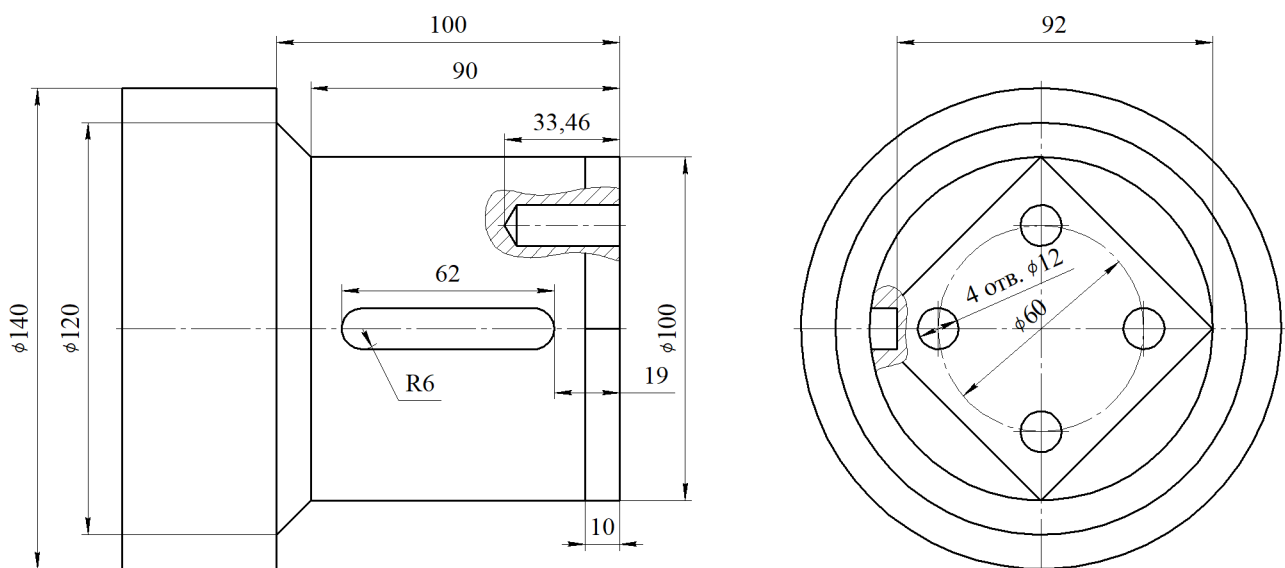


Рисунок 4.3 – Эскиз детали

Управляющая программа обработки к рисунку 4.3.

```

KON_1.SPF
G0 X100 Z0
G1 Z-90
X120 Z100
X140
M17

```

T1 D1  
 S500 F1 M3  
 G0 X144 Z0  
 G1 X-2  
 Z2  
 G0 X144  
**CYCLE95** ("KON\_1", 7, , , 0.6, 0.5, 0.2, 0.1, 9, , , )  
 T3 D1  
 S2=2000 M2=3 M8  
 G17 G0 X102 Z-10  
 SPCON  
 SPOS=0  
**TRANSMIT**  
 G450 G41 G0 X-90 Y0  
 G3 X-50 CR=20  
 G1 X0 Y50  
 X50 Y0  
 X0 Y-50  
 X-50 Y0  
 G3 X-78.248 CR=20  
 G40 G0 X-100 Z50  
 T4 D1  
 S2=1500 F0.5 M2=3  
 G0 X-30 Y0  
**MCALL CYCLE81**(2, 0, 2, -2,)  
 X-30 Y0  
 X0 Y30  
 X30 Y0  
 X0 Y-30  
**MCALL**  
 G0 X-100 Y50  
 T5 D1  
 S2=1200 F0.5 M2=3  
 G0 X-30 Y0  
**MCALL CYCLE81**(2, 0, 2, -33,46,)  
 X-30 Y0  
 X0 Y30  
 X30 Y0  
 X0 Y-30  
**MCALL**  
 G0 X-100 Z50  
 T6 D1  
 S2=1000 F0.1 M2=3  
 SPOS=0

**TRACYL(100)**

G0 X102 Y0 Z-25

G1 X98

Z-50

X96

Z-25

X94

Z-50

X92

Z-25

Z-50

X102

**(TRAFOOF)**

SPCOF

G0X200Z50

M30

***4.2 Порядок выполнения практического занятия***

- 1 Изучить особенности обработки профилей на токарных многоцелевых станках с ЧПУ.
- 2 Получить у преподавателя чертеж детали, разработать эскиз обработки.
- 3 Рассчитать геометрические параметры траекторий.
- 4 Построить траектории перемещения инструментов.
- 5 Разработать управляющую программу для выполнения обработки профилей.
- 6 Составить отчет и предоставить его преподавателю для проверки.

***4.3 Типовое содержание отчета по практическому занятию***

- 1 Цель практического занятия.
- 2 Используемое оборудование и инструмент.
- 3 Эскиз обработки детали.
- 4 Текст управляющей программы.
- 5 Ответы на контрольные вопросы.
- 6 Выводы.

***Контрольные вопросы***

- 1 Технологические задачи, решаемые токарной многоцелевой обработкой.
- 2 Какие трансформации системы координат применяются при многоцелевой токарной обработке?
- 3 Поясните понятие «виртуальные оси».



## 5 Практическое занятие № 5. Программирование контроля

**Цель занятия:** приобретение практических навыков программирования контроля на станках с ЧПУ.

### 5.1 Общие сведения о программировании контроля для СЧПУ семейства NC

*Измерение координат точки.*

Функция G72 измеряет при помощи щупа координаты точки в пространстве прямолинейным движением и заносит их в память системы как параметры «E», определённые в цикле (запоминание начинается с запрограммированного параметра). Измерение выполняется с корректировкой радиуса щупа.

Формат: G72 ось [ось] [ось] En,

где ось [ось] [ось] – контролируемые оси (до трех осей в кадре); En – определяет параметр, от которого необходимо начать запоминание размеров, вычисленных щупом.

**Пример**

G72 X100 Y50 E32.

В E32 и E33 запоминаются вычисленные величины для X и Y соответственно.

*Измерение параметров отверстия.*

Функция G73 измеряет при помощи щупа параметры отверстия в данной плоскости интерполяции и заносит их в память системы как параметры «E», определённые в цикле измерения (запоминание начинается с запрограммированного параметра). Щуп должен быть размещен в центре отверстия. Полученными параметрами являются координаты центра и радиус отверстия. Измерение осуществляется с корректировкой радиуса щупа.

Формат: G73 r En,

где r определяет теоретический радиус отверстия; En определяет параметр, от которого начинается запоминание параметров отверстия.

Результаты измерения окружности: координаты центра  $E_n=X$ ,  $E_{n+1}=Y$  и радиус окружности  $E_{n+2}=R$ .

