

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кузбасский государственный технический университет
имени Т. Ф. Горбачева»

Кафедра металлорежущих станков и инструментов

Составитель
Владимир Олегович Коротин

ПРОГРАММИРОВАНИЕ И НАЛАДКА СТАНКОВ С ЧИСЛОВЫМ ПРОГРАММНЫМ УПРАВЛЕНИЕМ

Методические материалы к лабораторным работам

Рекомендовано учебно-методической комиссией направления
подготовки 15.03.05 Конструкторско-технологическое
обеспечение машиностроительных производств
в качестве электронного издания для использования
в образовательном процессе

Кемерово 2025

Рецензенты: Коротков А. Н. – доктор технических наук, зав. кафедрой металлорежущих станков и инструментов ФГБОУ ВО «Кузбасский государственный технический университет имени Т. Ф. Горбачева»

Коротин Владимир Олегович

Программирование и наладка станков с числовым программным управлением : методические материалы к лабораторным работам для обучающихся направления подготовки 15.03.05 Конструкторско-технологическое обеспечение машиностроительных производств всех форм обучения / Кузбасский государственный технический университет имени Т. Ф. Горбачева, Кафедра металлорежущих станков и инструментов ; составитель В. О. Коротин. – Кемерово : КузГТУ, 2025. – 1 файл (1499 Кб). – Текст : электронный.

Приведено содержание лабораторных работ, порядок их оформления, а также материал, необходимый для успешного изучения дисциплины. Назначение издания – помощь обучающимся в получении знаний по дисциплине «Программирование и наладка станков с числовым программным управлением» и организация лабораторных работ.

©Кузбасский государственный
технический университет
имени Т. Ф. Горбачева, 2025
© Коротин В.О., составление,
2025

ОГЛАВЛЕНИЕ

Лабораторная работа №1 Проектирование траектории движения инструмента на фрезерном станке с ЧПУ	4
Лабораторная работа №2 Расчет координат опорных точек при подготовке управляющих программ	15
Лабораторная работа №3 Ручная подготовка УП	19
Лабораторная работа №4 Редакторы УП и проверка кода	34
Рекомендуемая литература	45

Лабораторная работа №1

Проектирование траектории движения инструмента на фрезерном станке с ЧПУ

Цель работы: Изучение основных принципов и получение практических навыков проектирования технологических операций обработки на фрезерных станках с ЧПУ.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Процесс разработки технологических процессов обработки деталей с использованием оборудования с числовым программным управлением связан с необходимостью учета большого числа факторов, основным из которых является детальное описание траектории движения инструмента в процессе обработки.

Разработка траектории движения инструмента осуществляется на основе рабочего чертежа детали, с учетом технологических и конструктивных возможностей используемого станка и режущего инструмента.

Так помимо сведений о максимально возможных габаритах обрабатываемых деталей или мощности приводов станка при построении траектории движения инструмента ключевую роль играют системы координат, которые используются в том или ином оборудовании. Главной расчетной системой, определяющей перемещения исполнительных органов станка, является система координат станка, общий вид которой представлен на рисунке 1.1.

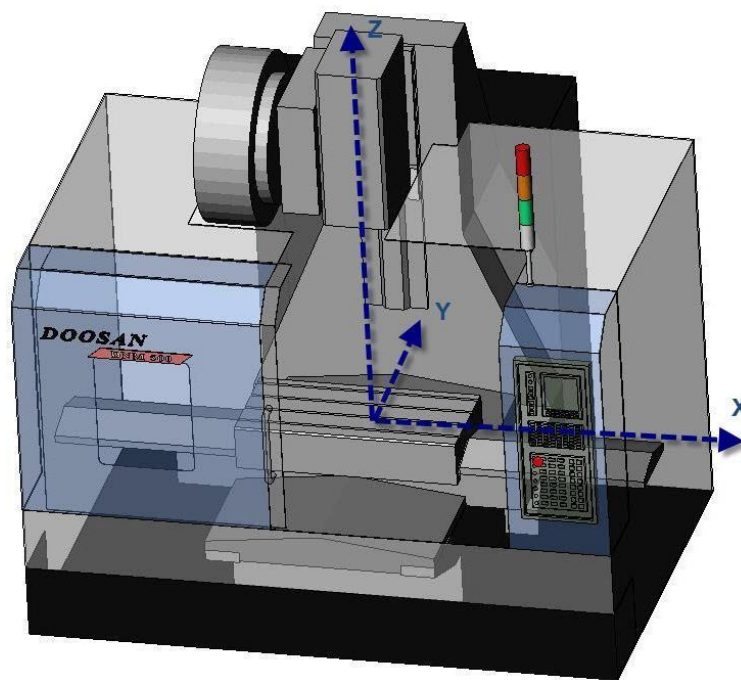


Рис. 1.1. Общий вид системы координат фрезерного станка с ЧПУ

Из рисунка 1.1 видно, что оси координат располагают параллельно направляющим станка, положительные направления осей определяются по правилу «правой руки», а сама система координат называется правой, и является стандартной для современных станков с ЧПУ.

По правилу «правой руки» большой палец указывает положительное направление оси X, указательный – оси Y, средний на положительное направление оси Z. В качестве положительного направления оси Z принимаю вертикальное направление вывода инструмента (например, сверла) из заготовки, то есть ось Z всегда связана со шпинделем станка. При этом ось X перпендикулярна оси Z и параллельна плоскости рабочего стола. Иллюстрация данного правила представлена на рисунке 1.2.

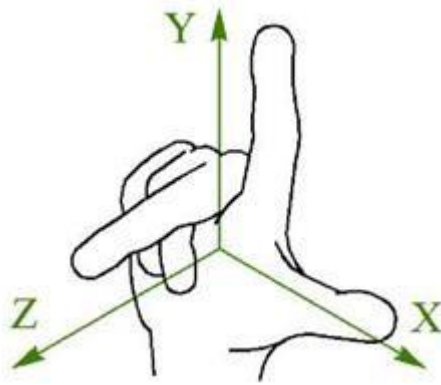


Рис. 1.2. Иллюстрация правила «правой руки»

Кроме того правило «правой руки» позволяет определить положительное направление вращения вокруг оси, для чего большой палец выставляется в положительном направлении оси, а согнутые пальцы показывают положительное направление вращения, пример чего представлен на рисунке 1.3.

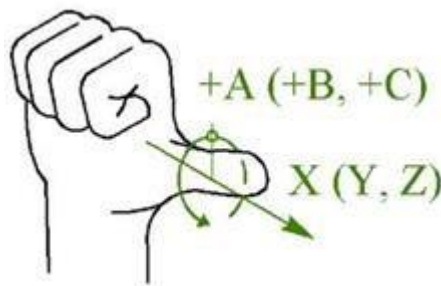


Рис. 1.3. Определение положительного направления вращения

Система координат каждого станка содержит в себе ряд ключевых точек, к которым относятся:

Нулевая точка станка – точка, принятая за начало системы координат станка, физическая величина, устанавливаем производителем станка при помощи концевых выключателей или датчиков. Как правило, на иллюстрациях обозначается буквой «М».

Исходная точка станка – точка, определенная относительно нулевой точки станка и используемая для начала работы по управляющей программе на иллюстрациях обозначается буквой «R».

Фиксированная точка станка – точка, определенная относительно нулевой точки станка и используемая для определения положения рабочего органа станка. Могут быть представлены в двух видах. Фиксированная точка стола в виде отверстия в столе станка (обозначается буквой «F»). Фиксированная точка шпинделя в позиции инструмента в виде пересечения оси шпинделя и его торцевой поверхности (обозначается буквой «N»).

Пример расположения указанных точек представлен на рисунке 1.4, однако для каждого конкретного станка положение ключевых точек необходимо уточнять в паспорте.

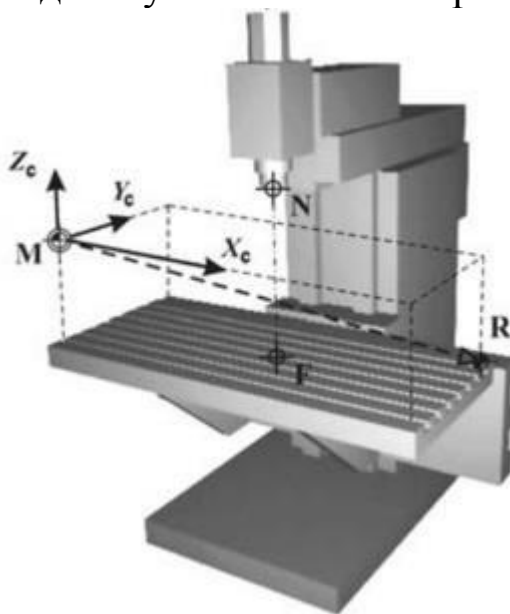


Рис. 1.4. Ключевые точки системы координат станка

В рамках системы координат станка системы числового программного управления позволяют создавать рабочие системы координат, относительно которых строится траектория перемещения инструмента и рассчитываются координаты опорных точек. Начало рабочей системы координат обозначается буквой «W», пример чего показан на рисунке 1.5, как правило, данная точка является нулевой точкой детали, относительно которой производится определение размеров детали и построение траектории.

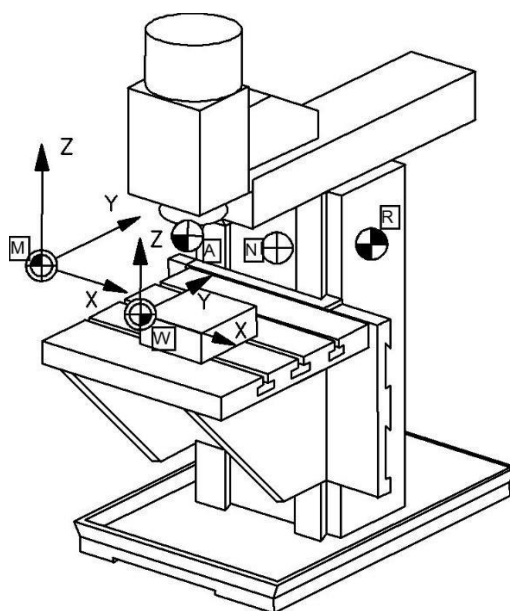


Рис. 1.5. Рабочая система координат

Наиболее часто встречаемые условные обозначения указанных точек представлены в таблице 1.1.

Таблица 1.1

Условные обозначения ключевых точек

Условное обозначение	Буквенное обозначение	Название	Пример
	M	Нулевая точка станка	
	R	Исходная точка станка	
	W	Нулевая точка детали	
	N	Фиксированная точка шпинделя в позиции инструмента	
	F	Фиксированная точка стола в центре базового отверстия	

	Ps	Точка начала обработка конкретной заготовки (исходная точка программы)	
--	----	---	--

Определение положения рабочей системы координат осуществляется с учетом геометрии детали или заготовки, способа ее базирования в приспособлении на столе станка и последовательности обработки.

Рассмотрим процесс построения траектории обработки детали пластика, эскиз которой показан на рисунке 1.6.

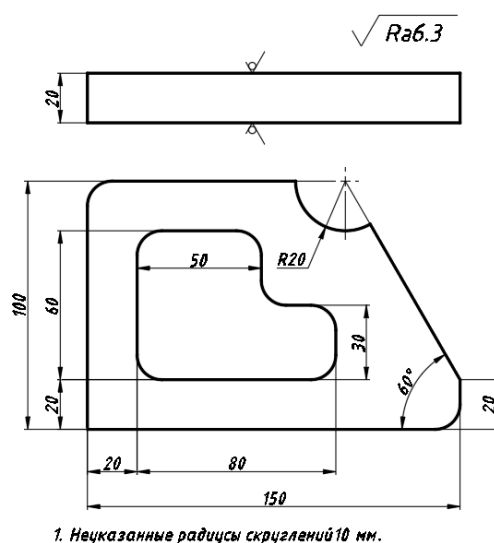


Рис. 1.6. Эскиз детали «Пластина»

На начальном этапе производится изучение конструкторской документации и определяется схема базирования детали. В рассматриваемом случае схема базирования представлена на рисунке 1.7. На основе схемы базирования детали производится назначение нулевой точки детали (начала рабочей системы координат).

В общем случае при назначении нулевой точки детали следует руководствоваться двумя правилами.

