МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФГБОУ ВО «Брянский государственный технический

Политехнический колледж (ПК БГТУ)

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 00DF212636081B4F2FEC2C849AFF43F11D
Владелец: Федонин Олег Николаевич
Действителен: с 27.07.2023 до 19.10.2024

УТВЕРЖДАЮ
Ректор ФГБОУ ВО БГТУ
О.Н. Федонин
«29» апреля 2022 г.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ
по выполнению лабораторных работ и практических работ
учебной дисциплины
ОП.10 «Программирование для автоматизированного оборудования»

Специальность:	**15.02.08 Технология машиностроения**
Уровень образования выпускника:	среднее профессиональное образование (СПО)
Программа подготовки специалиста среднего звена (ППССЗ):	Базовая
Присваиваемая квалификация:	Техник
Форма обучения:	очная
Срок получения СПО по ППССЗ:	3 года 10 месяцев
Уровень образования, необходимый для приема на обучение по ППССЗ:	основное общее образование
Год приема на обучение на 1-й курс:	2022

Брянск 2022

Методические рекомендации по выполнению практических занятий

по учебной дисциплине **ОП. 10 «Программирование для автоматизированного оборудования»** (далее — МУ)
для специальности ***15.02.08 Технология машиностроения***

Разработал:

– преподаватель ПК БГТУ О.А. Василенко

МР рассмотрена и одобрена на заседании
предметно-цикловой комиссии
«Технология машиностроения»
от «29» ____апреля___ 2022 г., протокол № 9

Председатель ПЦК И.А.Тарусова

Согласовано:

Заместитель директора ПК БГТУ
по учебно-методической работе

Т.Е.Балашова

© *Василенко О.А.*
© ФГБОУ ВО «Брянский государственный технический университет»

Лабораторная работа №1

Тема: «Разработка УП обработки группы отверстий на сверлильном станке с ЧПУ. Ввод УП и редактирование .
Цель: приобретение практических навыков разработки УП для обработки детали на сверлильном станке с ЧПУ.

1 Материальное обеспечение
1.1 Инструкция к практическому занятию.
1.2 Чертежи (эскизы) деталей.
1.3 Расчетно-технологическая карта.
1.4 Руководство программиста ТС для устройства числового программного управления типа NC-201M.

2 Порядок выполнения работы
2.1 Сообщение темы, плана, постановка цели урока.
2.2 Индивидуальная и групповая консультация студентов во время работы.
2.3 Выполнить технологический эскиз детали.
2.4 Выбрать систему координат.
2.5 Разработать управляющую программу для обработки детали на сверлильном станке с ЧПУ.
2.6 Оформить отчет.

3. Технологические рекомендации
3.1 Все размеры чертежа детали должны быть подсчитаны от единой базы.
3.2 База должна быть привязана к осям «плавающего» нуля.
3.3 Отверстия должны быть пронумерованы согласно оптимальному порядку обработки. Нумерация отверстий облегчает процесс программирования при большом количестве переходов.
3.4 Оптимальность порядка обработки заключается в минимальном времени, затраченном на обработку.

4 Характеристики программирования
4.1 <u>Системы измерения</u>
Миллиметры или дюймы, выбираемые посредством функции G71/G70.
4.2 <u>Программирование абсолютное или по приращениям</u>
Подготовительная функция G90 – абсолютное программирование, G91 – программирование по приращениям.
4.3 <u>Программирование относительно нуля станка</u>

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

4.4 Программирование с десятичной точкой

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

4.5 Код ленты

EIA RS244, ISO 840 с автоматическим распознанием.

4.6 Координаты осей

Координаты программируются в миллиметрах или в дюймах от +(-)0,001 до +(-) 99999,9999.

4.7 Координаты I,Y

Определяют координаты центра окружности в круговой интерполяции I по оси абсцисс и Y по оси ординат.

4.8 Функция F

Программируется 0,01 до 99999

G94 – определяет скорость подачи осей в мм/мин

G95 – определяет скорость подачи осей в мм/об

4.9 Функция S

Может выражать:

- число оборотов/мин шпинделя (G97)
- скорость резания в м/мин(G96)

4.10 Функция T

Определяет требуемы для обработки инструмент и номер коррекции для данного инструмента. Программируемая величина: 1,0 до 9999,999. Цифры перед десятичной точкой определяют инструмент, после – номер корректора.

4.11 Подготовительные функции

G00 – быстрое позиционирование;

G01 – линейная интерполяция;

G02 – интерполяция круговая по часовой стрелки;

G03 – интерполяция круговая против часовой стрелки;

G04 – выдержка времени заданная в кадре;

G09 – замедление в конце кадра;

G17 – выбирает плоскость интерполяции (X,Y);

G18 – выбирает плоскость интерполяции (Z,X);

G19 – выбирает плоскость интерполяции (Y,Z);

G27 – непрерывная обработка с автоматическим уменьшением скорости на углах;

G28 – непрерывная обработка без автоматического уменьшения скорости на углах;

G29 – позиционирование от точки к точки;

G33 – нарезание резьбы с постоянным или изменяющимся шагом;

G34 – нарезание резьбы с постоянным или изменяющимся шагом;
G35 – синхронизированное начало движение 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;
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;
G86 – цикл развертывания;
G89 – цикл развертывания с установкой;
G90 – абсолютное программирование;
G91 – программирование по приращениям;
G94 – скорость подачи осей, мм/мин;
G95 – скорость подачи осей мм/об;
G96 – скорость вращения шпинделя, м/мин;
G97 – скорость вращения шпинделя об/мин;

4.12 Вспомогательные функции М

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;
M06 – смена 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.

4.13 Постоянные циклы

С использованием подготовительных функций G81-G89 определение подготовительного цикла можно программировать ряд операций (сверление, нарезание резьбы метчиком, растачивание и т.д.) без повторения для каждой из них параметров отверстия, запрограммированную обработку которого надо осуществить. Характеристики постоянных постоянных циклов приведены в таблице 1

Таблица 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	в прерывистой работе	нет	рабочая скорость	быстрый ход
G84 нарезание резьбы метчиком	рабочая подача, начало вращения шпинделя	нет	изменение направления	рабочая подача
G85 рассверливание	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

5 Пример разработки УП для обработки детали «Фланец» на сверлильном станке с ЧПУ.

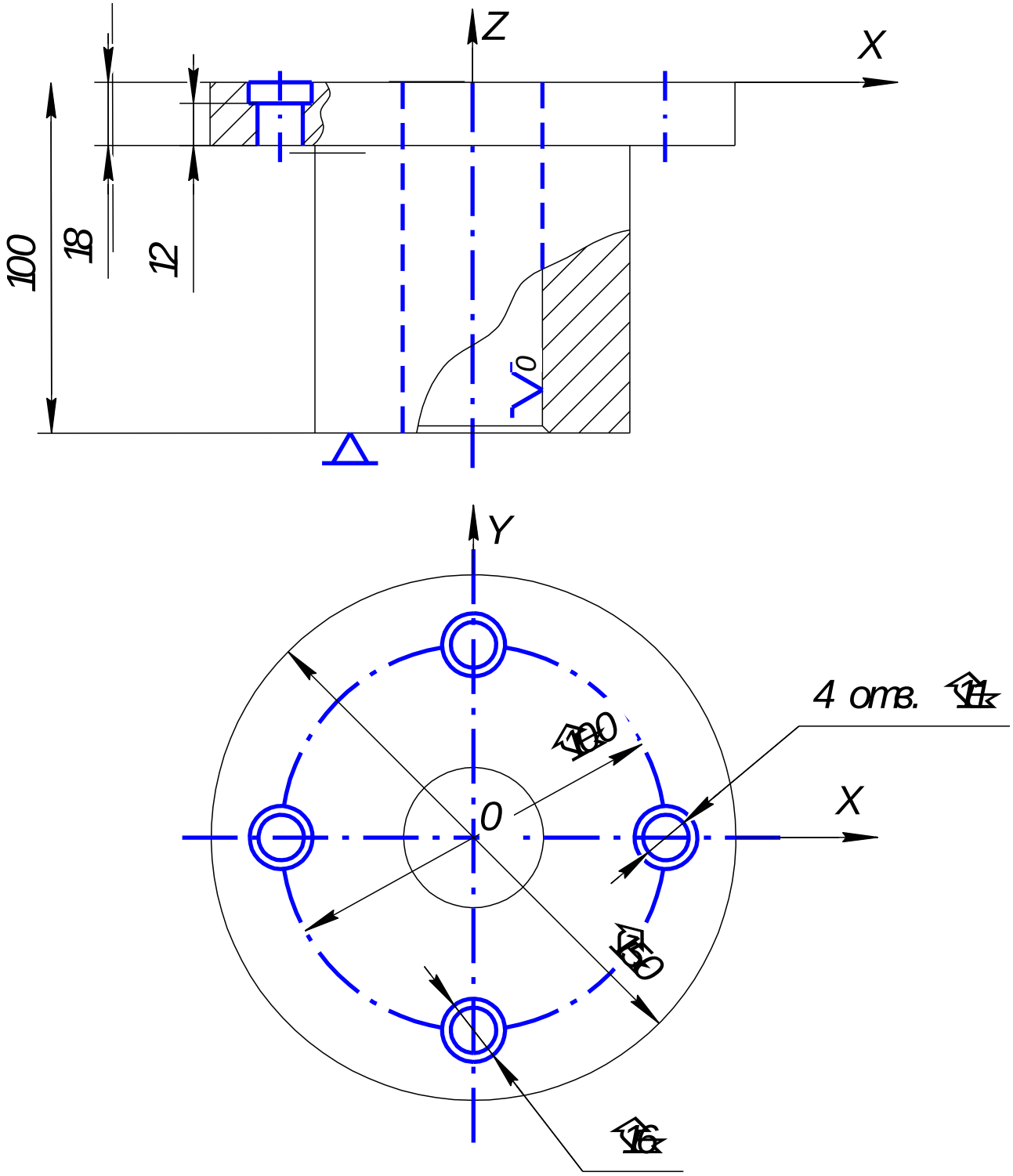


Рисунок 1 – Технологический эскиз детали

№1 T1.1 M06 (сверло диаметром 11мм)
№2 G97 M13 S710
№3 G0 G90 G95 X0 Y50
№4 G81 R2 Z-23 F0.2
№5 Y0 X-50
№6 X0 Y-50
№7 Y0 X50
№8 G80
№9 G0 Z500
№10 M5
№11 T2.2 M06 (цековка диаметром 16мм)
№12 G97 M13 S280
№13 G0 G90 G95 X0 Y50
№14 G8 R2 Z-14 F0.07
№15 Y0 X-50
№16 X0 Y-50
№17 Y0 X50
№18 G80
№19 G0 Z500
№20 M5
№21 M30

Лабораторная работа №2

Тема: «Разработка УП обработки детали на токарном станке с ЧПУ. Ввод УП и редактирование .

Цель: приобретение практических навыков разработки УП для обработки детали на токарном станке с ЧПУ.

1 Материальное обеспечение

1.1 Инструкция к практическому занятию.

1.2 Чертежи (эскизы) деталей.

1.3 Чертежи (эскизы) заготовок.

1.4 Расчетно-технологическая карта.

1.5 Руководство программиста ТС для устройства числового программного управления типа NC-201M.

2 Порядок выполнения работы

2.1 Сообщение темы, плана, постановка цели урока.

2.2 Индивидуальная и групповая консультация студентов во время работы.

2.3 Выполнить технологический эскиз детали.

2.4 Выбрать систему координат.

2.5 Разработать управляющую программу для обработки детали на токарном станке с ЧПУ.

2.6 Оформить отчет.

3 Характеристики программирования

3.1 <u>Системы измерения</u>

Миллиметры или дюймы, выбираемые посредством функции G71/G70.

3.2 <u>Программирование абсолютное или по приращениям</u>

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

3.3 <u>Программирование относительно нуля станка</u>

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

3.4 <u>Программирование с десятичной точкой</u>

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

3.5 <u>Код ленты</u>

EIA RS244, ISO 840 с автоматическим распознанием.

3.6 <u>Координаты осей</u>

Координаты программируются в миллиметрах или в дюймах от +(-)0,001 до +(-) 99999,9999.

3.7 Координаты I,Y

Определяют координаты центра окружности в круговой интерполяции I по оси абсцисс и Y по оси ординат.

3.8 Функция F

Программируется 0,01 до 99999

G94 – определяет скорость подачи осей в мм/мин

G95 – определяет скорость подачи осей в мм/об

3.9 Функция S

Может выражать:

- число оборотов/мин шпинделя (G97)
- скорость резания в м/мин(G96)

3.10 Функция T

Определяет требуемы для обработки инструмент и номер коррекции для данного инструмента. Программируемая величина: 1,0 до 9999,999. Цифры перед десятичной точкой определяют инструмент, после – номер корректора.