**Пример**

G73 r100 E55.

В E55, E56 и E57 заносятся абсцисса, ордината и радиус окружности соответственно.

*Определение параметров измерения DPT.*

Формат: (DPT, Qa, Qs, Vm),

где Qa – размер подхода, мм; Qs – размер безопасности, мм; Vm – скорость

измерения, мм/мин.

*Управление коррекцией инструмента RQU.*

Формат: (RQU, N.инстр., N.корр., Z..., K...),

где N.инстр. – номер инструмента (E10-E19); N.корр. – номер корректора; Z – инкремент (изменение) длины, который прибавляется к корректору оси Z (E30-En); K – инкремент диаметра инструмента, который прибавляется к корректору диаметра инструмента K (E30-En).

**Пример**

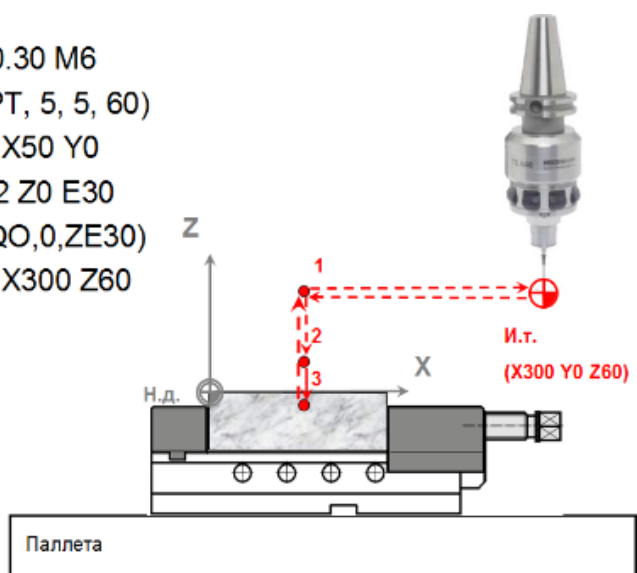
(RQU, 10, 10, ZE40, KE41).

Кадр модифицирует инструмент 10, корректор 10; инкремент для оси Z содержится в E40; инкремент для диаметра K содержится в E41.

Примеры программирования циклов контроля представлены на рисунке 5.1.

а)

N1 T30.30 M6  
N2 (DPT, 5, 5, 60)  
N3 G0 X50 Y0  
N4 G72 Z0 E30  
N5 (RQU,0,ZE30)  
N6 G0 X300 Z60  
N7 ...  
M30



б)

N1 T30.30 M6  
N2 (DPT, 5, 5, 60)  
N3 G0 X80 Y60 Z20  
N4 Z-13  
N5 G73 r50 E35  
N6 E35=E35-60  
N7 E36=E36-80  
N8 (RQU,0, XE35, YE36)  
N9 G0 Z20  
N10 X300 Y200 Z60  
N11 ...  
N100 M30

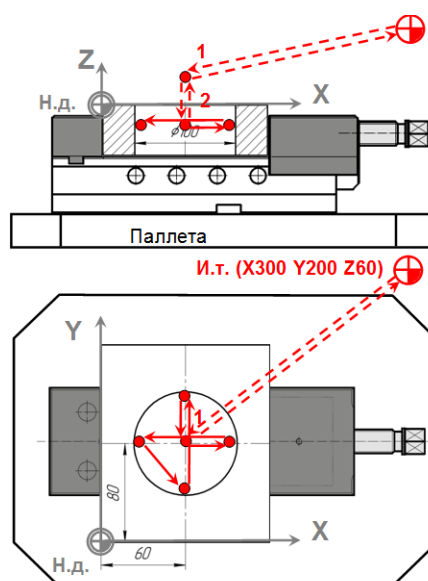


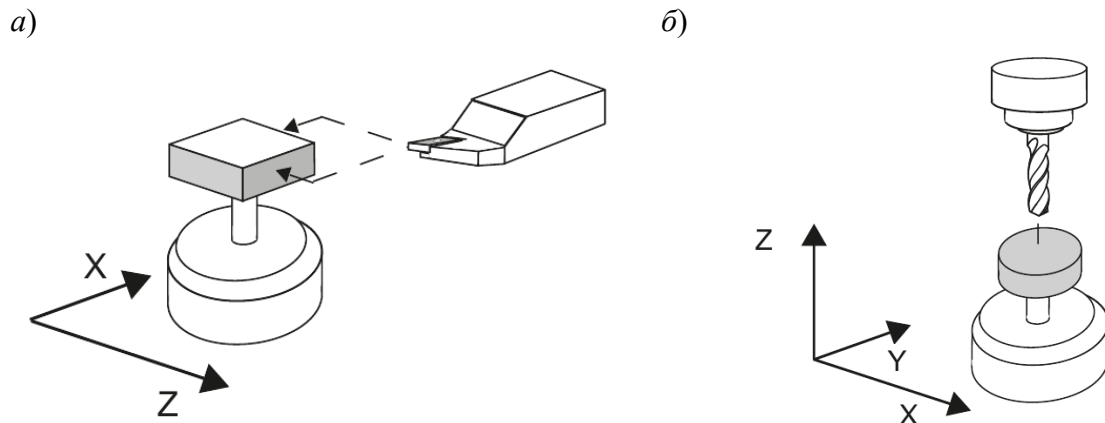
Рисунок 5.1 – Циклы контроля расположения точки (а) и точности отверстия (б)

## 5.2 Измерительные циклы для СЧПУ семейства SINUMERIK

Измерительные циклы – это общие подпрограммы для решения определенных задач измерения, которые через параметры могут быть адаптированы к конкретной проблеме.

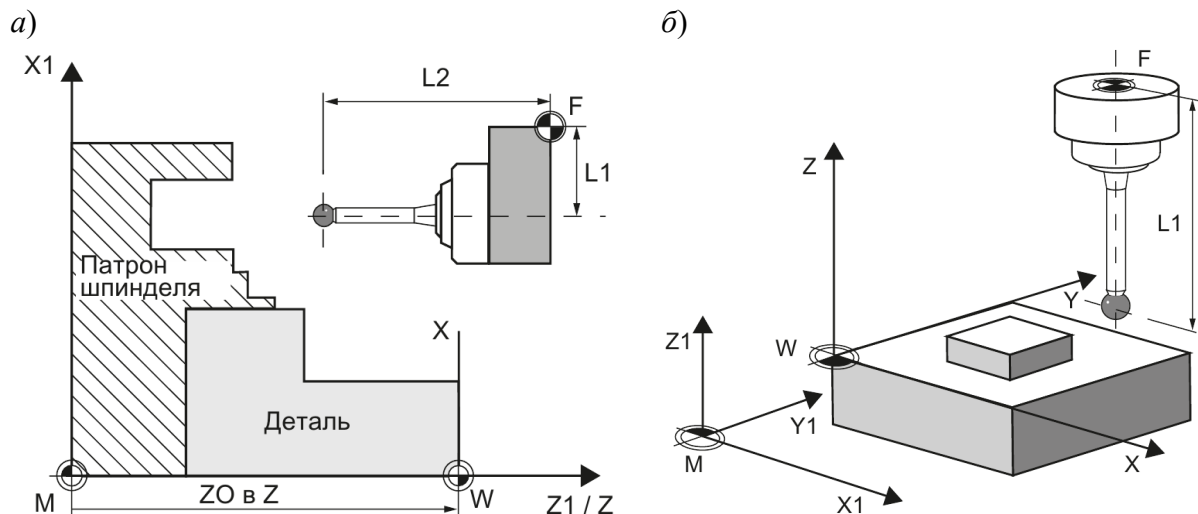
Базовыми вариантами измерения являются:

