А. А. Ловыгин,
Л. В. Теверовский

# Современный станок с ЧПУ и CAD/CAM-система

ДМК Пресс
издательство
Москва, 2012

УДК 004.451:004.43***
ББК 32.973.26-018.2***
Л68

Ловыгин А. А., Теверовский Л. В.
Л68 Современный станок с ЧПУ и CAD/CAM-система. – М.: ДМК Пресс, 2012. – 279 с.: ил.

ISBN 978-5-94074-560-0

Вы держите в руках уже третье, на этот раз полноцветное издание книги, которое адресовано всем, кто хочет быстро разобраться с основами автоматизированного программирования обработки на станках с ЧПУ: студентам и выпускникам технических колледжей и вузов, инженерам-технологам, инженерам-конструкторам, операторам и наладчикам станков с ЧПУ, руководителям участков механической обработки. Мы постарались не только максимально доходчиво описать конструкцию и принципы работы современного фрезерного станка с ЧПУ, процесс создания управляющей программы, основы работы в CAD/CAM-системе, но и сделать так, чтобы книгу было приятно держать в руках и пользоваться ею каждый рабочий день.

К книге прилагается DVD, содержащий авторские видеокурсы по CAM-системе и постпроцессоры на различные станки с ЧПУ, которые помогут вам оперативно перейти от теоретических знаний к практической работе с современным металлорежущим оборудованием.

УДК 004.451:004.43***
ББК 32.973.26-018.2***

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

ISBN 978-5-94074-560-0

© Ловыгин А. А., Теверовский Л. В., 2012
© Оформление, ДМК Пресс, 2012

# Содержание

**Глава 1**
**ОСНОВЫ ЧИСЛОВОГО ПРОГРАММНОГО УПРАВЛЕНИЯ** ..... 10

1.1. Автоматическое управление ..... 10
1.2. Особенности устройства и конструкции фрезерного станка с ЧПУ ..... 12
1.3. Функциональные составляющие (подсистемы) ЧПУ ..... 14
1.3.1. Подсистема управления ..... 14
1.3.2. Подсистема приводов ..... 16
Высокоточные ходовые винты ..... 16
Двигатели ..... 16
1.3.3. Подсистема обратной связи ..... 18
Датчики, используемые для определения положения ..... 18
Датчики состояния исполнительных органов ..... 20
1.3.4. Функционирование системы ЧПУ ..... 21
1.4. Языки для программирования обработки ..... 23
Краткое изложение главы ..... 24
Вопросы ..... 24

**Глава 2**
**ОСНОВЫ МЕТАЛЛООБРАБОТКИ** ..... 25

2.1. Процесс фрезерования ..... 25
2.2. Режущий инструмент ..... 27
2.3. Вспомогательный инструмент ..... 33
2.4. Основные определения и формулы ..... 34
2.5. Рекомендации по фрезерованию ..... 35
Краткое изложение главы ..... 38
Вопросы ..... 38

**Глава 3**
**ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ ОБРАБОТКИ** ..... 39

3.1. Прямоугольная система координат ..... 39
3.2. Написание простой управляющей программы ..... 40
3.3. Создание УП на персональном компьютере ..... 43
3.4. Передача управляющей программы на станок ..... 47

3.5. Проверка управляющей программы на станке ..... 49
Общие сведения ..... 49
Тестовые режимы станка с ЧПУ ..... 50
Последовательность полной проверки УП ..... 51
3.6. Советы по технике безопасности при эксплуатации станков с ЧПУ ..... 52
Краткое изложение главы ..... 54
Вопросы ..... 54

**Глава 4**
**СТАНОЧНАЯ СИСТЕМА КООРДИНАТ** ..... 55
4.1. Нулевая точка станка и направления перемещений ..... 55
4.2. Нулевая точка программы и рабочая система координат ..... 59
4.3. Компенсация длины инструмента ..... 62
4.4. Абсолютные и относительные координаты ..... 64
4.5. Комментарии в УП и карта наладки ..... 65
Краткое изложение главы ..... 68
Вопросы ..... 68

**Глава 5**
**СТРУКТУРА УПРАВЛЯЮЩЕЙ ПРОГРАММЫ** ..... 69
5.1. G- и M-коды ..... 69
5.2. Структура программы ..... 70
5.3. Слово данных, адрес и число ..... 73
5.4. Модальные и немодальные коды ..... 74
5.5. Формат программы ..... 75
5.6. Строка безопасности ..... 78
5.7. Важность форматирования УП ..... 79
Краткое изложение главы ..... 80
Вопросы ..... 81

**Глава 6**
**БАЗОВЫЕ G-КОДЫ** ..... 82
Введение ..... 82
6.1. Ускоренное перемещение – G00 ..... 83
6.2. Линейная интерполяция – G01 ..... 85
6.3. Круговая интерполяция – G02 и G03 ..... 86
Дуга с I, J, K ..... 87
Дуга с R ..... 88
Использование G02 и G03 ..... 89
Краткое изложение главы ..... 91
Вопросы ..... 91

**Глава 7**
**БАЗОВЫЕ М-КОДЫ** ..... 92

Введение ..... 92
7.1. Останов выполнения управляющей программы – M00 и M01 ..... 93
7.2. Управление вращением шпинделя – M03, M04, M05 ..... 94
7.3. Управление подачей СОЖ – M07, M08, M09 ..... 96
7.4. Автоматическая смена инструмента – M06 ..... 97
7.5. Завершение программы – M30 и M02 ..... 100
Краткое изложение главы ..... 100
Вопросы ..... 101

**Глава 8**
**ПОСТОЯННЫЕ ЦИКЛЫ СТАНКА С ЧПУ** ..... 102

Введение ..... 102
8.1. Стандартный цикл сверления и цикл сверления с выдержкой ..... 105
8.2. Относительные координаты в постоянном цикле ..... 107
8.3. Циклы прерывистого сверления ..... 108
8.4. Циклы нарезания резьбы ..... 110
8.5. Циклы растачивания ..... 111
8.6. Примеры программ на сверление отверстий при помощи постоянных циклов ..... 112
Пример № 1 ..... 112
Пример № 2 ..... 113
Краткое изложение главы ..... 115
Вопросы ..... 116

**Глава 9**
**АВТОМАТИЧЕСКАЯ КОРРЕКЦИЯ РАДИУСА ИНСТРУМЕНТА** ..... 117

9.1. Основные принципы ..... 117
9.2. Использование автоматической коррекции на радиус инструмента ..... 122
9.3. Активация, подвод и отвод ..... 123
Краткое изложение главы ..... 125
Вопросы ..... 125

**Глава 10**
**ОСНОВЫ ЭФФЕКТИВНОГО ПРОГРАММИРОВАНИЯ** ..... 127

10.1. Подпрограмма ..... 127
10.2. Работа с осью вращения (4-ой координатой) ..... 131
10.3. Параметрическое программирование ..... 134

Краткое изложение главы ..... 145
Вопросы ..... 146

**Глава 11**
**ПРИМЕРЫ УПРАВЛЯЮЩИХ ПРОГРАММ** ..... 147

11.1. Программирование в ISO ..... 147
Пример № 1. Контурная обработка ..... 147
Пример № 2. Контурная обработка с коррекцией на радиус инструмента ..... 148
Пример № 3. Контурная обработка ..... 150
Пример № 4. Контурная обработка с коррекцией на радиус инструмента ..... 151
Пример № 5. Фрезерование прямоугольного кармана ..... 152
Пример № 6. Фрезерование круглого кармана ..... 154
11.2. Программирование для Heidenhain ..... 155
Пример № 1. Контурная обработка ..... 156
Пример № 2. Контурная обработка с коррекцией на радиус инструмента ..... 157
Пример № 3. Сверление 7 отверстий диаметром 3 мм и глубиной 6,5 мм с помощью постоянного цикла Heidenhain ..... 158

**Глава 12**
**CAD/CAM** ..... 159

12.1. Методы программирования ..... 159
12.2. Что такое CAD и CAM? ..... 160
12.3. Общая схема работы с CAD/CAM-системой ..... 160
12.4. Виды моделирования ..... 162
12.5. Уровни CAM-системы ..... 165
12.6. Геометрия и траектория ..... 166
12.7. Алгоритм работы в CAM-системе ..... 167
12.7.1. Выбор геометрии ..... 167
12.7.2. Выбор стратегии и инструмента, назначение параметров обработки ..... 169
Плоская обработка ..... 170
Объемная обработка ..... 171
12.7.3. Бэкплот и верификация ..... 175
12.7.4. Постпроцессирование ..... 177
12.7.5. Передача УП на станок с ЧПУ ..... 180
12.8. Ассоциативность ..... 181
12.9. Пятикоординатное фрезерование и 3D-коррекция ..... 181

12.10. Высокоскоростная обработка (ВСО) ..... 183
12.11. Требования к современной САМ-системе ..... 185

**Глава 13**
**СИСТЕМА ТРЕХМЕРНОГО ТВЕРДОТЕЛЬНОГО МОДЕЛИРОВАНИЯ КОМПАС-3D** ..... 188

Классические твердотельные операции ..... 188
13.1. Твердотельное моделирование ..... 190
13.2. Поверхностное моделирование ..... 200
13.3. Моделирование деталей из листового материала ..... 207
13.4. Экспорт геометрии ..... 212

**Глава 14**
**ОСНОВЫ РАБОТЫ В САМ-СИСТЕМЕ ESPRIT** ..... 214

14.1. Общие сведения ..... 214
14.2. Системные требования ..... 215
14.3. Активация лицензии и запуск программы ..... 215
14.4. Интерфейс программы ..... 216
14.5. Порядок работы в программе ..... 218
14.6. Создание операций фрезерной обработки ..... 218

**Глава 15**
**УПРАВЛЕНИЕ СТАНКОМ С ЧПУ** ..... 232

15.1. Органы управления ..... 232
15.2. Основные режимы работы ..... 234
15.3. Индикация системы координат ..... 235
15.4. Установление рабочей системы координат ..... 235
15.4.1. Алгоритм нахождения нулевой точки детали по оси Z ..... 236
15.4.2. Алгоритм нахождения нулевой точки детали по осям X и Y ..... 237
15.4.3. Алгоритм нахождения нулевой точки в центре отверстия ..... 238
15.5. Измерение инструмента и детали ..... 239

**Глава 16**
**СПРАВОЧНИК КОДОВ И СПЕЦИАЛЬНЫХ СИМВОЛОВ ПРОГРАММИРОВАНИЯ** ..... 242

16.1. G-коды ..... 242
16.2. Адреса/слова данных ..... 259
16.3. M-коды ..... 262
16.4. Специальные символы в УП ..... 263

**Глава 17**
**ПОЛЕЗНЫЕ ПРОГРАММЫ** ..... 265
17.1. Мониторинг ЧПУ ..... 265
Возможности ..... 266
Контроль в режиме реального времени ..... 266
Формирование отчетов и графиков ..... 268
Ускорение работы цеховых служб ..... 269
Внедрение на предприятии ..... 270
17.2. Редактор УП Cimco Edit 6 ..... 270
17.3. Техтран® ..... 273
Фрезерная обработка ..... 273
Токарная обработка ..... 274
Токарно-фрезерная обработка ..... 275
Многошпиндельное сверление ..... 275
Раскрой листового материала ..... 276
Листовая штамповка ..... 277
Электроэрозионная обработка ..... 277
Контроль управляющих программ ..... 278

Авторы благодарят за помощь в издании книги:
Технический Центр ВариУс и лично Валерия Жовтобрюх,
компанию НИП-Информатика и лично Александра Лиферова,
компанию АСКОН и лично Дмитрия Оснача.

# Глава 1
# ОСНОВЫ ЧИСЛОВОГО ПРОГРАММНОГО УПРАВЛЕНИЯ

## 1.1. Автоматическое управление

На сегодняшний день практически каждое предприятие, занимающееся механической обработкой, имеет в своем распоряжении станки с числовым программным управлением (ЧПУ). Станки с ЧПУ выполняют все те же функции, что и обычные станки с ручным управлением, однако перемещения исполнительных органов этих станков управляются электроникой. В чем же основное преимущество станков с ЧПУ и почему все большее число заводов предпочитает вкладывать деньги именно в современное оборудование с автоматическим управлением, а не покупать относительно дешевые универсальные станки?

*Рис. 1.1. Универсальный сверлильно-фрезерный станок*

Первым, очевидным плюсом от использования станков с ЧПУ является более **высокий уровень автоматизации производства**. Случаи вмешательства оператора станка в процесс изготовления детали сведены к минимуму. Станки с ЧПУ могут работать практически автономно, день за днем, неделю за неделей, выпуская продукцию с неизменно высоким качеством. При этом главной заботой станочника-оператора являются в основном подготовительно-заключительные операции: установка и снятие детали, наладка инструмента и т. д. В результате один работник может обслуживать одновременно несколько станков.

Вторым преимуществом является производственная **гибкость**. Это значит, что для обработки разных деталей нужно всего лишь заменить программу. А уже проверенная и отработанная программа может быть использована в любой момент и любое число раз.

Третьим плюсом являются **высокая точность и повторяемость обработки**. По одной и той же программе вы сможете изготовить с требуемым

*Рис. 1.2. Фрезерный станок с ЧПУ фирмы Doosan*

качеством тысячи практически идентичных деталей. Ну и, наконец, числовое программное управление **позволяет обрабатывать такие детали, которые невозможно изготовить на обычном оборудовании**. Это детали со сложной пространственной формой, например штампы и пресс-формы.

Стоит отметить, что сама методика работы по программе позволяет более точно **предсказывать время обработки** некоторой партии деталей и соответственно более полно **загружать** оборудование.

Станки с ЧПУ стоят достаточно дорого и требуют больших затрат на установку и обслуживание, чем обычные станки. Тем не менее их высокая производительность легко может перекрыть все затраты при грамотном использовании и соответствующих объемах производства.

Давайте разберемся, что же такое ЧПУ. **Числовое программное управление – это автоматическое управление станком при помощи компьютера (который находится внутри станка) и программы обработки (управляющей программы).** До изобретения ЧПУ управление станком осуществлялось вручную или механически.

Осевыми перемещениями станка с ЧПУ руководит компьютер, который читает управляющую программу (УП) и выдает команды соответствующим двигателям. Двигатели заставляют перемещаться исполнительные органы станка – рабочий стол или колонну со шпинделем. В результате производится механическая обработка детали. Датчики, установленные на направляющих, посылают информацию о фактической позиции исполнительного органа обратно в компьютер. Это называется обратной связью. Как только компьютер узнает о том, что исполнительный орган станка находится в требуемой позиции, он выполняет следующее перемещение. Такой процесс продолжается, пока чтение управляющей программы не подойдет к концу.

По своей конструкции и внешнему виду станки с ЧПУ похожи на обычные универсальные станки. Единственное внешнее отличие этих двух типов станков заключается в наличии у станка с ЧПУ устройства числового программного управления (УЧПУ), которое часто называют **стойкой ЧПУ**.

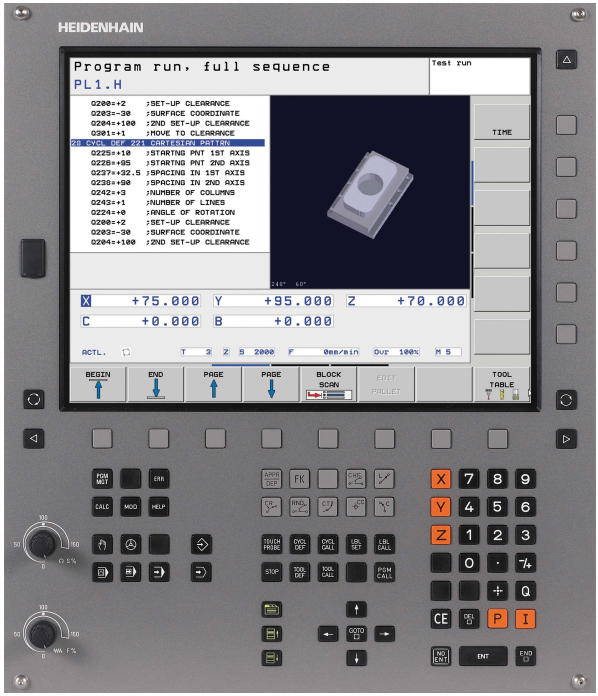



*Рис. 1.3. Стойка ЧПУ Heidenhain TNC*

# 1.2. Особенности устройства и конструкции фрезерного станка с ЧПУ

Фрезерные станки с ЧПУ можно классифицировать по различным признакам: по положению шпинделя (вертикальные или горизонтальные), по количеству управляемых осей или степеней свободы (2, 3, 4 или 5 осей), по точности позиционирования и повторяемости обработки, по количеству используемого инструмента (одно- или многоинструментальные) и т. д.

Рассмотрим конструкцию вертикально-фрезерного станка с ЧПУ (рис. 1.4, 1.5). *Станина* (1) предназначена для крепления всех узлов и механизмов станка. *Рабочий стол* (2) может перемещаться в продольном (влево/вправо) и поперечном (вперед/назад) направлениях по *направляющим* (3). *Пульт управления, или стой-*

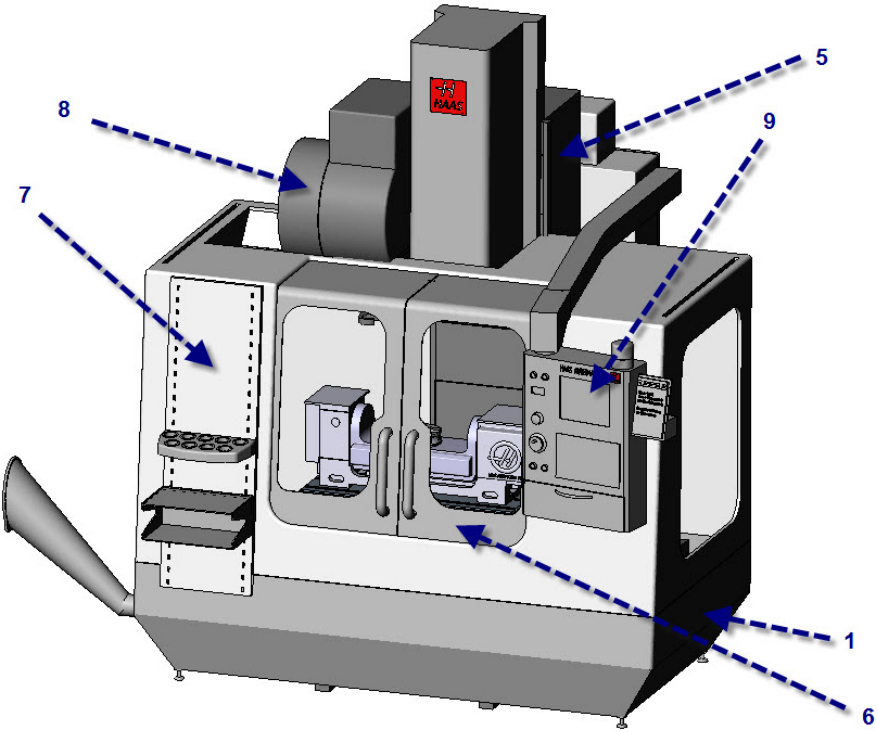



*Рис. 1.4. Корпус вертикально-фрезерного станка с ЧПУ*

*ка ЧПУ* (9), закреплен на кронштейне и может быть перемещен в удобное для оператора положение.

На рабочем столе закрепляют заготовки и различные технологические приспособления. Для этого на столе имеются специальные Т-образные пазы. *Шпиндель* (4) предназначен для зажима режущего инструмента и придания ему вращения. Шпиндель закреплен на *колонне* (5), которая может перемещаться в вертикальном направлении (вверх/вниз). От точности вращения шпинделя, его жесткости и виброустойчивости в значительной мере зависят точность и качество обработки. Таким образом, рассматриваемый станок является трехосевым.

*Защитные кожухи* (7) необходимы для обеспечения безопасности. Они защищают оператора станка от летящей стружки и смазывающе-охлаждающей жидкости (СОЖ), которая подается в зону обработки под давлением. Дверца (6) обеспечивает доступ в рабочую зону станка. В магазине инструментов (8) барабанного типа находится набор режущих инструментов. При этом взятие необходимого инструмента и фиксация его в шпинделе обеспечиваются устройством автоматической смены инструмента и производятся по определенной команде управляющей программы.

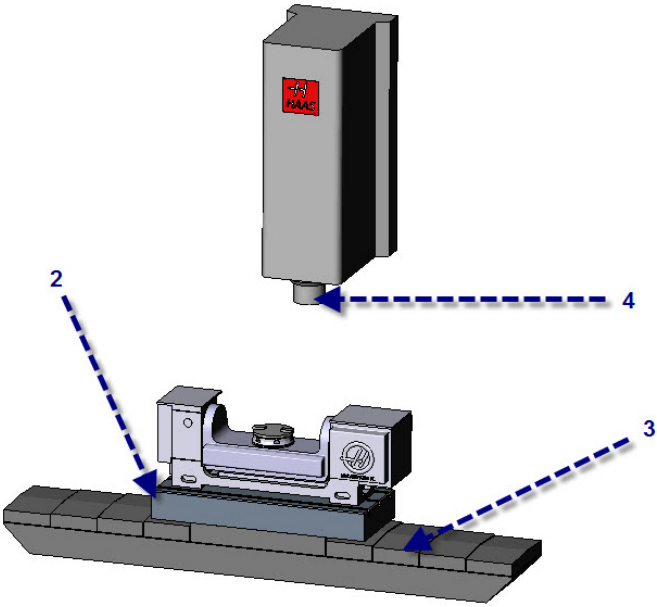



*Рис. 1.5. Конструктивные элементы станка*

# 1.3. Функциональные составляющие (подсистемы) ЧПУ

Для того чтобы сделать из обычного станка с ручным управлением станок с ЧПУ, необходимо внедрить определенные компоненты в его конструкцию. Недостаточно просто подключить станок к компьютеру, чтобы он работал по программе, – необходимо модернизировать механическую и электронную «начинку» станка. Давайте посмотрим, как устроена система ЧПУ (СЧПУ) на большинстве современных станков.

Условно СЧПУ можно разделить на три подсистемы:

- подсистему управления;
- подсистему приводов;
- подсистему обратной связи.

Далее в этом разделе мы подробнее остановимся на каждой из данных подсистем.

## *1.3.1. Подсистема управления*

Центральной частью всей СЧПУ является подсистема управления. С одной стороны, она читает управляющую программу и отдает команды различным агрегатам станка на выполнение тех или иных операций. С другой – взаимодействует с человеком, позволяя оператору станка контролировать процесс обработки.

**Сердцем подсистемы управления является контроллер (процессор), который обычно расположен в корпусе стойки ЧПУ**. Сама стойка имеет набор кнопок и экран (все вместе называется пользовательским интерфейсом) для ввода и вывода необходимой информации.

**Системы управления** могут быть **как закрытыми, так и открытыми**, ПК-совместимыми. Закрытые системы управления имеют собственные алгоритмы и циклы работы, собственную логику. Производители таких систем, как правило, не распространяют информацию об их архитектуре. Скорее всего, вы не сможете самостоятельно обновить программное обеспечение и редактировать настройки такой системы. У систем закрытого типа есть важное преимущество – они, как правило, имеют высокую надежность, так как все компоненты системы прошли тестирование на совместимость.

В последнее время стало появляться все больше открытых, ПК-совместимых систем управления. Их аппаратная начинка практически такая же, как и у вашего домашнего персонального компьютера. Преимущество такого метода – в доступности и дешевизне электронных компонентов, большинство из которых можно приобрести в обычном компьютерном магазине, и в возможности обновления внутреннего программного обеспечения.

Самые современные СЧПУ могут быть оснащены CAM-системой, позволяющей автоматизировать процесс написания УП прямо на станке. Наиболее яркий пример – системы ЧПУ серии MAPPS IV японских станков Mori Seiki содержат встроенное программное обеспечение ESPRIT от компании DP Technology (США) и позволяют оператору не только создать УП любой сложности, но и произвести ее всестороннюю проверку.

*Рис. 1.6. Стойка MAPPS IV с CAM-системой ESPRIT*

## 1.3.2. Подсистема приводов

Подсистема приводов включает в себя различные **двигатели** и **винтовые передачи** для окончательного выполнения команд подсистемы управления – для реализации перемещения исполнительных органов станка.

### *Высокоточные ходовые винты*

Важными компонентами подсистемы приводов являются высокоточные ходовые винты. Вы, наверное, знаете, что на станке с ручным управлением рабочий, вращая рукоятку, соединенную с ходовым винтом, перемещает рабочий стол. На днище стола укреплена гайка таким образом, что при повороте винта происходит линейное перемещение стола.

Усовершенствованный ходовой винт станка с ЧПУ позволяет выполнять перемещение исполнительного органа с минимальным трением и практически без люфтов. Устранение люфта очень важно по двум причинам. Во-первых, это необходимо для обеспечения сверхточного позиционирования. Во-вторых, только при соблюдении этого условия возможно нормальное попутное фрезерование.

### *Двигатели*

Второй составляющей подсистемы является двигатель (а точнее – несколько двигателей). Вращение вала двигателя приводит к повороту высокоточного ходового винта и линейному перемещению рабочего стола или колонны. В конструкции станков используются шаговые электродвигатели и серводвигатели.

**Шаговый электродвигатель – это электромеханическое устройство, преобразующее электрический сигнал управления в дискретное механическое перемещение.** Существует несколько основных видов шаговых двигателей, отличающихся конструктивным исполнением:

- шаговые двигатели с переменным магнитным сопротивлением;
- шаговые двигатели с постоянным магнитным сопротивлением;
- гибридные двигатели.

Принцип работы у всех этих двигателей примерно одинаков и достаточно прост.

**Шаговый двигатель с переменным магнитным сопротивлением** имеет несколько полюсов на статоре и ротор из магнитно-мягкого материала (реактивный ротор). На рис. 1.7 показан двигатель, имеющий шесть полюсов на статоре, ротор с четырьмя зубьями и три независимые обмотки, каждая из которых приходится на противоположные полюса статора.

При подаче электрического тока в одну из обмоток ротор стремится занять положение, при котором возникший магнитный поток будет замкнут. То есть зубья ротора будут находиться прямо напротив тех полюсов статора, на обмотки которого подан ток. Если выключить ток в этой обмотке и подать его в следующую обмотку, то ротор повернется, чтобы в очередной раз замкнуть магнитный поток своими зубьями. Для непрерывного вращения ротора необходимо попеременно подавать электрический ток в 1, 2 и 3 обмотки, при этом шаг вращения для представленного двигателя составит 30°.

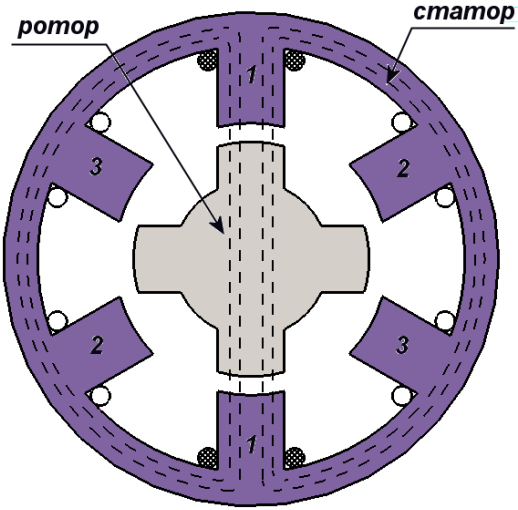



*Рис. 1.7. Устройство шагового двигателя с переменным магнитным сопротивлением*

**Шаговый двигатель с постоянными магнитами** состоит из статора с обмотками и ротора с постоянными магнитами. На рис. 1.8 показан двигатель, имеющий две пары полюсов статора и три пары полюсов ротора. При подаче электрического тока в одну из обмоток ротор займет положение, при котором разноименные полюса статора и ротора будут находиться напротив друг друга. Для непрерывного вращения ротора необходимо попеременно подавать электрический ток в 1 и 2 обмотки, при этом шаг вращения составит 30°.

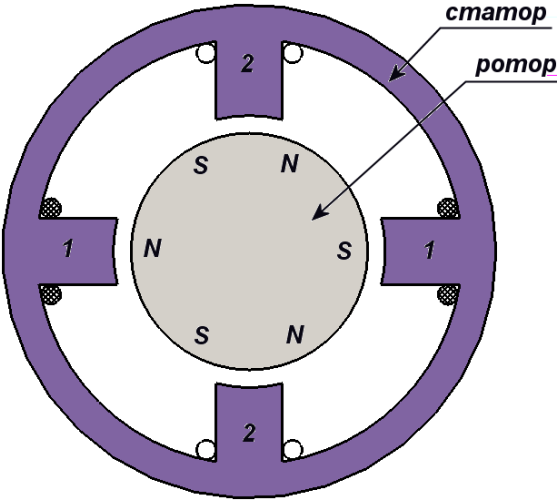



*Рис. 1.8. Устройство шагового двигателя с постоянными магнитами*

**Большинство современных шаговых электродвигателей являются гибридными**, то есть сочетают достоинства двигателей с переменным магнитным полем и двигателей с постоянными магнитами, имеют гораздо большее число полюсов статора и зубьев ротора, что обеспечивает меньший шаг вращения.

Когда подсистема управления посылает шаговому двигателю электрический импульс, то происходит поворот на определенный угол, который зависит от конструкции двигателя (например, 0,7°). Если ходовой винт имеет шаг 1 мм, то один импульс заставит исполнительный орган станка линейно переместиться на $0{,}7/360 \times 1 = 0{,}0019$ мм. Эта величина называется **разрешением системы**, или **ценой импульса**. Нельзя переместить исполнительный орган на величину, меньшую, чем разрешение системы. Таким образом, вы видите, что существует прямая взаимосвязь между двигателем, ходовым винтом и точностью перемещений станка.

Простота конструкции и легкость управления сделали шаговые электродвигатели очень популярными. Основным минусом двигателей этого типа является их толчковая или дискретная работа, которая может привести к ухудшению качества чистовой обработки поверхностей и эффекту «ступенек» при выполнении обработки по наклонной прямой или дуге. Однако шаговые двигатели могут работать без использования дорогостоящей и сложной обратной связи. Это позволяет создавать недорогие, хотя и не высокоточные станки.

Самые современные станки с ЧПУ не оснащаются шаговыми двигателями. На смену им пришли серводвигатели, которые имеют более сложную конструкцию. Серводвигатели, в отличие от шаговых двигателей, работают гладко, имеют лучшие характеристики, но ими тяжелее управлять.

Для работы с серводвигателем необходимо наличие специальных контроллеров и устройств обратной связи, что, несомненно, приводит к увеличению стоимости станка.

## 1.3.3. Подсистема обратной связи

Подсистема обратной связи главным образом призвана обеспечивать подсистему управления информацией о реальной позиции исполнительного органа станка и о скорости двигателей. Подсистема обратной связи может быть открытого или замкнутого типа.

**Системы открытого типа** регистрируют наличие или отсутствие сигнала из подсистемы управления. К сожалению, они не могут дать информации о реальной позиции исполнительного органа и скорости двигателей, поэтому в современных станках с ЧПУ практически не используются.

**Системы замкнутого типа** используют внешние датчики для проверки необходимых параметров.

### Датчики, используемые для определения положения

Как правило, в станках с ЧПУ для определения положения и состояния исполнительных органов используются два типа датчиков: **линейные** датчики положения и **вращающиеся** датчики положения.

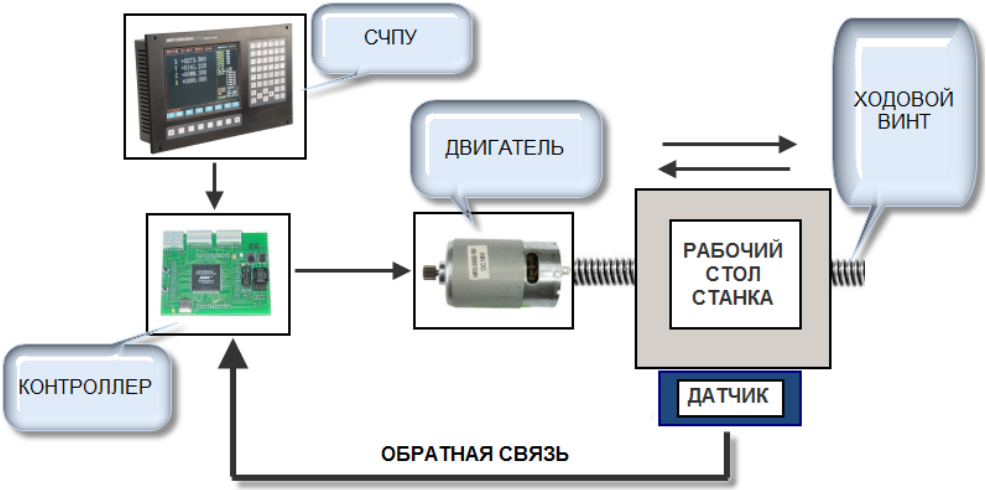



*Рис. 1.9. Схема обратной связи на станке с ЧПУ*

**Вращающийся датчик положения** (рис. 1.10) крепится на валу двигателя и позволяет определять его угловое положение. Этот датчик состоит из источника света, оптического датчика (приемника) и диска с маленькими радиальными прорезями (растрами). Растровый диск укреплен на валу, источник света и оптический датчик находятся с разных сторон от диска.

Когда диск вращается, то лучи проходят сквозь его прорези и падают на оптический датчик. Оптический датчик работает как переключатель, который включается или выключается при попадании на него лучей света. Это дает возможность определить относительное или абсолютное положение и направление вращения двигателя. Полученная информация отправляется в подсистему управления.

Все вращающиеся датчики имеют один существенный недостаток. Так как они устанавливаются непосредственно на валу двигателя, то не могут напрямую измерить линейное положение исполнительного органа станка. Они дают рассчитанное положение, основанное на данных о шаге ходового винта, и в высокоточных станках для определения линейного положения не применяются. Их можно использовать в конструкции шпинделя для определения числа оборотов при вращении и для нахождения его углового положения.

**Линейные датчики положения** используются практически во всех современных станках с ЧПУ для точного определения абсолютной или относительной позиции исполнительных органов. Датчики содержат два взаимосвязанных узла – растровую шкалу и считывающую головку (рис. 1.11). *Растровая шкала* (1), расположенная вдоль направляющих, представляет собой линейку с маленькими прямоугольными прорезями (растрами). Считывающая головка, перемещающаяся вместе с исполнительным органом станка, состоит из *осветителей* (2), *фотоприемников* (3) и *индикаторной пластины* (4). Причем осветители и индикаторная пластина находятся с одной стороны от растровой шкалы, а фотоприемники – с другой.

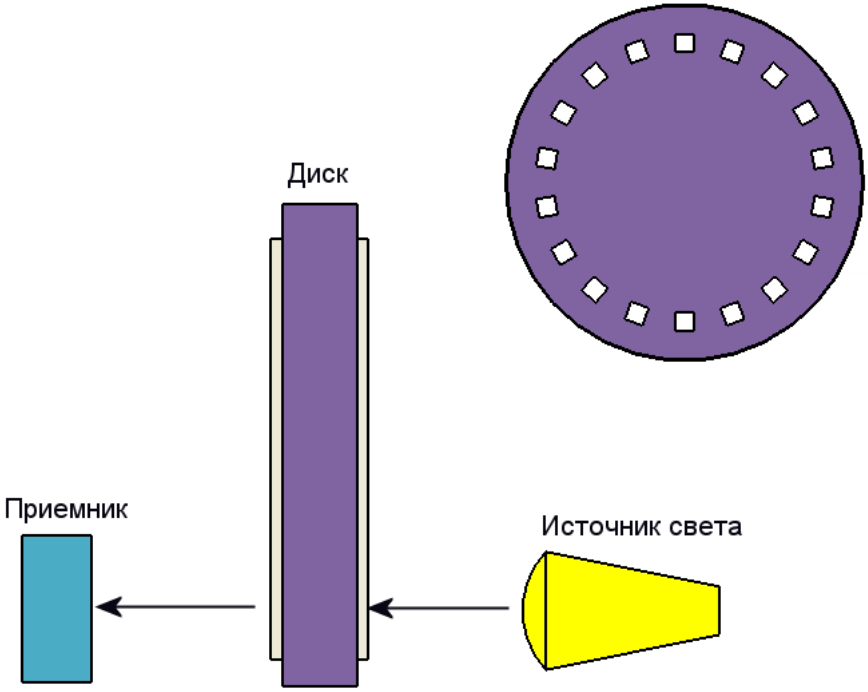



*Рис. 1.10. Вращающийся датчик положения*

На индикаторной пластине также присутствуют два растровых участка со смещенным шагом для формирования двух сигналов. Когда считывающая головка перемещается вдоль растровой шкалы, то световые сигналы от осветителей проходят через индикаторную пластину, затем через шкалу и регистрируются фотоприемниками. Полученные сигналы дают возможность определить величину и направление перемещения. На растровой шкале может находиться дополнительная дорожка референтных меток для задания собственного начала отсчета.

Системе ЧПУ также необходима информация о скорости, ускорении и замедлении исполнительного органа станка. Расчет величины ускорения и замедления необходим для точного позиционирования. Дело в том, что когда рабочий стол перемещается в требуемую позицию, он заранее замедляет скорость перемещения, чтобы «не промахнуться» мимо требуемой координаты.

## Датчики состояния исполнительных органов

Кроме вышеперечисленных датчиков, конечно же, используются и другие. **Температурные датчики** (термопары) применяют для определения температуры исполнительных органов, расчета температурного линейного расширения компонентов станка и для контроля над температурой масла и воздуха. **Инфракрасные датчики** используются в станочных системах автоматического измерения.

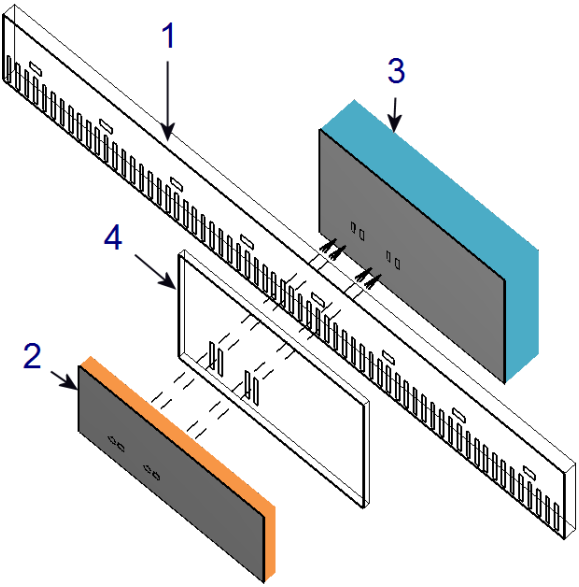



*Рис. 1.11. Линейный датчик положения*

## 1.3.4. Функционирование системы ЧПУ

Мы рассмотрели по отдельности работу трех подсистем, составляющих основу СЧПУ. Теперь поговорим о функционировании всей системы в целом.

Программист создает управляющую программу, в которой содержится закодированная информация о траектории и скорости перемещения исполнительных органов станка, частоте вращения шпинделя и другие данные, необходимые для выполнения обработки. Подсистема управления читает эту программу, расшифровывает ее и вырабатывает **профиль перемещения**.

Профиль перемещения можно представить в виде графика, который показывает, в какой точке должен находиться исполнительный орган станка через определенные промежутки времени. В соответствии с профилем перемещения подсистема управления посылает на соответствующий двигатель строго определенное количество электрических импульсов. Двигатель вращает ходовой винт, и исполнительный орган станка перемещается в указанную позицию (координату). Датчики обратной связи отправляют в подсистему управления информацию о действительной достигнутой позиции исполнительного органа. **Происходит сравнение фактической и требуемой (теоретической) позиций**. Если между ними есть разница (ошибка перемещения), то подсистема управления посылает скорректированное на величину ошибки число электрических импульсов на двигатель. Этот процесс повторяется снова и снова, пока исполнительный орган станка не достигнет требуемой позиции с определенной (очень высокой) точностью. Вообще, некоторая ошибка перемещения присутствовать будет всегда. Главное, чтобы она была настолько маленькой, что ей можно было бы пренебречь.

Мы рассмотрели простейший вариант – перемещение вдоль одной оси. Давайте усложним задачу. Пусть требуется переместить рабочий стол прямолинейно, но не параллельно ни одной из осей станка (рис. 1.12). Для того чтобы выполнить такое перемещение, система ЧПУ вынуждена строить между точками 1 и 2 множество опорных точек и двигать рабочий стол по этим точкам «ступенчато», попеременно включая подачу то по одной оси, то по другой. При этом нужно поддерживать такое соотношение скоростей движения по осям, чтобы траектория перемещения соответствовала заданной траектории. Работу по расчету этих промежуточных опорных точек выполняет специальное устройство, входящее в состав подсистемы управления, – интерполятор. **Интерполятор непрерывно в соответствии с заданными перемещениями поддерживает функциональную связь между опорными точками и оценивает отклонения от заданной траектории, стремясь свести их к минимуму.** Эти «ступеньки» имеют некоторое отклонение от заданной траектории перемещения. Величина отклонения полученной «ступенчатой» траектории будет равна или кратна цене импульса (разрешению) системы ЧПУ или импульса, формируемого датчиком обратной связи. Так как в современных станках разрешение системы ЧПУ приближается к 0,001 мм (1 мкм), то получаемое перемещение можно рассматривать как гладкое.

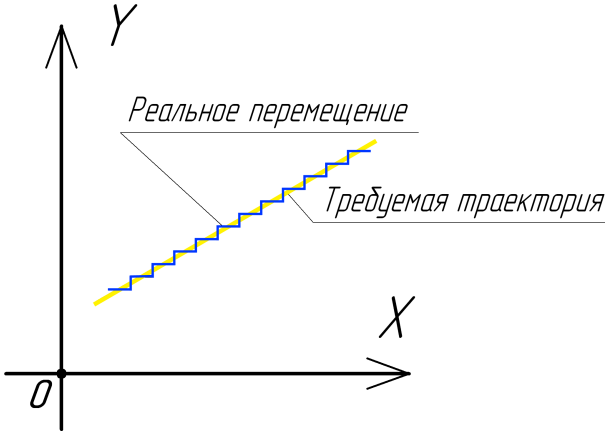



*Рис. 1.12. При перемещении из точки 1 в точку 2 система ЧПУ строит множество опорных точек и перемещает исполнительный орган «ступенчато», попеременно включая подачу то по одной оси, то по другой*

Рассмотренная интерполяция **называется линейной**. Если же необходимо выполнить перемещение по дуге, то интерполяция будет **круговой** (рис. 1.13). В случае перемещения по дуге выполняется так называемая линейная аппроксимация дуги, то есть замена дуги маленькими прямолинейными отрезками. Поэтому исполнительный орган станка также будет двигаться по «ступенчатой» траектории, которая визуально будет казаться абсолютно гладкой.

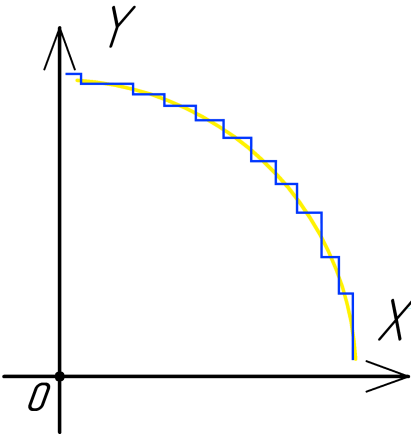



*Рис. 1.13. Схема круговой интерполяции*

# 1.4. Языки для программирования обработки

С момента появления первых станков с ЧПУ до внедрения новейших обрабатывающих центров появились различные языки для программирования обработки. Сегодня программирование в G- и M-кодах является наиболее популярным. **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». Однако производители систем ЧПУ хоть и придерживаются этих стандартов для описания основных функций, но допускают вольности и отступления от правил, когда речь заходит о каких-либо специальных возможностях своих систем.

Системы ЧПУ Fanuc (Япония) были одними из первых, адаптированных под работу с G- и M-кодами ISO и использующими этот стандарт наиболее полно. В настоящее время стойки Fanuc являются очень популярными и наиболее распространенными как за рубежом, так и в России. Поэтому в этой книге основой для описания программирования в G- и M-кодах будет именно стиль СЧПУ Fanuc.

Стойки ЧПУ других известных производителей, например Heidenhain и Sinumerik (Siemens), также имеют возможности по работе с G- и M-кодами, однако некоторые коды все же могут отличаться. Но не стоит этого пугаться. Нет никакой необходимости знать все коды всех систем ЧПУ. **Достаточно знать набор основных G- и M-кодов**, а о возникшей разнице в программировании специфических функций можно узнать из документации к конкретной системе. Освоив стиль программирования Fanuc, скорее всего, вы сможете работать на любом другом оборудовании с ЧПУ.

Некоторые производители систем ЧПУ предлагают диалоговый язык программирования. Этот язык упрощает общение с системой, особенно для новых операторов, так как основой для него служат англоязычные предложения, сокращения, вопросы и графические элементы, которые вводятся оператором станка в интерактивном режиме.

# Краткое изложение главы

- Станки с ЧПУ стремительно внедряются в производственную сферу. Теоретически любой станок с ручным управлением может быть модернизирован и превращен в станок с ЧПУ.
- Система ЧПУ состоит из подсистемы управления, подсистемы приводов и подсистемы обратной связи.
- Подсистема управления представляет собой компьютер, который читает управляющую программу и выдает электрические импульсы на двигатели для выполнения перемещения исполнительных органов станка.
- Получая электрические импульсы подсистемы управления, электродвигатель вращает ходовой винт, тем самым обеспечивая перемещение исполнительного органа.
- Датчики обратной связи регистрируют фактическое положение исполнительного органа станка и сообщают эту информацию подсистеме управления.
- Для программирования обработки на современных станках с ЧПУ используется язык G- и M-кодов (ISO).

# Вопросы

1. Чем станок с ЧПУ отличается от станка с ручным управлением?
2. Каковы преимущества от использования станков с ЧПУ?
3. Опишите конструкцию и принцип работы шагового электродвигателя.
4. Почему в современных станках в основном используются серводвигатели, а не шаговые электродвигатели?
5. Перечислите основные составляющие СЧПУ.
6. Как функционирует подсистема обратной связи?
7. Назовите основные типы датчиков, которые используются в подсистеме обратной связи.
8. Какой язык для программирования обработки на станках с ЧПУ сегодня является наиболее популярным?

# Глава 2
# ОСНОВЫ МЕТАЛЛООБРАБОТКИ

Эта глава в первую очередь предназначена для тех, кто слабо знаком с процессом механической обработки, режущим и вспомогательным инструментом. Если же вы хорошо представляете процесс резания, то можете смело приступить к чтению следующей главы, которая посвящена основам программирования ЧПУ.

## 2.1. Процесс фрезерования

Существуют различные виды механической обработки: точение, фрезерование, сверление, строгание и т. д. Несмотря на конструкционные отличия станков и особенности технологий, управляющие программы для фрезерных, токарных, электроэрозионных, деревообрабатывающих и других станков с ЧПУ создаются по одному принципу. В этой книге основное внимание будет уделено программированию фрезерной обработки. Освоив эту разностороннюю технологию, вероятнее всего, вы самостоятельно разберетесь и с программированием других видов обработки. Вспомним некоторые элементы теории фрезерования, которые вам обязательно пригодятся при создании управляющих программ и работе на станке.

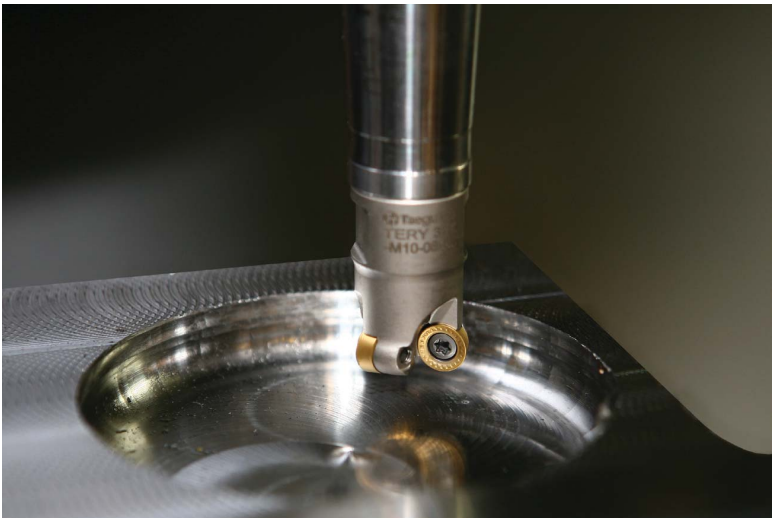


*Рис. 2.1. Процесс формирования кармана*

**Процесс фрезерования** заключается в срезании с заготовки лишнего слоя материала для получения детали требуемой формы, размеров и шероховатости обработанных поверхностей. При этом на станке осуществляется перемещение инструмента (фрезы) относительно заготовки или, как в нашем случае (для станка на рис. 1.4–1.5), перемещение заготовки относительно инструмента.

Для осуществления процесса резания необходимо иметь два движения – главное и движение подачи. При фрезеровании **главным движением** является вращение инструмента, а **движением подачи** – поступательное движение заготовки. В процессе резания происходит образование новых поверхностей путем деформирования и отделения поверхностных слоев с образованием стружки.

При обработке различают встречное и попутное фрезерование. **Попутное фрезерование**, или фрезерование по подаче, – способ, при котором направления движения заготовки и вектора скорости резания совпадают. При этом толщина стружки на входе зуба в резание максимальна и уменьшается до нулевого значения на выходе. При попутном фрезеровании условия входа пластины в резание более благоприятные. Удается избежать высоких температур в зоне резания и минимизировать склонность материала заготовки к упрочнению. Большая толщина стружки является в данном случае преимуществом. Силы резания прижимают заготовку к столу станка, а пластины – в гнезда корпуса, способствуя их надежному креплению. Попутное фрезерование является предпочтительным при условии, что жесткость оборудования, крепления и сам обрабатываемый материал позволяют применять данный метод.

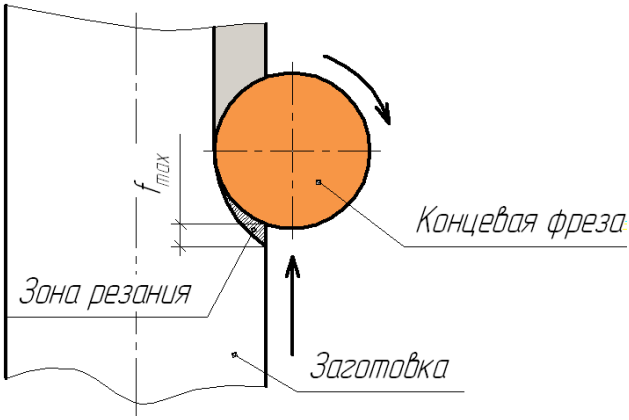



*Рис. 2.2. Попутное фрезерование*

**Встречное фрезерование**, которое иногда называют традиционным, наблюдается, когда скорости резания и движение подачи заготовки направлены в противоположные стороны. При врезании толщина стружки равна нулю, на выходе – максимальна. В случае встречного фрезерования, когда пластина начинает работу со стружкой нулевой толщины, возникают высокие силы трения, отжимающие фрезу и заготовку друг от друга. В начальный момент врезания зуба процесс ре-

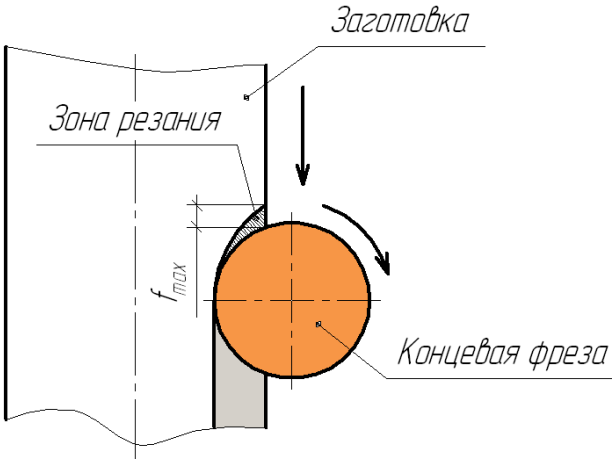



*Рис. 2.3. Встречное фрезерование*

зания больше напоминает выглаживание, с сопутствующими ему высокими температурами и повышенным трением. Зачастую это грозит нежелательным упрочнением поверхностного слоя детали. На выходе из-за большой толщины стружки в результате внезапной разгрузки зубья фрезы испытывают динамический удар, приводящий к выкрашиванию и значительному снижению стойкости.

В процессе фрезерования стружка налипает на режущую кромку и препятствует ее работе в следующий момент врезания. При встречном фрезеровании это может привести к заклиниванию стружки между пластиной и заготовкой и, соответственно, к повреждению пластины. Попутное фрезерование позволяет избежать подобных ситуаций. На современных станках с ЧПУ, которые обладают высокой жесткостью, виброустойчивостью и у которых отсутствуют люфты в сопряжении ходовой винт-гайка, применяется в основном попутное фрезерование.

**Припуск** – слой материала заготовки, который необходимо удалить при обработке. Припуск можно удалить в зависимости от его величины за один или несколько проходов фрезы.

Принято различать черновое и чистовое фрезерования. При **черновом фрезеровании** обработку производят с максимально допустимыми режимами резания для выборки наибольшего объема материала за минимальное время. При этом, как правило, оставляют небольшой припуск для последующей чистовой обработки. **Чистовое фрезерование** используется для получения деталей с окончательными размерами и высоким качеством поверхностей.

## 2.2. Режущий инструмент

Весь инструмент, использующийся в металлообработке, можно условно подразделить на **режущий** инструмент (фрезы, сверла, метчики и др.), непосредственно осуществляющий механическую обработку (резание), и **вспомогательный,**

служащий для закрепления режущего инструмента в шпинделе станка (патроны, державки, оправки).

Станки могут иметь различные базовые конусы шпинделя, а режущий инструмент, в свою очередь, изготавливается с различными видами хвостовиков.

**Базовый конус станка** – выход шпинделя, выполненный в соответствии с одним из стандартных вариантов исполнения. Различают метрические конусы (7:24 или ISO 7388.1), конусы Морзе (отечественные фрезерные станки или оборудование сверлильной группы), HSK (современные станки, предназначенные для высокоскоростной обработки).

Таким образом, вспомогательный инструмент является неким переходником между шпинделем станка и режущим инструментом. Совокупность режущего и вспомогательного инструментов называется **инструментальным блоком**. Отметим, что в инструментальном блоке могут находиться несколько вспомогательных инструментов и только один режущий (основной). Большие инструментальные блоки снижают жесткость технологической системы и уменьшают точность установки режущего инструмента, в результате чего ухудшаются условия обработки и качество изделия.

*Рис. 2.4. Фрезы с механическим креплением режущих пластин*

По **технологическому признаку** различают фрезы для обработки плоскостей, пазов и шлицев, зубчатых колес, резьбы, фасонных поверхностей, для разрезки материала и т. д.