3.11 Подготовительные функции

G00 – быстрое позиционирование;
G01 – линейная интерполяция;
G02 – интерполяция круговая по часовой стрелки;
G03 – интерполяция круговая против часовой стрелки;
G04 – выдержка времени заданная в кадре;
G09 – замедление в конце кадра;
G17 – выбирает плоскость интерполяции (X,Y);
G18 – выбирает плоскость интерполяции (Z,X);
G19 – выбирает плоскость интерполяции (Y,Z);
G27 – непрерывная обработка с автоматическим уменьшением скорости на углах;
G28 – непрерывная обработка без автоматического уменьшения скорости на углах;
G29 – позиционирование от точки к точки;
G33 – нарезание резьбы с постоянным или изменяющимся шагом;
G34 – нарезание резьбы с постоянным или изменяющимся шагом;
G35 – синхронизированное начало движение 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;
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;
G86 – цикл развертывания;
G89 – цикл развертывания с установкой;
G90 – абсолютное программирование;
G91 – программирование по приращениям;
G94 – скорость подачи осей, мм/мин;
G95 – скорость подачи осей мм/об;
G96 – скорость вращения шпинделя, м/мин;
G97 – скорость вращения шпинделя об/мин;

3.12 Вспомогательные функции М

М00 – остановка программы;
М01 – условная остановка программы;
М02 – конец программы;
М03 – вращение шпинделя по часовой стрелке;
М04 – вращение шпинделя против часовой стрелки;
М05 – остановка вращения шпинделя;
М06 – смена инструмента;
М07 – включение вспомогательного охлаждения;
М08 – включение основного охлаждения;
М09 – выключение охлаждения;
М10 – блокировка осей;
М11 – разблокировка осей;
М12 – блокировка вращающихся осей;
М13 – вращение шпинделя по часовой стрелке и охлаждение;
М14 – вращение шпинделя против часовой стрелки и охлаждение;
М19 – остановка вращения шпинделя с угловой ориентацией;
М30 – конец программы и возврат к первому кадру.

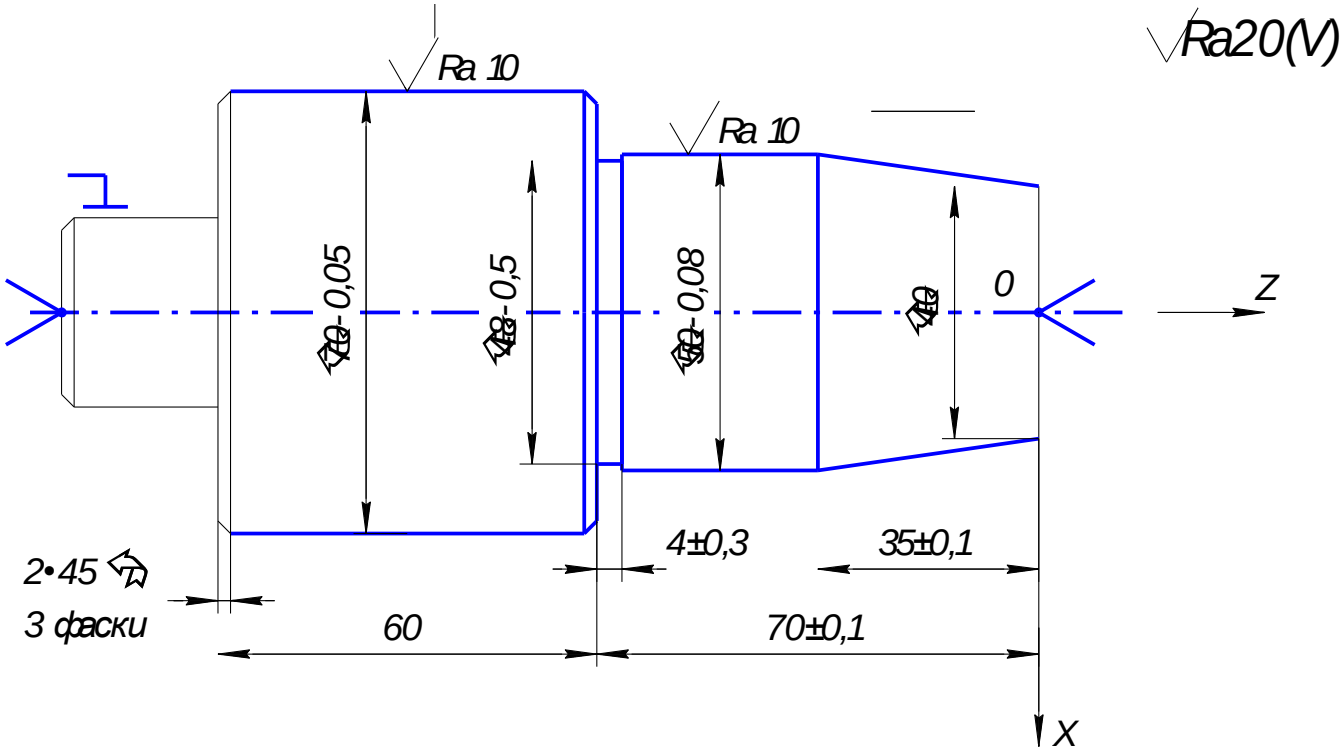
3.13 Постоянные циклы

С использованием подготовительных функций G81-G89 определение подготовительного цикла можно программировать ряд операций (сверление, нарезание резьбы метчиком, растачивание и т.д.) без повторения для каждой из них параметров отверстия, запрограммированную обработку которого надо осуществить. Характеристики постоянных постоянных циклов приведены в таблице 1

Таблица 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	в прерывистой работе	нет	рабочая скорость	быстрый ход
G84 нарезание резьбы метчиком	рабочая подача, начало вращения шпинделя	нет	изменение направления	рабочая подача
G85 рассверливание	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

4 Пример разработки УП для обработки детали «Вал» на токарном станке с УЧПУ типа NC-201M.



№1 Т1.1 М06 (проходной резец)

Рисунок 1 – Технологический эскиз детали.

№2 M04 S540
№3 G0 G90 G95 X40 Z2
№4 G1 Z0 F0.6
№5 X50 Z-35
№6 Z-70
№7 X70
№8 Z-132
№9 G0 X75
№10 Z-68
№11 X64
№12 G1 Z-70 F0.6
№13 Z70.2 Z-72.1
№14 G0 X70 Z20
№15 T2.2 M06 (канавочный резец)
№16 S630
№17 G0 Z-68 X54
№18 G1 Z-70 G4
№19 F0.15
№20 X48
№21X55
№22 Z20 X80
№23 M5

Практическое занятие №1

Тема: Определение последовательности обработки группы отверстий на сверлильном станке с ЧПУ. Назначение режима резания.

Цель занятия: Подготовка исходных данных для разработки УП обработки группы отверстий на сверлильном станке с ЧПУ.

1. Материальное обеспечение.

1.1 Инструкция к практическому занятию.

1.2 Чертеж (эскиз) детали.

1.3 Справочная и техническая литература.

2. Порядок выполнения работы.

2.1 Сообщение темы, плана, постановка цели урока.

2.2 Определение последовательности обработки отверстий. Выбор режущего инструмента.

2.3 Назначение режима резания.

2.4 Расчет величины перемещения инструмента по оси Z

2.5 Выполнить технологический эскиз.

3. Общие положения.

Для обработки отверстий на станках с ЧПУ по возможности режущий инструмент принимать с меньшим вылетом из шпинделя. Центрование рекомендуется производить спиральными сверлами $2\varphi = 90°$ или центровочными комбинированными.

Последовательность переходов сверлильной операции выбирают с учетом конфигурации отверстий, допустимых отклонений, формы и относительного положения осей отверстий, числа группы одинаковых отверстий, точности и времени позиционирования стола и поворота револьверной головки. Перед сверлением отверстия должны центрироваться, это повышает точность положения оси отверстия и предотвращает увод сверла при врезании. В отдельных случаях когда допуск на межцентровое расстояние более 0,2 мм по стали или 0,3 мм по чугуну, либо когда в револьверной головке не хватает позиции для центровочного сверла, допускается сверление без центрования.

Инструмент в револьверной головке располагается в порядке чередования переходов.

Возможны два варианта последовательности основных переходов сверлильной операции:

1) Каждый инструмент обрабатывает все отверстия данного размера, а потом производится смена инструмента.

2) Каждое отверстие обрабатывается всеми инструментами.

Вначале выполняются переходы связанные с большей нагрузкой на инструмент, затем с меньшей, т.е. сначала обрабатываются отверстия большего диаметра, затем меньшего. Переходы связанные с обеспечением высокой точности размера выполняются всегда в конце операции. Такая последовательность предохраняет уже обработанные отверстия от деформаций вызываемых обработкой следующих отверстий.

4. Пример выполнения практического занятия.

4.1 Определяем последовательность обработки отверстий. Выбор режущего инструмента.

Переход 1. Установить деталь, закрепить, снять.

Переход 2. Сверлить 4 отв. ∅11 /Т1/
Режущий инструмент: Сверло ∅11 ГОСТ 4010-77

Переход 3. Цековать 4 отв. ∅16 /Т2/
Режущий инструмент: цековка ∅16 ГОСТ 2И25-2-80

4.2 Назначаем режим резания.

Переход 2.

t=5,5мм; So=2мм/об; V=24,5м/мин; Kv=1,0

$$n = \frac{1000 \cdot V}{\pi \cdot D} = \frac{1000 \cdot 24{,}5}{3{,}14 \cdot 11} = 709\text{мин}^{-1}$$

$n=710\text{мин}^{-1}$

$$Vd = \frac{\pi \cdot D \cdot nd}{1000} = \frac{3{,}14 \cdot 11 \cdot 710}{1000} = 24{,}5\text{м/мин}$$

$Vs=So \cdot nd=0{,}2 \cdot 710=142$мм/мин

$Vs_d=125$мм/мин

Переход 3.

$$t = \frac{D - d}{2} = \frac{16 - 11}{2} = 2{,}5\text{мм}$$

*So=0,07*мм/об

*V=14*м/мин

$$n = \frac{1000 \cdot 14}{3{,}14 \cdot 16} = 278\text{мин}^{-1}$$

nd=280мин^{-1}

$$Vd = \frac{3{,}14 \cdot 16 \cdot 280}{1000} = 14{,}06\text{м/мин}$$

*Vs=0,07·280=19,6*мм/мин

$Vs_d = 20\text{мм/мин}$

4.3 Расчет величины перемещения инструмента по оси Z.

Переход 2.

$L = 2 + \gamma + e + \Delta, \text{мм}$

Где: 2 - недоход сверла до детали по программе 2мм;

у - величина врезания; у=0,3·d=0,3·11=3,3 – Для сверления;

е - длина обрабатываемой поверхности; е – 18мм;

Δ - перебег; Δ = 1…3 (принимаем Δ=2мм)

L=2+3,3+18+2=25,3мм

Переход 3.

2мм – недоход цековки по программе;

у=0, Т.к. $\varphi = 90°$;

е=12мм – из чертежа;

$\Delta = 0$

L=2+12=14мм

5. Технологический эскиз.

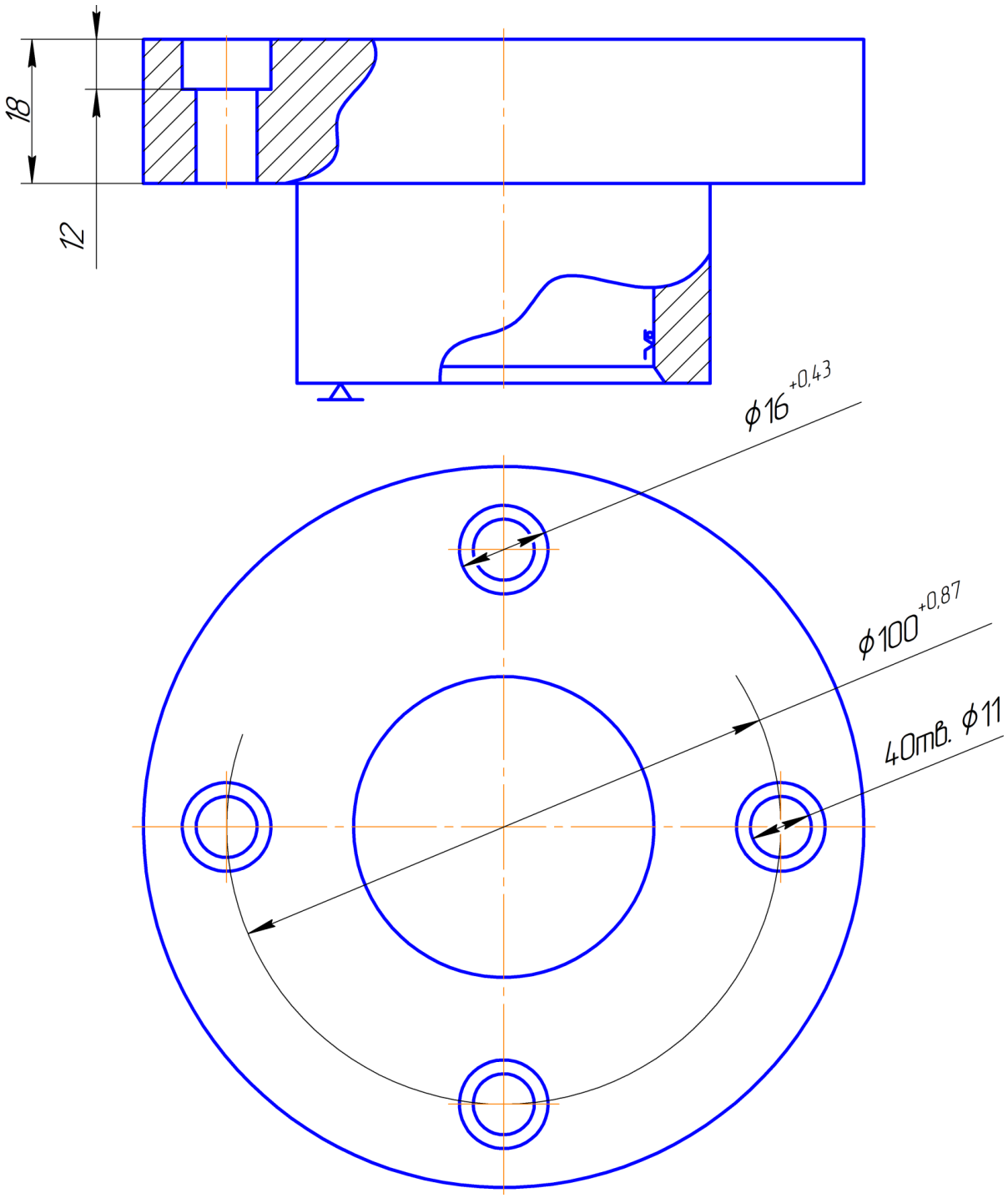


Рисунок 1 Технологический эскиз детали.