- измерение инструмента (рисунок 5.2);
- измерение детали (рисунок 5.3).



*a* – для токарного инструмента; *б* – для осевого инструмента

Рисунок 5.2 – Измерение инструмента



*a* – для токарного станка; *б* – для фрезерного станка

Рисунок 5.3 – Измерение детали

*Калибровка измерительного щупа инструмента (CYCLE971).*

С помощью этого измерительного цикла можно выполнить калибровку измерительного щупа инструмента и измерение длины инструмента и/или радиуса инструмента.

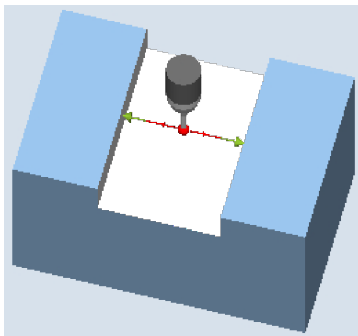
*Калибровка измерительного щупа детали (CYCLE976).*

С помощью этого измерительного цикла можно выполнить калибровку измерительного щупа детали.

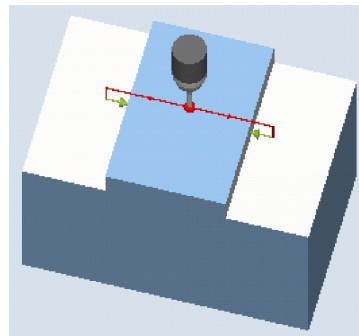
*Обмер пазов, перемычек, карманов, отверстий, цапф (CYCLE977).*

С помощью этого измерительного цикла можно определить центр в плоскости, а также ширину или диаметр. Возможные варианты измерения представлены на рисунке 5.4.

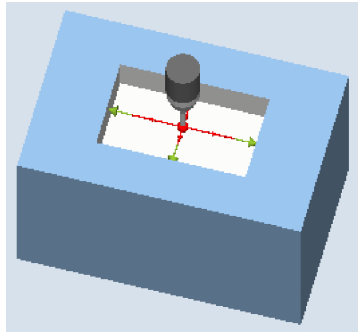
а)



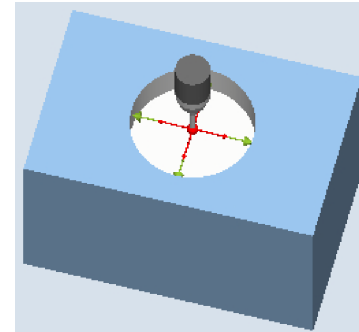
б)



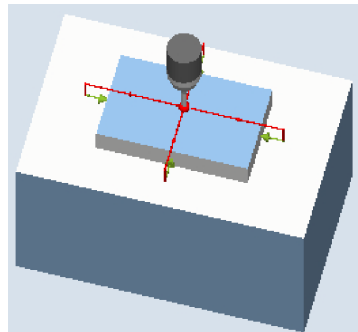
в)



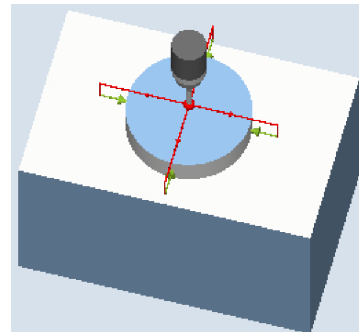
г)



д)



е)

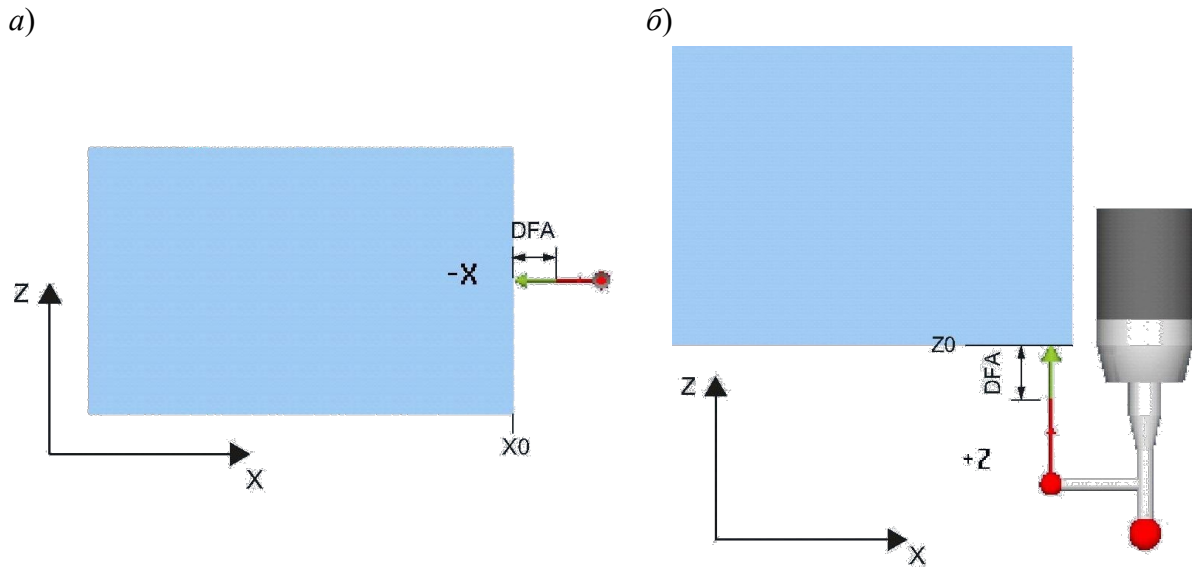


а – паз; б – перемычка; в – прямоугольный карман; г – отверстие; д – прямоугольная цапфа; е – круговая цапфа

Рисунок 5.4 – Измерительный цикл CYCLE977

*Обмер по первой точке или определение смещения нуля на плоскости (CYCLE978).*

С помощью этого измерительного цикла можно измерить кромку в системе координат детали. Возможные варианты измерения представлены на рисунке 5.5.