По **конструктивному признаку** различают:

- по устройству фрезы (цельные, составные, со вставными зубьями);
- по конструкции зуба (с острозаточенными, с затылованными зубьями);

- по направлению зуба (прямые, наклонные, винтовые зубья);
- по способу крепления (насадные, хвостовые – с цилиндрическим или коническим хвостовиком).

По **материалу**, из которого они изготовлены: быстрорежущая сталь, твердый сплав и др.

В современной инструментальной практике львиную долю составляет цельный твердосплавный или быстрорежущий инструмент, а также инструмент с механическим креплением режущих частей (пластин).

Твердые сплавы допускают работу со скоростями резания, превышающими в 5–10 раз скорости обработки быстрорежущими инструментальными сталями, обладают большей температурной стойкостью и износостойкостью.

При выборе фрезы технолог прежде всего руководствуется следующими параметрами:

- диаметр и длина рабочей части;
- форма профиля рабочей части;
- материал рабочей части;
- количество зубьев (режущих граней);
- форма и размер крепежной части.

*Рис. 2.5. Цельные концевые фрезы*

Обычная концевая фреза имеет несколько режущих зубьев (2, 3, 4, 6 или 8) и прямоугольный профиль режущей части. Зубья фрезы разделены винтовыми канавками, которые обеспечивают отвод стружки из зоны резания. В случае, когда необходимо получить переход от одной поверхности к другой с определенным радиусом, применяют фрезы со сферическим концом или с небольшим радиусом в основании профиля. Фрезы со сферическим концом и шаровые фрезы часто используются при обработке поверхностей сложной формы, например штампов и пресс-форм. Конические фрезы предназначены для фрезерования наклонных поверхностей и поднутрений.

Концевые фрезы наиболее универсальны – они позволяют обрабатывать плоскости, пазы и уступы. Существуют и другие типы фрез: торцовые, дисковые, пазовые. Эти фрезы, как правило, служат для выполнения фрезерных операций «узкой» направленности. Например, торцовая фреза – это лучший инструмент для фрезерования открытой плоскости, а дисковая – для обработки глубокого узкого паза за один проход.

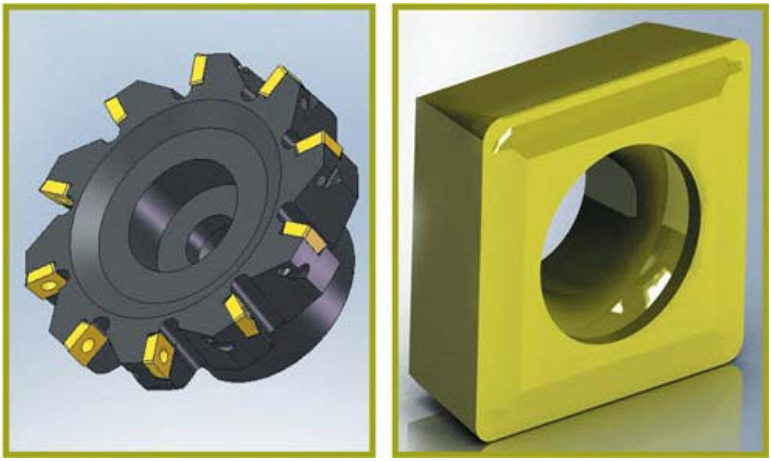


*Рис. 2.6. Торцовая фреза и режущая пластина*

Широкое распространение получили фрезы с механическим креплением пластин из твердого сплава и других инструментальных материалов. На корпусах таких фрез имеются специальные посадочные места, в которые устанавливаются пластины. Крепление пластин к стальному корпусу, как правило, осуществляется при помощи обычных винтов. Пластины имеют несколько граней, и в случае износа одной из них существует возможность развернуть пластину «свежей» гранью. Когда износятся все грани, то пластину можно выбросить и поставить новую. Получается очень экономичное решение, поскольку цельные твердосплавные фрезы стоят довольно дорого. Современные режущие пластины проектируются с учетом работы в различных условиях и отличаются геометрией передней поверхности.

**Шаг зубьев** фрезы может быть крупным, нормальным и мелким. Фрезы с различным шагом зубьев предназначены для различных условий обработки с точки зрения ее стабильности, энергозатрат и наличия склонности к вибрациям. Уменьшенное количество пластин – стандартное решение для производительной обработки при недостаточной мощности станка или низкой жесткости системы СПИД (станок – приспособление – инструмент – деталь). Фрезы с нормальным шагом универсальны для большинства операций. Мелкий шаг или максимальное число пластин на корпусе фрезы данного диаметра рекомендуется использовать для обработки при высокой жесткости системы СПИД, а также при фрезеровании материалов, дающих элементную стружку, титановых и жаропрочных сплавов.

На толщину срезаемого слоя при фрезеровании влияет **главный угол в плане**, который измеряется между главной режущей кромкой пластины и обрабатываемой поверхностью. Уменьшение угла в плане ведет к образованию более тонкой стружки для данного диапазона подач. Уменьшение толщины стружки происходит из-за распределения одного и того же объема снимаемого металла на большей длине режущей кромки. При меньшем угле в плане режущая кромка постепенно входит в работу и выходит из нее. Это уменьшает радиальную составляющую силы резания и защищает режущую кромку от возможных поломок. С другой стороны, неблагоприятным фактором является увеличение осевой составляющей силы резания, что вызывает ухудшение шероховатости поверхности тонкостенных деталей.

**При** угле в **плане 90°** сила резания направлена радиально в соответствии с направлением подачи. Основная область применения таких фрез – обработка прямоугольных уступов.

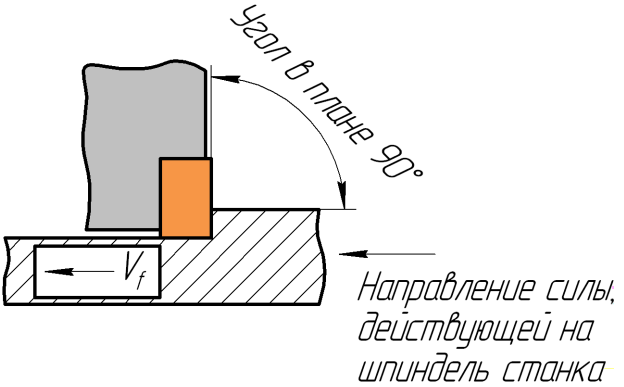



*Рис. 2.7. Угол в плане 90°*

При работе **фрезой** с углом **в плане 45°** осевые и радиальные силы резания практически одинаковы и потребляемая мощность невысока. Это фрезы универсального применения. Особенно они рекомендуются для обработки материалов, дающих элементную стружку и склонных к выкрашиванию при значительных радиальных усилиях на выходе инструмента. При врезании инструмента меньше нагрузка на режущую кромку и меньше склонность к вибрациям при закреплении в приспособлениях с небольшими усилиями зажима. Меньшая толщина срезаемого слоя при угле в плане 45° позволяет увеличивать минутную подачу стола, то есть повысить производительность обработки.

**Фрезы** с углом **в плане 10°** рекомендуются для продольного фрезерования с большими подачами и плунжерного фрезерования, когда характерны небольшие толщины стружки и высокие скоростные параметры. Преимуществом обработки такими фрезами являются низкие радиальные усилия резания. А также преобладание осевой составляющей силы резания как при радиальном, так и при осевом направлении подачи, что уменьшает склонность к вибрациям и предоставляет большие возможности для увеличения скоростей снятия материала.

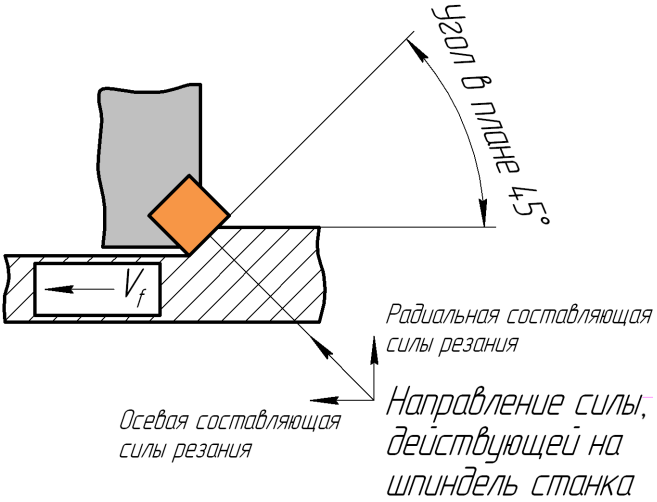



*Рис. 2.8. Угол в плане 45°*

**У фрез с круглыми пластинами** главный угол в плане меняется от 0 до 90° в зависимости от глубины резания. Эти фрезы имеют очень прочную режущую кромку и могут работать при больших подачах, поскольку образуют довольно тонкую стружку на большой длине режущей кромки. Фрезы с круглыми пластинами рекомендуется применять для обработки труднообрабатываемых материалов, таких как титан и жаропрочные сплавы. Направление сил резания меняется вдоль радиуса пластины, поэтому направление суммарной нагрузки зависит от глубины резания. Современная геометрия круглых пластин делает их более универсальными, обеспечивая стабильность процесса резания, меньшую потребляемую мощность и, соответственно, меньшие требования к жесткости оборудования. В настоящее время эти фрезы широко используются для снятия больших объемов металла.

*Рис. 2.9. Фрезы с круглыми пластинами*

# 2.3. Вспомогательный инструмент

Основная задача вспомогательного инструмента – надежная фиксация режущего инструмента в шпинделе и передача ему крутящего момента от станка. В качестве вспомогательного инструмента на операциях фрезерования используют патроны и оправки.

**Оправки** главным образом предназначены для операций с большими усилиями резания, таких как торцовое фрезерование, фрезерование пазов дисковыми фрезами, растачивание отверстий большого диаметра.

Элементом, передающим крутящий момент у оправок, является шпонка, которая предотвращает проворот режущего инструмента относительно оправки. Этим обеспечиваются надежное закрепление и передача крутящего момента. Однако оправки не способны обеспечить хорошее центрирование инструмента, поэтому основное их применение – черновые операции с удалением основного объема материала.

**Патроны** обеспечивают лучшее центрирование и обычно используются для зажатия режущих инструментов небольшого размера. Следует различать патроны с механическим креплением режущего инструмента (для сверл, инструментов с коническими хвостовиками типа конусов Морзе, Whistle Notch, Weldon и др.) и патроны с упруго-деформируемой зажимной частью (цанговые, гидромеханические, гидропластовые и др.).

Особое внимание следует уделить **цанговым патронам** (рис. 2.10), которые наиболее часто используются при работе на станках с ЧПУ. Принцип действия такого патрона очень прост. В коническое отверстие патрона вставляются смен-

*Рис. 2.10. Цанговые патроны и сменные цанги*

ные цанги. Цанга имеет цилиндрическое отверстие, в которое устанавливается цилиндрический хвостовик режущего инструмента (диаметр хвостовика режущего инструмента должен соответствовать номеру цанги). При затягивании гайки давление передается на торец цанги, что приводит к вдавливанию последней в коническое отверстие патрона и сжатию в радиальном направлении. Сжимаясь, цанга передает давление на цилиндрический хвостовик режущего инструмента и надежно его закрепляет. После снятия давления (откручивания гайки) с цанги она разжимается и позволяет извлечь режущий инструмент из патрона.

Основное преимущество цангового патрона – способность осуществлять закрепление широкого диапазона режущих инструментов при помощи комплекта сменных цанг. В комплекте цанг к одному и тому же патрону вы, как правило, найдете цанги для закрепления инструментов с хвостовиками от 6 до 30 мм и более. Цанга производит хорошее центрирование инструмента и надежное закрепление, однако плохо сбалансирована для скоростных методов обработки.

Следует уделять должное внимание вспомогательному инструменту, так как от него зависят: стойкость режущего инструмента, стабильность технологического процесса, точность и качество обработки.

## 2.4. Основные определения и формулы

**Скорость резания V (м/мин)** – это окружная скорость перемещения режущих кромок фрезы. Эта величина определяет эффективность обработки и лежит в рекомендованных для каждого инструментального материала пределах.

За один оборот фрезы точка режущей кромки, находящаяся на окружности фрезы диаметра D (мм), сможет пройти путь, равный длине окружности, то есть $\pi D$. Для того чтобы определить длину пути, пройденного точкой за одну минуту, нужно умножить длину пути за один оборот на частоту вращения фрезы, то есть $\pi DN$ (мм/мин). Таким образом, формула для определения скорости резания будет следующей:

$V = \pi DN/1000$ (мм/мин).

**Частота вращения шпинделя N** (об/мин) равняется числу оборотов фрезы в минуту. Вычисляется в соответствии с рекомендованной для данного типа обработки скоростью резания:

$N = 1000V/nD$ (об/мин).

При фрезеровании различают минутную подачу, подачу на зуб и подачу на оборот фрезы.

**Подача на зуб Fz** (мм/зуб) – величина перемещения фрезы или рабочего стола с заготовкой за время поворота фрезы на один зуб.

**Подача на оборот Fo** (мм/об) – величина перемещения фрезы или рабочего стола с заготовкой за один оборот фрезы. Подача на оборот равняется произведению подачи на зуб на число зубьев фрезы Z:

Fo = FzZ (мм/об).

**Минутной подачей Fm (мм/мин)** называется величина относительного перемещения фрезы или рабочего стола с заготовкой за одну минуту. Минутная подача равняется произведению подачи на оборот на частоту вращения фрезы:

Fm = FoN = FzZN (мм/мин).

**Глубиной фрезерования h (мм)** называется расстояние между обработанной и необработанной поверхностями, измеряемое вдоль оси фрезы.

**Шириной фрезерования b (мм)** называется величина срезаемого припуска, измеренная в радиальном направлении, или ширина контакта заготовки и инструмента.

**Производительность снятия материала Q (см³)** – это объем удаляемого материала в единицу времени, определяемый глубиной, шириной обработки и величиной подачи.

$Q = (h \times b \times Fm)/1000.$

# 2.5. Рекомендации по фрезерованию

**Выбор диаметра фрезы** зависит, как правило, от ширины обрабатываемой заготовки, а также от мощностных характеристик станка. При этом важным фактором, определяющим успешное выполнение операции фрезерования, является взаимное расположение обрабатываемой поверхности и фрезы.

Ширина фрезерования особенно сильно влияет на выбор диаметра фрезы при обработке торцовыми фрезами. В этом случае рекомендуется выбирать диаметр фрезы, превышающий ширину фрезерования на 20–50%.

Если обработка может быть произведена за несколько проходов, то ширина резания за каждый проход должна быть равной 3/4 диаметра фрезы. При этом формирование стружки и нагрузка на режущую кромку будут оптимальными.

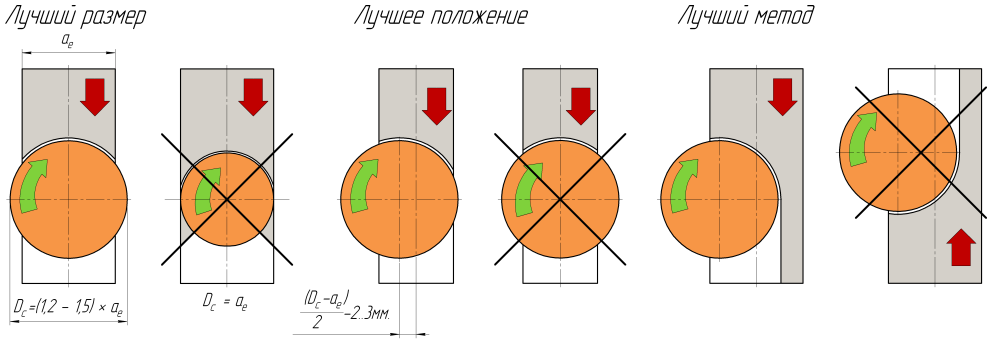



*Рис. 2.11. Варианты расположения фрезы относительно заготовки*

Когда диаметр фрезы значительно превышает ширину заготовки, то ось фрезы следует сместить с оси симметрии заготовки. Конечно, близкое расположение оси фрезы к оси заготовки позволяет обеспечить наикратчайший путь зубьев фрезы в металле, надежное формирование стружки на входе и благоприятную ситуацию относительно ударных нагрузок на пластину. Но когда ось фрезы расположена точно по оси симметрии заготовки, циклическое изменение силы резания при врезании и выходе может привести к возникновению вибраций, которые приведут к повреждению пластины и плохой шероховатости поверхности.

**При торцевом фрезеровании** по возможности избегайте фрезерования плоскостей с пересечением пазов и отверстий, так как при этом режущие кромки будут работать в неудовлетворительных условиях прерывистого резания. Выполняйте операцию изготовления отверстий после фрезерования. Если такой вариант невозможен, то при пересечении фрезой отверстия снижайте величину подачи на 50% от рекомендованной.

**При обработке больших плоскостей** старайтесь не прерывать контакт фрезы с заготовкой, обходя поверхность по периметру, а не за несколько параллельных проходов. Обработку углов необходимо осуществлять по радиусу, превышающему радиус фрезы, чтобы исключить возможность возникновения вибраций, связанных с резким увеличением угла охвата фрезы.

**При обработке закрытого паза** (замкнутой области) основная проблема заключается в трудности первоначального входа инструмента в материал заготовки, так как большинство концевых фрез плохо работают на засверливание. Существует несколько способов решения этой проблемы. Самый простой выход из ситуации – предварительно просверлить технологическое отверстие и затем спокойно опустить в него фрезу. Более интересные способы – маятниковое и спиральное

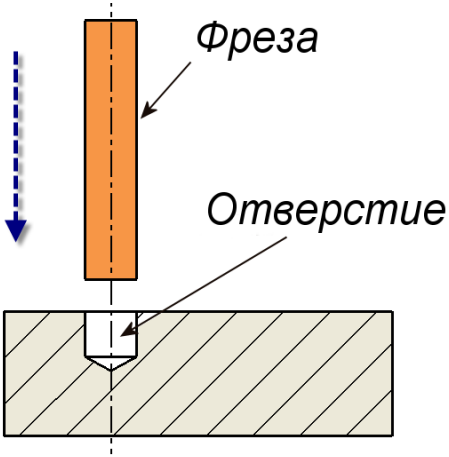



*Рис. 2.12. Для вертикального входа инструмента желательно заранее просверлить отверстие на глубину последующего фрезерования*

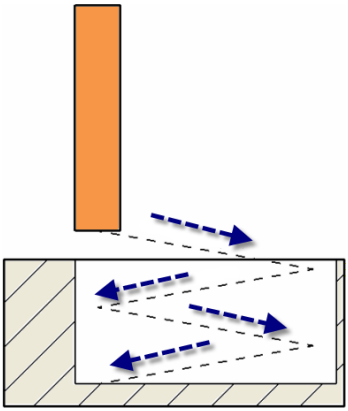


*Рис. 2.13. Маятниковое врезание (фрезерование)*

врезания. В этом случае отпадает необходимость предварительного сверления, режущий инструмент входит в материал заготовки плавно.

Особую осторожность нужно соблюдать при обработке **тонкостенных карманов**. Желательно, чтобы материал выбирался постепенно и обязательно при попутном способе фрезерования. В противном случае появляется вероятность «подрыва» тонкой стенки.

**Обрабатывая внутренние радиусы**, старайтесь, чтобы радиус фрезы был немного меньше, чем радиус в углу кармана (контура). Дело в том, что в момент, когда фреза входит в угол, **ширина** фрезерования возрастает скачкообразно, что может привести к «подхвату» инструмента и, как следствие, подрезать обрабатываемый контур или сломать фрезу. При назначении диаметра фрезы для черновой обработки внутренних радиусов желательно, чтобы оставляемый в углах припуск не **превышал** 0,20×D, где D – диаметр последующей чистовой фрезы.

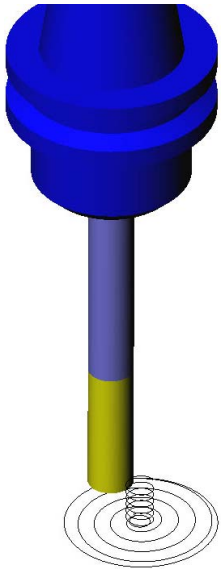


*Рис. 2.14. Траектория спирального врезания с выходом в плоскость обработки*

**При обработке глубоких контуров и уступов** необходимо обеспечить достаточную жесткость инструмента во избежание его отжима и исключения «конусности» обработанной поверхности. Желательно, чтобы диаметр инструмента D удовлетворял условию H < 2,5D, где H – максимальная высота стенки обрабатываемой детали.

Часто фрезерование производится в два этапа: черновой – контур обрабатывается послойно с небольшим припуском, чистовой – оставшийся припуск удаляется за один проход фрезы на финальной глубине.

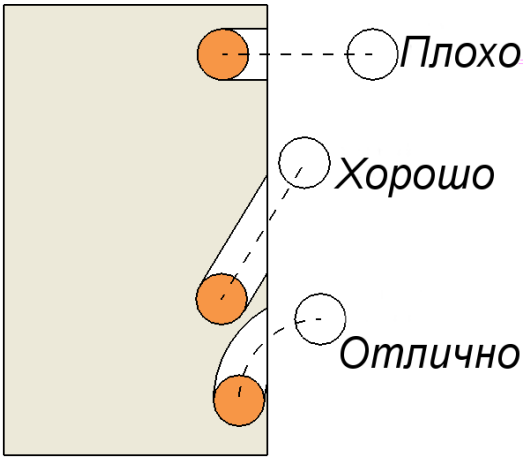



*Рис. 2.15. Простейшие способы подвода инструмента*

При выполнении чернового и особенно чистового фрезерования **инструмент следует подводить к обрабатываемой поверхности по касательной или по прямой линии под острым углом**. Следуя этому правилу, необходимо и отводить инструмент. Дело в том, что при первоначальном врезании в материал заготовки фреза подвергается резкой нагрузке, что может привести к ее поломке или к тому, что на поверхности детали в месте входа фрезы в материал останется след или неровность. Если врезание будет плавным, то нагрузка на инструмент будет возрастать постепенно и поверхность останется «чистой».

# Краткое изложение главы

- Процесс фрезерования заключается в срезании с заготовки лишнего слоя материала для получения детали требуемой формы, размеров и шероховатости обработанных поверхностей.
- При обработке различают встречное и попутное фрезерования.
- Весь инструмент, использующийся в металлообработке, можно условно подразделить на режущий инструмент (фрезы, сверла, метчики и др.), непосредственно осуществляющий механическую обработку (резание), и вспомогательный, служащий для закрепления режущего инструмента в шпинделе станка (патроны, державки, оправки).
- Твердые сплавы допускают работу со скоростями резания, превышающими в 5–10 раз скорости обработки быстрорежущими инструментальными сталями, обладают большей температурной стойкостью и износостойкостью.
- Широкое распространение получили фрезы с механическим креплением пластин из твердого сплава и других инструментальных материалов.
- Основная задача вспомогательного инструмента – надежная фиксация режущего инструмента в шпинделе и передача ему крутящего момента от станка.

# Вопросы

1. В чем отличие встречного фрезерования от попутного?
2. Для чего применяется вспомогательный инструмент?
3. Перечислите основные типы конусов шпинделя станка.
4. В чем преимущество фрез с механическим креплением режущих пластин перед цельными фрезами?
5. Опишите принцип действия цангового патрона.
6. Что такое скорость резания и чем она определяется?
7. Для чего применяются маятниковое и спиральное фрезерования?
8. Почему инструмент рекомендуется подводить к обрабатываемой поверхности по касательной?

# Глава 3
# ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ ОБРАБОТКИ

## 3.1. Прямоугольная система координат

Прежде чем приступить к созданию первой управляющей программы, вы должны вспомнить, что такое прямоугольная система координат. Ведь именно **прямоугольная система координат служит математической базой программирования обработки**. Более 300 лет назад французский математик Декарт придумал систему, которая позволяет человеку описать положение любой точки в пространстве. В простейшем случае прямоугольная система координат представляет собой две пересекающиеся под прямым углом линии. Эти линии называются осями, а точка их пересечения является **началом координат**.

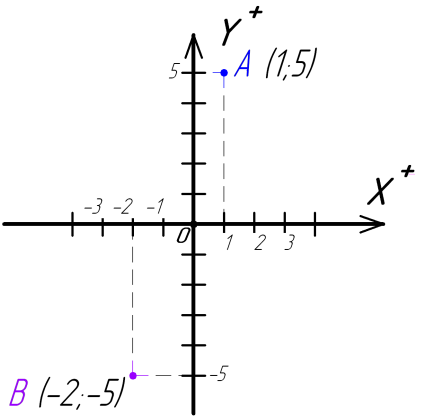



*Рис. 3.1. Прямоугольная система координат на плоскости*

Оси обозначаются буквами X и Y. Координатная система с двумя осями X и Y позволяет описать **положение точки на плоскости**. Расстояние от начала координат до точки A вдоль оси X является x-координатой этой точки. Расстояние от начала координат до точки A вдоль оси Y является y-координатой этой точки.

Координаты точки принято указывать в скобках. Сначала пишется координата по оси X, а затем по оси Y. Таким образом, на рис. 3.1 находится точка A(1; 5).

У каждой оси есть **положительное и отрицательное направления**. Когда координата имеет отрицательное значение, то это означает, что точка лежит либо левее начала координат, либо ниже. Например, точка B имеет следующие координаты: x = –2, y = –5. Если точка лежит на какой-либо оси, то одна из ее координат обязательно равна нулю.

Пересечение трех взаимно перпендикулярных плоскостей образует трехмерную систему координат, которая используется для описания **положения точки в пространстве**. К двум имеющимся осям X и Y добавляется третья ось Z. Координаты точки также указываются в скобках и идут в алфавитном порядке (x; y; z).

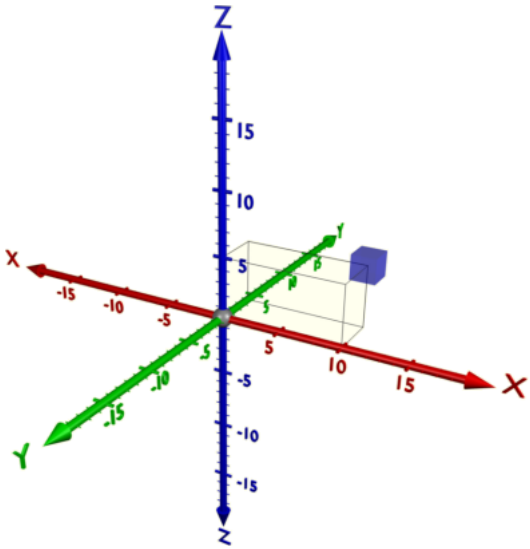



*Рис. 3.2. Прямоугольная пространственная система координат*

# 3.2. Написание простой управляющей программы

Детали, обрабатываемые на станке с ЧПУ, можно рассматривать как геометрические объекты. Во время обработки вращающийся инструмент и заготовка перемещаются относительно друг друга по некоторой траектории. УП описывает движение определенной точки инструмента – его центра. Траекторию инструмента представляют состоящей из отдельных, переходящих друг в друга участков. Этими участками могут быть прямые линии, дуги окружностей, кривые второго или высших порядков. Точки пересечения этих участков называются **опорны-**

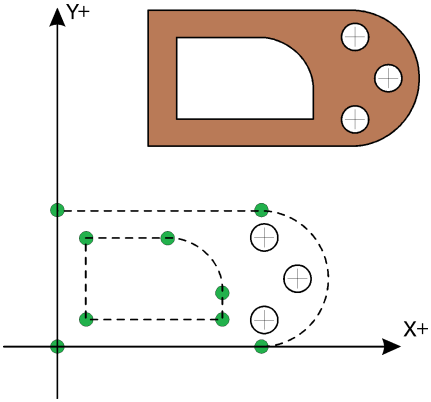



*Рис. 3.3. Любую деталь можно представить в виде совокупности геометрических элементов. Для создания программы обработки необходимо определить координаты всех опорных точек*

**ми**, **или узловыми**, **точками**. Как правило, в УП содержатся координаты именно опорных точек.

Попробуем написать небольшую программу для обработки паза, представленного на рис. 3.4. Зная координаты опорных точек, сделать это несложно. Мы не будем подробно рассматривать код всей УП, а обратим особое внимание на написание строк (кадров УП), непосредственно отвечающих за перемещение через опорные точки паза. Для обработки паза сначала нужно переместить фрезу в точку T1 и опустить ее на соответствующую глубину. Далее необходимо переместить фрезу последовательно через все опорные точки и вывести инструмент вверх из материала заготовки. Найдем координаты всех опорных точек паза и для удобства поместим их в табл. 3.1.

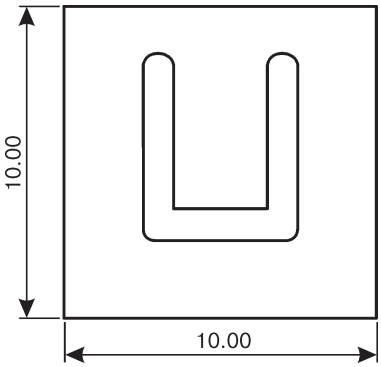



*Рис. 3.4. Необходимо создать программу для обработки паза. Глубина паза равна 1 мм*

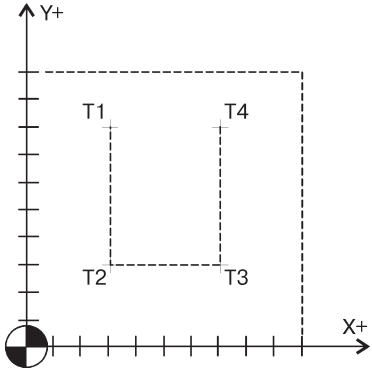



*Рис. 3.5. Поместим деталь в прямоугольную систему координат и найдем координаты четырех опорных точек*

***Таблица 3.1. Координаты опорных точек паза***

| Точка | Координата по оси X | Координата по оси Y |
|---|---|---|
| TI | 3 | 8 |
| T2 | 3 | 3 |
| T3 | 7 | 3 |
| T4 | 7 | 8 |

Подведем режущий инструмент к первой опорной точке:

N50 G00 X3 Y8

Следующие два кадра заставляют инструмент опуститься на требуемую глубину в материал заготовки.

N60 G00 Z0.5
N70 G01 Z-l F25

Как только инструмент окажется на нужной глубине (1 мм), можно перемещать его через все опорные точки для обработки паза:

N80 G01 X3 Y3
N90 G01 X7 Y3
N100 G01 X7 Y8

Теперь следует вывести инструмент из материала заготовки – поднять на небольшую высоту:

N110 G01 Z5

Соберем все кадры вместе, добавим несколько вспомогательных команд и получим окончательный вариант программы:

| **Кадры УП** | **Описание кадра** |
|---|---|
| % | Символ начала программы |
| O0001 (PAZ) | Номер программы (0001) и ее название (PAZ) |
| N10 G21 G40 G49 G54 G80 G90 | Строка безопасности |
| N20 M06 T01 (FREZA D1) | Вызов инструмента № 1 |
| N30 G43 H01 | Компенсация длины инструмента № 1 |
| N40 M03 S1000 | Включение оборотов шпинделя (1000 об/мин) |
| N50 G00 X3 Y8 | Ускоренное перемещение в опорную точку T1 |
| N60 G00 Z0.5 | Ускоренное перемещение инструмента в Z0.5 |
| N70 G01 Z-l F25 | Перемещение на глубину 1 мм на подаче 25 мм/мин |
| N80 G01 X3 Y3 | Перемещение инструмента в точку T2 (25 мм/мин) |
| N90 G01 X7 Y3 | Перемещение инструмента в точку T3 (25 мм/мин) |
| N100 G01 X7 Y8 | Перемещение инструмента в точку T4 (25 мм/мин) |

| | |
|---|---|
| N110 G01 Z5 | Подъем инструмента вверх в Z5 (25 мм/мин) |
| N120 M05 | Выключение оборотов шпинделя |
| N130 M30 | Завершение программы |
| % | Символ конца программы |

# 3.3. Создание УП на персональном компьютере

Существуют два способа для записи (набора) управляющих программ:

- при помощи стойки ЧПУ станка (цеховое программирование);
- используя персональный компьютер с последующей передачей в стойку ЧПУ.

**Цеховое программирование** в настоящее время считается малоэффективным и используется крайне редко. Во-первых, клавиши стойки ЧПУ менее удобны, чем клавиатура ПК. Во-вторых, программное обеспечение СЧПУ предоставляет меньшие возможности по редактированию программ. В-третьих, ручной ввод УП в память СЧПУ физически мешает оператору запустить процесс обработки деталей на этом станке. Исключение составляют, пожалуй, лишь стойки со встроенной CAM-системой.

**Набор текста программы обработки на компьютере** с последующей передачей в СЧПУ станка является гораздо более эффективным способом работы.

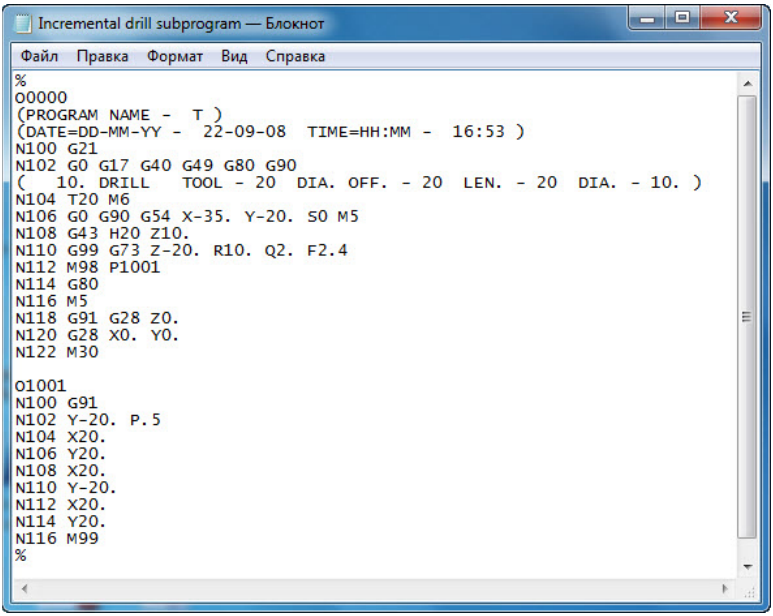



*Рис. 3.6. Записать УП можно в обычном текстовом редакторе. Например, в Блокноте операционной системы Windows*

Код УП можно набирать в любом текстовом редакторе и сохранять в соответствующем формате. Например, использовать всем известный Блокнот из стандартного набора операционной системы Windows.

Есть множество различных текстовых редакторов, которые были специально созданы для работы с кодом УП. Эти редакторы (назовем их редакторами УП) предоставляют широкие возможности по написанию и редактированию станочного кода. Например, они позволяют добавлять или удалять пробелы, автоматически нумеровать строки и перемещать курсор к коду смены инструмента. Эти функции не нужны обычному текстовому редактору, но очень полезны при создании и отладке программ. Наиболее продвинутые редакторы УП имеют инструменты для графической проверки кода и трансляции его на станок.

Стоит заметить, что некоторые текстовые редакторы сохраняют файлы в специальном формате, который содержит информацию о размере шрифта, полях, цвете и др. Код УП не содержит таких данных, а состоит исключительно из «чистого» текста в формате Американского стандартного кода для обмена информацией (ASCII). Стандарт ASCII является открытым и может читаться любым текстовым редактором. Файлы такого формата, скорее всего, будут иметь расширение «.txt».

Станки с ЧПУ работают в формате G- и M-кодов в соответствии со стандартами EIA/ISO. Код этого стандарта аналогичен ASCII, но есть ряд небольших отличий. Формат ASCII использует коды окончания строки и перевода каретки в конце каждой строки. В текстовом редакторе для перехода на новую строку вы просто нажимаете клавишу **Enter**, но не видите кода, который переводит вас на новую строку, хотя на самом деле он присутствует. **Система ЧПУ требует, чтобы в конце каждого кадра УП стоял знак конца кадра, например (;) или (*).** Поэтому если вы пишите программу на компьютере, то она выглядит так:

```
N50 G00 X230
N60 G01 Y300
N70 G01 Z100
```

А если та же программа вводится прямо со стойки ЧПУ, то она, возможно, будет выглядеть следующим образом (придется добавить символ конца кадра в каждую строку УП):

```
N50 G00 X230;
N60 G01 Y300;
N70 G01 Z100;
```

Если вы создаете программу на компьютере и затем передаете ее на станок, то знак конца кадра помещается в конец каждого кадра УП при передаче в большинстве случаев автоматически.

**Ошибка в программе обработки может повлечь за собой массу проблем.** В лучшем случае ошибка обернется сломанным инструментом или «запоротой» деталью, а в худшем – может привести к повреждению станка или травме оператора. Опытный программист знает, что **дешевле и проще проверить программу заранее на компьютере**, чем ошибиться при выполнении обработки на станке. Основной метод проверки УП на компьютере заключается в графической симуляции обработки. Такая симуляция может выглядеть как прорисовка траектории центра

инструмента или как полная имитация механической обработки на станке с демонстрацией процесса удаления материала. В первом случае программист может обнаружить ошибку в УП, просто наблюдая за траекторией перемещения центра инструмента на мониторе компьютера. Такая симуляция называется бэкплотом (Backplot). Бэкплот позволяет легко увидеть ошибку, которую тяжело распознать при простом просмотре кода УП.

Сравните два варианта одной и той же программы (табл. 3.2). Во втором варианте в кадре N90 можно заметить, что десятичная точка стоит не на своем месте. Программа обработки может иметь достаточно большой объем, и такую маленькую ошибку распознать в коде будет крайне тяжело. Бэкплот моментально покажет, что траектория перемещения инструмента качественно не соответствует тому, что было задумано программистом.

***Таблица 3.2. В одном из кадров УП допущена ошибка***

| Правильный вариант | Ошибочный вариант |
|---|---|
| N80 G01 X3. Y3. | N80 G01 X3. Y3. |
| N90 G01 X7. Y3. | N90 G01 X7. Y.3 |

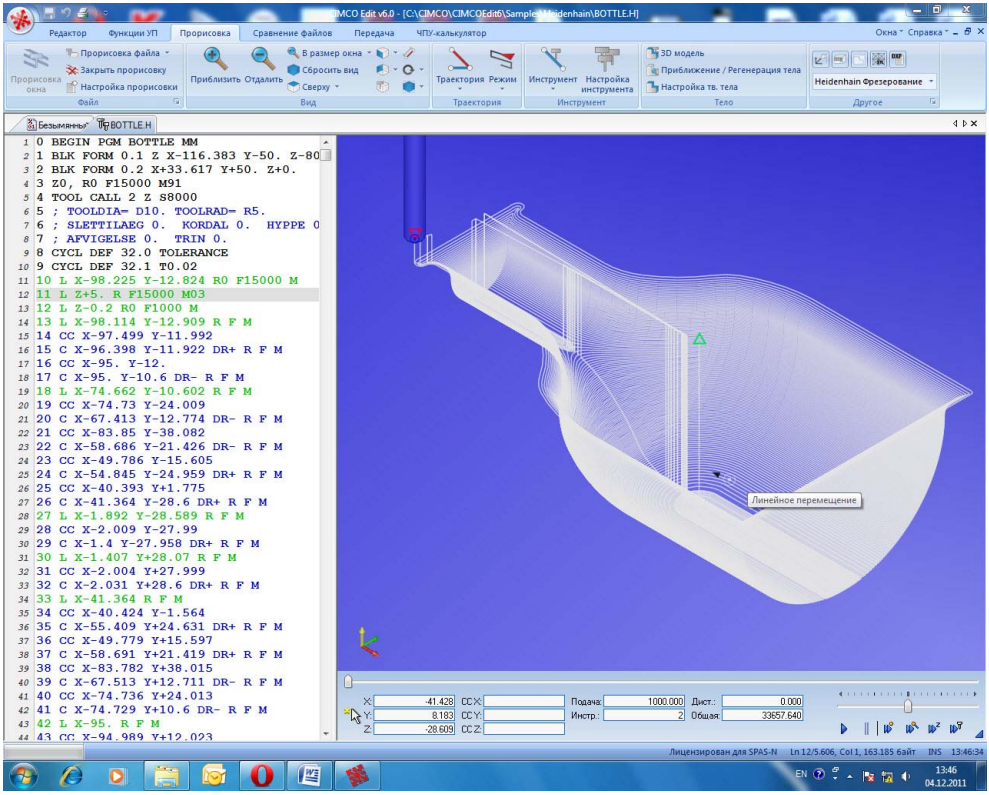



*Рис. 3.7. Прорисовка траектории в редакторе Cimco Edit*

**Твердотельная верификация** (от англ. *verification* – проверка) является мощным инструментом по проверке УП на компьютере. В отличие от бэкплота, программы твердотельной верификации демонстрируют процесс удаления материала заготовки и позволяют увидеть полный результат работы УП – модель готовой детали. На полученную «виртуальную» деталь можно посмотреть с разных сторон и понять, все ли элементы выполнены правильно, нет ли зарезов и столкновений инструмента с заготовкой.

Бэкплот обычно используют в процессе написания и отладки программы, а твердотельную верификацию – лишь на заключительном этапе проверки.

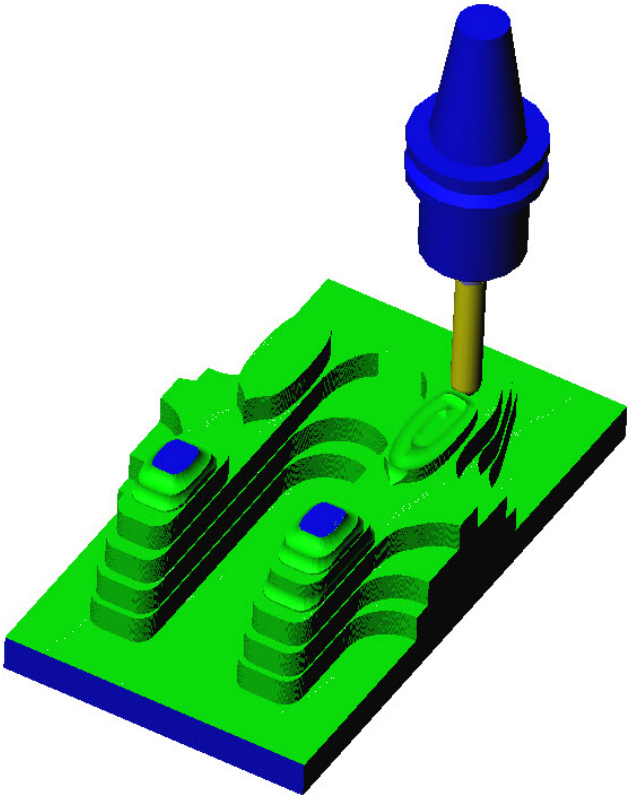


*Рис. 3.8. Твердотельная верификация – это удобный и наглядный способ проверки управляющей программы*

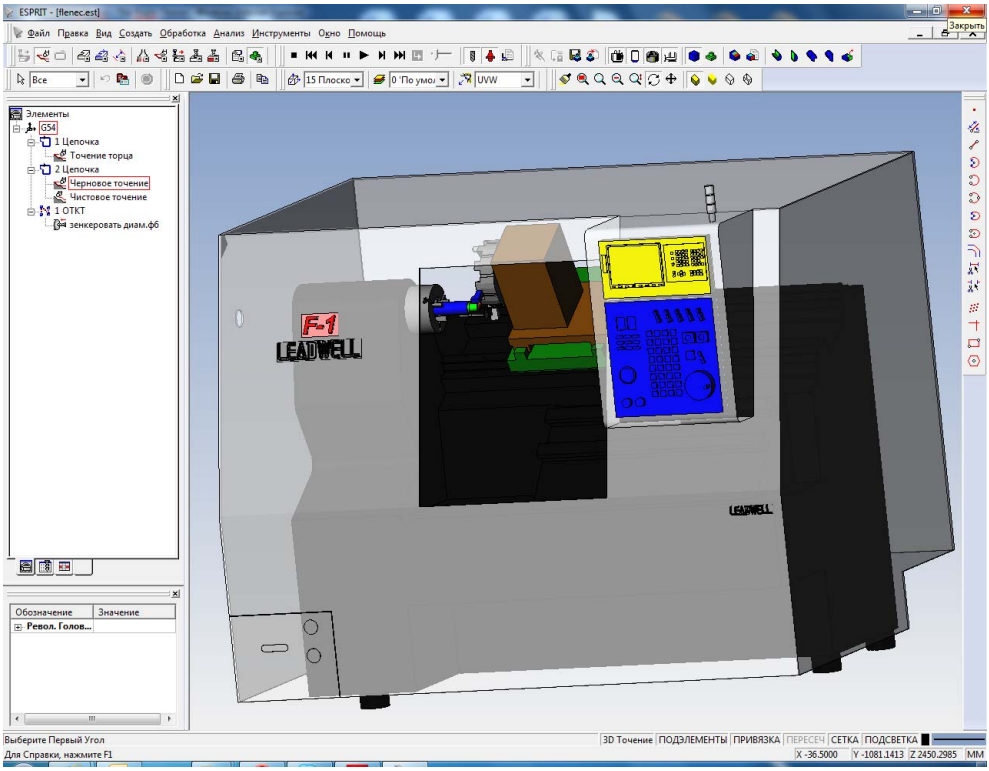



*Рис. 3.9. Твердотельная верификация с проверкой столкновений узлов станка называется **симуляцией обработки***

# 3.4. Передача управляющей программы на станок

После того как вы создали и проверили программу обработки при помощи ПК, ее необходимо передать на станок. Для передачи УП с компьютера в СЧПУ станка используется специальное **коммуникационное программное обеспечение**. В большинстве случаев связь осуществляется в соответствии со стандартом **RS-232**. При этом COM-порт компьютера соединяется кабелем со специальным разъемом на корпусе станка или панели УЧПУ. **Для передачи данных необходимо, чтобы УЧПУ станка и коммуникационная программа были синхронизированы.** Это достигается соответствующей настройкой параметров СЧПУ и комму-

никационной программы. Как правило, коммуникационная программа и кабель поставляются вместе со станком, а информацию о настройке параметров СЧПУ для связи станка и компьютера можно получить из «родной» документации. Стоит учитывать, что при передаче данных в соответствии с RS-232 желательно, чтобы длина кабеля не превышала 15 метров.

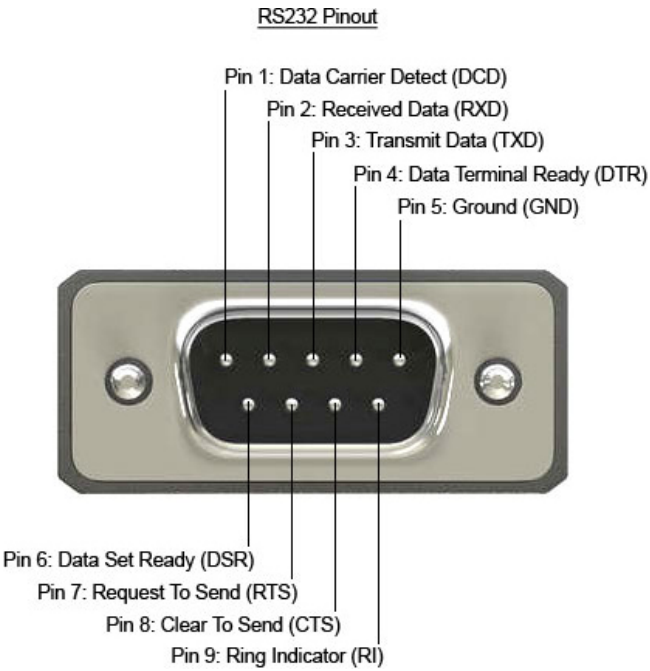



*Рис. 3.10. Вариант разъема RS-232*

Для передачи УП, размер которых превышает размер свободной памяти СЧПУ, используется режим DNC. **Режим DNC**, **или режим прямого числового управления**, позволяет выполнять программу обработки прямо из компьютера, не записывая ее в память СЧПУ. УП считывается из компьютера в буфер памяти СЧПУ кадр за кадром (точнее, порциями). Как только система определяет, что один кадр выполнен, она его удаляет и загружает следующий, и так далее – до конца программы. Для работы в режиме прямого числового управления необходимо, чтобы СЧПУ станка было соответствующим образом подготовлено производителем, а на персональном компьютере находилась коммуникационная программа с поддержкой DNC-режима.

Некоторые станки оборудованы собственными дисководами, что дает возможность передавать УП и другие данные в СЧПУ при помощи традиционных программных носителей – дискет и флэш-карт.

Самые «продвинутые» стойки ЧПУ поддерживают работу в локальной сети, что позволяет передавать данные более быстро и удобно, а некоторые из них позволяют выходить в Интернет и обеспечивают возможность дистанционного мониторинга системы и решения проблем непосредственно производителем станка в режиме реального времени.

# 3.5. Проверка управляющей программы на станке

## Общие сведения

После того как вы написали УП, проверили ее на компьютере и передали в память СЧПУ, **необходимо провести тест программы обработки прямо на станке**. Дело в том, что не все ошибки, содержащиеся в программе, могут быть распознаны инструментами верификации на компьютере. Вы можете легко увидеть, что фреза идет не в ту координату, но можете не заметить, что нет команды на включение вращения шпинделя или на подачу смазывающе-охлаждающей жидкости (СОЖ).

Но даже если в коде УП нет ни одной ошибки, оператор станка может случайно внести в корректора СЧПУ неверные значения длины и радиуса инструментов или ошибиться в «привязке» к детали, что неизбежно приведет к браку. **Для того чтобы исключить любую ошибку, перед изготовлением реальной детали прямо на станке производят пробный прогон или тестовую обработку.** В табл. 3.3 приведены типичные ошибки программиста или оператора станка с ЧПУ и их возможные последствия.

***Таблица 3.3. Возможные ошибки программиста и оператора***

| Ошибка | Последствия |
|---|---|
| Не включена подача СОЖ при обработке, когда это необходимо | Испорченный инструмент<br>Ухудшение качества поверхности |
| Неправильные режимы резания | Испорченный инструмент/<br>Заготовка сдвигается в приспособлении |
| В инструментальном магазине станка находится неправильный инструмент | Деталь с неверными размерами (брак) |
| Не включены обороты шпинделя при обработке | Испорченный инструмент/<br>Испорченная заготовка |
| Неправильный порядок выполнения операций. Например, сначала нарезание резьбы, затем сверление (должно быть наоборот) | Сломанный инструмент/<br>Испорченная заготовка |
| В корректорах стоят неправильные значения | Испорченный инструмент или заготовка/<br>Деталь с неверными размерами (брак) |
| Траектория перемещения инструмента пересекается с приспособлением | Сломанный инструмент/<br>Испорченное приспособление |
| При обработке нескольких карманов инструмент после обработки первого из них не был возвращен в безопасную позицию над деталью для перемещения к следующему карману | Испорченная деталь/<br>Сломанный инструмент |

***Таблица 3.3. Возможные ошибки программиста и оператора (окончание)***

| Ошибка | Последствия |
|---|---|
| Оператор станка неправильно «привязался» к обрабатываемой детали (неправильно установлена рабочая система координат) | Деталь с неверными размерами (брак)/ Испорченный инструмент/ Испорченная заготовка |

## *Тестовые режимы станка с ЧПУ*

Большинство современных станков с ЧПУ имеют тестовые режимы для дополнительной проверки УП. Основной тестовой функцией является **отработка УП на холостых ходах** (Dry run). При запуске УП в этом режиме станок блокирует перемещения исполнительных органов по оси Z, по осям X и Y или по всем осям одновременно. Например, если ваш станок блокирует ось Z, то будут осуществляться перемещения по осям X и Y, при этом шпиндель будет вращаться как обычно. Это позволит вам спокойно посмотреть, как работает вся УП, без опасения «зарезаться» в материал заготовки. Вообще, поведение станка в этом режиме может быть различным, поэтому перед работой в режиме отработки УП на холостых ходах внимательно прочтите руководство оператора станка.

**Режим покадровой отработки** (Single block) предназначен для выполнения программы обработки по отдельным кадрам. СЧПУ приостанавливает работу в конце каждого кадра и ожидает, когда оператор нажмет клавишу **Старт цикла** для исполнения следующего кадра. Пользуясь этим режимом, легко увидеть, пришел ли инструмент в правильную позицию или в программе есть ошибка. Обычно режим покадровой отработки УП применяется совместно с режимом отработки УП на холостых ходах или при «поднятой» нулевой точке детали.

Может показаться, что перечисленные тестовые режимы станка с ЧПУ немного устарели в эпоху современных компьютеров и мощных программ верификации. С одной стороны, это так. Но, с другой стороны, эти режимы позволяют новому оператору проще и спокойнее учиться программированию обработки и реальной работе на станке.

Применяются и иные методы для цеховой проверки УП. Контролировать правильность перемещений в программе обработки оператору помогает экранный режим, который называется **Distance to GO** (Оставшееся расстояние). Многие СЧПУ имеют функцию отображения оставшегося расстояния перемещения по каждой из осей. Перейдя в этот режим, оператор видит, сколько миллиметров (дюймов) осталось переместиться инструменту в текущем кадре УП. К примеру, вы знаете, что фрезе нужно пройти до касания с поверхностью заготовки по оси Z 50 мм, а в строке Z экранного режима **Distance to GO** стоит 100 мм. Это означает, что фреза пройдет лишних 50 мм и врежется в материал заготовки. Таким образом, информация об оставшемся расстоянии перемещения позволяет оператору сравнивать фактическое и заданное расстояния перемещения визуально и дает возможность избежать серьезной ошибки.

Зачастую для проверки УП применяются методы, не использующие тестовых режимов станка с ЧПУ. Например, после установки рабочей системы координат («привязки» к детали) **можно поднять нулевую точку на безопасную высоту над**

**поверхностью заготовки**. Эта высота должна быть немного больше, чем самое «глубокое» перемещение инструмента в данной УП. Таким образом, выполнение данной программы обработки ничем не будет отличаться от заданного, за исключением того, что инструмент будет «резать воздух».

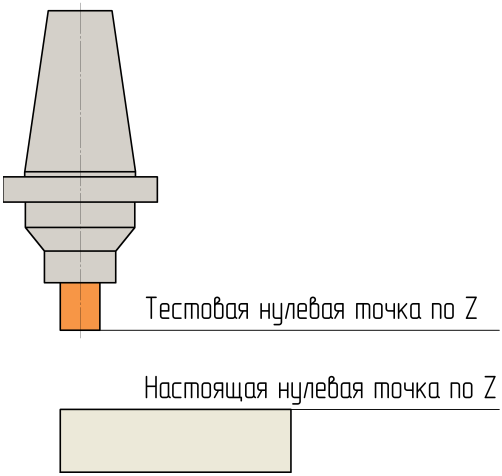



*Рис. 3.11. Очень часто перед настоящей обработкой детали «поднимают» нулевую точку по оси Z и УП «прогоняют по воздуху»*

Пожалуй, самым реалистичным способом проверки УП является настоящая обработка «ненастоящего» материала. **Обработав заготовки из специального воска или пластика, вы поймете, что на самом деле делает ваша программа.** При этом можно использовать очень высокие режимы резания и не бояться сломать инструмент или испортить заготовку из дорогостоящего металла.