1. Нулевая точка программы должна совпадать с конструкторской базой. Это значит, что если на чертеже размеры стоят от левого верхнего угла детали, то лучше, если именно в этом углу и будет находиться нулевая точка. А если размеры

указываются от центрального отверстия, то нулем детали следует назначить центр этого отверстия.

2. Вторым правилом является удобство программирования. Например, если расположить деталь в первом квадранте прямоугольной системы координат, то это немного упростит процесс расчета траектории из-за того, что все опорные точки этой детали будут описываться положительными координатами.

Из рисунка 1.7 видно, что в качестве нулевой точки детали целесообразно принять левый нижний угол детали. Пример, эскиза детали с нанесенной нулевой точкой детали и осями рабочей системы координат показан на рисунке 1.8.

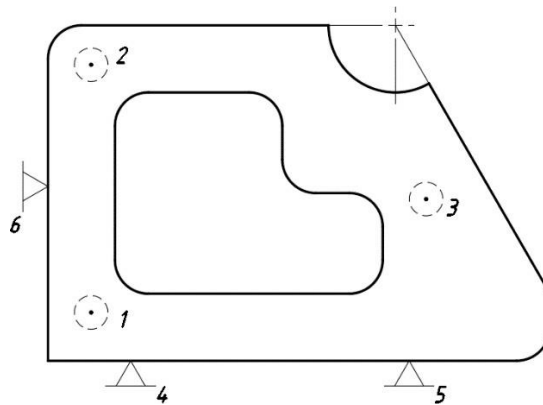


Рис. 1.7. Схема базирования детали

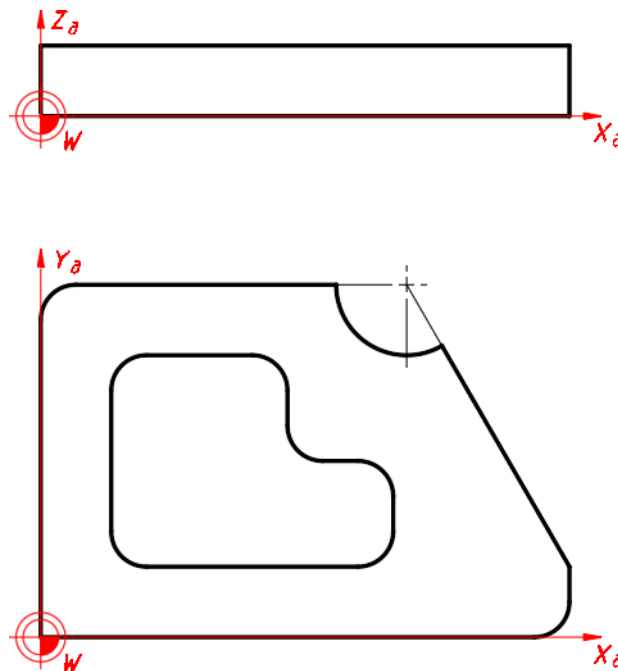


Рис. 1.8. Нулевая точка детали

Следующим этапом производится определение последовательности обработки, которая определяется конфигурацией детали и заготовки, но в общем случае предпочтительна следующая последовательность: обработка сначала внутренних контуров и примыкающих к ним плоскостей, а после обработка внешних контуров и примыкающих к ним плоскостей.

Также при построении траектории обработки необходимо учитывать геометрические параметры режущего инструмента, который в случае контурной обработки криволинейных профилей представляет собой концевую фрезу, диаметр которой выбирается исходя из условий жесткости и минимального размера радиуса скругления внутренних углов. В рассматриваемом случае принимается концевая фреза диаметром 16 мм.

Предположим, что заготовка пластины получена литьем и имеет вид, представленный на рисунке 1.9.

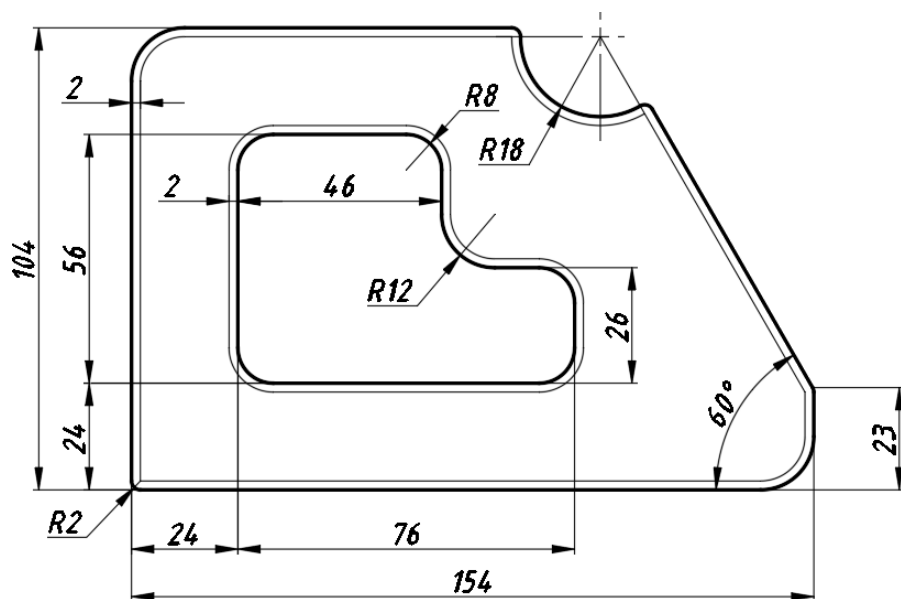


Рис. 1.9. Заготовка детали «Пластина»

Траектория движения инструмента при контурной обработке представляет собой эквидистанту, состоящую из элементарных технологических участков в виде отрезков и/или дуг, начинающихся и заканчивающихся в опорных точках. Размер смещения эквидистанты от исходного контура обработки

определяется размером инструмента.

Помимо рабочих движений инструмента при построении траектории необходимо учитывать еще и вспомогательные перемещения, при проектировании которых следует руководствоваться следующими рекомендациями:

- подвод и отвод инструмента от обрабатываемой поверхности осуществляется по специальным траекториям, обеспечивающим врезание по касательной со своевременным переходом с холостого хода на рабочий;

- остановка или резкое изменение подачи в процессе резания недопустимы, из-за возможности повреждения обрабатываемой поверхности, перед остановкой или резким изменением подачи, подъемом или опусканием инструмента его необходимо отвести от поверхности под малым углом или по касательной;

- длина холостых перемещений должна быть минимальной;

- расстояние между соседними проходами инструмента следует выбирать с учетом перекрытия, равного 10 % диаметра фрезы.

В рассматриваемом примере траектория обработки внутреннего и внешнего контура примет вид, показанный на рисунке 1.10.

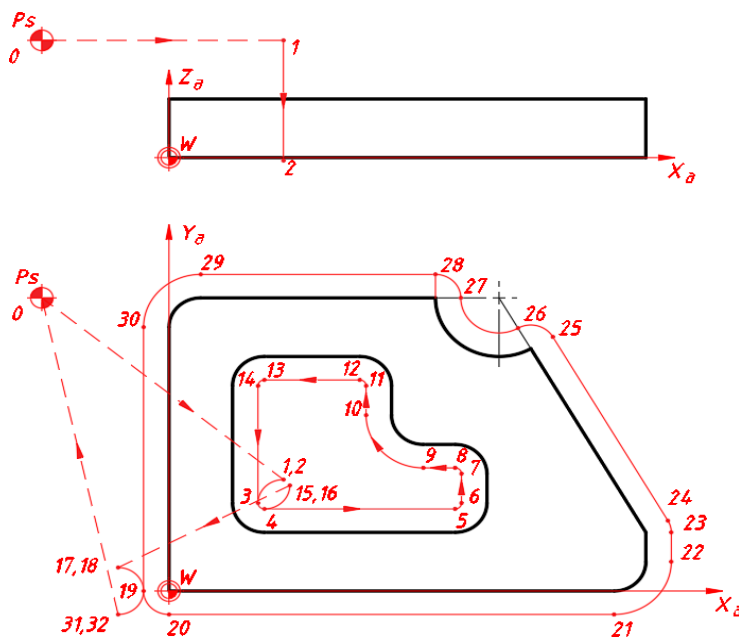


Рис. 1.10. Траектория обработки контуров детали

Из рисунка 1.10 видны следующие циклы обработки: подвод фрезы в точку начала программы Ps (0) на холостом ходу; перемещение на холостом ходу в точку 1; переход на рабочий ход при движении к опорной точке 2; выполнение контурной обработки в следующей последовательности: 2-3-4-5-6-7-8-9-10-11-12-13-14-3-4-15; подъем фрезы на безопасную высоту к точке 16; перемещение фрезы на холостом ходу к точке 17; переход на рабочий ход при движении к точке 18; выполнение контурной обработки в следующей последовательности: 18-19-20-21-22-23-24-25-26-27-28-29-30-19-31; подъем фрезы в точку 32 и возврат к точке начала программы Ps (0) на холостом ходу.

При построении траектории перемещения инструмента в расчетно-технологических картах (РТК) помимо условного обозначения ключевых точек имеется ряд обозначений позволяющих облегчить чтение.

Наиболее часто встречаемые условные обозначения для оформления РТК представлены в таблице 1.2.

Таблица 1.2

Условные обозначения в РТК

Наименование	Обозначение
Точка останова для смены инструмента	
Контрольная точка	
Точка вертикального подъема фрезы на 20 мм.	
Точка вертикального опускания фрезы на 15 мм.	
Перемещение фрезы с одновременным подъемом	
Перемещение фрезы с одновременным опусканием	
Траектория рабочего хода фрезы	
Траектория холостого хода	
Опорная точка с номером	

ЗАДАНИЯ

Разработать траекторию движения инструмента для детали, представленной в качестве задания согласно варианту.

Результаты проектирования траектории оформить в виде расчетно-технологической карты (РТК), в которой использовать условные обозначения, представленные в таблицах 1.1 и 1.2. В РТК отразить следующую информацию:

1. Изображение детали в прямоугольной системе координат с указанием размеров, направления осей и нулевой точки детали, точки начала обработки.

2. Схему базирования детали при обработке.

3. Траекторию движения центра инструмента, выполненную тонкими линиями отдельным цветом с указанием направления движения инструмента.

4. Опорные точки (отделяющие друг от друга отдельные участки траектории) и технологические точки (точки останова для измерения частоты вращения шпинделя, контроля размеров, точки изменения подачи).

СОДЕРЖАНИЕ ОТЧЕТА

В качестве отчета по лабораторной работе студент предоставляет расчетно-технологическую карту, выполненную с использованием системы автоматизированного проектирования в виде файла в нативном формате и PDF документа.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Дайте определение понятию нулевая точка станка.
2. Дайте определение понятию исходная точка станка.
3. Дайте определение понятию фиксированная точка станка.
4. Объясните принцип определения осей координат станка по правилу «правой руки».

5. Объясните принцип определения положительного направления вращения вокруг осей координат по правилу «правой руки».

6. Дайте определение понятию опорная точка.

Лабораторная работа №2

Расчет координат опорных точек при подготовке управляющих программ

Цель работы: Изучение основных методов определения координат опорных точек при подготовке информации для создания управляющих программ.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Создание управляющих программ основывается на использовании координат опорных точек траектории, определение которых может производиться двумя способами: аналитическим или графическим.

Аналитический метод определения координат основан на использовании несложных арифметических выражений, методов тригонометрии или аналитической геометрии.

Рассмотрим процесс определения координат опорных точек на примере точек траектории, представленной на рисунке 2.1. Так представленная траектория построения с учетом использования концевой фрезы диаметром 16 мм.