Практическое занятие №2

Тема: Определение последовательности обработки детали на токарном станке с ЧПУ. Назначение режима резания.

Цель занятия: Подготовка исходных данных для разработки УП обработки детали на токарном станке с ЧПУ.

1. Материальное обеспечение

1.1 Инструкция к практическому занятию.

1.2 Чертеж (эскиз) детали.

1.3 Справочная и техническая литература

2. Порядок выполнения работы

2.1 Сообщение темы, плана, постановка цели урока.

2.2 Определение последовательности обработки детали. Выбора режущего инструмента.

2.3 Назначение режима резания.

2.4 Построение циклограмм движения инструмента.

3. Пример выполнения практического занятия

3.1 Определяем последовательность обработки детали.

Переход 1. Установить деталь, закрепить, снять.

Переход 2. Точить поверхность ∅50, подрезать торец в размер , точить поверхность ∅70 на проход, точить фаску 2х45˚, точить конус.
Режущий инструмент: резец проходной, подрезной MWLNR 2525M08, $\varphi = 95°$, Т15К6,
ТУ 2-035-829-82.

Переход 3. Точить канавку.
Режущий инструмент: резец канавочный 035-2126-1183, Т15К6, ОСТ 2И10-7-84

3.2 Назначение режима резания.

Переход 2.

t=3мм; So=0,6мм/об; V=85,4м/мин

$$n = \frac{1000 \cdot V}{\pi \cdot D} = \frac{1000 \cdot 85{,}4}{3{,}14 \cdot 50} = 540\text{мин}^{-1}$$

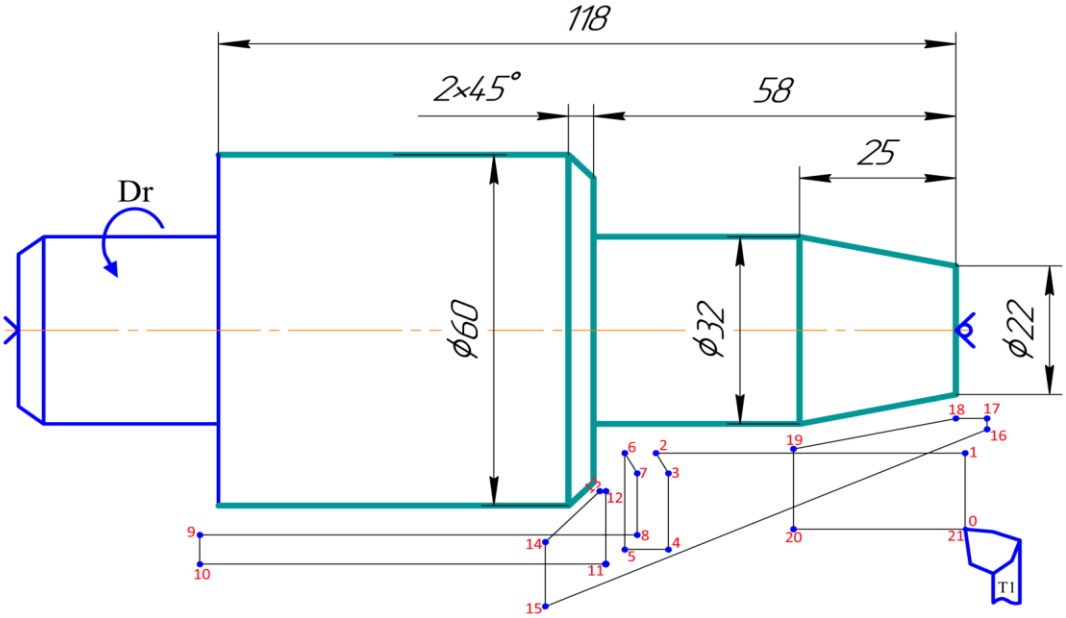
Переход 3.

t=3мм; So=0,15мм/об; V=100м/мин

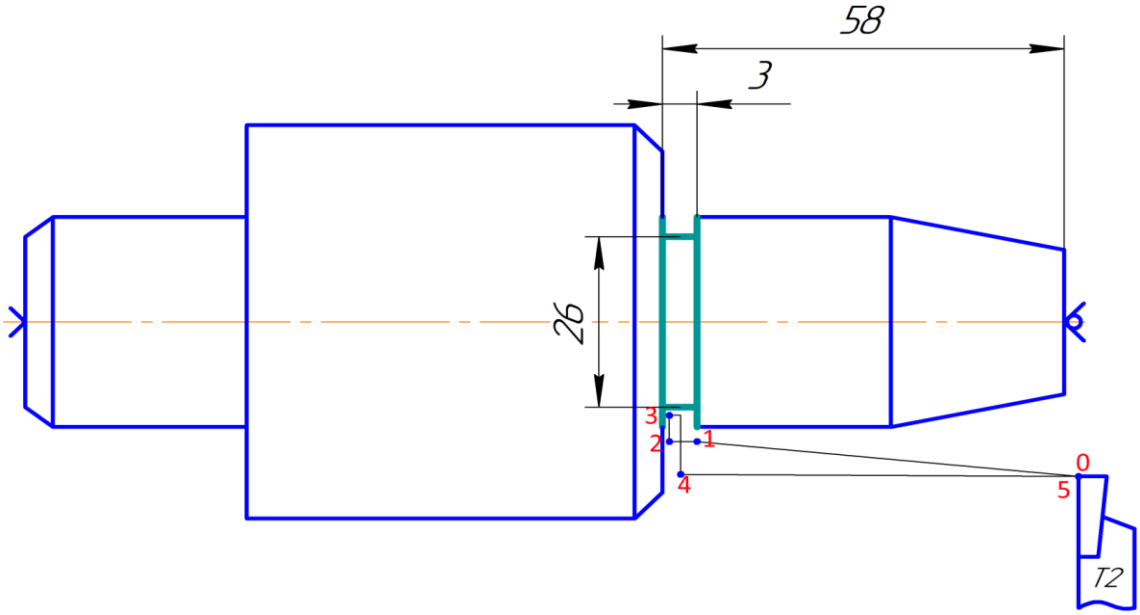
$$n = \frac{1000 \cdot V}{\pi \cdot D} = \frac{1000 \cdot 100}{3{,}14 \cdot 50} = 630\text{мин}^{-1}$$

4. Построение циклограмм движения инструмента.

Переход 2.



Переход 3.



Практическая работа №3

Тема: Разработка УП обработки детали на токарном станке с использованием САП.

Цель занятия: Приобретение практических навыков разработки УП для обработки детали на токарном станке с использованием САП.

1. Материальное обеспечение
1.1. Инструкция к лабораторным занятиям
1.2. Чертежи (эскизы) деталей
1.3. Токарный станок с ЧПУ типа NC-201N
2. порядок выполнения работы.
2.1. Сообщение темы, плана, постановка цели урока
2.2. Индивидуальные задания для выполнения работы
2.3. Анализ чертежа (эскиза) детали
2.4. Разработка УП с применением САП.
2.5. Отработка УП.
3. Геометрическое программирование высокого уровня (GTL)

В УЧПУ представляется возможным в программе описать геометрический профиль в плоскости, используя не только стандартный язык программирования (G1-G2-G3), но и язык программирования высокого уровня GTL. Этот язык позволяет программировать профиль, состоящий из прямых и окружностей, используя только информацию, полученную с чертежа; система сама вычисляет точки пересечения и точки касания геометрических элементов.

Язык программирования GTL и стандартный язык могут быть использованы одновременно в одной и той же программе, но не для одного и того же профиля. Геометрия GTL функционирует только при абсолютном программировании (G90).

Векторная геометрия

Определение профиля с использованием GTL основано на использовании 4 типов геометрических элементов:

- точки начала отсчета;
- точки;
- прямые;
- окружности.

Так как профиль определяется, как геометрическими элементами, так и направлением, то для определения геометрических элементов в языке GTL

используется особый тип геометрии - ВЕКТОРНАЯ ГЕОМЕТРИЯ. Для векторной геометрии определение элемента требует, кроме параметров, необходимых для установления позиции в плоскости, также назначение направления движения.

Например, прямая линия проходит через точки А и В (рисунок А.25), двигаясь от А к В, или прямая линия l', лежащая на l, но проходящая от В к А.

В векторной геометрии l и l' являются двумя различными линиями, имеющими противоположные направления. Программирование при помощи GTL, основанное на векторной геометрии, требует для каждой прямой линии назначение направления движения. Условимся, что направление движения прямой определяется углом, который она образует с положительной осью X. Правильный угол получается при вращении положительной оси X до наложения его на одну из прямых линий, которую надо определить. Руководство программиста ТС 54 Угол будет иметь положительный знак, если ось X будет вращаться против часовой стрелки, и отрицательный - в обратном случае (см. рисунок А.26).

Направление должно быть придано также и окружностям. Условно принимается за положительное направление движение против часовой стрелки и за отрицательное - по часовой стрелке (см. рисунок А.26).

По договоренности придается положительное значение радиуса окружностям с направлением движения против часовой стрелки и отрицательное - в обратном случае.

Направление, данное элементу, обычно соответствует направлению движения по профилю. Однако возможно направление элемента во время определения профиля, если это направление противоположно остальным элементам профиля (см. рисунок А.27).

Хранение в памяти геометрических элементов

Хранение в памяти геометрических элементов предусматривает использование строчных символов a-l-c-d-m-o-r-p-s-b для определения соответственно:

- ✓ углов;
- ✓ прямых линий;
- ✓ окружностей;
- ✓ расстояний;
- ✓ модулей;
- ✓ точек начала отсчета;
- ✓ радиуса;

- ✓ точки, числа пересечений;
- ✓ скоса.

Необходимость использования для этой информации строчных символов вызвана тем, что эти же заглавные символы используются в языке ЧПУ для другой информации. Запоминание геометрических элементов в памяти осуществляется до определения профиля. Элементами, рассматриваемыми в GTL, являются: прямые, окружности, точки, точки начала отсчета. Это - геометрические переменные, идентифицированные НАЗВАНИЕМ и ИНДЕКСОМ.

Геометрическая переменная определяется в кадре назначения.

Формат:

НАЗВАНИЕ ИНДЕКС = «выражение» ,

где:

НАЗВАНИЕ - одно из четырёх символических названий, предусмотренных для геометрических элементов:

1) o – для определения точки начала отсчёта;
2) p – для определения точки;
3) l - для определения прямой;
4) c - для определения окружности.

ИНДЕКС - определяет номер переменной геометрического элемента. Этот номер заключен между 0 и 255. Максимальный предел определяется при конфигурации;

выражение - содержит всю информацию, необходимую для описания геометрического элемента. Элементы могут быть определены:

- ✓ явным образом, программируя в кадре всю информацию, необходимую для определения геометрического элемента;
- ✓ неявным образом, вызывая другие геометрические элементы, определённые ранее.

Пример хранения в памяти элементов:

o1 = Z30 X30 a45

p1 = o1 Z15 X15

p2 = Z60 X30

L1 = p1, p2

L2 = Z30 X50, a45

c1 = L1, L2, r15

L3 = Z0 X0, Z100 X60

p3 = L3, c1

c2 = p3, r8

Число геометрических элементов, хранимых в памяти, определяется на стадии конфигурации системы. Формат геометрических определений предусматривает использование символа «,» (запятая) для разделения геометрического элемента (прямая - точка - окружность) от последующего геометрического элемента или информации, та- кой как радиус «r» или угол «a».