Выбор того или иного способа проверки УП на станке зависит от многих факторов. Для начинающего программиста рекомендуется выполнять полную проверку. Профессионал же может ограничиться частичной проверкой или вообще проводить тесты УП только на персональном компьютере, используя инструменты бэкплота и твердотельной верификации. **При написании программы и работе на станке с ЧПУ главное – быть внимательным и не торопиться, а в случае сомнений лучше обратиться к документации станка или за советом к специалисту.**

## *Последовательность полной проверки УП*

1. Выполните графическую проверку кода программы на компьютере методами бэкплота и твердотельной верификации.
2. Осуществите дополнительные проверки программы и настройки станка:
   - все ускоренные перемещения выполняются над поверхностью заготовки на безопасном расстоянии;
   - в УП номера инструментов и номера корректоров совпадают;

- инструмент, установленный в инструментальном магазине станка, соответствует инструменту, описанному в УП;
- инструмент надежно закреплен в патроне;
- в стойке ЧПУ находятся правильные значения компенсации длины и радиуса инструментов;
- назначены верные режимы резания;
- шпиндель вращается в правильном направлении;
- в УП присутствует команда на включение СОЖ при необходимости;
- операции обработки выполняются в правильном порядке;
- черновые операции производятся перед чистовыми операциями;
- заготовка надежно закреплена в приспособлении;
- инструмент перемещается от одного обрабатываемого элемента детали к другому на безопасном уровне по оси Z.

3. Выполните графическую проверку программы на стойке ЧПУ, если это возможно.
4. Отработайте УП на холостых ходах:
   - включите режим покадровой отработки УП;
   - уменьшите скорость рабочей подачи;
   - сместите нулевую точку детали на безопасное расстояние над поверхностью заготовки и прогоните программу «по воздуху».
5. Верните нулевую точку из безопасного положения в нормальное положение, отмените режим покадровой отработки УП и выполните обработку детали с уменьшенными рабочими режимами.
6. Проверьте размеры детали. При необходимости произведите корректировку значений длины и радиуса инструментов или самой программы.

## 3.6. Советы по технике безопасности при эксплуатации станков с ЧПУ

Станок с ЧПУ – это сложный агрегат, состоящий из различных механических узлов, электрических и электронных компонентов. Для обеспечения безопасности внимательно ознакомьтесь с общими правилами эксплуатации станков с ЧПУ:

1. К эксплуатации и ремонту станка допускается только персонал, ознакомленный с соответствующей документацией станка и СЧПУ.
2. Всегда необходимо пользоваться защитными очками и специальной обувью. При необходимости используйте перчатки, пылезащитную маску и средства защиты слуха.
3. При работе на станке не рекомендуется носить кольца, часы, различные украшения и галстуки. Длинные волосы должны быть собраны и прикрыты рабочим головным убором.
4. Эксплуатация станка разрешается лишь в том случае, если вы убедились, что регулярно производилось обслуживание станка и что он находится в отличном техническом состоянии.

5. Необходимо убедиться, что станок заземлен надлежащим образом.
6. Не приступать к работе на станке, если от него исходит необычное или слишком сильное тепло, шум, наблюдается выделение дыма или вибрация. При этом необходимо срочно обратиться к производителям станка или в сервисную службу.
7. Доступ к электрическим компонентам станка разрешается только специально обученному персоналу.
8. Нельзя применять в качестве очистителей и СОЖ ядовитые и воспламеняющиеся вещества.
9. Не открывать защитные дверцы и кожуха, пока какое-либо из устройств станка находится в движении.
10. Зона вокруг станка должна быть сухой и хорошо освещенной.
11. Перед закреплением инструмента необходимо убедиться, что все поверхности устройства крепления инструмента чистые.
12. Не применять в станке изношенные, хрупкие и дефектные инструменты.
13. Не превышать номинальную мощность станка.
14. Не оставлять инструмент и детали в местах, в которых они могли бы соприкоснуться с подвижными частями станка.
15. Регулярно проверять уровень смазки и охлаждающего средства.
16. Во время обработки не предпринимать наладку инструмента или крепление деталей.
17. Поддерживать достаточно безопасное расстояние до всех точек «защемления» (мест изгиба шлангов и проводов) и избегать опасных ситуаций.
18. Обязательно знать расположение клавиш аварийного останова станка.
19. Не оставлять станок без присмотра во время его работы.
20. При контакте с обработанными деталями обращать внимание на наличие острых кромок.
21. Не удалять стружку голыми руками. Использовать для этого крючок или другое приспособление, убедившись в полной остановке частей станка. Не производить очистку станка с помощью воздушного шланга.
22. Не пытаться остановить или затормозить перемещения исполнительных органов станка голыми руками или с помощью приспособлений.
23. Не применять для крепления инструмента и заготовки дефектные или грязные патроны, держатели и приспособления.
24. Запрещается предпринимать какие-либо конструктивные изменения станка без согласования с производителем станка или сервисной службой.
25. Не допускать, чтобы при работе со станком вас отвлекали другие люди.
26. Ни в коем случае не совершать на станке операций, которых вы не понимаете. В случае сомнений обязательно проконсультироваться со специалистом.
27. Никогда не работать внутри станка, если кто-либо работает со стойкой ЧПУ. Перейти в режим редактирования **Edit** для предотвращения случайного перемещения исполнительных органов станка.

28. Уделять особое внимание перемещениям на ускоренной подаче. Соблюдать безопасное расстояние над поверхностью заготовки для таких перемещений.
29. Перед началом обработки проверять УП всеми доступными способами.

# Краткое изложение главы

- Прямоугольная система координат позволяет описать положение любой точки в пространстве и используется для расчета УП на станках с ЧПУ.
- Детали, обрабатываемые на станке с ЧПУ, можно рассматривать как набор простых геометрических элементов с известными координатами опорных (узловых) точек.
- Существуют два способа для записи (набора) управляющих программ: при помощи стойки ЧПУ станка и на ПК.
- Основными методами проверки УП на персональном компьютере являются бэкплот и твердотельная верификация.
- Передача УП с компьютера на станок, как правило, осуществляется в соответствии со стандартом RS-232 при помощи специального кабеля и коммуникационной программы.
- Большинство современных станков с ЧПУ имеют тестовые режимы для проверки правильности УП.
- При работе на станке с ЧПУ необходимо соблюдать технику безопасности.

# Вопросы

1. Как в прямоугольной системе координат определяется положение точки?
2. Какое программное обеспечение используется для набора кода УП на персональном компьютере?
3. Как проверяют правильность УП на компьютере?
4. Для чего предназначен режим DNC?
5. Какова последовательность полной проверки УП?
6. Для чего предназначен экранный режим СЧПУ **Distance to go**?
7. Каково поведение станка при работе в режиме **Single block**?
8. Попытайтесь перечислить наиболее важные пункты основных правил техники безопасности при эксплуатации станков с ЧПУ.

# Глава 4
# СТАНОЧНАЯ СИСТЕМА КООРДИНАТ

## 4.1. Нулевая точка станка и направления перемещений

Система координат станка с ЧПУ является главной расчетной системой, определяющей перемещения исполнительных органов. Оси координат располагают параллельно направляющим станка, что позволяет при создании УП легко задавать направления и расстояния перемещений.

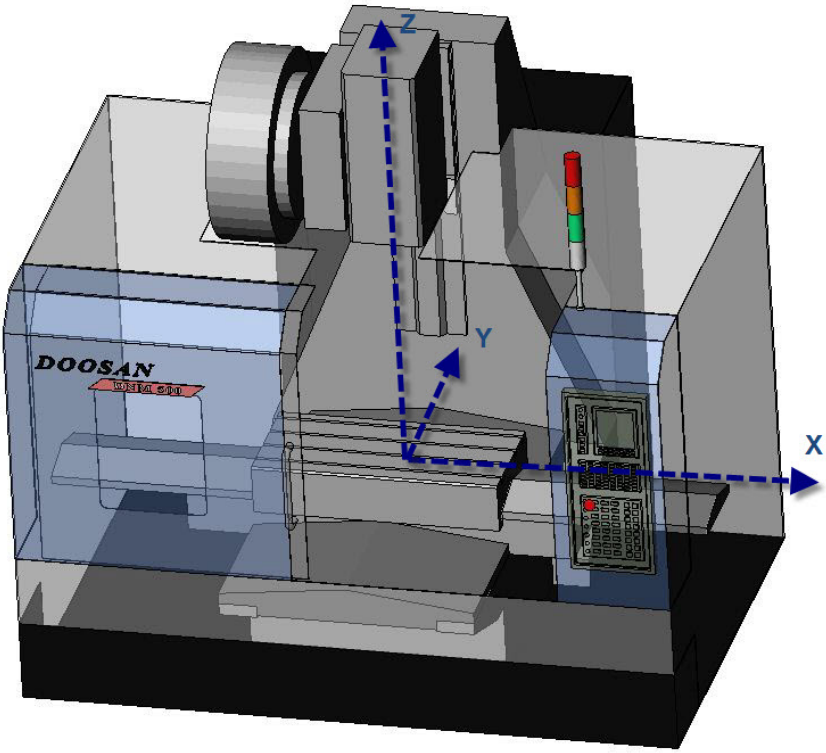



*Рис. 4.1. Оси координатной системы расположены параллельно направляющим*

**Правая система координат** является стандартной для всех станков с ЧПУ. В этой системе положительные направления координатных осей определяются по правилу «правой руки». Если большой палец указывает положительное направление оси X, указательный – оси Y, то средний укажет на положительное направление оси Z. В качестве положительного направления оси Z принимают вертикальное направление вывода инструмента (например, сверла) из заготовки. **То есть ось Z всегда связана со шпинделем станка.** Как правило, за X принимают ось, вдоль которой возможно наибольшее перемещение исполнительного органа станка. При этом ось X перпендикулярна оси Z и параллельна плоскости рабочего стола. Если вы определили на станке направления осей X и Z, то по правилу «правой руки» вы однозначно сможете сказать, куда «смотрит» ось Y. Оси X, Y, Z указывают положительные направления перемещений инструмента относительно неподвижных частей станка.

**При создании УП программист всегда исходит из правила, что именно инструмент перемещается относительно неподвижной заготовки.**

Дело в том, что одни станки с ЧПУ действительно перемещают колонну, шпиндель и, соответственно, вращающийся инструмент относительно неподвижной заготовки, а другие станки, наоборот, перемещают рабочий стол с заготовкой относительно вращающегося инструмента. Получаем противоположные направления перемещений. Если бы не было этого правила, то программист вынужден был бы думать: а что, собственно, перемещается и в какую сторону. А так все просто – система ЧПУ сама определит, в каком направлении нужно переместить тот или иной узел станка.

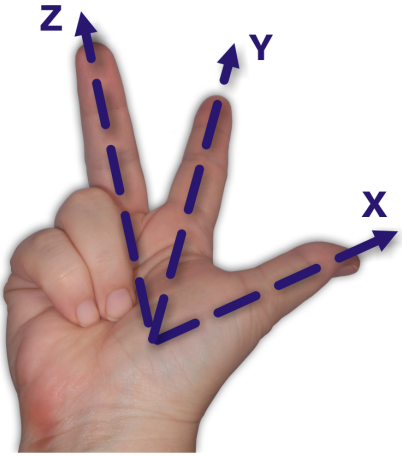



*Рис. 4.2. Воспользуйтесь правилом «правой руки» для определения положительных направлений осей координатной системы станка*

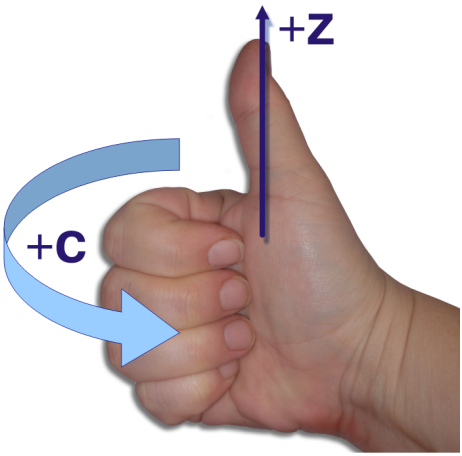



*Рис. 4.3. Если расположить большой палец правой руки в положительном направлении оси, то остальные согнутые пальцы обозначат положительное направление вращения вокруг этой оси*

Кроме линейных перемещений, конструкция некоторых станков позволяет совершать круговые перемещения. Под круговым перемещением подразумевается, например, поворот оси шпинделя фрезерного станка. Однако само рабочее вращение шпинделя не входит в это понятие. Круговые перемещения инструмента обозначают латинскими буквами A (вокруг оси X), B (вокруг оси Y) и C (вокруг оси Z). Положительные направления вращений вокруг этих осей определяются очень просто. Если расположить большой палец по направлению оси, то другие согнутые пальцы покажут положительное направление вращения.

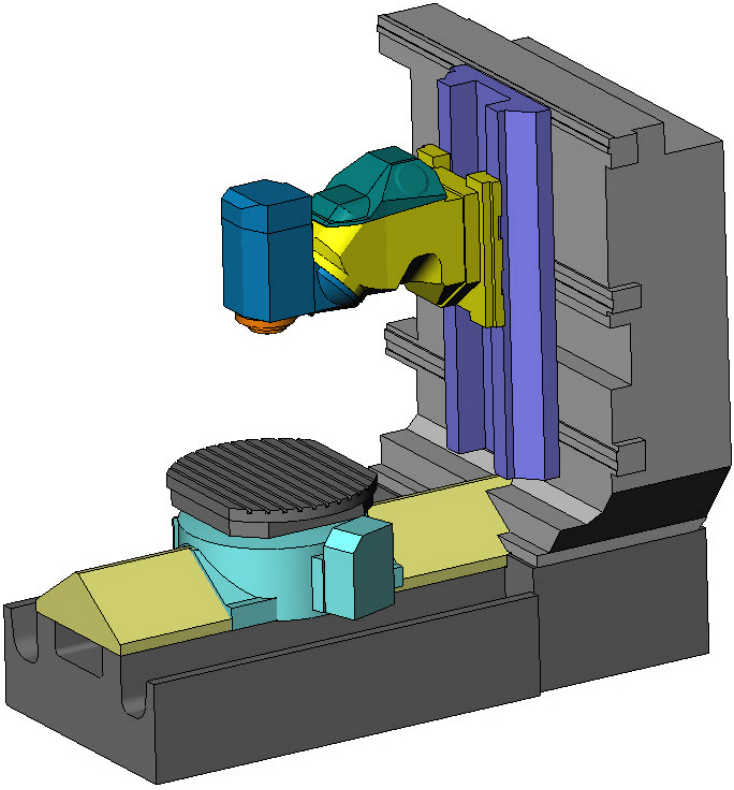


*Рис. 4.4. Головка этого станка способна поворачиваться вокруг своей оси*

**Положения исполнительных органов характеризуют их базовые точки, которые выбираются с учетом конструкции станка.** Например, базовой точкой для шпинделя фрезерного станка с ЧПУ является точка пересечения его торца с собственной осью вращения. Для рабочего стола – точка пересечения его диагоналей или один из углов. **Положение базовой точки относительно начала координат станка с ЧПУ (нулевой точки станка) называется позицией исполнительного органа в системе координат станка или машинной позицией** (от англ. *machine* –

станок). При работе станка в любой момент времени вы можете увидеть на экране стойки ЧПУ текущую машинную позицию (например, рабочего стола) по любой из осей относительно «нуля станка». В документации станка пределы возможных перемещений рабочих органов, как правило, указывают пределами смещений базовых точек. Эти данные являются очень важной характеристикой станка, так как они определяют максимально возможные габариты обрабатываемой заготовки.

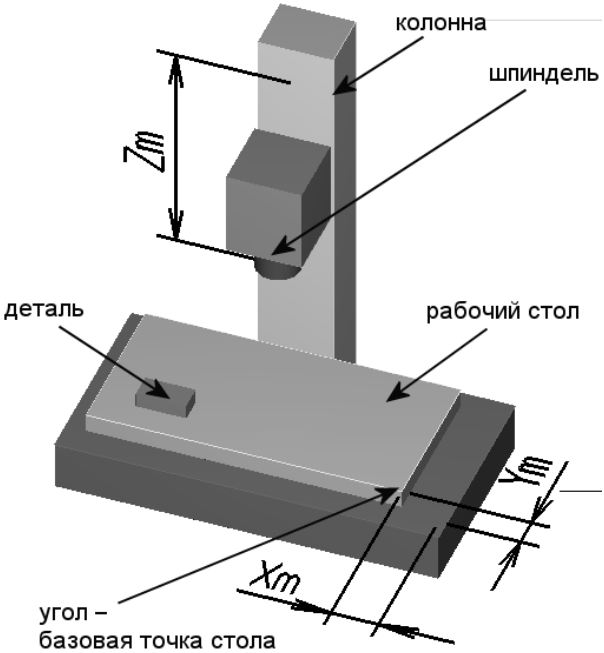



*Рис. 4.5. Расстояния Xm, Ym и Zm от нулевой точки станка до базовых точек исполнительных органов определяют машинные позиции*

**Нулевая точка станка – это физическая позиция, установленная производителем станка при помощи концевых выключателей или датчиков.** После включения станка необходимо переместить исполнительные органы в его нулевую точку, для того чтобы СЧПУ смогла определить или «обнулить» их машинную позицию, или, другими словами, нужно синхронизировать СЧПУ и станок. Дело в том, что в момент включения станка СЧПУ еще не знает реального положения исполнительных органов, и если не выполнить возврата в нуль, то станок просто «откажется» работать. Когда исполнительный орган приходит в нулевую точку станка, то происходит замыкание контактов специального датчика или конечного выключателя, СЧПУ получает электрический сигнал и машинная позиция обнуляется. **Процедура возврата в нуль станка является стандартной, и для ее осуществления любой станок имеет специальный режим и соответствующие клавиши на панели УЧПУ.**

# 4.2. Нулевая точка программы и рабочая система координат

Для того чтобы обработать заготовку на станке, необходимо соответствующим образом установить систему координат. Электронная природа СЧПУ позволяет нам легко помещать систему координат в любую позицию станка, просто нажимая определенные кнопки на панели УЧПУ. При написании УП программист «не забивает» себе голову вопросом о том, в каком именно месте рабочего стола будет установлена заготовка. Он знает, что перед выполнением обработки оператор станка «привяжет» к закрепленной на столе заготовке систему координат, в которой создана программа. Другими словами – **установит рабочую систему координат** или нулевую точку программы (детали).

Обычно рабочую систему координат по осям X и Y устанавливают в один из углов или центр детали, а за нуль по оси Z принимают самую верхнюю по-

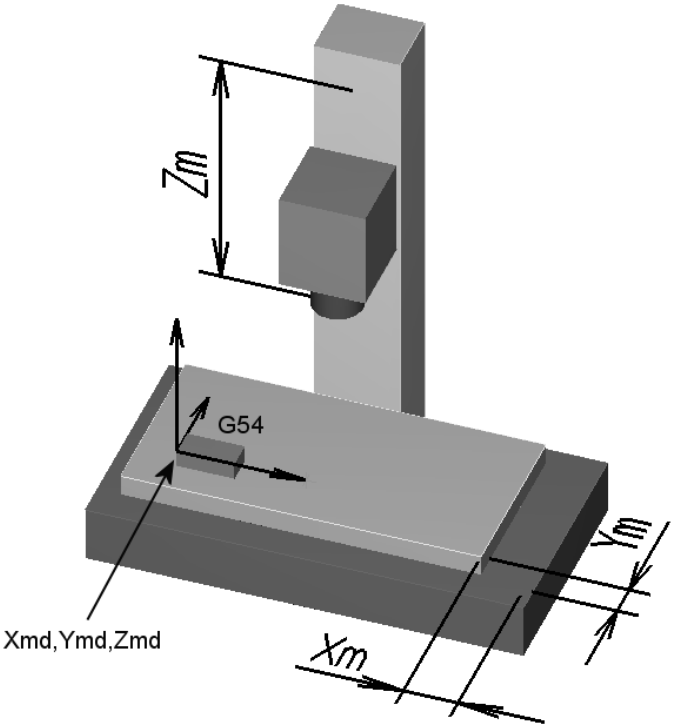



*Рис. 4.6. Задача оператора – определить координаты Xmd, Ymd, Zmd одного из углов детали в системе координат станка и занести их в регистры рабочих смещений, например в G54. После этого система ЧПУ принимает этот угол за нулевую точку для расчета всех перемещений по программе. В отличие от физического нуля станка, нулевая точка программы (детали) является логической*

верхность детали. Это облегчает программисту выполнение расчетов, а оператору проще «привязываться» и контролировать во время работы перемещения инструмента.

Предположим, что нужно обработать некоторую деталь. Программист решил, что нулевой точкой программы по осям X и Y будет нижний левый угол детали, а по оси Z – верхняя плоскость детали. После этого рассчитал координаты опорных точек, написал программу обработки и сообщил оператору станка о расположении нулевой точки программы. Оператор станка должен установить нулевую точку рабочей системы координат (нуль программы или детали) в определенный программистом угол детали. Для этого нужно найти координаты этого угла в системе координат станка (машинные координаты) и «объяснить» СЧПУ, что именно эта точка является исходной для расчетов всех перемещений. То есть можно сказать, что рабочая система координат находится внутри системы координат станка и зависит от нее.

Для нахождения машинной позиции элемента детали или «привязки» используются различные методы, о которых вы узнаете чуть позже.

После того как оператор определил все координаты (x, y, z) требуемого угла детали в системе координат станка, ему необходимо ввести значения этих координат в регистры рабочих смещений памяти СЧПУ. Под **рабочим смещением** понимается расстояние от нуля станка до нуля детали вдоль определенной оси. Оператор вводит эти координаты в память системы, используя цифровые клавиши стойки ЧПУ. Таким образом, обработка ведется в новой рабочей системе, координаты которой отличаются от соответствующих машинных координат на величины рабочих смещений. **Обратите внимание на то, что, в отличие от физического нуля станка, нулевая точка программы (детали) является логической.**

Современные СЧПУ позволяют запоминать множество смещений. Благодаря нескольким рабочим системам координат программист может использовать одну и ту же УП для обработки нескольких закрепленных на рабочем столе деталей. При этом нет необходимости выполнять программирование для каждой детали в отдельности. Вместо этого СЧПУ просто смещает рабочую координатную систему (нулевую точку программы) к следующей детали, подлежащей обработке.

Для установки различных рабочих систем координат используются соответствующие G-коды. В большинстве случаев G54 обозначает первую рабочую систему координат, G55 – вторую, G56 – третью и т. д. В программе обработки вы можете увидеть, например, такой кадр:

N20 G21 **G54** G90

Кадр N20 активирует первую рабочую систему координат G54.

**Назначение нулевой точки программы** – важный шаг при создании управляющей программы. Нулевая точка программы устанавливается для реализации требуемой последовательности и повторяемости обработки. Создание любой УП можно условно разбить на два этапа. На первом этапе технолог-программист анализирует информацию, полученную из конструкторской (чертежи, эскизы) и технологической документации (маршрутные карты, операционные карты), и, учитывая конструкционные и технические возможности станка с ЧПУ, окончательно

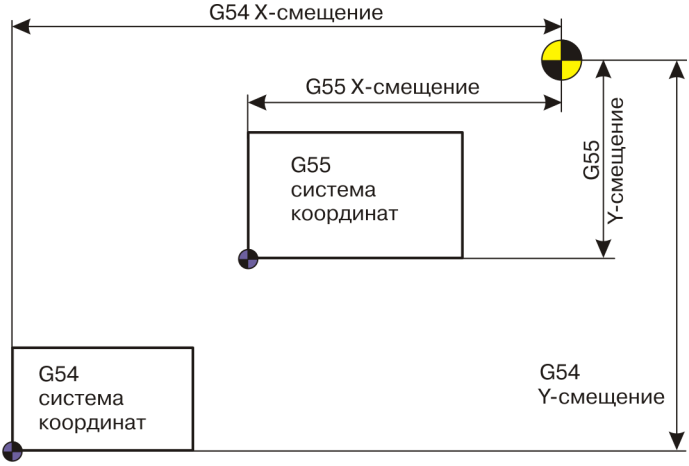



*Рис. 4.7. Многие станки с ЧПУ позволяют устанавливать несколько рабочих систем координат*

определяет технологические операции и маршрут обработки, назначает режущий и вспомогательный инструменты, выявляет комплекты конструкторских и технологических баз. И только на втором этапе производятся окончательный расчет траектории инструмента по опорным точкам и создание УП. Исходя из этого, при назначении нулевой точки программы используют несколько правил.

Первое, но не основное правило – 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.

Второе правило, более важное – 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 нуль детали. А если размеры указываются от центрального отверстия, то нулем детали следует назначить центр этого отверстия.

Если заготовка устанавливается в тиски, то вы должны учитывать несколько моментов. У тисков есть подвижная и неподвижная губки. Предположим, вы установили нулевую точку на поверхности (грани), примыкающей к подвижной губке тисков (рис. 4.8б). Размеры заготовок могут немного отличаться, и, соответственно, оператор станка для получения правильных размеров должен каждый раз «перепривязываться», то есть заново находить координаты нулевой точки. Если же нулевая точка установлена на поверхности, примыкающей к неподвижной губке тисков (рис. 4.8а), то координаты нулевой точки не изменятся при любых отклонениях размеров заготовки.

В большинстве случаев нулевая точка устанавливается относительно уже подготовленных поверхностей. Хорошо, когда на станок с ЧПУ приходит заготовка с обработанным «в размер» наружным контуром. Это позволяет точно и надежно ее закрепить и гарантировать постоянство координат нулевой точки.

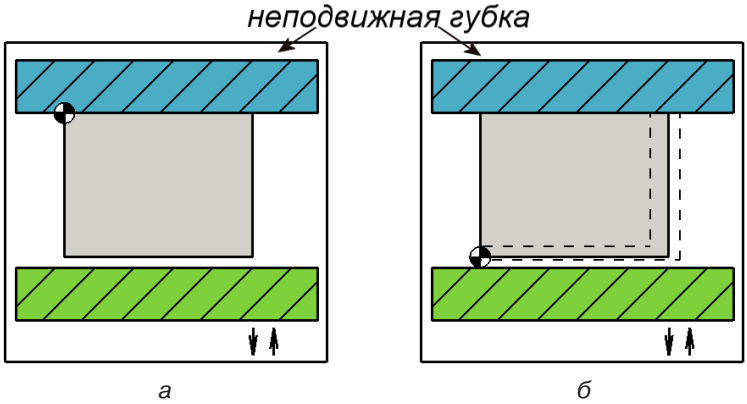



*Рис. 4.8. Варианты расположения нулевой точки в тисках*

Очень часто для обработки детали требуется несколько установов. Например, сначала нужно фрезеровать паз с одной стороны детали, а затем, после переустановки, – с другой стороны. В этом случае необходимо убедиться, что базирование осуществляется по одной и той же поверхности, иначе существует вероятность, что вы «промахнетесь» и не обеспечите требуемых размеров и точности.

Нулевая точка выбирается и устанавливается относительно определенного конструктивного элемента детали. Как правило, этим элементом является один из углов детали (заготовки) или центр отверстия. Будьте внимательны при обработке детали с несколькими установами и всегда «отслеживайте» расположение базовых поверхностей и нулевой точки.

## 4.3. Компенсация длины инструмента

При выполнении УП базовая позиция шпинделя (точка пересечения торца и оси вращения) определяется запрограммированными координатами. Проблема заключается в том, что в базовой позиции шпинделя обработка резанием не осуществляется. Обработка производится кромкой режущего инструмента, которая находится на некотором расстоянии от базовой точки шпинделя. Для того чтобы в запрограммированную координату приходила именно режущая кромка, а не шпиндель, необходимо «объяснить» СЧПУ, на какую величину по оси Z нужно сместить эту базовую точку.

Перед началом обработки оператор должен измерить длину каждого из инструментов, использующихся в программе, и ввести числовые значения длин в соответствующие регистры компенсации длины инструмента (или в таблицу инструментов). **Смещение базовой точки шпинделя на величину длины инструмента называется компенсацией длины инструмента.**

Компенсация длины инструмента на большинстве современных станков активируется командой G43, а отменяется при помощи G49 или H00. При создании

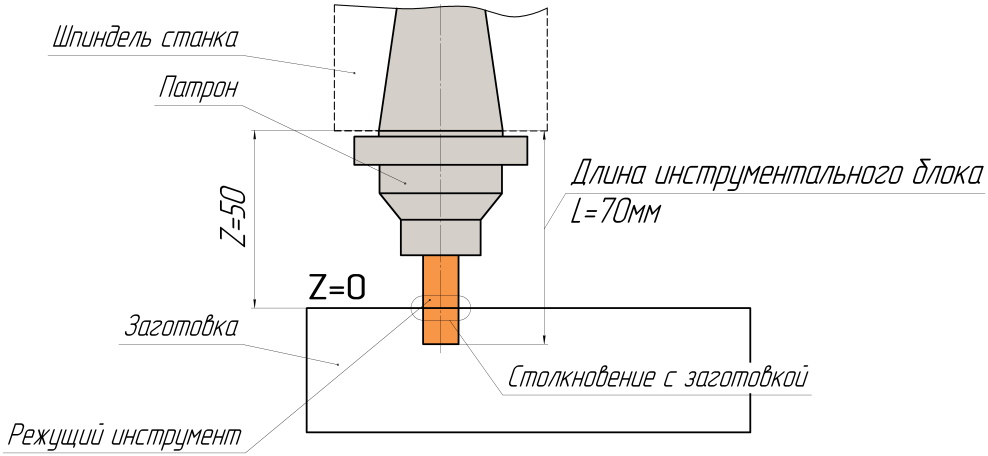



*Рис. 4.9. Если необходимо переместить инструмент длиной 70 мм в Z50, а компенсация длины инструмента не произведена, то произойдет опасное столкновение с деталью*

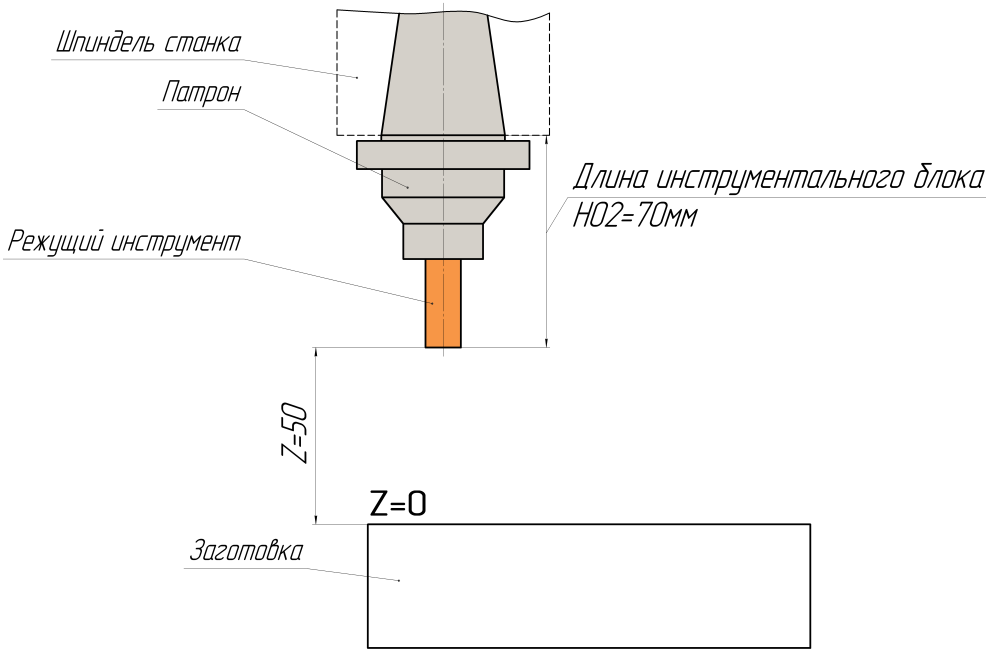



*Рис. 4.10. Если нужно переместить инструмент в Z50 и в программе есть код компенсации длины инструмента, то режущая кромка фрезы не дойдет до поверхности детали ровно 50 мм*

УП программист не указывает напрямую значение длины инструмента (он еще не знает точной длины), а использует «ссылку» на соответствующий регистр компенсации инструмента в памяти СЧПУ. Например, следующая строка программы активирует компенсацию длины инструмента № 2:

N025 G43 H02 Z50.

При этом H указывает на соответствующий регистр компенсации длины. H02 – на регистр инструмента № 2, H03 – на регистр инструмента № 3 и т. д. Вообще, компенсация длины инструмента сильно «упрощает жизнь» программисту – ему не нужно думать о точной длине инструментов – и дает возможность оператору станка, «играя» значением в регистре компенсации длины, добиваться требуемых размеров детали по оси Z.

# 4.4. Абсолютные и относительные координаты

Все приведенные ранее программные примеры и пояснения основывались на программировании в абсолютных координатах. При абсолютном способе программирования координаты точек отсчитываются от постоянного начала координат.

При относительном (инкрементальном) способе отсчета за нулевое положение каждый раз принимается положение исполнительного органа, которое он занимал перед началом перемещения к следующей опорной точке. Давайте вспомним самую первую УП для обработки паза из главы № 3. Взгляните на рис. 4.10, 4.11 и сравните значения координат в табл. 4.1.

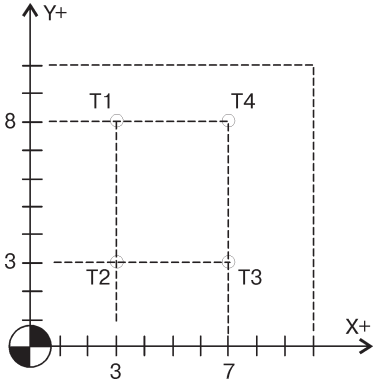



*Рис. 4.11. При абсолютном способе программирования (G90) координаты точек отсчитываются от одного «неподвижного» нуля*

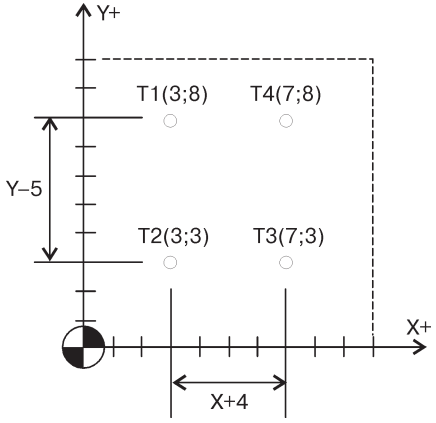



*Рис. 4.12. Относительный способ программирования (G91). Координаты точки T2 отсчитываются от точки T1, точки T3 – от T2, точки T4 – от T3*

***Таблица 4.1. Опорные точки паза при абсолютном и относительном программировании***

| Точка | Абсолютные координаты | | Относительные координаты | |
|---|---|---|---|---|
| | Координата по оси X | Координата по оси Y | Координата по оси X | Координата по оси Y |
| T1 | 3 | 8 | 3 | 8 |
| T2 | 3 | 3 | 0 | –5 |
| T3 | 7 | 3 | 4 | 0 |
| T4 | 7 | 8 | 0 | 5 |

Из этих примеров видно, как изменятся координаты опорных точек паза и программа обработки при относительном способе программирования.

**Абсолютное программирование**

```
%
O0001 (PAZ ABS)
N10 G21 G40 G49 G54 G80 G90
N20 M06 T01 (FREZA D1)
N30 G43 H01
N40 M03 S1000
N50 G00 X3 Y8
N60 G00 Z0.5
N70 G01 Z-1 F25
N80 G01 X3 Y3
N90 G01 X7 Y3
N100 G01 X7 Y8
N110 G01 Z0.5
N120 G91 G28 X0 Y0 Z0
N130 M05
N140 M30
%
```

**Относительное программирование**

```
%
O0001 (PAZ INCR)
N10 G21 G40 G49 G54 G80 G91
N20 M06 T01 (FREZA D1)
N30 G43 H01
N40 M03 S1000
N50 G00 X3 Y8
N60 G00 Z0.5
N70 G01 Z-1 F25
N80 G01 X0 Y-5
N90 G01 X4 Y0
N100 G01 X0 Y5
N110 G01 Z0.5
N120 G91 G28 X0 Y0 Z0
N130 M05
N140 M30
%
```

Программирование в относительных координатах было обязательным условием при работе на многих станках с ЧПУ старых моделей. Современные системы ЧПУ позволяют свободно работать и с абсолютными, и с относительными координатами. Для переключения из одного режима работы в другой используются команды G90 (абсолютное программирование) и G91 (относительное программирование). В случае появления одной координатной ошибки при относительном способе программирования все последующие перемещения будут неправильными.

# 4.5. Комментарии в УП и карта наладки

Если вы работаете постоянно, каждый день создаете хотя бы по одной программе для станка с ЧПУ, то через некоторое время обнаружите, что в вашем компьютере накопилось довольно много рабочих файлов. Практически невоз-

можно запомнить все технологические подробности и нюансы работы с той или иной программой. Оператор станка обязательно должен представлять, что делает конкретная УП, и обладать определенной информацией для настройки станка на работу. Например, где находится нулевая точка программы, какие режущие инструменты используются. В настоящее время эта информация сохраняется в основном двумя способами: с помощью комментариев в программе и карты наладки.

**Комментарии** представляют собой обычные предложения, при помощи которых программист доводит до оператора станка определенную технологическую информацию. Как правило, в комментариях содержатся следующие данные:

- дата и время создания УП;
- номер чертежа;
- материал заготовки;
- данные о рабочей системе координат;
- размеры инструмента;
- названия технологических операций.

Для того чтобы ввести комментарии в УП, необходимо использовать специальные символы (знаки) программирования. В качестве таких символов для большинства СЧПУ применяются круглые скобки или точка с запятой. Перед символами комментариев не принято ставить номера кадров, если комментарии занимают кадр полностью.

(TOOL #3)

или

; TOOL #3

У систем ЧПУ могут существовать различные ограничения на работу с комментариями. Например, некоторые стойки позволяют вводить комментарии длиной только до 32 символов.

Учтите, что существуют станки, имеющие полноразмерную клавиатуру, которая позволяет вводить комментарии прямо со стойки ЧПУ, и станки с ограниченной клавиатурой, СЧПУ которых не позволяет вводить и редактировать комментарии. Большинство СЧПУ работают с латинскими буквами, поэтому если вы будете писать комментарии на русском (при помощи ПК), то возможно, что на станке они будут нечитаемыми.

**Пример УП с комментариями**

```
%
O0045 (SKOBA)
(MATERIAL – ALUMINUM MM – 2024)
(DATE – NOV-14-04)
(TIME – 16:43)
(T2 | FREZA | H2 | D2 | D20.0000mm | | CONTOUR....)
(T4 | SVERLO | H4 | D4 | D10.0000mm | | PECK DRILL.)
N100 G00 G17 G21 G40 G49 G80 G90
N102 (FREZEROVANIE)
N104 T2
```

```
N106 M06 (FREZA 20)
N108 (MAX | Z100.)
N110 (MIN | Z-3.)
N112 G00 G90 G54 X-76.761 Y-42.321 S2000 M03
N114 G43 H2 Z100.
N116 Z10.
N118 G01 Z-3. F250.
N120 Y55.302
N122 X81.529
N124 Y-42.321
N126 X-76.761
N128 Z7.
N130 G00 Z100.
N136 M01
N138 (SVERLENIE)
N140 T4
N142 M06 (SVERLO 10)
N144 (MAX | Z100.)
N146 (MIN | Z-5.)
N148 G00 G90 G54 X-63.052 Y44.772 S1200 M03
N150 G43 H4 Z100.
N152 G98 G83 Z-5. R10. Q2. F45.
N154 X-40.798 Y53.25
N156 X8.213 Y47.421
N158 X52.19 Y49.806
N160 G80
N162 M05
N168 G90
N170 M30
%
```

При работе на станках с ЧПУ могут использоваться следующие виды документации: операционная карта, карта операционных эскизов, карта наладки станка, карта наладки инструмента, расчетно-технологическая карта. В настоящее время многие предприятия используют формальный подход к применению подобной стандартизованной документации, используют карту наладки произвольной формы. Исходя из этого, можно сказать, что карта наладки – документ, содержащий всю необходимую технологическую информацию для настройки станка на определенную работу, включая эскиз детали и приспособления, описание режущего инструмента, положения нулевой точки программы, режимов и времени обработки. Такая карта наладки помогает вспомнить о том, как нужно обрабатывать ту или иную деталь спустя некоторое время, когда программист и оператор станка с ЧПУ успели позабыть технологические особенности изготовления детали. Следует отметить, что комментарии в УП и карта наладки не заменяют друг друга, а используются совместно.

Карту наладки можно создать при помощи обычного листа бумаги и карандаша. Попробуйте схематично изобразить деталь, нулевую точку и крепежные элементы. Не забудьте записать порядок операций и данные режущего инструмента. Многие CAD/CAM-системы умеют автоматически генерировать карту наладки и инструментальную карту, в которой содержатся подробные сведения о применяемом в данной УП режущем инструменте.

# Краткое изложение главы

- Положительные направления координатных осей станка с ЧПУ определяются по правилу «правой руки».
- При написании программы обработки программист всегда исходит из правила, что именно инструмент перемещается относительно неподвижной заготовки.
- Положения исполнительных органов станка характеризуют их базовые точки. Например, базовой точкой для шпинделя фрезерного станка с ЧПУ является точка пересечения торца шпинделя с осью его вращения. Для рабочего стола – точка пересечения его диагоналей или один из углов.
- Положение базовой точки исполнительного органа относительно начала координат станка с ЧПУ (нулевой точки станка) называется позицией в системе координат станка, или машинной позицией.
- Сразу после включения станка необходимо переместить его исполнительные органы в нулевую точку для синхронизации с СЧПУ.
- Перед выполнением обработки оператор станка «привязывает» к закрепленной на столе заготовке систему координат, в которой создана программа.
- Смещение базовой точки шпинделя на величину длины инструмента называется компенсацией длины инструмента.
- При относительном способе отсчета за нулевое положение каждый раз принимается положение исполнительного органа, которое он занимал перед началом перемещения к следующей опорной точке.

# Вопросы

1. В чем заключается правило «правой руки» для определения направления осей координатной системы станка?
2. Какая точка является базовой для шпинделя?
3. Что необходимо сделать в первую очередь после включения станка?
4. Что такое рабочее смещение?
5. Какие коды используются для определения рабочей системы координат?
6. Для чего выполняется компенсация длины инструмента?
7. В чем разница между программированием в абсолютных и относительных координатах?
8. Для чего в УП используются комментарии?

# Глава 5
# СТРУКТУРА УПРАВЛЯЮЩЕЙ ПРОГРАММЫ

## 5.1. G- и M-коды

Программирование обработки на современных станках с ЧПУ осуществляется на языке, который обычно называют языком ИСО (ISO) 7 бит, или языком G- и M-кодов. **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.

Например, если программист хочет, чтобы инструмент перемещался по прямой линии, он использует G01. А если необходимо произвести смену инструмента, то в программе обработки он указывает M06.

Для управления многочисленными функциями станка с ЧПУ применяется довольно большое число различных кодов. Тем не менее, изучив набор основных G- и M-кодов, вы легко сможете создать управляющую программу.

В табл. 5.1 приведен список базовых кодов, которые мы подробно рассмотрим в этой и последующих главах. А в главе «Справочник G- и M-кодов» вы найдете подробное описание всех стандартных кодов и примеры их использования.

***Таблица 5.1. Базовые коды программирования обработки***

| Код (функция) | Назначение и пример кадра с кодом |
|---|---|
| | **Осевое перемещение** |
| G00 | Ускоренный ход – перемещение на очень высокой скорости в указанную точку<br>**G00 X10. Y20. Z25.** |
| G01 | Линейная интерполяция – перемещение по прямой линии на указанной скорости подачи<br>G01 X10. Y20. F100 |
| G02 | Круговая интерполяция – перемещение по дуге по часовой стрелке на указанной скорости подачи<br>G02 X10. Y20. R10. F100 |
| G03 | Круговая интерполяция – перемещение по дуге против часовой стрелки на указанной скорости подачи<br>G03 X10. Y20. R10. F100 |
| | **Настройка** |
| G20 | Ввод дюймовых данных<br>G20 G00 X10. Y20 |

***Таблица 5.1. Базовые коды программирования обработки (окончание)***

| Код (функция) | Назначение и пример кадра с кодом |
|---|---|
| G21 | Ввод метрических данных<br>G21 G00 X10. Y20 |
| G90 | Абсолютное позиционирование – все координаты отсчитываются от постоянной нулевой точки<br>G90 G00 X10. Y20 |
| G91 | Относительное позиционирование – все координаты отсчитываются от предыдущей позиции<br>G91 G00 X10. Y20 |
| **Обработка отверстий** | |
| G81 | Цикл сверления<br>G81 X10. Y20. Z-5. F30 |
| G82 | Цикл сверления с задержкой на дне отверстия<br>G82 X10. Y20. Z-5. R1. P2. F30 |
| G83 | Прерывистый цикл сверления<br>G83 X10. Y20. Z-5. Q0.25 R1. F30 |
| G85 | Цикл растачивания отверстия<br>G85 X10. Y20. Z-5. F30 |
| **Вспомогательные коды (функции)** | |
| M00 | Запрограммированный останов – выполнение программы временно прекращается |
| M01 | Запрограммированный останов по выбору – выполнение программы временно прекращается, если активирован режим останова по выбору |
| M03 | Прямое вращение шпинделя – шпиндель вращается по часовой стрелке |
| M04 | Обратное вращение шпинделя – шпиндель вращается против часовой стрелки |
| M05 | Останов шпинделя |
| M06 | Автоматическая смена инструмента<br>M06 T02 |
| M08 | Включение подачи охлаждающей жидкости |
| M09 | Выключение подачи охлаждающей жидкости |
| M30 | Конец программы, перевод курсора к началу программы |

# 5.2. Структура программы

Для знакомства со структурой УП давайте взглянем более пристально на уже созданную ранее программу обработки паза (см. главу 3):

```
%
O0001 (PAZ)
N10 G21 G40 G49 G54 G80 G90
N20 M06 T01 (FREZA D1)
N30 G43 H01
N40 M03 S1000
N50 G00 X3 Y8
```

```
N60 G00 Z0.5
N70 G01 Z-1 F25
N80 G01 X3 Y3
N90 G01 X7 Y3
N100 G01 X7 Y8
N110 G01 Z5
N120 M05
N130 M30
%
```

**Управляющая программа является упорядоченным набором команд, при помощи которых определяются перемещения исполнительных органов станка и различные вспомогательные функции.** Любая программа обработки состоит из некоторого количества строк, которые называются кадрами УП. **Кадр управляющей программы** – составная часть УП, вводимая и отрабатываемая как единое целое и содержащая не менее одной команды. Система ЧПУ считывает и выполняет программу кадр за кадром. Очень часто программист назначает каждому кадру свой номер, который расположен в начале кадра и обозначен буквой N. В нашей программе вы можете увидеть номера кадров с N10 до N130. Большинство станков с ЧПУ позволяют спокойно работать без номеров кадров, которые используются исключительно для удобства зрительного восприятия программы и поиска в ней требуемой информации. Поэтому наша программа обработки может выглядеть и следующим образом:

```
%
O0001 (PAZ)
G21 G40 G49 G54 G80 G90
M06 T01 (FREZA D1)
G43 H01
M03 S1000
G00 X3 Y8
G00 Z0.5
G01 Z-1 F25
G01 X3 Y3
G01 X7 Y3
G01 X7 Y8
G01 Z5
M05
M30
%
```

Программисту рекомендуется располагать номера кадров с интервалом в 5 или 10 номеров, чтобы при необходимости можно было вставить в программу дополнительные кадры.

В самом начале УП обязательно должен находиться код начала программы % и номер программы (например, O0001). Два этих первых кадра не влияют на

процесс обработки, тем не менее они необходимы для того, чтобы СЧПУ могла отделить в памяти одну программу от другой. Указание номеров для таких кадров не допускается.

```
%
O0001 (PAZ)
```

Кадр № 10 настраивает систему ЧПУ на определенный режим работы с последующими кадрами УП. Например, G21 означает, что станок будет работать в метрической системе, то есть перемещения исполнительных органов программируются и выполняются в миллиметрах, а не в дюймах. Иногда такие кадры называют строками безопасности, так как они позволяют перейти системе в некоторый стандартный режим работы или отменить ненужные функции.

```
N10 G21 G40 G49 G54 G80 G90
```

Следующие кадры говорят станку о необходимости подготовки к обработке. Для этого нужно поставить инструмент из магазина в шпиндель (кадр N20), активировать компенсацию длины инструмента (кадр N30) и заставить шпиндель вращаться в нужном направлении на указанной скорости (кадр N40). Также вы можете использовать символы комментариев. СЧПУ игнорирует любой текст, заключенный в круглые скобки, что позволяет, например, указать в кадре диаметр или наименование применяемой фрезы.

```
N20 M06 T01 (FREZA D1)
N30 G43 H01
N40 M03 S1000
```

Кадры с номерами от N50 до N110 непосредственно отвечают за обработку детали. В этой части УП содержатся коды, предназначенные для перемещения инструмента в указанные координаты. Например, кадр N80 перемещает инструмент в точку с координатами X3, Y3 со скоростью подачи, равной 25 миллиметров в минуту.

```
N50 G00 X3 Y8
N60 G00 Z0.5
N70 G01 Z-l F25
N80 G01 X3 Y3
N90 G01X7Y3
N100 G01 X7 Y8
N110 G01 Z0.5
```

Заключительные кадры предназначены для останова шпинделя (кадр N120) и завершения программы (кадр N130):

```
N120 M05
N130 M30
 %
```

Схематично любую УП можно представить в виде следующих областей:

| |
|---|
| Начало программы (шапка) |
| Вызов первого инструмента |
| Основная часть УП – рабочие перемещения |
| Смена инструмента |
| Основная часть УП – рабочие перемещения |
| Конец программы (шапка) |

# 5.3. Слово данных, адрес и число

Управляющая программа состоит из множества различных кадров. В свою очередь, каждый кадр УП состоит из слов данных. А слово данных строится из адреса (буквы) и относящегося к нему числа. Например, адрес Y относится к оси Y, а следующее за адресом число обозначает координаты вдоль этой оси.

G01 X3 Y3 – кадр УП
G – адрес
01 – число
G01 – слово данных (G-код)

X – адрес
3 – число
X3 – слово данных

Y – адрес
3 – число
Y3 – слово данных

Не обязательно, чтобы число, относящееся к G- или M-коду, имело ведущие нули (нули перед числом), например G01, G02, G03 и т. д. Можно писать просто G1, G2, G3. Тем не менее многие программисты в силу привычки предпочитают вариант с ведущими нулями.

Учтите, что система ЧПУ работает далеко не со всеми адресами. Например, стойка Fanuc откажется воспринимать латинские буквы E, U, V и W, если они не

относятся к макропрограмме. Это не значит, что вы не сможете загрузить в память станка программу, содержащую перечисленные адреса. Загрузить ее можно, но при попытке выполнить такую программу система ЧПУ, скорее всего, выдаст сообщение об ошибке.

# 5.4. Модальные и немодальные коды

Все станочные коды можно условно разделить на два класса в зависимости от их способности сохраняться в памяти СЧПУ. **Немодальные коды действуют только в том кадре, в котором они находятся. Модальные коды, напротив, могут действовать бесконечно долго, пока их не отменят другим кодом.**

Выделяют несколько групп кодов в зависимости от функции, которую они выполняют (табл. 5.2). Два модальных кода из одной группы не могут быть активными в одно и то же время. Например, G02 и G03 находятся в группе кодов осевых перемещений, и вы не можете применять оба этих кода сразу. Один из этих кодов обязательно отменит действие другого. Это как попытаться ехать на автомобиле одновременно и вправо, и влево. Однако вы можете одновременно использовать коды из разных функциональных групп. Например, в одном кадре можно написать G02 и G90.

***Таблица 5.2. Коды по группам***

| Функциональная группа | Коды |
|---|---|
| Перемещения | G00, G01, G02, G03 |
| Тип координатной системы | G90, G91 |
| Единицы ввода данных | G20, G21 |
| Постоянные циклы | G80, G81, G82, G83, G84, G85... |
| Рабочая система координат | G54, G55, G56, G57, G58... |
| Компенсация длины инструмента | G43, G44, G49 |
| Коррекция на радиус инструмента | G40, G41, G42 |
| Возврат в постоянных циклах | G98, G99 |
| Активная плоскость обработки | G17, G18, G19 |

Особенностью модальных кодов является то, что не нужно вводить активный код в последующие кадры. Например, код G01 используется для перемещения инструмента по прямой линии. Если нам необходимо совершить множество прямых перемещений, то не обязательно в каждом последующем кадре писать G01. Для отмены кода G01 следует применить один из кодов той же самой функциональной группы (G00, G02 или G03). Большинство из G-кодов являются модальными. Программист должен знать, к какой группе и к какому классу принадлежит тот или иной код.

Хотя M-коды обычно не делят на модальные и немодальные, однако этот термин все же можно применить и к ним. Например, можно выделить группу M-кодов, отвечающих за подачу охлаждающей жидкости (M07, M08, M09) или за вращение шпинделя (M03, M04, M05). Тем не менее большинство M-кодов нужно

рассматривать как немодальные. Некоторые стойки ЧПУ допускают программирование только одного М-кода в кадре.

## 5.5. Формат программы

Одна и та же управляющая программа может выглядеть по-разному. В качестве примера этого явления приведем фрагмент все той же программы обработки паза:

```
...
N70 G01 Z-1 F25
N80 G01 X3 Y3
N90 G01 X7 Y3
N100 G01 X7 Y8
N110 G01 Z0.5
...
```

Так как G01 является модальным кодом, то совсем не обязательно указывать этот код в каждом кадре линейных перемещений. Поэтому данный фрагмент УП может выглядеть следующим образом:

```
...
N70 G01 Z-1 F25
N80 X3 Y3
N90 X7 Y3
N100 X7 Y8
N110 Z0.5
...
```

СЧПУ читает первый кадр, в котором задается линейное перемещение по оси Z на глубину 1 мм (Z-1). Затем считывается второй кадр, в котором присутствуют координаты, но нет другого G-кода. Так как G01 является модальным кодом, то он сохраняется в памяти и используется СЧПУ для работы с новыми координатами. Можно сказать, что адреса X и Y также являются модальными. То есть значения координат сохраняются в памяти, пока СЧПУ не заменит (обновит) их другими значениями координат. Таким образом, данный фрагмент УП можно переписать еще раз:

```
...
N70 G01 Z-1 F25
N80 X3 Y3
N90 X7
N100 Y8
N110 Z0.5
...
```

Система ЧПУ читает программу обработки кадр за кадром. При этом в буфер памяти системы попадает один или несколько кадров целиком. Для современных

систем ЧПУ не принципиально, в каком месте кадра находится тот или иной код (слово данных). Однако некоторые станки, имеющие старые системы ЧПУ, могут быть очень придирчивы к порядку слов данных в кадре и к пробелам между ними. Для современной стойки три приведенных ниже кадра будут иметь совершенно одинаковый эффект:

N01 G55 G01 X30.45 Y2.35 M08
N02 M08 Y2.35 G55 X30.45 G01
N03 G01 X30.45 Y2.35 G55 M08

Для того чтобы программисту было легче создавать и читать УП, рекомендуется следующий порядок расположения слов данных и знаков программирования в кадре:

1) код пропуска кадра (/);
2) номер кадра (N);
3) подготовительные функции (G-коды);
4) адреса осевых перемещений (X, Y, Z, I, J, K, A, B, C);
5) команда подачи (F);
6) команда числа оборотов (S);
7) вспомогательные функции (M-коды).