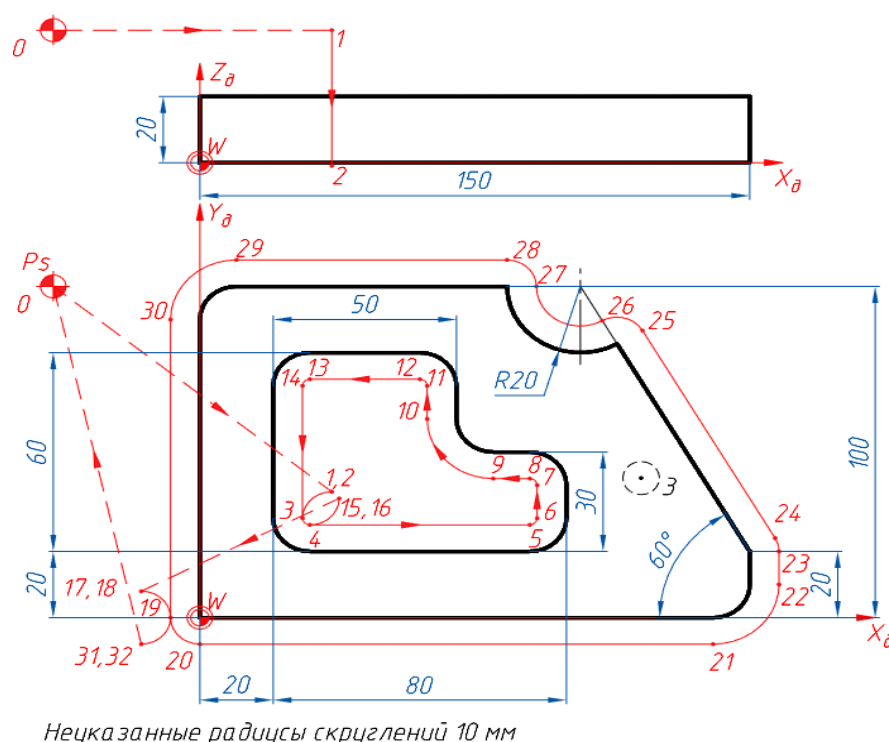


Рис. 2.1. Траектория обработки детали пластина

Координаты точки 0 (Точка начала обработки) задаются пользователем с учетом конструктивных особенностей станка и соображений безопасного подвода и отвода инструмента в рассматриваемом случае ее координаты: $X_0 = -40$; $Y_0 = 100$; $Z_0 = 40$.

Координаты X , Y опорной точки 3 можно определить при помощи арифметических выражений:

$$X_3 = 20 + R_{\phi} = 20 + 8 = 28 \text{ мм},$$

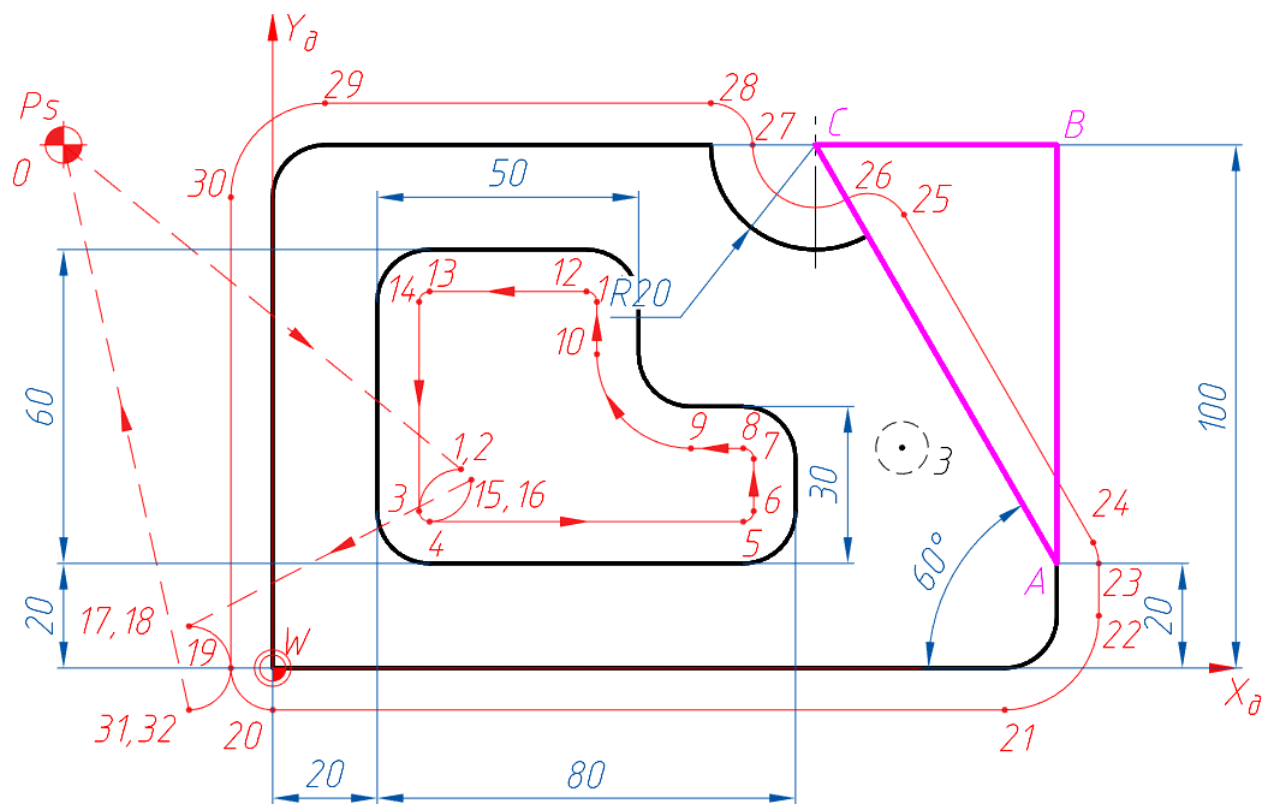
где 20 – размер до внутреннего контура детали по оси X , мм;
 R_{ϕ} – радиус фрезы, на который произведено смещение инструмента относительно контура детали, мм.

$$Y_3 = 20 + 10 = 30 \text{ мм},$$

где 20 – размер до внутреннего контура детали по оси X , мм;
 10 – радиус скругления, указанные в технических требованиях, мм.

Координата опорной точки 3 по оси Z , не требует расчета, так как задается исходя из технологических возможностей, в данном случае координат составит $Z_3 = -1$ мм с целью гарантированной обработки всей поверхности детали.

Определение координат опорной точки 28 потребует использования тригонометрических выражений. Так на начальном этапе потребуется определить положение центра дуги радиусом 20 мм. Для этого на начальном этапе определяется размер гипотенузы AC прямоугольного треугольника (рис. 2.2).



Неуказанные радиусы скруглений 10 мм

Рис. 2.2. Прямоугольный треугольник ABC

$$AC = AB / \sin 60 = 100 - 20 / \sin 60 = 92,376 \text{ мм},$$

где 100 – общая высота детали, мм; 20 – высота точки A, мм;
 $\sin 60$ – угол наклона гипотенузы AC к оси X, мм.

Зная размер гипотенузы треугольника ABC можно определить координату X опорной точки 28 при помощи следующего выражения:

$$X_{28} = 150 - AC \cdot \cos 60 - 20 = 150 - 92,376 \cdot \cos 60 - 20 = 83,812 \text{ мм},$$

где 150 – габаритный размер детали по оси X, мм; 20 – радиус дуги, мм.

Координата Y может быть определена арифметическим выражением:

$$Y_{28} = 100 + R_{\phi} = 100 + 8 = 108 \text{ мм},$$

где 150 – габаритный размер детали по оси Y, мм.

По оси Z координата опорной точки назначается аналогично опорной точке 3 и равна $Z_{28} = -1$ мм.

Координаты остальных опорных точек аналитическим методом определяются аналогичным образом.

ЗАДАНИЯ

Определить координаты опорных точек траектории для детали, представленной в качестве задания согласно варианту.

Результаты оформить в виде таблицы в расчетно-технологической карте в соответствии с примером, представленном на рисунке 2.3.

Участок	Тип движения	Вид движения	Координаты конечной точки		
			X	Y	Z
$R-0$	Холостой ход	Прямолинейное	-40	100	40
$0-1$	Холостой ход	Прямолинейное	36	38	40
$1-2$	Рабочий ход	Прямолинейное	36	38	-1
$2-3$	Рабочий ход (Врезание)	Круговое	28	30	-1

Рис. 2.3. Пример таблицы

СОДЕРЖАНИЕ ОТЧЕТА

В качестве отчета по лабораторной работе студент предоставляет расчетно-технологическую карту и отвечает на контрольные вопросы.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Укажите методы определения координат опорных точек траектории.
2. Укажите принцип определения координат опорных точек аналитическим методом.
3. Перечислите варианты определения координат опорных точек геометрическим методом.

Лабораторная работа №3 **Ручная подготовка УП**

Цель работы: Изучение основных команд языка программирования ИСО (ISO) 7-бит, а также методики ручной подготовки управляющих программ.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

С момента появления первых станков с ЧПУ до внедрения новейших обрабатывающих центров появились различные языки для программирования обработки. Однако на сегодняшний день наиболее распространенным языком является программирование в G- и M-кодах.

Язык G- и M-кодов основывается на положениях Международной организации по стандартизации (ISO) и Ассоциации электронной промышленности (EIA). Официально этот язык считается стандартом для американских и европейских производителей оборудования с ЧПУ, и иногда его называют «ИСО 7 бит».

Основные функции G- и M-кодов и структура программ регулируются стандартами ГОСТ 20999-83, ISO 6983-1:2009, однако производители станков могут дополнять или изменять значения отдельных команд в связи с чем основным источником при программировании является паспорт УЧПУ или станка с указанными в нем функциями.

Несмотря на возможные отличия отдельных функций для конкретных станков указанные выше стандарты позволяют получить представления о программировании.

Рассмотрим процесс составления, структуру и содержание управляющей программы на примере обработки паза, представленного на рисунке 3.1.

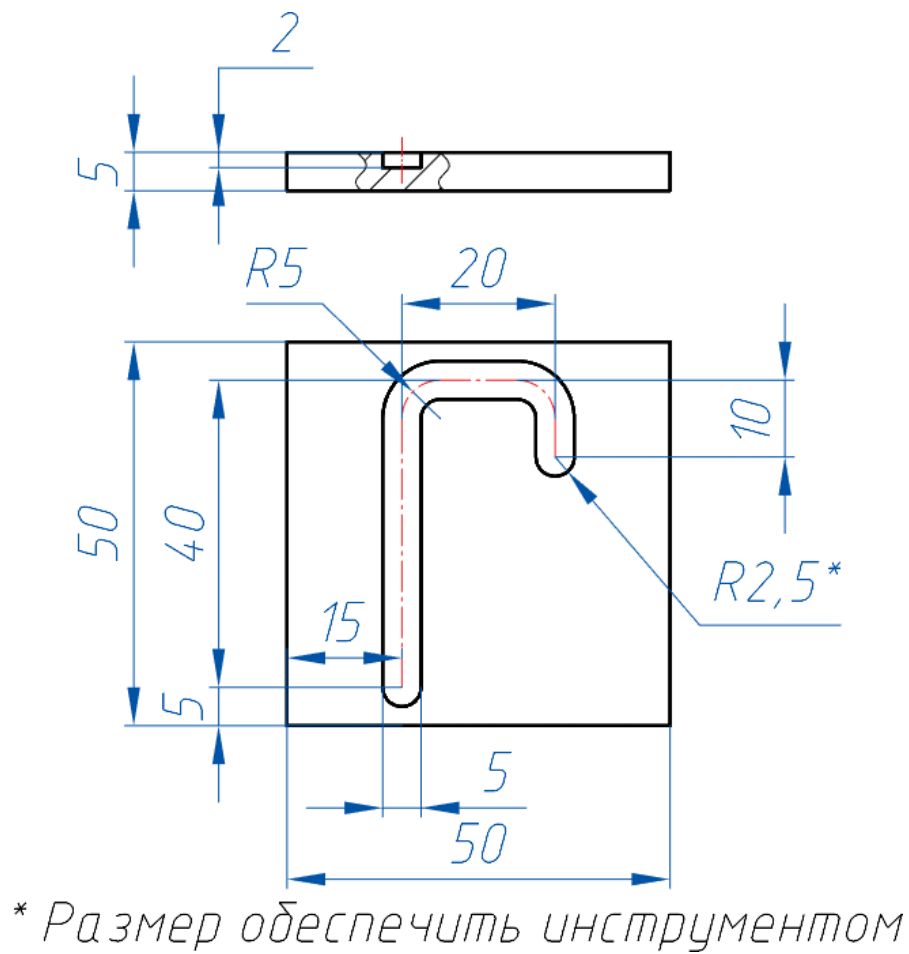


Рис. 3.1. Обрабатываемый паз

Траектория обработки паза имеет вид, представленный на рисунке 3.2. В качестве инструмента выбрана концевая цилиндрическая фреза диаметром 5 мм. Координаты опорных точек представлены в таблице 3.1.