Примеры

p1 = Z30 X30

c1 = I10 J20 r30 разделитель не требуется

```
L1 = Z20 X20, Z100 X-10
       |            |
     точка       точка

L2 = I30 J20 r10, Z80 X80
       |                |
   окружность  точка

L3 = Z100 X100, a45
       |              |
     точка         угол

c3 = L1, L2, r18
     |  |   |
     |  |   радиус
     |  прямая
   прямая
```

Дискриминатор s2 служит для выбора второго пересечения (s2). Иллюстрация при- мера приведена на рисунке А.28.

Список возможных геометрических определений

Точки начала отсчёта:

pn= Z.. X.. a..

Точки:

pn = [om] Z.. X..

pn = [om] m.. a..

pn = Lm, Lp

pn = [-]Lm, cp [,s2]

pn = cm, [-]Lp [,s2]

pn = cm, cp [,s2]

Прямые линии:

Ln = [om] Z.. X.., [op] Z.. X..

Ln = [om] Z.. X.., a..

Ln = [om] I.. J.. r.., [op] I.. J.. r..

Ln = [om] I.. J.. r.., a..

Ln = [om] I.. J.. r.., [op] Z.. X..

Ln = [om] Z.. X.., [op] I.. J.. r..

Ln = pm, pq Ln = pm, a..

Ln = [-]cm, [-]cp

Ln = [-]cm, a..

Ln = [-]cp, pm

Ln = pm, [-]cp

Ln = [-]lm, d..

Окружности:

cn = [om] l.. J.. r..

cn = [om] m.. a.. r..

cn = [-]lm, [-]lp, r..

cn = [-]lm, [-]cp, r..

cn = [-]cp, [-]lm, r..

cn = pm, [-]lp, r..

cn = [-]lp, pm, r..

cn = [-]cm, [-]cp, r

cn = pm, [-]cp, r..

cn = [-]cp, pm, r..

cn = pm, pq, r..

cn = pm, [-]lp

cn = pm, [-]cp [,s2]

cn = pm, pq, pr

cn = pm, r..

cn = [-]cm, [-]d..

Примечание - Последовательность двух точек (..) указывает на необходимость объявления цифровых величин. Элементы в квадратных скобках - необязательные и могут быть опущены.

Определение точек начала отсчёта

Функция определения точек начала отсчета дает возможность определить точки начала отсчёта в прямом формате (явным образом).

Обычно информация, находящаяся в программе, относится к системе осей, совпадающих с осями станка. Однако при проектировании деталь может быть выполнена на чертеже с использованием различных декартовых систем: абсолютной системы и других систем (начальных точек) отсчёта, которые могут быть приведены к абсолютной системе вращением и смещением осей. Геометрия GTL может быть определена при лю- бой системе начала отсчёта.

Формат:

on = Z.. X.. a..,

где: on - определяет название точки начала отсчета;

Z..X.. - координаты новой начальной точки;

a.. - угол вращения (положительный против часовой стрелки).

Пример приведён на рисунке А.29.

Определение точек

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

Система полярного начала отсчёта состоит из начальной точки, названной полю- сом, из которой начинается ось X, названная полярной осью (рисунок А.30).

Любая точка плоскости может быть определена при помощи длины отрезка (P), названной модулем, который соединяет её с полярной точкой при помощи величины угла, который образует отрезок прямой с полярной осью (рисунок А.31).

Формат:

1) Прямой

Точка в декартовых координатах (рисунки А.34-А.35):

pn = [on] Z.. X..

Точка в полярных координатах (рисунок А.36):

pn = [on] m.. a.. 2)

2) Косвенный

Точка пересечения двух прямых, определённых ранее (рисунок А.37):

pn = lm,lp

Точка пересечения прямой и окружности, определённых ранее (рисунок А.38):

pn = [-] lm, cp [,s2]

pn = cm, [-] lp [,s2]

Точка пересечения двух окружностей (рисунок А.39):

pn = cm, cp [,s2],

где: Pn - определяет название точки индекса n (n - число, заключённое между 1 и максимально конфигурируемым числом);

Z.. X.. - координаты точки;

[on] - начальная точка отсчёта индекса n, определённая ранее, к которой относятся координаты Z и X;

m.. - модуль полярного вектора;

a.. - угол полярного вектора;

cm cp - ранее определённые элементы окружности индекса m и p;

[-] lm - ранее определённые элементы прямой индекса m и p;

[-] lp - возможно изменить направление, вставляя знак «-»;

[,s2] - индикатор второго пересечения.

В случае пересечения прямая-окружность или наоборот, существуют два возможных решения (рисунок 47): окружность c1 и прямая l2 пересекаются в точках p1 и p2. Проходя прямую l2, следуя её направлению, сначала встречаем точку p1 (1-е пере- сечение), а затем - точку p2 (2-е пересечение). Для выбора второго пересечения (p2) следует использовать индикатор s2. Если он опущен, то выбирается первое пересечение (p1). Пример пересечения прямая-окружность показан на рисунке А.38.

Также в случае пересечения окружность-окружность существуют два возможных решения: окружности c1 и c2 пересекаются в точках p1 и p2 (рисунок А.48). Рассматривается сориентированная прямая, соединяющая центр 1-ой окружности с центром 2-ой окружности. Она делит плоскость на две полуплоскости. Для выбора точки в правой полуплоскости (p2) следует использовать индикатор s2. Если он опущен, то автоматически выбирается точка в левой полуплоскости (p1).

Пример пересечения окружность-окружность приведён на рисунке А.39.

Определение прямой линии

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

Направление прямой всегда от первого ко второму определённому элементу. В случае если прямая касается с окружностью, возможны два решения, т.к. прямая может быть касательной с одной или с другой стороной окружности. Для выбора требуемого решения следует убедиться в том, что в точке касания окружность и прямая имеют одно и то же направление.

Несовместимость направления движения геометрических элементов показана на ри- сунке А.40. Совместимость направления движения показана рисунке А.41.

Формат:

1) Прямой в явном виде

Прямая, проходящая через две точки (рисунок А.42):

ln = [om] Z..X.., [op] Z..X..

Прямая, проходящая через одну точку и образующая угол с осью абсциссы (рисунки А.44-А.45):

ln = [on] Z..X..,a..

Прямая, касательная к одной окружности и образующая угол с осью абсциссы (рисунки А.46-А.47):

ln = [om] I..J.. r..,a..

Прямая, касательная к двум окружностям (рисунки А.48-А.49):

ln = om I..J.. r.., op I.. J.. r..

Прямая, касательная к окружности и проходящая через точку (рисунок А.43):

ln = [om] I..J..r.., [op] Z..X..

ln = [om] Z..X.., [op] I.. J..r..

2) Косвенный

Прямая, проходящая через две точки (рисунок А.50):

ln = pm, pq

Прямая, проходящая через точку и образующая угол с осью абсциссы (рисунок А.54):

ln = pm, a..

Прямая, касательная к двум окружностям (рисунки А.52-А.53):

ln = [-]cm, [-] cp

Прямая, касательная к одной окружности и образующая угол с осью абсциссы (рисунок А.55):

ln = [-] cm, a..

Прямая, касательная к окружности и проходящая через точку (рисунок А.51):

ln = [-] cp,pm ln = pm, [-] cp

Прямая, параллельная прямой на расстоянии d (рисунки А.56-А.57):

ln = [-] lm, d..,

где: Ln - определяет название прямой с индексом n (n - число, заклю- чённое между 1 и максимально конфигурируемым номером);

Z..X.. - координаты точки;

a.. - угол, образованный осью абсциссы и прямой (положительный против часовой стрелки);

r.. - радиус окружности (положительный против часовой стрелки; pm pg - предопределённые элементы точки с индексом m и q;

[-] cm [-] cp - предопределённые элементы окружности с индексом m и q.

Направление движения окружности может быть изменено при помощи отрицательного знака для гарантирования совместимости направлений прямой и окружности в точке касания.

[-] lm - предопределённый элемент прямой с индексом m;

d.. - расстояние между двумя прямыми положительное, если прямая находится слева, отрицательное - в обратном случае, смотря в направлении предопределённом прямой.

Определение окружностей

Функция определения окружностей позволяет определить окружности в прямой форме (явной) или в косвенной (неявной) форме.

Определяя окружности в косвенной форме, программист должен учитывать совместимость направлений элементов (знак «-» может изменить направление предопределенных элементов). Если направление элементов не учитывается, то, задавая окружность известного радиуса и прямую линию, возможно от 1 до 8 решений получения окружности, касательной к двум элементам.

Окружности, касательные к прямой и окружности показаны на рисунке А.58.

Учитывая совместимость направлений движения предопределённых элементов и окружности, которую следует определить, количество возможных решений уменьшается до двух. Для различения двух возможных окружностей, имеющих одно и то же направление и один и тот же радиус, необходимо учитывать направление элементов, к которым от- носится определение, и две дуги окружности, в которой, если необходимо, элемент делится точками касания с предраспределёнными элементами: GTL всегда производит окружность с направлением от первого ко второму элементу, с дугой, имеющей меньший центральный угол. Окружности, касательные к меньшей дуге см. рисунок А.59. Окружность с3 получается при введении в определение прямой l1 в первой позиции и окружность с2 - во второй, т.к. элемент с3 позволяет движение от прямой l1 к окружности с2, с дугой, имеющей меньший центральный угол. Окружность с4 полу- чается при введении в определение окружности с2 в первой позиции и прямой l1 во второй, т.к. элемент с4 позволяет движение от окружности с2 к прямой l1, с дугой, имеющей меньший центральный угол. То же самое можно сказать для определения окружности, касательной к двум предопределённым окружностям: в этом случае возможны от 1 до 8 решений см. рисунок А.60.

Окружности, касательные к двум предопределённым окружностям, показаны на рисунке А.61.

Совместимость направлений движения предопределённых элементов и окружности, которую надо определить, сводит количество возможных решений до двух. Для разли- чения двух окружностей, имеющих одинаковое направление и одинаковый радиус, рас- сматриваются две дуги, в которых новый элемент разделён точками, касательными к элементам начала; GTL создаёт окружность, двигаясь от первой окружности ко вто- рой, с дугой, имеющей меньший центральный угол (рисунок А.61). Для получения окружности с3 при определении следует установить в правой позиции элемент с1, а затем с2. Для получения окружности с4 элемент с2 должен предшествовать элементу с1 в определении.

Формат прямого программирования

Окружность с декартовыми координатами центра и радиуса (рисунки А.62-А.63):

cn = [om] m.. a.. r..

Окружность с полярными координатами центра и радиуса (рисунок А.64):

cn = [om] m.. a.. r,

где: cn - устанавливает название окружности индекса n (n - номер, заключенный между 1 и макс. конфигурируемым номером);

I..J.. - координаты центра окружности;

r.. - радиус окружности (положительный для направления против часовой стрелки, отрицательный для направления по часовой стрелке); [-]

lm - предопределённые элементы прямой с индексом m и p;

[-] lp - может принять противоположное направление при использовании знака «-»;

pm pq pr - предопределённые элементы точки индекса m, q, r;

[-] cm [-] cp - предопределённые элементы окружности индексом m, p. Может принять противоположное направление при использовании знака «-»;

[s2] - атрибут для наибольшей из двух возможных окружностей;

d.. - расстояние между двумя окружностями положительное, если, смотря из [-] cm, cn находится слева от неё, отрицательное, если справа.

Формат косвенного программирования

Окружность с данным радиусом и касательная к двум предопределенным прямым (рисунок А.65):

cn = [-] lm, lp,r..

Окружность с данным радиусом и касательная к прямой и окружности (предопределенным) (рисунки А.66-А.68):

cn = [-] lm, [-] cp,r..

cn = [-] cp, [-] lm,r..

Окружность с данным радиусом, проходящая через предопределенную точку и касательная к предопределенной линии (рисунок А.69):

cn = pm, [-] lp,r..

cn = [-] lp,pm,r..

Окружность с данным радиусом и касательная к двум предопределенным окружностям (рисунки А.70-А.71):

cn = [-] cm, [-] cp,r..

Окружность с данным радиусом, проходящая через одну предопределенную точку и касательная к предопределенной окружности (рисунок А.72):

cn = pm, [-] cp,r..

cn = [-] cp,pm,r..

Окружность с данным радиусом, проходящая через две предопределенные точки (рисунок А.73):

cn = pm, pq, r..

Окружность с центром в предопределенной точке и касательная к предопределенной прямой (рискнок А.74):

cn = pm, [-] lp

Окружность с центром в предопределенной точке и касательная к предопределен- ной окружности (рисунок А.75):

cn = pm, [-] cp [,s2]

Окружность, проходящая через три точки (рисунок А.76):

cn = pm,pq,pr

Окружность с данным радиусом и с центром в точке (рисунок А.77):

cn = pm,r..

