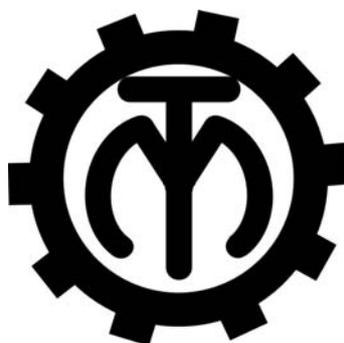


МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Технология машиностроения»

ПЕРСПЕКТИВЫ ПРИМЕНЕНИЯ СТАНКОВ С ЧПУ И РОБОТОВ ПРИ ПРОИЗВОДСТВЕ МАШИН

*Методические рекомендации к лабораторным работам
для студентов специальности
1-36 80 02 «Инновационные технологии в машиностроении»
очной и заочной форм обучения*



Могилев 2020

УДК 621.9.06
ББК 34.63
П 27

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Технология машиностроения» «27» мая 2020 г.,
протокол № 14.

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

Рецензент канд. техн. наук И. В. Лесковец

Приведены методические рекомендации к проведению лабораторных работ по дисциплине «Перспективы применения станков с ЧПУ и роботов при производстве машин» для студентов специальности 1-36 80 02 «Инновационные технологии в машиностроении» очной и заочной форм обучения.

Учебно-методическое издание

ПЕРСПЕКТИВЫ ПРИМЕНЕНИЯ СТАНКОВ С ЧПУ И РОБОТОВ ПРИ ПРОИЗВОДСТВЕ МАШИН

Ответственный за выпуск

В. М. Шеменков

Корректор

А. А. Подошевка

Компьютерная верстка

Н. П. Полевничая

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. . Уч. -изд. л. . Тираж 26 экз. Заказ №

Издатель и полиграфическое исполнение:

Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».

Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.

Пр-т Мира, 43, 212022, Могилев.

© Белорусско-Российский
университет, 2020

Содержание

Введение.....	4
Инструкция по охране труда.....	5
1 Лабораторная работа № 1. Разработка 3D-модели в среде трехмерного моделирования.....	6
2 Лабораторная работа № 2. Программирование токарной обработки в САМ-системе.....	11
3 Лабораторная работа № 3. Программирование фрезерной обработки в САМ-системе.....	17
4 Лабораторная работа № 4. Программирование электроэрозионного проволочно-вырезного станка с ЧПУ.....	23
5 Лабораторная работа № 5. Программирование токарно-фрезерной обработки в САМ-системе.....	32
6 Лабораторная работа № 6. Проектирование операций с применением промышленных роботов в среде офлайн-программирования	33
Список литературы.....	48

Введение

Целью учебной дисциплины является формирование у студентов знаний, умений и навыков, позволяющих решать задачи, связанные с применением промышленных роботов, технологического оборудования, оснащенного устройствами числового программного управления в действующих и проектируемых технологических процессах.

Целью лабораторных работ является формирование умений и навыков разработки управляющих программ технологического оборудования.

Лабораторные работы проводятся в компьютерном классе, а также в лабораториях кафедры.

По лабораторным работам каждый студент оформляет отчет, содержащий тему работы, цель, основные теоретические сведения, описание выполняемых задач, результаты работы и выводы.

Защита лабораторной работы проводится во время занятий устно, письменно или в форме тестирования.

Инструкция по охране труда

Общие требования безопасности

Допуск студентов к лабораторным занятиям производится только после инструктажа по технике безопасности, о чем делается соответствующая запись в специальном журнале (бланке).

Требования безопасности перед началом работы

- 1 Внимательно изучить содержание и порядок проведения лабораторной работы, а также безопасные приемы его выполнения.
- 2 Перед каждым включением оборудования предварительно убедиться, что его пуск безопасен.

Требования безопасности во время работы

- 1 Точно выполнять все указания преподавателя.
- 2 Не прикасаться к находящимся под напряжением элементам электрических цепей, к корпусам стационарного электрооборудования, не производить переключения в цепях до отключения источника питания.
- 3 Запрещается во время работы оборудования снимать ограждения и предохранительные устройства, а также держать их открытыми.
- 4 Во время работы запрещается касаться руками вращающихся и перемещающихся частей, вводить руки в зону движения.

Требования безопасности по окончании работы

- 1 Полностью выключить оборудование.
- 2 Привести в порядок рабочее место.
- 3 Предупредить преподавателя обо всех, даже малейших и незначительных, неисправностях оборудования.

Требования безопасности в аварийных ситуациях

- 1 В случае травмирования кого-либо немедленно доложить преподавателю.
- 2 При выходе оборудования из строя (нет освещения, возгорание токопроводов, поломка механических деталей и др.) необходимо:
 - отключить оборудование (обесточить);
 - доложить преподавателю о случившемся, а в случае возгорания приступить к немедленной его ликвидации первичными средствами пожаротушения.

1 Лабораторная работа № 1. Разработка 3D-модели в среде трехмерного моделирования

Цель работы: ознакомление с системой трехмерного моделирования SolidWorks и изучение принципов построения трехмерных моделей деталей.

1.1 Теоретические сведения

1.1.1 Создание нового документа.

Новый документ в SolidWorks можно создать несколькими способами:

- нажать кнопку «Создать» на стандартной панели инструментов;
- выбрать в меню пункты «Файл», «Новый».

Откроется окно «Новый документ SolidWorks».

В диалоговом окне «Новый документ SolidWorks» представлены три вида стандартных шаблонов документов: «Деталь», «Сборка» и «Чертеж».

Шаблон представляет собой основу нового документа, которая содержит сведения о формате, свойствах документа, а также параметры пользователя, такие как единицы измерения, примечания, или стандарты по оформлению чертежей.

Шаблон документа «Деталь» предназначен для создания трехмерных моделей отдельных компонентов (деталей).

Шаблон «Сборка» предназначен для создания трехмерных моделей сборок нескольких компонентов (деталей).

Шаблон «Чертеж» предназначен для создания двухмерных чертежей по имеющимся моделям отдельных деталей или же сборок.

1.1.2 Ознакомление с пользовательским интерфейсом SolidWorks.

Графическая область представляет собой пространство, в котором на экране отображается деталь, сборка или чертеж, а также происходит построение и оформление вышеперечисленных объектов.

В панели менеджера отображается следующая комбинация:

- Дерево конструирования FeatureManager.
- PropertyManager (Менеджер свойств).
- ConfigurationManager.
- Приложения третьих фирм.

Дерево конструирования (FeatureManager) расположено в левой части окна SolidWorks и представляет собой набор строк, содержащих полную информацию о трехмерном объекте.

Для большинства эскизов, элементов и инструментов рисования в SolidWorks можно открыть окно PropertyManager в левой панели. В окне PropertyManager (Менеджера свойств) отображаются свойства объекта или элемента. Поэтому можно указать необходимые свойства, не открывая диалоговое окно, закрывающее графическую область.

Менеджер конфигураций (ConfigurationManager) предназначен для создания нескольких конфигураций деталей и сборок в документе.

Панели инструментов отображают различные кнопки, активизирующие команды. На панели инструментов «Элементы» сосредоточены все кнопки команд, позволяющие различными способами строить твердотельные модели.

Менеджер команд представляет собой контекстную панель инструментов, которая обновляется автоматически.

Строка состояния расположена в нижнем правом углу окна и предназначена для отображения информации о выполняемой функции.

1.1.3 Отображение трехмерных моделей объектов.

Команды для отображения трехмерных моделей располагаются в меню «Вид», а наиболее часто использованные команды продублированы кнопками на панели инструментов «Вид».

1 Предыдущий вид – позволяет вернуть предыдущий вид объекта.

2 Изменить размер экрана – производит масштабирование объекта до размеров окна графической области.

3 Увеличить элемент вида – увеличивает область объекта при помощи граничной рамки.

4 Увеличить/уменьшить вид – приводит к увеличению масштаба вида при перетаскивании указателя мыши вверх и, соответственно, к уменьшению масштаба при перетаскивании указателя вниз.

5 Вращать вид – позволяет вращать вид модели в графической области. Данная команда активируется, если нажать на колесо мыши.

6 Перемещать – позволяет перемещать вид параллельно самому себе в пределах окна графической области.

7 Панель инструментов «Стандартные виды» – обеспечивает ориентацию модели, сборки или эскиза, используя один из стандартных видов по умолчанию.

8 Панель инструментов «Стандартные виды» обеспечивает ориентацию модели, сборки или эскиза, используя один из стандартных видов по умолчанию.

Панель инструментов «Стандартные виды» открывается при нажатии на треугольник справа от иконки.

1 Спереди – эта команда разворачивает модель, отображая вид спереди.

2 Сзади – эта команда разворачивает модель, отображая вид сзади.

3 Слева – эта команда разворачивает модель, отображая вид слева.

4 Справа – эта команда разворачивает модель, отображая вид справа.

5 Сверху – эта команда разворачивает модель, отображая вид сверху.

6 Снизу – эта команда разворачивает модель, отображая вид снизу.

7 Изометрия – эта команда разворачивает деталь в изометрическую ориентацию.

8 Триметрия – эта команда поворачивает деталь в триметрическую ориентацию.

9 Диметрия – модель разворачивается в диметрическую ориентацию.

10 Перпендикулярно – эта команда располагает выбранную плоскость или грань детали параллельно экрану дисплея.

11 Каркасное представление – отображает все кромки трехмерной модели.

12 Невидимые линии отображаются – приводит к отображению всех кромок модели, при этом невидимые в текущем виде кромки изображаются другим цветом.

13 Удалить невидимые линии – отображает на модели только те кромки, которые можно видеть в текущей ориентации вида.

14 Закрасить с кромками – эта команда позволяет увидеть модель в закрашенном виде с четко прорисованными кромками.

15 Закрасить – позволяет увидеть закрашенное отображение модели.

16 Тени в режиме «Закрасить» – эта команда отображает тени под моделью.

17 Разрез – отображает вырез детали или сборки с помощью указанных плоскостей или граней.

1.2 Порядок выполнения лабораторной работы

Создать новый файл, выбрать шаблон «Деталь» и сохранить созданный документ.

Используя команду «Осевая линия», создать ось симметрии вала (рисунок 1.1), проходящую через ось координат. Используя команду «Линия», вычертить половину контура вала. Установить необходимые размеры, используя команду «Автоматическое нанесение размеров».

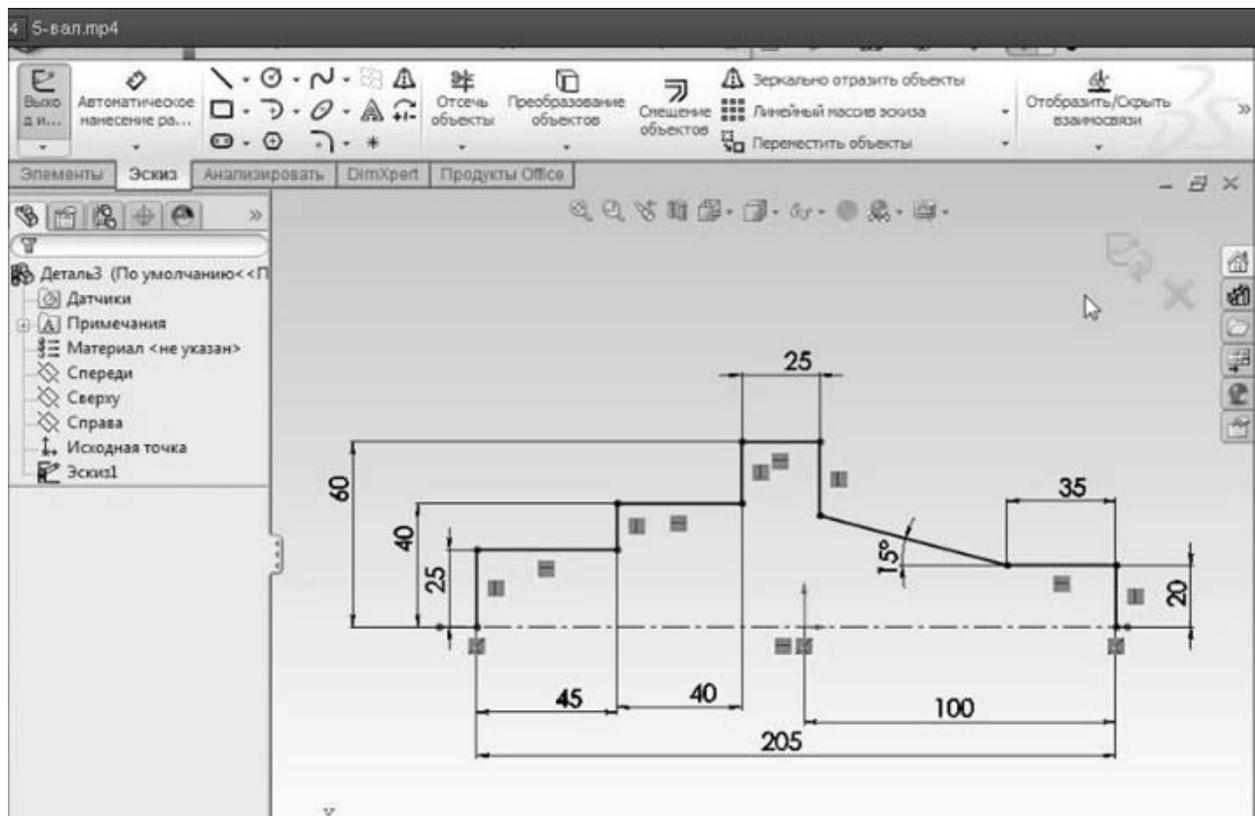


Рисунок 1.1 – Пример эскиза вала с проставленными размерами

Во вкладке «Элементы» выбрать команду «Повернутая бобышка».

Проставить фаски, используя команды «Скругление», «Фаска».

Для создания шпоночного паза в дереве конструирования выбрать плоскость «Спереди». Удерживая кнопку «Ctrl» и левую кнопку мыши, переместить данную плоскость вверх, указывая направление на величину радиуса шейки вала, на которой расположен шпоночный паз.

На перемещенной плоскости в режиме «Эскиз» построить эскиз шпоночного паза шириной b .

Перейти в режим «Элементы». Используя команду «Вытянутый вырез», прорезать паз на глубину t_1 в соответствии с заданием.

Сохранить созданную трехмерную модель вала.

Задание

В соответствии с заданием построить трехмерную модель вала (рисунок 1.2).

Варианты заданий представлены в таблице 1.1.

Таблица 1.1 – Варианты заданий

Номер варианта	Эскиз проектируемого вала (рисунок 1.2)	$d_{вал}$	b	t_1
1	a	42	12	5
2	b	28	8	4
3	a	35	10	5
4	b	36	10	5
5	a	50	16	5,5
6	b	42	12	5
7	a	38	10	5
8	b	22	6	4
9	a	42	12	5
10	b	34	10	5
11	a	25	8	4
12	b	30	8	4
13	a	40	12	5
14	b	54	16	6
15	a	32	10	5
16	b	63	18	7
17	a	28	8	4
18	b	42	12	5
19	a	34	10	5
20	b	35	10	5

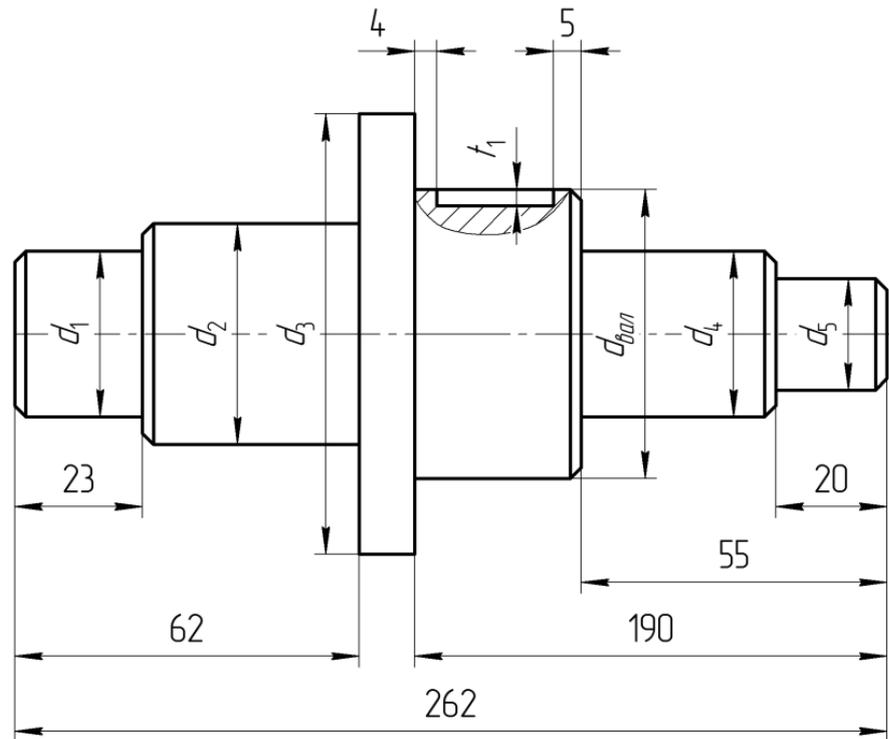
Для вала, изображенного на рисунке 1.2, a , принять следующее:
 $d_1 = 0,6d_{вал}$; $d_2 = 0,75d_{вал}$; $d_3 = 1,5d_{вал}$; $d_4 = 0,6d_{вал}$; $d_5 = 0,35d_{вал}$.