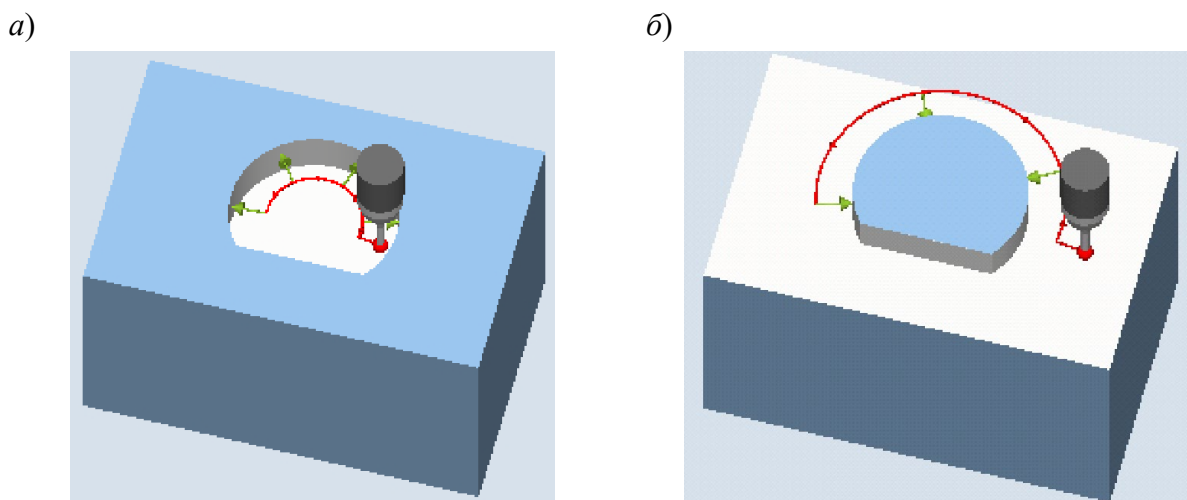
*a* – направление измерения (-X); *б* – направление измерения (+Z)

Рисунок 5.5 – Измерительный цикл CYCLE978

*Обмер кругового сегмента (CYCLE979).*

Возможные варианты представлены на рисунке 5.6:

- внутренний круговой сегмент;
- внешний круговой сегмент.



*a* – внутренний круговой сегмент; *б* – внешний круговой сегмент

Рисунок 5.6 – Цикл CYCLE979 определяет диаметр и центр сегмента в плоскости

*Выверка кромки (только определение смещения нуля) (CYCLE998).*

С помощью этого измерительного цикла можно определить угловое положение поверхности (плоскости) относительно рабочей плоскости и угол кромок в системе координат детали, как показано на рисунке 5.7. Измерительный цикл CYCLE998 может измерить максимальный угол в  $-45^{\circ} \dots +45^{\circ}$ .

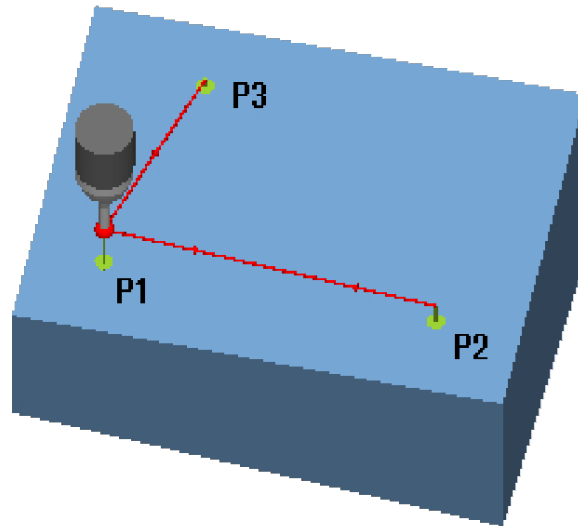


Рисунок 5.7 – Измерительный цикл CYCLE998

### ***5.3 Порядок выполнения практического занятия***

- 1 Изучить особенности программирования контроля на станках с ЧПУ.
- 2 Получить у преподавателя чертеж детали, разработать схему контроля.
- 3 Построить траектории перемещения измерительных щупов.
- 4 Разработать управляющую программу для контроля элементов детали.
- 5 Составить отчет и предоставить его преподавателю для проверки.

### ***5.4 Типовое содержание отчета по практическому занятию***

- 1 Цель практического занятия.
- 2 Используемое оборудование и инструмент.
- 3 Схема контроля и траектории перемещения измерительных щупов.
- 4 Текст управляющей программы.
- 5 Ответы на контрольные вопросы.
- 6 Выводы.

### ***Контрольные вопросы***

- 1 Как работает измерительный щуп?
- 2 Перечислите типовые измерительные циклы.
- 3 Какие задачи способен решить контроль на станках с ЧПУ?

## 6 Практическое занятие № 6. Нормирование операций, выполняемых на станках с ЧПУ

**Цель занятия:** приобретение навыков определения затрат времени на операции, выполняемые на станках с ЧПУ.

### 6.1 Расчёт норм времени для операций, выполняемых на станках с ЧПУ

Норма времени на выполнение операций на станках с ЧПУ при работе на одном станке состоит из нормы подготовительно-заключительного времени  $t_{п.з.}$  и нормы штучного времени  $t_{шт.}$ :

$$t_{шт.} = (t_{ца} + t_{с} \cdot k_{тс}) \left( 1 + \frac{a_{тех} + a_{орг} + a_{отд}}{100} \right), \quad (6.1)$$

где  $t_{ца}$  – время цикла автоматической работы станка по программе, мин;

$k_{тс}$  – поправочный коэффициент на время выполнения ручной вспомогательной работы в зависимости от партии обрабатываемых деталей;

$a_{тех}$ ,  $a_{орг}$ ,  $a_{отд}$  – время на техническое и организационное обслуживание рабочего места, на отдых и личные потребности при одностаночном обслуживании; определяется как процент от оперативного времени.

$$t_{ца} = t_o + t_{мс}, \quad (6.2)$$

где  $t_o$  – основное (технологическое) время на обработку одной детали, мин,

$$t_o = \sum \frac{L_i}{S_{mi}}; \quad (6.3)$$

$L_i$  – длина пути, проходимого инструментом или деталью в направлении подачи при обработке  $i$ -го технологического участка (с учетом врезания и перебега), мм;

$S_{mi}$  – минутная подача на данном технологическом участке, мм/мин;

$t_{мс}$  – машинно-вспомогательное время по программе (на подвод детали или инструмента от исходных точек в зоны обработки и отвод, установку инструмента на размер, смену инструмента, изменение величины и направления подачи, время технологических пауз (остановок) и т. п.), мин.