После номера кадра N обычно следует G-код. Это как глагол в предложении – G-коды говорят нам, какую функцию несет каждый кадр. Далее следуют адреса и координаты позиций осевых перемещений. M-коды обычно ставятся в конец кадра. Это правило действует, когда в кадре присутствует G-код. Тем не менее если в кадре нет G-кода, то многие программисты предпочитают ставить M-код в начало:

...
N40 M03 S1000
N50 G00 X3 Y8
...

В УП не допускаются пробелы между адресом (буквой) и числом или внутри G- и M-кодов. В приведенном кадре есть несколько ошибок, и СЧПУ станка обязательно будет на них «ругаться»:

N50 G 00 X 3Y 8

Большинство современных стоек прекрасно работает и без пробелов между словами данных. Удаление пробелов позволяет сократить размер управляющей программы. Однако человеку, в отличие от компьютера, будет непривычно читать УП в таком варианте. Сравните два варианта одного и того же кадра:

1) N50 G00 XЗ Y8
2) N50G00X3Y8

Первый вариант явно читается легче, что означает меньшую вероятность ошибки при написании или проверке программы обработки.

Номера кадров для большинства современных СЧПУ не обязательны. Они используются для облегчения поиска требуемой информации в УП и для соз-

дания переходов к определенному кадру в некоторых особых случаях. Поэтому фрагмент программы, с которым мы работаем:

```
...
N70 G01 Z-1 F25
N80 X3 Y3
N90 X7
N100 Y8
N110 Z0.5
...
```

перепишем следующим образом:

```
...
G01 Z-1 F25
X3 Y3
X7
Y8
Z0.5
...
```

Необходимо уделить особое внимание **числовому формату**, с которым ваша стойка ЧПУ сможет работать. Обычно система ЧПУ работает с десятичным форматом и позволяет использовать несколько знаков до десятичной точки и несколько знаков после нее (например, 999.999). Возможны различные варианты употребления ведущих (перед десятичной точкой) и последующих (после десятичной точки) нулей. Сравните:

```
Z0.1
Z.1
Z0.100
```

В некоторых случаях наличие десятичной точки в определенных словах данных обязательно, а в других случаях недопустимо. Поэтому внимательно ознакомьтесь с разделом документации станка с ЧПУ, в котором говорится о формате программирования.

При работе с положительными числами не требуется вводить знак «+», так как СЧПУ исходит из положительного значения числа, если не введен никакой знак. Но при необходимости ввода отрицательного числового значения знак «–» должен быть запрограммирован обязательно.

Теперь мы можем сравнить первоначальный вариант программы обработки паза и новый вариант, созданный в этой главе (табл. 5.3). Несмотря на то что второй вариант УП имеет меньший размер (экономия программной памяти системы ЧПУ), его гораздо труднее читать.

Следовательно, при работе с «экономичной» версией УП появляется вероятность сделать ошибку или ее не заметить. Так как современные СЧПУ и компьютеры обладают достаточно большим объемом памяти, то нет смысла «экономить байты», убирая пробелы между словами данных, не ставя номеров кадров и забывая про комментарии.

***Таблица 5.3. Два варианта одной программы***

| Обычная УП | Сжатая УП |
|---|---|
| % | % |
| O0001 (PAZ) | O0001 (PAZ) |
| N10 G21 G40 G49 G54 G80 G90 | G21G40G49G54G80G90 |
| N20 M06 T01 (FREZA D1) | M6T1 |
| N30 G43 H01 | G43H1 |
| N40 M03 S1000 | M3S1000 |
| N50 G00 X3 Y8 | G0X3Y8 |
| N60 G00 Z0.5 | Z.5 |
| N70 G01 Z-1 F25 | G1Z-1F25 |
| N80 G01 X3 Y3 | X3Y3 |
| N90 G01 X7 Y3 | X7 |
| N100 G01 X7 Y8 | Y8 |
| N110 G01 Z0.5 | Z.5 |
| N120 G91 G28 X0 Y0 Z0 | G91G28X0Y0Z0 |
| N130 M05 | M5 |
| N140 M30 | M30 |
| % | % |

# 5.6. Строка безопасности

**Строкой безопасности** называется кадр, содержащий G-коды, которые переводят СЧПУ в определенный стандартный режим, отменяют ненужные функции и обеспечивают безопасную работу с управляющей программой. В нашей программе для обработки паза строкой безопасности является кадр N10.

N10 G21 G40 G49 G54 G80 G90

Как вы уже знаете, многие коды являются модальными и остаются активными в памяти СЧПУ до тех пор, пока их не отменят. Возможны ситуации, когда ненужный модальный G-код не был отменен. Например, если программа обработки была прервана по каким-либо причинам в середине. Строка безопасности, которая обычно находится в начале УП или после кадра смены инструмента, позволяет «восстановить» забытые G-коды и выйти в привычный режим работы.

Давайте поближе познакомимся с G-кодами, находящимися в типичной строке безопасности.

Код **G21** говорит станку о том, что все перемещения и подачи рассчитываются и осуществляются в миллиметрах, а не в дюймах (G20). Так как станки производятся и работают в разных странах, то существует возможность переключения между дюймовым и метрическим режимами. Поэтому включение этого кода в состав строки безопасности гарантирует работу в правильном режиме.

Код **G40** отменяет автоматическую коррекцию на радиус инструмента. Коррекция на радиус инструмента предназначена для автоматического смещения инструмента от запрограммированной траектории. Коррекция может быть активна, если вы в конце предыдущей программы забыли ее отменить (выключить). Результатом этого может стать неправильная траектория перемещения инструмента и, как следствие, испорченная деталь.

Код **G49** отменяет компенсацию длины инструмента.

Код **G54** на большинстве современных станков позволяет активировать одну из нескольких рабочих систем координат. Предыдущая управляющая программа могла работать в другой системе координат, например в G55. Как и большинство G-кодов, G-код рабочей системы координат является модальным и сохраняется активным в памяти СЧПУ до тех пор, пока его не отменят. Для того чтобы избежать ошибки, в строку безопасности включают код требуемой рабочей системы координат (G54–G59).

Код **G80** отменяет все постоянные циклы (например, циклы сверления) и их параметры. Отмена постоянных циклов необходима, так как все координаты после G-кода постоянного цикла относятся непосредственно к нему и для выполнения других операций нужно «сказать» системе ЧПУ, что цикл закончен.

Код **G90** активирует работу с абсолютными координатами. Хотя большинство программ обработки создаются в абсолютных координатах, возможны случаи, когда требуется выполнять перемещения инструмента в относительных координатах (G91).

# 5.7. Важность форматирования УП

В начале этой главы мы рассмотрели пример, наглядно демонстрирующий, что нет особого смысла уменьшать размер УП, а есть смысл сделать так, чтобы программа обработки была хорошо читаемой. Хорошая читаемость программы обеспечивается четкой структурой, комментариями, номерами кадров и пробелами между словами данных, то есть ее форматом. Однако это не единственная причина для форматирования УП. Второй причиной является совместимость. Если все программисты в вашей компании будут использовать одинаковый формат, то каждый из них без особых хлопот разберется в программе своего коллеги, сможет найти ошибку и исправить ее.

Пожалуй, самой важной причиной для форматирования УП является специфика многоинструментальной обработки на современных станках с ЧПУ. Особенность этой работы заключается в частой смене инструмента и в многократном использовании одного и того же инструмента. У оператора станка с ЧПУ может возникнуть необходимость перезапуска программы с определенного номера инструмента или операции. Для этого требуется особая технология написания УП, нужна определенная избыточность информации.

Опытный программист всегда включает в УП некоторый набор дополнительных команд, позволяющих оператору станка «стартовать» из определенных кадров программы. Этими командами могут быть не только команды включения требуемых оборотов шпинделя S и M03, но и строки безопасности, команды на выполнение компенсации длины и коррекции на радиус инструмента.

Это означает, что одна управляющая программа может состоять из множества «мини-программ»:

```
%
O0002
```

```
(PROGRAM NAME – T)
(DATE=DD-MM-YY – 15-09-04 TIME=HH:MM – 22:55)
N100 G21
N102 G00 G17 G40 G49 G80 G90
(1 OPERATION)
N104 T1 M6
N106 G00 G90 G54 X-88.783 Y47.985 A0. S1000M3
N108 G43 H1 Z100.
...
(2 OPERATION)
N134 T2 M6
N136 G00 G90 G54 X-88.783 Y47.985 A0. S1000M3
N138 G43 H2 Z100.
...
(3 OPERATION)
N164 T3 M6
N166 G00 G90 G54 X-88.783 Y47.985 A0. S1000M3
N168 G43 H3 Z100.
...
(4 OPERATION)
N194 T4 M6
N196 G00 G90 G54 X-88.783 Y47.985 A0. S1000M3
N198 G43 H4 Z100.
..
(5 OPERATION)
N224 T5 M6
N226 G00 G90 G54 X-88.783 Y47.985 A0. S2000M3
N228 G43 H5 Z100.
...
N248 M08
N250 G28 X0 Y0 Z0
N252 M30
%
```

# Краткое изложение главы

- Коды (функции) с адресом G, называемые подготовительными, определяют настройку СЧПУ на определенный вид работы.
- Коды (функции) с адресом M называются вспомогательными и предназначены для управления режимами работы станка.
- Любая программа обработки состоит из некоторого количества строк, которые называются кадрами УП.
- В самом начале УП обязательно должны находиться код начала программы % и номер программы (например, O0001).

- Немодальные коды действуют только в том кадре, в котором они находятся. А модальные коды активны до тех пор, пока их не отменят другим кодом из их функциональной группы.
- Строкой безопасности называется кадр, содержащий G-коды, которые переводят СЧПУ в определенный стандартный режим, отменяют ненужные функции и обеспечивают безопасную работу с управляющей программой.

# Вопросы

1. Что такое кадр УП?
2. Для чего нужны номера кадров?
3. Для чего в начале программы находятся код начала программы и номер программы?
4. Из чего состоит слово данных?
5. Перечислите функциональные группы кодов.
6. В чем преимущество модальных G-кодов перед немодальными?
7. Для чего нужны строки безопасности?
8. Назовите причины для форматирования УП.

# Глава 6
# БАЗОВЫЕ G-КОДЫ

## Введение

Современные системы ЧПУ понимают более сотни различных команд, однако в своей повседневной работе технолог-программист использует лишь ограниченный, довольно узкий набор G- и M-кодов. Этому есть простое объяснение. Дело в том, что основная задача УП заключается в перемещении инструмента по заданным координатам. Для реализации таких перемещений нужно воспользоваться всего несколькими кодами, которые можно смело назвать базовыми. В этой главе вы подробно ознакомитесь с базовыми G-кодами, которые понимают практически все станки с ЧПУ. Для большей наглядности все программные примеры снабжены рисунками и схемами. Обратите внимание на графический символ, обозначающий нулевую точку.

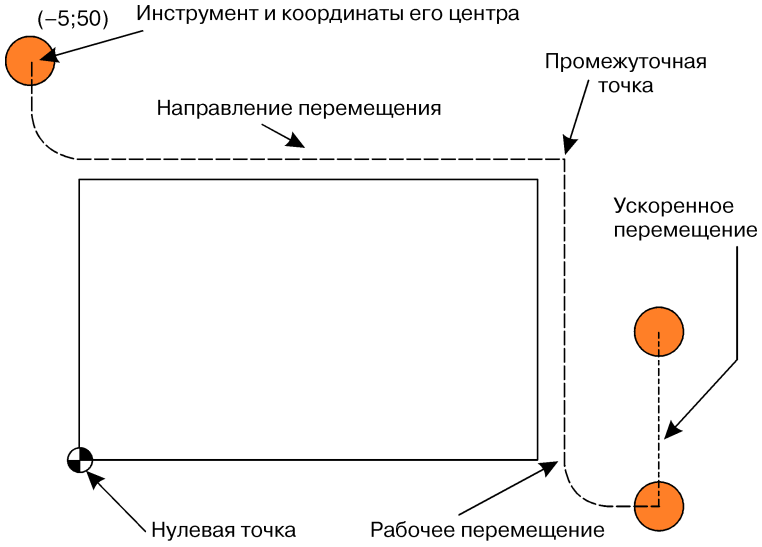



*Рис. 6.1. Условные обозначения на рисунках*

# 6.1. Ускоренное перемещение – G00

Код G00 используется для ускоренного перемещения. **Ускоренное перемещение, или позиционирование**, необходимо для быстрого перемещения инструмента к позиции обработки или безопасной позиции. **Ускоренное перемещение никогда не используется для выполнения обработки, так как скорость движения инструмента слишком высока и непостоянна.** Применение кода G00 позволяет существенно снизить общее время обработки.

На рис. 6.2 показана фреза, которая перемещается из некоторой безопасной позиции к заготовке для обработки паза. Вы уже знаете, что для фрезерования паза сначала нужно подвести фрезу максимально близко к поверхности заготовки, а затем осуществить вертикальное врезание в материал на нужную глубину. Для экономии времени на выполнение холостых ходов в программе обработке мы должны указать координаты точки у поверхности заготовки, в которую инструмент должен переместиться максимально быстро. Непосредственное врезание инструмента в материал заготовки осуществляется со скоростью рабочей подачи и при помощи другого кода.

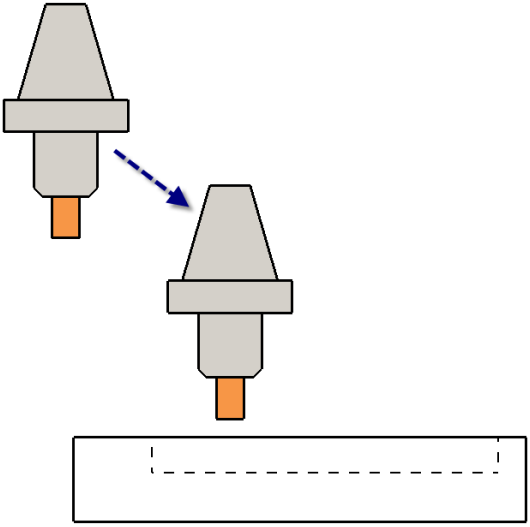


*Рис. 6.2. Ускоренное перемещение G00*

Для выполнения ускоренного перемещения достаточно указать в кадре код G00 и координаты требуемой позиции. Условно кадр ускоренного перемещения выглядит так:

G00 Xn.n Yn.n Zn.n

Посмотрим, как будет перемещаться инструмент из точки с координатами (0;0) в данном фрагменте УП:

```
...
N05 G00 X10.0 Y20.0
N10 X40.0
N15 Y0.0
...
```

Кадр N05 обеспечивает ускоренное перемещение инструмента в точку с координатами (10;20). Следующий кадр выполняет позиционирование в точку (40;20). Кадр N15 перемещает инструмент в точку (40;0). Так как G00 является модальным кодом, то нет никакой необходимости указывать его еще раз в кадрах N10 и N15.

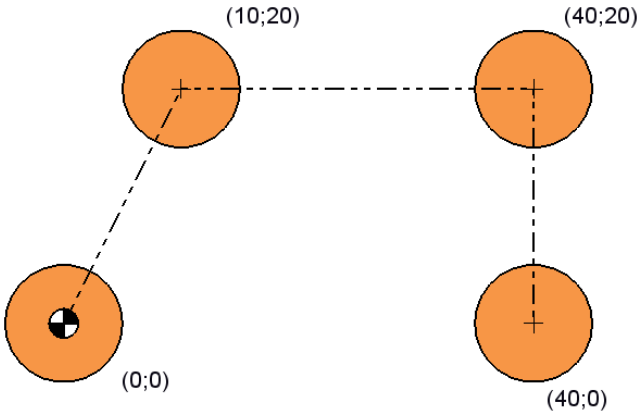



*Рис. 6.3. Ускоренные перемещения инструмента*

Всегда необходимо оставлять небольшое расстояние между поверхностью заготовки и точкой, в которую должен быть перемещен инструмент при помощи кода G00. Если это расстояние будет близким к нулю, то возникает опасность столкновения инструмента с заготовкой так как размеры заготовки или инструмента могут оказаться немного большими, чем вы ожидаете. Обычно безопасное расстояние находится в пределах от 0.5 до 5 мм.

При ускоренном подводе инструмента к детали по трем осям сначала лучше выполнить позиционирование по осям X и Y, а уже затем по Z. При отводе инструмента – обратный порядок перемещений. В любом случае, рекомендуется разделять ускоренное перемещение на два кадра – на позиционирование по X, Y в одном кадре и по оси Z в другом. Дело в том, что при позиционировании с кодом G00 по трем осям одновременно траектория движения инструмента может не являться прямой линией. СЧПУ отдает команду двигателям на перемещение колонны или рабочего стола на максимальной скорости в указанную координату. Как только достигается требуемая координата по одной из осей, то ускоренное перемещение по этой оси прекращается, хотя перемещение по другим осям все еще может происходить. Так как расстояния перемещения по трем осям различны, то траектория перемещения может выглядеть как ломаная линия. Результатом такого перемеще-

ния может стать неожиданное столкновение инструмента с крепежными элементами и приспособлениями.

Сравните два фрагмента УП:

```
...
N10 G00 X100.0 Y200.0 Z0.5
...
```

```
...
N10 G00 X100.0 Y200.0
N20 Z0.5
...
```

Во втором фрагменте ускоренное перемещение разбито на два кадра: сначала инструмент быстро перемещается в точку (100;200), а уже после опускается по оси Z. Этот вариант является более предпочтительным, так как основное позиционирование осуществляется по осям X и Y далеко от заготовки и приспособления, тогда как в первом случае существует вероятность столкновения инструмента с деталью.

При работе с ускоренными перемещениями нужно проявлять повышенную внимательность. Современные станки с ЧПУ в этом режиме могут иметь скорость 30 метров в минуту и более. Скорее всего, оператор станка просто не успеет среагировать на неправильное перемещение на такой скорости, и может произойти серьезное столкновение. Опытный программист старается не допускать ускоренных перемещений инструмента ниже поверхности заготовки и всегда проверяет кадры, содержащие код G00.

# 6.2. Линейная интерполяция – G01

Код G01 предназначен для выполнения **линейной интерполяции**, или, говоря простым языком, **для перемещения инструмента по прямой линии с заданной скоростью**. Условно кадр для линейной интерполяции записывается следующим образом:

G01 Xn.n Yn.n Zn.n F n.n

Как видите, в этом кадре появилось слово данных F. Основное отличие кода G01 от G00 заключается в том, что при линейной интерполяции инструмент перемещается с заданной скоростью (скоростью рабочей подачи), при которой возможна механическая обработка материала. При этом СЧПУ поддерживает прямолинейное перемещение даже по трем осям одновременно.

```
N10 G01 X10.0 Y30.0 F100
N20 X40.0 Y40.0 Z40.0
```

В кадре N10 инструмент перемещается в точку (10;30) со скоростью 100 миллиметров в минуту. Следующий кадр выполняет линейное перемещение в точку (40;40;40). Так как код G01 является модальным, то его не нужно указывать еще раз в кадре N20. То же самое относится и к скорости подачи F. Если в кадре N10 указана скорость F100, то она остается неизменной, пока не будет запрограммировано новое значение F. Линейная интерполяция используется не только для обработки в плоскости X–Y, но и для вертикального врезания в материал заготовки.

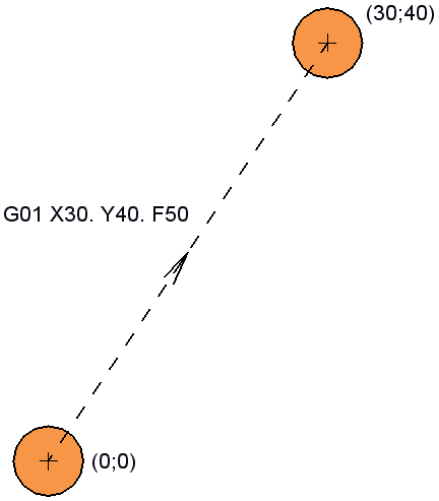



*Рис. 6.4. Перемещение инструмента в точку (30; 40) со скоростью рабочей подачи 50 мм/мин*

# 6.3. Круговая интерполяция – G02 и G03

Если обработку по прямой линии несложно производить и на простом станке с ручным управлением, то перемещение инструмента по дуге точнее и проще выполнять на станке с ЧПУ.

**Коды G02 и G03 предназначены для выполнения круговой интерполяции.** Код G02 используется для перемещения по дуге по часовой стрелке, а G03 – против часовой стрелки. Направление перемещения определяется, когда мы смотрим на инструмент со стороны шпинделя, в отрицательном направлении оси Z. Как и при выполнении линейной интерполяции, в кадре круговой интерполяции необходимо указать скорость рабочей подачи F.

Существуют два способа для формирования кадра круговой интерполяции. Сравните структуру следующих кадров:

G02 Xn.n Yn.n Zn.n In.n Jn.n Kn.n Fn.n. G02 Xn.n Yn.n Zn.n Rn.n Fn.n.

В первом варианте для выполнения кругового перемещения указывают: код G02 (G03); координаты конечной точки дуги; I, J, K – слова данных и скорость рабочей подачи. А во втором варианте вместо I, J, K указывают R. Выбор варианта записи кадра кругового перемещения зависит от возможностей ЧПУ и привычки программиста. Большинство современных станков с ЧПУ поддерживают оба варианта записи.

В кадре с кодом круговой интерполяции необходимо указать координаты конечной точки перемещения (дуги). Если, кроме X и Y, в кадре находится Z-слово данных, то это значит, что производится винтовая интерполяция. Винтовая ин-

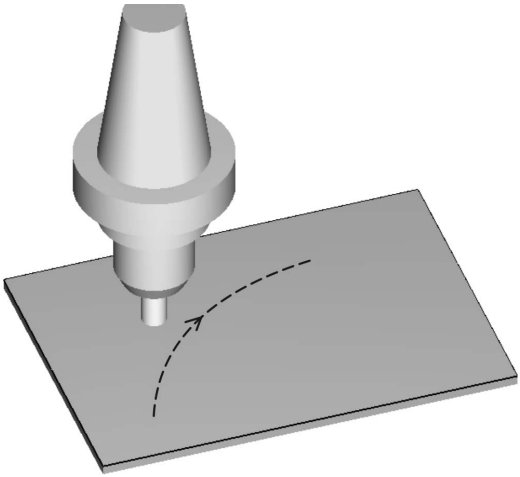


*Рис. 6.5. Направление перемещения по дуге можно определить, если посмотреть на заготовку со стороны инструмента. В данном случае фреза перемещается по часовой стрелке, значит, используем код G02*

терполяция, которая поддерживается не всеми системами ЧПУ, позволяет выполнять фрезерование резьбы и обеспечивает плавное винтовое врезание инструмента в материал заготовки.

## Дуга с I, J, K

Для полного описания дуги недостаточно задать только координаты ее конечной точки. Необходимо также указать радиус и координаты центра.

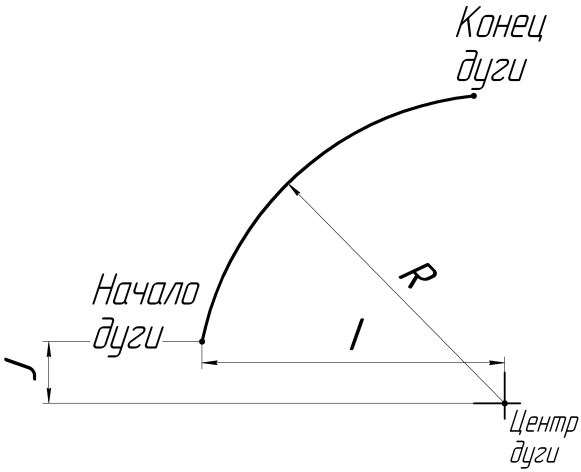



*Рис. 6.6. Адреса I, J, K используются для определения центра дуги*

При помощи I, J и K вы указываете относительные (инкрементальные) расстояния от начальной точки дуги до ее центра. Слово данных с I относится к оси X, слово данных с J – к оси Y, а слово данных с K – к оси Z. При этом в зависимости от расположения дуги значения могут быть положительными или отрицательными.

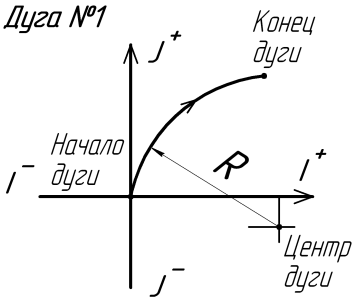



*Рис. 6.7. Для описания дуги № 1 необходимо указать положительное значение для I и отрицательное для J*

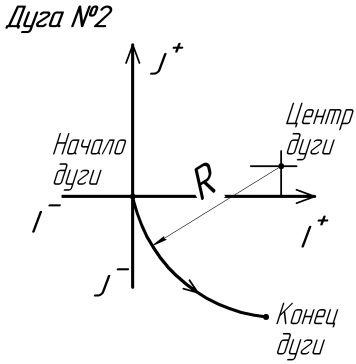



*Рис. 6.8. Для описания дуги № 2 необходимо указать положительное значение для I и положительное для J*

## Дуга с R

Более простой способ задания центра дуги основан на применении адреса R (радиуса). Если ваша стойка поддерживает такой формат для круговой интерполяции, то СЧПУ самостоятельно производит необходимые расчеты для определения координат центра дуги. Многие СЧПУ при работе с R требуют, чтобы окружность была разбита на несколько сегментов.

**Для однозначного определения формы дуги нужно указывать соответствующий знак перед числовым значением радиуса R.** Для дуги, которая больше 180°,

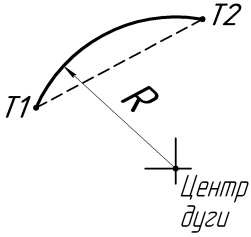



*Рис. 6.9. Так как дуга меньше 180° (ее центр расположен снаружи хорды), то R будет иметь положительное значение*

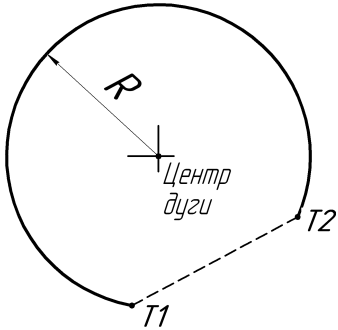



*Рис. 6.10. Так как дуга больше 180° (ее центр расположен внутри хорды), то R будет иметь отрицательное значение*

значение R будет отрицательным. Для дуги, которая меньше 180°, значение R будет положительным.

## Использование G02 и G03

Давайте разберемся, как работает круговая интерполяция, на примере. Приведенный ниже фрагмент управляющей программы перемещает инструмент по дуге с радиусом 3 мм из точки A (0;0) в точку B (3;3) со скоростью рабочей подачи 100 мм/мин.

N10 G02 X3.0 Y3.0 I3.0 J0.0 F100

Так как центр дуги находится на расстоянии 3 мм по оси X и 0 мм по оси Y относительно начальной точки A, то I будет равно 3.0, а J равно 0. Полученная дуга составляет всего четверть от полной окружности. Попытаемся описать всю окружность постепенно. Следующий кадр перемещает инструмент из точки B (B1) в точку B2. Так как скорость рабочей подачи не изменяется, то нет необходимости повторно указывать F-слово данных.

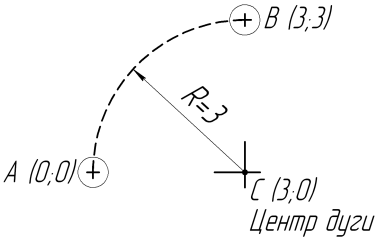



*Рис. 6.11. Перемещение по дуге с R3 из точки A (0;0) в точку B (3;3)*

Так как центр дуги находится на расстоянии 3 мм по оси X и 0 мм по оси Y относительно начальной точки A, то I будет равно 3.0, а J равно 0. Полученная дуга составляет всего четверть от полной окружности. Попытаемся описать всю окружность постепенно. Следующий кадр перемещает инструмент из точки B1 в точку B2. Так как скорость рабочей подачи не изменяется, то нет необходимости повторно указывать F-слово данных.

Так как центр дуги находится на расстоянии 0 мм по оси X и 3 мм по оси Y относительно точки B, то I будет равно 0, а J равно –3.

Таким образом, нам удалось создать перемещение по дуге из точки A в точку B2 при помощи двух кадров. Этот пример не случаен. Дело в том, что многие станки требуют именно такого разбиения окружности. То есть для описания полной окружности может потребоваться до четырех кадров.

В настоящее время большинство систем ЧПУ позволяют выполнить операцию по описанию полной окружности за два или даже за один кадр. Поэтому перемещение из точки A в точку C можно записать следующим образом:

N05 G02 X6.0 Y0.0 I3.0 J0.0

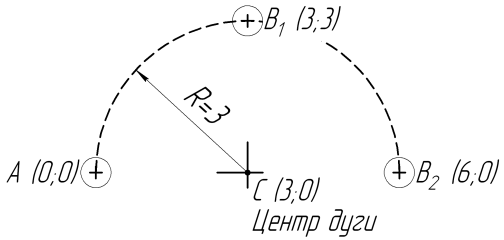



*Рис. 6.12. Современные системы ЧПУ допускают описание подобной дуги в одном кадре*

А для полной окружности с радиусом 3 мм и центром в точке с координатами (0; 0) справедливым будет следующий кадр:

N15 G02 X-3.0 Y0.0 13.0 J0.0

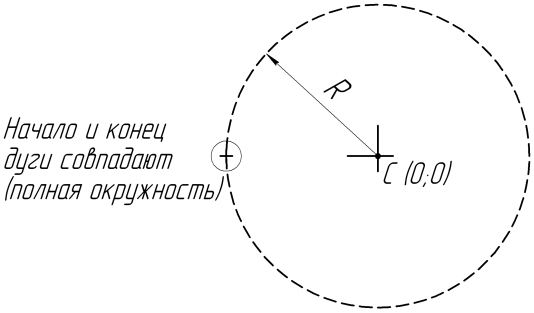



*Рис. 6.13. Описание полной окружности в одном кадре также возможно*

Дуги такого типа несложно описать математически. Однако если начальная и конечная точки дуги образуют некоторый сложный угол или эти точки находятся в разных квадрантах, то для нахождения значений I, J, K требуются определенные тригонометрические вычисления (рис. 6.14). При этом необходимо, чтобы расче-

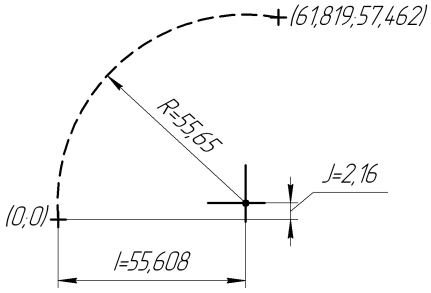



*Рис. 6.14. Часто для расчета дуги «вручную» необходимо приложить некоторые усилия*

ты были достаточно точными, иначе СЧПУ может выдать сообщение о невозможности построения дуги.

На рис. 6.15 изображена дуга, которую необходимо описать при помощи кодов круговой интерполяции с R-словом данных. В случае, когда инструмент перемещается по дуге по часовой стрелке (G02) из точки А в точку В , в УП должен присутствовать следующий кадр: G02 X0 Y-10 R10. Если инструмент перемещается по дуге против часовой стрелки (G03) из точки В в точку А, в УП должен присутствовать следующий кадр: G03 X10 Y0 R10.

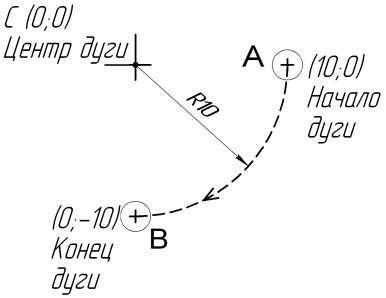



*Рис. 6.15. Дуга, которую необходимо описать при помощи кодов круговой интерполяции с R-словом данных*

# Краткое изложение главы

- Код G00 используется для выполнения ускоренного перемещения.
- Ускоренное перемещение никогда не применяется для выполнения обработки, так как скорость движения инструмента слишком высока и непостоянна.
- Код G01 предназначен для выполнения линейной интерполяции.
- Коды G02 и G03 предназначены для выполнения круговой интерполяции.
- Современные системы ЧПУ допускают описание полной дуги в одном кадре.
- Если дуга меньше 180°, то R-слово данных будет положительным. Если дуга больше 180°, то R будет отрицательным.

# Вопросы

1. Для чего применяется ускоренное перемещение?
2. Зачем нужен зазор между поверхностью и точкой, в которую перемещается инструмент с помощью кода G00?
3. Почему при работе с G00 следует проявлять повышенную осторожность?
4. В чем разница между G01 и G00?
5. В чем разница между G02 и G03?
6. Для чего в кадре круговой интерполяции указывают I-, J-, K-слова данных?
7. Как проще описать дугу – при помощи R- или I-, J-, K-слов данных?

# Глава 7
# БАЗОВЫЕ М-КОДЫ

## Введение

Коды, обозначающиеся буквой М (Miscellaneous), называются **вспомогательными** и предназначены для управления режимами работы станка. М-код может стоять как отдельно, так и находиться в кадре с G-кодами. Некоторые М-коды работают совместно с другими адресами. Например, М-код, отвечающий за направление вращения шпинделя, обычно указывается с адресом S, который необходим для задания числа его оборотов при вращении:

N10 S1000 M03

В табл. 8 помещены базовые М-коды, которые должен знать каждый технолог-программист. Когда М-код находится в кадре с G-кодом, то порядок выполнения зависит от модели ЧПУ. Возьмем, например, следующий кадр:

N10 G01 X100.0 Y100.0 Z100.0 F50.0 M08

Этот кадр выполняет линейное перемещение и включает подачу охлаждающей жидкости (M08). Одни станки включают подачу СОЖ сразу, другие – только после перемещения в указанную позицию. Зная это, опытный программист старается указывать код M08 перед выполнением перемещения на рабочей подаче:

N05 M08
N10 G01 X100.0 Y100.0 Z100.0 F50.0

Нужно учесть, что некоторые системы ЧПУ позволяют задавать в кадре только один М-код. В этом случае если в кадре находятся несколько М-кодов, то СЧПУ выдаст сообщение об ошибке. Для избежания ошибок внимательно прочтите раздел документации станка и системы ЧПУ о работе с М-кодами.

***Таблица 7.1. Базовые М-коды***

| М код | Действие |
|---|---|
| M00 | Запрограммированный останов |
| M01 | Останов по выбору |
| M02 | Конец программы |
| M03 | Прямое вращение шпинделя |
| M04 | Обратное вращение шпинделя |
| M05 | Останов шпинделя |

***Таблица 7.1. Базовые М-коды (окончание)***

| М код | Действие |
|---|---|
| M06 | Автоматическая смена инструмента |
| M08 | Включение подачи охлаждающей жидкости |
| M09 | Выключение подачи охлаждающей жидкости |
| M30 | Конец программы, перевод курсора в начало программы |

# 7.1. Останов выполнения управляющей программы – M00 и M01

Очень часто возникают ситуации, когда необходимо временно прервать выполнение программы. Например, для того чтобы удалить стружку, проверить размеры обрабатываемой детали или переставить крепежные элементы в другое положение.

Коды M00 и M01 временно приостанавливают выполнение программы обработки, или, говоря другими словами, делают паузу в производственном цикле станка. Когда СЧПУ читает код M00, то происходит так называемый запрограммированный останов. Все осевые перемещения останавливаются и возобновляются лишь после того, как оператор станка нажмет клавишу **Старт цикла** на панели УЧПУ. При этом шпиндель продолжает вращаться, и другие функции остаются активными. Если оператор станка нажимает клавишу **Старт цикла**, то выполнение программы будет продолжено с кадра, следующего за M00.

...
N200 G01 X200
N210 G00 Z100
N220 **M00**
N230 G00 Z5
N240 G01 Z-1 F50
...

Для того чтобы безопасно удалить стружку из зоны обработки или снять ее со сверла, необходимо остановить шпиндель. То есть перед командой M00 нужно указать код выключения вращения шпинделя M05. Однако не забудьте снова включить шпиндель, иначе инструмент будет перемещаться без вращения, что приведет к его поломке.

...
N200 G01 X200
N210 G00 Z100
N215 **M05**
N220 **M00**
N225 **M03 S1000**

N230 G00 Z5
N240 G01 Z-1 F50
...

Код M01 предназначен для останова по выбору. Действует он аналогично коду M00, однако предоставляет выбор оператору – нужно или не нужно прерывать выполнение управляющей программы. На панели УЧПУ практически любого станка имеется клавиша (или переключатель) «M01». Если эта клавиша нажата, то при чтении кадра с M01 происходит останов. Если же клавиша не нажата, то команда M01 пропускается и выполнение УП не прерывается.

...
N200 G01 X200
N210 G00 Z100
N220 M01
N230 G00 Z5
N240 G01 Z-1 F50
...

***Таблица 7.2. Поведение станка при работе с кодом M01***

| Клавиша «M01» на панели УЧПУ | Поведение станка |
|---|---|
| Вкл. | Выполнение программы приостанавливается и будет продолжено только после того, как оператор станка нажмет клавишу **Старт цикла** |
| Выкл. | Выполнение программы не будет прервано |

# 7.2. Управление вращением шпинделя – M03, M04, M05

Вспомогательные коды M03 и M04 предназначены для управления вращением шпинделя. Единственная разница между двумя этими М-кодами заключается в направлении вращения. **Код M03 отвечает за прямое (по часовой стрелке), а M04 – за обратное вращение шпинделя (против часовой стрелки).** Направление вращения определяется, если смотреть в отрицательном направлении оси Z (со стороны шпинделя в сторону заготовки). При фрезеровании режущие инструменты должны иметь прямое вращение (M03). При выводе метчика из отверстия, при нарезании левой резьбы, в циклах автоматического измерения диаметра инструмента может потребоваться обратное вращение пшинделя (M04). В конце программы обработки и перед сменой инструмента нужно остановить вращение шпинделя при помощи команды M05.

...
N40 Z5.0
N45 G00 Z100.0
N50 M05

M55 M30
%

Для задания частоты вращения шпинделя используется S-адрес. За S следует числовое значение, выражающее скорость вращения шпинделя в оборотах за одну минуту. Большинство СЧПУ воспринимают только целочисленное значение S. Обычно код M03 и S находятся в одном кадре. Например, кадр N20 заставляет вращаться шпиндель в прямом направлении со скоростью 1000 оборотов в минуту:

```
%
O0002
N05 G21 G40 G49 G54 G80 G90 G98 G00
N10 T1 M06
N15 G43 H1 Z100.0
N20 M03 S1000
N25 X100.0 Y150.0 Z5
N30 G01 Z-0.5
N35 X200.0 Y250.0
N40 Z5.0
N45 G00 Z100.0
N50 M05
M55 M30
%
```

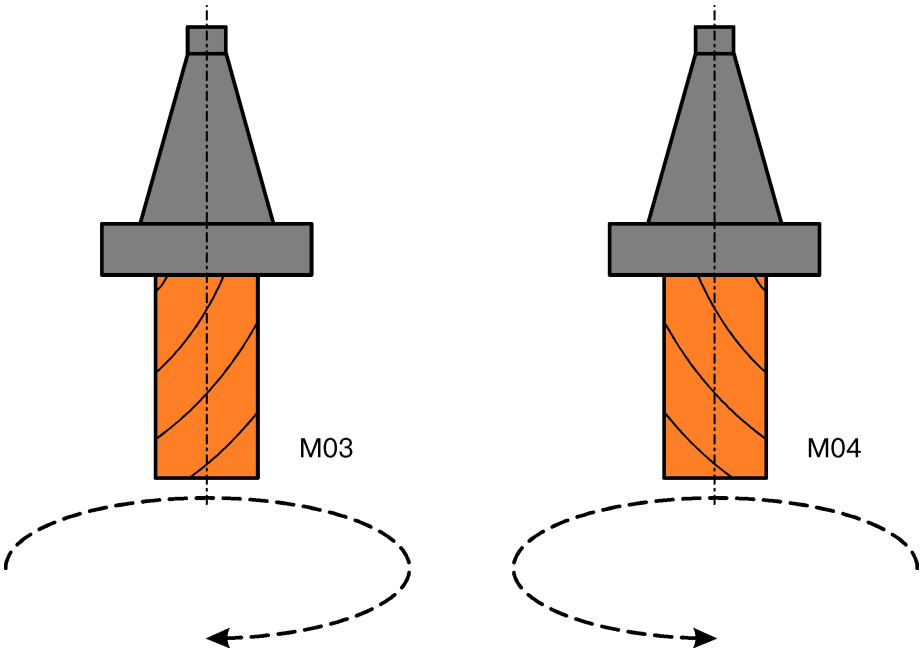



*Рис. 7.1. При фрезеровании используется прямое вращение шпинделя (M03)*

Некоторые станки (преимущественно старых моделей) оснащены специальной коробкой скоростей. Нужная передача для соответствующего диапазона скоростей вращения шпинделя выбирается автоматически или с помощью М-кодов. Обычно для включения низкой передачи используется код M41, а для включения более высоких передач – M42, M43 и т. д. Переключение передач в этом случае необходимо, чтобы двигатель, вращающий шпиндель, не подвергался чрезмерным нагрузкам (вспомните процесс переключения передач в автомобиле). Приведем небольшой программный пример. В кадре N20 включается низкая передача, а в кадре N25 шпиндель начинает вращаться со скоростью 900 оборотов в минуту:

```
...
N10 T5 M06
N15 G43 H5
N20 M41
N25 M03 S900
...
```

# 7.3. Управление подачей СОЖ – M07,M08, M09

В процессе фрезерования охлаждающая жидкость подается в зону обработки для увеличения стойкости инструмента, улучшения качества обрабатываемой поверхности и удаления (вымывания) стружки. Станки с ЧПУ оснащаются системой автоматической подачи СОЖ. Управление этой системой осуществляется при помощи нескольких М-кодов. **Обычно код M08 используется для включения подачи охлаждающей жидкости, а код M09 – для выключения.** Некоторые станки позволяют подавать СОЖ в зону обработки в различном виде. Например, код M08 может вызывать подачу охлаждающей жидкости в виде струи, а код M07 – в распыленном виде.

Подачу СОЖ принято отключать перед сменой инструмента и в конце программы обработки. Многие современные станки делают это автоматически при чтении кода M06 (смена инструмента), кодов M30 и M02 (конец программы). Кроме программного управления системой автоматической подачи СОЖ, существует и ручное управление, позволяющее оператору станка при помощи определенных клавиш на панели УЧПУ включать или выключать подачу охлаждающей жидкости в случае необходимости.

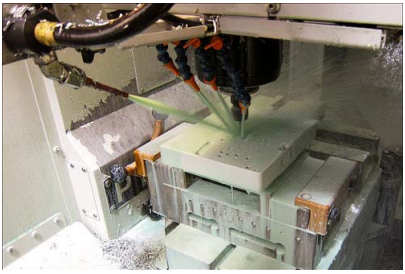


*Рис. 7.2. Подача СОЖ в зону резания*

Для того чтобы струя охлаждающей жидкости точно попадала в нужное место зоны обработки и на режущий инструмент, используют гибкие шланги.

# 7.4. Автоматическая смена инструмента – M06

Код M06 предназначен для автоматической смены инструмента. Некоторые станки с ЧПУ прошлых поколений или недорогие настольные станки не имеют устройства автоматической смены инструмента. В этом случае оператор станка вынужден останавливать программу и вручную менять один инструмент на другой, что конечно же неудобно.

Многие современные станки с ЧПУ имеют это полезное устройство, освобождающее оператора от лишнего вмешательства в производственный цикл станка. Инструменты находятся в ячейках специального барабана, который обычно называют **магазином** инструментов. В большинстве станков каждая из ячеек магазина инструментов имеет собственный номер. Специальные датчики и устройство обратной связи помогают системе ЧПУ определить положение магазина инструментов и наличие инструмента в ячейках.

*Рис. 7.3. Режущие инструменты в магазине станка с ЧПУ*

Обычно для выполнения автоматической смены инструмента программист напрямую указывает номер инструмента, который необходимо взять (номер инструмента в большинстве случаев совпадает с номером ячейки инструментального магазина). Такой способ смены инструмента называется абсолютным. Некоторые старые станки использовали относительный способ смены инструмента. В этом

случае номер инструмента отсчитывался от номера текущего инструмента, что менее удобно.

Производители станков постоянно совершенствуют конструкцию устройств автоматической смены инструмента. Сегодня наиболее популярными являются следующие конфигурации:

- магазин инструментов перемещается при смене инструмента, тип «зонтик»;
- магазин инструментов не перемещается при смене инструмента, тип «рука».

Сначала рассмотрим алгоритм работы устройства автоматической смены инструмента с перемещающимся магазином. Когда управляющая программа доходит до кадра смены инструмента, шпиндель перемещается в определенную точку, находящуюся рядом с магазином инструментов. Магазин инструментов перемещается в эту же точку до «сцепления» инструмента с пустой ячейкой. Шпиндель немного поднимается вверх, освобождая отработавший инструмент. Магазин инструментов поворачивается таким образом, чтобы выбранный инструмент находился под шпинделем. Шпиндель опускается, зажимает новый инструмент и отводится вверх. Магазин инструментов перемещается обратно на свое место.

Если магазин инструментов не перемещается, то возможен следующий алгоритм работы. Сначала шпиндель перемещается в определенную точку, находящуюся рядом с магазином инструментов. Затем магазин инструментов поворачивается таким образом, чтобы выбранный инструмент находился напротив шпинделя. Механический захват («рука»), находящийся между магазином и шпинделем, захватывает отработавший инструмент в шпинделе и новый инструмент в магазине. Захват опускается вниз, освобождает инструмент и меняет их местами. Захват поднимается вверх, при этом новый инструмент зажимается в шпинделе, а старый остается в магазине инструментов.

Внимательно ознакомьтесь с соответствующим разделом документации станка, для того чтобы хорошо понимать, как на вашем станке производится смена инструмента.

Обычно станки с ЧПУ производят смену инструмента при указании в программе следующей команды:

```
M06 T1
```

Адрес T обозначает номер вызываемого инструмента (в данном случае инструмент № 1), а M06 обеспечивает смену. Например, если в УП запрограммировать M06 T5, то будет вызван инструмент № 5. Большинство СЧПУ допускают любой порядок слов данных в кадре смены инструмента. То есть вы можете вызвать инструмент № 1 и таким образом:

```
T1M06
```

Некоторые СЧПУ требуют, чтобы адрес T и команда M06 находились в разных кадрах, иначе автоматическая смена инструмента может быть выполнена неправильно:

```
N10 T1
N20 M06
```

Сразу после смены необходимо выполнить компенсацию длины нового инструмента. Как вы уже знаете, компенсация длины инструмента осуществляется при помощи кода G43 и следующего за ним H-слова данных. Для удобства номер корректора на длину совпадает с номером инструмента. Например, для выполнения компенсации длины инструмента № 1 в УП необходимо указать:

G43 H1

Некоторые станки старых моделей требовали указывать **направление компенсации** длины инструмента. При этом код G43 обозначал положительное направление, а G44 – отрицательное направление компенсации. К счастью, сегодня такой неудобный способ используется крайне редко.

Кадры смены инструмента и активации компенсации длины нового инструмента в управляющей программе:

```
%
O0002
N05 G21 G40 G49 G54 G80 G90 G98 G00
N10 T1 M06
N15 G43 H1 Z100.0
N20 M03 S1000
N25 X100.0 Y150.0 Z5
N30 G01 Z-0.5
N35 X200.0 Y250.0
N40 Z5.0
N45 G00 Z100.0
N50 M05
M55 M30
%
```

В кадре N10 производится смена инструмента (вызов инструмента № 1), а в кадре N15 выполняется компенсация длины инструмента № 1 и инструмент перемещается в точку Z100.0.

Перед тем как вызвать новый инструмент, принято отменять компенсацию длины активного инструмента. Это действие производится при помощи кода G49, хотя многие современные СЧПУ отменяют компенсацию длины автоматически при указании команды M06. **Если произведена смена инструмента, а компенсация его длины не выполнена, то возможно столкновение инструмента с заготовкой или частями станка.**

Многие программисты для обеспечения безопасности перед сменой инструмента выполняют возврат в исходную позицию по оси Z:

```
...
G91 G28 Z0
T3 M06
G43 H3
...
```

Чтобы избежать серьезных ошибок, при смене инструмента оператору станка необходимо быть особенно внимательным.

## 7.5. Завершение программы – M30 и M02

В конце любой управляющей программы должен находиться код ее завершения – M30 или M02. При выполнении любого из этих кодов станок останавливается независимо от того, какую функцию он выполнял. Разница между M30 и M02 заключается лишь в том, что код M30, помимо завершения программы, «перематывает» или «сбрасывает» ее на начало, а код M02 не делает этого. Проще говоря, при окончании программы обработки с M30 курсор текущего положения переводится в самое начало программы, а с M02 остается в конце.

```
...
N40 Z5.0
N45 G00 Z100.0
N50 M05
M55 M30
%
```

Обычно при завершении программы обработки производится перемещение рабочего стола или инструмента в позицию, которая облегчает оператору снятие готовой детали со станка. Такое перемещение совершается с помощью **кода возврата в исходную позицию G28**:

```
...
N120 G91 G28 X0 Y0 Z0
N130 M05
N140 M30
%
```

## Краткое изложение главы

- Коды или функции, обозначающиеся буквой M, называются вспомогательными и предназначены для управления режимами работы станка.
- Большинство современных станков с ЧПУ снабжены устройством автоматической смены инструмента, которое устраняет необходимость ручного вмешательства оператора в процесс замены одного инструмента на другой.
- Сразу после смены инструмента необходимо выполнить компенсацию длины нового инструмента.
- Коды M03 и M04 предназначены для включения вращения шпинделя.
- Код M05 останавливает вращение шпинделя.
- Для задания частоты вращения шпинделя используется адрес S.
- Станки с ЧПУ оснащаются системой автоматической подачи СОЖ.

- В конце любой управляющей программы должен находиться код завершения программы – M03 или M02.
- Коды M00 и M01 временно останавливают выполнение управляющей программы.

# Вопросы

1. Перечислите основные M-коды.
2. Опишите типичное поведение станка при смене инструмента.
3. В чем разница между кодами M03 и M04?
4. С какой скоростью будет вращаться шпиндель при условии, что в УП находится кадр M03 S1200?
5. Для чего нужно подавать охлаждающую жидкость в зону обработки при фрезеровании?
6. В чем разница между кодами M30 и M02?
7. В чем разница между кодами M00 и M01?
8. Назовите команду для автоматической смены инструмента.

# Глава 8
# ПОСТОЯННЫЕ ЦИКЛЫ СТАНКА С ЧПУ

## Введение

**Постоянными циклами** называются специальные макропрограммы, заложенные в УЧПУ для выполнения стандартных операций механической обработки. Практически все станки с ЧПУ имеют набор циклов для обработки отверстий – циклы сверления, растачивания и нарезания резьбы. Эти циклы упрощают процесс написания УП и экономят время, так как позволяют при помощи одного кадра выполнить множество перемещений.

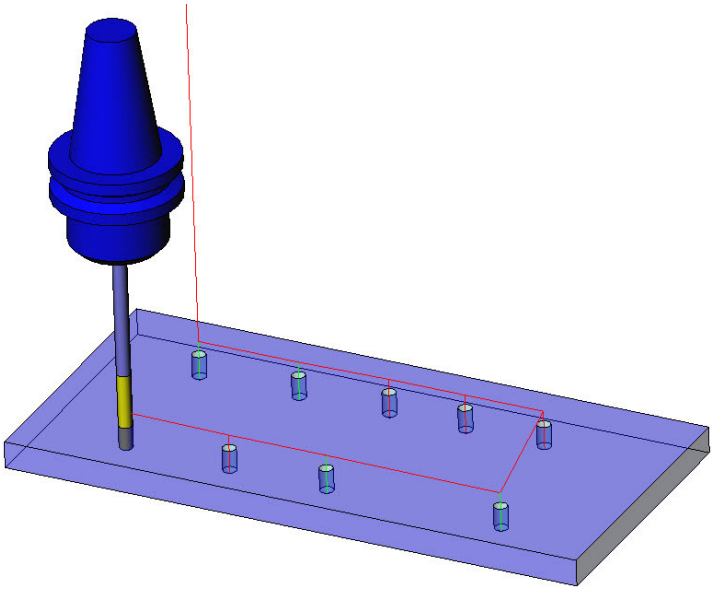


*Рис. 8.1. Постоянные циклы обычно используются для обработки отверстий*

Предположим, что необходимо просверлить несколько отверстий в детали. Чтобы просверлить одно отверстие, нужно на рабочей подаче опустить сверло на требуемую глубину, затем вывести его вверх на ускоренной подаче и переместить к следующему отверстию. Следующая программа демонстрирует, как просверлить несколько отверстий без использования постоянных циклов:

```
%
O0005                                  Начало программы
N100 G21
N102 G0 G17 G40 G49 G80 G90            Строка безопасности
N104 T1 M6                             Вызов инструмента
N106 G0 G90 G54 X5. Y5. S1000 M3       Перемещение к отверстию № 1
N108 G43 H1 Z100.                      Коррекция на длину инструмента
N110 Z10.
N112 G1Z-8. F70.                       Сверление отверстия № 1
N114 G0 Z10.                           Вывод сверла на ускоренной подаче
N116 X15.                              Перемещение к отверстию № 2
N118 G1 Z-8. F70                       Сверление отверстия № 2
N120 G0 Z10.                           Вывод сверла на ускоренной подаче
N122 X-5.                              Перемещение к отверстию № 3
N124 G1 Z-8. F70                       Сверление отверстия № 3
N126 G0 Z10.                           Вывод сверла на ускоренной подаче
N128 X-15.                             Перемещение к отверстию № 4
N130 G1 Z-8. F70                       Сверление отверстия № 4
N132 G0 Z10.                           Вывод сверла на ускоренной подаче
N134 X5. Y-5.                          Перемещение к отверстию № 5
N136 G1 Z-8. F70                       Сверление отверстия № 5
N138 G0 Z10.                           Вывод сверла на ускоренной подаче
N140 X15.                              Перемещение к отверстию № 6
N142 G1 Z-8. F70                       Сверление отверстия № 6
N144 G0 Z10.                           Вывод сверла на ускоренной подаче
N146 X-5.                              Перемещение к отверстию № 7
N148 G1 Z-8. F70                       Сверление отверстия № 7
N150 G0 Z10.                           Вывод сверла на ускоренной подаче
N152 X-15.                             Перемещение к отверстию № 8
N154 G1 Z-8. F70                       Сверление отверстия № 8
N156 G0 Z10.                           Вывод сверла на ускоренной подаче
N158 Z100.
N160 M5
N166 M30                               Конец программы
%
```

Использование постоянного цикла упрощает процесс создания программы для обработки отверстий, делает ее легко читаемой и существенно уменьшает в размере. Создадим новую УП для обработки этих же отверстий с постоянным циклом сверления:

```
%
O0005                                  Начало программы
N100 G21
N102 G0 G17 G40 G49 G80 G90            Строка безопасности
N104 T1 M6                             Вызов инструмента
```

```
N106 G0 G90 G54 X5. Y5. S1000 M3      Перемещение к отверстию № 1
N108 G43 H1 Z100.                     Коррекция на длину инструмента
N110 Z10.
N112 G99 G81 Z-8. R10. F70.           Вызов цикла сверления
N114 X15.                             Координаты отверстия № 2
N116 X-5.                             Координаты отверстия № 3
N118 X-15.                            Координаты отверстия № 4
N120 X5. Y-5.                         Координаты отверстия № 5
N122 X15.                             Координаты отверстия № 6
N124 X-5.                             Координаты отверстия № 7
N126 X-15.                            Координаты отверстия № 8
N128 G80                              Отмена цикла сверления
N130 Z100.
N132 M5
N138 M30                              Конец программы
%
```

Очевидно, что новая программа имеет меньший размер. В кадре N112 находится код G81 для вызова цикла сверления. В этом же кадре находятся адреса, отвечающие за настройку параметров цикла. Адрес Z обозначает глубину сверления, а R определяет высоту отвода сверла из отверстия относительно нулевой плоскости. В последующих кадрах находятся координаты обрабатываемых отверстий. В них не нужно ставить коды вызова цикла сверления, так как G81 будет оставаться активным, пока его не отменят при помощи кода G80.