Диаметры d_1 и d_4 должны быть кратны 5.

Для вала, изображенного на рисунке 1.2, b , принять следующее:
 $d_1 = 0,4d_{вал}$; $d_2 = 0,65d_{вал}$; $d_3 = 0,65d_{вал}$; $d_4 = 0,55d_{вал}$; $d_5 = 0,4d_{вал}$.

Диаметры d_2 и d_3 должны быть кратны 5.

a)



б)

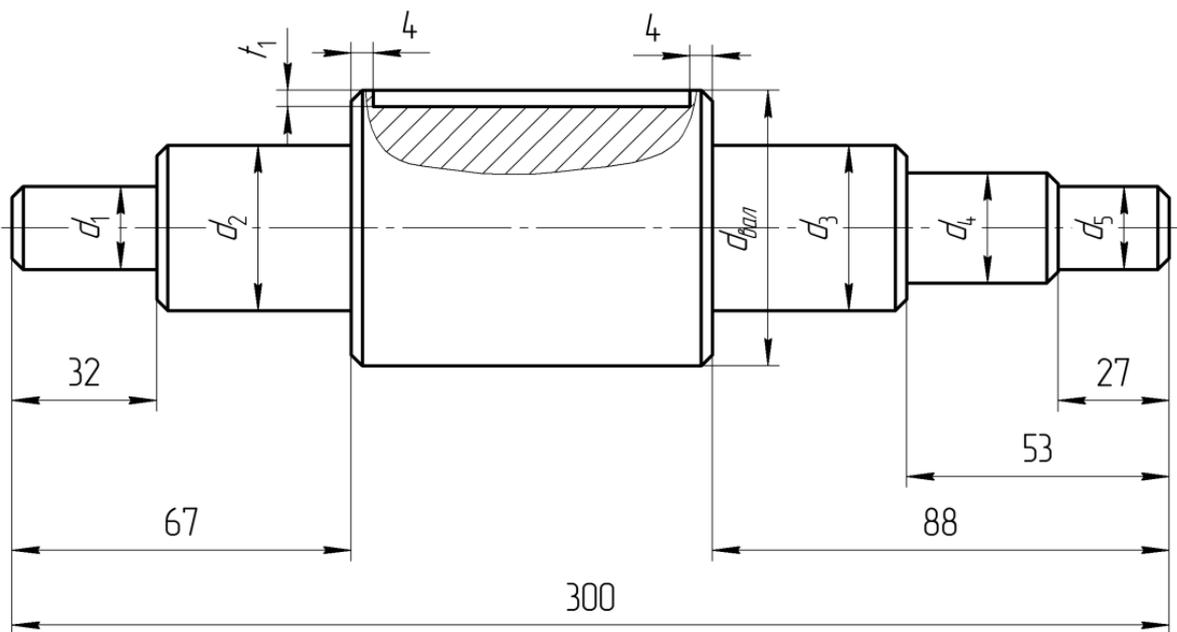


Рисунок 1.2 – Эскизы проектируемых валов

Контрольные вопросы

- 1 Что такое шаблон документа SolidWorks? Какие основные шаблоны вы знаете?
- 2 Назовите основные виды панелей инструментов SolidWorks.
- 3 Назовите основные объекты проектирования в SolidWorks.
- 4 Какие инструменты построения эскиза вы знаете?
- 5 Назовите основные этапы построения модели вала в SolidWorks.

2 Лабораторная работа № 2. Программирование токарной обработки в САМ-системе

Цель работы: овладение практическими навыками автоматизированного проектирования управляющих программ для токарных станков с ЧПУ с использованием САМ-системы.

2.1 Теоретические сведения

САМ-система (сокращение от англ. Computer-Aided Manufacturing) – система автоматизированного программирования (САП) станков с ЧПУ – общий термин для обозначения программных систем подготовки управляющих программ (УП) для станков с числовым программным управлением.

Область применения САМ-систем определяется конструктивно-технологическими признаками деталей и технологической группой станков. Различают четыре типа САМ-систем, областями применения которых, являются:

1) обработка тел вращения со ступенчатым и криволинейным профилями на токарных станках с 2-координатным управлением;

2) обработка отверстий на сверлильных станках с позиционным управлением и обработка поверхностей, параллельных координатным плоскостям, на фрезерных станках с 2,5-координатным управлением;

3) комплексная (многоцелевая) обработка корпусных деталей на сверлильно-фрезерно-расточных станках и обрабатывающих центрах;

4) обработка поверхностей деталей сложной формы (штампы, пресс-формы, турбинные лопатки и т. п.) на 3–5-координатных фрезерных станках.

SprutCAM – современная полнофункциональная САМ-система, предназначенная для разработки управляющих программ для обработки деталей различной сложности на обрабатывающих центрах, фрезерных, гравировальных, токарных станках и токарно-фрезерных обрабатывающих центрах с ЧПУ.

SprutCAM используется при программировании обработки резанием штампов, пресс-форм, литейных форм, прототипов изделий, мастер-моделей, различных деталей машин и конструкций, шаблонов, при гравировке и вырезке надписей и изображений.

В данной лабораторной работе будет рассмотрен порядок формирования программ для токарной обработки с использованием модуля «Токарная обработка» (LATHE). Этот модуль позволяет программировать следующие типы технологических переходов:

- подрезка торцов и отрезка;
- послойное черновое точение и растачивание наружных и внутренних поверхностей;
- чистовое точение и растачивание наружных и внутренних поверхностей;
- точение и растачивание канавок;
- сверление осевых отверстий;

- сверление несоосных отверстий и фрезерование поверхностей на токарно-фрезерных обрабатывающих центрах;
- нарезание резьбы.

Основная задача модуля LATHE – формирование траектории движения режущего инструмента – решается с учетом геометрии обрабатываемой детали и заготовки, выбранных технологических переходов и геометрии режущего инструмента.

Геометрия детали и заготовки может быть получена как при помощи инструментов черчения, имеющихся в системе, так и путем импорта данных из САД-систем.

Геометрическая модель изготавливаемой детали, заготовки, оснастки и т. п. может быть подготовлена в любой САД-системе и импортирована в SprutCAM.

SprutCAM имеет множество функций для последующего преобразования модели, а также встроенную среду двумерных параметрических построений для создания и редактирования 2D-элементов.

Для учета геометрии режущего инструмента в системе предусмотрена библиотека инструмента с возможностью редактирования и расширения.

В процессе расчета траектории система осуществляет контроль зарезания поверхностей. Сформированная траектория может просматриваться и редактироваться. Режим реалистичного просмотра (симуляция) позволяет пользователю увидеть на модели процесс обработки детали и конечный результат.

Система имеет в наличии расширяемую и редактируемую библиотеку постпроцессоров, позволяющих транслировать получаемые с использованием системы траектории режущих инструментов в G-коды конкретных устройств ЧПУ.

2.2 Порядок выполнения лабораторной работы

Рассмотрим последовательность проектных операций процедуры разработки управляющей программы для токарной обработки с использованием модуля токарной обработки (LATHE) системы SprutCAM. В качестве иллюстрации примеров выполнения отдельных задач будет использована деталь, эскиз которой представлен на рисунке 2.1.

Предполагаемая последовательность переходов обработки рассматриваемой детали:

- подрезка правого торца;
- черновое точение наружного контура детали;
- чистовое точение наружного контура детали;
- точение канавок;
- нарезание резьбы;
- сверление отверстия.

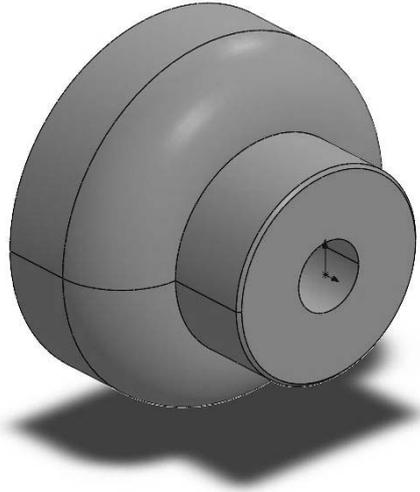


Рисунок 2.1 – Трехмерный эскиз детали

Для загрузки программы SprutCAM необходимо двойным щелчком нажать по ярлыку «SprutCAM» на рабочем столе.

Для загрузки учебника SprutCAM необходимо двойным щелчком нажать по ярлыку «Учебник SprutCAM» на рабочем столе.

Из меню Учебника SprutCAM выбрать раздел Токарная обработка, Обработка торца (урок № 1).

После выбора урока выполните последовательно все предусмотренные в нем упражнения, перемещаясь по ним с помощью кнопок навигации.

Завершив изучение упражнений урока 1, перейдите к уроку 2 и выполните остальные пять уроков.

Задание

С использованием САМ-системы SprutCAM спроектировать управляющую программу для токарной обработки детали, эскиз которой выбирается по варианту индивидуального задания (рисунки 2.2–2.11).

Выполнение индивидуального задания включает.

- 1 Создание контуров детали и заготовки в среде САД-системы.
- 2 Экспорт геометрической информации из САД-системы.
- 3 Импорт геометрической информации из файла экспорта САД-системы в САМ-систему.
- 4 Создание 3D-моделей детали и заготовки.
- 5 Задание технологической информации (определение последовательности технологических переходов, выбор инструментов и задание режимов резания).
- 6 Геометрическое моделирование технологических переходов и устранение возможных ошибок.
- 7 Выбор постпроцессора и генерация управляющей программы для УЧПУ металлорежущего станка.
- 8 Составление отчета по лабораторной работе.