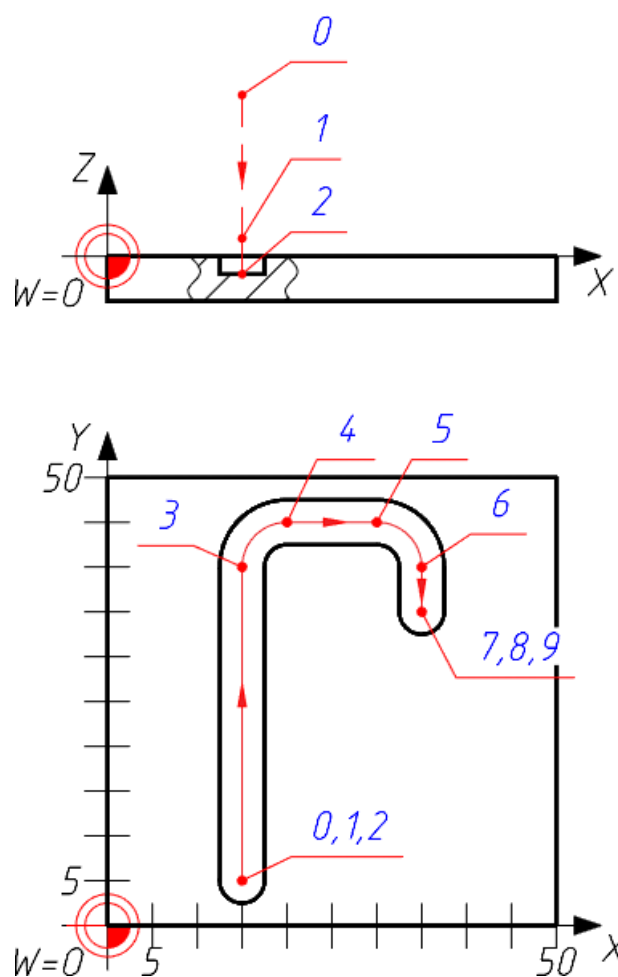


Рис. 3.2. Траектория обработки

Таблица 3.1

Координаты траектории

№ опорной точки	Координата X	Координата Y	Координата Z
0	15	5	30
1	15	5	2
2	15	5	-2
3	15	40	-2
4	20	45	-2
5	30	45	-2
6	35	40	-2
7	35	35	-2
8	35	35	2
9	35	35	30

Управляющие программы на языке G- и M-кодов представляет собой упорядоченным набором команд и специальных символов, сгруппированных в строки. Строка управляющей команды называется кадром, которые в свою очередь состоят из команд. В каждом кадре должна находиться как минимум одна команда, называемая словом данных. Команда или слово данных состоит из адреса и числа. На рисунке 3.3 представлена структура и состав управляющей программы.

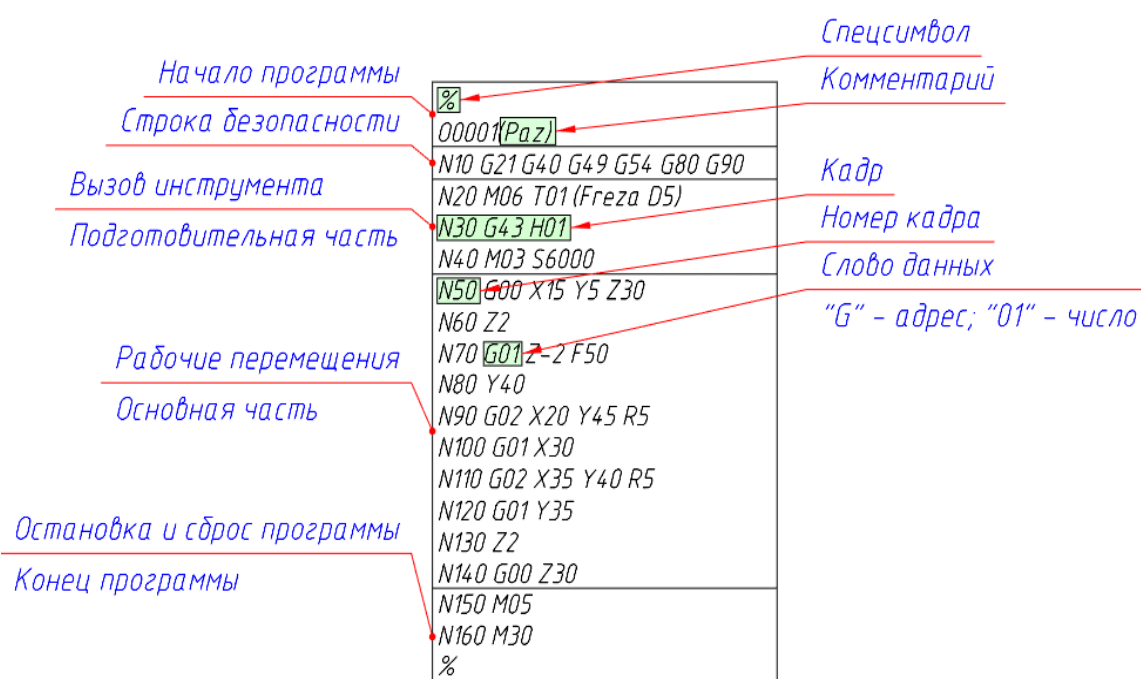


Рис. 3.3. Структура и состав управляющей программы

Каждая команда начинается с символа «%» на следующей строке записывается имя программы начинающейся с латинской буквы «O» (все команды обозначаются латинскими буквами). После имени программы может быть дан комментарий, который заключается в «()» и не обрабатывается устройством ЧПУ. Для рассматриваемого примера начало программы примет вид:

%

O0001 (Paz).

Следом за именем программы следует кадр, называемый строкой безопасности, данный кадр производит настройку системы ЧПУ на параметры программы. Необходимость в данном кадре возникает в связи с тем, что часть команд является

модельными и выполняются системой ЧПУ до тех пор, пока не будут отменены иными командами.

Например, при обработке предыдущей детали в качестве единиц измерения использовались дюймы, заданные командой G20, в случае отсутствия строки безопасности, содержащей команду G21 в текущей программе, система ЧПУ продолжит исчисление в дюймах, что может привести к порче заготовки, поломке инструмента или же станка.

Как правило, строка безопасности содержит следующие команды:

G21/G20 – указание единиц измерения при обработке;

G40 – отмена коррекции на радиус инструмента;

G49 – отмена коррекции на длину инструмента;

G54–G59 – указание на использование конкретной рабочей системы координат;

G80 – отмена постоянного цикла;

G90/G91 – указание режима позиционирования в абсолютных или относительных координатах.

В рассматриваемом случае строка безопасности примет вид:
G21 G40 G49 G54 G80 G90.

Следует отметить, что команды с адресом G группируются по функциональным признакам, пример чего представлен в таблице 3.2. Один кадр может содержать только одну команду группы, например, один кадр не может содержать команды G20 и G21, так как они противоречат друг другу.

Таблица 3.2

Функциональные группы команд с адресом G

Функциональная группа	Команды группы
Перемещения	G00, G01, G02, G03
Тип позиционирования	G90, G91
Единицы измерения	G20, G21
Рабочие системы координат	G54, G55, G56, G57, G58, G59
Коррекция радиуса инструмента	G40, G41, G42
Коррекция длины инструмента	G43, G44, G49
Активная плоскость обработки	G17, G18, G19
Постоянные циклы	G80, G81, G82, G83, G84, G85, G86, G87
Вызов подпрограмм	G98, G99

Начиная со строки безопасности может быть введена нумерация N10, N20 и т. д., данные номера не являются обязательными, не обрабатываются УЧПУ и необходимы для удобства составления и контроля программы. Интервал между номерами кадров выбирается произвольно по аналогии с номерами технологических операций, и задается на случай вставки дополнительных кадров.

За строкой безопасности вводится подготовительная часть, в рамках которой производится выбор инструмента, активация компенсации его длины, включение шпинделя с заданным числом оборотов.

В рамках рассматриваемого примера обработки кадр с выбором инструмента принимает вид:

M06 T01 (Freza D5).

В котором M06 – команда смены инструмента; T01 – указание на 1 инструмент в магазине. В случае необходимости выбора другого инструмента, например, №3 кадр примет вид: M06 T03. Коды с адресом G, называемые подготовительными, определяют настройку СЧПУ на определенный вид работы. Коды с адресом M называются вспомогательными и предназначены для управления режимами работы станка. В некоторых системах ЧПУ допускается только одна вспомогательная команда с адресом M.

Далее следует кадр, активирующий компенсацию длины инструмента, который имеет вид:

G43 H01.

В котором команда G43 активирует режим коррекции длины инструмента в положительном направлении, а команда H01 определяет преднастроенное значение компенсации длины первого инструмента.

Компенсация длины инструмента необходима в связи с тем, что фиксированная точка шпинделя в позиции инструмента, обозначаемая как N, как правило, находится на пересечении торца шпинделя с его осью вращения, и задавая координату по оси Z программист определяет положение именно этой точки. Рассмотрим пример, в котором необходимо переместить инструмент в точку с координатой по оси Z равной 50 мм. Предположим, что длина инструментального блока равна 70 мм.

В случае отсутствия компенсации длины инструмента в точку с координатой $Z = 50$ перемещена точка N в результате чего произойдет столкновение инструмента с заготовкой, пример чего представлен на рисунке 3.4.

В случае активации компенсации длины инструментального блока при перемещении в точку с координатой $Z = 50$ столкновения не произойдет, пример чего представлен на рисунке 3.5.

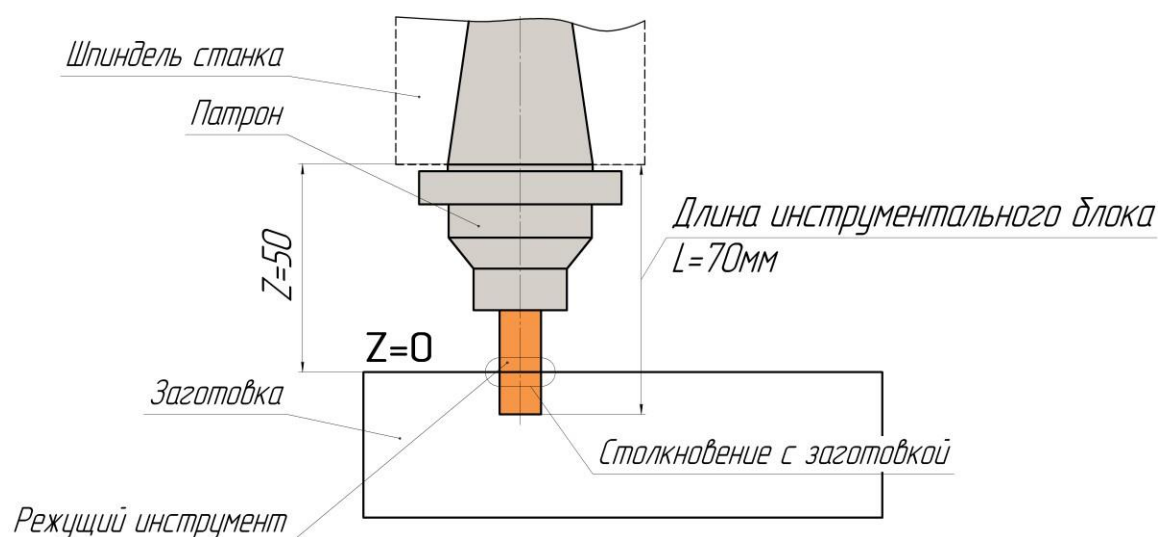


Рис. 3.4. Положение инструмента при отсутствии компенсации

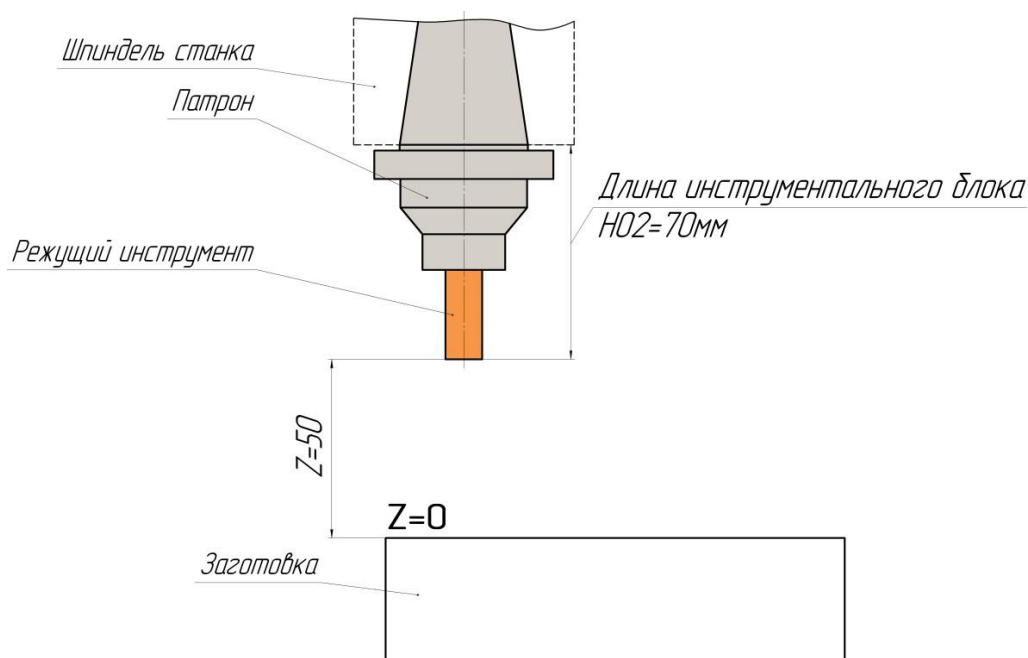


Рис. 3.5. Положение инструмента при активной компенсации

При составлении программы программист не вводит конкретное значение длины того или иного инструмента ограничиваясь соответствующей командой H01, H02 и т. д. Непосредственное указание значения компенсации производится оператором путем измерения, перед обработкой. После измерения значений оператор присваивает их конкретным регистрам в самом УЧПУ.