Вспомогательное время, связанное с операцией, не вошедшее во время цикла автоматической работы станка по программе:

$$t_{с} = t_{с.у} + t_{с.он} + t_{с.изм}, \quad (6.4)$$

где  $t_{e,y}$  – время на установку заготовки и снятие детали вручную или подъемником, мин;

$t_{e,on}$  – вспомогательное время, связанное с операцией (не вошедшее в управляющую программу), мин;

$t_{e,uzm}$  – вспомогательное неперекрываемое время на измерения, мин.

Нормативное время на установку заготовки и снятие детали предусматривает выполнение следующей работы: взять и установить заготовку, выверить и закрепить; включить и выключить станок; открепить, снять деталь и уложить в тару; очистить приспособление от стружки, протереть базовые поверхности салфеткой. При установке в специальных приспособлениях вспомогательное время определяют как сумму времени на установку заготовки и снятие одной детали; на установку заготовки и снятие каждой последующей детали свыше одной в многоместных приспособлениях; на закрепление заготовки с учетом количества зажимов; на очистку приспособления от стружки, на протирку базовых поверхностей салфеткой.

Вспомогательное время, связанное с операцией, подразделяется на:

– вспомогательное время, связанное с операцией, не вошедшее во время цикла автоматической работы станка по программе и предусматривающее выполнение следующей работы: установить заданное взаимное положение заготовки и инструмента по координатам X, Y, Z и в случае необходимости произвести поднастройку; проверить приход заготовки или инструмента в заданную точку после обработки; установить щиток от забрызгивания эмульсией и снять;

– машинно-вспомогательное время, связанное с переходом, включенное в программу и относящееся к автоматической вспомогательной работе станка, предусматривающее подвод заготовки или инструмента от исходной точки в зону обработки и отвод; установку инструмента на размер обработки; автоматическую смену инструмента; включение и выключение подачи; холостые ходы при переходе от обработки одних поверхностей к другим; технологические паузы, необходимые при резком изменении направления подачи, проверке размеров, для осмотра инструмента и переустановки или перезакрепления заготовки.

Необходимые размеры деталей, обрабатываемых на станках с числовым программным управлением, обеспечиваются конструкцией станка или режущего инструмента и точностью их настройки. В связи с этим время на контрольные измерения (после окончания работы по программе) должно включаться в норму штучного времени только в том случае, если это предусмотрено технологическим процессом и с учетом необходимой периодичности таких измерений в процессе работы, и только в тех случаях, если оно не может быть перекрыто временем цикла автоматической работы станка по программе.

Время на обслуживание рабочего места дано по типам и размерам оборудования с учетом одностаночного и многостаночного обслуживания в



процентах от оперативного времени. Техническое обслуживание рабочего места предусматривает выполнение следующих работ:

- смену инструмента (или блока с инструментом) вследствие его затупления; регулировку и подналадку станка в процессе работы (изменение величины коррекции инструмента);

- сметание и периодическую уборку стружки в процессе работы (кроме сметания стружки с базовых поверхностей установочных приспособлений, время на которое учтено во вспомогательном времени на установку заготовки и снятие детали).

Организационное обслуживание рабочего места включает работу по уходу за рабочим местом (основным и вспомогательным оборудованием, технологической и организационной оснасткой, тарой), относящуюся к рабочей смене в целом:

- осмотр и опробование оборудования в процессе работы;

- раскладку инструмента в начале и уборку его в конце смены (кроме многоцелевых станков);

- смазку и чистку станка в течение смены;

- получение инструктажа мастера, бригадира в течение смены;

- уборку станка и рабочего места в конце смены.

Время на отдых и личные потребности для условий обслуживания одним рабочим одного станка отдельно не выделяется и учтено во времени на обслуживание рабочего места.

Норма подготовительно-заключительного времени представляется как время на приемы подготовительно-заключительной работы на обработку партии одинаковых деталей независимо от партии и определяется по формуле

$$t_{n.з.} = t_{n.з.1} + t_{n.з.2} + t_{np.обр.} \quad (6.5)$$

где  $t_{n.з.1}$  – норма времени на организационную подготовку, мин;

$t_{n.з.2}$  – норма времени на наладку станка;

$t_{np.обр.}$  – норма времени на пробную обработку.

Время на организационную подготовку предусматривает:

- получение наряда, чертежа, технологической документации, программоносителя, режущего, вспомогательного и контрольно-измерительного инструмента, приспособлений, заготовок до начала и сдачу их после окончания обработки партии деталей на рабочем месте или в инструментальной кладовой;

- ознакомление с работой, чертежом, технологической документацией, осмотр заготовки;

- инструктаж мастера.

В состав работы по наладке станка, инструмента и приспособлений включаются приемы работы наладочного характера, зависящие от назначения станка и его конструктивных особенностей:

- установка и снятие крепежного приспособления;

- установка и снятие блока или отдельных режущих инструментов;
  - установка исходных режимов работы станка;
  - установка программносителя в считывающее устройство и снятие его;
- настройка нулевого положения и др.

Время на пробную обработку деталей на станках токарной (до 630 мм) и револьверной групп включает затраты времени на изготовление детали по программе (время цикла), вспомогательное время на выполнение дополнительных приемов, связанных с измерением детали, вычислением коррекции, введением величин коррекций в систему ЧПУ, и вспомогательное время на приемы управления станком и системой ЧПУ.

### ***6.2 Порядок выполнения практического занятия***

- 1 Изучить особенности нормирования операций, выполняемых на станках с ЧПУ.
- 2 Получить у преподавателя чертеж детали.
- 3 Выбрать оборудование, инструмент.
- 4 Установить состав переходов при обработке детали.
- 5 Определить способ установки детали на станке, установить нуль детали, положение исходной точки. Построить траектории перемещений каждого инструмента.
- 6 Определить координаты опорных точек, заполнить таблицу.
- 7 Определить режимы резания для каждого инструмента.
- 8 Заполнить таблицу затрат времени на работу по программе и определить составляющие штучно-калькуляционного времени. Определить штучно-калькуляционное время.
- 9 Составить отчет и предоставить его преподавателю для проверки.