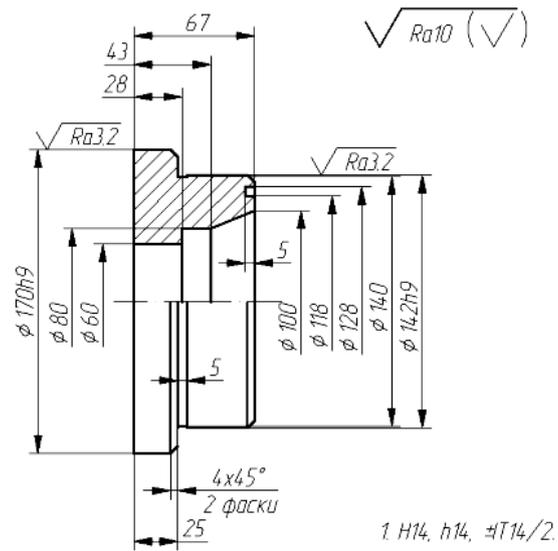


Рисунок 2.2 – Вариант 1

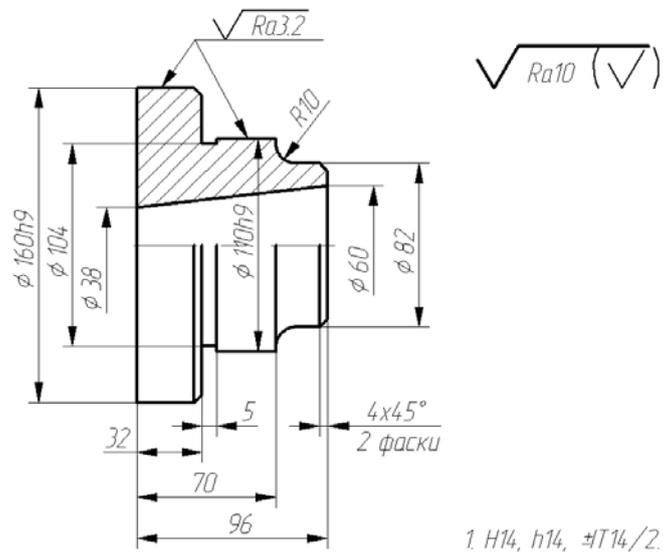


Рисунок 2.3 – Вариант 2

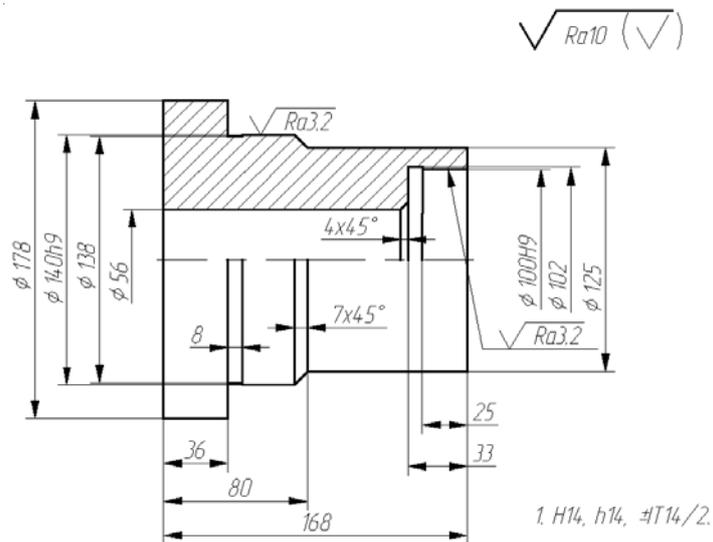


Рисунок 2.4 – Вариант 3

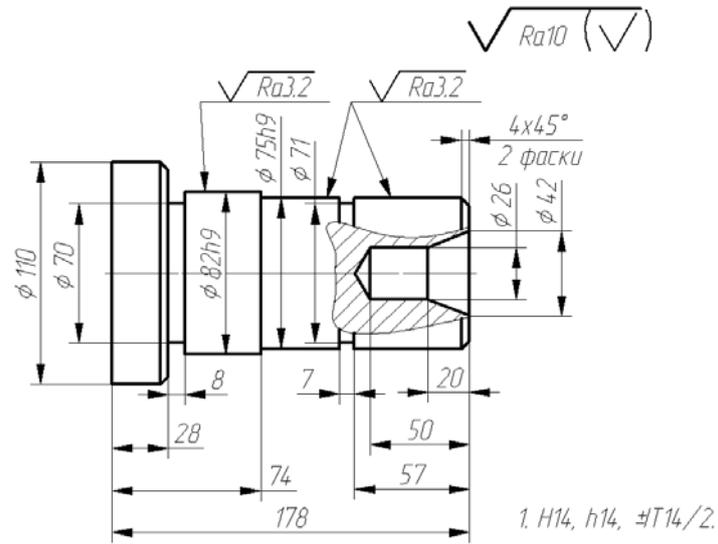


Рисунок 2.5 – Вариант 4

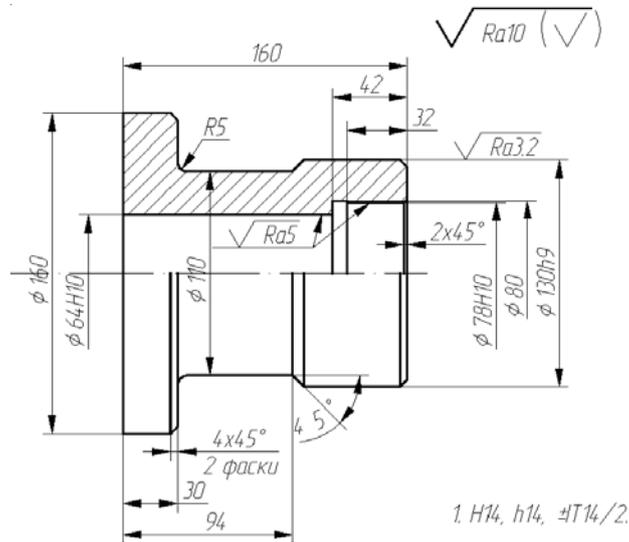


Рисунок 2.6 – Вариант 5

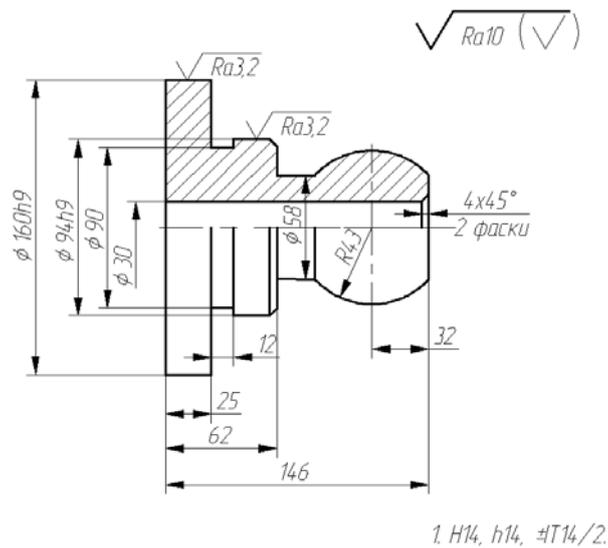


Рисунок 2.7 – Вариант 6

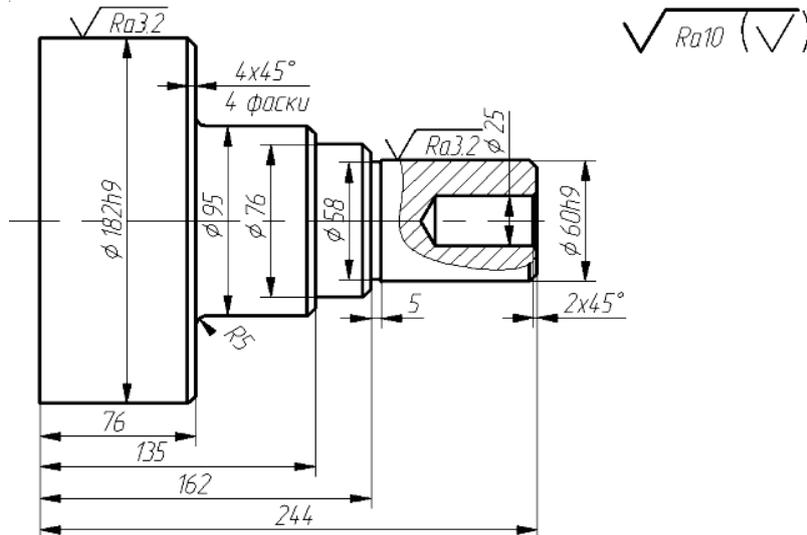


Рисунок 2.11 – Вариант 10

Контрольные вопросы

- 1 Области применения САМ-систем?
- 2 Программирование каких типов металлорежущих станков возможно с использованием САМ-системы SprutCAM?
- 3 Каковы отличительные особенности САМ-системы SprutCAM?
- 4 Каков общий порядок подготовки УП для токарного станка с ЧПУ с использованием САМ-систем?
- 5 Какие существуют способы подготовки исходной геометрической информации об обрабатываемой детали?

3 Лабораторная работа № 3. Программирование фрезерной обработки в САМ-системе

Цель работы: овладение практическими навыками автоматизированного проектирования управляющих программ для фрезерных станков с ЧПУ с использованием САМ-системы.

3.1 Теоретические сведения

Модуль Mill интегрированной системы SprutCAM предназначен для разработки управляющих программ для фрезерных станков с ЧПУ.

Обработка детали проектируется на основе геометрической модели, построенной в любой САД-системе. Также геометрическая модель может быть построена и в самой системе SprutCAM, так как система имеет три основных режима работы:

- 1) подготовка геометрической модели;
- 2) двухмерные построения;
- 3) формирование процесса обработки.

Управление режимами работы производится выбором соответствующих закладок на панели главного окна системы (3D-модель, 2D-геометрия, Технология, Моделирование).

В режиме 2D формируются плоские контуры, которые используются как зоны ограничения при обработке и как направляющие кривые при определении стратегии, а также для определения геометрии заготовки.

В режиме Технология формируется процесс обработки детали, который представляет собой последовательность технологических переходов различных типов. Изменение их очередности и редактирование параметров возможны на любом этапе проектирования техпроцесса. Имеется возможность доработки остаточного материала, обработки по заданной высоте гребешка, оптимизации обработки пологих и крутых участков, оптимизации по направлению уклона поверхности и прочие средства для получения оптимальной траектории движения инструмента.

Порядок действий проектирования управляющей программы для станка с ЧПУ, в общем случае, сводится к последовательности действий: импортировать или создать модель обрабатываемых поверхностей, создать последовательность технологических переходов, назначить их параметры, рассчитать и сгенерировать управляющую программу (УП).

Управляющая программа сначала формируется в виде команд на языке CLDATA, а затем автоматически преобразуются в УП для конкретного станка в соответствии с правилами, заложенными в постпроцессоре.

Постпроцессор – это программа, содержащая правила преобразования последовательности команд на языке CLDATA в УП для конкретного станка.

Для создания оригинальных постпроцессоров существует специальный модуль, называемый генератором постпроцессоров (Postprocessors generator).

Ассоциативная связь геометрии детали и процесса моделирования обработки позволяет при изменении геометрии автоматически получить новую УП.

Моделирование процесса обработки на экране позволяет проверить и отладить УП до ее передачи на станок. Имеется возможность автоматического определения областей, оказавшихся недоступными для заданного инструмента, и обработки их инструментом меньшего диаметра.

Любую деталь, подлежащую обработке, можно представить набором конструктивных элементов (КЭ). Например, для фрезерной обработки это плоскость, колодец, стенка и т. д.

Конструктивный элемент – это геометрическая модель, на базе которой разрабатывается УП. Она может быть создана в системе конструкторского проектирования SprutCAD или непосредственно в SprutCAM .

Проектирование обработки детали в SprutCAM выполняется поэлементно. Технологический объект (ТО) – это единица информации, который содержит данные об обработке одного КЭ. Эти данные могут быть разделены на две части: геометрическая и технологическая информация.

Геометрическая информация:

– геометрия конструктивного элемента (контур);

- тип конструктивного элемента (выступ, впадина, инвариантная область, гребень и т. д.);

- параметры конструктивного элемента (глубина, припуск и др.).

Технологическая информация:

- тип технологического перехода (фрезерование, сверление и т. д.);

- параметры технологических переходов (инструмент, подача, подход-отход, стратегия и т. д.).

Система сохраняет связь между технологическими объектами и геометрическими элементами, на базе которых они созданы. Если обрабатываемый контур модифицирован в SprutCAD, то не нужно заново проектировать обработку, достаточно автоматически пересчитать траекторию инструмента со старыми параметрами технологических переходов. Это справедливо и для параметрических геометрических моделей.

Существуют также технологические объекты, не связанные с непосредственной обработкой (снятием металла). Такие ТО называются технологическими параметрами.

Технологические параметры задают:

- координаты начального положения инструмента и безопасной плоскости;

- плоскость холостых ходов при перемещении инструмента от одного КЭ к другому.

Маршрут обработки – это последовательность технологических переходов (по терминологии SprutCAM). При желании пользователь может изменить порядок технологических переходов, что приведет к изменению маршрута обработки.

После того, как создан маршрут обработки запускается команда Пуск, которая на основе информации, содержащейся в ТО, рассчитывает перемещения инструмента, необходимые для обработки детали. Эта последовательность перемещений инструмента называется траекторией движения инструмента.

Результатом работы команды Пуск является последовательность команд на языке CLDATA. Текст на языке CLDATA содержит команды перемещения инструмента (собственно траекторию инструмента), команды не связанные с перемещением инструмента (например, включение/выключение шпинделя или охлаждения) и справочную информацию (название УП, модель станка и др.).

Режим Постпроцессор конвертирует последовательности команд на языке CLDATA в УП в соответствии с выбранным постпроцессором (моделью станка). При формировании текста управляющей программы постпроцессор:

- переводит данные, подготовленные процессором, в координатную систему станка;

- формирует перемещения в абсолютной системе координат или в приращениях;

- проверяет данные расчета;

- приводит величины подач и скоростей шпинделя в соответствие с паспортными данными станка.

Результатом работы системы SprutCAM является управляющая программа (УП) – последовательность команд в формате станка с ЧПУ.

3.2 Порядок выполнения лабораторной работы

Загрузить программу SprutCAM двойным щелчком по ярлыку SprutCAM на рабочем столе.

Загрузить учебник двойным щелчком по ярлыку Учебник SprutCAM на рабочем столе. Из меню Учебника SprutCAM выбрать раздел Фрезерная обработка, 2D-обработка.

После выбора урока выполните последовательно предусмотренные в нем два упражнения Построение 2D-контуров и Обработка 2D-контуров. Для перемещения по разделам используйте кнопки навигации.

После выполнения всех упражнений урока 2D-обработка выполните упражнения урока 2,5D-обработка. Завершив изучение уроков, выполните индивидуальное задание согласно заданному варианту и составьте отчет по лабораторной работе.

Задание

1 Получить индивидуальное задание – вариант детали и номер набора параметров (таблица 3.1).

2 Проанализировать чертеж детали. Выделить контуры заготовки и обрабатываемых поверхностей и присвоить им имена.

3 В режиме «2D» построить контуры заготовки и обрабатываемых поверхностей в плоскости X–Y.

4 Определить последовательность переходов фрезерной операции.

5 Задать технологические параметры переходов фрезерной операции (режущий инструмент, режимы резания, подвод, отвод инструмента, стратегию).

6 Выполнить генерацию программы в виде последовательности команд языка CLDATA.

7 Выбрать режим «Постпроцессор». Установить заданную систему ЧПУ и сгенерировать управляющую программу.

Таблица 3.1 – Варианты заданий

Вариант	Длина	Высота	Радиус	Диаметр	Рисунок
1	2	3	4	5	6
1	$L = 160,$ $L1 = 80,$ $L2 = 130$	$H = 70, H1 = 30,$ $H2 = 30, H3 = 30,$ $H4 = 60$	$R = 10,$ $R1 = 15$	$d = 10,$ $D2 = 20,$ $D4 = 40$	3.1
2	$L = 80,$ $L1 = 40,$ $L2 = 54$	$H = 35, H1 = 15,$ $H2 = 10, H3 = 15,$ $H4 = 30$	$R = 5,$ $R1 = 8$	$d = 5,$ $D2 = 16,$ $D4 = 20$	3.1
3	$L = 160,$ $L1 = 80,$ $L2 = 130$	$H = 70, H1 = 30,$ $H2 = 30, H3 = 40,$ $H4 = 27, H5 = 30$	$R = 10,$ $R1 = 8$	$d = 10,$ $D = 14,$ $D1 = 20$	3.2

Окончание таблицы 3.1

1	2	3	4	5	6
4	$L = 112,$ $L1 = 56,$ $L2 = 91$	$H = 49, H1 = 21,$ $H2 = 16, H3 = 32,$ $H4 = 19, H5 = 25$	$R = 7,$ $R1 = 3$	$d = 7,$ $D = 10,$ $D1 = 14$	3.2
5	$L = 100,$ $L1 = 50,$ $L2 = 74$	$H = 38, H1 = 18,$ $H2 = 16, H3 = 16$	$R1 = 7,$ $R2 = 25,$ $R3 = 10$	$D = 14,$ $D1 = 8$	3.3
6	$L = 150,$ $L1 = 80,$ $L2 = 111$	$H = 57, H1 = 26,$ $H2 = 22, H3 = 24$	$R1 = 10,$ $R2 = 38,$ $R3 = 15$	$D = 21,$ $D1 = 12$	3.3
7	$L = 100,$ $L1 = 50,$ $L2 = 74$	$H = 32, H1 = 30,$ $H2 = 18, H3 = 17,$ $H4 = 24$	$R1 = 7,$ $R2 = 25,$ $R3 = 10$	$D = 12,$ $D1 = 8$	3.4
8	$L = 200,$ $L1 = 100,$ $L2 = 148$	$H = 64, H1 = 60,$ $H2 = 36, H3 = 34,$ $H4 = 48$	$R1 = 14,$ $R2 = 50,$ $R3 = 20$	$D = 24,$ $D1 = 16$	3.4
9	$L = 160,$ $L1 = 130$	$H = 68, H1 = 30,$ $H2 = 36$	$R = 80$	$D1 = 10,$ $D2 = 30,$ $D3 = 30,$ $D4 = 52$	3.5
10	$L = 240,$ $L1 = 195$	$H = 102, H1 = 45,$ $H2 = 50$	$R = 120$	$D1 = 15,$ $D2 = 45,$ $D3 = 50,$ $D4 = 78$	3.5