Окружность концентрическая к предопределенной окружности и отдаленная от нее на данную величину (рисунок А.78):

cn = [-] cm,d..,

где: cn - устанавливает название окружности индекса n (n - номер, заключённый между 1 и макс. конфигурируемым номером);

I..J.. - координаты центра окружности;

r.. - радиус окружности; положительный для направления против часовой стрелки, отрицательный для направления по часовой стрелке;

[-] lm - предопределённые элементы прямой с индексом m и p;

[-] lp - может принять противоположное направление при использовании знака «-»;

pm pq pr - предопределённые элементы точки индекса m, q, r;

[-] cm [-] cp - предопределённые элементы окружности индексом m, p. Может принять противоположное направление при использовании знака «-»;

[s2] - атрибут для наибольшей из двух возможных окружностей;

d.. - расстояние между двумя окружностями, положительное, если, смотря из [-] cm, cn находится слева от неё, отрицательное, если справа.

Определение профиля

Под профилем подразумевается последовательность геометрических элементов, накопленных в памяти системы до начала обработки. Профиль может быть открытым и закрытым.

Начало и конец профиля

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

G21 - устанавливает начало профиля;

G20 - устанавливает конец профиля.

Открытый профиль

Если профиль открытый, то он должен начинаться с точки (pn) и заканчиваться точкой, отличной от первой. 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, программируя в кадре функции G21 G41/G42, и закрыта на последней точке с функциями G20 G40. Пример приведён на рисунке А.79.

Закрытый профиль

Если профиль закрытый, то сначала следует запрограммировать последний элемент (см. рисунки А.80-А.81), а затем, после последнего элемента - вызвать первый элемент профиля. Первая точка скорректированного профиля - пересечение первого и последнего смещенных элементов (первая точка = последней точке). Компенсация радиуса должна быть в начале профиля в кадре вызова последнего элемента, программируя функции G21 G41/G42, и закрыта в конце профиля в кадре вызова первого элемента с функциями G20 G40. Если первый и/или второй элементы являются окружностями, то возможны два пересечения. Если не дается никакой дополнительной информации, система выбирает первое. В случае если необходимо второе пересечение, следует программировать дискриминатор s2. Дискриминатор s2 программируется в кадре вызова последнего элемента в начале профиля и на последнем элементе в кон- це профиля (см. рисунок А.81).

Примеры открытых и закрытых профилей

Если в кадре начала профиля запрограммировано pn, то это означает, что про- филь открытый; pn может быть запрограммировано в начале и в конце профиля, но не внутри профиля. Открытый профиль (см. рисунок А.79):

```
 ............
l1 = ZX 25,a
p1 = Z-20 X25
p2 = Z90 X25
c1 = I30 J25 r-14
c2 = I45 J25 r15
```

.....

G21 G42 p1 - первая точка

l1

r3

c1

s2

c2

s2 l1

G20

G40

p2 - последняя точка

.....

Компенсация радиуса должна быть открыта на первой точке профиля и закрыта на последней. Компенсация радиуса отменяется в первом кадре движения осей в плоскости профиля, следующего за функцией G40. Закрытый профиль (см. рисунок А.80):

.....

l5 = Z X-15,a180

.....

l1 = Z-30 X-15,a135

.....

G21 G42 15 - последний элемент

l1 - первый элемент

.....

l5 - последний элемент

G20 G40 l1 - первый элемент

Закрытый профиль (см. рисунок А.81):

c1 = I.. J.. r..

.....

l1 = Z.. X..,a90

.....

l5 = Z.. X..,a180

.....

G21 G42 15 s2 - последний элемент

c1 s2 - первый элемент

l1

.....

l5 s2 - последний элемент

G40 c1 - первый элемент G40 c1

.....

Компенсация радиуса должна быть открыта в начале профиля, в кадре вызова последнего элемента и закрыта в конце профиля, в кадре вызова первого элемента. Руководство программиста ТС 62 Компенсация радиуса отменяется в первом кадре движения осей в плоскости профиля, следующего за функцией G40.

Движение осей шпинделя В любой точке профиля представляется возможным двигать оси, не участвующие в контурной обработке, даже на первой точке, например, для входа в деталь. В случае открытых профилей движение на первой точке программируется после программирования точки. В случае закрытых профилей оно должно быть запрограммировано между определением последнего элемента профиля и первым элементом.

.....

G21 G42 p1

Z-10

l1

G21 G42 L5 последний элемент

Z-10

l1 первый элемент

.....

.....

Соединение геометрических элементов

Геометрические элементы профиля могут быть связаны между собой за счёт тангенциального сопряжения, пересечения или присутствия автоматического соединения или фаски.

Пересечение между элементами

В случае пересечения двух прямых, возможно только одно решение. В случае пересечения прямая-окружность или окружность-окружность всегда возможны два решения. Система автоматически выбирает первое, если необходимо получить второе, следует запрограммировать дискриминатор s2 после определения первого элемента.

Примеры пересечения прямая-окружность приведены на рисунке А.82. В случае пересечения прямой с окружностью первое и второе пересечения определяются направлением движения прямой линии. В случае пересечения окружности с окружностью, как показано на рисунке А.83, первое пересечение то, что слева от прямой, соединяю- щей центр первой окружности с центром второй, а второе пересечение то, что справа от той же прямой линии.

Соединения между элементами при помощи автоматического радиуса

Если элементы пересекаются, можно определить соединение между ними (прямые линии или окружности), программируя значение радиуса: положительное - в направлении против часовой стрелки, отрицательное - в направлении по часовой стрелке. Пример приведён на рисунке А.84.

Соединение r не может быть запрограммировано в кадре, следующем сразу же за кадром с G21 или же в кадре, предшествующем кадру с G20 (т.е. профиль не может начинаться и закончиться с соединения). В случае активизации компенсации радиуса инструмент размещается на пересечении двух геометрических элементов, смещённых радиусом инструмента. Если необходимо ввести радиус между двумя элементами, сл дует программировать нулевой радиус (r=0). Примеры приведены на рисунках А.84- А.85.

Скосы

Можно определить скосы между прямолинейными элементами, программируя значение скоса без знака, рассматриваемое как расстояние от точки пересечения. Пример приведён на рисунке А.86.

Скос не может быть запрограммирован в кадре, который непосредственно следует за кадром G21 или предшествует G20 (т.е. профиль не может начинаться или закончиться со скосом).

В геометрическом программировании GTL перемещения всегда осуществляются с рабочей подачей, для программирования быстрого перемещения необходимо программировать скорость рабочей подачи F с высоким значением.

Примеры программирования при помощи GTL

Рассмотрим пример:

примеру соответствует рисунок А.87:

N1 (UCG,2,Z-90Z5,X-5X125)

N2 p1=Z0 X50

N3 l1=p1,a180

N4 c1=I-40 J30 r20

N5 l2=Z-40 X30,a100

N6 p2=Z-85 X120

N7 l3=p2,a180

N8 T1.1 M6 S… F… M3

N9 G X100 Z10

N10 G21 G42 p1

N11 l1

N12 r-8

N13 c1

N14 r-10

N15 l2

N16 r8

N17 l3

N18 G20 G40 p2

N19 G X150 Z55

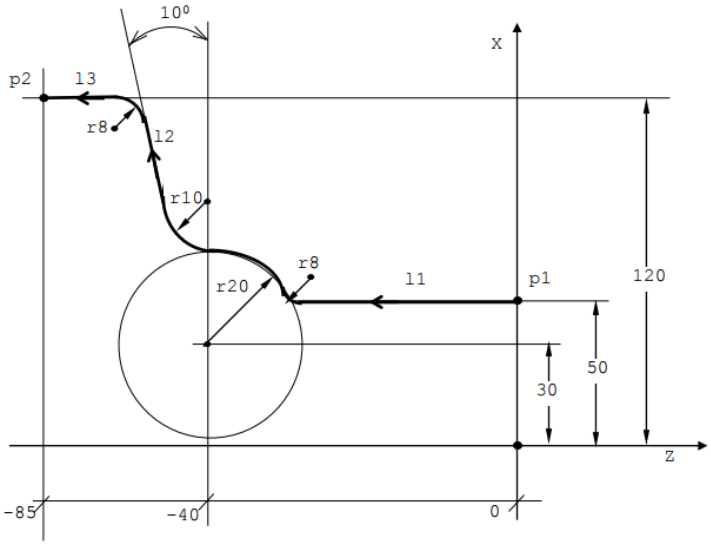


Рисунок А.87

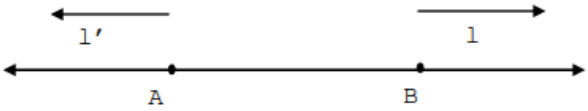
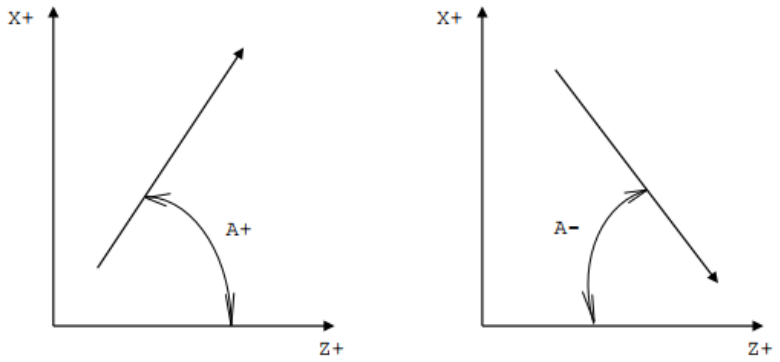
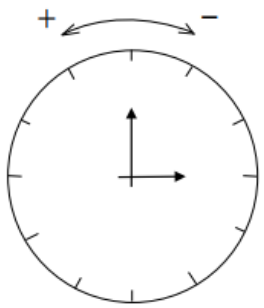


Рисунок А.25 – Представление прямой линии в векторной геометрии



а) для прямой линии



б) для окружности

Рисунок А.26 - Определение направления движения в геометрическом программировании