Следующим кадром подготовительной части программы является включение шпинделя и определение числа его оборотов. Команда начала вращения шпинделя против часовой стрелки (в положительном направлении вращения вокруг оси) M03 (по часовой стрелке M04), команда определения скорости вращения шпинделя имеет адрес S, число после которой определяет количество оборотов в минуту. Для рассматриваемого паза кадр примет вид:

M03 S6000.

В результате выполнения кадра начнется движение шпинделя с числом оборотов 6000 в минуту.

В основной части программы содержатся команды определяющие рабочие перемещения. Управление перемещениями исполнительных органов станка осуществляется специальным устройством интерполятором. Интерполяторы бывают двух видов: линейными (позволяют выполнять перемещение между соседними опорными точками по прямым линиям под любым углом) и линейно-круговыми (позволяют выполнять перемещение между соседними опорными точками, как по прямой, так и дуге окружности). В настоящее время на большинстве станков с ЧПУ применяются линейно-круговые интерполяторы позволяющие выполнять два вида интерполяции: линейную и круговую.

Линейной интерполяцией называется прямолинейное движение с заданной скоростью. Задается командой G01, а общий вид кадра такого движения можно представить в виде:

G01 Xn.n Yn.n Zn.n Fn.n.

В котором Xn.n, Yn.n, Zn.n – координаты конечной опорной точки по соответствующим осям; Fn.n – скорость перемещения.

Предположим, что необходимо задать перемещение из точки А с координатами $X = 20$, $Y = 10$, $Z = 20$ в точку В с координатами $X = 80$ $Y = 50$ $Z = 90$ со скоростью 100 мм/мин (рис. 3.6). Тогда кадр примет вид:

G01 X80 Y50 Z90 F100.

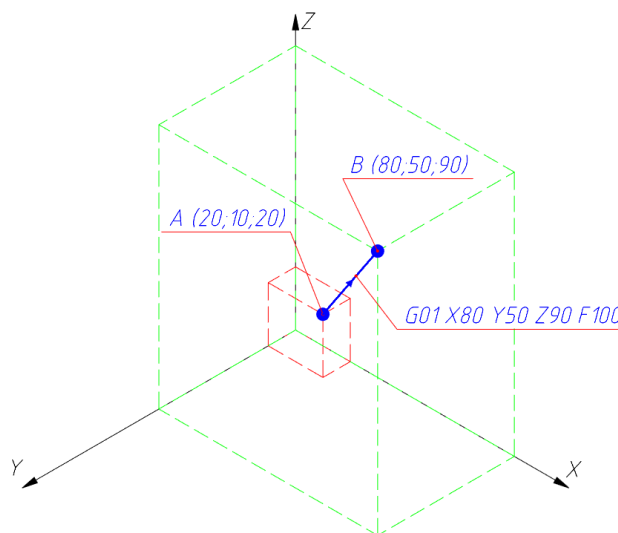


Рис. 3.6. Положение точек А и В

Круговой интерполяцией называется движение по дуге окружности с заданной скоростью. Задается круговая интерполяция двумя командами: G02 для движения по часовой стрелке; G03 против часовой стрелки (рис. 3.7).

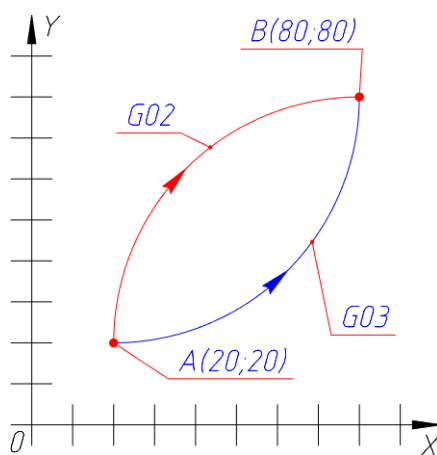


Рис. 3.7. Круговая интерполяция

Вне зависимости от направления существует два способа задания круговой интерполяции: при помощи указания

координат конечной точки и координат центра дуги относительно начальной точки; при помощи координат конечной точки и радиуса.

Общий вид кадра с указанием центра дуги можно представить в виде:

`G02 Xn.n Yn.n In.n Jn.n Fn.n.`

В котором $X_{n.n}$, $Y_{n.n}$ координаты конечной опорной точки по соответствующим осям; $I_{n.n}$ смещение до центра дуги по оси X относительно начальной точки; $J_{n.n}$ смещение до центра дуги по оси Y от начальной точки; $F_{n.n}$ скорость перемещения.

Предположим, что необходимо задать перемещение по дуге радиусом 60 мм против часовой стрелки от точки A с координатами $X = 20$ $Y = 20$ в точку B с координатами $X = 80$, $Y = 80$ со скоростью 100 мм/мин (рис. 3.8). Тогда кадр примет вид:

`G02 X80 Y80 I60 J0 F100.`

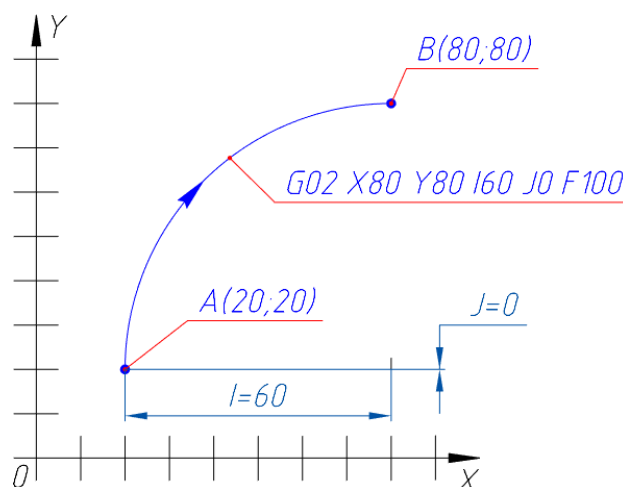


Рис. 3.8. Круговая интерполяция при помощи I и J

Следует отметить, что отсчет координат I и J осуществляется именно от начальной точки дуги, а не от начала координат, как показано на рисунке 3.9, в рамках рассмотренного примера.

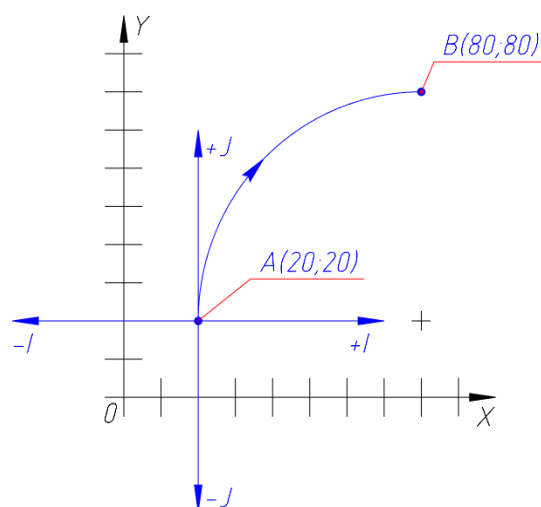


Рис. 3.9. Правила расчета координат I и J

Если добавить в кадр команды с координатами по оси Z и смещением центра по этой же оси K (I, J, K – параметры круговой интерполяции, и в коде программы радиус обозначается как ось K) круговая интерполяция станет винтовой, однако не все УЧПУ поддерживают винтовую интерполяцию. Кадр винтовой интерполяции для аналогичного примера, но с приращением 10 мм по оси Z точки B относительно точки A примет вид:

G02 X80 Y80 Z10 I60 J0 K10 F100.

Расчеты положения центра дуги в ряде случаев достаточно сложны, при этом современные УЧПУ поддерживают помимо представленного способа и способ задания при помощи радиуса. В данном случае для рассмотренного примера кадр принимает вид (рис. 3.10):

G02 X80 Y80 R60 F100.

В котором R60 указание на радиус дуги.

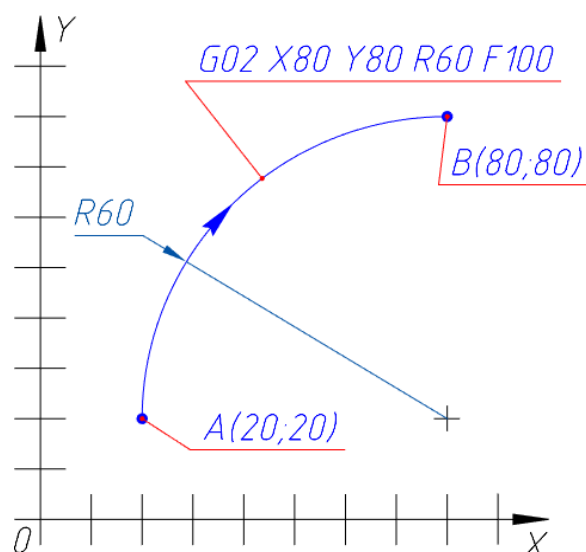


Рис. 3.10. Круговая интерполяция при радиусе R

В данном способе есть особенность. Если угол дуги меньше 180 градусов, а ее центр находится за пределами сегмента, образованного дугой и хордой, то указывается положительное значение радиуса; в обратном случае – отрицательное, пример чего продемонстрирован на рисунке 3.11.

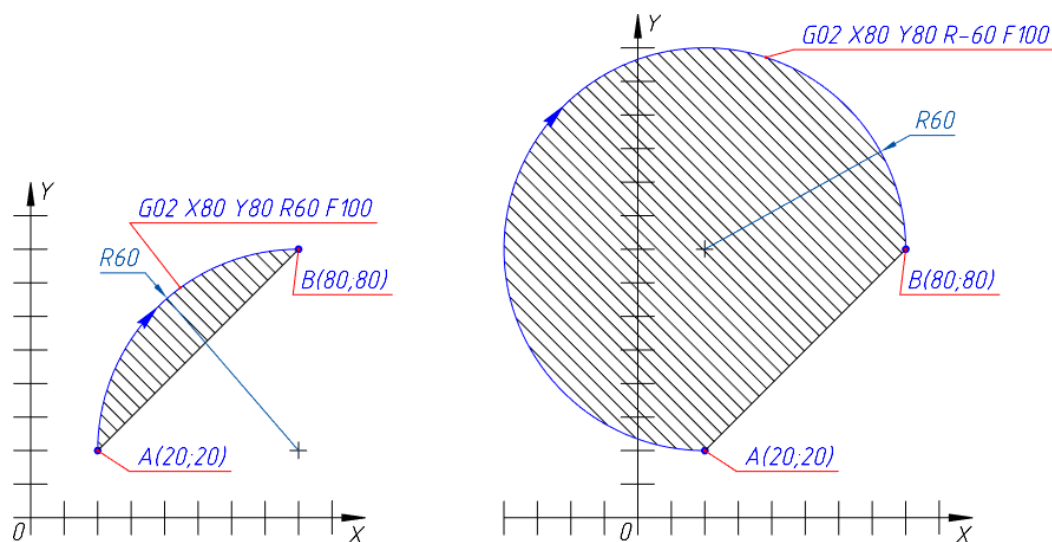


Рис. 3.11. Значение радиуса R

В случае необходимости выполнения винтовой интерполяции в кадр добавляется координата Z:

G02 X80 Y80 Z10 R60 F100.