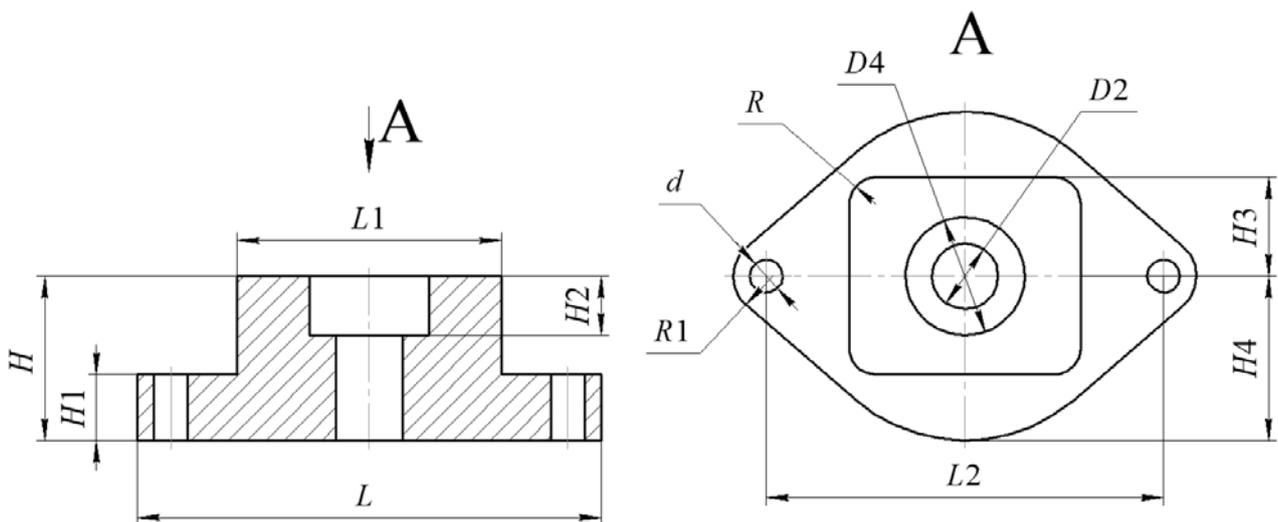


Рисунок 3.1 – Варианты 1 и 2, станок: BOSCH с системой ЧПУ M8

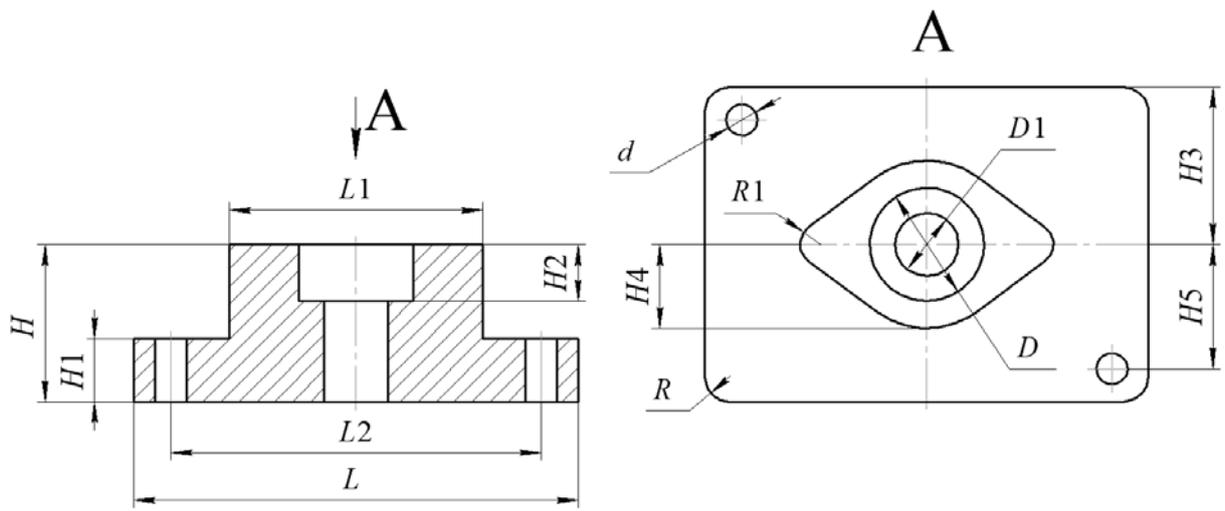


Рисунок 3.2 – Варианты 3 и 4, станок: BOSCH с системой ЧПУ M8

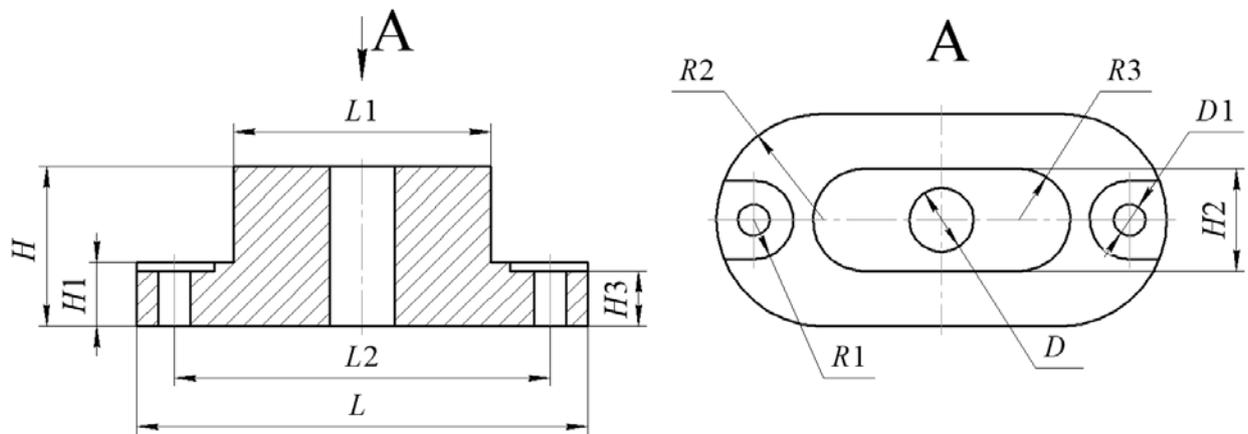


Рисунок 3.3 – Варианты 5 и 6, станок: Fanuc 3000

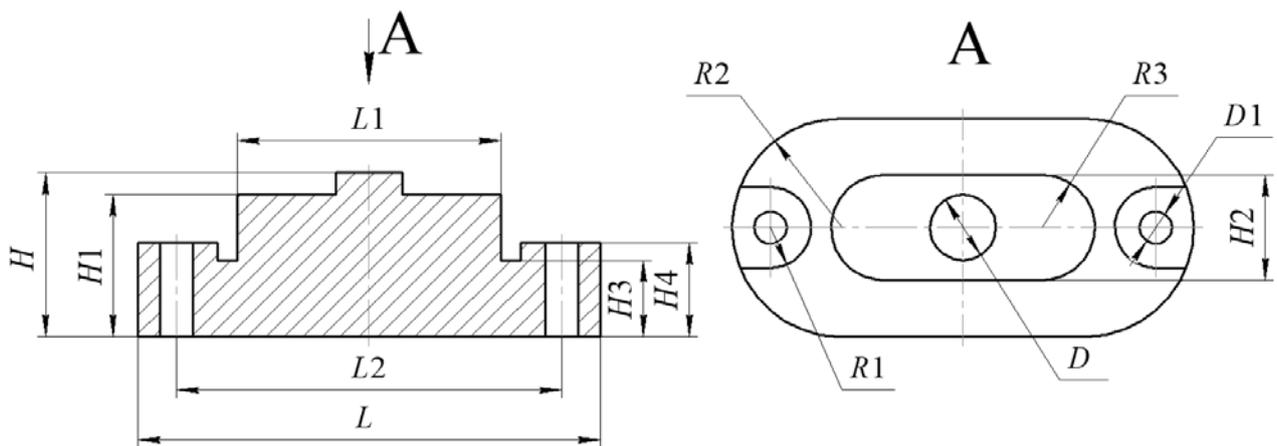


Рисунок 3.4 – Варианты 7 и 8, станок: H33

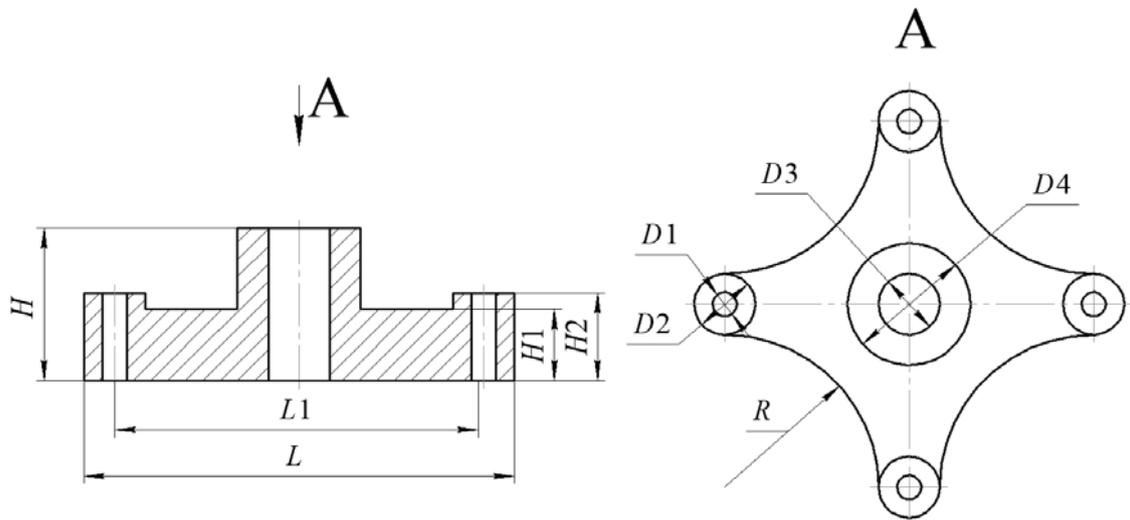


Рисунок 3.5 – Варианты 9 и 10, станок: BOSCH

Контрольные вопросы

- 1 Что такое САМ-системы, их назначение?
- 2 Какие САМ-системы вам известны?
- 3 Каков общий порядок подготовки УП для фрезерного станка с ЧПУ с использованием САМ-систем?
- 4 Какая технологическая информация задается в процессе подготовки управляющей программы?
- 5 Что такое симуляция? Для чего используется этот процесс?

4 Лабораторная работа № 4. Программирование электроэрозионного проволочно-вырезного станка с ЧПУ

Цель работы: ознакомление с работой системы подготовки программ для электроэрозионных проволочно-вырезных станков, с основными правилами описания деталей и составления программ для их изготовления.

4.1 Теоретические сведения

4.1.1 Описание и характеристики станка.

Станок модели DK7725 предназначен для точной обработки всех типов металлических изделий малых и средних размеров, а также изделий сложной формы, изготовленных из проводящих и твердых материалов (закаленная сталь, высокопрочные сплавы). Станок управляется в цифровом режиме и обеспечивает автоматическую и точную обработку изделия с широкими возможностями настройки. Таким образом, он может применяться для обработки оборудования, счетчиков, двигателей, тракторов, мотоциклов, бытовой техники, эксплуатироваться в испытательных мастерских и инструментальных цехах в таких отраслях как производство предметов первой необходимости.

Электроэрозионный станок с ЧПУ серии DK77 состоит из корпуса станка

(основания), координатного рабочего стола, системы нитепротяжки, стойки электродной нити, системы циркуляции рабочей жидкости, системы электроуправления и цифровой системы управления. Контроль перемещения координатного стола осуществляется цифровой системой контроля, которая обеспечивает точный ход заготовки и её продольное перемещение по направляющим и обеспечивает возвратно-поступательное передвижение нити по прямой с высокой скоростью. При обработке импульс электроразряда проходит между электродной нитью, рабочей жидкостью и заготовкой. В центре электроразрядного канала создается высокая температура, что приводит к плавлению металла и формированию эрозии, которая применяется для обработки заготовки.

Установленное на подвижной консоли стойки катушки конусное устройство при необходимости отклоняет электродную нить в направлении U и V . Благодаря взаимодействию четырех X , Y и U , V осей в вертикальном и горизонтальном направлении, существует возможность создавать профили с различными типами верхних и нижних поверхностей или же с конической поверхностью.

Корпус станка представляет собой железный закрытый корпус, по двум сторонам которого размещены электрический блок и блок рабочей жидкости, рабочий стол, системы протяжки нити, освещения и другие компоненты на внешней части станка.

Рабочий стол состоит из движущейся платформы, на которой он расположен, средней тяговой платформы, винтов точной настройки и редуктора (коробка передач). Вертикальный и горизонтальный ход тяговой платформы осуществляется по роликовым направляющим. Движение рабочего стола (подача заготовки) осуществляется посредством шагового двигателя через передаточный редуктор на шариковый винт точной настройки.

Винтовая передача обладает такими преимуществами как высокая точность, высокая эффективность и длительный срок эксплуатации.

Устройство протяжки нити выполняет прямолинейное возвратно-поступательное движение электродной нити. Катушка хранения нити состоит из полого цилиндра, который отделен от основной оси. Двигатель протяжки нити соединен с катушкой хранения нити посредством муфты. Вращение катушки придает нити скорость 11 м/с, которая передается на винтовую передачу точной настройки через шкив синхронной ленточной передачи, осуществляя ровное возвратно-поступательное движение нити. В винтовой передаче точной настройки применяются главные или вспомогательные винтовые гайки для регулировки зазоров. Частое изменение направления хода катушки нити контролируется рядом путевых переключателей. В случае сбоя в работе переключателя, изменения направления, путевой переключатель отключает питание и работа станка останавливается.

Стойка нитедержателя состоит из колонны, неподвижной консоли, подвижной консоли и направляющих колес. Фронтальная часть стойки нитедержателя оборудована блоком подачи питания из высокопрочного сплава, направляющего колеса с подачей питания от консоли стойки нитедержателя. В

конструкции реализован метод полностью открытой протяжки нити. Электродная нить выходит из катушки хранения проводной нити, направляется в главное направляющее колесо в голове стойки нитедержателя через верхние и нижние распыляющие форсунки. Перед началом эксплуатации следует проверить вертикальность расположения электродной нити.

Внизу колонны также находится клапан подачи эмульсии с двумя кранами для регулирования объема подачи воды верхней и нижней форсунки.

При вращении винтов вверх или вниз колонна перемещается. Для регулирования зазора между верхним и нижним главным направляющим колесом при обработке заготовки подвижная консоль может двигаться вверх или вниз.

4.1.2 Система программируемого управления AutoCut WEDM.

Система программируемого управления AutoCut WEDM (далее по тексту – система AutoCut) на базе операционной системы Windows XP состоит из: программного обеспечения системы (CAD и CAM), карты управления движением по четырем осям, платы драйвера шагового мотора и платы высокой частоты.

Пользователь использует CAD для построения графики обработки в соответствии с чертежом и задает WEDM-методы для графики CAD, создает данные двухмерной или трехмерной обработки и выполняет изготовление детали. Во время обработки система может интеллектуально управлять скоростью мотора, параметрами импульсного тока и т. д. Метод обработки на базе графического процессора представляет собой динамическую интеграцию CAD и CAM в поле WEDM.

В число функций системы входит автоматическое управление скоростью резания, отображение в режиме реального времени во время резания, предварительный просмотр обработки и др. При этом она способна обеспечить защиту от любых типов непредвиденных ситуаций (обрыв питания, остановка системы и т. п.) во избежание отбраковки изготавливаемой детали.

4.1.3 Использование AutoCut для AutoCAD.

Главный интерфейс представлен на рисунке 4.1.

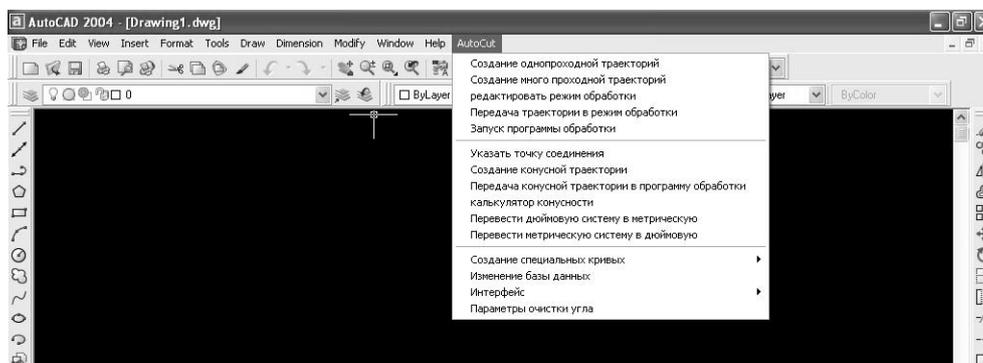


Рисунок 4.1 – Главный интерфейс AutoCAD с модулем AutoCut WEDM

4.1.4 Построение кривой WEDM.

Функция «Построение кривой WEDM» в AutoCut для AutoCAD включает в себя построение архимедовой спирали, циклоиды, гиперболы, эвольвенты и параболы, шестерни и т. д.

Архимедова спираль. Щелкните команду «Построение архимедовой спирали» в подменю «Построение кривой» в меню «AutoCut», появится диалоговое окно построения архимедовой спирали. После ввода параметров архимедовой спирали нажмите кнопку «ОК» для завершения построения архимедовой спирали.

Параметрическое уравнение архимедовой спирали:

$$\begin{cases} x = r \cdot t \cdot \cos(t); \\ y = r \cdot t \cdot \sin(t). \end{cases} \quad (4.1)$$

Параметры включают в себя параметр t , значение параметра r , угол вращения и координату базовой точки архимедовой спирали в пространстве чертежа.

Парабола. Щелкните команду «Построение параболы» в подменю «Построение кривой» в меню «AutoCut», появится диалоговое окно построения параболы. После ввода параметров параболы нажмите кнопку «ОК» для завершения построения параболы. Параметрическое уравнение параболы: $y = k \cdot x^2$, параметры включают в себя диапазон координаты x и значение коэффициента k . Кроме того, пользователь может задать вращение и параллельное перемещение параболы в пространстве чертежа.

Эвольвента. Щелкните команду «Построение эвольвенты» в подменю «Построение кривой» в меню «AutoCut», появится диалоговое окно построения эвольвенты. После ввода параметров эвольвенты нажмите кнопку «ОК» для завершения построения эвольвенты.

Параметрическое уравнение эвольвенты:

$$\begin{cases} x = r \cdot (\cos(t) + t \cdot \sin(t)); \\ y = r \cdot (\sin(t) + t \cdot \cos(t)). \end{cases} \quad (4.2)$$

Параметры включают в себя: радиус базовой окружности, угол развертки и угол вращения эвольвенты в пространстве чертежа, положение центра базовой окружности.

Гипербола. Щелкните команду «Построение гиперболы» в подменю «Построение кривой» в меню «AutoCut», появится диалоговое окно построения гиперболы. После ввода параметров нажмите кнопку «ОК» для завершения построения гиперболы.

Параметрическое уравнение гиперболы:

$$\begin{cases} x = \frac{a}{\cos(t)}; \\ y = b \cdot \operatorname{tg}(t). \end{cases} \quad (4.3)$$

Параметры включают в себя: a , b , диапазон параметра $t - t_1 \sim t_2$ ($t_1 < t < t_2$). Кроме того, пользователь может задать угол вращения гиперболы в пространстве чертежа и положение базовой точки.

Шестерня. Щелкните команду «Построение зубчатого колеса» в подменю «Построение специальной кривой» в меню «AutoCut», появится диалоговое окно построения зубчатого колеса (рисунок 4.2). После ввода параметров и просмотра зубчатого колеса нажмите кнопку «ОК» для завершения построения зубчатого колеса.