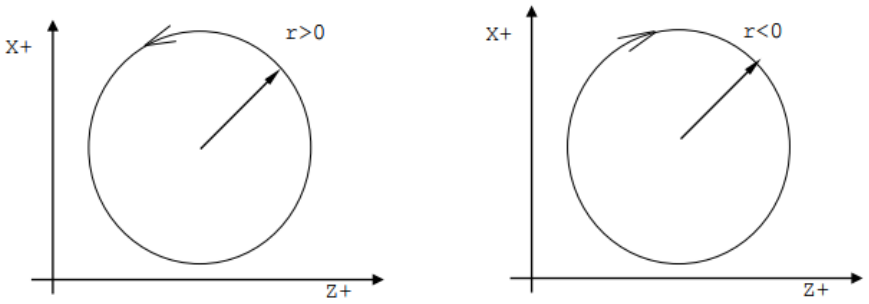


Рисунок А.27

p1=c1,l3

p2=c1,l3,s2

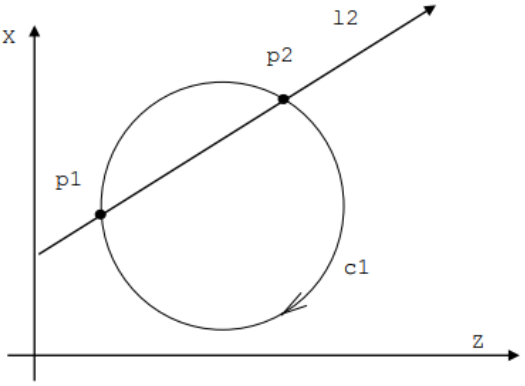


Рисунок А.28

o4=Z20X30a-45

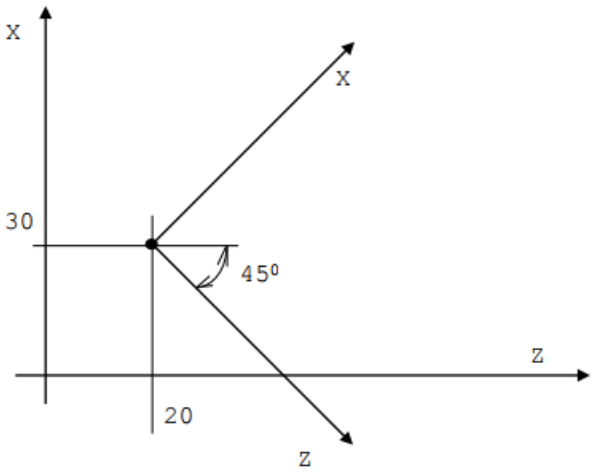


Рисунок А.29

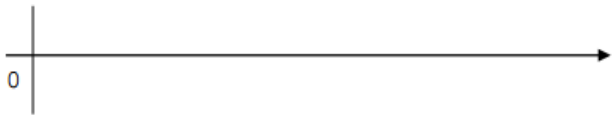


Рисунок А.30 - Полярные оси

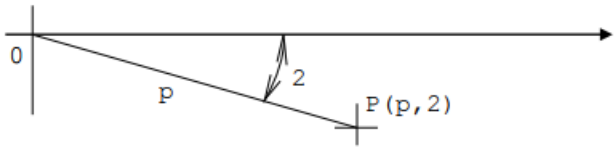


Рисунок А.31 - Полярные координаты

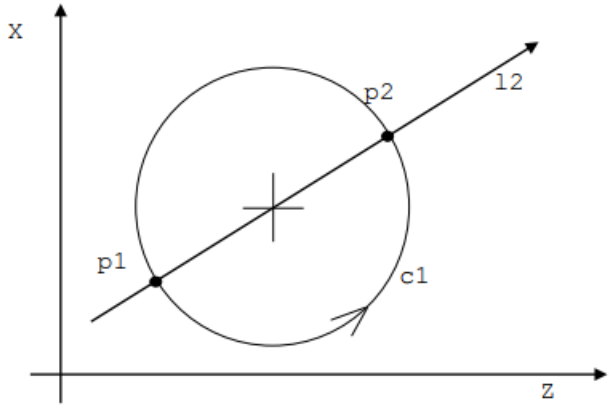


Рисунок А.32

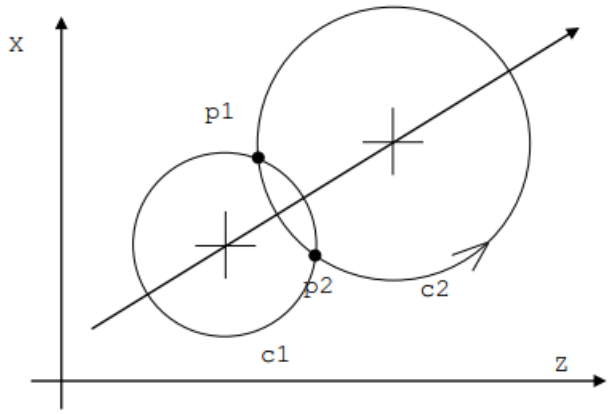


Рисунок А.33

p1=Z30X160

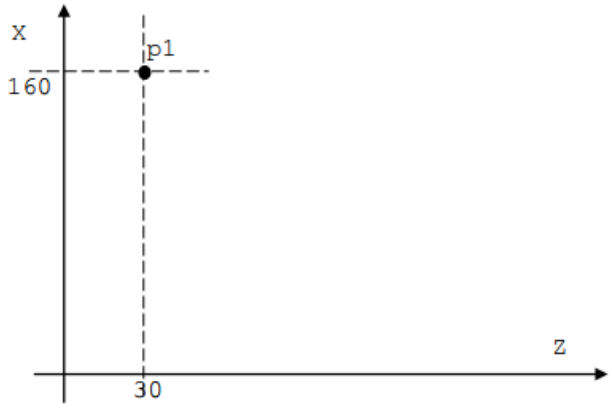


Рисунок А.34

o1=Z30X30a-20
p5=o1Z20X10

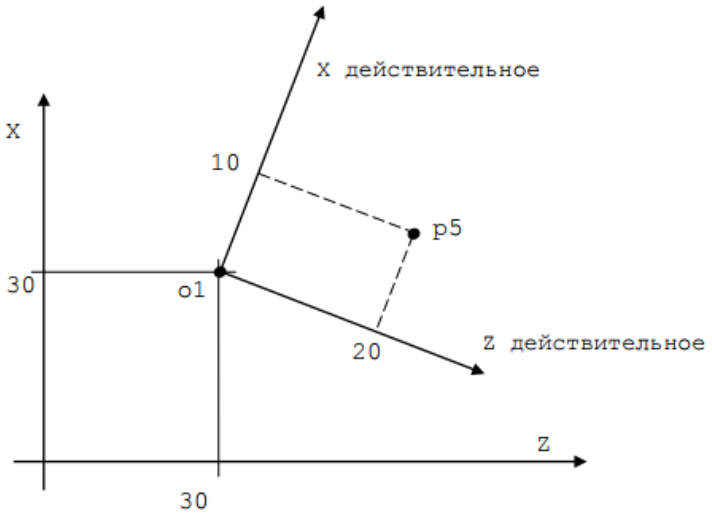


Рисунок А.35

p2=m55a60

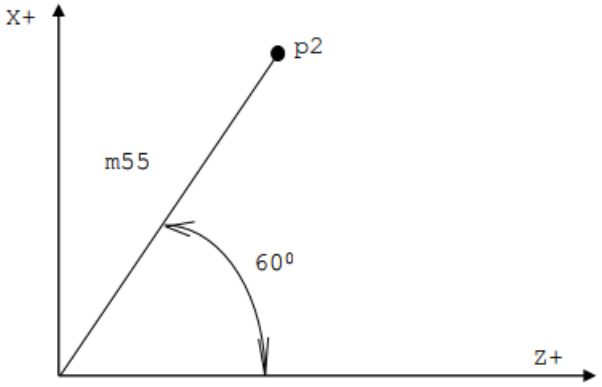


Рисунок А.36

p1=l1,l2

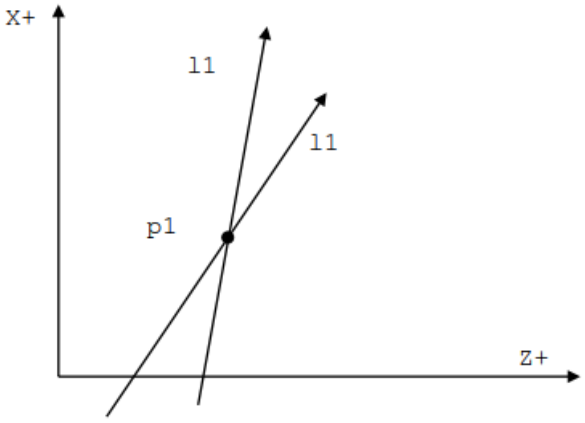


Рисунок А.37

p1=l4,c3
p2=l4,c3,s2
p1=-l4,c3,s2

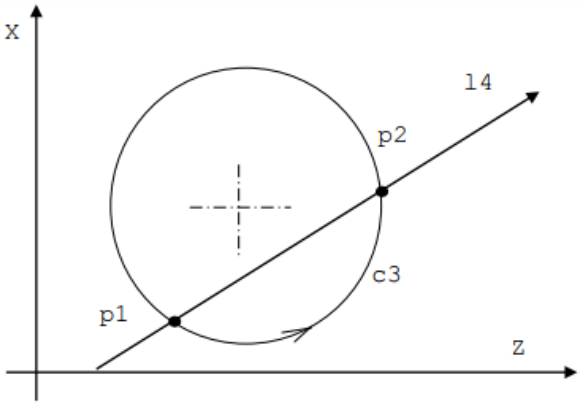


Рисунок А.38

p1=c1,c2
p2=c1,c2,s2
p1=c2,c1,s2

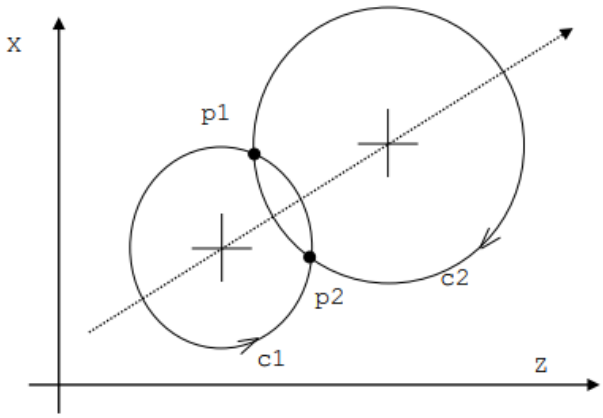


Рисунок А.39

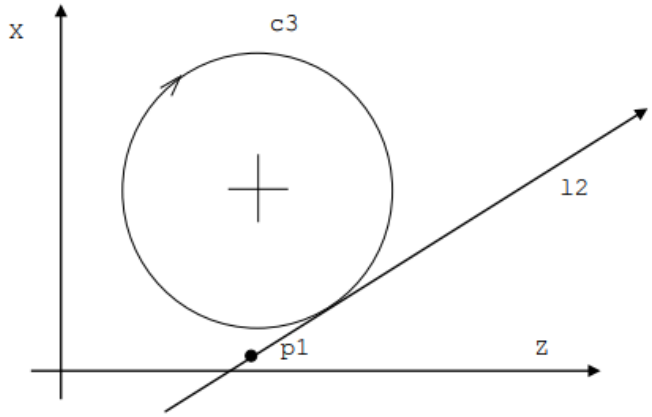


Рисунок А.40

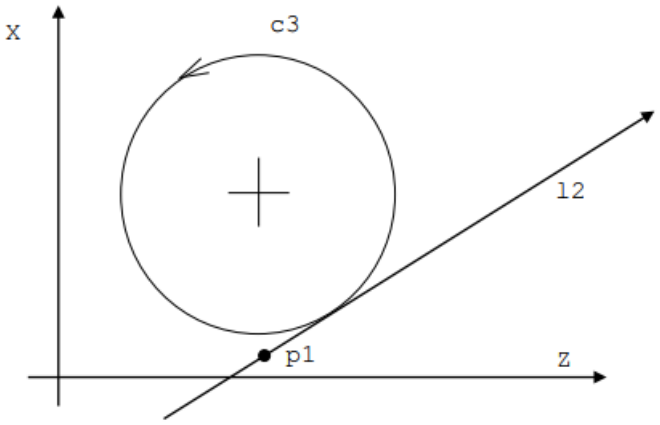


Рисунок А.41

```
l1=Z40X20,Z60X70
```

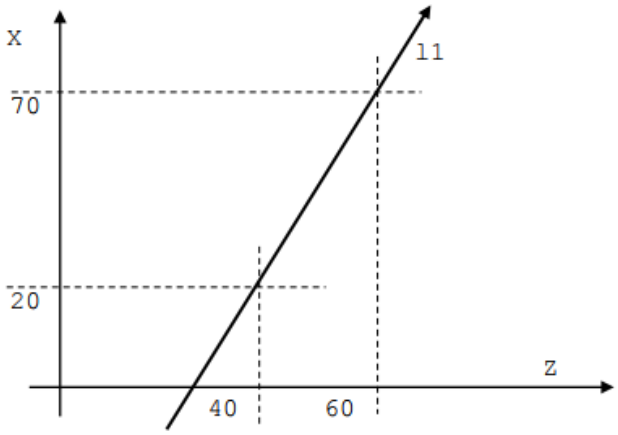


Рисунок А.42

```
l1=Z10X15,I45J30r-15
l2=Z10X15,I45J30r15
```