Также в основной части программы помимо рабочих перемещений исполнительных органов включают команды ускоренного перемещения или позиционирования. Данные перемещения необходимы для максимально быстрого подвода инструмента к зоне обработки и перехода от одного контура обработки к другому. Данный режим не используется для обработки в связи с высокими скоростями перемещения, их непостоянством и непостоянством траектории. Ускоренное перемещение активируется командой G00 после которой указывают координаты точки. Обобщенный кадр подвода инструмента имеет следующий вид:

G00 Xn.n Yn.n Zn.n.

Скорость движения при использовании ускоренного перемещения не указывается, так как перемещения производятся на максимальной скорости.

В связи с непостоянством скорости по осям и траектории перемещения рекомендуется разбивать подвод инструмента на две части: сначала выполнять позиционирование по двум или трем осям в безопасной точке, после чего подводить к зоне обработки по одной оси, как правило, по оси Z. Отвод инструмента из зоны резания рекомендуется осуществлять в обратном порядке.

На примере рассматриваемого паза подвод инструмента к зоне резания разбит на две части: сначала выполняется подвод инструмента из исходной точки станка (R) в точку 0 с координатами $X = 15$, $Y = 5$, $Z = 30$; после чего выполняется подвод инструмента по оси Z к точке 1 с координатами $X = 15$, $Y = 5$, $Z = 2$. Кадры программы в данном случае примут вид:

G00 X15 Y15 Z30 Z2.

Во втором кадре режим движения и координата по осям X и Y не указываются, так как они неизменны, а команды, их определяющие, – модальные.

В результате преобразования траектории в команды, текст основной части управляющей программы примет вид, представленный в таблице 3.3.

Таблица 3.3

Основная часть управляющей программы

Содержание кадра	Пояснение
G00 X15 Y15 Z30	Ускоренное перемещение из исходной точки станка в опорную точку 0 (15, 5, 30)
Z2	Ускоренное перемещение из опорной точки 0 (15, 5, 30) в опорную точку 1 (15, 5, 2)
G01 Z-2 F50	Рабочее перемещение (линейная интерполяция) со скоростью 50 мм/мин из опорной точки 1 (15, 5, 2) в опорную точку 2 (15, 5, -2) (врезание в деталь на глубину паза)
Y40	Рабочее перемещение (линейная интерполяция) со скоростью 50 мм/мин из опорной точки 2 (15, 5, -2) в опорную точку 3 (15, 40, -2)
G02 X20 Y45 R5	Рабочее перемещение (круговая интерполяция по часовой стрелке с радиусом 5 мм) со скоростью 50 мм/мин из опорной точки 3 (15, 40, -2) в опорную точку 4 (20, 45, -2)
G01 X30	Рабочее перемещение (линейная интерполяция) со скоростью 50 мм/мин из опорной точки 4 (20, 45, -2) в опорную точку 5 (30, 45, -2)
G02 X35 Y40 R5	Рабочее перемещение (круговая интерполяция по часовой стрелке с радиусом 5 мм) со скоростью 50 мм/мин из опорной точки 5 (30, 45, -2) в опорную точку 6 (35, 40, -2)
G01 Y35	Рабочее перемещение (линейная интерполяция) со скоростью 50 мм/мин из опорной точки 6 (35, 40, -2) в опорную точку 7 (35, 35, -2)
Z2	Рабочее перемещение (линейная интерполяция) со скоростью 50 мм/мин из опорной точки 7 (35, 35, -2) в опорную точку 8 (35, 35, 2) вывод инструмента из зоны резания
G00 Z30	Ускоренное перемещение из опорной точки 8 (35, 35, 2) в опорную точку 9 (35, 35, 30)

По завершению всех необходимых перемещений программа завершается. В завершающей части программы производится остановка шпинделя командой M05 и сброс программы из памяти УЧПУ командой M30, а также указывается специальный символ %, обозначающий конец программы.

Весь текст программы для обработки паза (рис. 3.1), по разработанной траектории (рис. 3.2) примет вид:

```
%
O0001(Paz)
N10 G21 G40 G49 G54 G80 G90 N20 M06 T01 (Freza D5)
N30 G43 H01 N40 M03 S6000
N50 G00 X15 Y5 Z30 N60 Z2
N70 G01 Z-2 F50 N80 Y40
N90 G02 X20 Y45 R5 N100 G01 X30
N110 G02 X35 Y40 R5 N120 G01 Y35
N130 Z2
N140 G00 Z30 N150 M05 N160 M30
%
```

ЗАДАНИЯ

Разработать управляющую программу для обработки детали, представленной в качестве задания согласно варианту.

Справка по основным командам языка G- и M-кодов представлена в приложении 3.

СОДЕРЖАНИЕ ОТЧЕТА

В качестве отчета по лабораторной работе студент предоставляет программу обработки в виде файла с расширением *.txt созданную с использованием любого текстового редактора.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Укажите способы описания параметров круговой интерполяции.
2. Укажите способы описания винтовой интерполяции.
3. Укажите отличия между линейной интерполяцией и ускоренным перемещением.

Лабораторная работа №4

Редакторы УП и проверка кода

Цель работы: Изучение специализированных редакторов и их возможностей по ускорению процесса ручного программирования, а также проверке программ.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

На сегодняшний день существует три способа создания управляющих программ для станков с числовым программным управлением: ручное программирование; написание программы при помощи стойки ЧПУ станка; написание программы при помощи САМ-системы.

Запись управляющей программы в ручном режиме может выполняться при помощи большинства текстовых редакторов, например, «Блокнот» в операционной системе Windows. Помимо стандартных текстовых редакторов имеются специализированные редакторы, имеющие широкий набор дополнительных функций, обеспечивающих ускорения процесса написания и редактирования управляющих программ.

Примерами таких редакторов могут являться программы HSM Edit или Cimco Edit, которые позволяют автоматизировать нумерацию кадров, изменение форматирования кода программы, организовывать поиск по инструментам или строке, а кроме того, глобальное изменение кода, например, выполнять поворот всей траектории относительно одной из осей, вывод статистической информации о времени обработки и т. д.

Кроме редактирования данный класс программ, как правило, позволяет выполнять проверку корректности программы путем визуализации траектории движения инструмента. Визуализация с одной стороны позволяет проверить имеющиеся программы на наличие каких-либо ошибок, а с другой стороны позволяет выполнять построение траектории по мере написания программы.

Рассмотрим интерфейс редактора HSM Edit на примере программы обработки паза (рис. 3.1) по траектории

представленной в предыдущей работе (рис 3.2). Внешний вид редактора представлен на рисунке 4.1.

Редактор имеет привычный «ленточный» интерфейс с вкладками, панелями инструментов и командами. Так к основным вкладкам можно отнести следующие: «Editor» (Редактор); «NC Function» (Функции УП); «Backplot» (Бэкплот) и «File Compare» (Сравнение файлов). Под лентой расположена рабочая область, наполнение которой зависит от выполняемых действий. Так в представленном примере продемонстрирована рабочая область, содержащая код программы и визуализацию траектории.

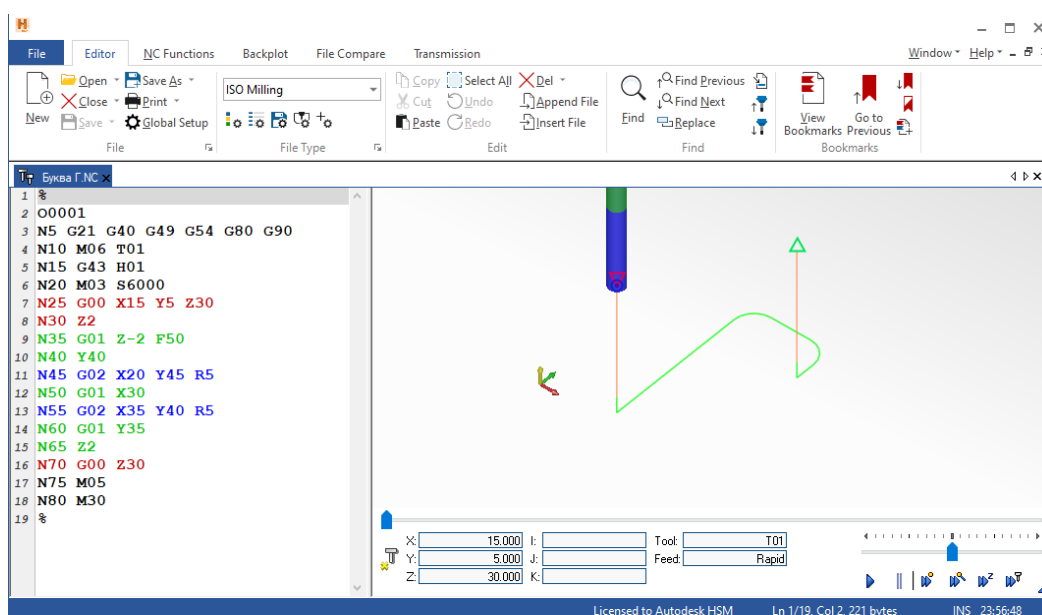


Рис. 4.1. HSM Edit

Вкладка «Editor» (рис. 4.2) позволяет производить типовые действия с программой, к которым относится создание, сохранение, печать и т. д., выполнять настройку редактора на определенный тип обработки, например, фрезерование (ISO Milling) или токарную обработку (ISO Turning), производить поиск в коде программы, в том числе и по специальным функциям, поиска инструмента, значения подачи и числа оборотов.

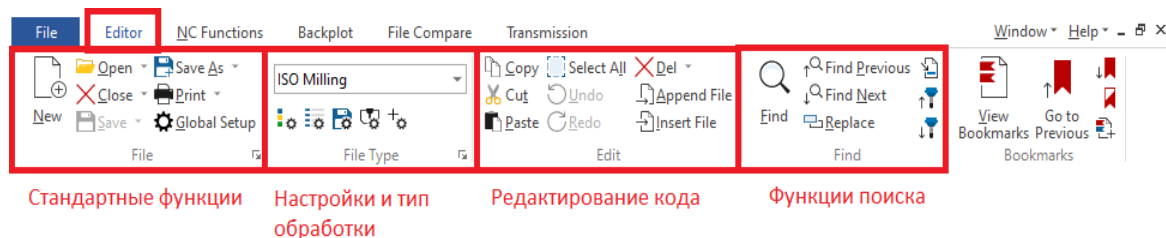


Рис. 4.2. Вкладка «Editor»

Вкладка «NC Function» (рис. 4.3) содержит специальные инструменты для работы с кодом, к которым относятся: настройка нумерации кадров; операции с кадрами, их форматом и представлением, функции редактирования и изменения кода, а также статистические данные о программе.

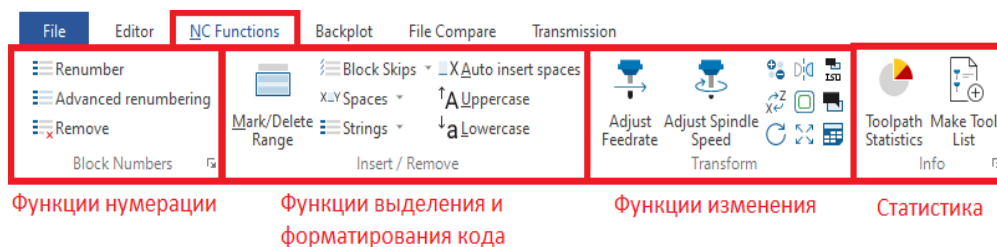


Рис. 4.3. Вкладка «NC Function»

Вкладка «Backplot» (рис. 4.4) содержит инструменты визуализации траектории движения инструмента в различных режимах и настройки визуализации.