Работать с постоянными циклами очень удобно. Например, вы решили изменить глубину сверления и высоту вывода сверла из отверстия. При работе с программой без постоянного цикла вам придется отредактировать ее практически полностью. Если же вы используете постоянный цикл сверления, то для достижения нужного эффекта достаточно изменить несколько параметров.

Станки с ЧПУ могут иметь разнообразные циклы: от довольно простых – для сверления, растачивания и нарезания резьбы до более сложных – для обработки контуров и карманов. Некоторые циклы стандартизированы, хотя большинство из них разрабатываются производителями станков и систем ЧПУ самостоятельно. Поэтому на разных станках одинаковые по сути циклы могут записываться поразному, что конечно же затрудняет программирование. В этой главе мы рассмотрим циклы для обработки отверстий, использующиеся на подавляющем большинстве современных станов с ЧПУ.

***Таблица 8.1. Постоянные циклы для обработки отверстий***

| G код | Описание |
|---|---|
| G80 | Отмена постоянного цикла |
| G81 | Стандартный цикл сверления |
| G82 | Сверление с выдержкой |
| G83 | Цикл прерывистого сверления |
| G73 | Высокоскоростной цикл прерывистого сверления |

***Таблица 8.1. Постоянные циклы для обработки отверстий (окончание)***

| G код | Описание |
|---|---|
| G84 | Цикл нарезания резьбы |
| G74 | Цикл нарезания левой резьбы |
| G85 | Стандартный цикл растачивания |

# 8.1. Стандартный цикл сверления и цикл сверления с выдержкой

Код G81 предназначен для вызова **стандартного цикла сверления**. Следующий кадр демонстрирует типичный формат этого цикла:

G81 X10.0 Y15.3 Z-3.0 R0.5 F50.

Адреса X и Y определяют координаты обрабатываемых отверстий. Адрес Z указывает конечную глубину сверления, а R применяется для установления плоскости отвода. Плоскость отвода – это координата по оси Z, с которой начинается сверление на рабочей подаче. Плоскость отвода устанавливается немного выше поверхности детали, поэтому значение при R обычно положительное. Не стоит устанавливать плоскость отвода очень высоко, иначе сверло на рабочей подаче будет перемещаться слишком долго. Рабочая подача для цикла устанавливается с помощью F-слова данных.

**Постоянные циклы и их параметры являются модальными.** Вызвав цикл при помощи соответствующего G-кода, в следующих кадрах вы указываете координаты отверстий, которые необходимо обработать, не программируя никаких других кодов и параметров. После кадра, содержащего координаты последнего отверстия, необходимо запрограммировать G80 – код отмены (окончания) постоянного цикла. Если этого не сделать, то все последующие координаты перемещений будут считаться координатами обрабатываемых отверстий.

**Цикл сверления с выдержкой** вызывается при помощи команды G82. Функционирует этот цикл аналогично стандартному циклу сверления, с единственной разницей в том, что при G82 на дне отверстия запрограммировано время ожидания (выдержка). Цикл сверления с выдержкой часто применяется для сверления глухих отверстий, так как запрограммированное время ожидания обеспечивает лучшее удаление стружки со дна отверстия. Адрес P устанавливает время ожидания на дне отверстия. Как правило, время выдержки указывается в 1/1000 сек. без десятичной точки. Например, в следующем кадре выполняется цикл сверления с выдержкой на дне отверстия, равной 6.5 секунды:

G82 X10.0 Y15.3 Z-3.0 P6500 R0.5 F50.

При выполнении механической обработки отверстий при помощи постоянных циклов вам необходимо знать, что такое исходная плоскость и плоскость отвода. Две эти плоскости используются для управления перемещениями по оси Z между отверстиями. О плоскости отвода мы уже говорили. **Плоскость отвода** – это координата (уровень) по оси Z, устанавливаемая R-адресом, с которой начинается

сверление на рабочей подаче и в которую возвращается инструмент после того, как он достиг дна обрабатываемого отверстия. **Исходная плоскость** – это координата (уровень), по оси Z в которой располагался инструмент перед вызовом постоянного цикла. Код G98 используется для работы с исходной плоскостью, а код G99 – с плоскостью отвода.

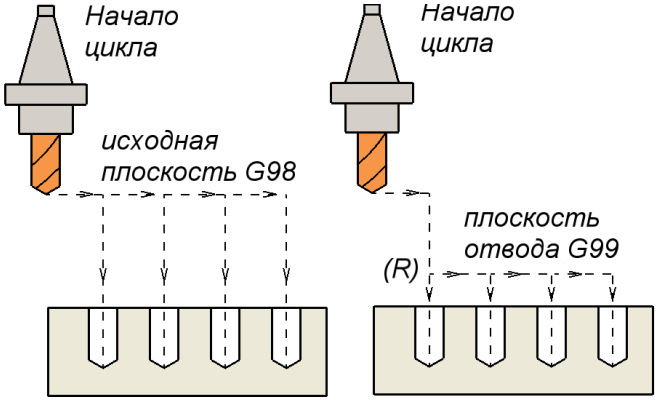



*Рис. 8.2. При использовании кода G98 в постоянном цикле инструмент каждый раз возвращается в исходную плоскость, а при использовании G99 – в плоскость отвода, установленную R-адресом*

Предположим, что инструмент находится в координате Z20.0 в момент вызова цикла сверления. Тогда исходная плоскость будет располагаться на расстоянии 20 мм выше нулевой точки по оси Z. То есть для установления исходной плоскости не требуется указывать какие-либо специальные адреса. Однако для установления плоскости отвода необходимо использовать адрес R. Формат кадра для цикла сверления выглядит следующим образом:

**G98** G81 X10.0 Y15.3 Z-3.0 R0.5 F50.

или

**G99** G81 X10.0 Y15.3 Z-3.0 R0.5 F50.

Если цикл сверления работает совместно с кодом G98, то инструмент возвращается к исходной плоскости в конце каждого цикла и между всеми обрабатываемыми отверстиями. Код G98 применяется, когда требуется увеличенное расстояние отвода, для того чтобы избежать столкновения инструмента с деталью. Учтите, что если вы работаете с G98 сразу после смены инструмента, то исходная плоскость, скорее всего, будет установлена очень высоко, и инструмент будет перемещаться к отверстию непозволительно долго.

Когда нет опасности столкновения инструмента с деталью, то обычно используют код G99, который позволяет сократить время при обработке множества отверстий. В этом случае инструмент перемещается между отверстиями и выводится вверх в конце цикла до координаты по Z, установленной R-словом данных.

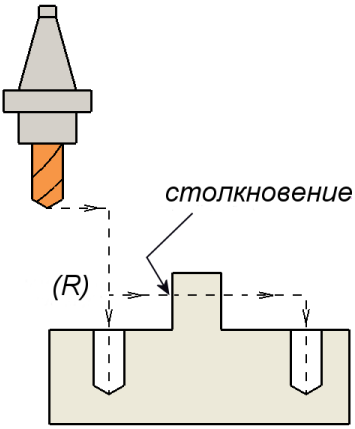



*Рис. 8.3. Будьте особенно внимательны при использовании постоянного цикла с G99. Если плоскость отвода (R) установлена неправильно, то может произойти столкновение инструмента с деталью*

Обычно системы ЧПУ позволяют переключаться между G98 и G99 прямо внутри постоянного цикла между обрабатываемыми отверстиями:

...
**G99** G81 X10.0 Y15.3 Z-3.0 R0.5 F50.
X20 Y20
**G98** X30 Y30
X40 Y40
...

# 8.2. Относительные координаты в постоянном цикле

При необходимости использования относительных координат (G91) при работе с постоянным циклом следует учитывать следующее:

1) плоскость отвода устанавливается относительно исходной плоскости;
2) глубина сверления по Z устанавливается относительно плоскости отвода.

Давайте создадим две программы на сверление отверстий для одной и той же детали. Первая программа будет использовать абсолютные координаты, а вторая – относительные (табл. 8.2).

Как видите, в случае относительного программирования глубина сверления (6 мм ниже нулевой плоскости) устанавливается относительно плоскости отвода, поэтому в программе указано Z-16.

Плоскость отвода (10 мм выше нулевой плоскости) задана относительно координаты Z100, поэтому в цикле сверления указано R-90.

***Таблица 8.2. Абсолютные и относительные координаты в постоянном цикле***

| Абсолютные координаты | Относительные координаты |
|---|---|
| %<br>O0001<br>N100G21<br>N102G0G17G40G49G80**G90**<br>N104T3M6<br>N106G0G90G54X-15.Y2.5S1500M3<br>N108G43H3Z100.<br>**N110G98G81Z-6.R10.F80.**<br>N112X15.<br>N114G80<br>N116M5<br>N118G91G0G28Z0.<br>N120G28X0.Y0.<br>N122M30<br>% | %<br>O0002<br>N100G21<br>N102G0G17G40G49G80G90<br>N104T3M6<br>N106G0G90G54X-15.Y2.5S1500M3<br>N108**G91**<br>N110G43H3Z100.<br>**N112G98G81Z-16.R-90.F80.**<br>N114X30.<br>N116G80<br>N118M5<br>N120G0G28Z0.<br>N122G28X0.Y0.<br>N124M30<br>% |

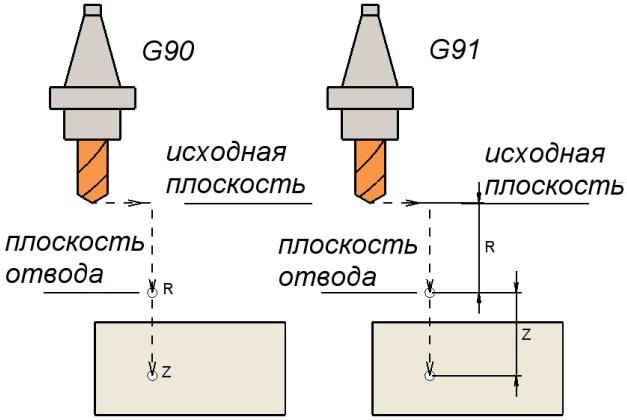



*Рис. 8.4. Разница между G90 и G91 в постоянных циклах сверления*

# 8.3. Циклы прерывистого сверления

Код G83 вызывает **цикл прерывистого сверления**. Прерывистое сверление часто используется при обработке глубоких отверстий. Если при обычном сверлении инструмент на рабочей подаче перемещается ко дну отверстия непрерывно, то в цикле прерывистого сверления инструмент поднимается вверх через определенные интервалы для удаления стружки. Если вы сверлите глубокое отверстие (глубина отверстия больше трех диаметров сверла), то есть вероятность, что стружка не успеет выйти из отверстия и инструмент сломается. При обработке отверстий технолог-программист должен решить, какой именно цикл ему необходим в каждом конкретном случае.

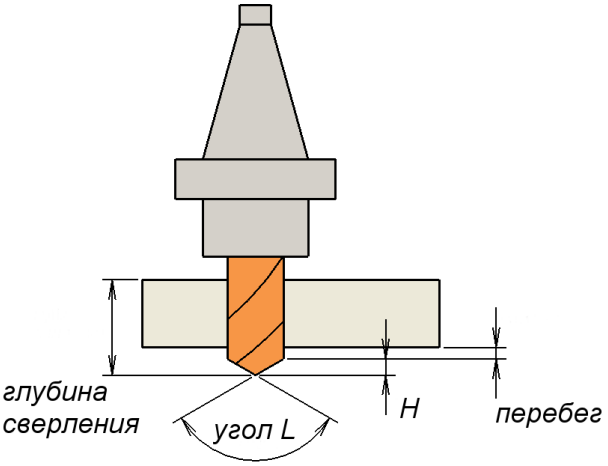



*Рис. 8.5. На чертежах длину отверстия указывают по прямой части. Однако режущая кромка сверла заточена под определенным углом (обычно 118°). Так как в программе указываются координаты Z для кромки сверла, то инструменту необходимо пройти дополнительное расстояние H = R сверла/tan(L/2). При сверлении сквозных отверстий нужно задать небольшой перебег (0.5–1 мм) для прямой части сверла*

Формат кадра для цикла прерывистого сверления похож на формат обычного цикла сверления:

G83 X10.0 Y10.0 Z-25.0 Q2.0 R0.5 F45

Обратите внимание на Q-адрес, который определяет **относительную глубину каждого рабочего** хода сверла. В данном случае сверление происходит по такому алгоритму:

1. Сверло от исходной плоскости перемещается к плоскости отвода (R0.5) на ускоренной подаче.
2. От плоскости отвода R сверло подается на глубину 2 мм (Q2.0) со скоростью подачи (F45).
3. Сверло ускоренным ходом перемещается к плоскости отвода (R0.5).
4. Сверло ускоренным ходом перемещается к ранее достигнутой позиции по глубине (или немного не доходит до этой глубины во избежание столкновения сверла с материалом детали).
5. Сверло подается на глубину 4 мм (2 + 2) со скоростью подачи (F45).
6. Шаги 3, 4 и 5 повторяются до тех пор, пока сверло не достигнет координаты Z-25. Затем сверло выводится из отверстия до плоскости отвода (G99) или исходной плоскости (G98).

**Высокоскоростной цикл прерывистого сверления** G73 работает аналогично циклу G83. Единственная разница заключается в том, что при высокоскоростном цикле сверло для удаления стружки выводится из отверстия не полностью. Это

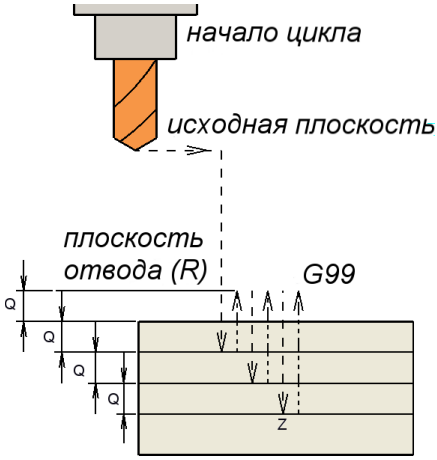



*Рис. 8.6. Схема цикла прерывистого сверления*

позволяет уменьшить машинное время обработки. Формат кадра для высокоскоростного цикла прерывистого сверления:

G73 X10.0 Y10.0 Z-25.0 Q2.0 R0.5 F45

Многие СЧПУ позволяют указывать дополнительные адреса для более гибкой работы с циклами сверления. Внимательно прочитайте документацию к станку для понимания работы циклов и уточните использующиеся в них адреса.

Когда программист задает глубину сверления в программе обработки, он рассчитывает ее относительно крайней кромки сверла. Очень часто на чертежах глубина отверстия указывается относительно прямой части сверла. В этом случае необходимо произвести несложный расчет для нахождения глубины крайней кромки.

Высота кромки сверла H = радиус сверла R/tan (угол L/2).

Если на чертеже указана глубина до прямой части 40 мм, диаметр сверла равен 10 мм, а угол кромки равен 118°, тогда высота кромки H = 5/tan 59 (град.) = = 5/1.664 = 3.004 мм. Следовательно, глубина сверления, которую необходимо указать в управляющей программе, равна 40 + 3.004 = 43.004 мм (Z-43.004).

# 8.4. Циклы нарезания резьбы

Код G84 используется для вызова **цикла нарезания резьбы**. В этом случае при каждой подаче оси Z на значение шага метчика шпиндель поворачивается на один оборот. Когда метчик достигает дна отверстия, шпиндель, вращаясь в обратную сторону, выводит метчик из отверстия. УЧПУ самостоятельно синхронизирует подачу и скорость вращения шпинделя во избежание повреждения резьбы и поломки инструмента. Благодаря этому нарезание резьбы можно выполнить без плавающего патрона с высокой скоростью и точностью.

Формат кадра для цикла нарезания резьбы следующий:

G98 G84 X10.0. Y10.0 Z-6.0 R10.0 F10

Код G74 вызывает **цикл нарезания резьбы при помощи метчика левой резьбы**. Формат этого цикла аналогичен формату для G84. Единственная разница между двумя этими циклами заключается в направлении вращения шпинделя.

G98 G74 X10.0. Y10.0 Z-6.0 R10.0 F10

Некоторые СЧПУ позволяют программировать циклы нарезания резьбы за несколько рабочих операций, аналогично циклу прерывистого сверления. При нарезании резьбы при помощи постоянных циклов станка программисту следует проявлять особую внимательность, назначая режимы резания и глубину обработки.

# 8.5. Циклы растачивания

Код G85 вызывает **стандартный цикл растачивания**. Операция растачивания применяется для получения отверстий высокой точности с хорошей чистотой поверхности. В качестве инструмента используется расточной патрон с настроенным на определенный радиус резцом. Формат для цикла G85 похож на формат цикла сверления:

G98 G85 X10.0 Y10.0 Z-10.0 R10.0 F30

Цикл G85 выполняет перемещение расточного резца до дна отверстия на рабочей подаче с вращением шпинделя. Когда резец достигает дна, инструмент выводится из отверстия также на рабочей подаче.

*Рис. 8.7. Процесс растачивания цилиндрического отверстия*

Существует множество разновидностей цикла растачивания, которые отличаются друг от друга поведением при выводе инструмента из обработанного отверстия. В табл. 8.3 приведены наиболее распространенные расточные циклы.

***Таблица 8.3. Расточные циклы***

| Цикл растачивания | Описание цикла |
|---|---|
| G76 | При достижении дна отверстия расточной резец ориентируется определенным образом, и сдвигается от боковой поверхности (стенки) отверстия и выводится на ускоренной подаче. Для правильной работы с этим циклом необходимо правильно сориентировать инструмент при настройке и установке, иначе можно сломать инструмент или испортить деталь |
| G85 | Стандартный расточной цикл. Инструмент вводится в отверстие на рабочей подаче. При достижении заданной координаты инструмент выводится из отверстия на рабочей подаче |
| G86 | При достижении дна отверстия шпиндель прекращает вращаться и выводится из отверстия на ускоренной подаче. На боковой поверхности (стенке) отверстия, скорее всего, останется вертикальная риска |
| G87 | Поведение цикла может быть различным. У одних станков этот цикл выполняет растачивание за несколько рабочих операций, аналогично циклу прерывистого сверления. У других станков шпиндель останавливается на дне отверстия и выводится из него вручную. На большинстве современных ОЦ является циклом обратного растачивания |
| G88 | Аналогично G87. На дне отверстия можно задать время выдержки |
| G89 | Аналогично G85. На дне отверстия можно задать время выдержки |

# 8.6. Примеры программ на сверление отверстий при помощи постоянных циклов

## Пример № 1

| **Код программы** | **Описание** |
|---|---|
| % | |
| O0001 | Номер программы |
| N100 G21 | Работа в метрической системе |
| N102 G0 G17 G40 G49 G80 G90 | Строка безопасности |
| N104 T1 M6 | Вызов сверла диаметром 3 мм |
| N106 G54 X5. Y5. S1000 M3 | Перемещение к отверстию № 1 |
| N108 G43 H1 Z100. | Компенсация длины инструмента |
| N110 Z10. | Ускоренное перемещение к Z10. |
| N112 G99 G81 Z-6.5 R1. F45. | Стандартный цикл сверления |
| N114 X10. | Сверление отверстия № 2 |
| N116 X15. | Сверление отверстия № 3 |
| N118 X20. | Сверление отверстия № 4 |

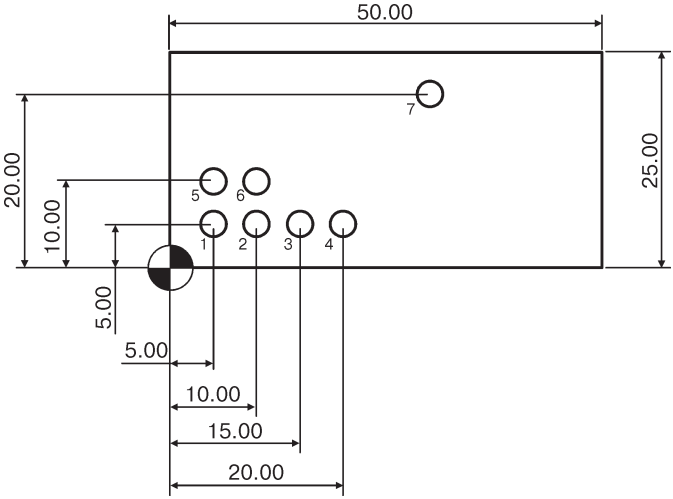



*Рис. 8.8. Необходимо просверлить 7 отверстий диаметром 3 мм и глубиной 6,5 мм*

| | |
|---|---|
| N120 X5. Y10. | Сверление отверстия № 5 |
| N122 X10. | Сверление отверстия № 6 |
| N124 X30. Y20. | Сверление отверстия № 7 |
| N126 G80 | Отмена постоянного цикла |
| N128 Z100. | Перемещение к Z100. |
| N130 M5 | Останов шпинделя |
| N132 G91 G28 Z0. | Возврат в исходную позицию по Z |
| N134 G28 X0. Y0.. | Возврат в исходную позицию по X, Y |
| N136 M30 | Конец программы |
| % | |

## Пример № 2

| № отверстия | Координаты в прямоугольной системе координат (X; Y) |
|---|---|
| 1 | X21.651 Y12.5 |
| 2 | X12.5 Y21.651 |
| 3 | X0. Y25. |
| 4 | X-12.5 Y21.651 |
| 5 | X-21.651 Y12.5 |
| 6 | X-25. Y0. |
| 7 | X-21.651 Y-12.5 |
| 8 | X-12.5 Y-21.651 |
| 9 | X0. Y-25. |
| 10 | X12.5 Y-21.651 |
| 11 | X21.651 Y-12.5 |
| 12 | X25. Y0. |

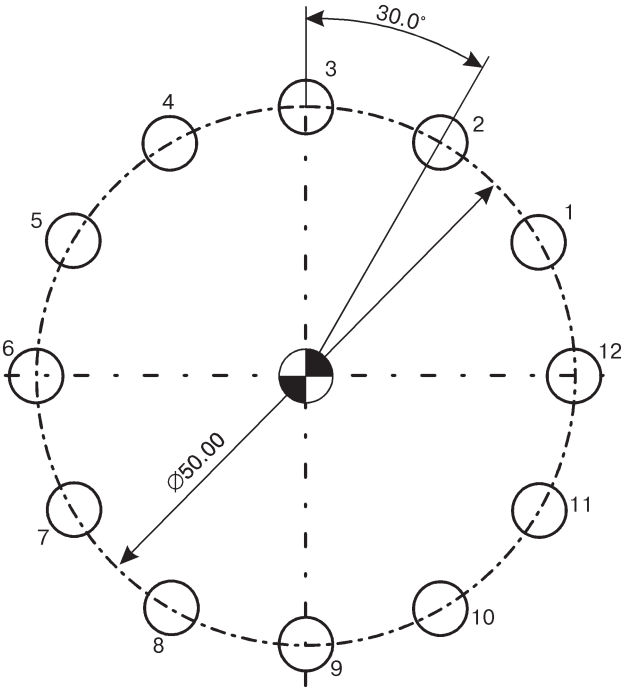



*Рис. 8.9. Необходимо просверлить 12 отверстий диаметром 5 мм и глубиной 40 мм, предварительно выполнить операцию центрования отверстий*

| **Код программы** | **Описание** |
|---|---|
| % | |
| O0002 | Номер программы |
| (PROGRAM NAME – HOLES2 ) | Название программы |
| N100 G21 | Работа в метрической системе |
| N102 G0 G17 G40 G49 G80 G90 | Строка безопасности |
| ( CENTROVKA ) | Комментарий |
| N104 T1 M6 | Вызов центровки |
| N106 G54 X21.651 Y12.5 S1200 M3 | Перемещение к отверстию № 1 |
| N108 G43 H1 Z100. | Компенсация длины инструмента |
| N110 Z2. | Ускоренное перемещение к Z2. |
| N112 G99 G81 Z-.8 R2. F70. | Стандартный цикл сверления |
| N114 X12.5 Y21.651 | Центрование отверстия № 2 |
| N116 X0. Y25. | Центрование отверстия № 3 |
| N118 X-12.5 Y21.651 | Центрование отверстия № 4 |
| N120 X-21.651 Y12.5 | Центрование отверстия № 5 |
| N122 X-25. Y0. | Центрование отверстия № 6 |
| N124 X-21.651 Y-12.5 | Центрование отверстия № 7 |

| | |
|---|---|
| N126 X-12.5 Y-21.651 | Центрование отверстия № 8 |
| N128 X0. Y-25. | Центрование отверстия № 9 |
| N130 X12.5 Y-21.651 | Центрование отверстия № 10 |
| N132 X21.651 Y-12.5 | Центрование отверстия № 11 |
| N134 X25. Y0. | Центрование отверстия № 12 |
| N136 G80 | Отмена постоянного цикла |
| N138 Z100. | Перемещение к Z100. |
| N140 M5 | Останов шпинделя |
| N142 G91 G28 Z0. | Возврат в исходную позицию по Z |
| N144 G28 X0. Y0. | Возврат в исходную позицию по X, Y |
| N146 M01 | Временный останов |
| ( DRILL 12 HOLES ) | Комментарий |
| N148 T2 M6 | Вызов сверла диаметром 5 мм |
| N150 G54 X21.651 Y12.5 S1000 M3 | Перемещение к отверстию № 1 |
| N152 G43 H2 Z100. | Компенсация длины инструмента |
| N154 Z2. | Ускоренное перемещение к Z2. |
| N156 G99 G83 Z-40. R2. Q2. F45. | Цикл прерывистого сверления |
| N158 X12.5 Y21.651 | Сверление отверстия № 2 |
| N160 X0. Y25. | Сверление отверстия № 3 |
| N162 X-12.5 Y21.651 | Сверление отверстия № 4 |
| N164 X-21.651 Y12.5 | Сверление отверстия № 5 |
| N166 X-25. Y0. | Сверление отверстия № 6 |
| N168 X-21.651 Y-12.5 | Сверление отверстия № 7 |
| N170 X-12.5 Y-21.651 | Сверление отверстия № 8 |
| N172 X0. Y-25. | Сверление отверстия № 9 |
| N174 X12.5 Y-21.651 | Сверление отверстия № 10 |
| N176 X21.651 Y-12.5 | Сверление отверстия № 11 |
| N178 X25. Y0. | Сверление отверстия № 12 |
| N180 G80 | Отмена постоянного цикла |
| N182 Z100. | Перемещение к Z100. |
| N184 M5 | Останов шпинделя |
| N186 G91 G28 Z0. | Возврат в исходную позицию по Z |
| N188 G28 X0. Y0. | Возврат в исходную позицию по X, Y |
| N190 M30 | Конец программы |
| % | |

# Краткое изложение главы

- Практически все станки с ЧПУ имеют набор циклов для обработки отверстий – циклы сверления, растачивания и нарезания резьбы. Эти циклы упрощают процесс написания УП и экономят время, так как позволяют при помощи одного кадра выполнить множество перемещений.
- На станках с различными УЧПУ функционально одинаковые циклы могут записываться по-разному, что усложняет создание программы обработки.

- Постоянные циклы и их параметры являются модальными, поэтому при применении циклов легко просверлить множество отверстий, используя минимальное число кадров.
- При работе с постоянным циклом возможно использование абсолютных или относительных координат.
- Плоскость отвода и исходная плоскость используются для управления перемещениями по оси Z между отверстиями.
- Когда программист задает глубину сверления в программе обработки, он рассчитывает ее относительно крайней кромки сверла. Очень часто на чертежах глубина отверстия указывается относительно прямой части сверла. В этом случае необходимо произвести несложный расчет для нахождения глубины крайней кромки.

# Вопросы

1. Что называется постоянным циклом?
2. Перечислите основные циклы сверления.
3. Что такое плоскость отвода?
4. В чем разница между плоскостью отвода и исходной плоскостью?
5. Для чего необходимо указывать код G80 в УП?
6. Для чего используют цикл прерывистого сверления?
7. Что определяется при помощи Q- и R-слов данных?
8. В чем заключается разница между G98 и G99 в постоянных циклах?

# Глава 9
# АВТОМАТИЧЕСКАЯ КОРРЕКЦИЯ РАДИУСА ИНСТРУМЕНТА

## 9.1. Основные принципы

Для того чтобы вам проще было понять, что такое автоматическая коррекция радиуса инструмента и как с ней работать, составим программу для обработки наружного контура детали на рис. 9.1.

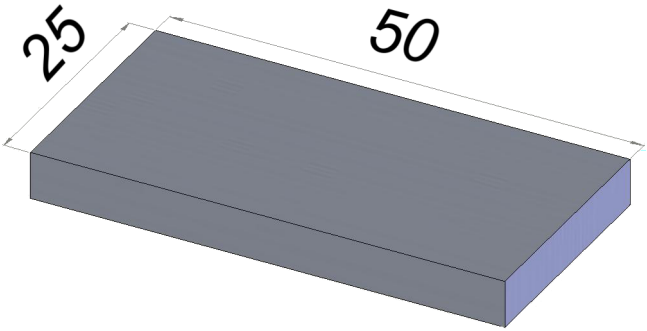



*Рис. 9.1. Необходимо обработать наружный контур этой детали*

Итак, необходимо фрезеровать наружный контур детали концевой фрезой диаметром 10 мм. Так как резание происходит боковой (цилиндрической) частью фрезы, то центр фрезы будет смещен на расстояние, равное радиусу (5 мм), влево относительно обрабатываемого контура. Фреза перемещается последовательно через точки 1, 2, 3 и 4 (рис. 9.2). Так как в программе обработки указываются координаты центра инструмента, то ее можно представить упрощенно в таком виде:

```
...
G01 X-5 Y-5
G01 X-5 Y30
G01 X55 Y30
G01 X55 Y-5
...
```

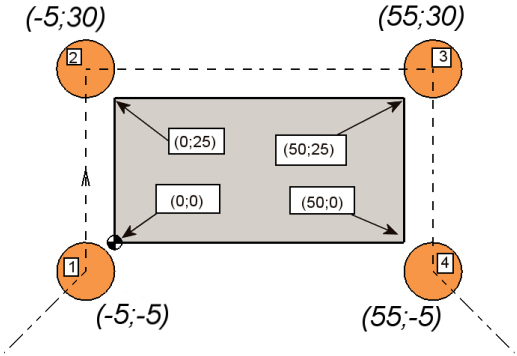



*Рис. 9.2. Исходный контур и эквидистантная траектория для фрезы диаметром 10 мм*

Мы рассчитали координаты опорных точек для фрезы диаметром 10 мм. Предположим, что по каким-либо причинам такой фрезы не оказалось в наличии. Зато в инструментальной кладовой нашлась концевая фреза диаметром 9 мм. Придется составить еще одну программу для новой фрезы, то есть пересчитать все координаты траектории. При обработке контура новой фрезой ее центр будет смещен влево уже не на 5, а на 4.5 мм (рис. 9.3). Новую программу можно представить в следующем виде:

```
...
G01 X-4.5 Y-4.5
G01 X-4.5 Y29.5
G01 X54.5 Y29.5
G01 X54.5 Y-4.5
...
```

Мы столкнулись с довольно простым контуром, поэтому перерасчет программы не занял много времени. Однако если обрабатываемый контур окажется

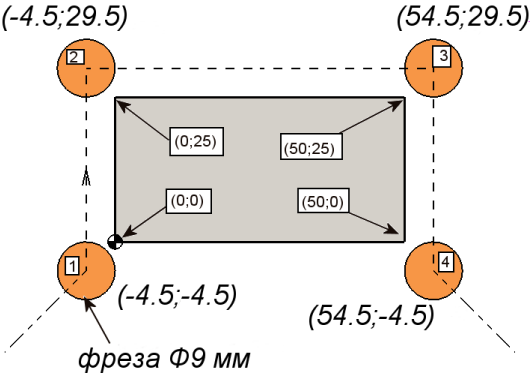



*Рис. 9.3. Исходный контур и эквидистантная траектория для фрезы диаметром 9 мм*

сложным, а диаметр инструмента будет не целочисленным, то расчет координат траектории может представлять собой трудную математическую задачу. Функция станка с ЧПУ для автоматической коррекции радиуса инструмента упрощает такие вычисления и позволяет обрабатывать один и тот же контур различными инструментами по одной управляющей программе.

Теперь создадим программу обработки нашего контура, используя функцию автоматической коррекции радиуса инструмента. Представьте себе, что центр фрезы перемещается прямо по контуру (рис. 9.4) без каких-либо смещений. Тогда управляющая программа будет выглядеть так:

```
...
G01 X0 Y0
G01 Y25
G01 X50
G01 Y0
...
```

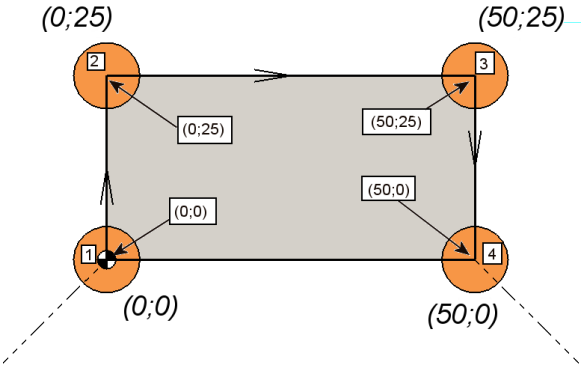



*Рис. 9.4. Опорные точки траектории совпадают с опорными точками контура детали*

Очевидно, что такая программа не обеспечивает правильной обработки. Необходимо, чтобы траектория центра инструмента была смещена относительно контура на величину радиуса. Система ЧПУ способна самостоятельно рассчитать и выполнить такое смещение – коррекцию на радиус инструмента. Для осуществления автоматической коррекции на радиус инструмента нужно сообщить системе величину радиуса инструмента и в управляющей программе указать соответствующий G-код. В табл. 9.1 сведены коды и адреса для автоматической коррекции радиуса инструмента.

Автоматическая коррекция радиуса инструмента активируется при помощи G-кода, который вызывает смещение инструмента относительно исходной траектории вправо или влево. Код G41 используется для коррекции инструмента слева (рис. 9.7), а код G42 – для коррекции справа (рис. 9.8). Направление смещения определяется, если смотреть на траекторию сверху вниз, то есть со стороны «+Z» в направлении «–Z». Отмена коррекции осуществляется при помощи G40.

***Таблица 9.1. Коды и адреса, использующиеся для автоматической коррекции на радиус инструмента***

| Код | Действие | Код | Действие |
|---|---|---|---|
| G41 | Коррекция слева | G40 | Отмена коррекции |
| G42 | Коррекция справа | D | Номер корректора на радиус инструмента |

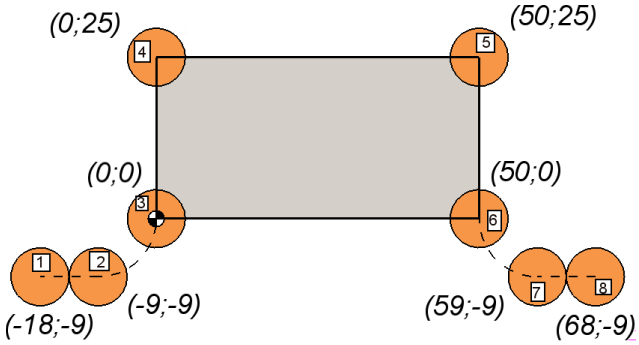



*Рис. 9.5. Обработка контура с коррекцией, участками подвода и отвода*

Теперь мы можем создать полную программу с коррекцией, которая обеспечит правильную обработку нашего контура фрезой диаметром 9 мм.

| **Код программы** | **Пояснение** |
|---|---|
| % | |
| O0004 | Номер программы |
| N100 G21 | Работа в метрической системе |
| N102 G0 G17 G40 G49 G80 G90 | Строка безопасности |
| N104 T2 M6 | Вызов инструмента № 2 |
| N106 G54 X-18. Y-9. S1200 M3 | Ускоренное перемещение в позицию № 1 |
| N108 G43 H2 Z100. | Компенсация длины инструмента |
| N110 Z10. | Ускоренное перемещение в Z10 |
| N112 G1 Z-1. F120 | Перемещение в Z-1 с F = 120 мм/мин |
| N114 **G41 D2** X-9. F200 | Перемещение с коррекцией в позицию № 2 |
| N116 G3 X0. Y0. R9. | Подход к контуру по дуге в позицию № 3 |
| N118 G1 Y25. | Перемещение в позицию № 4 |
| N120 X50. | Перемещение в позицию № 5 |
| N122 Y0. | Перемещение в позицию № 6 |
| N124 G3 X59. Y-9. R9. | Отвод от контура по дуге в позицию № 7 |
| N126 G1 G40 X68. | Перемещение и отмена коррекции |
| N128 Z9. F300 | Подъем инструмента в Z9 |
| N130 G0 Z100. | Ускоренное перемещение в Z100 |
| N132 M5 | Останов шпинделя |
| N138 M30 | Конец программы |
| % | |

В кадре N114 активируется автоматическая коррекция радиуса инструмента. Код G41 включает коррекцию слева, а D-слово данных обозначает номер корректора, содержащего значение радиуса инструмента. **В памяти любой современной СЧПУ существует область (таблица инструментов), в которой хранятся значения длин и радиусов инструментов.**

Если мы работаем фрезой № 2 диаметром 9 мм, то в соответствующем корректоре № 2 должно находиться значение 4,5 мм (табл. 9.4). Именно на эту величину и происходит смещение центра инструмента влево относительно запрограммированного контура.

***Таблица 9.4. Область корректоров (таблица инструментов) системы ЧПУ***

| № инструмента | Длина | Радиус | Другие параметры |
|---|---|---|---|
| 1 | 100,82 | 5 | ... |
| 2 | 87,67 | **4,5** | ... |
| 3 | 133,45 | 6 | ... |
| ... | ... | ... | ... |

Таким образом, если на рис. 9.5 вы видите **расчетную траекторию**, совпадающую с исходным контуром, то на рис. 9.6 показана **реальная траектория** обработки фрезой диаметром 9 мм, смещенная на 4,5 мм влево относительно исходного контура.

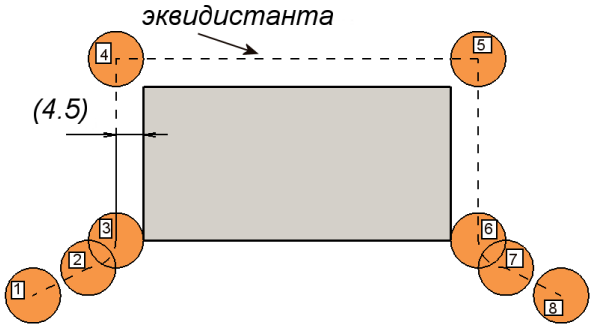



*Рис. 9.6. Реальная траектория при обработке с коррекцией*

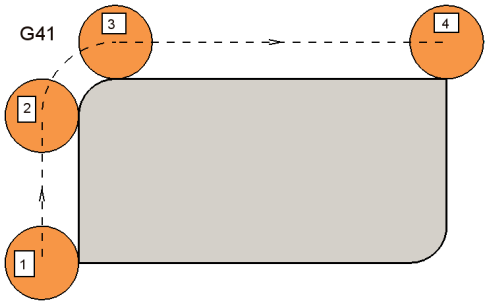



*Рис. 9.7. Коррекция слева (G41) от контура*

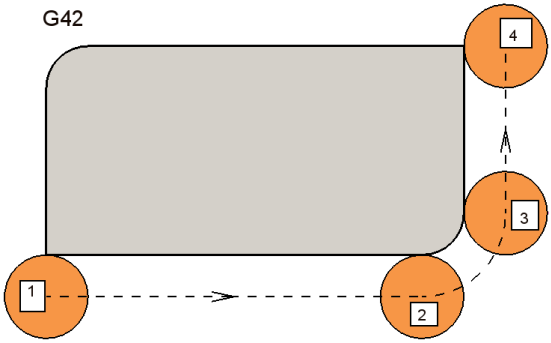



*Рис. 9.8. Коррекция справа (G42) от контура*

# 9.2. Использование автоматической коррекции на радиус инструмента

Для чего на самом деле применяется функция автоматической коррекции радиуса инструмента? Для работы разными инструментами по одной программе? Не только. **Главное назначение этой функции заключается в управлении размерами детали при фрезеровании.**

Предположим, вы обработали контур фрезой диаметром 9 мм по программе с коррекцией на радиус. Естественно, что в корректоре находится значение радиуса фрезы, равное 4,5 мм. При измерении размеров детали вы обнаружили, что размер 25 мм (стороны прямоугольного контура, рис. 9.1) выполнен «в плюсе», например 25,02 мм. Может быть, диаметр фрезы оказался чуть меньшим (последствия износа или погрешность измерения), а может, ее немного отжимало при обработке. В любом случае, инструмента с идеальными размерами не существует. Однако чертеж требует, чтобы этот размер был выполнен «в номинале» или «в минусе», например 25–0,02 мм. Что же делать? Нужно просто уменьшить значение радиуса в корректоре с 4,5 мм до 4,49 мм (табл. 9.5) и запустить программу на выполнение еще раз. В этом случае система ЧПУ будет считать, что мы используем инструмент с меньшим радиусом, и сместит (приблизит) траекторию центра инструмента на 4,49 мм относительно исходного контура. Так как на самом деле используется фреза с радиусом 4,5 мм, то контур после обработки окажется меньше на 0,02 мм (при обработке уменьшится по 0,01 мм с каждой стороны), то есть мы получим требуемый размер 25 мм.

***Таблица 9.5. Новое значение в корректоре для инструмента № 2***

| № инструмента | Длина | Радиус | Другие параметры |
|---|---|---|---|
| 1 | 100,82 | 5 | ... |
| 2 | 87,67 | 4,49 | ... |
| 3 | 133,45 | 6 | ... |
| ... | ... | ... | ... |

Если увеличить числовое значение в корректоре, например, до 5 мм, то фреза не дойдет 0,5 мм до обрабатываемого контура. Таким образом, «играя» значениями радиусов в корректорах, можно получать размеры детали в пределах допусков, указанных на чертеже, без пересчета программы.

Еще одним преимуществом от использования функции автоматической коррекции радиуса инструмента является возможность работать непосредственно с чертежным контуром детали. То есть координаты практически всех опорных точек очевидны, их можно «взять» прямо с чертежа без каких-либо дополнительных расчетов. Это также в значительной мере позволяет упростить процесс написания УП.

# 9.3. Активация, подвод и отвод

Для того чтобы система ЧПУ успела выполнить смещение относительно запрограммированного контура, **необходимо добавить к исходной траектории участок подвода**. На этом участке происходит активация автоматической коррекции радиуса инструмента.

Большинству систем для активации коррекции требуется пройти расстояние, не меньшее величины радиуса инструмента. То есть если диаметр инструмента равен 9 мм, то, прежде чем приступить к обработке контура с коррекцией, необходимо запрограммировать прямолинейное перемещение на расстояние не менее 4,5 мм. Кстати, прежде чем активировать коррекцию на радиус, не забудьте выполнить компенсацию длины инструмента. Посмотрите внимательно на кадр N114 в созданной ранее программе:

N114G41D2X-9.F200

В этом кадре находятся код G41, слово данных D2 и запрограммировано линейное перемещение в X-9 (перемещение из позиции № 1 в позицию № 2). Код G41 включает коррекцию слева, D2 считывает значение радиуса инструмента из корректора № 2, а на участке прямолинейного перемещения происходит смещение инструмента влево относительно запрограммированного контура. Таким образом, к позиции № 2 инструмент подходит уже по смещенной (эквидистантой) траектории. Далее обычно программируют участок плавного подхода инструмента к обрабатываемому контуру – по касательной (из позиции № 2 в позицию № 3, к первой точке контура). Следует заметить, что если подход к контуру осуществляется по прямой линии, то угол подхода должен быть не менее 90°.

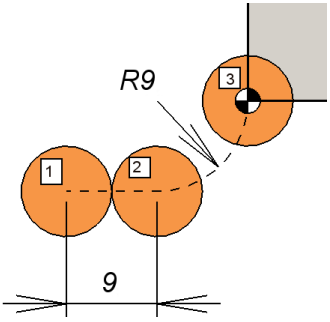



*Рис. 9.9. Прямолинейный участок для активации коррекции и участок плавного подхода (по касательной) к контуру*

**Обязательным условием для активации коррекции является наличие именно прямолинейного перемещения на рабочей подаче.** При попытке активировать коррекцию на радиус вместе с перемеще-

нием по дуге СЧПУ выдаст сообщение об ошибке. Поэтому сначала активируют коррекцию вместе с прямолинейным перемещением, а затем по дуге (или по прямой) осуществляют подход к контуру.

**При работе с коррекцией на радиус есть ряд ограничений.** Если запрограммированный радиус контура окажется меньше значения радиуса инструмента в корректоре, то СЧПУ может выдать сообщение об ошибке. Многие станки позволяют выполнять коррекцию на радиус только в плоскости X–Y. В некоторых случаях ошибкой будет считаться линейное перемещение с шагом, меньшим, чем значение радиуса инструмента в соответствующем корректоре. Ряд СЧПУ не позволяют, чтобы участок подхода к контуру был менее 90°.

Будьте внимательны при работе с адресом D. Многие СЧПУ хранят информацию о радиусе и длине инструмента в разных корректорах. В табл. 9.4 и 9.5 вы видите, что значения радиуса и длины каждого инструмента находятся в одной строке. Поэтому в УП мы указывали H- и D-слова данных с одинаковыми числовыми значениями:

```
...
N108 G43 H2 Z100.
...
N114 G41 D2 X-9. F200
...
```

или

```
...
N108 G43 H1 Z100.
...
N114 G41 D1 X-9. F200
...
```

Возможна ситуация, когда значения радиуса и длины инструмента находятся в разных корректорах (строках) таблицы инструментов (табл. 9.6):

***Таблица 9.6. Область корректоров СЧПУ***

| № корректора | Значение |
|---|---|
| 1 | 100. 82 |
| 2 | 87.67 |
| 3 | 133.45 |
| … | … |
| 101 | 5 |
| 102 | 4.5 |
| 103 | 23.08 |
| .. | … |

В этом случае числовые значения для D- и H-слов данных будут разными:

```
...
N108 G43 H2 Z100.
```

...
N114 G41 D102 X-9. F200
...

После того как инструмент обошел контур полностью, необходимо плавно отвести инструмент и отменить коррекцию при помощи кода G40 или D00. Как и в случае начального подвода инструмента к контуру, отвод от контура выполняется с теми же принципами и условиями. Оптимальный вариант – сначала отход по касательной, затем кадр, содержащий линейное перемещение с кодом отмены коррекции:

...
N124 G3 X59. Y-9. R9. Отход от контура по касательной
N126 G1 **G40** X68. Отмена коррекции на радиус
...

Функция автоматической коррекции позволяет работать по одной программе инструментами с различным диаметром. Однако, исходя из перечисленных ограничений, рекомендуется, чтобы расчетный и фактический диаметры инструмента отличались несильно. Например, вы рассчитывали работать фрезой диаметром 10 мм и соответствующим образом составили программу обработки. Вы смело можете изменить значение радиуса в корректоре с 5 мм до 4,5 мм и работать фрезой диаметром 9 мм. Но изменение значения радиуса в корректоре, например, на 30 мм может привести к сообщению об ошибке или зарезу контура.

# Краткое изложение главы

- Автоматическая коррекция радиуса инструмента применяется для автоматического смещения траектории инструмента относительно исходного контура.
- Код G41 активирует коррекцию слева, код G42 активирует коррекцию справа.
- Адрес D определяет номер корректора, в котором находится значение радиуса инструмента.
- Изменяя значения радиусов в корректорах, можно получать размеры детали в пределах допусков, указанных на чертеже, без пересчета программы.
- Для активации коррекции обычно нужно запрограммировать линейное перемещение на расстояние, не меньшее, чем радиус инструмента.
- После того как инструмент обошел контур полностью, необходимо плавно отвести его от контура и отменить коррекцию при помощи кода G40 или D00.
- При работе с автоматической коррекцией на радиус есть некоторые ограничения, которые обязательно нужно учитывать.

# Вопросы

1. Для чего используют функцию автоматической коррекции на радиус инструмента?

2. Как вы думаете, функцию автоматической коррекции на радиус инструмента чаще применяют при черновой или чистовой обработке?
3. Перечислите G-коды для автоматической коррекции радиуса инструмента.
4. Откуда система ЧПУ узнает о диаметре используемого инструмента?
5. Можно ли активировать коррекцию в кадре с перемещением по дуге?
6. Когда нужно отменить автоматическую коррекцию радиуса инструмента?
7. Какие существуют ограничения при работе с функцией автоматической коррекции на радиус инструмента?
8. Что принято указывать в УП раньше – компенсацию длины инструмента или автоматическую коррекцию радиуса инструмента?

# Глава 10
# ОСНОВЫ ЭФФЕКТИВНОГО ПРОГРАММИРОВАНИЯ

## 10.1. Подпрограмма

Язык G- и M-кодов, как и любой другой язык программирования, позволяет работать с подпрограммами и совершать переходы. Посредством функции подпрограммы основная (главная) управляющая программа может вызывать из памяти другую программу (подпрограмму) и выполнить ее определенное число раз. Если УП содержит часто повторяемое действие или работает по определенному шаблону, то использование подпрограмм позволяет упростить программу обработки и сделать ее гораздо меньшей в размере.

Существуют два вида подпрограмм – внутренние и внешние. **Внутренние подпрограммы** вызываются при помощи кода M97 и содержатся внутри главной программы. То есть они находятся в одном файле. **Внешние подпрограммы** вызываются кодом M98 и не содержатся в теле главной программы. В этом случае главная программа и подпрограмма находятся в разных файлах.

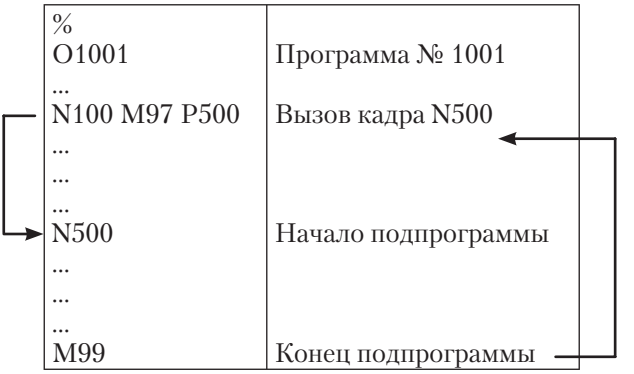



*Рис. 10.1. Схема внутренней подпрограммы*

Внутренняя подпрограмма выполняется, когда СЧПУ встречает код M97. При этом адрес P указывает на номер кадра, к которому нужно перейти, то есть туда, где начинается внутренняя подпрограмма. Когда СЧПУ находит кадр с кодом окончания подпрограммы M99, то выполнение внутренней подпрограммы

| Главная программа | Подпрограмма |
|---|---|
| %<br>O1023<br>N10 G21 G40 G49 G54 G80 G90<br>N20 T3 M06<br>N30 G43 H3<br>N40 M03 S1000<br>N50 G00 X0 Y0<br>N60 Z0.5<br>N70 M98 P2000<br>N80 M05<br>N90 M02<br>% | %<br>O2000<br>N10 G01 X-0.5 F50<br>N20 X10 Y10<br>N30 Z0.5<br>N40 M99<br>% |

*Рис. 10.2. Схема внешней подпрограммы*

завершается и управление передается кадру главной программы, следующему за кадром, вызвавшим завершенную подпрограмму.

Внешние подпрограммы работают похожим образом. Когда в главной программе встречается кадр с кодом M98, то вызывается подпрограмма с номером, установленным при помощи P-адреса. При нахождении кода M99 управление возвращается главной программе, то есть выполняется кадр главной программы, следующий за кадром с M98. Учтите, что внешняя подпрограмма находится в отдельном файле. По сути, внешняя подпрограмма – это отдельная программа с индивидуальным номером, которая при желании может быть выполнена независимо от главной программы. Для вызова подпрограммы необходимо, чтобы она находилась в памяти СЧПУ.

Пример УП с внутренней подпрограммой:

| | |
|---|---|
| % | |
| O1023 | Программа № 1023 |
| N10 G21 G40 G49 G54 G80 G90 | Строка безопасности |
| N20 T3 M06 | Вызов инструмента № 3 |
| N30 G43 H3 | Компенсация длины инструмента |
| N40 M03 S1000 | Включение оборотов шпинделя |
| N50 G00 X0 Y0 | Позиционирование в X0 Y0 |
| N60 Z0.5 | Позиционирование в Z0.5 |
| N70 **M97 P200** | Вызов внутренней подпрограммы |
| N80 M05 | Выключение оборотов шпинделя |
| N90 M02 | Окончание программы |
| **N200** G01 X-0.5 F50 | Начало внутренней подпрограммы |
| N210 X10 Y10 | ... |
| N220 Z0.5 | ... |
| M230 **M99** | Конец внутренней подпрограммы |
| % | |

При помощи L-адреса определяется, сколько раз нужно вызвать ту или иную подпрограмму. Если подпрограмму нужно вызвать всего один раз, то L в кадре можно не указывать.

M98 P1000 L4 – подпрограмма будет вызвана 4 раза.

Большим преимуществом от использования подпрограмм является возможность удобной и эффективной работы с программными массивами и шаблонами. Например, для обработки детали, изображенной на рис. 11.3, мы создадим главную программу и подпрограмму, и вы увидите, насколько удобнее и проще будет работать в этом случае.