Рисунок 4.2 – Шестерня

Фигурный текст. Щелкните команду «Фигурный текст» в подменю «Построение специальной кривой» в меню «AutoCut», появится диалоговое окно построения векторного текста (рисунок 4.3).

Наберите текст в текстовом редакторе «Текст», нажмите кнопку «Предварительный просмотр» рисунок появится на черном поле диалогового окна, нажмите кнопку «Вставить» для полной вставки векторного текста в главный интерфейс.

4.1.5 Построение траектории.

Есть три способа построения траектории в модуле AutoCAD WEDM: создание траектории обработки, создание траектории многократной обработки и создание траектории конуса.

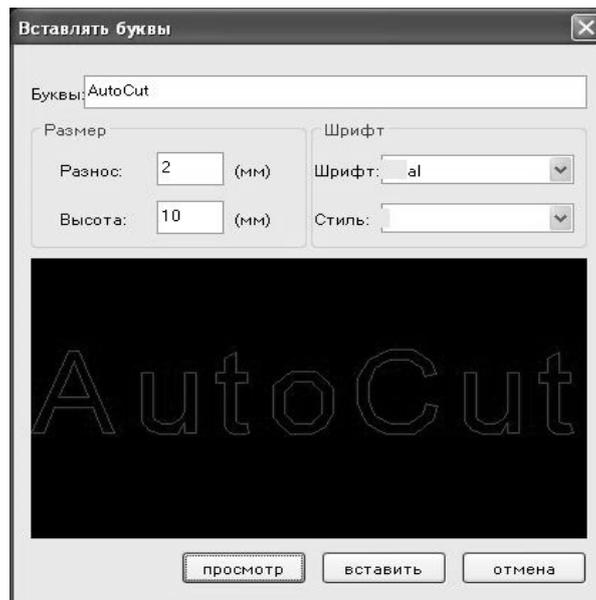


Рисунок 4.3 – Векторный текст

Создание траектории обработки. Щелкните пункт «Создание траектории обработки» в меню «AutoCut», появится диалоговое окно (рисунок 4.4) с параметрами для создания траектории высокоскоростной вырезки.

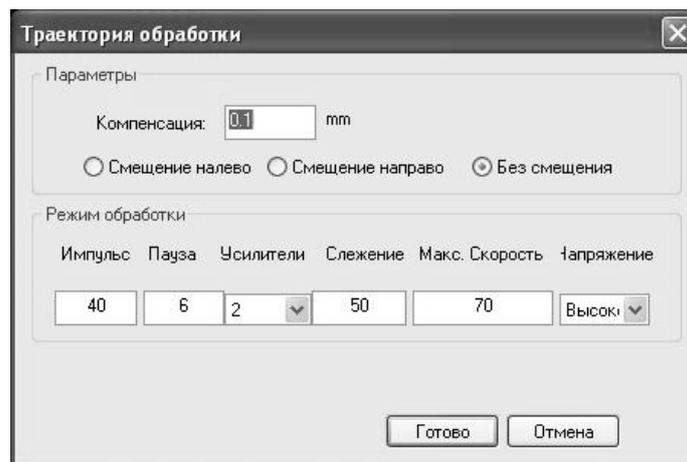


Рисунок 4.4 – Траектория высокоскоростной обработки: задание параметров процесса

После выбора направления коррекции задайте значение коррекции и параметр, нажмите кнопку «ОК». В командной строке появится «Введите точку старта». Пользователь может ввести координаты точки старта с помощью ручного ввода относительных или абсолютных координат, либо может с помощью щелчка левой кнопкой мыши выбрать точку на экране в качестве точки старта. После подтверждения заправки в командной строке появится «Введите точку резания».

Внимание: точка резания должна находиться в пределах чертежа, в противном случае она будет не действительной.

Пользователь может ввести координаты точки резания с помощью ручного ввода или с помощью мыши выбрать одну точку на чертеже в качестве точки

резания. После подтверждения точки резания в командной строке появится «Выберите направление резания, нажмите <Enter>».

Пользователь может видеть альтернативные преобразования, обозначенные красной и зеленой стрелками на траектории обработки при перемещении мыши. Щелкните левой клавишей мыши на зеленой стрелке для подтверждения направления резания или нажмите клавишу <Enter> для завершения выбора направления траектории обработки. Направлением траектории будет направление зеленой стрелки.

Внимание: в случае замкнутой фигуры описанной выше процедурой пользователь может завершить создание траектории.

Но в случае незамкнутой фигуры после описанной выше процедуры в командной строке появится «Введите конечную точку, нажмите <Enter>» (рисунок 4.5).

Введите вручную или выберите мышью одну точку в качестве координат точки выхода, либо нажмите клавишу <Enter> для завершения выбора точки выхода (по умолчанию точка выхода и точка заправки являются одной и той же точкой). Создание траектории обработки незамкнутой фигуры завершено.

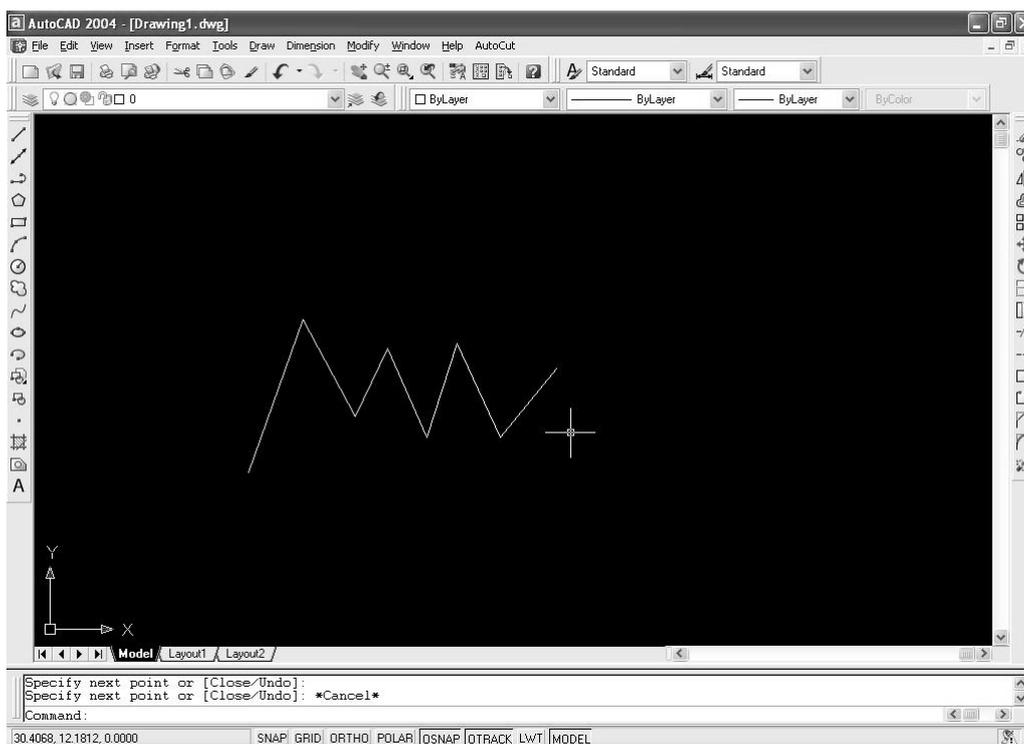


Рисунок 4.5 – Незамкнутая фигура: создание траектории обработки

4.1.6 Траектория обработки.

В модуле AutoCAD WEDM предусмотрены три метода обработки траектории: отправить траекторию обработки непосредственно в управляющую программу AutoCut через программу AutoCAD, отправить задание обработки конуса в управляющую программу AutoCut, запустить управляющую программу AutoCut непосредственно и загрузить файл выполнения детали в управляющую программу.

Передача траектории в режим обработки. Щелкните пункт «Передача траектории в режим обработки» в меню «AutoCut», появится диалоговое окно «Выбор карты» (рисунок 4.6).

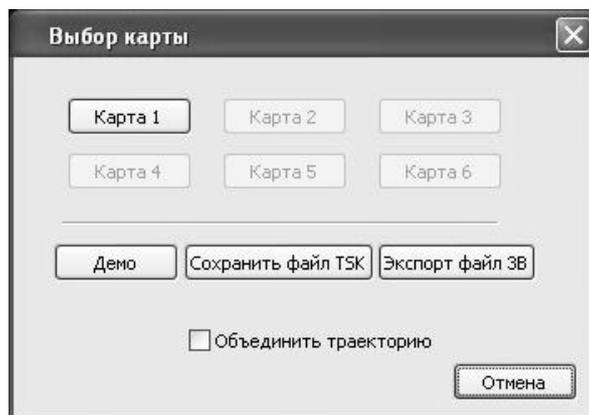


Рисунок 4.6 – Выбор карты

Нажмите кнопку «Карта 1» (в отсутствие карт управления пользователь может выбрать «Демонстрационная карта» для демонстрации), в командной строке программы AutoCAD появится «Выберите объект» выберите розовую траекторию левой кнопкой мыши

Щелкните правой клавишей мыши или <Enter>, откроется следующий интерфейс управления (рисунок 4.7).

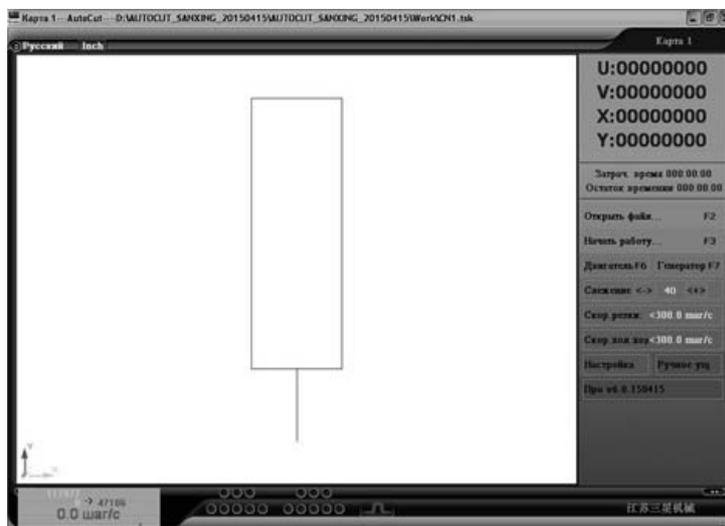


Рисунок 4.7 – Главный интерфейс управляющей программы

4.2 Порядок выполнения лабораторной работы

1 Получить индивидуальное задание – вариант набора параметров (таблица 4.1) и чертеж детали (рисунок 4.8).

2 Проанализировать чертеж детали.

3 В режиме «2D» построить контур заготовки и обрабатываемых поверхностей в плоскости X–Y.

4 Определить характерные точки начала и конца обработки.

5 Задать технологические параметры переходов электроэрозионной операции.

6 Выполнить операции создания контура и передачи контура на обработку.

7 Произвести обработку. Измерить полученные размеры элементов детали и сравнить с исходными.

Таблица 4.1 – Варианты заданий

Вариант	Элемент <i>A</i>	Элемент <i>B</i>	Элемент <i>C</i>	Элемент <i>D</i>
1	$3 \times 45^\circ$	$R3$	$R4$	6×3
2	$R3$	$3 \times 45^\circ$	6×3	$R4$
3	$R4$	6×3	$3 \times 45^\circ$	$R3$
4	6×3	$R4$	$R3$	$3 \times 45^\circ$
5	7×4	$R5$	$4 \times 45^\circ$	$R2$
6	$4 \times 45^\circ$	7×4	$R5$	$R6$
7	$R5$	$R6$	7×4	$4 \times 45^\circ$

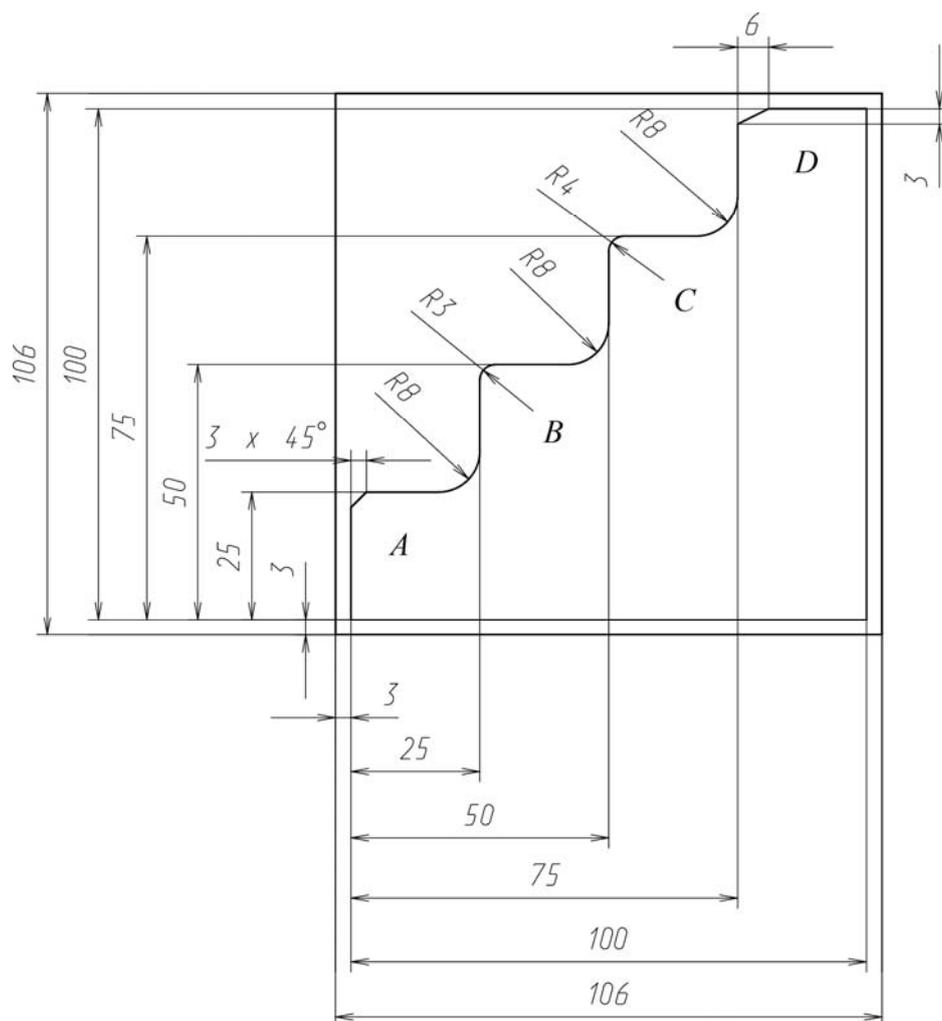


Рисунок 4.8 – Эскиз проектируемой детали

Контрольные вопросы

- 1 Какими способами можно определить контур спирали?
- 2 Какими способами можно определить контур параболы?
- 3 Какими способами можно определить контур эвольвенты?
- 4 Какими способами можно определить контур гиперболы?
- 5 Какими способами можно определить контур шестерни?
- 6 Какими способами можно определить контур фигурного текста?

5 Лабораторная работа № 5. Программирование токарно-фрезерной обработки в САМ-системе

Цель работы: овладение практическими навыками автоматизированного проектирования управляющих программ для многоцелевых станков с ЧПУ с использованием САМ-системы.

5.1 Порядок выполнения лабораторной работы

Рассмотрим последовательность проектных операций процедуры разработки управляющей программы для токарно-фрезерной обработки с использованием системы SprutCAM. В качестве иллюстрации примеров выполнения отдельных задач будет использована деталь, эскиз которой представлен на рисунке 5.1.

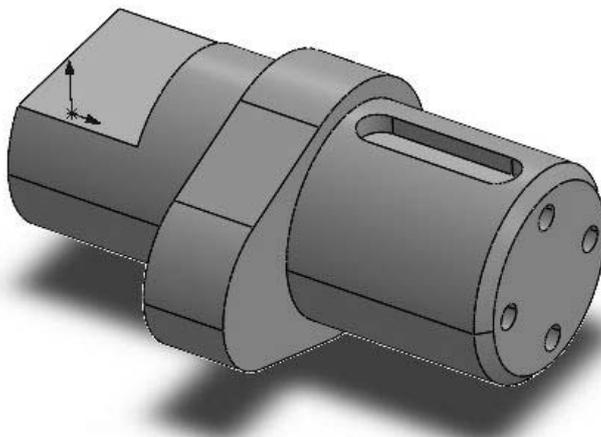


Рисунок 5.1 – Трехмерный эскиз детали

Загрузить программу SprutCAM двойным щелчком по ярлыку SprutCAM на рабочем столе.