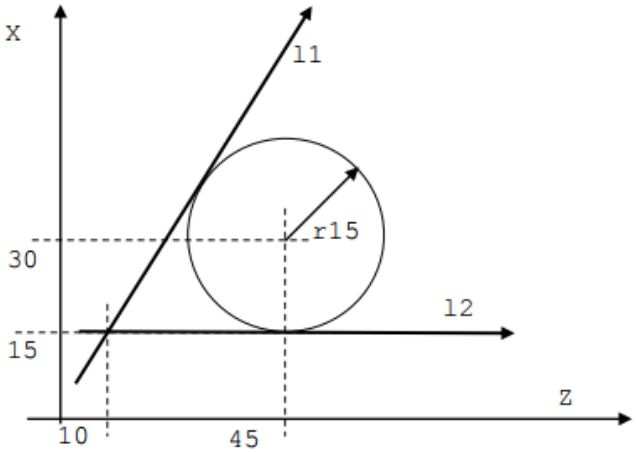


Рисунок А.43

l2=Z20X20,a-20

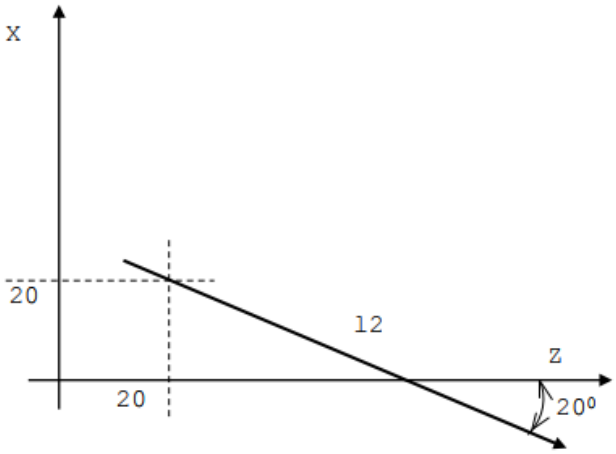


Рисунок А.44

o1=Z30X30a-40
l5=o1Z25X30,a60

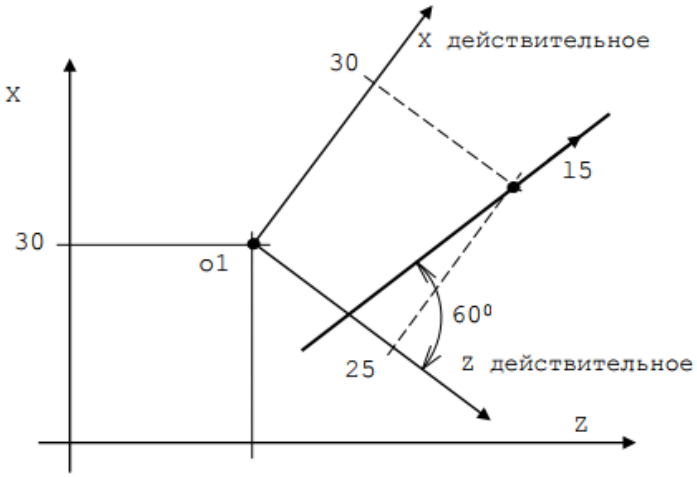


Рисунок А.45

l1=I60J80r20,a45

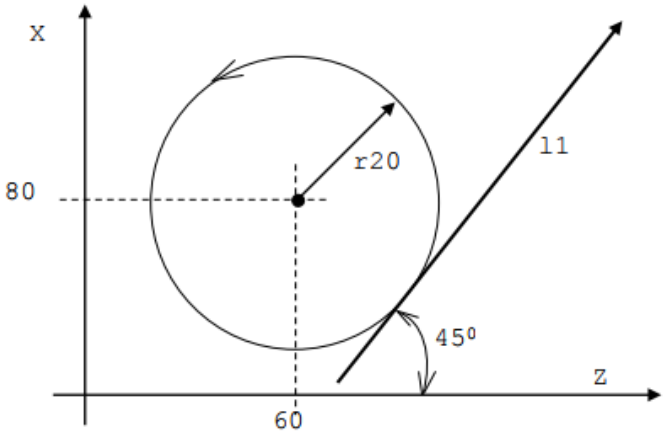


Рисунок А.46

o3=Z25X10a20
l4=o3I25J15r10,a115

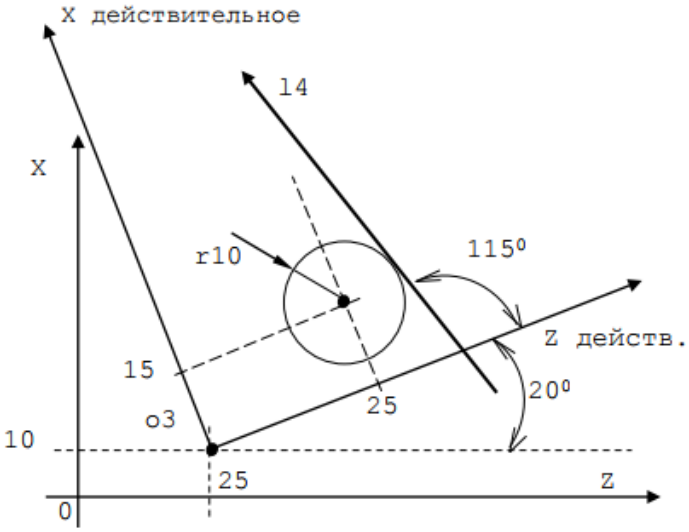


Рисунок А.47

l3=I25J35r-17,I70J20r13

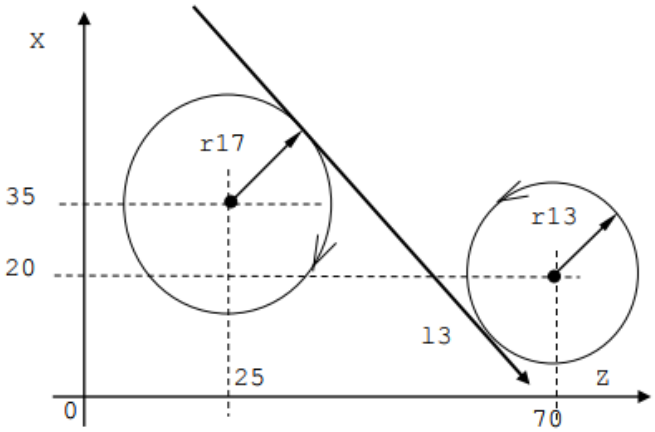


Рисунок А.48

l3=I25J35r17,I70J20r13

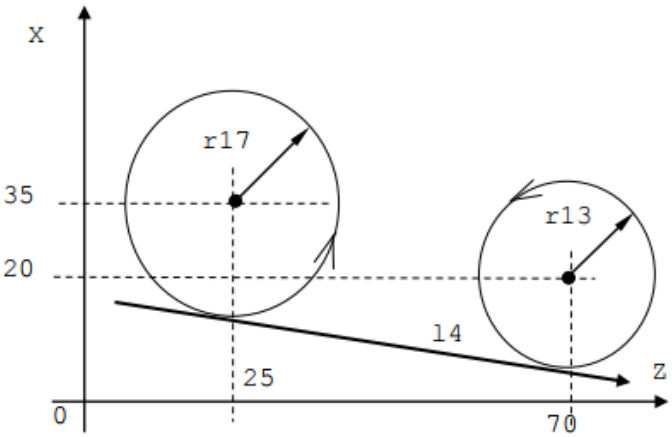


Рисунок А.49

l9=p7,p8

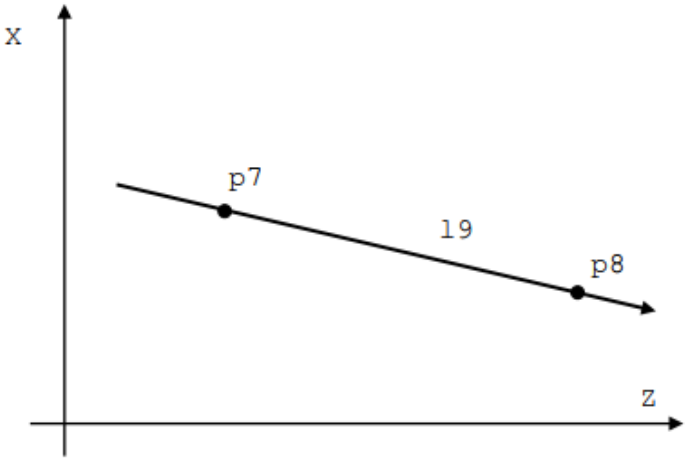


Рисунок А.50

l1=p1,c1
l2=p1,-c1

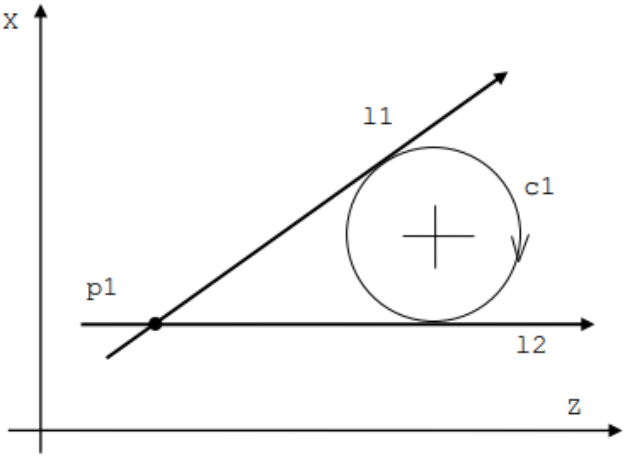


Рисунок А.51

l3=c1,c2

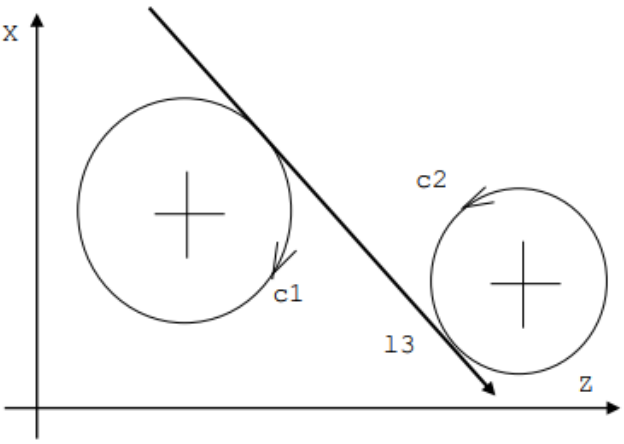


Рисунок А.52

l4=-c1,c2

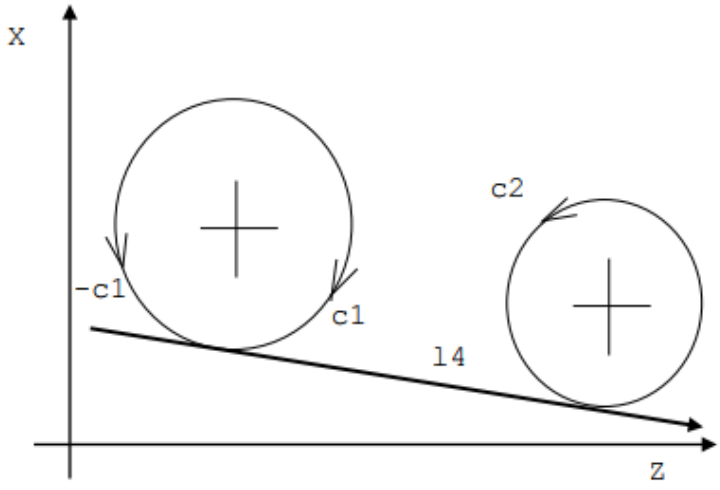


Рисунок А.53

l3=p1,a45

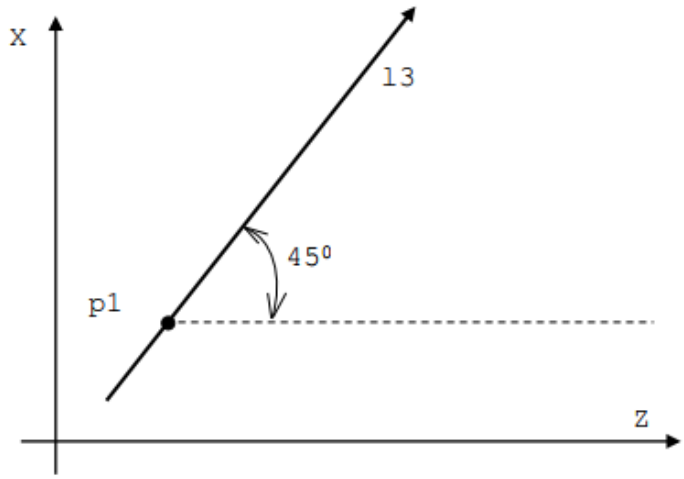


Рисунок А.54

l1=c1,a50
l2=-c1,a50

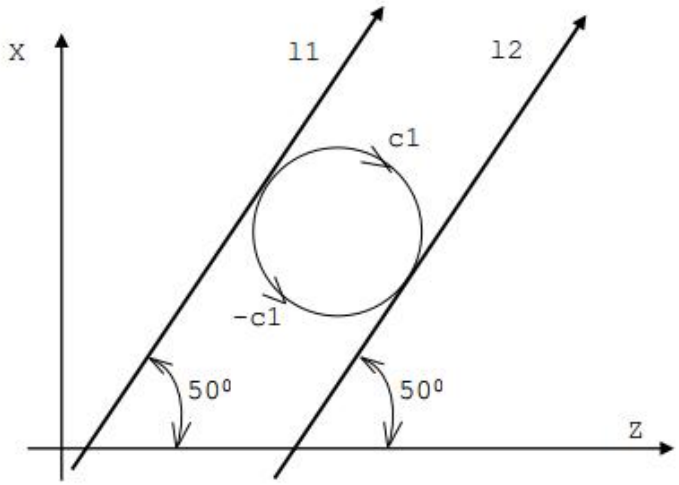


Рисунок А.55

12=11,d20
13=11,d-15

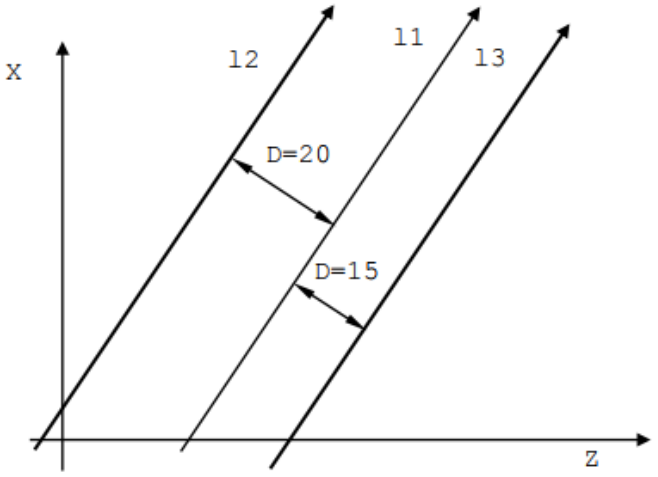


Рисунок А.56

12=-11,d-50

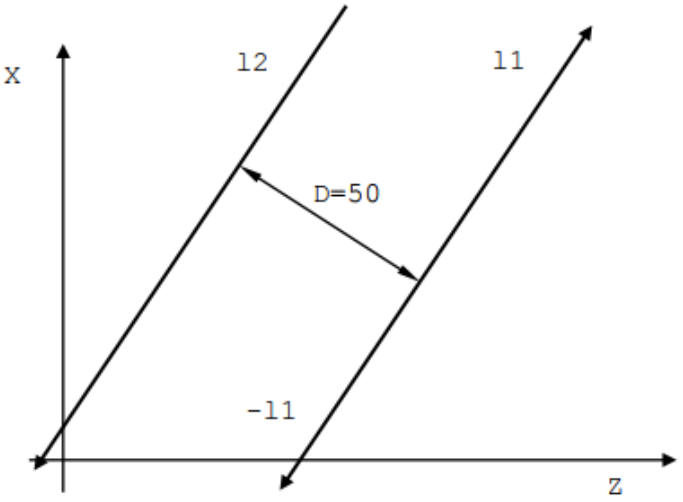


Рисунок А.57

Практическая работа №4

Тема: Разработка УП обработки деталей на фрезерном станке с использованием САП.