Итак, на рис. 10.3 изображена деталь с 4 группами отверстий диаметром 3 мм. Нулевой точкой является верхний левый угол детали. Сначала создадим главную программу, которая будет позиционировать инструмент к каждой группе отверстий. Затем напишем подпрограмму, необходимую для сверления 4 отверстий в одной группе. Учтите, что в подпрограмме используются относительные координаты, а смена инструмента и основные команды находятся в главной программе.

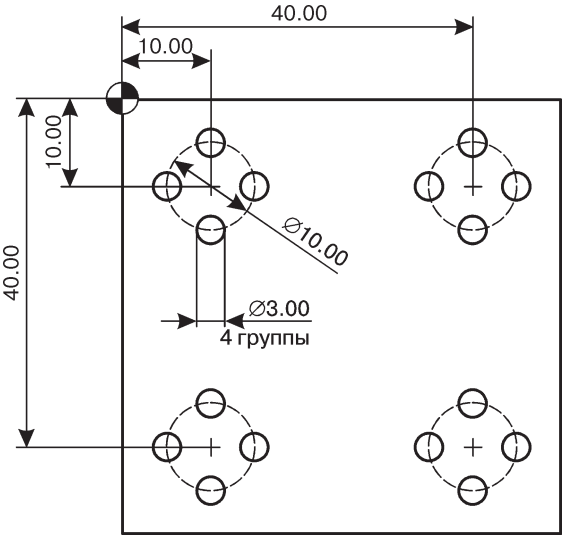



*Рис. 10.3. Использование подпрограмм при обработке повторяющихся элементов позволяет уменьшить размер программы*

| **Главная программа** | **Пояснение** |
|---|---|
| % | |
| O0001 | Программа O0001 |
| N10 G90 G40 G80 G49 G98 G21 | Строка безопасности |
| N20 T1 M06 | Вызов сверла Ф3 мм |
| N30 G43 H1 | Компенсация длины инструмента |
| N40 M03 S1000 | Включение оборотов шпинделя |

| | |
|---|---|
| N50 G00 X10 Y-10 | Позиционирование к 1-ой группе |
| N60 Z0.5 | Подвод сверла к детали по Z |
| N70 M98 P1000 | Вызов подпрограммы O1000 |
| N70 **G00 X40 Y-10** | Позиционирование ко 2-ой группе |
| N80 M98 P1000 | Вызов подпрограммы O1000 |
| N90 G00 X10 Y-40 | Позиционирование к 3-ей группе |
| N100 M98 P1000 | Вызов подпрограммы O1000 |
| N110 G00 X40 Y-40 | Позиционирование к 4-ой группе |
| N120 M98 P1000 | Вызов подпрограммы O1000 |
| N130 G91 G28 Z0 | Отвод инструмента по Z |
| N140 M05 | Останов шпинделя |
| N150 M30 | Конец программы |
| % | |

| **Подпрограмма** | **Пояснение** |
|---|---|
| % | |
| **O1000** | Подпрограмма O1000 |
| N10 G91 | Режим относительных координат |
| N20 G99 G81 X5 Y0 Z-5 R0.5 | Цикл сверления и 1-ое отверстие |
| N30 X-5 Y-5 | Координаты 2-го отверстия |
| N40 X-5 Y5 | Координаты 3-го отверстия |
| N50 X5 Y5 | Координаты 4-го отверстия |
| N60 G80 | Отмена цикла сверления |
| N70 G90 | Режим абсолютных координат |
| N80 **M99** | Возврат в главную программу |
| % | |

Обе эти программы необходимо передать в СЧПУ, оператор станка должен вызвать программу O0001 и запустить ее. Главная программа работает с абсолютными координатами и перемещает инструмент последовательно в центр каждой из четырех групп отверстий. Когда СЧПУ встречает кадр с M98 P1000, то происходит передача управления подпрограмме с номером 01000. В этот момент инструмент уже находится в центре группы отверстий. Переключаемся в режим относительного (инкрементального) программирования и используем постоянный цикл сверления. После завершения сверления четырех отверстий одной группы выключаем цикл сверления командой G80 и переходим в режим абсолютных координат G90, для того чтобы правильно выполнить позиционирование в главной программе. Код M99 передает управление кадру главной программы, который следует за кадром, вызвавшим эту подпрограмму. Затем инструмент перемещается в центр следующей группы отверстий, и снова вызывается подпрограмма 01000. И так далее, пока не просверлим все отверстия и СЧПУ не прочтет код завершения программы M30.

Если бы мы создавали обычную программу обработки, то ее размер был бы значительно больше, так как пришлось бы указывать координаты всех 16 отверстий. Работая в таком формате, легче производить изменения. Например, если из-

менится диаметр окружности, на которой находятся отверстия группы, то в случае работы с подпрограммой достаточно пересчитать координаты центров четырех отверстий только в подпрограмме.

Из главной программы можно вызвать несколько различных подпрограмм. Более того, из каждой подпрограммы можно вызвать несколько других подпрограмм. Системы ЧПУ могут накладывать ограничения на вложенность и количество выполняемых подпрограмм, поэтому внимательно прочитайте документацию станка и стойки, прежде чем начнете работать с подпрограммами.

## 10.2. Работа с осью вращения (4-ой координатой)

Нередки случаи, когда на трехкоординатный станок с ЧПУ дополнительно монтируют управляемый поворотный стол (делительную головку). **Управляемый поворотный стол** – это устройство, которое способно поворачивать закрепленную в нем деталь на требуемый угол по определенной команде. Обычно 4-ая ось управляется при помощи адресов A или B, а числовое значение определяет угол поворота в градусах.

*Рис. 10.4. Управляемые поворотные столы HAAS*

Существуют два варианта работы с управляемым поворотным столом. Первый вариант – нам просто необходимо повернуть его на определенный угол и затем выполнить какую-либо технологическую операцию (индексация). Второй вариант – нужно выполнить фрезерование одновременно с поворотом стола. В этом случае мы имеем синхронное линейное перемещение исполнительного органа станка по трем (или менее) координатам с вращением стола. При этом СЧПУ станка должна поддерживать данный вид интерполяции.

Для управления поворотным столом достаточно в кадр с линейной интерполяцией, позиционированием или постоянным циклом добавить адрес A (B):

- G00 X_Y_Z_A_ – позиционирование;
- G01 X_Y_Z_A_F_ – линейная интерполяция.

Типичный формат для работы с постоянным циклом:

G81 X0 Y0 Z-5 A0 F45 R0.5
A15
A30
A45
G80

Программирование 4-ой оси не должно вызвать у вас особых трудностей. Просто нужно учесть несколько технических особенностей при работе с управляемым поворотным столом. Во-первых, поворотный стол может вращаться как в положительном, так и в отрицательном направлении. Направление вращения и соответствующий знак определяются по правилу правой руки. Во-вторых, поворот стола может быть запрограммирован как в абсолютных, так и в относительных координатах. В-третьих, у многих станков существует ограничение на числовое значение угла поворота. Например, вам нужно повернуть стол на 400°, а СЧПУ позволяет указывать угол не более 360°. Придется запрограммировать дополнительный кадр с углом в 40° относительно предыдущего положения стола. Ну и напоследок учтите, что чем дальше мы удалимся от центра вращения, тем большей будет ошибка линейного перемещения.

Следующие примеры помогут вам понять, как программируется дополнительная ось вращения. В первом случае необходимо просверлить отверстия на периферии диска. Во втором случае нужно получить винтовую канавку на поверхности вала, используя одновременное линейное перемещение фрезы и вращение поворотного стола.

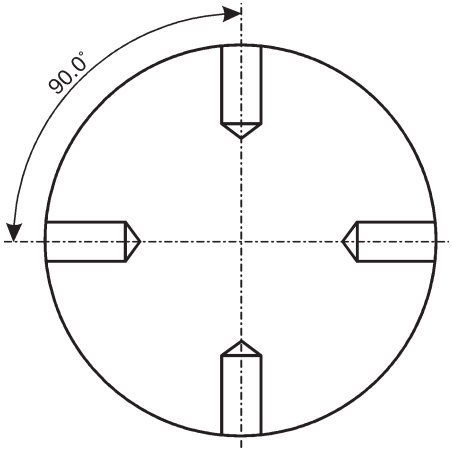



*Рис. 10.5. Требуется просверлить 4 отверстия на периферии диска, закрепленного в кулачках поворотного стола. Чтобы просверлить такие отверстия, нужно поворачивать стол через 90°*

```
%
O3000                                  Программа O3000
N10 G21 G40 G49 G80 G90 G98            Строка безопасности
N20 G54                                Рабочая система координат
N30 T2 M06                             Вызов инструмента № 2
N40 M03 S1000                          Включение оборотов шпинделя
N50 G00 X0 Y0                          Позиционирование в X0 Y0
N60 Z5                                 Позиционирование в Z5
N70 G81 X0 Y0 Z-10 A0 F45 R1           Цикл сверления, сверление 1-го отверстия
N80 A90                                Поворот стола на 90°, сверление
N90 A180                               Поворот стола на 180°, сверление
N100 A270                              Поворот стола на 270°, сверление
N110 G80                               Отмена цикла сверления
N120 G91 G00 A-270                     Поворот стола в начальное положение
N130 M05                               Выключение оборотов шпинделя
N140 M30                               Конец программы
%
```

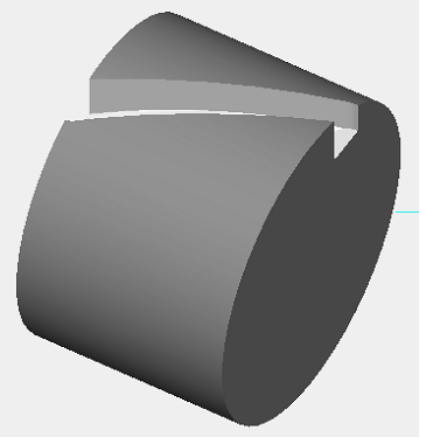


*Рис. 10.6. Необходимо получить винтовую канавку на поверхности вала. Вал закреплен в кулачках управляемого поворотного стола. Самый простой способ обработки такой канавки – расчет при помощи CAD/CAM-системы*

```
%
O0001
N100 G21
N102 G0 G17 G40 G49 G80 G90
N104 T1 M6
N106 G0 G90 G54 X-16.612 Y0. A-2.993 S1000 M3
N108 G43 H1 Z125.171 M8
N110 Z35.605 A-10.578
```

```
N112 G1 Z33.932 A13.459 F200.
N234 G0 Z123.253 A3.674
N236 M5
N238 G91 G28 Z0. M9
N240 G28 X0. Y0. A0.
N242 M30
%
```

# 10.3. Параметрическое программирование

Одним из самых интересных и эффективных методов программирования обработки является параметрическое программирование. Удивительно, но большинство технологов-программистов хоть и слышали об этом методе, но совершенно не умеют его использовать. В этом разделе вы познакомитесь с теорией параметрического программирования и коснетесь основ макроязыка системы ЧПУ современного станка.

Большинство станочных систем ЧПУ имеют в своем распоряжении специальный язык для параметрического программирования (макропрограммирования). Например, в СЧПУ Fanuc этот язык называется Macro B. Если вы хоть немного знакомы с языком программирования Бейсик (Basic), то вы без труда разберетесь и с Macro B. Команды и функции именно этого языка мы рассмотрим подробно.

В обычной управляющей программе вы указываете различные G-коды, а также направления и величины перемещений при помощи числовых значений. Например, G10 или X100. Однако СЧПУ станка может делать то же самое при помощи переменных.

Символом переменной в Macro B является знак #. Например, в программе можно указать следующие выражения:

```
...
#1=100
#2=200
#3=#1+#2
...
```

Это означает, что переменной #1 присваивается значение 100, а переменной #2 – значение 200. Переменная #3 будет являться результатом суммы переменной #1 и переменной #2. С таким же успехом можно записать и G-код:

```
...
#25=1
G#25
...
```

Переменной #1 присвоено значение 1. Тогда вторая строка по своей сути будет обозначать код линейной интерполяции G1. С переменными можно производить

различные арифметические и логические операции, что позволяет создавать «умные» программы обработки или различные станочные циклы.

В памяти системы ЧПУ существует область, в которой хранятся значения переменных. Вы можете заглянуть в эту область, если найдете раздел памяти СЧПУ, который обычно называется MACRO или VARIABLES. Присваивать значения переменным можно не только внутри программы, но и непосредственно – вводя значения в регистры этой памяти. Приведу несколько примеров. Можно составить такую программу:

#1=25
#2=30
#3=#2+#1

В этом случае значения присваиваются переменным внутри программы. Чтобы в будущем изменить числовые значения переменных #1 и #2, придется отредактировать программу.

Можно реализовать более удобный вариант, который позволит изменять значения переменных в любой момент, не прибегая к изменению самой программы:

#3=#2+#1

Как видите, переменным #1 и #2 в программе не присвоено никаких значений. Оператор станка может войти в область переменных MACRO и ввести любое числовое значение для любой переменной.

| № переменной | Значение |
|---|---|
| 0 | 0 |
| 1 | 10 |
| 2 | 12 |
| 3 | 0 |
| 4 | 0 |
| … | |
| 700 | 0 |
| 701 | 0 |
| … | |

После того как оператор станка присвоил переменной #1 значение 10, а переменной #2 значение 12 и выполнил программу, значение переменной #3 станет равным 22.

| № переменной | Значение |
|---|---|
| 0 | 0 |
| 1 | 10 |
| 2 | 12 |
| 3 | 22 |
| 4 | 0 |
| … | |
| 700 | 0 |
| 701 | 0 |
| … | |

Все переменные системы ЧПУ можно условно разделить на 4 типа:

- нулевые;
- локальные;
- общие;
- системные.

**Локальные переменные** могут быть использованы внутри макросов для хранения данных. При выключении электропитания локальные переменные обнуляются. У большинства станков с СЧПУ Fanuc нулевой серии локальными являются переменные с номерами от 1 до 33.

**Общие переменные** могут работать внутри различных параметрических программ и макросов. При выключении электропитания некоторые общие переменные обнуляются, а некоторые сохраняют свои значения. У большинства станков с СЧПУ Fanuc нулевой серии общими являются переменные с номерами от 100 до 999.

**Системные переменные** используются для чтения и записи различной системной информации – данных о позиции инструмента, величинах компенсации, времени и др. Номера системных переменных для Fanuc нулевой серии начинаются с 1000.

**Нулевые переменные** всегда равны нулю.

Для выполнения арифметических и логических операций язык Macro B предоставляет набор команд и операторов.

***Таблица 10.1. Основные арифметические и логические команды***

| Функции | Формат |
|---|---|
| Равенство | #a=#b |
| Сложение | #c=#a+#b |
| Вычитание | #c=#a-#b |
| Умножение | #c=#a*#b |
| Деление | #c=#a/#b |
| Синус | #c=SIN[#b] |
| Косинус | #c=COS[#b] |
| Тангенс | #c=TAN[#b] |
| Арктангенс | #c=ATAN[#b] |
| Квадратный корень | #c=SQRT[#b] |
| Абсолютное значение | #c=ABS[#b] |
| Округление | #c=ROUND[#b] |
| ИЛИ (OR) | #c=#a OR #b |
| И (AND) | #c=#a AND #b |

Для управления переменными и для выполнения различных логических операций служат макрокоманды. Макрокоманды языка Macro B похожи на команды Бейсика.

**Команда безусловного перехода GOTO** предназначена для передачи управления определенному кадру программы. Формат команды следующий:

- GOTO N – безусловный переход к кадру N;
- GOTO #A – безусловный переход к кадру, установленному переменной #A.

Пример:

```
...
N10 G01 X100
N20 G01 X-100
N30 GOTO 10
...
```

После выполнения кадра N30 система ЧПУ переходит к кадру N10. Затем снова работает с кадрами N20 и N30 – получается бесконечный цикл.

**Команда условия IF** позволяет выполнять различные действия с условием. После IF указывается некоторое выражение. Если это выражение оказывается справедливым, то выполняется команда (например, команда безусловного перехода), находящаяся в кадре с IF. Если выражение оказывается несправедливым, то команда, находящаяся в кадре с IF, не выполняется, а управление передается следующему кадру.

Формат команды следующий:

```
IF [#a GT #b] GOTO N
```

Пример:

```
...
#1=100
#2=80
N10 G01 X200
N20 IF [#1 GT #2] GOTO 40
N30 G01 X300
N40 M30
...
```

В начале программного примера переменным #1 и #2 присваиваются значения 100 и 80 соответственно. В кадре N20 происходит проверка условия. Если значение переменной #1 больше значения переменной #2, то выполняется команда перехода GOTO к кадру окончания программы N40. В нашем случае выражение считается справедливым, так как 100 больше, чем 80. В результате после выполнения кадра N10 происходит переход к кадру N40, то есть кадр N30 не выполняется.

В этой же программе можно изменить значения переменных:

```
#1=100
#2=120
N10 G01 X200
N20 IF [#1 GT #2] GOTO 40
N30 G01 X300
N40M30
```

Во втором случае условие в кадре N20 не будет справедливым, так как 100 не больше, чем 120. В результате после выполнения кадра N10 не происходит переход к кадру N40, то есть кадр N30 выполняется как обычно.

В выражении [#1 GT #2] используются операторы сравнения. В табл. 10.2 сведены операторы для сравнения переменных языка Macro B.

***Таблица 10.2. Операторы сравнения***

| Оператор | Смысл |
|---|---|
| EQ | Равно (=) |
| NE | Не равно |
| GT | Больше (>) |
| GE | Больше или равно |
| LT | Меньше (<) |
| LE | Меньше или равно |

**Команда WHILE** позволяет повторять различные действия с условием. Пока указанное выражение считается справедливым, происходит выполнение части программы, ограниченной командами DO и END. Если выражение не справедливо, то управление передается кадру, следующему за END.

```
%
O1000
#1=0
#2=1
WHILE [#2 LE 10] DO 1;
#1=#1+#2
#2=#2+1
END 1
M30
%
```

**Макропрограммой** называется программа, которая находится в памяти СЧПУ и содержит различные макрокоманды. Макропрограмму можно вызывать из обычной программы с помощью G-кода, аналогично постоянным циклам. При вызове макропрограммы существует возможность прямой передачи значений для переменных макропрограммы.

Команда G65 предназначена для немодального вызова макропрограммы. Формат для этой команды следующий:

G65 P_L_A_B_

где G65 – команда вызова макропрограммы; P_ – номер вызываемой макропрограммы; L_ – число повторений макропрограммы; A_ и B_ – адреса и значения локальных переменных.

Пример:

G65 P9010 L2 A121 B303 – макропрограмма 9010 вызывается 2 раза, соответствующим локальным переменным присваиваются значения 121 и 303.

Необходимо знать, какой локальной переменной присваивается значение с помощью того или иного адреса. Например, для СЧПУ Fanuc 0-MD будут справедливы следующие зависимости:

***Таблица 10.3. Соответствие адресов локальным переменным***

| Адрес | Переменная |
|---|---|
| A | #1 |
| B | #2 |
| C | #3 |
| D | #7 |
| E | #8 |
| F | #9 |
| H | #11 |
| I | #4 |
| J | #5 |
| K | #6 |
| M | #13 |
| Q | #17 |
| R | #18 |
| S | #19 |
| T | #20 |
| U | #21 |
| V | #22 |
| W | #23 |
| X | #24 |
| Y | #25 |
| Z | #26 |

Теперь можно приступить к созданию несложной, но очень полезной параметрической программы. Довольно часто возникает необходимость в обработке нескольких отверстий, находящихся на некотором радиусе и следующих через определенный угол (рис. 10.7). Чтобы освободить программиста от утомительного переделывания программы в случае изменения радиуса, угла или количества отверстий, создадим такую программу обработки, которая позволит оператору вводить значения радиуса и угла и выполнять операцию сверления по окружности с любыми размерами.

Для сверления отверстий будем использовать стандартный цикл G81. Угол, на котором находятся отверстия, отсчитывается от оси X против часовой стрелки (положительный угол).

Необходимо задать:

- радиус окружности, на которой находятся отверстия;
- начальный угол (угол, на котором находится первое отверстие);
- относительный угол (угол, через который следуют остальные отверстия);
- общее количество отверстий.

Все эти данные должны быть представлены в параметрическом виде, то есть при помощи переменных.

Пусть

#100= радиус окружности, на которой находятся отверстия;

#101= начальный угол;

#102= относительный угол;

#103= общее количество отверстий.

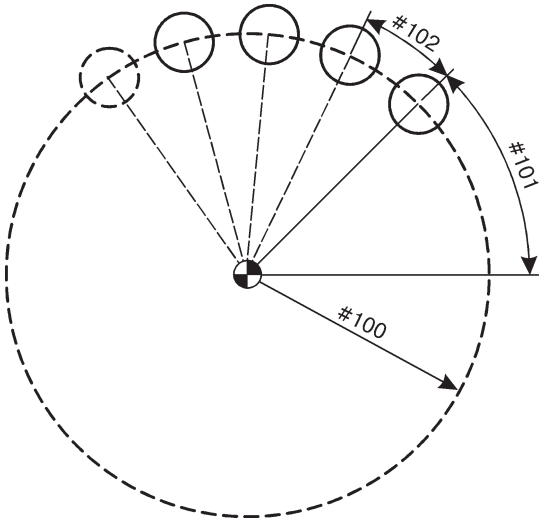



*Рис. 10.7. Создадим параметрическую программу для обработки детали с неизвестными размерами*

Для того чтобы создать параметрическую программу, необходимо придумать алгоритм, позволяющий изменять поведение программы обработки в зависимости от значений указанных переменных. В нашем случае основой УП является стандартный цикл сверления G81. Остается найти закон, по которому описываются координаты центров отверстий при любых первоначальных значениях радиуса, углов и произвольном количестве отверстий.

```
%
O2000
N10 G21 G90 G80 G54 G40 G49 G00
N20 G17
```

Первые кадры программы будут стандартными. Это номер программы, строка безопасности и код G17 выбора плоскости XY.

```
N30 G16
```

Так как координаты центров отверстий задаются с помощью радиуса и угла, то есть в полярной системе координат, то в кадре N30 укажем код G16.

Далее следуют вызов инструмента (сверла), компенсация его длины и включение оборотов шпинделя.

```
N40 T1 M6
N45 G43 HI Z100
N50 S1000 M03
#120=0
```

В кадр N60 поставим цикл сверления G81 и координаты центра первого отверстия. Как вы помните, в случае работы с полярными координатами X обозначает радиус, а Y определяет угол. Значения радиуса и начального угла известны, они устанавливаются переменными #100 (радиус) и #101 (начальный угол). Вводится некоторая переменная #120 с нулевым значением. Эта переменная представляет собой счетчик. Чуть позже вы поймете назначение данной переменной.

N60 G98 G81 X#100 Y#101 Z-5 R0.5 F50

Переменная #103 отвечает за общее количество отверстий. Так как первое отверстие мы уже просверлили, то уменьшим #103 на 1. Таким образом, кадр N70 обеспечивает подсчет оставшихся отверстий. А кадр N75 увеличивает значение переменной #120 на 1.

N70 #103=#103-1
N75 #120=#120+1

Если количество отверстий, которые осталось просверлить, равно нулю, то следует отменить цикл сверления, выключить обороты шпинделя и завершить программу.

N80 IF [#103 EQ 0] GOTO 120

В кадре N80 происходит сравнение значения переменной #103 с нулем. Если переменная #103 равна нулю, то управление передается кадру N120 в конце программы. Если же переменная #103 не равна нулю, то выполняется следующий кадр.

N90 #130=#102*#120
N95#110=#101+#130

Кадр N90 предназначен для определения углового приращения. Новая переменная #110 является суммой #101 (начального угла) и #130 (углового приращения). Кадр N95 обеспечивает расчет угла последующего отверстия.

Затем указывается новый угол для сверления, и управление передается кадру N70.

N100 Y#110
N110 GOTO 70

При помощи кадра N70 образуется замкнутый цикл, который обеспечивает расчет координат центров отверстий и сверление до тех пор, пока значение переменной #103 не будет равно нулю. Если значение #103 станет равным нулю, то управление будет передано кадру N120.

N120 G80
N125 M05
N130 G15
N140 M30
%

Заключительные кадры программы предназначены для отмены постоянного цикла (G80), выключения оборотов шпинделя (M05), выключения режима полярных координат (G15) и завершения программы (M30).

```
%
O2000
N10 G21 G90 G80 G54 G40 G49 G00
N20 G17
N30 G16
N40 T1 M6
N45 G43 H1 Z100
N50 S1000 M03
#120=0
N60 G98 G81 X#100 Y#101 Z-5 R0.5 F50
N70 #103=#103-1
N75 #120=#120+1
N80 IF [#103 EQ 0] GOTO 120
N90 #130=#102*#120
N95 #110=#101+#130
N100 Y#110
N110 GOTO 70
N120 G80
N125 M05
N130 G15
N140 M30
%
```

Любая параметрическая программа должна быть тщательно проверена, прежде чем она попадет на станок. Скорее всего, у вас не получится проверить такую программу при помощи редактора УП и бэкплота, так как в ней присутствуют переменные. Самая надежная проверка в данном случае – это подстановка значений для входных переменных и «раскручивание» алгоритма уже с конкретными числами.

Предположим, что оператор станка получил чертеж детали (рис. 10.8) для обработки отверстий. Он должен установить нулевую точку G54 в центр детали, замерить длину сверла и установить его в шпиндель. Затем следует войти в область переменных MACRO и ввести следующие числовые значения:

| № переменной | Значение |
|---|---|
| ... | ... |
| 100 | 12.5 |
| 101 | 45 |
| 102 | 20 |
| 103 | 4 |
| 104 | 0 |
| 105 | 0 |
| ... | ... |

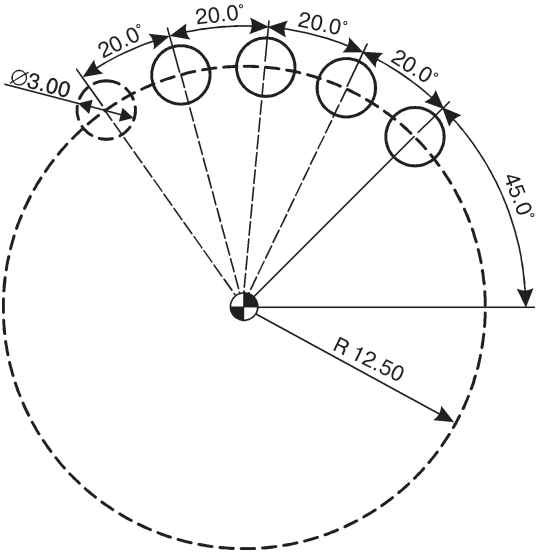



*Рис. 10.8. Вместо переменных на чертеже стоят конкретные размеры и известно количество отверстий*

Для проверки созданной параметрической программы достаточно подставить конкретные значения переменных и, «прокручивая» алгоритм, получить обычную программу.

```
%
O2000
N10 G21 G90 G80 G54 G40 G49 G00
N20 G17
N30 G16
N40 T1 M6
N45 G43 H1 Z100
N50 S1000 M03
#120=0
N60 G98 G81 X12.5 Y45 Z-5 R0.5 F50
N70 #103=#103-1=4-1=3 – количество оставшихся отверстий
N75 #120=#120+1=0+1=1
N80 Переменная #103 не равна 0
N90 #130=#102*#120=20*1=20
N95 #110=#101+#130=45+20=65
N100 Y65
N70 #103=#103-1=3-1=2 – количество оставшихся отверстий
N75 #120=#120+1=1+1=2
N80 Переменная #103 не равна 0
N90 #130=#102*#120=20*2=40
```

```
N95 #110=#101+#130=45+40=85
N100 Y85
N70 #103=#103-1=2-1=1– количество оставшихся отверстий
N75 #120=#120+1=2+1=3
N80 Переменная #103 не равна 0
N90 #130=#102*#120=20*3=60
N95 #110=#101+#130=45+60=105
N100 Y105
N70 #103=#103-1=1-1=0 – количество оставшихся отверстий
N75 #120=#120+1=3+1=4
N80 Переменная #103 равна 0, переход к кадру N120
N120 G80
N125 M05
N130 G15
N140 M30
%
```

Эту же программу можно записать и в привычном виде:

```
%
O2000
N10 G21 G90 G80 G54 G40 G49 G00
N20 G17
N30 G16
N40 T1 M6
N45 G43 H1 Z100
N50 S1000 M03
N60 G98 G81 X12.5 Y45 Z-5 R0.5 F50
N100 Y65
N100 Y85
N100 Y105
N120 G80
N125 M05
N130 G15
N140 M30
%
```

Теперь попробуем создать макропрограмму, которая будет функционировать аналогично постоянному циклу. Для обработки детали, показанной на рис. 10.8, оператор станка должен ввести и отработать следующую команду:

```
G65 P9010 I12.5 A45 B20 H4
```

При этом наша параметрическая программа (с новым номером O9010) уже должна находиться в памяти СЧПУ. Как правило, макропрограммы имеют номера с 9000 и выше, недоступны для свободного редактирования. Команда G65 предназначена для немодального вызова макропрограммы. При этом адреса I, A, B, H в кадре с G65 передают свои числовые значения определенным локальным пере-

менным. Для нахождения соответствия адресов локальным переменным можно воспользоваться табл. 10.3.

Можно подстроить переменные в нашей программе, вставив следующие строки в программу:

```
#100=#4
#101=#1
#102=#2
#103=#11
```

В результате получаем макропрограмму:

```
%
O9010
#100=#4
#101=#1
#102=#2
#103=#11
N10 G21 G90 G80 G54 G40 G49 G00
N20 G17
N30 G16
N40 T1 M6
N45 G43 H1 Z100
N50 S1000 M03
#120=0
N60 G98 G81 X#100 Y#101 Z-5 R0.5 F50
N70 #103=#103-1
N75 #120=#120+1
N80 IF [#103 EQ 0] GOTO 120
N90 #130=#102*#120
N95 #110=#101+#130
N100 Y#110
N110 GOTO 70
N120 G80
N125 M05
N130 G15
N140 M30
%
```

Хотя созданная нами параметрическая программа и не является оптимальной, однако она наглядно демонстрирует широкие возможности этого метода по созданию эффективных УП и различных станочных циклов.

# Краткое изложение главы

- Если УП содержит часто повторяемое действие или работает по определенному шаблону, то использование подпрограмм позволяет упростить УП и сделать ее гораздо меньшей в размере.

- Существуют два вида подпрограмм – внутренние и внешние.
- При помощи L-слова данных определяется, сколько раз нужно вызвать ту или иную подпрограмму. Если подпрограмму нужно вызвать всего один раз, то L-слово данных можно не указывать.
- Очень часто трехкоординатные станки с ЧПУ дополнительно снабжаются поворотным столом или делительной головкой.
- Для управления поворотным столом достаточно в кадр с линейной интерполяцией, позиционированием или постоянным циклом добавить A-слово данных.
- Большинство станочных систем ЧПУ предоставляют специальный язык для параметрического программирования (макропрограммирования).
- В памяти СЧПУ существует область, в которой хранятся значения переменных. Вы можете заглянуть в эту область, если найдете раздел памяти СЧПУ, который обычно называется MACRO или VARIABLES.

# Вопросы

1. Для чего нужны подпрограммы?
2. Чем отличаются внутренние подпрограммы от внешних подпрограмм?
3. Для чего используется код M98?
4. Для чего используется код M99?
5. Можно ли из одной подпрограммы вызвать другую подпрограмму?
6. Назовите особенности программирования 4-ой оси (поворотного стола).
7. Какой символ используется для обозначения переменной в Macro B?
8. Что такое макропрограмма?

# Глава 11
# ПРИМЕРЫ УПРАВЛЯЮЩИХ ПРОГРАММ

## 11.1. Программирование в ISO

### *Пример № 1. Контурная обработка*

Необходимо создать УП для обработки наружного контура детали (рис. 11.1) фрезой диаметром 5 мм без коррекции на радиус инструмента. Глубина фрезерования – 4 мм. Подвод к контуру осуществляется по прямолинейному участку.

| **Управляющая программа** | **Пояснение** |
|---|---|
| % | |
| O0001 | Программа O0001 |
| (PROGRAM NAME – CONTOUR1) | Комментарий – имя программы |
| N100 G21 | Режим ввода метрических данных |
| N102 G0 G17 G40 G49 G80 G90 | Строка безопасности |
| (FREZA D5) | Комментарий – фреза Ф5 мм |
| N104 T1 M6 | Вызов инструмента № 1 |

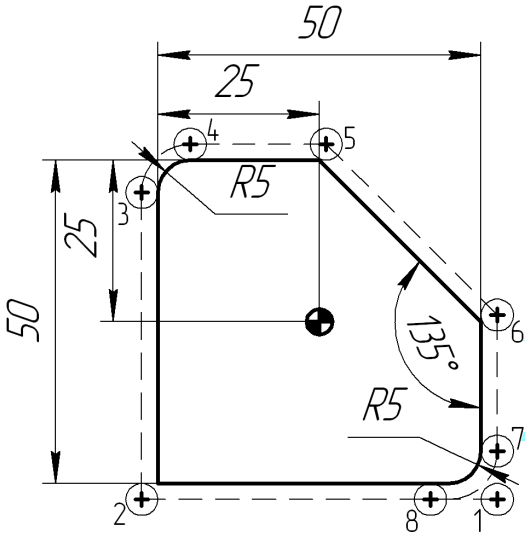



*Рис. 11.1. Контурная обработка*

| | |
|---|---|
| N106 G0 G90 G54 X25. Y-27.5 S2000 M3 | Позиционирование в начальную точку траектории (1), включение оборотов шпинделя 2000 об/мин |
| N108 G43 H1 Z100. | Компенсация длины инструмента № 1 |
| N110 Z10. | Позиционирование в Z10 |
| N112 G1 Z-4. F100. | Фреза опускается до Z-4 на рабочей подаче 100 мм/мин |
| N116 X-27.5 | Линейное перемещение в точку (2) |
| N118 Y20. | Линейное перемещение в точку (3) |
| N120 G2 X-20. Y27.5 R7.5 | Перемещение по дуге в точку (4) |
| N122 G1 X1.036 | Линейное перемещение в точку (5) |
| N124 X27.5 Y1.036 | Линейное перемещение в точку (6) |
| N126 Y-20. | Линейное перемещение в точку (7) |
| N128 G2 X20. Y-27.5 R7.5 | Перемещение по дуге в точку (8) |
| N130 G1 Z6. | Фреза поднимается к Z6 |
| N132 G0 Z100. | Фреза поднимается на ускоренной подаче к Z100 |
| N134 M5 | Останов шпинделя |
| N136 G91 G28 Z0. | Возврат в исходную позицию по Z |
| N138 G28 X0. Y0. | Возврат в исходную позицию по X и Y |
| N140 M30 | Конец программы |
| % | |

## *Пример № 2. Контурная обработка с коррекцией на радиус инструмента*

Необходимо создать УП для обработки наружного контура детали (рис. 11.2) фрезой диаметром 5 мм с коррекцией на радиус инструмента. Глубина фрезерования – 4 мм. Подвод к контуру осуществляется по касательной.

| **Управляющая программа** | **Пояснение** |
|---|---|
| % | |
| O0002 | Программа O0002 |
| (PROGRAM NAME – CONTOUR2 ) | Комментарий – имя программы |
| N100 G21 | Режим ввода метрических данных |
| N102 G0 G17 G40 G49 G80 G90 | Строка безопасности |
| ( FREZA D5 ) | Комментарий – фреза Ф5 мм |
| N104 T1 M6 | Вызов инструмента № 1 |
| N106 G0 G90 G54 X25. Y-35. S2000 M3 | Позиционирование в начальную точку траектории (1), включение оборотов шпинделя 2000 об/мин |
| N108 G43 H1 Z100. | Компенсация длины инструмента № 1 |
| N110 Z10. | Позиционирование в Z10 |

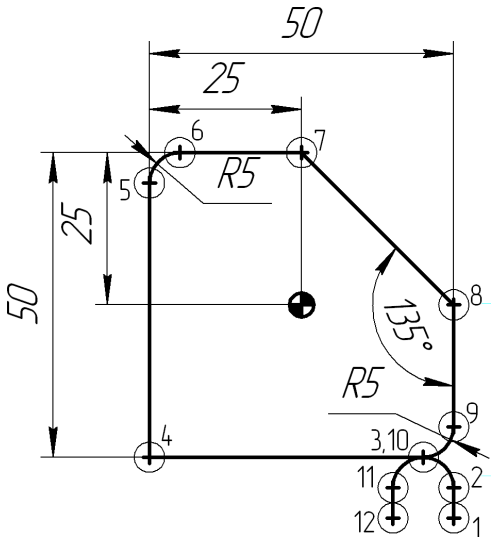



*Рис. 11.2. Контурная обработка с коррекцией*

| | |
|---|---|
| N112 G1 Z-4. F100. | Фреза опускается до Z-4 на рабочей подаче 100 мм/мин |
| N114 G41 D1 Y-30. | Коррекция слева, перемещение в точку (2) |
| N116 G3 X20. Y-25. R5. | Подвод инструмента по касательной к точке (3) |
| N118 G1 X-25. | Линейное перемещение в точку (4) |
| N120 Y20. | Линейное перемещение в точку (5) |
| N122 G2 X-20. Y25. R5. | Перемещение по дуге в точку (6) |
| N124 G1 X0. | Линейное перемещение в точку (7) |
| N126 X25. Y0. | Линейное перемещение в точку (8) |
| N128 Y-20. | Линейное перемещение в точку (9) |
| N130 G2 X20. Y-25. R5. | Перемещение по дуге в точку (10) |
| N132 G3 X15. Y-30. R5. | Отвод инструмента от контура по касательной к точке (11) |
| N134 G1 G40 Y-35. | Линейное перемещение в точку (12) с отменой коррекции |
| N136 Z6. | Фреза поднимается к Z6 |
| N138 G0 Z100. | Фреза поднимается на ускоренной подаче к Z100 |
| N140 M5 | Останов шпинделя |
| N142 G91 G28 Z0. | Возврат в исходную позицию по Z |
| N144 G28 X0. Y0. | Возврат в исходную позицию по X и Y |
| N146 M30 | Конец программы |
| % | |

## *Пример № 3. Контурная обработка*

Необходимо создать УП для чистовой обработки кармана (рис. 11.3) без коррекции на радиус инструмента фрезой диаметром 5 мм. Глубина фрезерования – 2 мм. Подвод к контуру осуществляется по касательной.

| **Управляющая программа** | **Пояснение** |
|---|---|
| % | |
| O0003 | Программа O0003 |
| (PROGRAM NAME – FINISH POCKET) | Комментарий – имя программы |
| N100 G21 | Режим ввода метрических данных |
| N102 G0 G17 G40 G49 G80 G90 | Строка безопасности |
| ( FREZA D5 ) | Комментарий – фреза Ф5 мм |
| N104 T1 M6 | Вызов инструмента № 1 |
| N106 G0 G90 G54 X-2.5 Y-2.5 S1000 M3 | Позиционирование в начальную точку траектории (1), включение оборотов шпинделя |
| N108 G43 H1 Z100. | Компенсация длины инструмента № 1 |
| N110 Z10. | Позиционирование в Z10 |
| N112 G1 Z-2. F100. | Фреза опускается до Z-2 на рабочей подаче 100 мм/мин |
| N114 Y-5. | Линейное перемещение в точку (2) |
| N116 G3 X0. Y-7.5 R2.5 | Подвод инструмента по касательной к точке (3) |
| N118 G1 X10. | Линейное перемещение в точку (4) |
| N120 G3 X17.5 Y0. R7.5 | Перемещение по дуге в точку (5) |

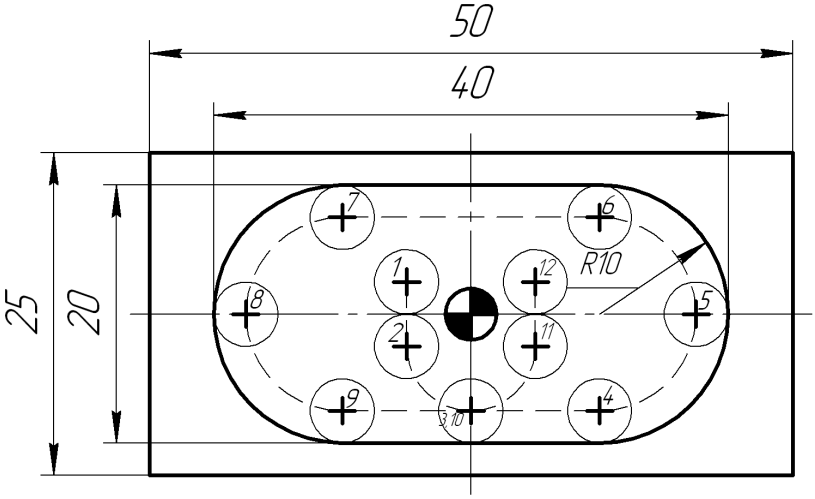



*Рис. 11.3. Чистовая обработка кармана*

| | |
|---|---|
| N122 X10. Y7.5 R7.5 | Перемещение по дуге в точку (6) |
| N124 G1 X-10. | Линейное перемещение в точку (7) |
| N126 G3 X-17.5 Y0. R7.5 | Перемещение по дуге в точку (8) |
| N128 X-10. Y-7.5 R7.5 | Перемещение по дуге в точку (9) |
| N130 G1 X0. | Линейное перемещение в точку (10) |
| N132 G3 X2.5 Y-5. R2.5 | Отвод инструмента по касательной к точке (11) |
| N134 G1 Y-2.5 | Линейное перемещение в точку (12) |
| N136 Z8. | Фреза поднимается к Z8 |
| N138 G0 Z100. | Фреза поднимается на ускоренной подаче к Z100 |
| N140 M5 | Останов шпинделя |
| N146 M30 | Конец программы |
| % | |

## Пример № 4. Контурная обработка с коррекцией на радиус инструмента

Необходимо создать УП для чистовой обработки кармана с коррекцией на радиус инструмента. Глубина фрезерования – 2 мм. Подвод к контуру осуществляется по касательной.

| **Управляющая программа** | **Пояснение** |
|---|---|
| % | |
| O0004 | Программа O0004 |
| (PROGRAM NAME – FINISH POCKET2) | Комментарий – имя программы |
| N100 G21 | Режим ввода метрических данных |

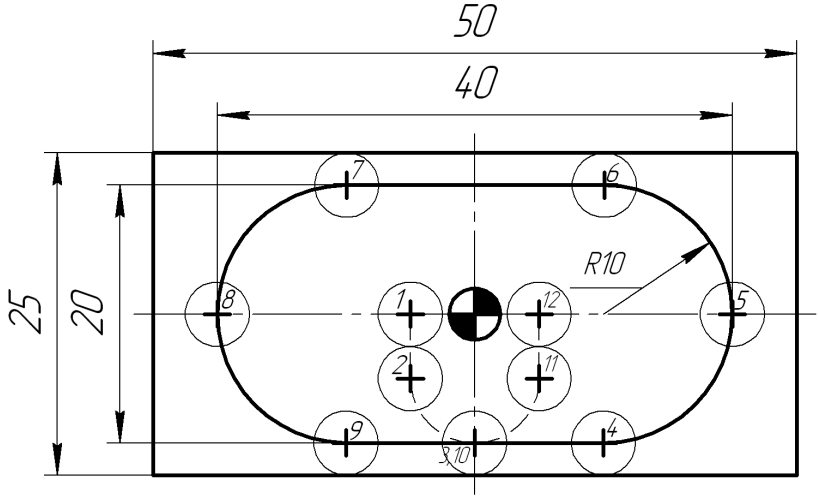



*Рис. 11.4. Чистовая обработка кармана с коррекцией*

| | |
|---|---|
| N102 G0 G17 G40 G49 G80 G90 | Строка безопасности |
| N104 T1 M6 | Вызов инструмента № 1 |
| N106 G0 G90 G54 X-2.5 Y-5. S1000 M3 | Позиционирование в начальную точку траектории (1), включение оборотов шпинделя |
| N108 G43 H1 Z100. | Компенсация длины инструмента №1 |
| N110 Z10. | Позиционирование в Z10 |
| N112 G1 Z-2. F100. | Фреза опускается до Z-2 на рабочей подаче 100 мм/мин |
| N114 G41 D1 Y-7.5 | Коррекция слева, перемещение в точку (2) |
| N116 G3 X0. Y-10. R2.5 | Подвод инструмента по касательной к точке (3) |
| N118 G1 X10. | Линейное перемещение в точку (4) |
| N120 G3 X20. Y0. R10. | Перемещение по дуге в точку (5) |
| N122 X10. Y10. R10. | Перемещение по дуге в точку (6) |
| N124 G1 X-10. | Линейное перемещение в точку (7) |
| N126 G3 X-20. Y0. R10. | Перемещение по дуге в точку (8) |
| N128 X-10. Y-10. R10. | Перемещение по дуге в точку (9) |
| N130 G1 X0. | Линейное перемещение в точку (10) |
| N132 G3 X2.5 Y-7.5 R2.5 | Отвод инструмента по касательной к точке (11) |
| N134 G1 G40 Y-5. | Линейное перемещение в точку (12) с отменой коррекции |
| N136 Z8. | Фреза поднимается к Z8 |
| N138 G0 Z100. | Фреза поднимается на ускоренной подаче к Z100 |
| N140 M5 | Останов шпинделя |
| N146 M30 | Конец программы |
| % | |

## *Пример № 5. Фрезерование прямоугольного кармана*

Необходимо создать УП для обработки прямоугольного кармана фрезой диаметром 10 мм. Глубина фрезерования – 1 мм.

| **Управляющая программа** | **Пояснение** |
|---|---|
| % | |
| O0005 | Программа O0005 |
| (PROGRAM NAME – ROUGH POCKET) | Комментарий – имя программы |
| N100 G21 | Режим ввода метрических данных |
| N102 G0 G17 G40 G49 G80 G90 | Строка безопасности |
| N104 T1 M6 | Вызов инструмента № 1 |

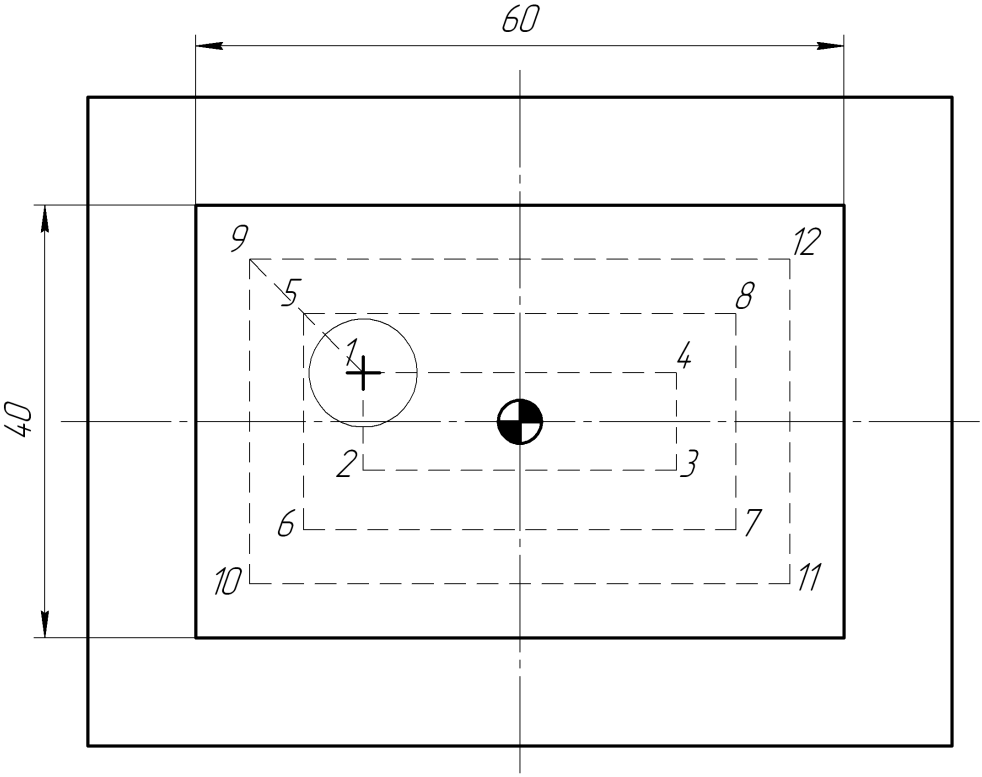



*Рис. 11.5. Черновое фрезерование прямоугольного кармана*

| | |
|---|---|
| N106 G0 G54 X-13.75 Y3.75 S1000 M3 | Позиционирование в начальную точку траектории (1), включение оборотов шпинделя |
| N108 G43 H1 Z100. | Компенсация длины инструмента № 1 |
| N110 Z10. | Позиционирование в Z10 |
| N112 G1 Z-1. F100. | Фреза опускается до Z-1 на рабочей подаче 100 мм/мин |
| N114 Y-3.75 | Линейное перемещение в точку (2) |
| N116 X13.75 | Линейное перемещение в точку (3) |
| N118 Y3.75 | Линейное перемещение в точку (4) |
| N120 X-13.75 | Линейное перемещение в точку (1) |
| N122 X-17.5 Y7.5 | Линейное перемещение в точку (5) |
| N124 Y-7.5 | Линейное перемещение в точку (6) |
| N126 X17.5 | Линейное перемещение в точку (7) |
| N128 Y7.5 | Линейное перемещение в точку (8) |
| N130 X-17.5 | Линейное перемещение в точку (5) |

| | |
|---|---|
| N132 X-25. Y15. | Линейное перемещение в точку (9) |
| N134 Y-15. | Линейное перемещение в точку (10) |
| N136 X25. | Линейное перемещение в точку (11) |
| N138 Y15. | Линейное перемещение в точку (12) |
| N140 X-25. | Линейное перемещение в точку (9) |
| N142 Z9. | Фреза поднимается к Z9 |
| N144 G0 Z100. | Фреза поднимается на ускоренной подаче к Z100 |
| N146 M5 | Останов шпинделя |
| N152 M30 | Конец программы |
| % | |

## *Пример № 6. Фрезерование круглого кармана*

Необходимо создать УП для обработки круглого кармана фрезой диаметром 10 мм. Глубина – 0.5 мм.

| **Управляющая программа** | **Пояснение** |
|---|---|
| % | |
| O0000 | Программа O0006 |
| (PROGRAM NAME – N6) | Комментарий – имя программы |
| N100 G21 | Режим ввода метрических данных |
| N102 G0 G17 G40 G49 G80 G90 | Строка безопасности |

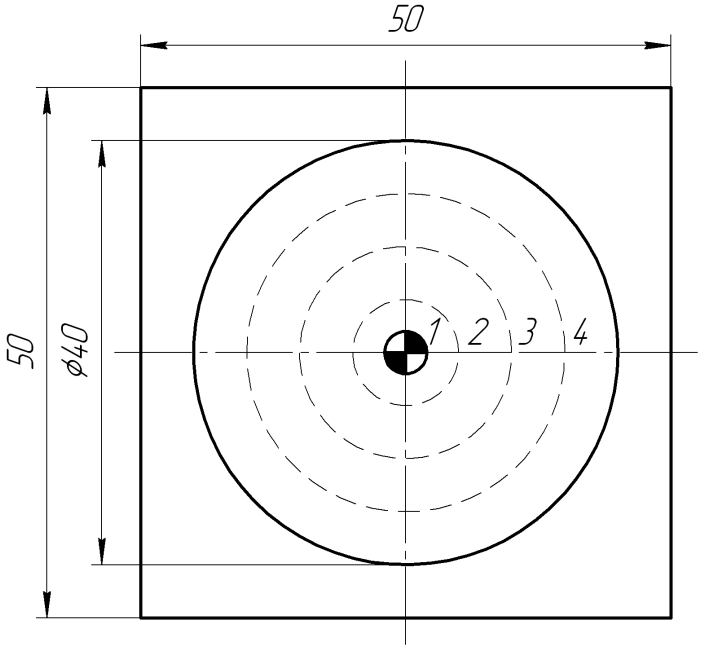



*Рис. 11.6. Черновое фрезерование круглого кармана*

| | |
|---|---|
| N104 T1 M6 | Вызов инструмента № 1 |
| N106 G0 G90 G54 X0. Y0. S1000 M3 | Позиционирование в начальную точку траектории (1), включение оборотов шпинделя |
| N108 G43 H1 Z100. | Компенсация длины инструмента № 1 |
| N110 Z10. | Позиционирование в Z10 |
| N112 G1 Z-.5 F100. | Фреза опускается до Z-0.5 на рабочей подаче 100 мм/мин |
| N120 X5. F200 | Перемещение в точку (1) |
| N122 G3 X-5. R5. | Круговое перемещение по 1-ой «орбите» |
| N124 X5. R5. | ... |
| N126 G1 X10. | Перемещение в точку (2) |
| N128 G3 X-10. R10. | Круговое перемещение по 2-ой «орбите» |
| N130 X10. R10. | ... |
| N132 G1 X15. | Перемещение в точку (3) |
| N134 G3 X-15. R15. | Круговое перемещение по 3-ей «орбите» |
| N136 X15. R15. | ... |
| N138 G1 Z10 F300. | Фреза поднимается к Z10 |
| N140 G0 Z100. | Фреза поднимается на ускоренной подаче к Z100 |
| N142 M5 | Останов шпинделя |
| N148 M30 | Конец программы |
| % | |

# 11.2. Программирование для Heidenhain

При работе со стойкой ЧПУ фирмы Heidenhain существуют два варианта программирования: с помощью традиционных G-, M-кодов или на специальном языке Heidenhain. Несмотря на популярность ИСО 7 бит, многие станочники-операторы предпочитают второй вариант, который дает возможность вести работу в диалоговом режиме и имеет эффективные отличия от ISO-кода. Главная особенность диалогового языка Heidenhain заключается в замене привычных G- и M-кодов на специальные слова данных. Постоянные циклы этого языка также могут оказаться непривычными для вас.

Запомните некоторые принципы диалогового языка Heidenhain:

- **Линейная интерполяция** указывается при помощи L-слова данных (LINE – линия).
  L X+10 Y+40 F100 – перемещение в точку (10; 40) с подачей 100 мм/мин.
- **Ускоренный ход** обозначается F MAX.
- **Круговая интерполяция** (координаты конечной точки дуги) определяется с помощью C-слова данных (CIRCLE – окружность), при этом координаты центра дуги обозначаются CC (CIRCLE CENTER). Если образующая-

ся дуга больше 180°, то в кадре круговой интерполяции указывается DR–, а если меньше 180° – то DR+.
122 CC X-20 Y+20 – координаты центра дуги.
124 C X-20 Y+27.5 DR- – перемещение по дуге в точку (–20; 27.5).

- **Коррекция на радиус инструмента** слева указывается при помощи RL-, а справа – при помощи RR-слова данных.
  116 L Y-30 RL – линейное перемещение и включение коррекции слева.

Особое отличие – описание стандартных циклов, которые указываются набором из нескольких кадров (см. пример № 3 этого раздела).