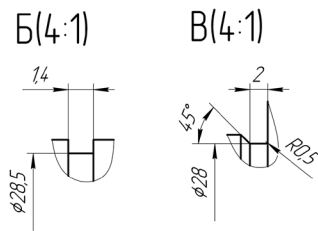
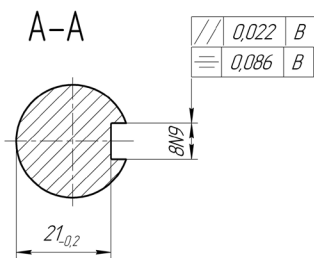
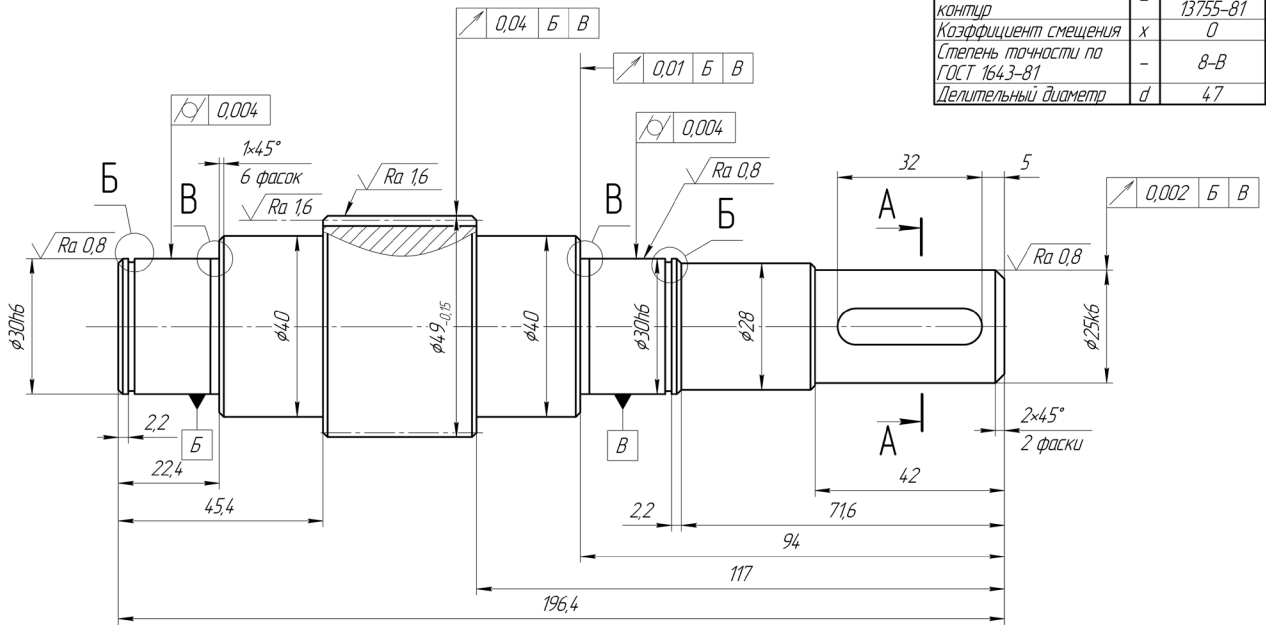
### ***6.3 Типовое содержание отчета по практическому занятию***

- 1 Цель практического занятия.
- 2 Чертеж обрабатываемой детали (представлен на рисунке 6.1).
- 3 Форма и размеры заготовки: прокат  $\text{Ø}50 \times 196,4$ .
- 4 Тип используемого оборудования: токарный станок с ЧПУ модели СК6140А.
- 5 Состав инструмента, закрепляемого в револьверной головке, с указанием точки привязки инструмента (рисунок 6.2).
- 6 На основе содержания операции устанавливается последовательность обработки. Для обработки левой стороны вала, представленного на рисунке 6.1, последовательность следующая (рисунок 6.3):
  - а) переход 1 – точить начерно  $\text{Ø}30$  мм,  $\text{Ø}40$  мм,  $\text{Ø}49$  мм;
  - б) переход 2 – точить начисто  $\text{Ø}30$  мм,  $\text{Ø}40$  мм,  $\text{Ø}49$  мм, точить фаски;
  - в) переход 3 – точить канавки под стопорные кольца, канавки под выход шлифовального круга.

Схема базирования: обработка в центрах – задний центр вращающийся, передний центр плавающий с упором в торец, вращение передается поводком.

$\sqrt{Ra\ 6,3\ (\checkmark)}$

Модель	m	1
Число зубьев	z	47
Нормальный исходный контур	-	ГОСТ 13755-81
Коэффициент смещения	x	0
Степень точности по ГОСТ 1643-81	-	8-B
Делительный диаметр	d	47



1. 230...260 HB
2. Общие допуски по ГОСТ 30893.1-02: H14, h14,  $\pm IT14/2$
3. Остальные технические требования по СТБ 1014-95