Загрузить учебник двойным щелчком по ярлыку Учебник SprutCAM на рабочем столе.

Из меню Учебника SprutCAM выбрать раздел Токарная-фрезерная обработка.

После выбора урока выполните последовательно все предусмотренные в нем упражнения, перемещаясь по ним с помощью кнопок навигации.

Задание

1 Получить индивидуальное задание – вариант детали и номер набора параметров.

2 Проанализировать чертеж детали. Выделить контуры заготовки и обрабатываемых поверхностей и присвоить им имена.

3 В режиме «3D» построить контуры заготовки и обрабатываемых поверхностей.

4 Определить последовательность переходов токарно-фрезерной операции.

5 Задать технологические параметры переходов токарно-фрезерной операции (режущий инструмент, режимы резания, подвод, отвод инструмента, стратегию).

6 Выполнить генерацию программы в виде последовательности команд языка CLDATA.

7 Выбрать режим «Постпроцессор». Установить заданную систему ЧПУ и сгенерировать управляющую программу.

Контрольные вопросы

1 Основные типы САМ-систем?

2 Каков общий порядок подготовки УП для многоцелевого станка с ЧПУ с использованием САМ-систем?

3 Какая технологическая информация задается в процессе подготовки управляющей программы?

4 Что означает термин «постпроцессирование»?

6 Лабораторная работа № 6. Проектирование операций с применением промышленных роботов в среде офлайн-программирования

Цель работы: изучение методики создания моделей роботизированных технологических комплексов на основе цифровых прототипов основного и вспомогательного технологического оборудования и их офлайн-программирования.

6.1 Теоретические сведения

Управление роботами производства компании ABB Group (Швеция) обеспечивается блоком модулей управления (Control module) (рисунок 6.1) связанным с силовыми блоками управления отдельными роботами (Drive

module). Блок модулей управления обеспечивает управление синхронной работой нескольких роботов (до четырех) с помощью функции MultiMove.

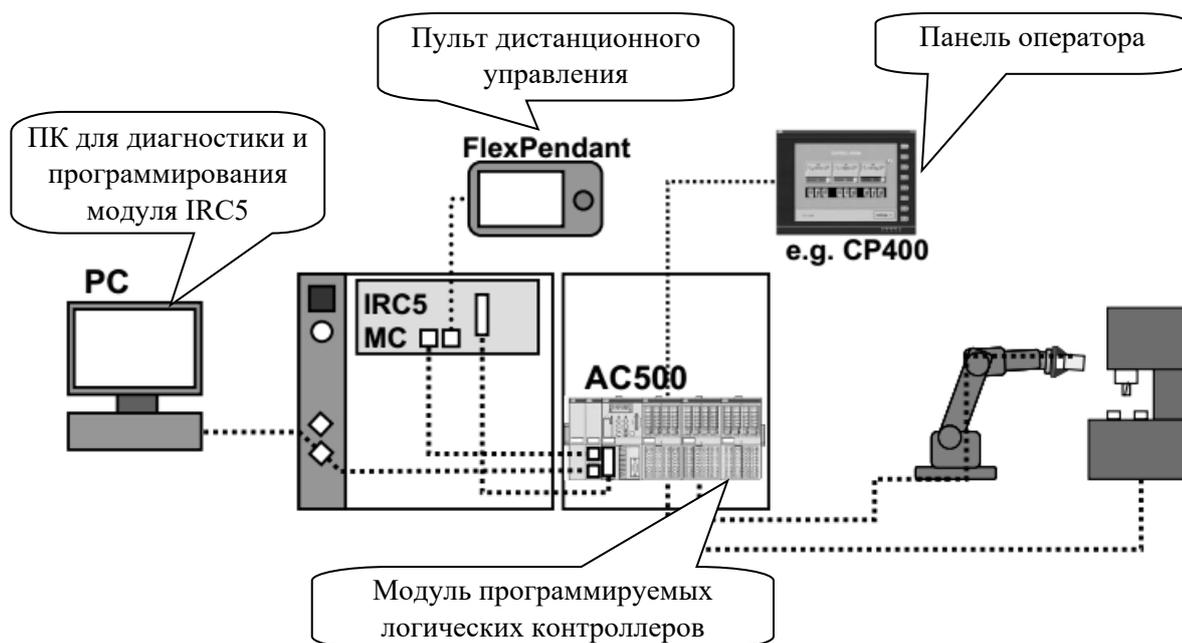


Рисунок 6.1 – Общая концепция использования модуля IRC5 для управления роботом и технологическим оборудованием

При создании блока модулей управления пятого поколения использована новая концепция модульного проектирования. Модульная конструкция блока управления (Control module) обеспечивает не только управление роботами, но и управлением технологическим процессом. Подобная гибкость делает возможным создавать оптимальные компоновки гибких роботизированных комплексов, а также выполнять модернизацию или производить замену одного из модулей при минимальном вмешательстве в другие модули.

Для исключения простоя оборудования при программировании роботов на выполнение других операций к модулю IRC5 может быть подключен персональный компьютер для офлайн-программирования РТК на языке RAPID.

Для управления технологическим оборудованием и анализа его состояния к блоку управления могут быть подсоединены различные аналоговые и цифровые устройства, которые обеспечивают включение и выключение отдельных механизмов станков, конвейеров и выполняют оценку состояния этих механизмов после исполнения управляющих команд. Для этих целей блок управления стандартной комплектации имеет 16 портов ввода-вывода цифровых сигналов (Digital I/O 16 DI/16 DC), восемь портов релейных сигналов для включения контакторов и реле (8 DI /8 DO relay) и четыре аналоговых канала ввода-вывода (Analog 4 AI/4 AO).

Программирование модуля управления IRC5 в условиях гибкого производства для исключения простоев оборудования выполняется в режиме офлайн с использованием виртуального контроллера. С целью сокращения трудоемкости программирования гибких роботизированных комплексов

концерном ABB Group разработана система моделирования и программирования РТК RobotStudio.

RAPID-программа содержит данные и инструкции, которые управляют роботом и периферийным оборудованием. RAPID-программа обычно состоит из четырех частей:

- 1) главная программа, которая вызывает для выполнения подпрограммы;
- 2) нескольких подпрограмм, которые разделяют программу на меньшие части, вызываемые из главных или других подпрограмм;
- 3) условно-постоянные данные программы, которые используются для определения положения элементов РТК, числовых значений и систем координат;
- 4) системные модули: программы, которые постоянно находятся в оперативной памяти и предназначены для управления исполнительными механизмами робота.

При моделировании производственного процесса в среде RobotStudio формирование программы на языке RAPID происходит автоматически. При необходимости в программу могут быть внесены изменения с использованием редактора системы программирования RAPID Editor.

Как и большинство языков программирования высокого уровня RAPID содержит функции определения переменных, массивов и констант (Var, Dim, Const), математические функции, функции организации циклов и условных переходов (While, Do, For, If, GoTo), синтаксис которых подобен синтаксису языка Basic.

Специфическими особенностями языка RAPID является наличие в нем специальных функций – инструкций для управления перемещениями руки робота, периферийного оборудования и обработки сигналов цифровых и аналоговых датчиков.

6.2 Порядок выполнения лабораторной работы

6.2.1 Создание компоновки РТК на основе цифровых прототипов технологического оборудования.

Необходимо скопировать файлы цифровых прототипов заготовки и кассеты **Rolling.sat**, **Cartridge.sat** из своей рабочей папки в папку **Geometry**.

Для использования оригинальных цифровых прототипов оборудования необходимо скопировать файлы **Lathe.rslib** и **Gripper.rslib** из своей рабочей папки в папку **Equipment**.

Загрузить систему **RobotStudio** и создать РТК на базе шаблона **IRB 2600_12kg_1.8m** в соответствии с компоновкой, показанной на рисунке 6.2.

Установить силовой блок и блок контроллера **IRC5** в пространстве станции на безопасном расстоянии от робота.

Из раздела меню **Home** командой **Import Library** из библиотеки цифровых прототипов **Equipment** на площадку РТК установить станок, файл **Lathe.rslib**. Если не все элементы конструкции станка отображаются на экране, для их визуализации используйте команду **Visible** из контекстного меню узла **Lathe**.

Командой **Set Position** определить положение станка с координатами $X = 1400$, $Y = 1152$, $Z = 0$, угол поворота вокруг оси $Z = 180$ град.

Командой **Import Library** из библиотеки цифровых прототипов **Equipment** загружаем файл схвата **Gripper.rslib**. Для точной установки схвата на фланец робота на вкладке **Layout** выбираем узел **Gripper** и из контекстного меню выполняем команду **Attach to...** (Присоединить к...). В открывшемся списке жмем пункт меню **IRB 2600_12_1.8**. На запрос «Следует ли сохранить текущую позицию инструмента» отвечаем **Нет**. Перемещением и вращением изображения РТК оцениваем точность установки схвата.

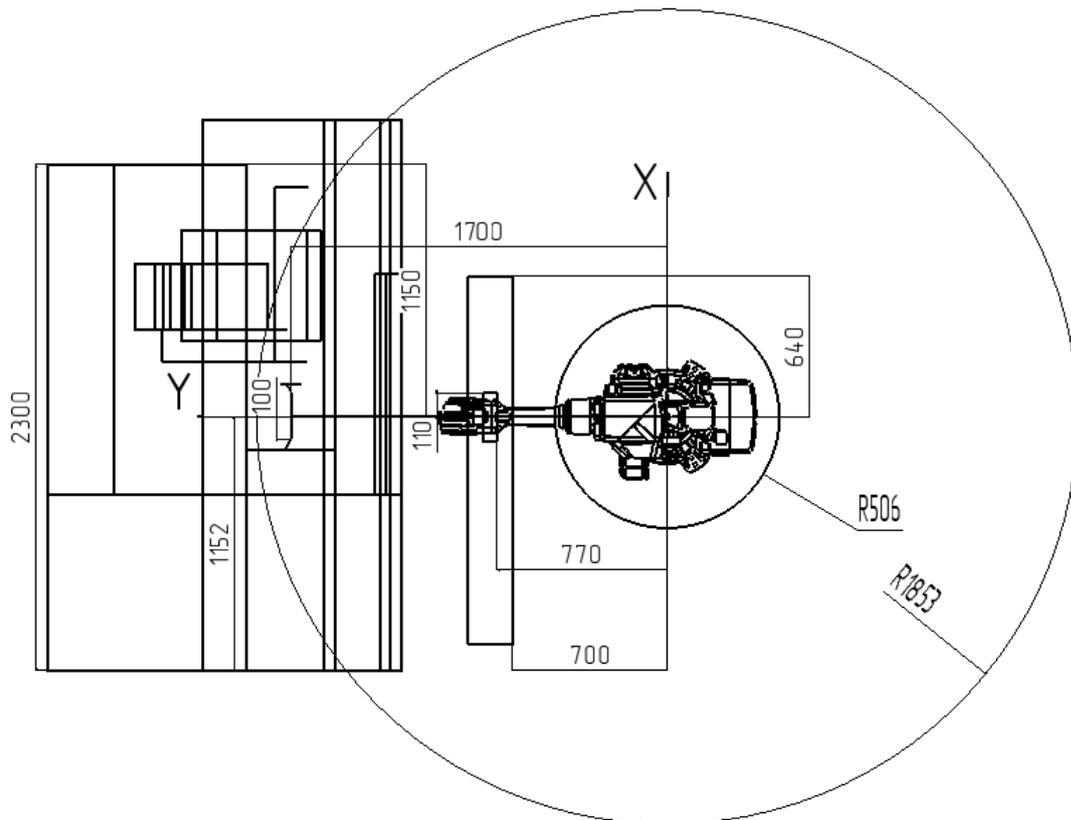


Рисунок 6.2 – Компоновка РТК на базе робота IRB 2600_12kg_1.8m

Для имитации секции шагового конвейера создадим параллелепипед с размерами $4000 \times 200 \times 500$ мм. Из раздела меню **Modeling** командой **Solid** → **Box** строим параллелепипед, начиная из точки с координатами $X = 800$, $Y = -1000$, $Z = 0$ по заданным размерам $Length = 200$, $Width = 4000$, $Height = 500$. В закладке **Layout** на дереве проекта узел **Part_1** переименовываем в **Conveyor**.

Из библиотеки цифровых прототипов **Geometry** командой **Import Geometry** загружаем файл модели кассеты **Cartridge.sat**. Перемещаем модель кассеты на конвейер командой **Set Position**, задавая координаты $X = 850$, $Y = 110$, $Z = 500$, угол поворота вокруг оси $X = 90$ град. Чтобы изображения конвейера и картриджа не сливались, перекрасим их командой **Set Color** из контекстного меню узлов **Conveyor** и **Cartridge** дерева проекта.

Аналогичным образом загружаем файл модели заготовки **Rolling.sat** и размещаем заготовку в кассете с координатами $X = 885$, $Y = 100$, $Z = 550$, угол поворота вокруг оси $X = 90$ град.

Для создания второй загрузочной позиции на шаговом конвейере командой **Import Geometry** загружаем файл модели кассеты **Cartridge.sat**, а затем перемещаем модель кассеты на конвейер командой **Set Position**, задавая координаты $X = 850$, $Y = 500$, $Z = 500$, угол поворота вокруг оси $X = 90$ град. Аналогичным образом загружаем модель второй заготовки **Rolling.sat** и размещаем заготовку в кассете с координатами $X = 885$, $Y = 490$, $Z = 550$, угол поворота вокруг оси $X = 90$ град. Заготовка должна разместиться между крайними упорами кассеты.

Перемещением, увеличением и вращением изображения оцениваем точность установки компонентов РТК.

6.2.2 Создание опорных точек траектории для создания пути схвата.

Для упрощения задачи определения координат при задании опорных точек (**Target**) пути схвата робота (**Path**) создадим систему координат рабочего объекта **Workobject** для заготовки. Повернем станцию в графическом окне так, чтобы ее конвейер был ориентирован, как показано на рисунке 6.3. Для точного определения системы координат объекта включаем привязку к центру окружности **Snap Center** в верхней части графического окна системы и выбираем на закладке **Layout** узел **Rolling**.

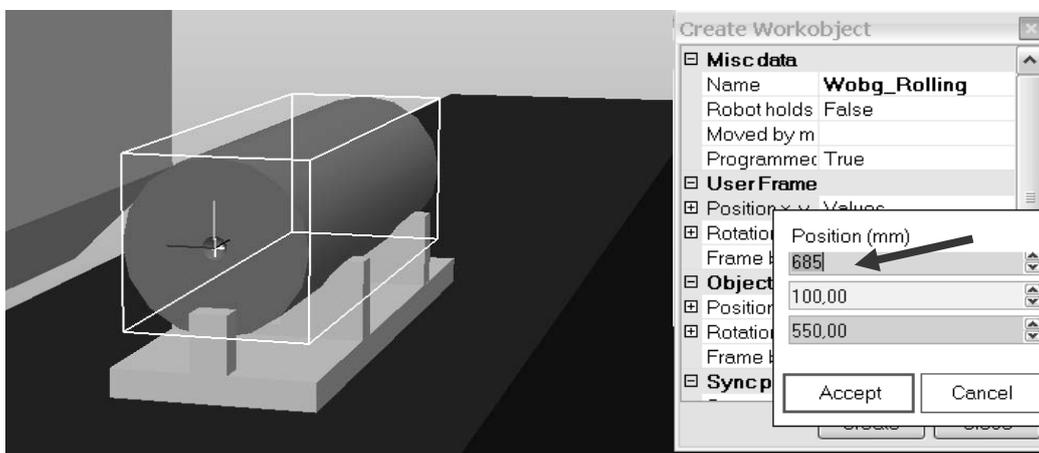


Рисунок 6.3 – Положение заготовки при создании рабочей системы координат **Workobject**

На вкладке **Home** в подразделе **Path Programming** (программирование пути) нажимаем кнопки **Other** → **Create WorkObject**. В появившемся окне **Create WorkObject** (Создание рабочего объекта) в поле **Name** (Имя) присваиваем имя **Wobj_Rolling**, создаваемому рабочему объекту. В поле **Position X, Y** раздела меню **User Frame** щелкаем справа кнопку выпадающего списка, устанавливаем курсор в первую позицию поля **Position (mm)** для ввода координаты X (см. рисунок 6.3). На изображении заготовки, используя

объектную привязку, находим центр торца и щелкаем левой клавишей мыши. Для завершения процедуры щелкаем кнопки **Accept**, а затем **Create**.