Обычно при работе со стойками Heidenhain не нужно программировать компенсацию длины инструмента, но нужно выполнить его инициализацию, то есть активацию геометрических характеристик.

TOOL DEF 3 – инициализация данных инструмента № 3.
TOOL CALL 3 – вызов инструмента № 3.

## *Пример № 1. Контурная обработка*

Необходимо создать УП для обработки наружного контура детали (рис. 11.1) фрезой диаметром 5 мм без коррекции на радиус инструмента. Глубина фрезерования равна 4 мм. Подвод к контуру осуществляется по прямолинейному участку.

| **Управляющая программа** | **Пояснение** |
|---|---|
| 100 BEGIN PGM 0 MM | Номер программы |
| ; PROGRAM NAME – SAMP1H | Комментарий – имя программы |
| 102 TOOL DEF 1 | Инициализация инструмента № 1 |
| ; FREZA D5 | Комментарий – фреза Ф5 |
| 104 TOOL CALL 1 Z S2000 | Вызов инструмента № 1, S = 2000 об/мин |
| 106 L Z+100 R0 F MAX | Позиционирование в Z100 |
| 108 L X+25 Y-27.5 R0 F MAX M03 | Позиционирование в начальную точку траектории (1), включение оборотов |
| 112 L Z+10 F MAX | Позиционирование в Z10 |
| 114 L Z-4 F100 | Фреза опускается до Z-4 на рабочей подаче 100 мм/мин |
| 118 L X-27.5 | Линейное перемещение в точку (2) |
| 120 L Y+20 | Линейное перемещение в точку (3) |
| 122 CC X-20 Y+20 | Координаты центра дуги |
| 124 C X-20 Y+27.5 DR- | Перемещение по дуге в точку (4) |
| 126 L X+1.0355 | Линейное перемещение в точку (5) |
| 128 L X+27.5 Y+1.0355 | Линейное перемещение в точку (6) |
| 130 L Y-20 | Линейное перемещение в точку (7) |
| 132 CC X+20 Y-20 | Координаты центра дуги |
| 134 C X+20 Y-27.5 DR- | Перемещение по дуге в точку (8) |
| 136 L Z+6 | Фреза поднимается к Z6 |
| 138 L Z+100 F MAX | Фреза поднимается на ускоренной подаче к Z100 |

| | |
|---|---|
| 142 M2 | Завершение программы |
| 144 END PGM 0 MM | |

## Пример № 2. Контурная обработка с коррекцией на радиус инструмента

Необходимо создать УП для обработки наружного контура детали (рис. 11.2) фрезой диаметром 5 мм с коррекцией на радиус инструмента. Глубина фрезерования – 4 мм. Подвод к контуру осуществляется по касательной.

| **Управляющая программа** | **Пояснение** |
|---|---|
| 100 BEGIN PGM 0 MM | Номер программы |
| ; PROGRAM NAME – SAMPLE2H | Комментарий – имя программы |
| 102 TOOL DEF 1 | Инициализация инструмента № 1 |
| ; FREZA D5 | Комментарий – фреза Ф5 |
| 104 TOOL CALL 1 Z S2000 | Вызов инструмента № 1, S = 2000 об/мин |
| 106 L Z+100 R0 F MAX | Позиционирование в Z100 |
| 108 L X+25 Y-35 R0 F MAX M03 | Позиционирование в начальную точку траектории (1), включение оборотов |
| 112 L Z+10 F MAX | Позиционирование в Z10 |
| 114 L Z-4 F100 | Фреза опускается до Z-4 на рабочей подаче 100 мм/мин |
| 116 L Y-30 RL | Коррекция слева, перемещение в точку (2) |
| 118 CC X+20 Y-30 | Координаты центра дуги |
| 120 C X+20 Y-25 DR+ | Подвод инструмента по касательной к точке (3) |
| 122 L X-25 | Линейное перемещение в точку (4) |
| 124 L Y+20 | Линейное перемещение в точку (5) |
| 126 CC X-20 Y+20 | Координаты центра дуги |
| 128 C X-20 Y+25 DR- | Перемещение по дуге в точку (6) |
| 130 L X+0 | Линейное перемещение в точку (7) |
| 132 L X+25 Y+0 | Линейное перемещение в точку (8) |
| 134 L Y-20 | Линейное перемещение в точку (9) |
| 136 CC X+20 Y-20 | Координаты центра дуги |
| 138 C X+20 Y-25 DR- | Перемещение по дуге в точку (10) |
| 140 CC X+20 Y-30 | Координаты центра дуги |
| 142 C X+15 Y-30 DR+ | Отвод инструмента от контура по касательной к точке (11) |
| 144 L Y-35 R0 | Линейное перемещение в точку (12) с отменой коррекции |
| 146 L Z+6 F100 | Фреза поднимается к Z6 |
| 148 L Z+100 F MAX | Фреза поднимается на ускоренной подаче к Z100 |
| 152 M2 | Завершение программы |
| 154 END PGM 0 MM | |

## *Пример № 3. Сверление 7 отверстий диаметром 3 мм и глубиной 6,5 мм с помощью постоянного цикла Heidenhain (см. рис. 8.8)*

| **Код программы** | **Описание** |
|---|---|
| 00 BEGIN PGM 0 MM | Номер программы |
| ; PROGRAM NAME – 1 | Комментарий – имя программы |
| 102 TOOL DEF 1 | Инициализация инструмента № 1 |
| ; DRILL 7 HOLES | Комментарий – операция |
| 104 TOOL CALL 1 Z S1000 | Вызов сверла диаметром 3 мм |
| 106 L Z+100 R0 F MAX | Ускоренное перемещение к Z100. |
| 108 L X+5 Y+5 R0 F MAX M03 | Перемещение к отверстию № 1 |
| 110 L Z+10 R0 F MAX | Ускоренное перемещение к Z10. |
| 112 CYCL DEF 200 DRILLING ~ | Цикл № 200 для сверления |
| Q200=10. ;SET-UP CLEARANCE ~ | Исходная плоскость |
| Q201=-6.5 ;DEPTH ~ | Глубина сверления |
| Q206=45 ;FEED RATE FOR PLUNGING ~ | Рабочая подача |
| Q202=6.5 ;PLUNGING DEPTH ~ | Приращение по глубине |
| Q210= 0 ;DWELL TIME AT TOP ~ | Время выдержки наверху |
| Q203= +0 ;SURFACE COORDINATE ~ | Уровень припуска |
| Q204= 100 ;2ND SET-UP CLEARANCE ~ | Плоскость отвода |
| Q211=0. ;DWELL TIME AT DEPTH | Время выдержки на дне |
| 116 L Z10. R F MAX M9 M99 | |
| 118 L X+10 R0 F MAX M99 | Сверление отверстия № 2 |
| 120 L X+15 R0 F MAX M99 | Сверление отверстия № 3 |
| 122 L X+20 R0 F MAX M99 | Сверление отверстия № 4 |
| 124 L X+5 Y+10 R0 F MAX M99 | Сверление отверстия № 5 |
| 126 L X+10 R0 F MAX M99 | Сверление отверстия № 6 |
| 128 L X+30 Y+20 R0 F MAX M99 | Сверление отверстия № 7 |
| 130 L Z+100 F MAX | Ускоренное перемещение к Z100. |
| 134 M2 | Конец программы |
| 136 END PGM 0 MM | |

# Глава 12
# CAD/CAM

## 12.1. Методы программирования

Существуют три метода программирования обработки для станков с ЧПУ:

- ручное программирование;
- программирование на пульте УЧПУ;
- программирование при помощи CAD/CAM-системы.

Прочитав предыдущие главы, вы, наверное, заметили, что **ручное программирование** является довольно утомительным занятием. Однако все технологи-программисты должны иметь хорошее представление о технике ручного программирования независимо от того, как на самом деле они работают. Это как начальные классы в школе, обучение в которых дает нам базу для последующего образования. В нашей стране существует еще немало предприятий, на которых используется метод ручного программирования. Действительно, если завод имеет несколько станков с ЧПУ, а изготавливаемые детали просты, то грамотный программист способен довольно успешно работать и без средств автоматизации собственного труда.

Метод **программирования на пульте УЧПУ** приобрел особую популярность лишь в последние годы. Это связано с техническим развитием систем ЧПУ, улучшением их интерфейса и возможностей. В этом случае программы создаются и вводятся прямо на стойке ЧПУ с использованием клавиатуры и дисплея. Современные системы ЧПУ действительно позволяют работать очень эффективно. Например, оператор станка может произвести верификацию УП или выбрать требуемый постоянный цикл при помощи специальных пиктограмм и вставить его в код УП. Некоторые системы ЧПУ предлагают диалоговый язык программирования, который значительно упрощает процесс создания УП, делает «общение» с ЧПУ удобным для оператора.

Третий метод – **программирование при помощи CAD/CAM-системы** – позволяет «поднять» процесс написания программ обработки на более высокий уровень. Работая с CAD/CAM-системой, технолог-программист избавляет себя от трудоемких математических расчетов и получает инструменты, значительно повышающие скорость написания УП.

## 12.2. Что такое CAD и CAM?

Сегодня для достижения успеха на рынке промышленное предприятие вынуждено работать над сокращением срока выпуска продукции, снижением ее себестоимости и повышением качества. Стремительное развитие компьютерных и информационных технологий привело к появлению CAD/CAM/CAE-систем, которые являются наиболее продуктивными инструментами для решения этих задач.

Под **CAD-системами** (computer-aided design – компьютерная поддержка проектирования) понимают программное обеспечение, которое автоматизирует труд инженера-конструктора и позволяет решать задачи проектирования изделий и оформления технической документации при помощи персонального компьютера.

**CAM-системы** (computer-aided manufacturing – компьютерная поддержка изготовления) автоматизируют расчеты траекторий перемещения инструмента для обработки на станках с ЧПУ и обеспечивают выдачу управляющих программ с помощью компьютера.

**CAE-системы** (computer-aided engineering – компьютерная поддержка инженерных расчетов) предназначены для решения различных инженерных задач, например для расчетов конструктивной прочности, анализа тепловых процессов, расчетов гидравлических систем и механизмов.

Развитие CAD/CAM/CAE-систем продолжается уже несколько десятилетий. За это время произошло некоторое разделение, или, точнее, «ранжирование» систем на уровни. Появились системы верхнего, среднего и нижнего уровней. Системы **верхнего уровня** обладают огромным набором функций и возможностей, но с ними тяжелее работать. Системы **нижнего уровня** имеют довольно ограниченные функции, но очень просты в изучении. Системы **среднего уровня** – это «золотая середина». Они обеспечивают пользователя достаточными для решения большинства задач инструментами, при этом не сложны для изучения и работы.

## 12.3. Общая схема работы с CAD/CAM-системой

**Этап 1.** В CAD-системе создается электронный чертеж или 3D-модель детали.

**Этап 2.** Электронный чертеж или 3D-модель детали импортируется в CAM-систему. Технолог-программист определяет поверхности и геометрические элементы, которые необходимо обработать, выбирает стратегию обработки, режущий инструмент и назначает режимы резания. Система производит расчеты траекторий перемещения инструмента.

**Этап 3.** В CAM-системе производится верификация (визуальная проверка) созданных траекторий. Если на этом этапе обнаруживаются какие-либо ошибки, то программист может легко их исправить, вернувшись к предыдущему этапу.

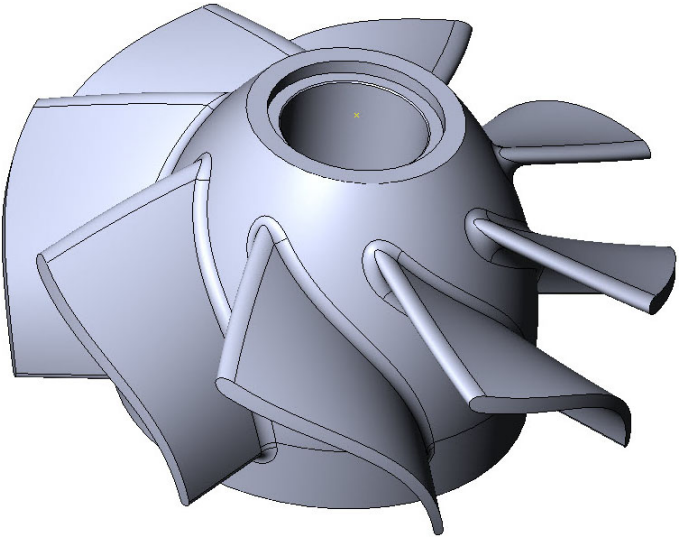


*Рис. 12.1. Трехмерная модель детали*

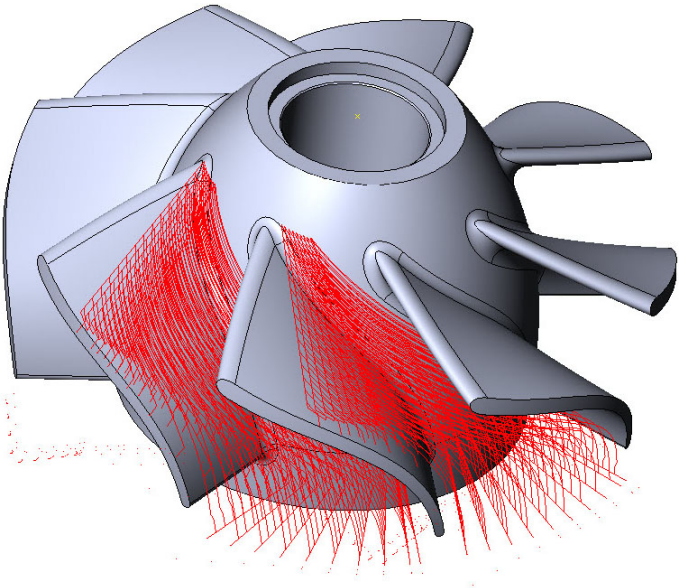


*Рис. 12.2. CAM-система рассчитала траекторию*

**Этап 4.** Финальным продуктом CAM-системы является код управляющей программы. Этот код формируется при помощи постпроцессора, который форматирует УП под требования конкретного станка и системы ЧПУ.

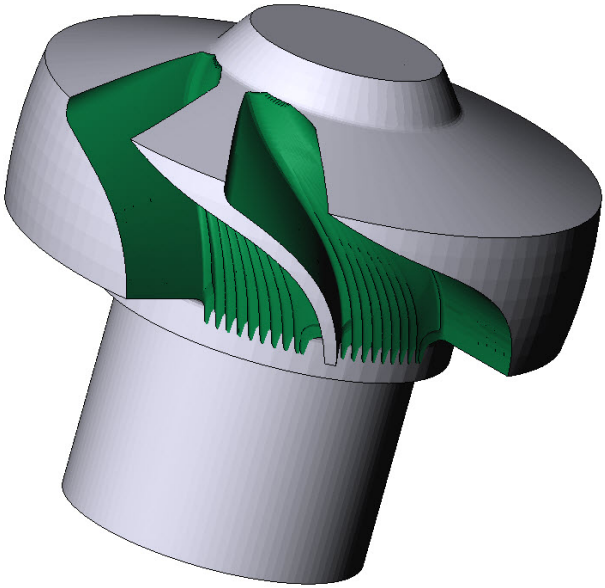


*Рис. 12.3. Результат верификации траектории*

# 12.4. Виды моделирования

Существуют несколько вариантов геометрического представления детали в CAD-системе. Выбор того или иного варианта зависит от возможностей системы и от необходимости его применения для создания управляющей программы.

Еще не так давно основным инструментом инженера-конструктора был кульман. С появлением первых персональных компьютеров началась настоящая революция в области автоматизации проектирования. Инженеры-конструкторы сразу же оценили преимущества «плоских чертилок». Даже самая простая CAD-система **для двумерного проектирования** позволяет быстро создавать различные геометрические элементы, копировать фрагменты, автоматически наносить штриховку и проставлять размеры.

Основными инструментами при плоском проектировании являются линии, дуги и кривые. При помощи операций продления, обрезки и соединения геометрических элементов происходит создание «электронного чертежа». Для полноценной работы с плоской графикой в CAM-системе необходима дополнительная информация о глубине геометрии.

**Каркасная модель** представляет геометрию детали в трехмерном пространстве, описывая положение ее контуров и граней. Каркасная модель, в отличие от плоского электронного чертежа, предоставляет CAM-системе частичную информацию о глубине геометрии.

С развитием автомобильной и авиационной промышленности и необходимостью аналитического описания деталей сложной формы на ПК сформировались

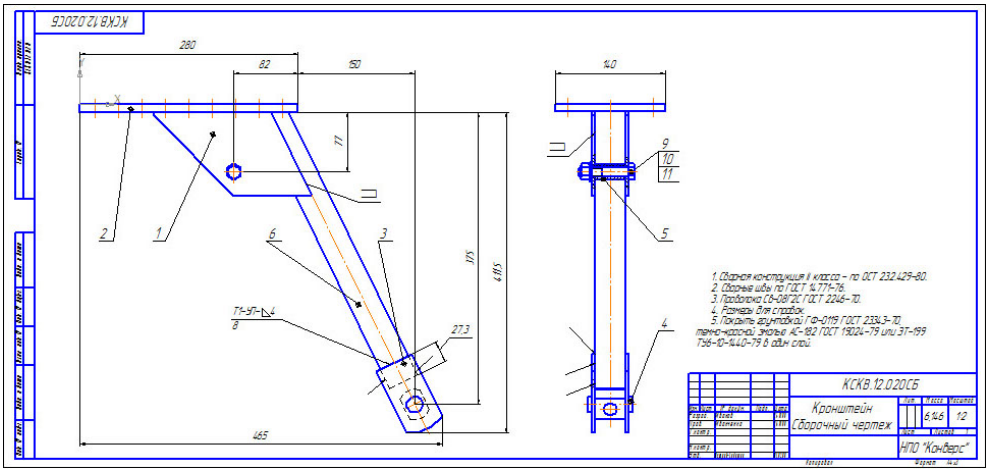



*Рис. 12.5. Электронный чертеж (2D-геометрия)*

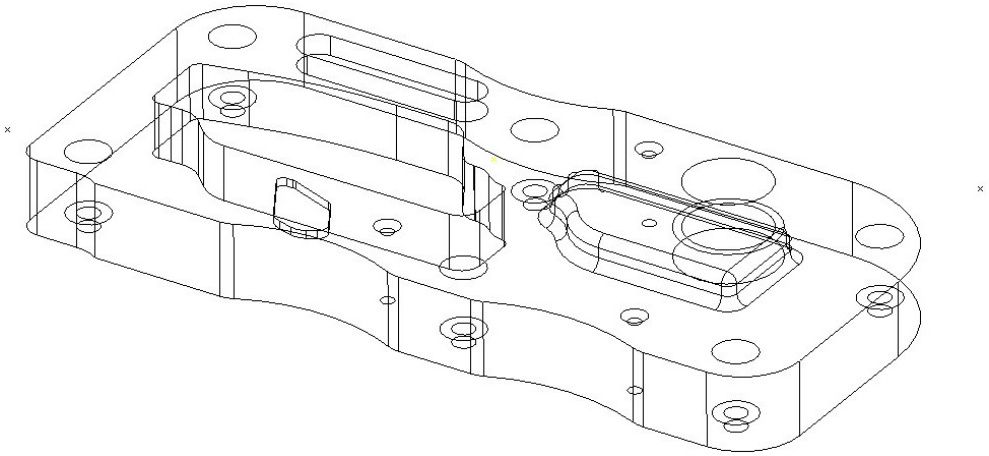


*Рис. 12.6. Каркасная модель*

основные предпосылки для перехода от плоского к объемному моделированию. Объемная, или 3D-модель, предназначена для однозначного определения геометрии всей детали.

Системы объемного моделирования базируются на методах построения поверхностей и твердотельных моделей на основе плоских и неплоских эскизов. Эскиз, в свою очередь, состоит из простых геометрических элементов – линий, дуг и кривых. Инженер-конструктор принимает в качестве эскизов сечения, виды и осевые линии деталей.

**Поверхностная модель** очень похожа на каркасную. Представьте себе, что между гранями каркасной модели натянута тонкая ткань. Это и будет поверхност-

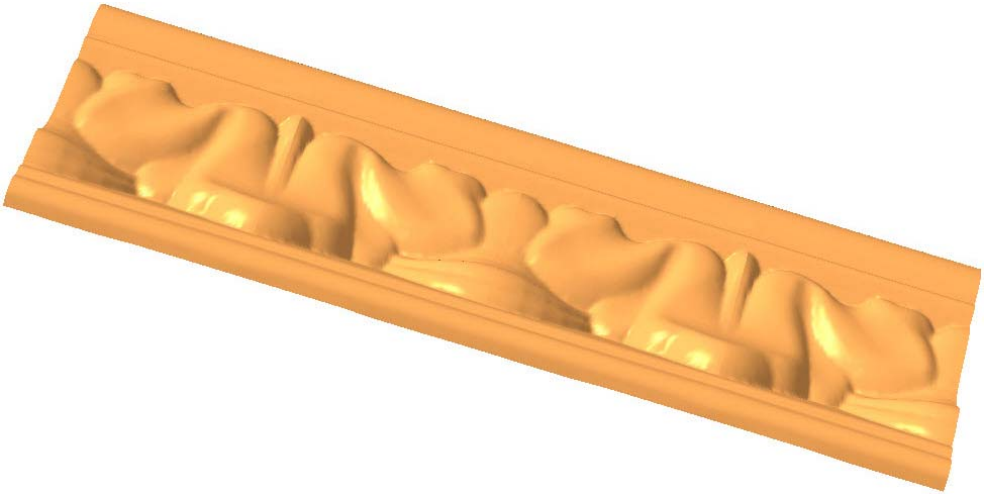


*Рис. 12.7. Поверхностная модель*

ной моделью. Таким образом, любое изделие может быть представлено в виде набора ограничивающих поверхностей.

В настоящее время поверхностные модели широко используются для работы с САМ-системами, особенно когда речь идет об инструментальном производстве.

**При твердотельном способе моделирования** основными инструментами являются тела, созданные на основе эскизов. Для построения твердого тела используются такие операции, как выдавливание, вырезание и вращение эскиза. Булевы

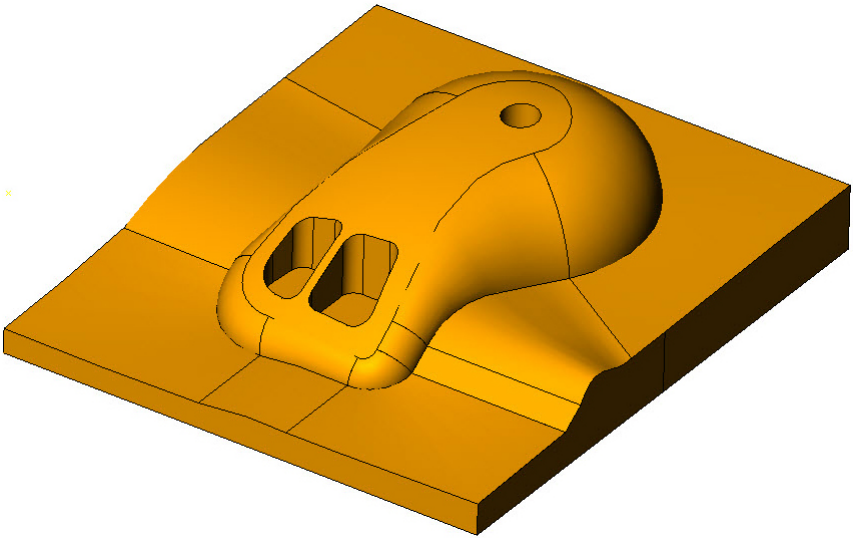


*Рис. 12.8. Твердотельная модель*

операции позволяют складывать, вычитать и объединять различные твердые тела для создания 3D-модели детали. В отличие от поверхностных моделей, твердотельная модель не является пустой внутри. Она обладает некоторой математической плотностью и массой. На сегодняшний день твердотельные модели – это самая популярная основа для расчета траекторий в CAM-системе.

Одним из главных преимуществ этого способа является так называемая параметризация. **Параметризация** означает, что в любой момент вы можете изменить размеры и характеристики твердого тела, просто изменив числовые значения соответствующих параметров.

Современная CAD/CAM-система должна обладать инструментами для создания как поверхностных, так и твердотельных моделей.

## 12.5. Уровни CAM-системы

CAM-система предназначена для автоматического создания управляющих программ на основе геометрической информации, подготовленной в CAD-системе. **Главные преимущества, которые получает технолог при взаимодействии с системой, заключаются в наглядности работы, удобстве выбора геометрии, высокой скорости расчетов, возможности проверки и редактирования созданных траекторий.**

Различные CAM-системы могут отличаться друг от друга областью применения и возможностями. К примеру, существуют системы для токарной, фрезерной, электроэрозионной обработки, деревообработки и гравировки. Несмотря на то что большинство современных CAD/CAM-систем умеют создавать УП для любого типа производства, такое разделение по областям применения остается актуальным. Если предприятию нужна фрезерная обработка, то оно приобретает модуль фрезерования. Если же нужна только токарная обработка, то достаточно приобрести токарный модуль этой же системы. Модульность построения CAM-систем является частью маркетинговой политики разработчиков и позволяет предприятию-пользователю экономить значительные средства для приобретения только необходимых конструкторско-технологических возможностей.

В свою очередь, модули системы отличаются определенным уровнем возможностей. Обычно для фрезерной обработки разработчики вводят следующие уровни:

- 2.5-осевая обработка.
  На этом уровне система позволяет рассчитывать траектории для простого двухкоординатного фрезерования и обработки отверстий;
- трехосевая обработка с позиционированием 4-ой оси.
  На этом уровне вы сможете работать с 3D-моделями. Система способна генерировать УП для объемной обработки;
- многоосевая обработка.
  В этом случае система предназначена для работы с самым современным оборудованием и способна создавать УП для 5-осевого фрезерования самых сложных деталей.

Чем выше уровень модуля, тем большими возможностями он обладает. Естественно, что для разработки алгоритмов 5-координатной обработки требуются большие инвестиции (как финансовые, так и интеллектуальные), чем для разработки алгоритмов 3-координатной обработки. Следовательно, и стоимость модулей будет разной. Если у вашего предприятия нет оборудования для 5-координатной обработки, то нет смысла приобретать самый дорогостоящий модуль.

## 12.6. Геометрия и траектория

Прежде чем начать работу с CAD/CAM-системой, вы должны понять, что геометрия детали, изготовленной на станке с ЧПУ, может отличаться от истинной геометрии CAD-модели. Несомненно, что 3D-модель служит базой для расчета траекторий, но готовая деталь является результатом работы CAM-системы и станка с ЧПУ, которые по-своему интерпретируют исходную геометрическую информацию.

Возьмем эллипс, который может быть создан в любой CAD-системе очень просто – достаточно пары кликов мышкой. Однако станок с ЧПУ не способен напрямую описать эллипс, ведь он умеет перемещать инструмент только по прямой или дуге. CAM-система знает это и решает возникшую проблему при помощи аппроксимации эллипса прямыми линиями с определенной точностью. В результате траекторию эллипса можно получить и на станке с ЧПУ, но уже при помощи линейной интерполяции.

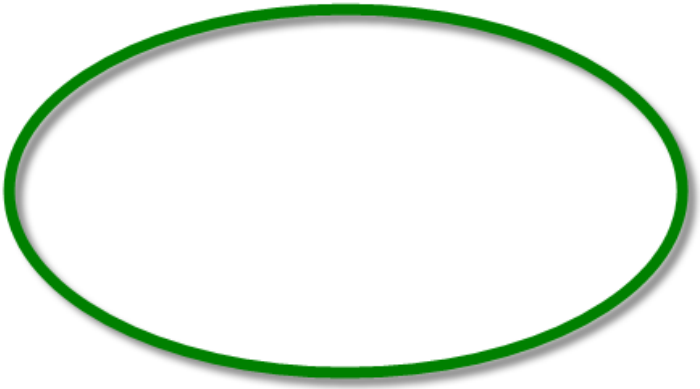


*Рис. 12.9. Эллипс легко построить в любой CAD-системе*

Программист сам устанавливает ограничивающую зону для аппроксимации, то есть определяет, с какой точностью нужно «приблизиться» к исходной геометрии. Чем выше задана точность, тем больше будет произведено отдельных сегментов и тем больший размер будет иметь программа обработки. Особенно ярко этот эффект проявляется при обработке 3D-моделей.

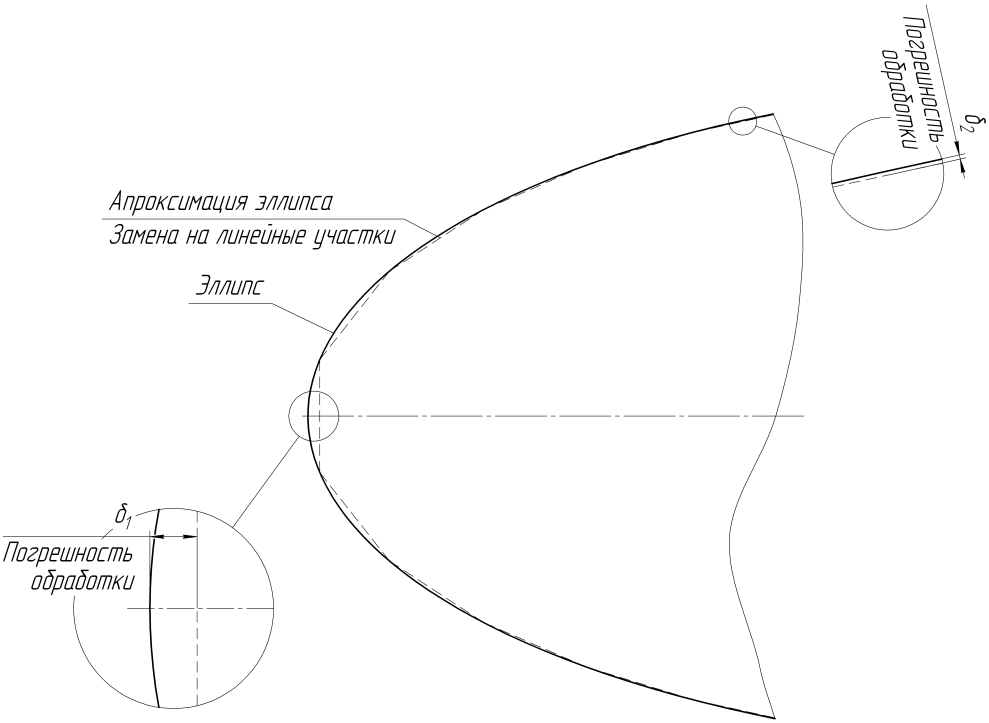



*Рис. 12.10. Линейная аппроксимация эллипса в САМ-системе выполняется с заданной точностью*

# 12.7. Алгоритм работы в САМ-системе

Несмотря на то что сегодня существует не один десяток САМ-систем и все они отличаются интерфейсом и возможностями, порядок работы с ними примерно одинаков.

## *12.7.1. Выбор геометрии*

Самым первым действием технолога-программиста является выбор геометрических элементов, подлежащих обработке. Такие геометрические элементы **называются рабочими**, или **обрабатываемыми**. Это могут быть точки, линии, поверхности, грани и ребра 3D-моделей.

Современные САМ-системы позволяют контролировать перемещение инструмента не только относительно рабочих элементов, но и относительно других геометрических элементов. Можно выбрать **контролируемые** элементы, которых инструмент не должен касаться ни при каких условиях.

Некоторые CAM-системы **требуют определения геометрии заготовки** уже на начальном этапе проектирования обработки. Это означает, что система «видит» заготовку и рассчитывает траектории, исходя из действительного припуска. Заготовка может быть определена несколькими способами:

- в виде цилиндра;
- в виде параллелепипеда (куба);
- произвольной формы.

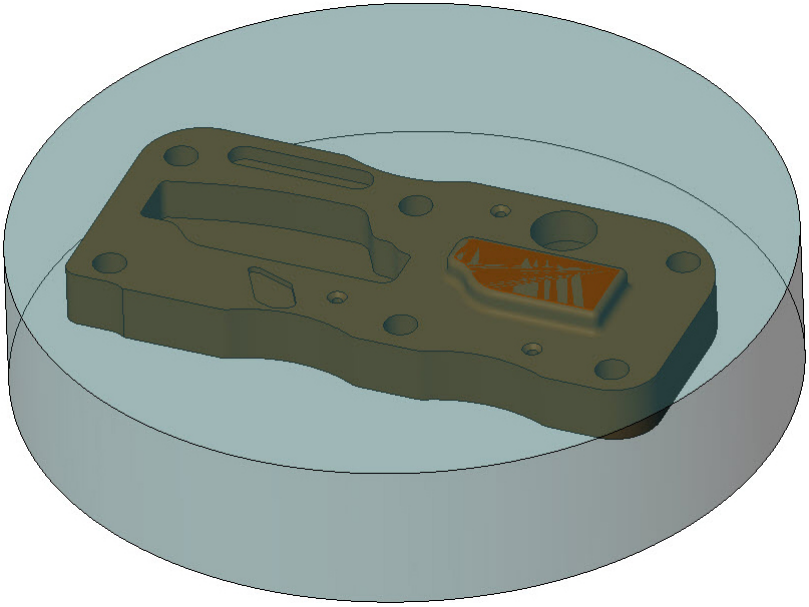


*Рис. 12.11. Цилиндрическая заготовка*

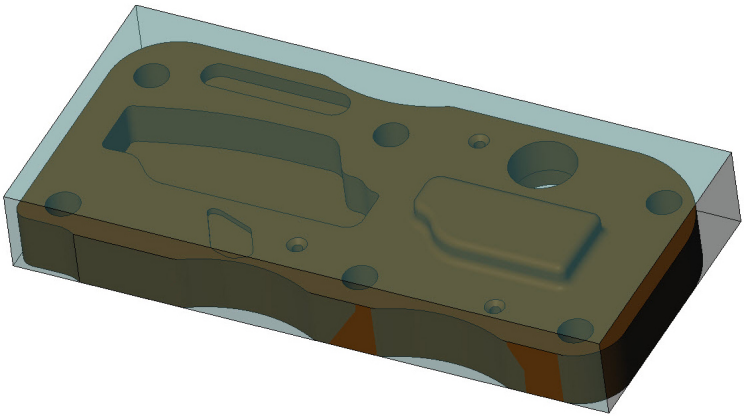


*Рис. 12.12. Заготовка в виде параллелепипеда*

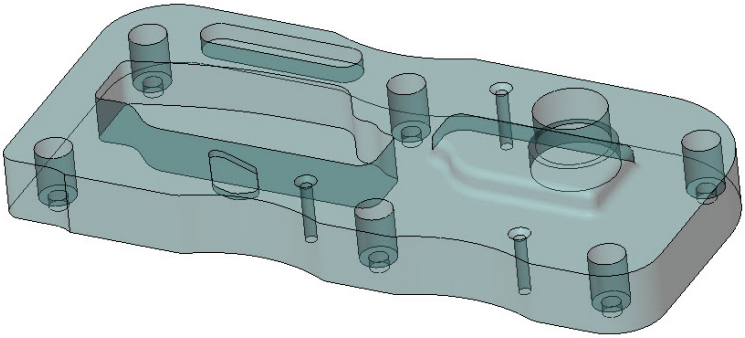


*Рис. 12.13. Для построения заготовок сложных форм, имитирующих отливки и поковки, САМ-система может создавать модели заготовок, смещая внешние границы 3D-модели детали на указанное расстояние. Большинство современных CAD/CAM-систем позволяют импортировать 3D-модель заготовки, которая была создана в другой или в этой же системе*

Существуют системы, которые **не требуют определения геометрии заготовки** на этом этапе. Заготовка назначается и используется только на этапе верификации и не участвует в процессе расчета траекторий. Такое решение является недостаточно эффективным, так как при расчетах система всегда исходит из заготовки правильной формы – параллелепипеда или цилиндра. Если же на самом деле заготовка имеет неправильную форму, то система может сформировать довольно много холостых ходов.

Выбирая геометрические элементы, подлежащие обработке, технолог-программист должен **учитывать положение детали и заготовки относительно нулевой точки**. Здесь есть два пути. Во-первых, можно сместить 3D-модель детали относительно нулевой точки таким образом, чтобы выбранный элемент совпал с ней. Во-вторых, можно смещать нулевую точку относительно модели, «привязывая» ее к определенному геометрическому элементу. Однако это не означает, что первоначальный выбор нулевой точки является окончательным. Система позволяет изменять любые параметры, в том числе и положение нулевой точки детали, в любой момент и на любом этапе проектирования обработки.

## 12.7.2. Выбор стратегии и инструмента, назначение параметров обработки

На втором этапе работы с САМ-системой технолог-программист выбирает стратегию и параметры обработки, назначает инструмент и режимы резания. Современная система обычно имеет солидный набор стратегий и позволяет выполнить обработку одной и той же детали разными способами.

Условно все стратегии можно разделить на черновые и чистовые, стратегии плоской и объемной обработки. Рассмотрим стратегии, характерные для большинства современных САМ-систем.

## Плоская обработка

Стратегии плоской обработки применяются при работе с 2D-геометрией. В этом случае не требуется большого разнообразия – вся обработка сводится к фрезерованию контура или плоскости, выборке кармана и обработке отверстий (рис. 12.14–12.17).

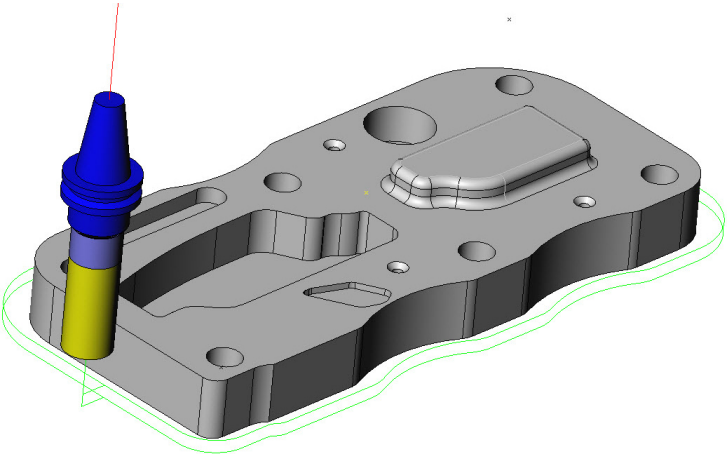


*Рис. 12.14. Контурная стратегия (Contour). Для чернового фрезерования указываются количество проходов и шаг между ними (перекрытие)*

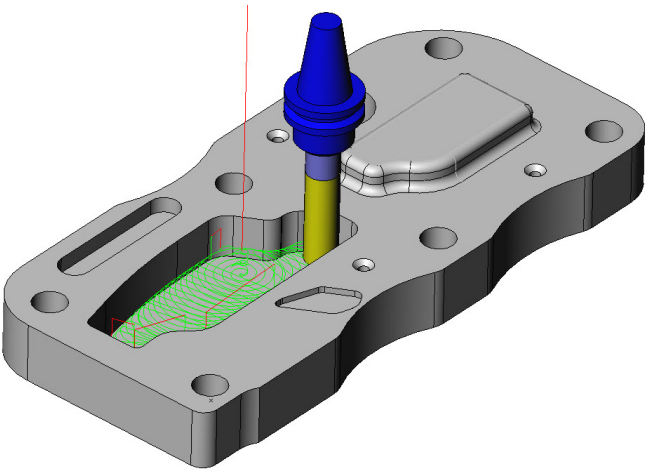


*Рис. 12.15. Обработка кармана (Pocket). Эта стратегия предназначена для выборки замкнутых областей. Основными параметрами являются шаг между проходами фрезы и тип траектории (параллельная, спиральная, зигзаг и др.)*

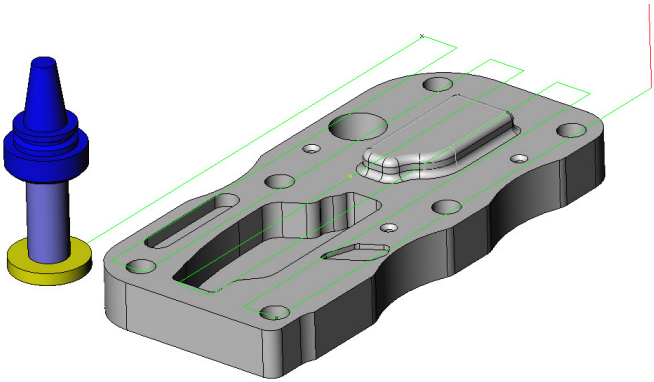


*Рис. 12.16. Обработка торца (Face). Основными параметрами для этой стратегии являются шаг между проходами фрезы и угол обработки*

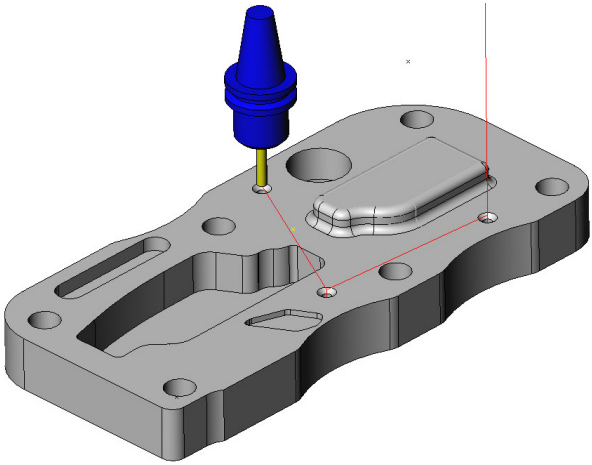


*Рис. 12.17. Обработка отверстий: сверление (Drill), нарезание резьбы (Threading), растачивание (Boring). Основные параметры – тип операции и глубина обработки*

## Объемная обработка

Стратегии объемной обработки предназначены для работы с 3D-моделями. Эти стратегии отличаются большим разнообразием, однако все они условно могут быть разделены на черновые и чистовые.

**Стратегии объемной черновой обработки** предназначены для быстрой послойной выборки большого объема материала и подготовки детали к последующей чистовой обработке. **Стратегии объемной чистовой обработки** используются для окончательного фрезерования поверхностей с требуемым качеством. Зачас-

тую при объемном чистовом фрезеровании управление перемещением режущего инструмента осуществляется одновременно минимум по трем координатам. Как правило, при объемной обработке используют сферические фрезы. В этом случае произвести расчет перемещения инструмента самостоятельно, без использования CAD/CAM-системы чрезвычайно трудно.

В качестве примеров рассмотрим некоторые стратегии объемной обработки более подробно.

**Обработка кармана** – стратегия, предназначенная для эффективного удаления материала из закрытых или открытых карманов. Существует множество схем карманной обработки: зигзаг, в одном направлении, параллельная и круговая спираль. Современная CAD/CAM-система выбирает оптимальную схему фрезерования, обеспечивая максимальную производительность и минимальное число холостых ходов. Как правило, эта стратегия заключается в последовательной послойной выборке материала и выполнении заключительного чистового обхода контура на окончательной глубине. При зигзагообразной схеме обработки кармана происходит изменение вида фрезерования – с встречного на попутное или наоборот.

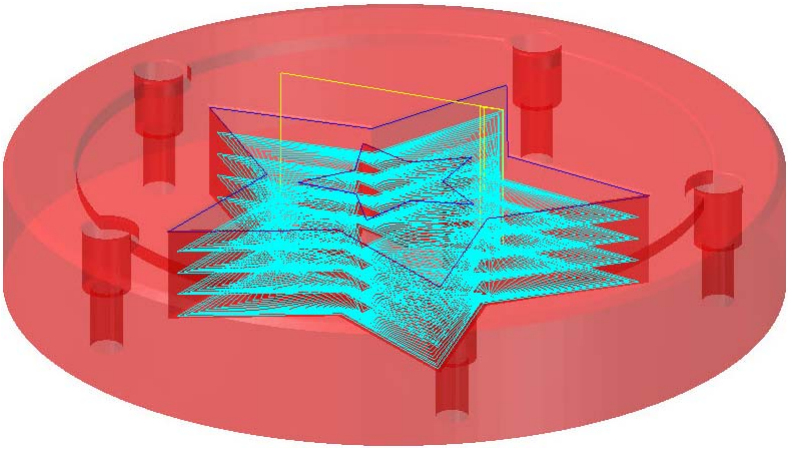


*Рис. 12.18. Послойная обработка кармана*

**Стратегия радиальной обработки** обычно применяется для черновой или чистовой обработки деталей круглой формы. Перемещение инструмента в этой стратегии производится от центра детали к ее внешним границам (или наоборот) с постепенным изменением угла в плоскости обработки.

С помощью **черновой вертикальной выборки** можно быстро обработать деталь, используя движения, аналогичные сверлению. Производства, использующие такую стратегию, приобретают специальные плунжерные фрезы, позволяющие быстро выбрать материал и имеющие подвод охлаждающей жидкости через ось инструмента. Стратегия черновой вертикальной выборки отлично подходит для обработки глубоких впадин и карманов.

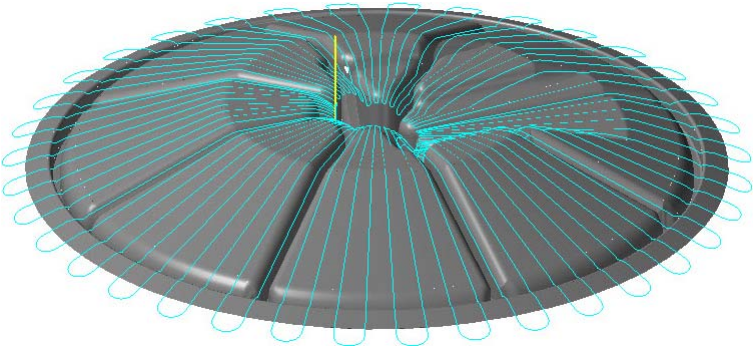


*Рис. 12.19. Радиальная обработка*

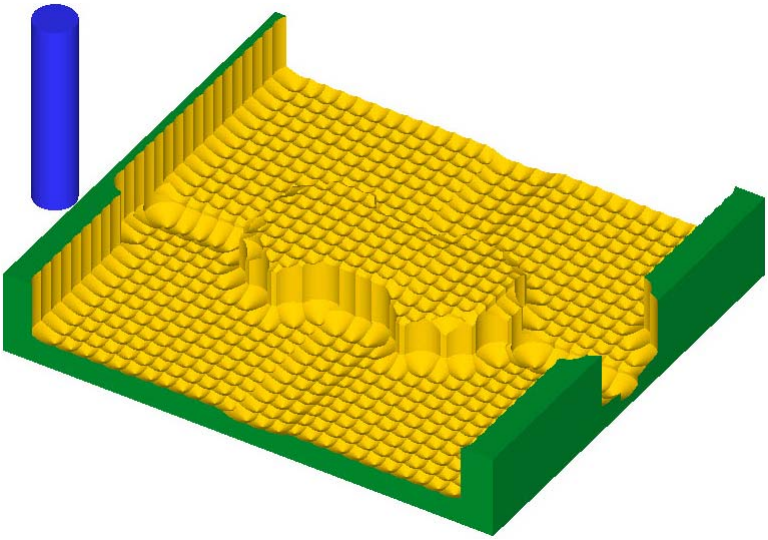


*Рис. 12.20. Результат применения стратегии черновой вертикальной выборки*

**Стратегия фрезерования остатков (дообработка)** позволяет автоматически удалить материал, оставшийся от предыдущей операции. Для увеличения производительности обработки принято выбирать материал сначала инструментом большого диаметра, а уже затем производить фрезерование в труднодоступных местах инструментом меньшего диаметра. Система, анализируя объем снятого и оставшегося материала, автоматически генерирует траекторию для выборки материала, который не был удален в предыдущей операции обработки.

**Стратегия контурной обработки** используется для черновой или чистовой контурной обработки деталей произвольной формы. Суть стратегии заключается в удалении припуска за счет проходов фрезы по контурам, созданным путем «смещения» границ текущего слоя по Z.

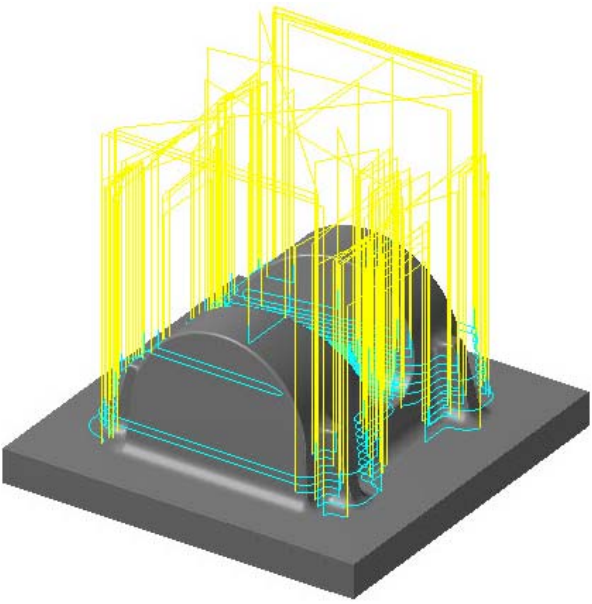


*Рис. 12.21. Стратегия дообработки меньшим инструментом*

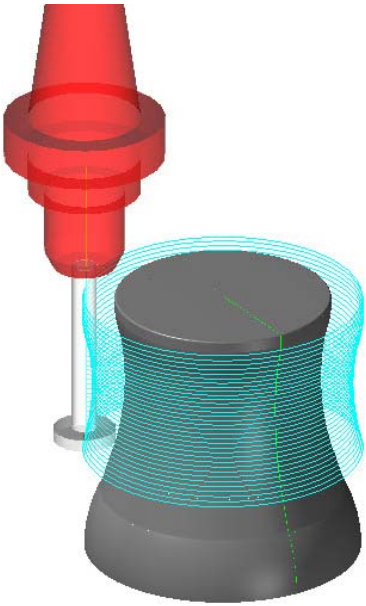


*Рис. 12.22. Объемная контурная обработка*

**Стратегию обработки по потоковым линиям** применяют для чистовой обработки любых поверхностей. Система создает траектории с учетом формы и направления поверхностей.

**Стратегия проекционной обработки** позволяет наложить на любой участок обрабатываемой поверхности своеобразный шаблон траектории движения инструмента или спроецировать плоскую траекторию на 3D-модель.

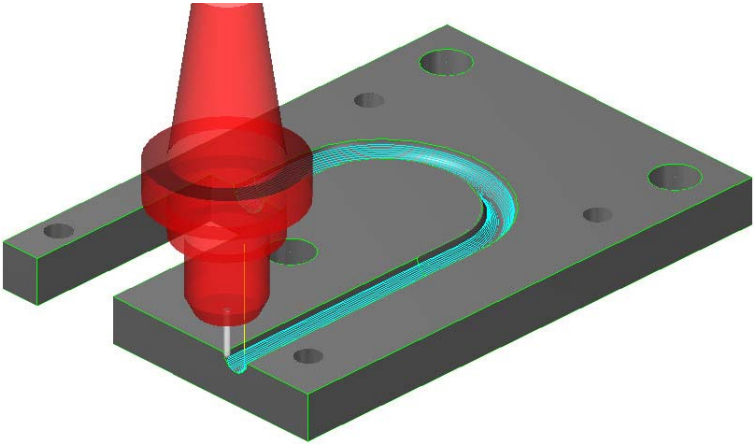


*Рис. 12.23. Обработка по потоковым линиям, схема – зигзаг*

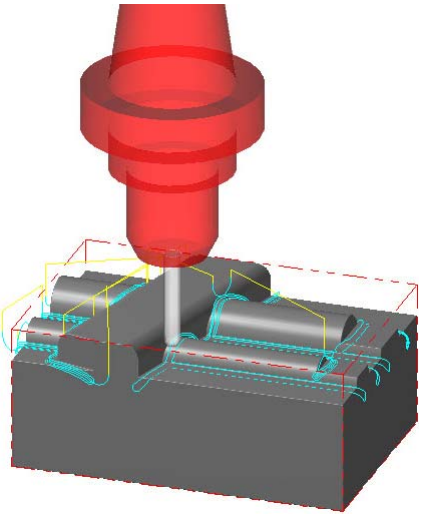


Рис. 12.24. Карандашная обработка

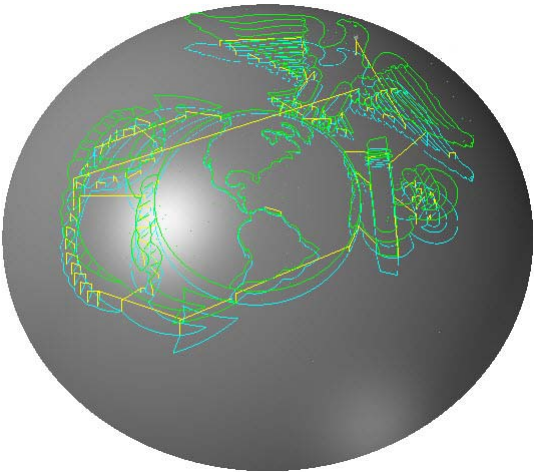


Рис. 12.25. Проецирование траектории на поверхность

После выбора стратегии и определения основных параметров обработки необходимо **назначить** режущий инструмент либо **выбрать** его из библиотеки инструментов. На этом же этапе определяются режимы резания: скорость рабочей подачи, обороты шпинделя – и программируется включение или выключение СОЖ. Результатом второго этапа является сформированная траектория. В дереве операций САМ-системы должна появиться новая технологическая операция.

## 12.7.3. Бэкплот и верификация

В настоящее время любая САМ-система имеет функции для проверки правильности созданных траекторий. **Функция бэкплота** (Backplot) позволяет программисту отслеживать перемещения режущего инструмента. При этом он может наблюдать за траекторией центра инструмента и самим инструментом прямо на 3D-модели. Как правило, бэкплот используется для предварительной проверки рассчитанных траекторий и настройки технологических параметров операции. Окончательная проверка обычно осуществляется с помощью верификации.

Инструменты верификации предоставляют программисту прекрасные возможности для наглядной проверки траектории движения инструмента, для оценки качества и общей технологии изготовления детали. **Основной смысл верификации** заключается в демонстрации процесса удаления материала заготовки и возможности посмотреть на окончательный результат работы УП – модель изготовленной детали. Полученную «виртуальную» деталь можно рассмотреть с разных сторон. Можно увидеть, все ли элементы выполнены правильно, и даже разглядеть гребешки на материале, оставшиеся от инструмента.