Цель занятия: Приобретение практических навыков разработки УП для обработки детали на фрезерном станке за использованием САП.

4. Материальное обеспечение

4.1. Инструкция к лабораторным занятиям

4.2. Чертежи (эскизы) деталей

4.3. Фрезерный станок ЧПУ типа NC-201M.

5. порядок выполнения работы.

5.1. Сообщение темы, плана, постановка цели урока

5.2. Индивидуальные задания для выполнения работы

5.3. Анализ чертежа (эскиза) детали

5.4. Разработка УП с применением САП.

5.5. Отработка УП.

3 Параметрическое программирование

Используя коды E, можно параметрически 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. Пара- метры E имеют различные индексы для переменных различного формата. Описание параметров E для различных форматов представлено в таблице 2.8.

Таблица 2.8 – Описание параметров E для различных форматов

Формат	Параметры	Мин/макс величина
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

Примечание – В таблице 2.8 (*)означает максимальное число параметров E, определённое во время конфигурации.

Параметры E получают значения в кадрах назначения.

Формат кадра назначения:

En = «выражение»,

где: «выражение» - может быть цифровой величиной или математическим выражением, результат которого будет запомнен под параметром E индекса n. «Выражение» - это математическое выражение, состав- ленное из арифметических операторов, функций и операндов (параметры E, числовые константы).

Арифметические операции:

1) + сложение;
2) - вычитание;
3) * умножение;
4) / 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 индекса B (1,2,3) из геометрического элемента линии (прямой) индекса A (1 = синус угла, 2 = косинус, 3 = расстояние от прямой линии до начальной точки);
- FEP (A,B) - извлекает элемент индекса B (1,2) из геометрического элемента точки индекса A (1 = абсцисса точки, 2 = ордината);
- FEC (A,B) - извлекает элемент индекса B (1,2,3) из геометрического элемента окружности индекса A (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 Е, указанный слева от знака «=».

Примеры

E30 = FEL(5,1) придаёт E30 значение синуса угла, который образует прямая l5 с абсциссой.

E34 = FEP(4,2) придаёт E34 значение ординаты точки p4.

E42 = FEC(8,3) придаёт E42 значение радиуса окружности c8.

Значения вычисления параметров:

N1 E37=(E31*SIN(E30)+123.4567)/SQR(16) - выполняет математическое решение выражения и придает результат параметру E37.

"LAB1"E51=-0.00000124+5 - выполняет вычисление выражения и придает результат параметру E51.

E40=TAN(35) - извлекает тангенс 35 градусов и придает результат параметру E40.

/E35=FEP(37,1) - извлекает абсциссу точки p37, ранее занесенной в память и придает значение параметру E35.

E31=NEG(E31) - меняет знак параметра E31. E7=81 - придаёт значение параметру E7.

E25=E25+30 - новым значением параметра E25 будет сумма константы 30 и текущего значения параметра E25.

E2=SK396 - придаёт E2 содержимое байта 396 пакета K.

E8=SYVAR1 - придаёт E8 значение переменной SYVAR1.

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

Параметры Е могут быть использованы как внутри программы, так и внутри под- программы и могут быть воспроизведены.

Пример

(DIS,E54) - воспроизводит на экране величину E54=...

Обычно параметры остаются запомненными при выключении станка, могут быть приведены к нулю, если это предусмотрено в фазе характеризации. Использование параметров E сведено в таблицу 2.9.

Таблица 2.9 - Использование параметров E

Параметры -(Формат)	Данные (геометрические технологические)	Примеры программирования
E0..E9 (BY)	функции G функции M коды RPT	GE1 ME3 (RPT,E9)
E10..E19 (IN)	функции T	TE14.E15
E20..E24 (LI)		
E25..E29 (RE)	функции F коды URT коды SCF индексированные оси код UGF	FE27 (URT,E25) (SCF,E26) PE29 UGF=E28
E30...(*) (LR)	координаты осей C X Z координаты R составные операции IJK глобальные переменные системы: TMR UOV	XE32 RE33 KE34 TMR=E38 UOV=E40

Кадры с трёхбуквенными операторами

Этот раздел описывает функциональность и синтаксис кадров, имеющих в качестве операторов трёхбуквенные коды. Возможно, установить семь классов трёхбуквенных операторов:

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

Трёхбуквенные операторы, модифицирующие систему отсчёта осей

Ими являются все операторы, позволяющие изменять в плоскости декартовую систему отсчёта, по отношению, к которой был запрограммирован профиль. К этому классу принадлежат следующие операторы: UAO,UOT,UIO,MIR,URT,SCF,RQO.

Использование абсолютных исходных точек - UAO

Выбирает одну из абсолютных исходных точек, ранее определенных командой ORA.

Формат:

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

где: N - определяет номер исходной 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 исходная точка «n». Для необъявленных осей остаётся в силе текущая исходная точка. Если «название оси» не присутствует, исходная точка «n» приводится в действие для всех осей, для которых была объявлена эта исходная точка.

Пример

(UAO,1) - абсолютная исходная точка 1, приведенная в действие для всех осей.

..... - программа, отнесенная к исходной точке 1 для всех осей.

(UAO,2,Z,X) - абсолютная исходная точка 2, приведенная в действие для осей Z и X.

(UAO,3,B) - абсолютная исходная точка 3, приведенная в действие для оси B.

..... - программа, отнесенная к исходной точке 2 для осей ZX, к точке 3 для оси B и к точке 1 для всех остальных осей.

(UAO,O) - возвращение в нулевую начальную точку для всех осей.

При включении и после команды «СБРОС» автоматически приводится в действие ну- левая исходная точка для всех осей. Максимально могут присутствовать 8 «названий осей». Не могут быть определены одинаковые «названия осей». Если требуется при- вести в действие различные исходные точки для различных осей, необходимо программировать столько кадров с

этими операторами, сколько имеется исходных точек. Если выбранная начальная точка (-n) загружена в файл альтернативной системы измерения, она автоматически переводится в текущую систему измерения.

Определение и использование временных исходных точек - UOT

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

Формат:

(UOT,n,VAR-1 [,VAR-2]),

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 исходной точке для той оси. Для необъявленных осей остаётся в силе текущая начал ная точка.

Пример

(UAO,O) - активизируется абсолютная исходная точка 0. Программа, отнесенная к абсолютной исходной точке 0 для всех осей:

1) (UOT,O,Z100,X100) ..... применяется временная исходная точка к исходной точке 0 с корректировками Z100 и X100 (временная исходная точ- ка).
2) (UOT,1.Z-250,X-50) ..... применяет временная исходная точка к абсолютной исходной точке 1 с корректировками Z-250 и X-50. (UAO,O) активизируется абсолютная исходная точка 0 для всех осей.

Пример приведён на рисунке А.89.

По крайней мере, должен присутствовать один операнд оси. Максимально могут присутствовать 8 осей. Не могут быть определены операнды осей с одним и тем же названием. Временная исходная точка остаётся активной до того, как определяется новая временная исходная точка или до вызова абсолютной исходной точки, или до команды «СБРОС». Размер в операторе UOT необходимо программировать в текущей размерности (G70/G71).

Определение и использование исходных точек по приращениям - UIO

Эта команда позволяет инкрементально переместить текущую исходную точку для всех программированных в команде осей.

Формат:

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

где VAR-i - представляет ось и размер. Система берет размер как абсолютное смещение и прибавляет его к абсолютной исходной точке для данной оси. Для необъявленных осей текущая исходная точка остается в силе.

Пример приведён на рисунке А.90.

Инкрементальная исходная точка остаётся в силе до её переопределения с новой командой UIO, или восстанавливается абсолютная исходная точка при помощи (UAO,O) или [СБРОС].

Если команда UIO относится к оси X, то программируемое значение является радиальным, а не диаметральным.

Зеркальная обработка - MIR

Оператор MIR инвертирует запрограммированные направления перемещений, объявленных в операторе. Для необъявленных осей предыдущая функция MIR остаётся 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,Z)

.....

N42 (MIR,Z,X)

.....

N84 (MIR,X)

.....

N99 (MIR)

Зеркальная обработка на программированную ось начинает действовать с первого движения данной оси после команды MIR. Инверсия осуществляется вокруг текущей начальной точки. Максимально может быть запрограммировано 8 осей. Не представляется возможным программировать 2 раза одну и ту же ось. Если присутствуют коман- ды вращения (URT) и зеркальной обработки (MIR), то они устанавливаются в следую- щем порядке: MIR и URT.

Поворот плоскости - URT

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

Формат:

(URT , ОПЕРАНД) ,

где: ОПЕРАНД - представляет величину угла, выраженную в градусах и десятичных долях градуса; может быть выражен явно или неявно (параметр E типа от E25 до E29). Если операндом является «0», то 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), то они устанавливаются в следующем порядке: MIR и URT.

Пример программирования для повторения профиля (8 раз) при виртуальных осях (рисунок А.93):

N1 (DIS,"GTL WITH ROTATION")

N2 M21

N3 G0 G94 G97 C0 X120 Z5

N4 (UAV,1,XC,UV,30)

N5 (DPI,U,V)

```
N6 F.. S.. T2.2 M6
N7 UOV=2
N8 p1=U50 V0
N9 C1=I0 J0 r50
N10 C2=I0 J0 r10
N11 l1=C2,a180
N12 l3=U0 V0,a45
N13 l2=C2,a45
N14 p2=l3,C1,S2
N15 U60 V0
N16 Z-10 "START"
N17 E25=0
N18 (RPT,8)
N19 (URT,E25)
N20 G21 G42 p1
N21 c1
N22 r3
N23 l1
N24 r-3
N25 l2
N26 r3
N27 c21
N28 G20 G40 p2
N29 E25=E25+45
N30 (ERP)
```

N31 (URT,0)

"END" N32

N33 UOV=0

N34 (EPP,START,END)

N35 (UAV,0)

N36 (DPI,Z,X)

N37 G0 M20

N38 Z5

N39 G79 X Z M30

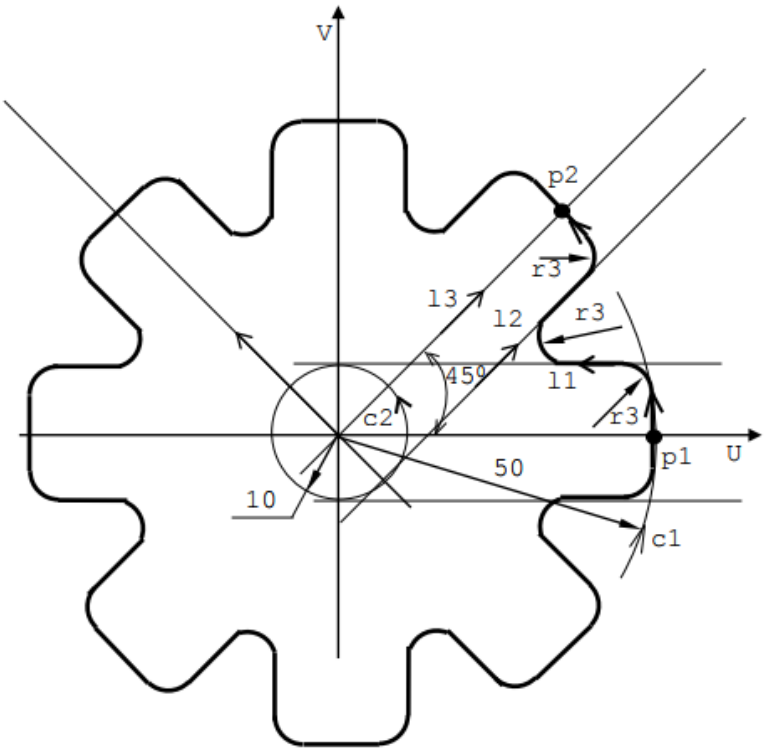


Рисунок А.93