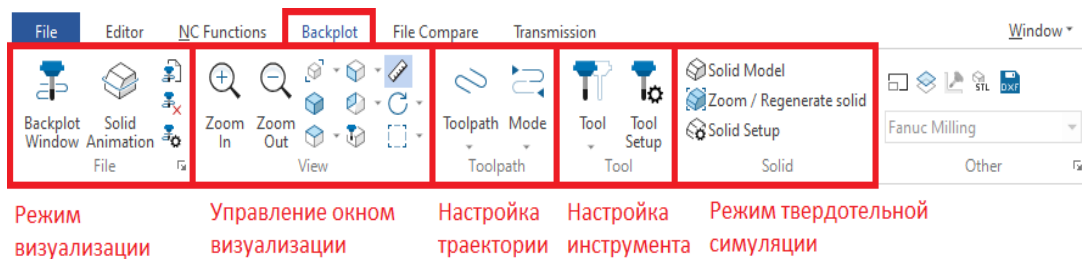


Рис. 4.4. Вкладка «Backplot»

Вкладка «File Compare» (рис. 4.5) позволяет выполнять сравнение различных программ с выделением отличий, навигацией между ними, а также синхронизацией.

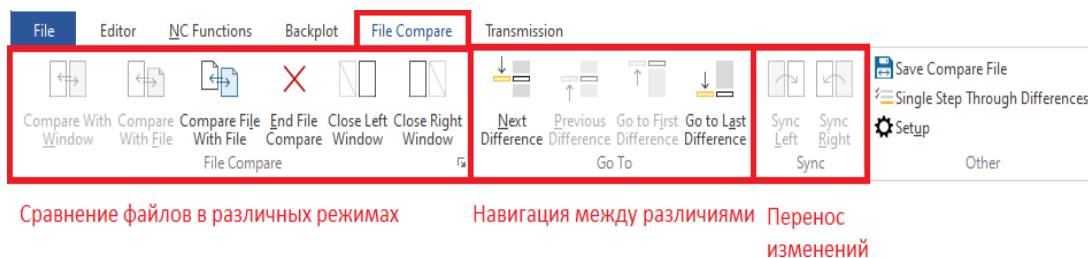


Рис. 4.5. Вкладка «File Compare»

В левой части рабочей области располагается код управляющей программы. Для удобства анализа элементы кода, относящиеся к перемещениям, имеют различные цвета, по умолчанию красный для быстрого перемещения G00; зеленый для линейной интерполяции G01 и синий для круговой или винтовой интерполяции G02, G03, черным цветом обозначены вспомогательные и технологические команды, не задающие перемещения (рис. 4.6). Из рисунка 4.6 хорошо видна модальность перемещений, примером чего могут служить кадры N35 и N40, в рамках которых выполняется линейная интерполяция.

```

%
O0001
N5 G21 G40 G49 G54 G80 G90
N10 M06 T01
N15 G43 H01
N20 M03 S6000
N25 G00 X15 Y5 Z30
N30 Z2
N35 G01 Z-2 F50
N40 Y40
N45 G02 X20 Y45 R5
N50 G01 X30
N55 G02 X35 Y40 R5
N60 G01 Y35
N65 Z2
N70 G00 Z30
N75 M05
N80 M30
%
```

Рис. 4.6. Цветовая схема управляющей программы

Наибольший интерес при написании программы и ее проверке представляет возможность визуализации траектории, включение которой осуществляется при помощи функции «Backplot Window» расположенной во вкладке «Backplot». Так

после активации данной команды в правой части рабочей области открывается специальное окно, пример которого показан на рисунке 4.7.

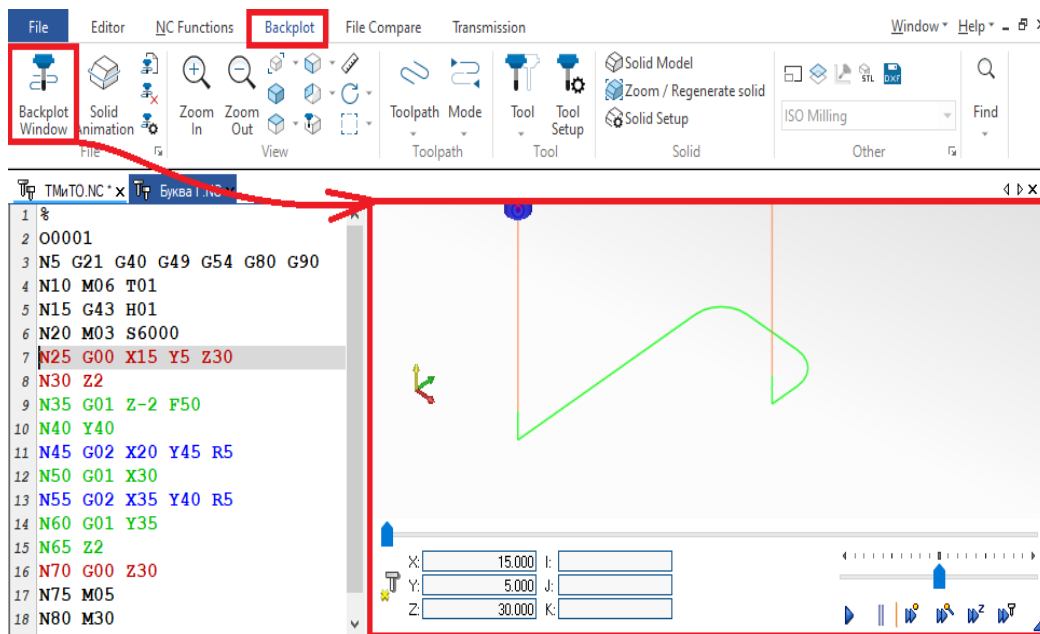


Рис. 4.7. Включение визуализации кода

Данный режим позволяет проверять правильность траектории как готовой программы, так и в процессе ее написания. В зависимости от активного кадра показывает положение инструмента и набор данных, к которым относятся: координаты точек; координаты центра дуги в случае круговой интерполяции, скорость движения инструмента и номер инструмента, пример чего представлен на рисунке 4.8.

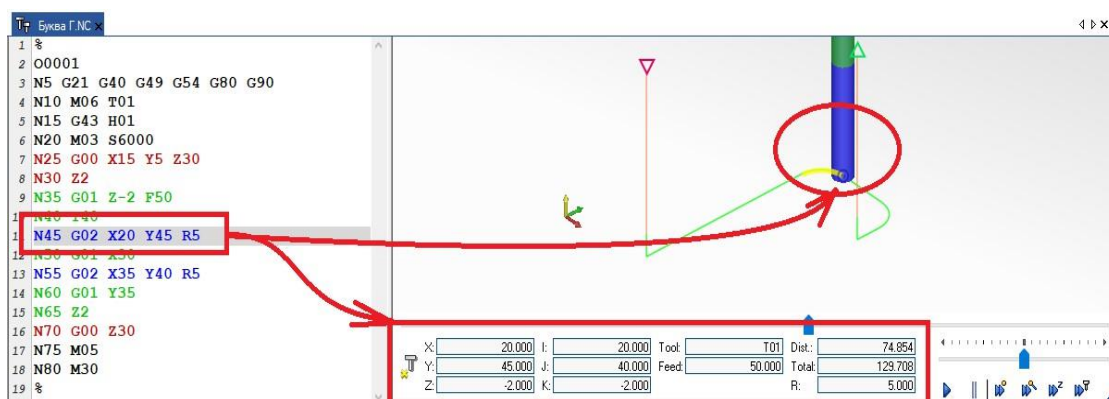


Рис. 4.8. Справочные параметры

Кроме визуализации траектории вкладка «Backplot» позволяет проводить твердотельную визуализацию. Для этого перед включением визуализации необходимо произвести настройку размеров твердого тела при помощи функции «Solid setup». Так для рассматриваемого примера обработки паза параметры твердого тела представлены на рисунке 4.9.

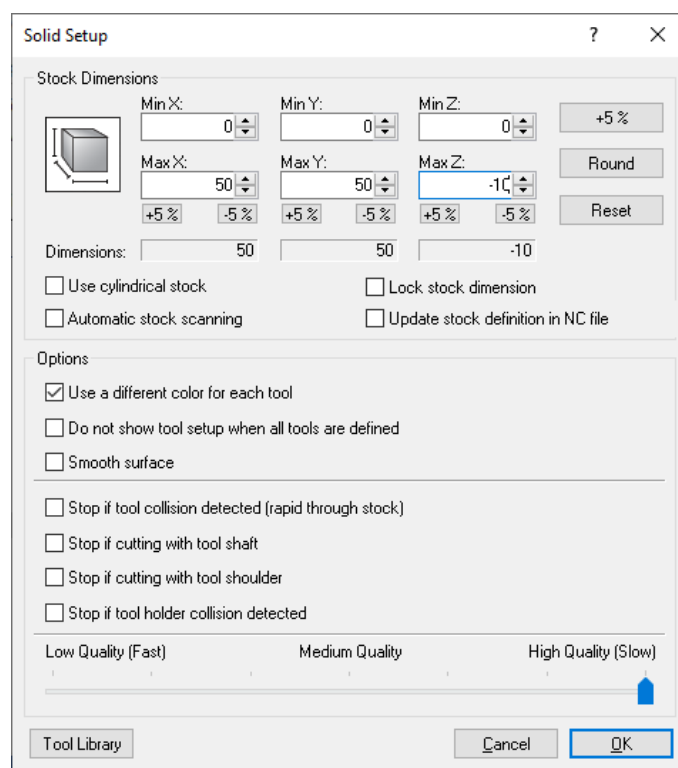


Рис. 4.9. Настройки размеров заготовки

Режим твердотельной симуляции включается при помощи функции «Solid Animation», а результат продемонстрирован на рисунке 4.10.

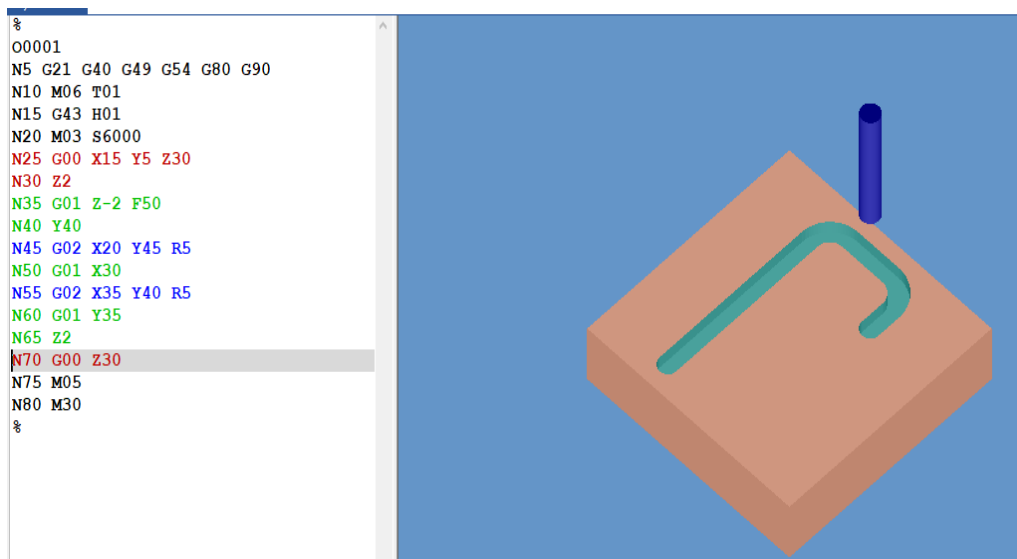


Рис. 4.10. Визуализация твердотельной обработки

В случае необходимости функционал редактора позволяет в некоторых пределах изменять код программы, например, поворачивать или отражать зеркально траекторию движения инструмента относительно какой-либо точки. Рассмотрим пример поворота траектории. Для выполнения данного действия необходимо перейти к вкладке «NC Function» и активировать команду «Rotate», после чего задать параметры поворота в появившемся окне. Так в рассматриваемом примере необходимо повернуть траекторию движения инструмента на 90 градусов относительно центра детали. До момента поворота рабочая система координат была выбрана таким образом, чтобы нулевая точка детали совпадала с левым нижним углом детали, пример чего представлен на рисунке 4.11.