На основе компоновки РТК и размеров конвейера, картриджа, заготовки (рисунок 6.4) направления их осей координат, определяем координаты опорных точек (**Target**) пути схвата при загрузке и выгрузке заготовки со станка (таблица 6.1).

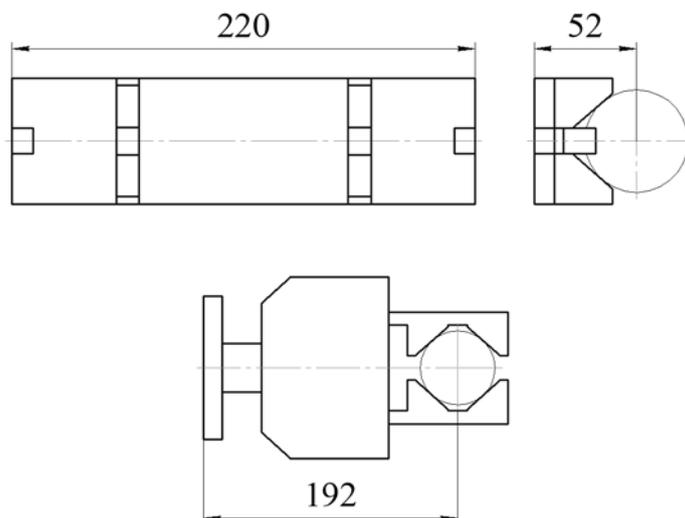


Рисунок 6.4 – Положения заготовки при создании рабочей системы координат

Таблица 6.1 – Координаты опорных точек (**Target**) пути схвата в системе координат заготовки **Wobj_Rolling**

Имя точки	Значение координаты в Wobj_Rolling			Угол поворота вокруг оси объекта Wobj_Rolling		
	X	Y	Z	Rx	Ry	Rz
Target_10	0	-100	192	180	0	180
Target_20	0	-100	282	180	0	180
Target_30	720	-100	545	180	0	180
Target_40	620	-100	595	180	0	180
Target_50	20	-100	595	180	0	180

Для задания точек траектории (**Target**) выбираем рабочий объект **Wobj_Rolling** на закладке **Path&Targets**, открываем вкладку **Home** главного меню системы. В разделе **Path Programming** нажмем кнопки **Target** → **Create Target** (Цель, целевая, опорная точка). Вверху окна ввода данных **Create Target** в поле **Reference** из выпадающего списка выбираем в поле ссылку на системы координат рабочих объектов **WorkObject**. Внизу окна в поле **WorkObject** выбираем рабочий объект **Wobj_Rolling** в системе координат, которого определены опорные точки. Заполняем поля **Position** значениями координат целевой точки **Target_10** $X = 0$, $Y = -100$, $Z = 192$. В этом случае точка захвата заготовки будет располагаться по ее центру на оси (рисунок 6.5). Для того, чтобы схват в точке захвата заготовки располагался требуемым

образом (вертикально) и оси его системы координат совпадали с осями координат фланца робота, задаем углы поворота вокруг осей $R_x = 180^\circ$, $R_y = 0^\circ$, $R_z = 180^\circ$. Нажимаем кнопку **Add** – Добавить точку.

В закладке **Path&Targets** на дереве проекта должна появиться точка под именем **Target_10** (рисунок 6.5).

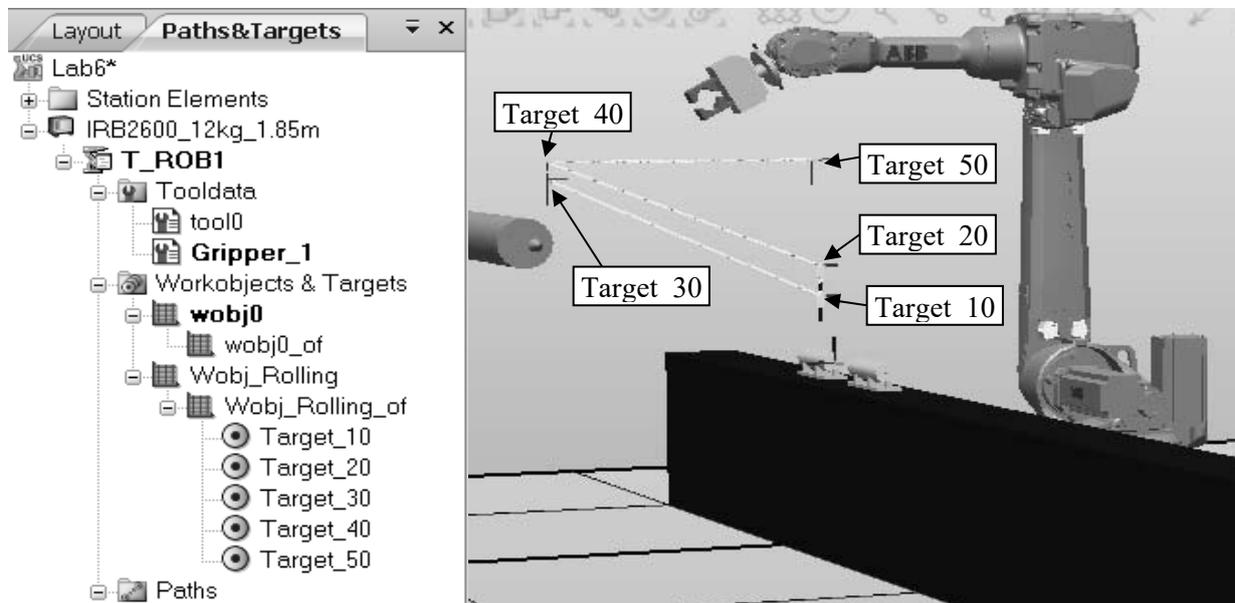


Рисунок 6.5 – Задание точек **Target** траектории перемещения схвата и его пути

Не закрывая окно **Create Target**, аналогичным образом последовательно вводим координаты и углы поворота для создания других опорных точек пути схвата (см. таблицу 6.1). После задания координат всех пяти точек нажимаем кнопку **Create**. Закроем окно **Create Target** нажатием кнопки **Close**.

Для оценки пространственной ориентации схвата в заданных точках откроем закладку **Layout** и отключим видимость станины станка и двери кабины командой **Visible** (Видимый) из контекстного меню объектов **Lathe**, **L1**, **L2** на дереве проекта.

Открываем закладку **Path&Targets** и командой **View Tool at Target** (Вид на инструмент в выбранной точке), из контекстного меню узлов точек, последовательно проверим ориентацию схвата в точках **Target_10**, **Target_20**, **Target_30**, **Target_40**, **Target_50**.

Если в некоторых точках положение схвата неправильное, то корректировку положения можно выполнить командой **Modifi** → **Target** → **Set_Position** → **Parent** из контекстного меню узла точки на дереве проекта.

6.2.3 Создание пути схвата робота.

Для создания пути схвата, по ранее определенным точкам, нажимаем в разделе меню **Path Programming** кнопки **Path** → **Empty Path** (Пустой Путь). На дереве путей и точек **Paths&Targets** в разделе **Path** появляется пустой путь **Path_10**. Добавим в пустой путь **Path_10** целевые точки. Для этого выделяем в

меню **Paths&Targets** первую целевую точку **Target_20**. В контекстном меню выбранной точки нажимаем пункт меню **Add to path** → **Path 10** → **<First>**. На дереве путей и точек **Paths&Targets** в разделе **Path_10** появляется участок пути к заданной точке **MoveL Target_20** (линейное перемещение в точку 10). Создайте полный путь командой **Add to path** → **Path 10** → **<Last>**, добавляя точки в последовательности, показанной на рисунке 6.6.

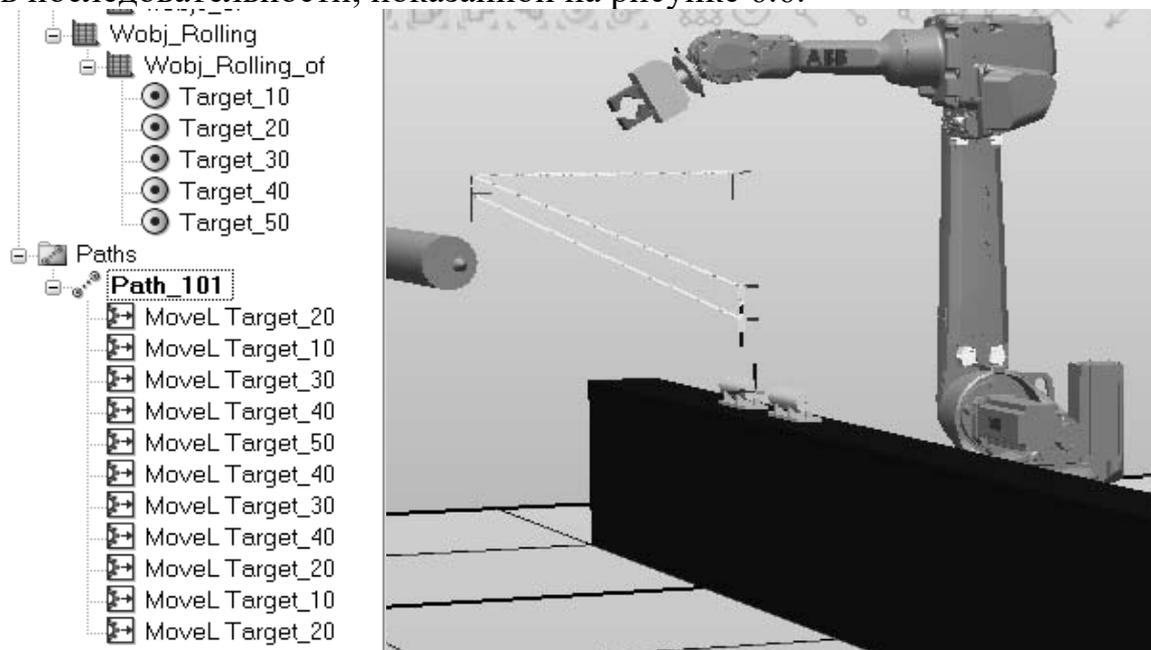


Рисунок 6.6 – Создание пути схвата Path_10

Для конфигурирования звеньев руки робота в каждой точке пути на дереве проекта выбираем путь **Path_10** и из контекстного меню выполняем команду **Auto Configuration**. Если после автоматического конфигурирования руки робота не появилось предупреждающих сообщений о невозможности создать конфигурацию руки робота в некоторых точках, выполняем синхронизацию созданной конфигурации с виртуальным контроллером командой **Synchronization to VC** из контекстного меню узла **Path_10**. Эту команду необходимо всегда выполнять после изменения конфигурации робота. Для проверки созданного пути инструмента выбираем вкладку **Simulation** (Моделирование), раздел меню **Configure** (Формирование) и нажимаем пункт меню **Simulation Setup** (Установки Моделирования).

Появляется окно **Setup Simulation**. В списке **Available Procedures** (Доступные Процедуры) выделяем путь **Path_10** и нажимаем на стрелку влево, а затем кнопки **Apply** и **OK**.

В результате выполнения эти команд будет сформирована программа управляющая движением робота вдоль заданной траектории.

Для моделирования движения робота в меню **Simulation Control** (Контроль Моделирования) нажимаем кнопку **Play**. В графическом окне системы наблюдаем, как робот перемещается по заданному пути.4 Создание цифровых сигналов портов ввода-вывода для управления механизмами РТК и контроля их состояния.

Для управления периферийным основным и вспомогательным технологическим оборудованием РТК **назначаем порты вывода** цифровых выходных управляющих сигналов (Digital Output) контроллера (таблица 6.2).

В соответствии с таблицей 6.2 создадим сигналы управления основным и вспомогательным технологическим оборудованием.

Таблица 6.2 – Сигналы управления технологическим оборудованием РТК

Механизм	Команда управления	Сигнал	Логическое имя порта вывода
Станок	Включить	1	Do1Lathe
	Выключить	0	
Дверь кабины	Открыть	1	Do2Door
	Закрыть	0	
Пинопль	Подвести	1	Do3Pinole
	Отвести	0	
Схват	Открыть	1	Do4Gripper
	Закрыть	0	
Конвейер	Включить	1	Do5Conveyor
	Выключить	0	
Аварийный сигнал	Включить	1	Do6Failure
	Выключить	0	

Из группы меню **Offline** выполняем команду **Configuration Editor** и выбираем пункт **I/O**. В открывшемся окне слева выбираем пункт **Signal**. Щелкаем правой клавишей мыши в правой части окна и из контекстного меню выполняем команду **Add Signal**. В окне **Edit Signal(s)** выбираем пункт **Name** и в поле **Type: RAPID** вводим внутрисистемное логическое имя сигнала **Do2Door** (см. таблицу 6.2).

Для задания типа выходного сигнала выбираем пункт **Type of Signal** и в поле **Value** (значение) из выпадающего списка выбираем тип сигнала **Digital Output**. Для задания уровня доступа выбираем пункт **Access Level** и в поле **Value** из выпадающего списка выбираем **All**. Для завершения нажимаем кнопку **Ok**. Аналогичным образом создаем остальные сигналы в соответствии с данными в таблице 6.2.

Для контроля в ходе выполнения технологической операции состояния основного и вспомогательного технологического оборудования РТК **назначаем порты ввода** контроллера **IRC5** для цифровых сигналов датчиков контроля (таблица 6.3).

В соответствии с таблицей 6.3 создадим сигналы управления основным и вспомогательным технологическим оборудованием. Создание входных цифровых сигналов (Digital Input) выполняется таким же образом, как и выходных. Имена сигналов устанавливаем в соответствии с таблицей 6.3, а тип сигнала (Type of Signal) **Digital Output**.

Таблица 6.3 – Сигналы контроля состояния технологическим оборудованием РТК

Механизм	Состояние механизма	Сигнал	Логическое имя порта ввода
Станок	Включен	1	Di1Lathe
	Выключен	0	
Дверь кабины	Открыта	1	Di2Door
	Закрыта	0	
Пиноль	Подведена	1	Di3Pinole
	Отведена	0	
Схват	Открыт	1	Di4Gripper
	Закрыт	0	
Конвейер	Есть заготовка	1	Di5Conveyor
	Нет заготовки	0	

После задания всех сигналов выполняем «горячую» перезагрузку контроллера из меню **Offline** командой **Restart** → **Warmstart**.

6.2.4 Использование цифровых сигналов ввода-вывода для управления механизмами РТК.

Для задания определенной последовательности работы оборудования РТК может быть составлен алгоритм управления основным и вспомогательным технологическим оборудованием. В алгоритме предусматривается выполнение функций управления оборудованием и контроля его состояния для исключения аварийных ситуаций и предотвращения выполнения операции без загрузки заготовки на станок.

Для срабатывания механизмов, в зависимости от значений управляющих выходных цифровых сигналов (Digital Output), свяжем их с перемещениями механизмов (таблица 6.4).

Таблица 6.4 – Выходные сигналы порта контроллера

Назначение выходного сигнала	Name (Внутрисистемное имя для RAPID)	Type of Signal (Тип сигнала)	Access Level (Уровень доступа)
Открыть – Закрыть дверь	Do2Door	Digital Output	All
Подвести – Отвести пиноль	Do3Pinole	Digital Output	All
Открыть – Закрыть левый схват	Do4GripperL	Digital Output	All
Открыть – Закрыть правый схват	Do4GripperR	Digital Output	All
Присоединить – отсоединить заготовку в картридже	AttachK	Digital Output	All
Присоединить – отсоединить заготовку в центрах	AttachS	Digital Output	All

Открываем меню **Simulation** и выполняем команду **Event Manager**. В окне **Event** нажимаем кнопку **Add**, в открывшемся окне устанавливаем в поле **Activation** значение **On** и нажимаем кнопку **Next**. В открывшемся после этой команды окне в списке справа выбираем сигнал управления дверью **Do2Door** и устанавливаем триггер **Signal is True** в **1** (открытие двери, см. таблицу 6.4) и нажимаем кнопку **Next** (далее). Затем определяем тип действия путем выбора из списка **Set Action Type** пункта **Move Mechanism to Pose** (перемещение механизма в позицию) и нажимаем кнопку **Next**. В следующем окне в списке **Mechanism** выбираем **Lathe** (токарный станок), а в списке **Pose** выбираем открытие двери **DoorOpened** и жмем кнопку **Finish**.