Рисунок 6.1 – Чертеж детали

T1 – черновой  
упорно-проходной

T3 – чистовой  
упорно-проходной

T5 – канавочный  
B = 1,4 мм

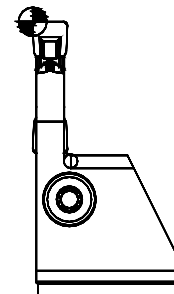
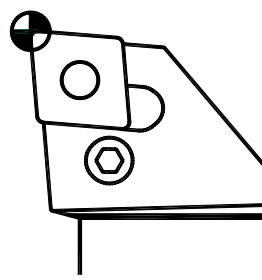
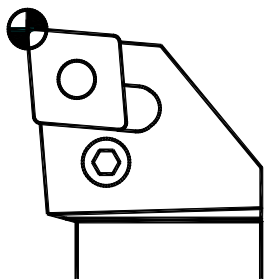
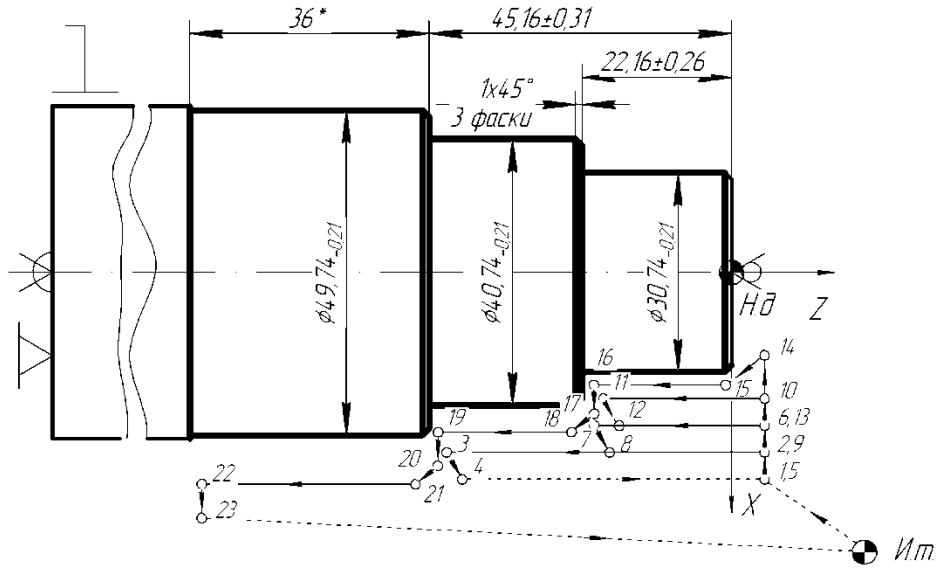
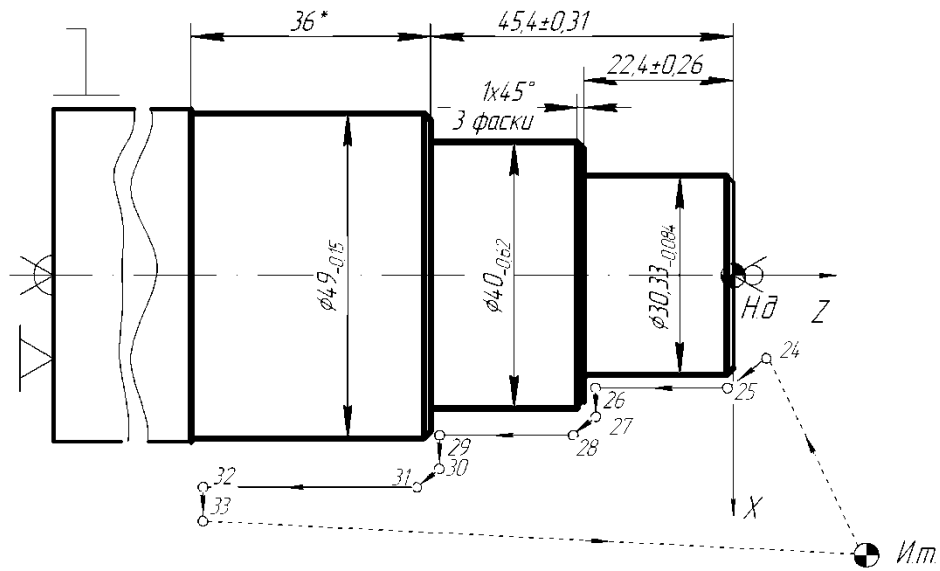


Рисунок 6.2 – Режущий инструмент

Переход 1



Переход 2



Переход 3

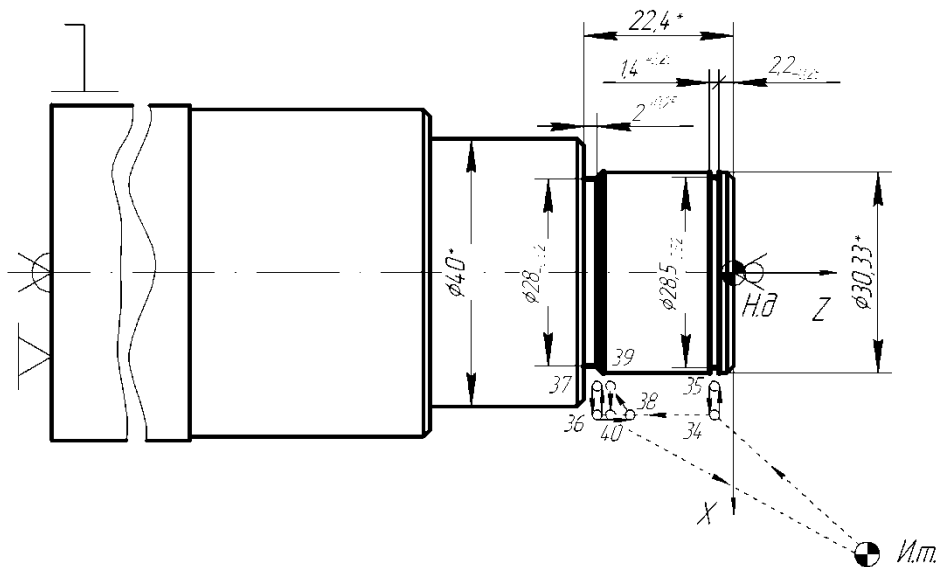


Рисунок 6.3 – Траектории перемещения инструмента при обработке вала

На основе траекторий перемещения инструмента при обработке вала определяются и заносятся в таблицу 6.1 координаты опорных точек.

Таблица 6.1 – Координаты опорных точек

Номер опорной точки	X	Z	Номер опорной точки	X	Z	Номер опорной точки	X	Z
И.т.	60	20	28	39,7	-23,4	35	28,24	-3,6
1	53	2	29	39,7	-45,4	36	32	-22,4
23	26	-83	30	46,92	-45,4	37	27,74	-22,4
24	28,29	1	31	48,92	-46,4	38	32	-19,8
25	30,29	-1	32	48,92	-82	39	27,74	-21,8
26	30,29	-22,4	33	53	-82	40	32	-21,8
27	37,7	-22,4	34	32	-3,6			

7 Расчет затрат времени для токарной операции (см. рисунок 6.3). Время  $t_{ца}$  определится как сумма затрат времени на элементарные перемещения. Расчеты сведены в таблицу 6.2, по паспорту станка время смены инструмента по программе – 0,5 с на позицию, скорость быстрых перемещений – 5000 мм/мин.

Таблица 6.2 – Затраты времени на работу по программе  $t_{ца}$

Элементарное перемещение	Длина перемещения, мм	Характер перемещения	Подача, мм/об	Частота вращения, мин <sup>-1</sup>	Подача, мм/мин	Время, мин
1	2	3	4	5	6	7
Переход 1						
T1	–	$t_{мс}$	–	–	–	0,017
ИТ→1	18,248	$t_{мс}$	–	–	5000	0,004
1→2	3,5	$t_{мс}$	–	–	5000	0,001
2→3	46,9	$t_o$	0,4	784,145	313,66	0,150
3→4	4,95	$t_{мс}$	–	–	5000	0,001
4→5	43,4	$t_{мс}$	–	–	5000	0,009
5→6	7	$t_{мс}$	–	–	5000	0,001
6→7	23,9	$t_o$	0,4	877,583	351,03	0,068
7→8	4,95	$t_{мс}$	–	–	5000	0,001
8→9	20,4	$t_{мс}$	–	–	5000	0,004
9→10	7	$t_{мс}$	–	–	5000	0,001
10→11	23,9	$t_o$	0,4	996,302	398,52	0,060
11→12	4,95	$t_{мс}$	–	–	5000	0,001
12→13	20,4	$t_{мс}$	–	–	5000	0,004
13→14	7	$t_{мс}$	–	–	5000	0,001
14→15	4,243	$t_o$	0,4	996,302	398,52	0,011
15→16	20,4	$t_o$	0,4	996,302	398,52	0,051
16→17	4	$t_o$	0,4	996,302	398,52	0,010
17→18	1,414	$t_o$	0,4	877,583	351,03	0,004
18→19	22	$t_o$	0,4	877,583	351,03	0,063