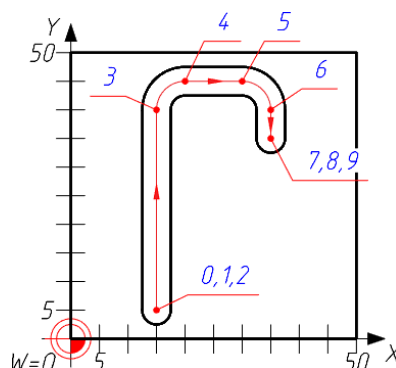


Рис. 4.11. Траектория обработки

Соответственно чтобы выполнить поворот траектории обработки на 90 градусов относительно центра детали в качестве координат точки поворота необходимо указать координаты $X = 25$ и $Y = 25$, как продемонстрировано на рисунке 4.12.

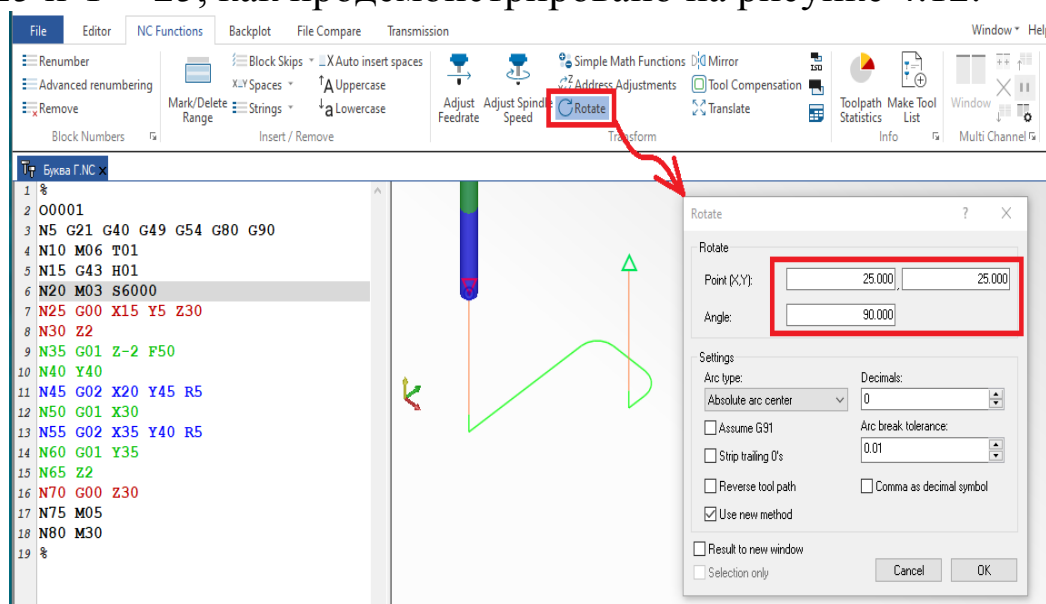


Рис. 4.12. Настройка операции поворота

В результате выполнения поворота траектория и код управляющей программы примут вид, показанный на рисунке 4.13.

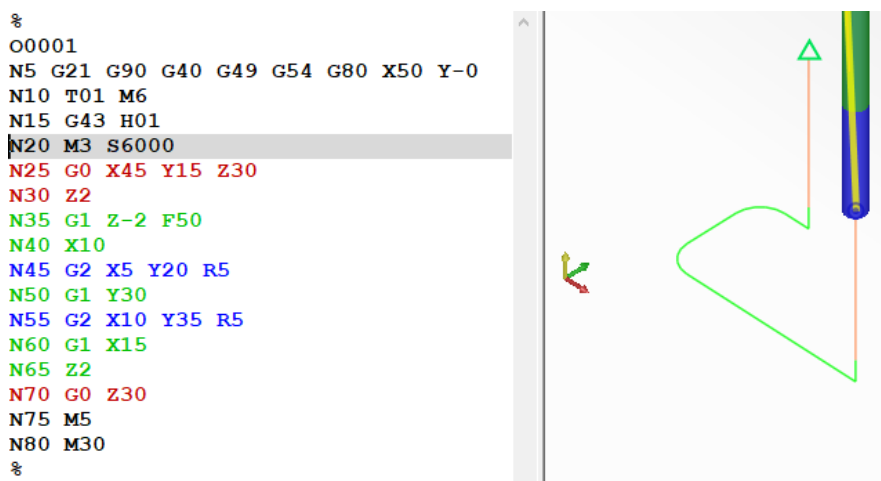


Рис. 4.13. Результат поворота траектории

Аналогичным образом выполняется и команда зеркального отражения, которая позволяет существенно ускорить процесс разработки управляющих программ для деталей, геометрия

которых зеркально отражена, либо обработку которых требуется выполнить в зеркальном виде.

Между написанием программистом ЧПУ управляющей программы и ее выполнением непосредственно на станке в программу могут быть внесены изменения, например, оператор станка может изменить число оборотов или скорость движения инструмента. Отслеживание подобных изменений в управляющих программах с большим числом кадров является достаточно трудоемкой задачей, однако в специальных редакторах имеется функционал сравнения.

В рассматриваемом редакторе для сравнения файлов используется вкладка «File Compare». Перед проведением сравнения программ необходимо открыть соответствующие файлы, после чего перейти к вкладке «File Compare» и активировать функцию «Compare With Window» (рис. 4.14).

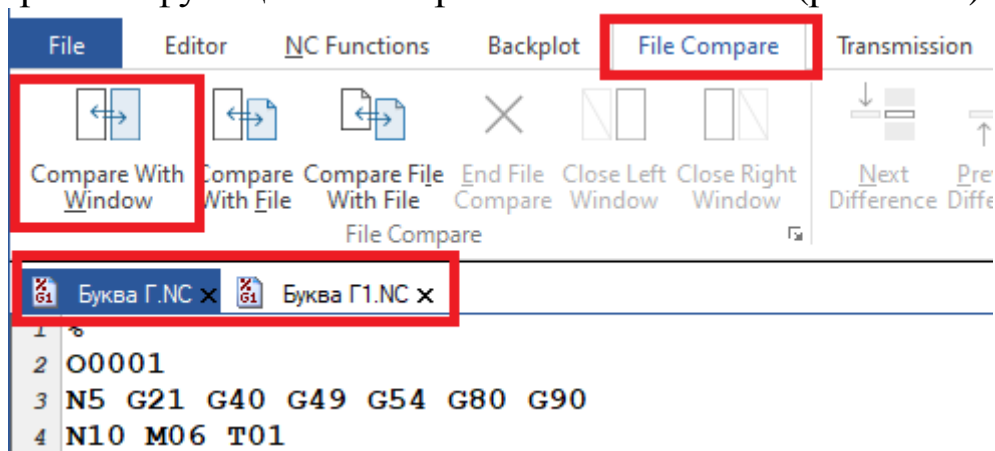


Рис. 4.14. Подготовка к сравнению

В результате окно редактора примет вид, представленный на рисунке 4.15.

Буква Г.1С / Буква Г1.1С			
1	%	1	%
2	O0001	2	O0001
3	N5 G21 G40 G49 G54 G80 G90	3	N5 G21 G40 G49 G54 G80 G90
4	N10 M06 T01	4	N10 M06 T01
5	N15 G43 H01	5	N15 G43 H01
6	N20 M03 S6000	6	N20 M03 S600
7	N25 G00 X15 Y5 Z30	7	N25 G00 X15 Y5 Z30
8	N30 Z2	8	N30 Z2
9	N35 G01 Z-2 F50	9	N35 G01 Z-2 F500
10	N40 Y40	10	N40 Y40
11	N45 G02 X20 Y45 R5	11	N45 G02 X20 Y45 R5
12	N50 G01 X30	12	N50 G01 X30
13	N55 G02 X35 Y40 R5	13	N55 G02 X35 Y40 R5
14	N60 G01 Y35	14	N60 G01 Y35
15	N65 Z2	15	N65 Z2
16		16	N70 G00 Z10
17	N70 G00 Z30	17	N75 G00 Z30
18	N75 M05	18	N80 M05
19	N80 M30	19	N85 M30
20	%	20	%

Рис. 4.15. Результат сравнения

Отличающиеся участки кода управляющих программ будут подсвечены фиолетовым цветом, а в случае, если в одной из программ нет совпадающих кадров, будет пропущена строка или несколько строк и это будет выделено красным. В процессе проверки у пользователя имеется возможность навигации между отличиями, а также синхронизации кода между программами. Кроме того пользователь может произвести экспорт отличий в текстовый файл.

ЗАДАНИЯ

Разработать управляющую программу для вычерчивания траектории движения инструмента на станке СК Роутер 6040s.

В качестве траектории использовать собственные Имя и Фамилию, буквы которых выполнить в соответствии с представленным в приложении 5 шрифтом.

При создании координат опорных точек управляющей программы придерживаться следующих требований к траектории:

- в качестве нулевой точки детали принимается точка 0, отмеченная для первой буквы имени в приложении 5;
- траекторию выполнять в две строки: 1-я – имя; 2-я – фамилия;
- расстояние между соседними буквами 5 мм;

- расстояние между нижней точкой первой строки и верхней точкой второй строки 10 мм;
- в случае необходимости отрыва инструмента от плоскости обработки высота подъема не менее 5 мм;
- глубина обработки при вычерчивании должна обеспечивать прижим инструмента к полотну, для чего все рабочие перемещения должны осуществляться с координатой – 1 мм по оси Z. За 0 по оси Z принимать точку контакта инструмента с полотном;
- вычерчивание осуществляется с подачей 400 мм/мин без включения вращения шпинделя.

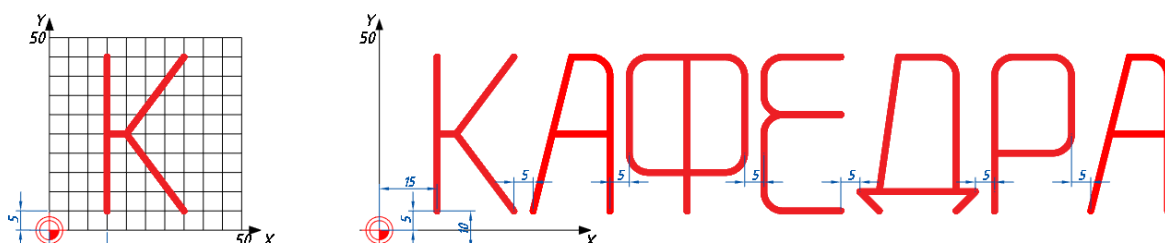


Рис. 4.16. Пример расположения элементов траектории

СОДЕРЖАНИЕ ОТЧЕТА

В качестве отчета по лабораторной работе студент предоставляет программу обработки в виде файла с расширением *.txt или *.nc, созданную с использованием любого специального редактора.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Укажите назначение вкладки «NC Function» и перечислите основные команды.
2. Укажите назначение вкладки «Backplot» и перечислите основные команды.
3. Укажите назначение вкладки «File Compare» и перечислите основные команды.

Рекомендуемая литература

Основная литература

1. Мейер, Б. Объектно-ориентированное программирование и программная инженерия : учебное пособие / Б. Мейер. – 2-е изд., испр. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 286 с. : ил. – URL: <https://biblioclub.ru/index.php?page=book&id=429034> (дата обращения: 13.04.2025).

2. Проектирование технологических процессов машиностроительных производств : учебник / В. А. Тимирязев, А. Г. Схиртладзе, Н. П. Солнышкин, С. И. Дмитриев. – Санкт-Петербург : Лань, 2022. – 384 с. – ISBN 978-5-8114-1629-5. – URL: <https://e.lanbook.com/book/211652> (дата обращения: 13.04.2025).

Дополнительная литература

1. Баженова, И. Ю. Введение в программирование : учебное пособие : / И. Ю. Баженова, В. А. Сухомлин. – Москва : Интернет-Университет Информационных Технологий (ИНТУИТ) : Бином. Лаборатория знаний, 2007. – 327 с. : ил. – (Основы информационных технологий). – ISBN 5-9556-0077-9. – URL: <https://biblioclub.ru/index.php?page=book&id=232982> (дата обращения: 13.04.2025).

2. Программирование технологических контроллеров в среде Unity : учебное пособие / Г. В. Саблина, А. В. Суворов, В. В. Медведков, В. Г. Шайхшнейдер. – 4-е изд. – Новосибирск : Новосибирский государственный технический университет, 2017. – 207 с. : ил., табл. – ISBN 978-5-7782-3386-7. – URL: <https://biblioclub.ru/index.php?page=book&id=575617> (дата обращения: 13.04.2025).