Используя этот же сигнал **Do2Door**, связываем его с событием «Закреть дверь» **DoorClosed** в том же порядке, изменяя только состояние сигнала **Signal is True** с **1** на **0** и действие с **DoorOpened** на **DoorClosed** (таблица 6.5)

Таблица 6.5 – Параметры настройки выходных сигналов контроллера

Event (Событие)	Name (Имя сигнала)	Trigger Condition (Состояние Сигнала)	Set Action Type (Действие)	Mechanism (Механизм)	Pose (Положение)
1	2	3	4	5	6
Открыть дверь	Do2Door	1	Move Mechanism to Pose	Lathe	DoorOpened
Закреть дверь	Do2Door	0	Move Mechanism to Pose	Lathe	DoorClosed
Подвести пиноль	Do3Pinole	1	Move Mechanism to Pose	Lathe	PinoleOn
Отвести пиноль	Do3Pinole	0	Move Mechanism to Pose	Lathe	PinoleOff
Открыть схват левый	Do4GripperL	1	Move Mechanism to Pose	Gripper	Lever_L_On
Закреть схват левый	Do4GripperL	0	Move Mechanism to Pose	Gripper	Lever_L_Off
Открыть схват правый	Do4GripperR	1	Move Mechanism to Pose	Gripper	Lever_R_On
Закреть схват правый	Do4GripperR	0	Move Mechanism to Pose	Gripper	Lever_R_Off
Присоединить заготовку в картридже	AttachK	1	Attach Objekt	IRB2600	Target_10

Окончание таблицы 6.5

1	2	3	4	5	6
Отсоединить заготовку в картридже	AttachK	0	Detach Objekt	IRB2600	Target_10
Присоединить заготовку в центрах	AttachS	1	Attach Objekt	IRB2600	Target_30
Отсоединить заготовку в центрах	AttachS	0	Detach Objekt	IRB2600	Target_30

Повторяем эту процедуру для задания остальных сигналов, руководствуясь данными, которые приведены в таблице 6.5.

Перед созданием сигналов присоединения и отсоединения заготовки схват и руку робота необходимо переместить в соответствующую точку **Target_10**, **Target_30** (рисунок 6.7).

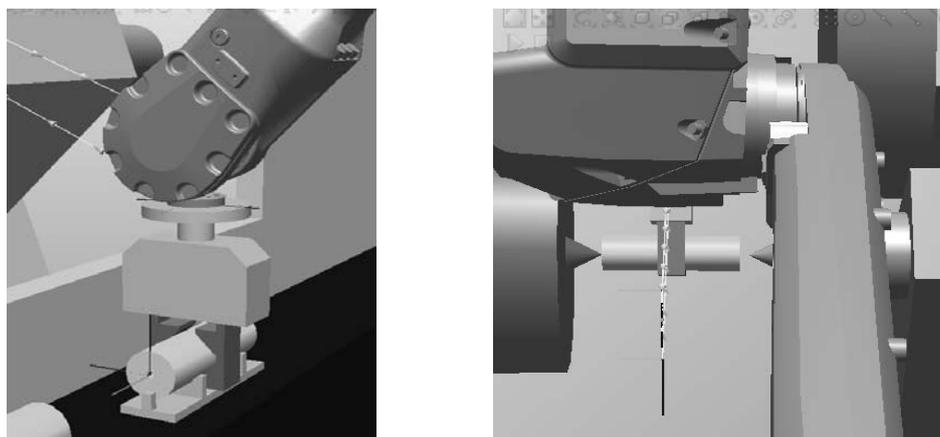


Рисунок 6.7 – Положение схвата и руки робота в точках **Target_10**, **Target_30**

При создании сигналов присоединения и отсоединения заготовки **AttachS** и **AttachR** в поле **Attach Object** (присоединяемый объект) выбираем **Rolling**, а в поле **Attach to** (Присоединить к) **IRB 2600**. Перед нажатием кнопки **Finish** необходимо установить опцию **Keep position** (Сохранить положение).

Для проверки действия созданных сигналов управления механизмами выбираем вкладку **Simulation**, нажимаем кнопку **I/O Simulator**. В окне управления сигналами портов вывода (**Outputs**) проверяем реакцию механизмов, изменяя значения сигналов с **0** на **1** и обратно.

6.2.5 Добавление сигналов ввода-вывода для управления механизмами РТК.

Созданные сигналы **ввода-вывода (I/O)** для управления дверью, пинолью станка, схватом могут быть добавлены на дереве путей **Patch&Targets** и затем на основе этих данных автоматически создается **RAPID**-программа управления РТК для выполнения технологической операции.

Для добавления сигналов ввода-вывода откроем раздел меню **Home**, закладку **Patch&Targets**, а затем на дереве проекта открываем узел **Path_10**.

Для добавления сигнала **Do4Gripper = 0** открытие схвата выбираем первый пункт пути **MoveL Target_20** и из контекстного меню выполняем команду добавить инструкцию **Add Action Instruction**. В окне **Create Action Instruction** (Создание инструкции управления, рисунок 6.8) заполняем поле **Instruction Templates** (Типовая инструкция управления), выбирая из выпадающего списка инструкцию задания значения *выходного* цифрового сигнала **SetDO**. В поле **Signal** из выпадающего списка выбираем логическое имя сигнала **Do4GripperR**, а в поле **Value** (значение) выбираем из выпадающего списка **0** и нажимаем кнопку **Create** (Создать).

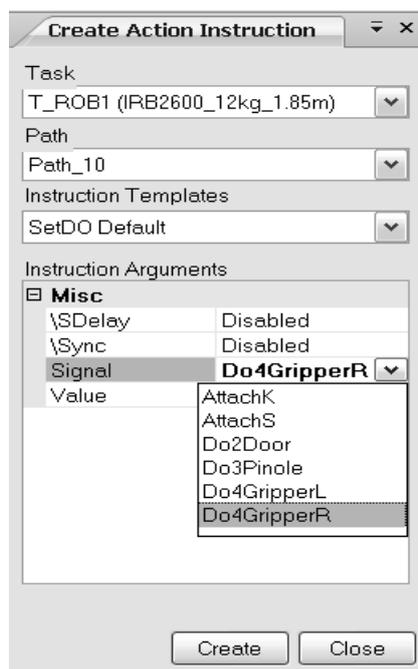


Рисунок 6.8 – Окно **Create Action Instruction** (Создание инструкции управления)

Не закрывая окно, добавляем следующий сигнал управления для проверки наличия заготовки в кассете **Di5Rolling = 1**. Заполняем поле **Instruction Templates** (Типовая инструкция управления), выбирая из выпадающего списка инструкцию анализа состояния *входного* цифрового сигнала **WaitDI** (Ожидание до тех пор, пока значение входного сигнала не станет равным заданному значению). В поле **Signal** из выпадающего списка выбираем логическое имя сигнала **Di5Rolling**, а в поле **Value** (значение) выбираем из выпадающего списка **1** и нажимаем кнопку **Create** (Создать).

Аналогичным образом в соответствующих узлах дерева пути **Path_10** добавляем остальные сигналы управления в соответствии со списком приведенным ниже.

- 1 Перемещение в точку 20 и выполнение команды на открытие схвата **Do4Gripper = 0**.
- 2 Проверка наличия заготовки в кассете **Di5Rolling = 1**.
- 3 Контроль открытия схвата **Di4Gripper = 1**.

4 Перемещение в точку 10 (MoveL Target_10) и закрытие схвата **Do4Gripper = 4.**

5 Присоединение заготовки к схвату **AttachK.**

6 Контроль закрытия схвата **Di4Gripper = 0.**

7 Перемещение в точку 20 (MoveL Target_20) для извлечения заготовки из кассеты.

8 Контроль состояния двери кабины в положении «Открыта» **Di2Door = 1.**

9 Если дверь закрыта $Di2Door = 0$ тогда, открыть дверь **Do2Door = 1.**

10 Контроль состояния пиноли станка в положении «Отведена» **Di3Pinole = 0.**

11 Перемещение схвата в точку 40 (MoveL Target_40).

12 Перемещение в точку 30 (MoveL Target_30) загрузки станка и подвод пиноли для закрепления заготовки **Do3Pinole = 1.**

13 Контроль состояния пиноли станка в положении «Подведена» **Di3Pinole = 1.**

14 Если (IF) пиноль подведена $Di3Pinole = 1$, тогда (THEN) открыть схват **Do4Gripper = 1.**

15 Контроль состояния схвата «Открыт» **Di4Gripper = 1.**

16 Отсоединение заготовки от схвата **AttachS.**

17 Перемещение схвата в точку 40 (MoveL Target_40).

18 Перемещение схвата из рабочей зоны станка в точку 50 (MoveL Target_50).

19 Закрытие двери кабины **Do2Door = 0.**

20 Обработка заготовки (задержка времени) **WaitTime 40 сек.**

21 Открытие двери кабины станка **Do2Door = 1.**

22 Контроль состояния двери кабины в положении «Открыта» $Di2Door = 1.$

23 Перемещение схвата в точку 40 (MoveL Target_40), а затем в точку 30 (MoveL Target_30).

24 Закрытие схвата **Do4Gripper = 4.**

25 Присоединение заготовки к схвату **AttachS.**

26 Отвод пиноли **Do3Pinole = 0.**

27 Контроль состояния пиноли станка в положении «Отведена» **Di3Pinole = 0.**

28 Перемещение схвата в точку 20 (MoveL Target_20) над кассетой.

29 Перемещение в точку 10 (MoveL Target_40), и открытие схвата **Do4Gripper = 0.**

30 Отсоединение заготовки от схвата **AttachK.**

31 Перемещение схвата в точку 20 (MoveL Target_20) над кассетой.

32 Проверка наличия заготовки в кассете **Di5Rolling = 1.**

33 Включение конвейера и перемещение его на шаг.

34 Далее рабочий цикл повторяется, начиная с пункта 2.

В тех случаях, когда условия контроля состояния оборудования не выполняются, происходит включение сигнала «Аварийный останов» $Do6 = 1$ и переход в программе происходит к точке включения аварийной сигнализации, например,

```

IF Di2Door = 1 THEN
MoveL Target_40,v7000,z20,Gripper_1\WObj:=Wobj_Rolling;
ELSE
GOTO <метка>
ENDIF

```

После ввода сигналов управления и контроля проверяем состояние контроллера. Если прямоугольник в нижнем правом углу окна RobotStudio желтый и статус контроллера **0/0**, открываем раздел меню **Offline** и выполняем перезагрузку контроллера командами **Restart** → **WarmStart**.

Для синхронизации заданного пути с контроллером открываем раздел меню **Home**, закладку **Patch&Targets** и из контекстного меню узла дерева пути **Patch_10** выполняем команду **Synchronization VC**.

Для конфигурирования пути на этапе offline-моделирования открываем раздел меню **Simulation** и выполняем команду **Simulation Setup**. Если путь **Patch_10** не подключен, подключаем его, выбрав его в правой части окна и нажав зеленую стрелку, направленную влево.

Поскольку при моделировании работы РТК используются виртуальные датчики контроля, то перед выполнением программы присвоим им значение 1. Для этого выполняем команду **I/O** и в открывшемся окне в разделе **Inputs** изменяем значения их с **0** на **1**.

Для проверки созданной программы и ее редактирования откроем раздел меню **Offline**. На дереве проекта открываем узел программного модуля **Path_10** и, выделив его щелчком мыши, нажимаем кнопку **Rapid Editor**. Откроется окно с текстом автоматически созданной программы.

Для того чтобы иметь возможность одновременно наблюдать выполнение программы и имитацию работы РТК под ее управлением, из контекстного меню заголовка окна **IRB2600_12kg...** (рисунок 6.9) выполняем команду **New Vertical Tab Group** для расположения всех открытых окон группой по вертикали.

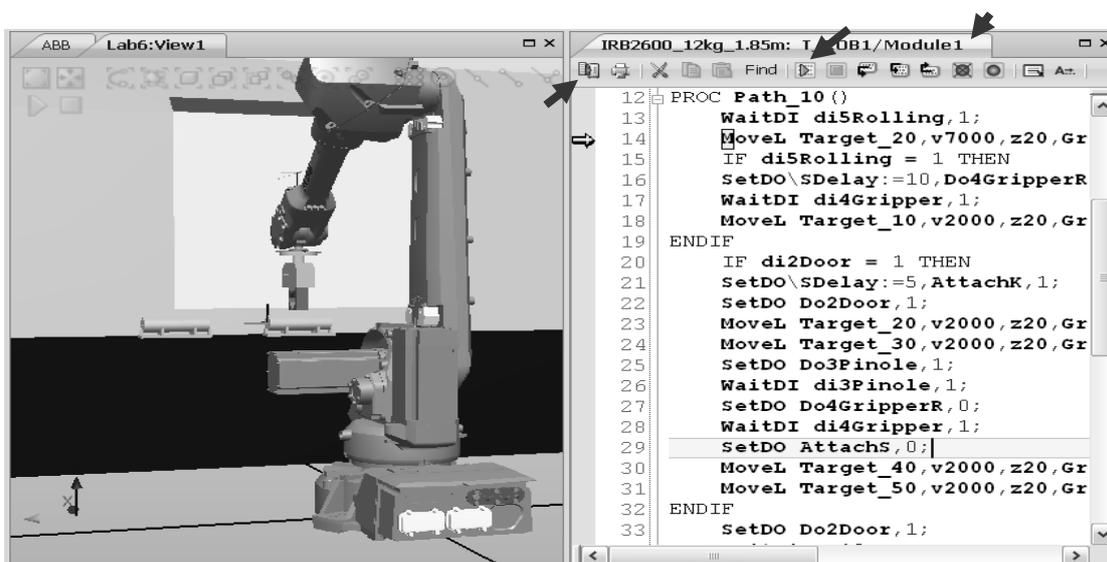


Рисунок 6.9 – Окно редактирования программы

Располагаем в окне графического редактора изображение РТК таким образом, чтобы удобно было наблюдать за всеми действиями робота и технологического оборудования. В окне редактора программ нажимаем кнопку Start для начала выполнения программы и отслеживаем последовательность перемещения указателя по строкам программы и синхронное выполнение роботом действий по загрузке станка.

При ошибках в программе или для расширения ее функциональных возможностей допускается внесение исправлений с последующим их сохранением командой **Apply Changes**.

При сохранении отредактированной программы выполняется синтаксический и семантический анализ текста программы и возможны сообщения о наличии в программе ошибок. Сообщения об ошибках выводятся в окне **Output** внизу главного окна RobotStudio.

Контрольные вопросы

- 1 Для чего предназначена программа RAPID?
- 2 Из каких частей состоит программа RAPID?
- 3 Какие функции содержит программирование высокого уровня RAPID?
- 4 Назовите специфические особенности языка RAPID.

Список литературы

1 **Ефремов, В. Д.** Металлорежущие станки: учебник / В. Д. Ефремов, В. А. Горохов, А. Г. Схиртладзе; под общ. ред. П. И. Ящерицына. – 2-е изд., стер. – Старый Оскол: ТНТ, 2018. – 696 с.

2 **Лукинов, А. П.** Проектирование мехатронных и робототехнических устройств: учебное пособие / А. П. Лукинов. – Санкт-Петербург; Москва; Краснодар: Лань, 2016. – 608 с. : ил. + CD-ROM.

3 **Акулович, Л. М.** Основы автоматизированного проектирования технологических процессов в машиностроении: учебное пособие для вузов / Л. М. Акулович, В. К. Шелег. – Минск: Новое знание, 2016. – 488 с.

4 **Дьяконов, А. А.** CAD/CAM/CAE/CAPP-системы в машиностроении: учебное пособие / А. А. Дьяконов, А. Х. Нуркенов. – Челябинск: ЮУрГУ, 2017. – 168 с.: ил.

5 **Берлинер, Э. М.** САПР технолога-машиностроителя: учебник / Э. М. Берлинер, О. В. Таратынов. – Москва: Форум; ИНФРА-М, 2015. – 336 с.

6 Основы автоматизированного проектирования: учебник / Под ред. А. П. Карпенко. – Москва: ИНФРА-М, 2017. – 329 с.