Окончание таблицы 6.2

1	2	3	4	5	6	7
19→20	3,5	$t_o$	0,4	784,145	313,66	0,011
20→21	1,414	$t_o$	0,4	784,145	313,66	0,005
21→22	35	$t_o$	0,4	784,145	313,66	0,112
22→23	2	$t_o$	0,4	784,145	313,66	0,006
23→ИТ	90,988	$t_{ме}$	–	–	5000	0,018
Переход 2						
Т3	–	$t_{ме}$	–	–	–	0,017
ИТ→24	24,759	$t_{ме}$	–	–	5000	0,005
24→25	2,828	$t_o$	0,2	1697,65	339,53	0,003
25→26	21,4	$t_o$	0,2	1697,65	339,53	0,025
26→27	4	$t_o$	0,2	1340,25	268,05	0,006
27→28	1,414	$t_o$	0,2	1273,24	254,65	0,002
28→29	22	$t_o$	0,2	1273,24	254,65	0,035
29→30	3,5	$t_o$	0,2	1083,61	216,72	0,006
30→31	1,414	$t_o$	0,2	1039,38	207,88	0,003
31→32	35	$t_o$	0,2	1039,38	207,88	0,067
32→33	1,5	$t_o$	0,2	960,936	192,19	0,003
33→ИТ	101,479	$t_{ме}$	–	–	5000	0,020
Переход 3						
Т5	–	$t_{ме}$	–	–	–	0,017
ИТ→34	27,44	$t_{ме}$	–	–	5000	0,005
34→35	1,75	$t_o$	0,1	1787	178,7	0,010
Пауза		$t_o$				0,008
35→34	1,75	$t_{ме}$	–	–	5000	0,000
34→36	18,8	$t_{ме}$	–	–	5000	0,004
36→37	2	$t_o$	0,1	1818,91	181,89	0,011
Пауза	–	$t_o$	–	–	–	0,008
37→36	2	$t_{ме}$	–	–	5000	0,000
36→38	2,6	$t_o$	0,1	1818,91	181,89	0,014
38→39	2,828	$t_o$	0,1	1818,91	181,89	0,016
Пауза		$t_o$				0,008
39→40	2	$t_{ме}$	–	–	5000	0,000
40→ИТ	42,757	$t_{ме}$	–	–	5000	0,009
					Итого	0,915

Время на установку и снятие заготовки:  $t_{в.у} = 0,13$  мин [6, с. 59].

Вспомогательное время, связанное с операцией:  $t_{в.он} = 0,32 + 0,15 + 0,03 = 0,5$  мин [6, с. 79].

Время на измерения:  $t_{в.изм} = 0,08 + 0,08 + 0,1 = 0,26$  мин [6, с. 84].

Так как время на измерения меньше времени работы по программе, в дальнейшем его не учитываем. Тогда вспомогательное время:

$$t_g = 0,13 + 0,5 = 0,63 \text{ мин.}$$

Принимаем  $K_{тв} = 1$  ([6, с. 50], партия – 100 шт.),  $a_{мех} + a_{опз} = 2,3 \%$  [6, с. 91],  
 $a_{омд} = 2 \%$  [6, с. 93].

Штучное время

$$t_{ум.} = (0,915 + 0,63 \cdot 1) \cdot \left(1 + \frac{2,3 + 2}{100}\right) = 1,545 \cdot 1,043 = 1,61 \text{ мин.}$$

Составляющие подготовительно-заключительного времени:

– норма времени на организационную подготовку [6, с. 96]: получить наряд, чертеж, технологическую документацию, программоноситель, инструмент, приспособление, заготовки и сдать их после окончания обработки на рабочем месте – 4 мин; ознакомиться с работой, чертежом, технологической документацией, осмотреть заготовки – 2 мин; инструктаж мастера – 2 мин. Тогда

$$t_{н.з.1} = 4 + 2 + 2 = 8 \text{ мин;}$$

– норма времени на наладку станка [6, с. 96]: установить и снять центр – 0,8 мин; сместить заднюю бабку – 0,3 мин; установить и снять инструментальный блок (инструмент) –  $3 \cdot 0,5$  мин; установить программоноситель в считывающее устройство и снять – 1 мин; настроить нулевое положение –  $3 \cdot 2$  мин; настроить устройство для подачи СОТС – 0,2 мин. Тогда

$$t_{н.з.2} = 0,8 + 0,3 + 1,5 + 1 + 6 + 0,2 = 9,8 \text{ мин.}$$

Подготовительно-заключительное время

$$t_{н.з.} = t_{н.з.1} + t_{н.з.2} + t_{пр.обр} = 8 + 9,8 + 0,915 = 18,71 \text{ мин.}$$

Штучно-калькуляционное время токарной операции

$$t_{ум-к.} = t_{ум.} + \frac{t_{н.з.}}{n} = 1,61 + \frac{18,71}{100} = 1,79 \text{ мин.}$$

8 Ответы на контрольные вопросы.

9 Выводы.

### **Контрольные вопросы**

- 1 Дайте определение нормы времени.
- 2 Чем штучно-калькуляционное время отличается от штучного?
- 3 Перечислите составляющие нормы времени.

## Список литературы

- 1 **Жолобов, А. А.** Программирование процессов обработки поверхностей на станках с ЧПУ : учебное пособие / А. А. Жолобов, Ж. А. Мрочек, А. М. Федоренко. – Могилев : Белорус.-Рос. ун-т, 2009. – 339 с.
- 2 Автоматизация подготовки управляющих программ для станков с ЧПУ / В. И. Аверченков [и др.]. – Брянск : БГТУ, 2010. – 212 с.
- 3 Станки с ЧПУ в машиностроительном производстве : учебное пособие: в 2 ч. / В. И. Аверченков [и др.]. – Брянск : БГТУ, 2010. – Ч. 1. – 303 с.
- 4 Технология изготовления деталей на станках с ЧПУ : учебное пособие / Ю. А. Бондаренко [и др.]. – Старый Оскол : ТНТ, 2011. – 292 с.
- 5 Станки с ЧПУ : устройство, программирование, инструментальное обеспечение и оснастка : учебное пособие / А. А. Жолобов [и др.]. – Москва : ФЛИНТА; Наука, 2017. – 360 с. : ил.
- 6 Общемашиностроительные нормативы времени и режимов резания для нормирования работ, выполняемых на универсальных и многоцелевых станках с числовым программным управлением. Ч: 1 : Нормативы времени. – Москва : Экономика, 1990. – 206 с.