Верификация может быть твердотельной или растровой. В случае **твердотельной верификации** система работает с трехмерной моделью заготовки и позволяет

*Рис. 12.26. В режиме «Backplot» можно наблюдать за траекторией перемещения центра инструмента*

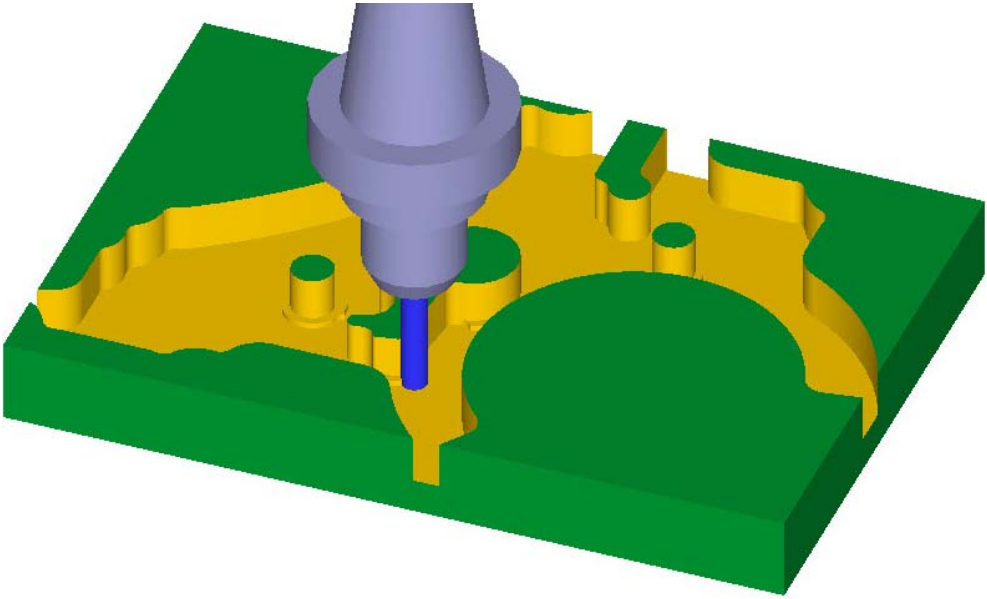


*Рис. 12.27. Процесс верификации*

реализовать множество полезных функций. К примеру, измерить обработанную деталь или экспортировать ее в CAD-систему для дальнейшей работы. **Растровая верификация** лишь имитирует работу с трехмерной моделью и применяется сейчас достаточно редко.

Учтите, что на этом этапе мы еще не имеем готовой УП, то есть самого кода. Код программы обработки появится позже, после постпроцессирования. Но что же мы тогда проверяем?

Результатом предыдущего этапа является сформированная траектория перемещения инструмента для определенной операции. Информацию об этой траектории, всех координатах и инструменте система записывает в специальный промежуточный файл. Этот файл не похож на обычную программу обработки, то есть в нем нет привычных G- и M-кодов. В настоящий момент именно этот файл является объектом для верификации и бэкплота.

## 12.7.4. Постпроцессирование

**Постпроцессор** – программа, которая преобразует файл траектории движения инструмента и технологических команд (промежуточный файл), сформированный CAD/CAM-системой, в файл УП в соответствии с требованиями конкретного комплекса станок – СЧПУ. В ряде отечественных систем постпроцессоры называются **паспортами**.

Для того чтобы абстрагироваться от большого разнообразия станков, систем ЧПУ и языков программирования обработки, CAM-система генерирует промежуточный файл, содержащий информацию о траектории, угле поворота инструмента (в случае многокоординатной обработки) и обобщенные команды управления станком. Обычно этот промежуточный файл называется CL-файлом (Cutter Location) или CLDATA-файлом.

Далее в работу вступает постпроцессор. Он преобразует этот промежуточный файл в программу обработки в строгом соответствии с форматом программирования конкретного станка с ЧПУ.

**Такая технология позволяет программисту во время проектирования обработки в CAD/CAM-системе не задумываться о том, на какой конкретно станок попадет УП и каков будет ее формат.** Ему необходимо лишь выбрать постпроцессор, соответствующий определенному станку с ЧПУ, и тот возьмет на себя всю работу по созданию программы обработки определенного формата.

Откуда же взялась идея постпроцессирования и почему до сих пор в этой области существуют проблемы? По идее, разработчики станков и систем ЧПУ должны соблюдать стандарты Ассоциации электронной промышленности (EIA) и Международной организации стандартизации (ISO). То есть одинаковые G-коды на разных станках с ЧПУ должны выполнять одну и ту же функцию. В принципе, эти стандарты соблюдаются, но только для основных команд станка, например для включения СОЖ, линейной и круговой интерполяции.

Если же дело доходит до других команд, постоянных циклов и специальных функций, то приверженность определенному стандарту практически отсутству-

ет. Это приводит к невозможности правильного исполнения одной и той же УП на разных станках с ЧПУ. К этой проблеме прибавляется другая – постоянная «гонка» производителей оборудования с ЧПУ. В условиях жесткой конкуренции станкостроительные компании создают все более сложные станки, а разработчики систем ЧПУ придумывают новые циклы и функции. В результате серьезные отличия в формате УП могут быть замечены даже у станков одной фирмы, но разных моделей.

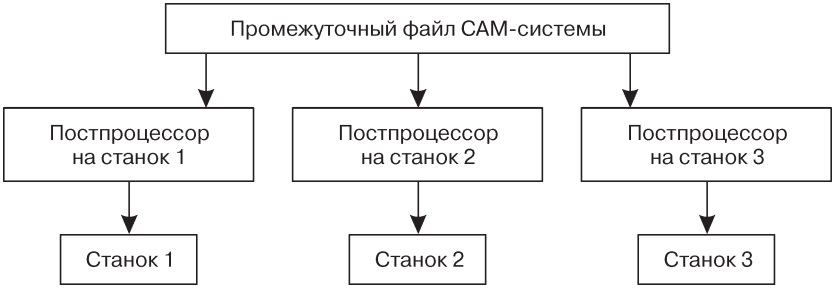



*Рис. 12.28. Схема получения УП для конкретного комплекса «Станок – система ЧПУ»*

Когда появились первые САМ-системы, то для работы с конкретным станком с ЧПУ разрабатывался **индивидуальный постпроцессор**, который представлял собой исполняемый файл. Индивидуальный постпроцессор мог быть создан только опытным программистом (именно программистом, а не технологом-программистом) путем длительного общения со станочником для выяснения всех нюансов работы с определенным станком. После этого проходили испытания и доводка индивидуального постпроцессора «до ума». В результате постпроцессор выполнял свои функции, но процесс его создания был очень долгим, мучительным и дорогим. Изменения в таком постпроцессоре мог сделать только сам автор-разработчик.

В 70–80-х годах прошлого века наблюдался значительный подъем автоматизированного машиностроения. Как грибы после дождя стали появляться новые станки с различными системами ЧПУ. Возник огромный спрос на технологическое программное обеспечение и CAD/CAM-системы.

Программисты просто не успевали разрабатывать новые индивидуальные постпроцессоры, что подтолкнуло их к автоматизации собственного труда. Появилась идея создания обобщенных постпроцессоров для разных станков с одинаковой системой ЧПУ. В этом был смысл – при работе с системой ЧПУ одной фирмы даже на разных станках отличия в УП будут минимальными, значит, и изменения, которые нужно внести в исполняемый файл, тоже будут незначительными.

**Классический постпроцессор**, находящийся на службе современной CAD/CAM-системы, состоит из нескольких файлов. Во-первых, это исполняемый файл – программа. Исполняемый файл занимается преобразованием данных промежуточного CL-файла в кадры УП. Преобразование осуществляется по некото-

рым правилам, отличным для разных станков и систем ЧПУ. Эти правила, или алгоритмы преобразования, находятся во втором файле – текстовом.

**Текстовый файл написан на специальном макроязыке, который может быть изменен в случае необходимости самим технологом-программистом в любом текстовом редакторе.** Вносить какие-либо изменения в исполняемый файл не требуется. Как правило, с CAD/CAM-системой поставляется набор таких текстовых файлов, которые описывают правила для преобразования промежуточных файлов в программу обработки для нескольких десятков различных станков и систем ЧПУ. В данном случае постпроцессором можно смело называть именно текстовый файл. Исполняемый же файл является модулем CAM-системы и работает незаметно для пользователя. В некоторых CAM-системах присутствует и третий файл, необходимый для постпроцессирования. Этот файл также является текстовым. Он предназначен для ввода дополнительных условий и передачи специальной информации в управляющую программу.

К сожалению, **разработчики CAD/CAM-систем не придерживаются единого стандарта** для формирования промежуточных CL-файлов. В результате текстовый файл с описанием алгоритмов преобразования (постпроцессор) одной CAM-системы будет абсолютно бесполезен для использования внутри другой CAM-системы. Для разработки даже такого текстового постпроцессора «с нуля» необходимы глубокие знания самой системы и принципов преобразования исходных данных. Поэтому пользователю предоставляют набор базовых постпроцессоров с редактируемыми переменными. Работая с этими переменными, пользователь может самостоятельно настроить постпроцессор для имеющегося станка.

Сегодня наиболее эффективным и простым решением проблем постпроцессирования на предприятии является использование универсального постпроцессора (генератора).

**Универсальные постпроцессоры** позволяют работать со многими известными CAD/CAM-системами и одновременно не зависеть от них. Дело в том, что, в отличие от классических постпроцессоров, которые работают с промежуточным файлом только «родной» системы, универсальные постпроцессоры способны «перерабатывать» CL-файлы разных CAM-систем. Прибавьте сюда продвинутый интуитивно-понятный графический интерфейс, позволяющий строить наглядную кинематическую схему станка и задавать типовые параметры системы ЧПУ, и вы получите мощное и универсальное средство для решения проблем совместимости любого оборудования с любой CAD/CAM-системой.

Как оценить результаты работы постпроцессора? Очень просто. Если полученная программа не требует ручной правки и доводки, значит, постпроцессор работает правильно. Это главный принцип. Хороший постпроцессор способен сформировать УП с учетом максимальных возможностей станка с ЧПУ и CAD/CAM-системы и должен грамотно работать с постоянными циклами. К примеру, если система ЧПУ станка поддерживает винтовую интерполяцию, то желательно, чтобы постпроцессор не заменял ее множеством линейных перемещений. Естественно, что программа обработки не должна содержать лишних кодов и координат.

## 12.7.5. Передача УП на станок с ЧПУ

Итак, программа обработки готова. После постпроцессирования CAD/CAM-система открывает файл УП в специальном редакторе или в стандартном «Блокноте» операционной системы Windows. Многие системы поставляются со специальными редакторами УП, которые являются простым и удобным инструментом для правки и передачи программ на станок.

**В большинстве случаев связь осуществляется в соответствии со стандартом RS-232.** При этом COM-порт персонального компьютера соединяется кабелем со специальным разъемом на корпусе станка или панели УЧПУ. **Для передачи данных необходимо, чтобы УЧПУ станка и коммуникационная программа были синхронизированы.** Это достигается соответствующей настройкой параметров СЧПУ и коммуникационной программы. Например, если значение для стоповых битов (stop bits) системы ЧПУ станка равно двум, то такое же значение должно находиться и в соответствующем параметре коммуникационной программы.

После настройки основных параметров УП можно передать на станок. Вам придется переключить систему ЧПУ станка в режим приема/передачи данных, нажать на панели УЧПУ клавишу **Прием данных**, а в коммуникационной программе выбрать пункт меню **Передать**. Для передачи данных со станка на ПК нужно действовать наоборот. Сначала переключаем СЧПУ станка в режим приема/передачи данных, затем в коммуникационной программе выбираем пункт меню **Принять** и нажимает клавишу на панели УЧПУ станка **Передать данные**. Внимательно читайте документацию к станку – в ней вы обязательно найдете информацию о настраиваемых параметрах для приема и передачи данных, а также последовательность действий при работе с внешними устройствами хранения информации.

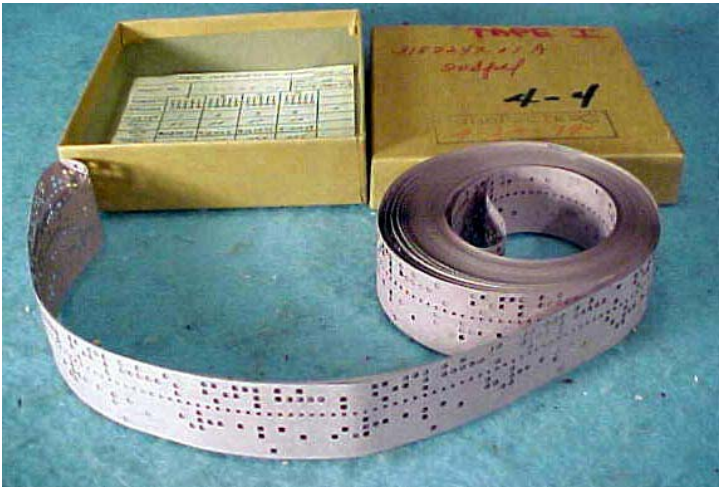



*Рис. 12.29. На многих заводах до сих пор программы загружают в станок с ЧПУ при помощи перфоленты*

**Для передачи УП, размер которых превышает размер свободной памяти СЧПУ, используется режим DNC.** В этом режиме программа обработки не записывается в память системы ЧПУ, а выполняется прямо с компьютера. Система ЧПУ имеет (или создает в памяти) специальный программный буфер, в который приходит часть программы. Как только этот буфер заполняется, система посылает сигнал на компьютер, чтобы тот прекратил передавать данные. В это время СЧПУ станка выполняет кадры УП, находящиеся в буфере. Когда буфер освобождается, система ЧПУ посылает новый сигнал на ПК, который разрешает дальнейшую передачу данных. Этот процесс продолжается до полного выполнения УП. Для работы в режиме DNC необходимо, чтобы СЧПУ станка было соответствующим образом подготовлено производителем, а на персональном компьютере находилась коммуникационная программа с поддержкой этого режима.

Наиболее продвинутые системы дают возможность передавать данные по локальной сети, а не по RS-232. В этом случае пользователь прямо с ПК может «видеть» содержимое памяти станка. А прием и передача данных осуществляются привычным для Windows способом – перетаскиванием файлов из одной папки в другую, что очень удобно. Некоторые станки с ЧПУ снабжены встроенным дисководом или умеют читать информацию с флэш-карт.

## 12.8. Ассоциативность

Ассоциативность CAD/CAM-системы заключается в ее способности связать геометрию с траекторией обработки, инструментом, материалом, параметрами и сформировать завершенную операцию. Если какая-либо часть операции изменяется, то другие ее части остаются нетронутыми и могут быть использованы для дальнейших расчетов и создания обновленной операции. Ассоциативность предоставляет технологу-программисту небывалые возможности по отладке технологии обработки и защищает его от утомительного исправления своих ошибок. Изменяя параметры операции, можно следить, как меняются траектория и машинное время обработки, и в результате выбрать наилучший вариант. Как правило, такая ассоциативность действует в пределах только одной CAD/CAM-системы.

## 12.9. Пятикоординатное фрезерование и 3D-коррекция

В последнее время наблюдается значительный интерес к многоосевой обработке. Это вызвано, с одной стороны, повышением спроса на изготовление деталей сложной формы, с другой – снижением стоимости 5-координатных станков с ЧПУ и развитием математического аппарата CAD/CAM-систем.

Традиционной областью применения этой технологии является авиационная промышленность, где 5-координатные обрабатывающие центры служат для механической обработки турбинных лопаток, лопастей и других деталей сложной

формы. Постепенно эта прогрессивная технология внедряется в обычное производство для изготовления инструмента и пресс-форм.

При 5-координатном фрезеровании инструмент может обрабатывать поверхность детали торцевой или боковой частью. При такой обработке обычно используют концевые сферические фрезы, поэтому в первом случае контакт инструмента с обрабатываемой поверхностью будет точечным, а во втором – линейным.

Существуют два вида 5-координатной обработки: **одновременная (непрерывная)** и **обработка с индексированием**. В первом случае в каждом кадре УП действительно находятся пять адресов осевого перемещения, например: X, Y, Z, A, B. Во втором случае нельзя говорить об одновременном перемещении по всем пяти осям – в каждом кадре обычно содержатся только три координаты. Остальные адреса стоят отдельно и используются как вспомогательные – для поворота инструмента или детали в определенное положение и дальнейшей 3-координатной обработки.

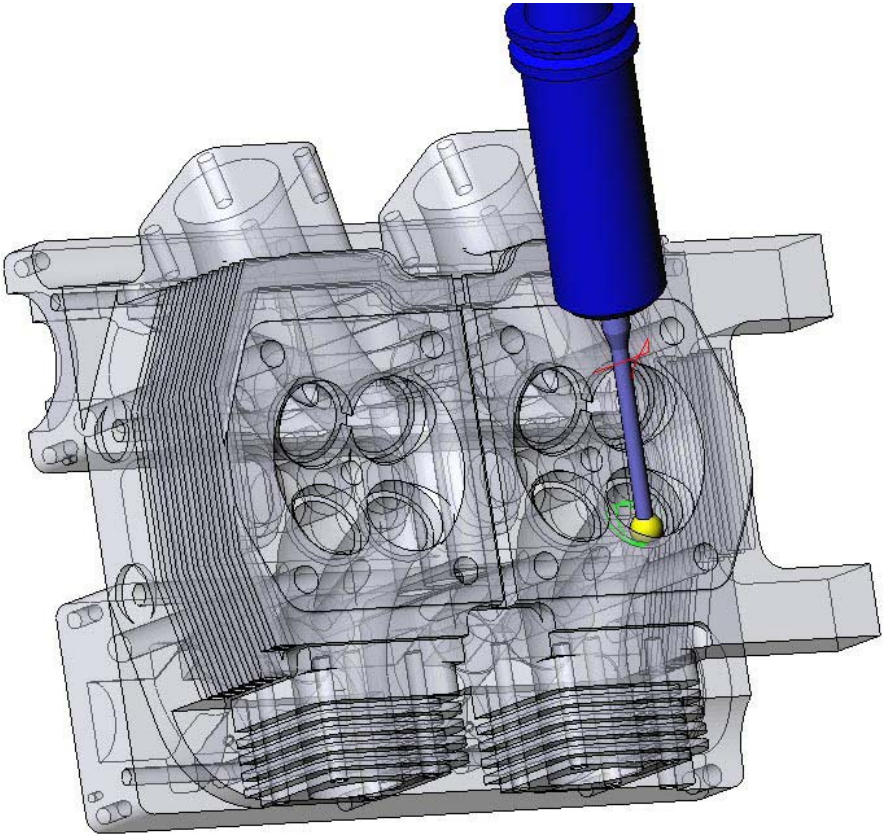


*Рис. 12.30. Для расчета траекторий 5-координатной обработки используются CAM-системы. В данном примере производится обработка канала двигателя в системе ESPRIT*

Управляющие программы для многоосевой обработки создаются исключительно при помощи CAD/CAM-систем. Зачастую технологу-программисту приходится строить дополнительные направляющие поверхности и ограничивать угол наклона режущего инструмента. Для получения корректной программы требуется тщательная настройка постпроцессора, создание которого может обойтись предприятию довольно дорого.

Термин **3D-коррекция** часто используется, когда речь заходит о технологии объемной обработки. При обычном плоском фрезеровании существует возможность выполнить коррекцию на радиус инструмента слева или справа от запрограммированного контура при помощи кодов G41 и G42. А как поступить в случае объемной поверхностной обработки, например при изготовлении матриц и пуансонов?

Когда производится обработка плоского контура, корректирующее смещение указывается справа или слева, то есть по нормали к обрабатываемой поверхности в точке контакта с инструментом. При использовании 3D-коррекции ситуация аналогичная, просто необходимо знать вектор ориентации инструмента и вектор нормали поверхности в точке контакта с инструментом. Исходя из взаимного положения этих векторов и корректирующего значения, **система ЧПУ рассчитывает пространственное смещение режущего инструмента** с сохранением его ориентации и заданной точки контакта.

Если система ЧПУ на вашем станке поддерживает 3D-коррекцию инструмента, то оператор может влиять на размеры деталей со сложной геометрией, изменяя значение радиуса инструмента как «в плюс», так и «в минус». Однако в этом случае CAM-система и СЧПУ станка, скорее всего, не смогут контролировать возможные столкновения и ошибки позиционирования.

## 12.10. Высокоскоростная обработка (ВСО)

Теория использования высоких режимов резания при механической обработке металлов возникла достаточно давно. Однако только сейчас, с появлением самых современных станков с ЧПУ, частота вращения шпинделя у которых достигает 60 000 об/мин, а скорость рабочей подачи переваливает за 5000 мм/мин, из области научных предположений мы переходим к реальному производству.

Экспериментально установлено, что при увеличении скорости резания температура в зоне резания постепенно возрастает и достигает некоторого максимального значения. При дальнейшем увеличении скорости резания происходит некоторое падение температуры и крутящий момент, необходимый для выполнения резания, тоже снижается. Это означает, что существует некоторая область сверхвысоких скоростей обработки, в которой процесс резания происходит спокойно и режущий инструмент не подвергается катастрофическим нагрузкам. В настоящее время для каждого обрабатываемого материала и инструмента эта область определяется только опытным путем.

Работа в условиях ВСО имеет ряд нюансов и предъявляет особые требования к оборудованию, инструменту и управляющей программе.

При работе в таком режиме, во-первых, станок должен иметь частоту вращения шпинделя не менее 15 000 об/мин и подачу не менее 2500 мм/мин, во-вторых, он должен успевать за программой, то есть быстро ускорять и замедлять рабочую подачу. Желательно, чтобы станок обладал высокой жесткостью и имел эффективные средства для удаления стружки.

Режущий инструмент является главным «ограничителем» скорости обработки. Как правило, режущая часть инструмента для ВСО имеет специальное износостойкое покрытие. Уделите особое внимание патрону, так как малейшая погрешность установки вызывает биение, особо опасное на таких скоростях вращения шпинделя. Вылет инструмента должен быть минимальным.

Теперь поговорим об особенностях программирования. В случае ВСО глубина и шаг обработки гораздо меньшие, чем при обычном фрезеровании. Траектория перемещения должна быть плавной, без резких смен направления и скорости подачи. Часто линейные перемещения заменяют на петлеобразные, используют трохоидальную траекторию. Врезание инструмента в металл должно проходить по спирали или под небольшим углом, но никак не вертикально. Конечно же все это делается для поддержания неизменных условий резания, уменьшения нагрузки на инструмент и исключения его поломки.

Так как УП для ВСО содержит очень много перемещений, то ее размер может превышать размер обычной программы обработки в десятки или сотни раз. Система ЧПУ станка должна успевать отрабатывать кадры и иметь достаточно большой программный буфер для подготовки к последующим перемещениям. Если система не имеет значительного объема памяти для хранения программ, то не обойтись без DNC-режима. При этом предъявляются особые требования по скорости и надежности к персональному компьютеру, коммуникационному программному обеспечению и линии связи.

Не стоит увлекаться методом ВСО для обработки всей детали целиком. В большинстве случаев черновую обработку можно выполнить на обычных режимах «грубым» инструментом. Серьезная CAM-система обязательно должна иметь инструменты дообработки, средства определения излишков материала и возможность сравнить результаты обработки с исходной моделью. Используя различные программные фильтры и оптимизаторы, производящие анализ перемещений в УП, можно значительно сократить размер программы, сделать ее наиболее подходящей для метода ВСО.

Каковы преимущества от использования ВСО? Меньший шаг и глубина фрезерования в сочетании с большими значениями рабочей подачи и оборотов шпинделя при чистовой обработке существенно повышают качество поверхности и значительно сокращают машинное время. Это позволяет избежать ручной доводки, например при изготовлении пресс-форм. При ВСО можно использовать инструмент меньшего размера и фрезеровать такие мелкие детали и острые углы, которые в другом случае пришлось бы получать электроэрозионной обработкой.

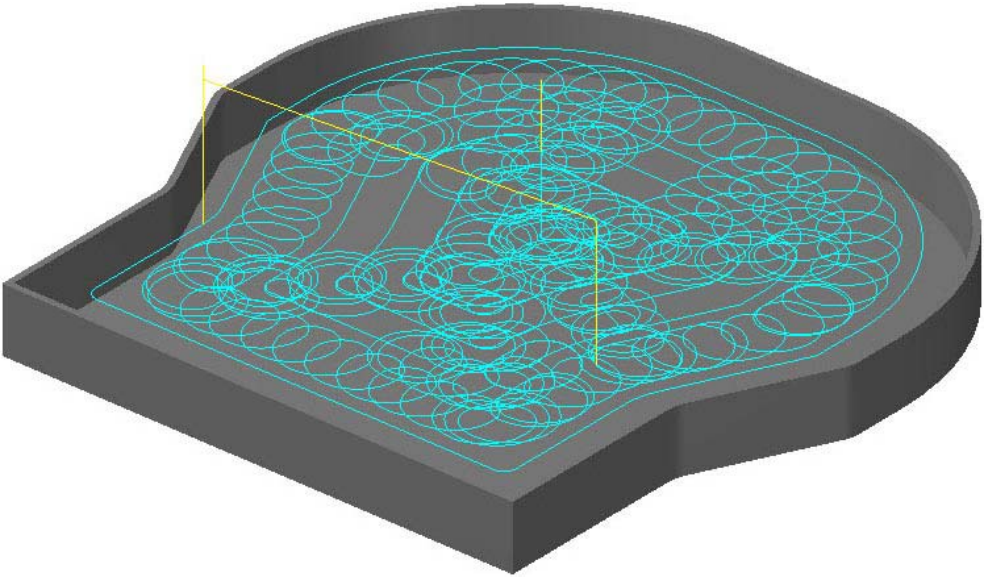


*Рис. 12.31. Трохоидальная траектория состоит из множества «петелек»*

Что касается черновой обработки, то положительный эффект от ВСО не так очевиден и проявляется лишь при обработке деталей небольшого размера или высокой твердости и при достаточной загрузке станка.

## 12.11. Требования к современной CAM-системе

Сегодня на рынке CAD/CAM представлен не один десяток систем, отличающихся возможностями, интерфейсом и стоимостью. Как сделать правильный выбор и приобрести систему, которая бы полностью устраивала технолога и позволяла максимально автоматизировать процесс создания УП? Есть ряд параметров, по которым можно составить первоначальное мнение о CAM-системе.

Первое, что бросается в глаза при знакомстве с системой, – ее интерфейс, то есть совокупность меню, пиктограмм и окон. **Главное требование к интерфейсу – удобство.** Подавляющее большинство современных CAM-систем имеет привычный Windows-интерфейс, позволяющий сформировать удобную для пользователя атмосферу. Что же касается наглядности и графического оформления – то это дело вкуса. Обратите внимание на язык интерфейса. Если вы не владеете иностранным языком, то предпочтительнее, чтобы все меню были представлены на русском языке.

CAM-система не должна ограничивать технолога выбором только стандартного инструмента. **Функция поддержки оригинального (пользовательского) инструмента** позволит вам спроектировать инструмент любой формы прямо в сис-

теме и правильно рассчитать траекторию его перемещения. Поверьте, рано или поздно эта функция обязательно понадобится.

Вряд ли вам понравится, если вы обнаружите зарезы на детали после черновой или чистовой обработки. Хорошая CAM-система обязана иметь механизм, обеспечивающий **предотвращение зарезов и столкновений инструмента с заготовкой и элементами крепежа**.

Если траектории перемещения инструмента рассчитываются с учетом **заданной заготовки произвольной формы**, то это говорит о высокой эффективности созданных системой операций. Другими словами – система должна «видеть» заготовку. Преимущество этой функции проявляется при работе с заготовками в виде отливок и штамповок. Когда система не имеет этой функции, она производит расчет траекторий на основе цилиндра или параллелепипеда. Полученная траектория окажется неоптимальной, в ней может присутствовать довольно большое количество холостых перемещений.

Способность системы «помнить», сколько материала было снято в предыдущих операциях, говорит об «интеллектуальном» потенциале системы. **Функция дообработки** позволяет автоматически находить недоработанные области и гарантирует получение максимально эффективных траекторий.

К верификатору CAM-системы нужно отнестись очень внимательно. Во-первых, система должна обеспечивать **возможность импорта трехмерной модели заготовки** из CAD-модуля и верификацию этой модели. Во-вторых, верификатор должен уметь вращать заготовку, динамически масштабировать и перемещать по экрану. В-третьих, при верификации система должна отмечать цветом места зарезов и столкновений инструмента с заготовкой.

Верификаторы большинства современных CAD/CAM-систем имеют несколько режимов работы и множество полезных настроек. **Режим верификатора «Turbo»** используется для работы со сложными заготовками и большими траекториями, если ожидаемое время процесса верификации слишком велико. При работе в режиме Turbo верификатор производит расчеты и показывает только конечный результат – модель обработанной детали.

Графическое качество процесса верификации сильно зависит от производительности компьютера и от соответствующих настроек верификатора. Как правило, между качеством и скоростью существует прямая зависимость – **чем выше качество верификации, тем медленнее она производится**.

Обычно верификаторы позволяют осуществлять визуальную проверку 3-осевой обработки. Верификация многоосевой обработки может быть опцией. Следовательно, если вы собираетесь работать с четырьмя или пятью координатами в УП, то лучше проверьте возможности верификатора заранее.

Наиболее продвинутые верификаторы позволяют интенсивно работать с 3D-моделью заготовки. При наличии такого верификатора вы сможете экспортировать модель обработанной детали в другую CAD-систему, проверить ее геометрические размеры или сделать сечение. **Дополнительным преимуществом является способность осуществлять проверку не только промежуточного CL-файла, но и кода управляющей программы.**

Часто возникают ситуации, когда технологу-программисту приходится работать с некачественной 3D-моделью детали. Например, модель может попасть в CAM-модуль «дырявой», то есть имеющей поверхностные нестыковки или исчезнувшие геометрические элементы. Некоторые CAM-системы способны «закрыть глаза» на это, другие же просто не смогут работать с такими моделями и потребуют их «лечения».

Если вы создаете геометрию детали в CAD-системе, которая не является «родной» для CAM-системы, то вам не обойтись без помощи конверторов (трансляторов). **Конверторы предназначены для преобразования графических файлов одного формата в графические файлы другого (требуемого) формата.** О формате вы можете судить по расширению файлов. Например, .DWG, .DXF, .IGS, .STL, .SAT. При знакомстве с CAM-системой убедитесь в наличии требуемых конверторов и проверьте качество конвертации файлов.

# Глава 13
# СИСТЕМА ТРЕХМЕРНОГО ТВЕРДОТЕЛЬНОГО МОДЕЛИРОВАНИЯ КОМПАС-3D

Системы автоматизированного проектирования (САПР-системы, CAD-системы) изделий машиностроения, приборостроения и строительства существуют уже несколько десятков лет. Сегодня есть целый ряд таких систем, различающихся по принципам проектирования и возможностям. Одни из них больше «заточены» под поверхностное моделирование отдельных деталей, другие являются «твердотельными» и используются как при геометрическом моделировании, так и при имитации сборочных моделей. Программные комплексы имеют различные математические ядра и функциональность.

Современный уровень компьютерной техники, его возможности по обработке массивов данных и визуализации достаточно высоки, поэтому и возможности систем одного класса приблизительно соответствуют друг другу. Далее в главе мы рассмотрим возможности геометрического моделирования деталей с помощью одной из самых известных на российском рынке САПР – КОМПАС-3D.

КОМПАС-3D – классическая система трехмерного моделирования «среднего уровня». КОМПАС предназначен как для моделирования деталей и сборок различной сложности, так и для выпуска полного комплекта конструкторской документации на разрабатываемое изделие в полном соответствии с отечественными (а с недавних пор – и зарубежными) стандартами. Поддерживаются экспорт и импорт документов в разных обменных форматах для передачи инженерных данных. Это позволяет пользователям работать комфортно, не теряя информации.

Так как предлагаемая книга в большей степени посвящена САМ-системам, то и данная глава ориентирована на то, чтобы дать читателю представление о подходах к геометрическому представлению деталей в системе КОМПАС, а также о передаче его в системы подготовки управляющих программ для станков с ЧПУ.

Как уже было сказано, САПР КОМПАС-3D объединяет в себе как твердотельные операции (рис. 13.1), так и операции работы с поверхностями (рис. 13.2).

## *Классические твердотельные операции*

Операция выдавливания, при которой формообразование (наращивание или удаление материала) происходит путем перемещения эскиза в направлении, перпендикулярном его плоскости. Примерами могут служить плиты, основания, станины и т. п.

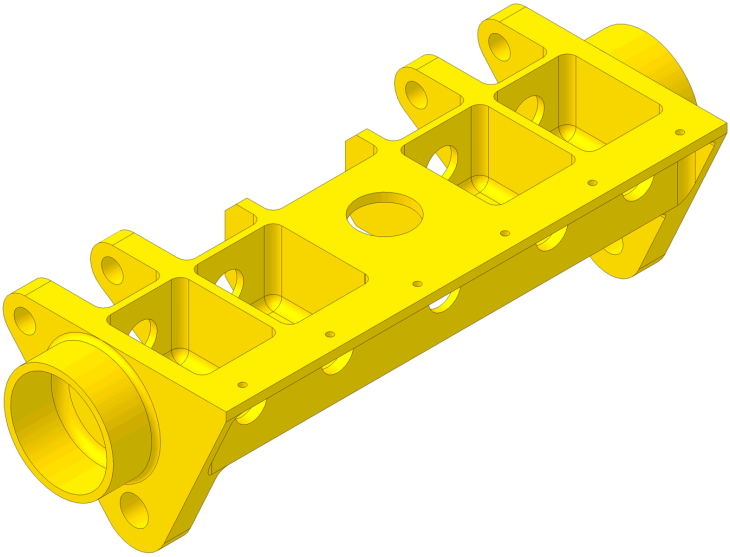


*Рис. 13.1. Твердотельная модель*

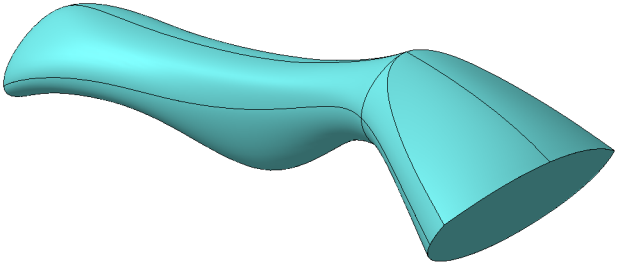


*Рис. 13.2. Поверхностная модель*

Операция вращения, когда эскиз является образующей тела вращения. Эскиз «закручивается» вокруг специально отрисованной осевой линии. Наиболее характерные детали – валы, оси, стержни.

Операция по сечениям, в которой новый объем образуется последовательно по набору поперечных сечений. Это фюзеляжи, корпуса судов простой формы и т. п.

Операция кинематическая, когда постоянное поперечное сечение как бы протягивается вдоль сложной пространственной траектории. Самый простой пример – это трубопроводы, пружины, кабели и провода.

К твердотельным относятся и прочие – «вспомогательные» – операции, такие как **Ребро жесткости**, **Отверстие**, **Оболочка**, **Фаска**, **Скругление**, **Уклон**, **Масштабирование** и некоторые другие. Создание одинаковых повторяющихся элементов осуществляется с помощью таких операций, как **Зеркальный массив**, **Массивы по прямоугольной или концентрической сетке, по кривой и по точкам эскиза**.

Отдельным блоком можно выделить набор операций для работы с листовым материалом – гибка и формовка, которые применяются при разработке корпусных деталей, приборов, емкостей и многого другого.

В состав системы включен ряд булевых операций, таких как **Объединение**, **Вычитание** или **Пересечение тел**, есть также возможность **Масштабировать деталь**.

Постоянное развитие функционала системы КОМПАС-3D позволяет постепенно заменять используемые ранее операции твердотельного моделирования на формообразование с помощью поверхностного моделирования в случае работы со сложными пространственными формами. Особенно это касается замены **Операции по сечениям** на **Линейчатые поверхности** и **Поверхности по сети кривых**. Тем самым построение, например, корпусов судов, фюзеляжей летательных аппаратов существенно облегчается.

Далее мы рассмотрим построение одной из деталей в основном с помощью операций твердотельного моделирования, а затем построим другую деталь с помощью поверхностного моделирования. В продолжение главы мы познакомимся с построением детали, изготавливаемой из листового материала, а также с получением развертки. Именно развертка и используется впоследствии при разработке управляющей программы.

# 13.1. Твердотельное моделирование

Формирование геометрического представления детали с помощью твердотельного моделирования рассмотрим на примере корпуса водозапорного клапана (рис. 13.3).

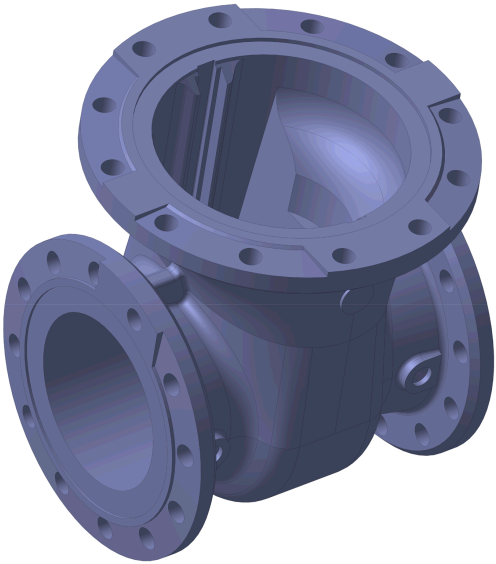


*Рис. 13.3. 3D-модель детали «Водозапорный клапан»*

Начнем работу с создания верхнего фланца. Для этого выберем плоскость, в которой вычертим эскиз основания фланца – окружность с диаметром 430 мм. Рекомендуется в качестве исходной плоскости выбирать одну из базовых конструктивных плоскостей модели. В нашем случае это плоскость Z0X. Эскиз выполняется стилем линии **Основная**, ассоциативный диаметральный размер позволит в дальнейшем управлять изображением с помощью изменения значения размера. Размеру можно также назначить переменную и сделать ее **Внешней**. Меняя ее значение, – даже не редактируя самого эскиза – можно управлять геометрией детали (рис. 13.4).

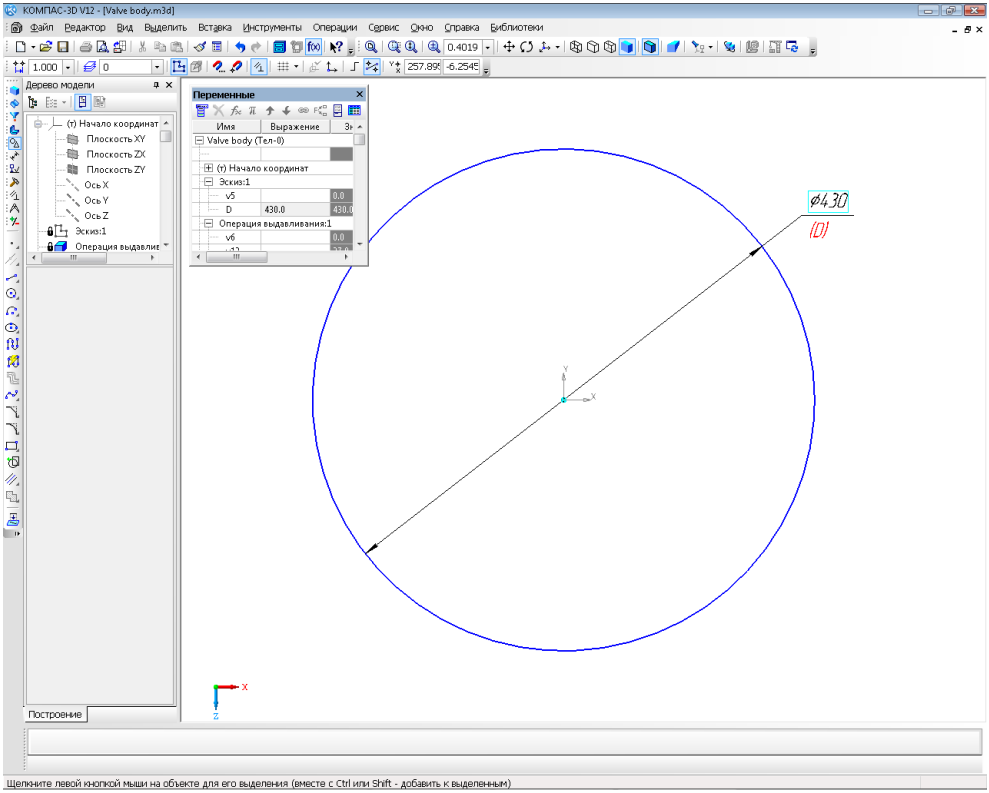



*Рис. 13.4. Первая операция – создание окружности*

Первой операцией моделирования станет **Операция выдавливания**. Мы можем выбрать направление выдавливания относительно исходной плоскости эскиза, величину выдавливания в этом направлении (в нашем случае – 37 мм). При необходимости можно установить уклон боковых граней получаемого тела для имитации литейных или иных технологических уклонов (рис. 13.5).

Далее нам необходимо добавить к построенному фланцу часть корпуса, представляющую собой переход с переменными овальными поперечными сечениями.

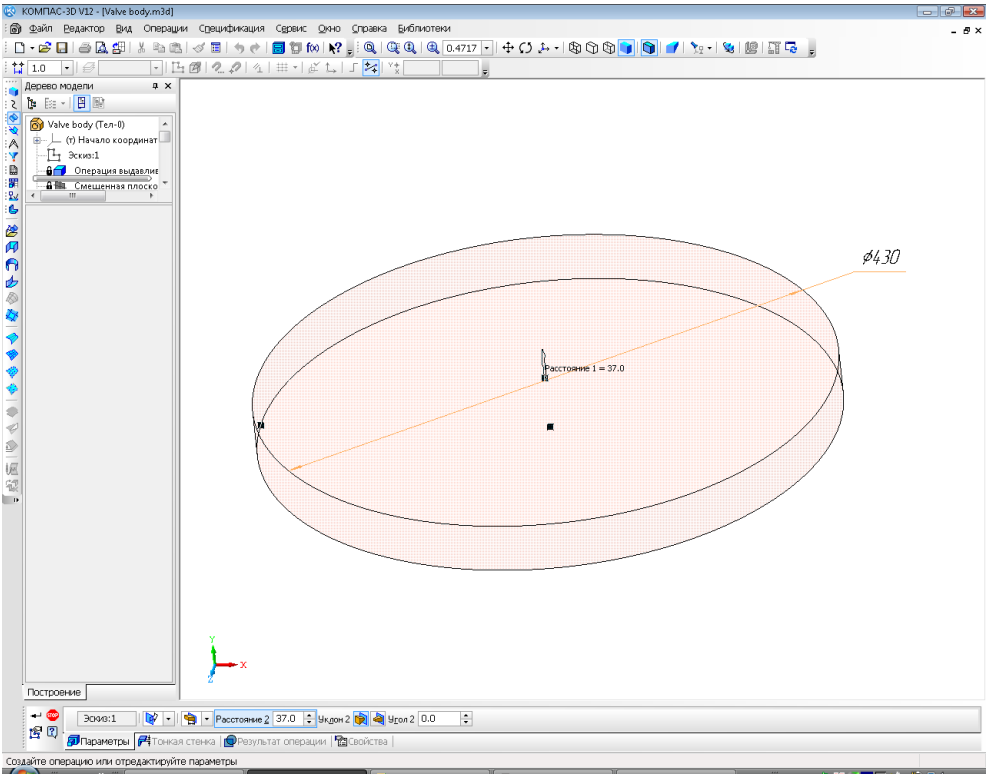



*Рис. 13.5. Применение операции выдавливания*

Для построения эскизов поперечных сечений нам нужно сначала построить несколько смещенных плоскостей, в которых последовательно отрисовываются поперечные сечения. Смещенные плоскости мы построим параллельно одному из оснований ранее построенного фланца. Желательно, чтобы формы поперечных сечений содержали равное количество точек переходов между кривыми. В этом случае построение будет корректным.

После того как построены поперечные сечения, воспользуемся операцией **Приклеить элемент по сечениям**. Выделим в дереве детали (или прямо в модели) последовательно построенные сечения и выполним данную операцию. В результате мы получим новое построение (рис. 13.6).

Аналогичные действия можно использовать, чтобы смоделировать продолжение перехода. Однако мы «усложним» себе задачу. Если внимательно изучить модель корпуса, можно увидеть, что в рассматриваемом нами сейчас продолжении перехода можно создать только «четвертинку» объема, а затем «повторить» ее зеркальными отображениями. Для создания «четвертинки» создадим параллельную торцу ранее построенного перехода смещенную плоскость, затем в плоскости торца и в смещенной плоскости отрисуем эскизы поперечного сечения «четвертин-

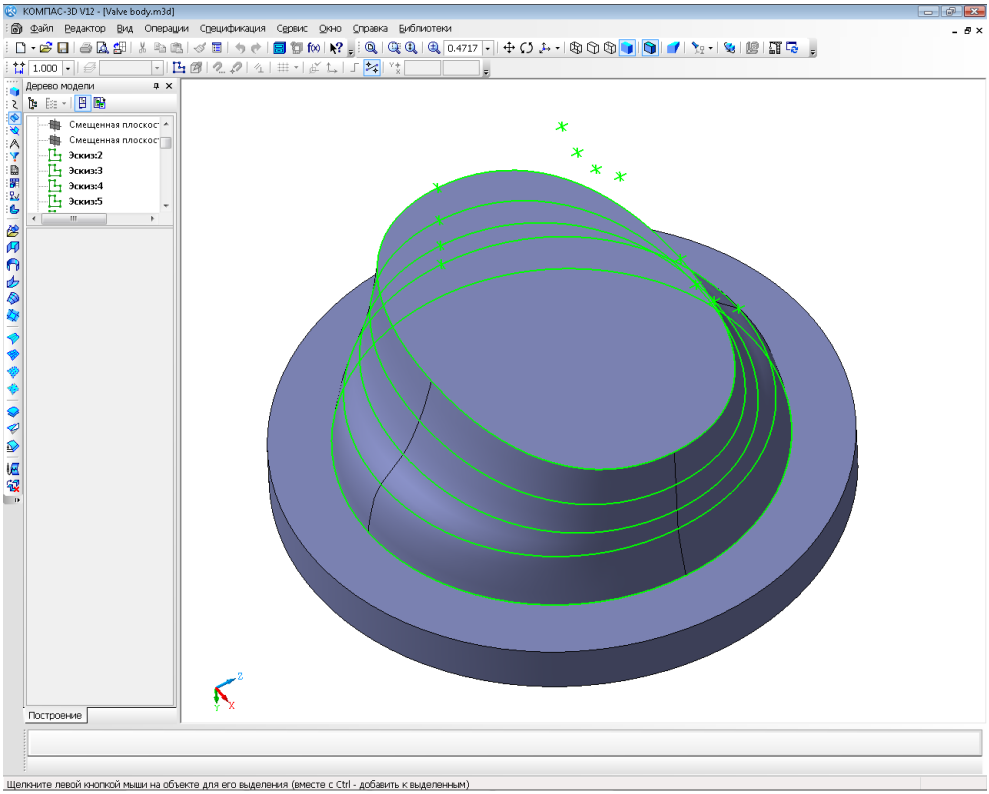



*Рис. 13.6. Добавление части корпуса*

ки». После создания эскизов с помощью команды **Приклеить элемент по сечениям** смоделируем ее (рис. 13.7).

Теперь, как мы и хотели, выполним зеркальное отражение полученного объема относительно одной из вертикальных плоскостей. Для этого применим команду **Зеркальный массив**. Выберем команду, выделим в дереве построения детали результат предыдущей операции (**Приклеить элемент по сечениям**), а затем укажем мышью вертикальную плоскость, относительно которой мы хотим отобразить объем. В результате мы получим уже не четвертинку объема, а ее половину (рис. 13.8).

Повторно произведя такую операцию, закончим построение продолжения перехода. Мы можем увидеть, что команда **Зеркальный массив** применима и к результату другой такой же операции, выполненной ранее.

Нижняя часть корпуса напоминает половинку цилиндра, «приставленную» к торцу ранее полученного объема. Цилиндр как тело вращения мы и получим с помощью операции **Приклеить вращением**. Сначала на торце детали построим эскиз, который затем будем вращать. Этот эскиз можно построить вручную, однако в нашем случае гораздо проще получить его, спроецировав на плоскость

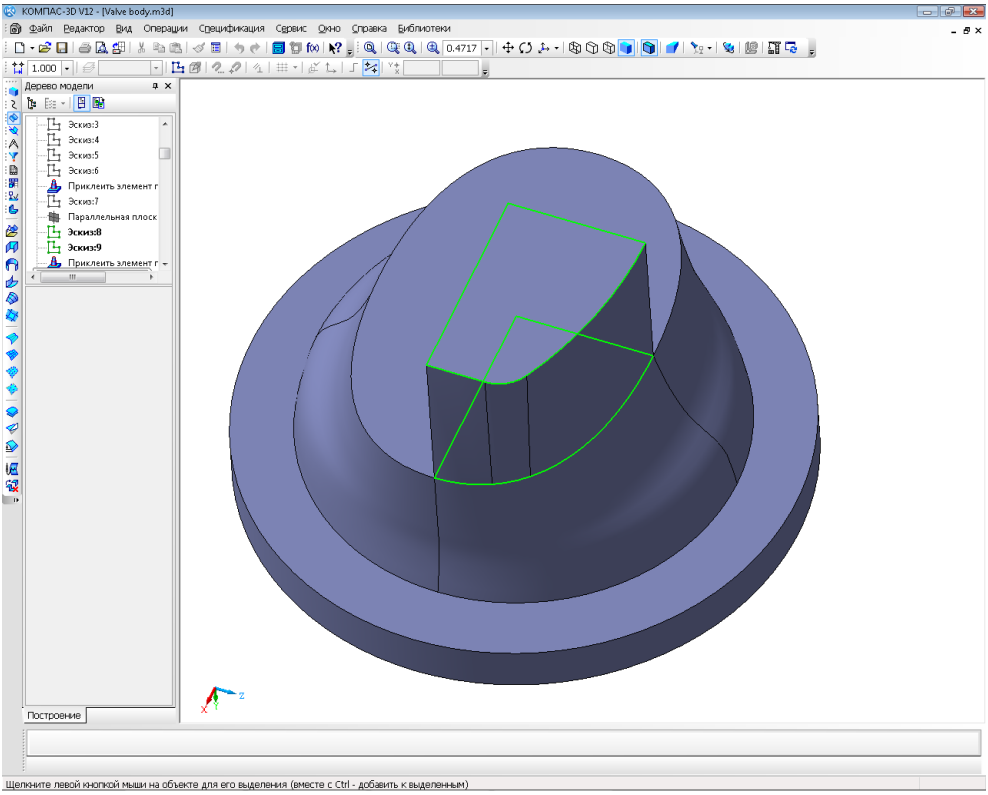



*Рис. 13.7. Создание «четвертинки»*

торца половинку его контура с помощью операции **Спроецировать объект**. Также в эскизе необходимо начертить ось вращения стилем линии **Осевая**. Теперь можно приступить к формообразованию. На рис. 13.9 мы видим, что получилось в итоге.

Аналогично, с помощью **Операции вращения** добавляем два поперечных фланца. Поскольку проектируемая деталь наверняка будет изготавливаться с помощью литья, то необработанные (и не обрабатываемые в дальнейшем) переходы между гранями должны быть скруглены. Для выполнения плавного перехода между поверхностями детали применим функцию **Скругление** (рис. 13.10). Мы указываем скругляемое ребро, можем задать тип скругления – постоянным или переменным радиусом, типы обработки сложных случаев прохода стыкующихся ребер, а также определяем некоторые другие параметры.

Формообразование внутренних полостей, карманов, выступов и других элементов мы выполним с помощью набора ранее рассмотренных операций. А вот для создания отверстий под крепеж на фланцах нашего корпуса мы применим не операцию вырезания выдавливанием, а воспользуемся современным инструментом – **Библиотекой стандартных изделий**. Пусть вас не пугает в ее названии слово ИЗДЕЛИЯ. В нашем случае, при моделировании конструктивных элементов де-

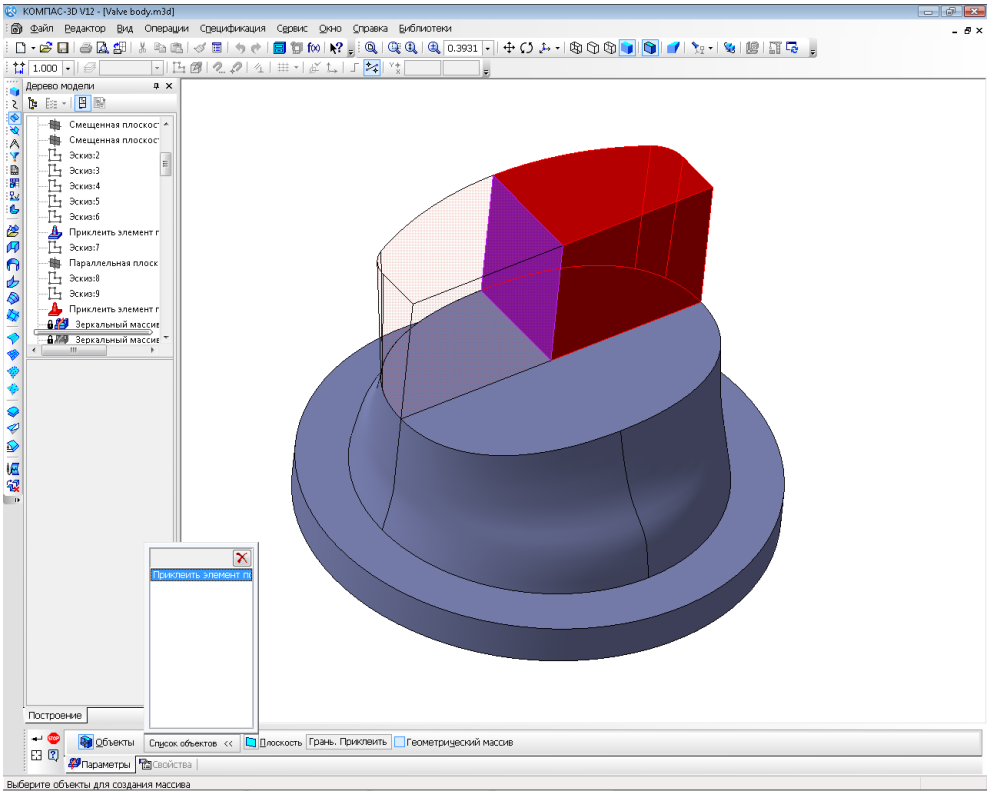



*Рис. 13.8. Применение операции* ***Зеркальный массив***

тали, мы выберем из библиотеки именно один из вариантов подобных элементов – **Отверстие цилиндрическое**. Нам понадобится указать две поверхности, которые ограничивают отверстие, ввести его диаметр, задать координаты (в декартовой или полярной системе) либо просто указать точку, ранее построенную в эскизе на плоскости торца фланца. В дереве построения детали отверстие будет отображено не как операция, а как специальный макроэлемент с возможностью редактирования (рис. 13.11).

Вообще говоря, в **Библиотеке стандартных изделий** мы находим не только отверстия (цилиндрические, конические, резьбовые, центровые и др., но также массу очень часто применяемых конструктивных элементов:

- канавки для выхода различного режущего инструмента;
- канавки для размещения уплотнений и уплотнительных колец;
- проточки под запорные кольца;
- проточки для выхода резьб (трубной, метрической, конической, трапецеидальной);
- прямобочные, треугольные, эвольвентные шлицы;
- шпоночные пазы под призматические и сегментные шпонки.

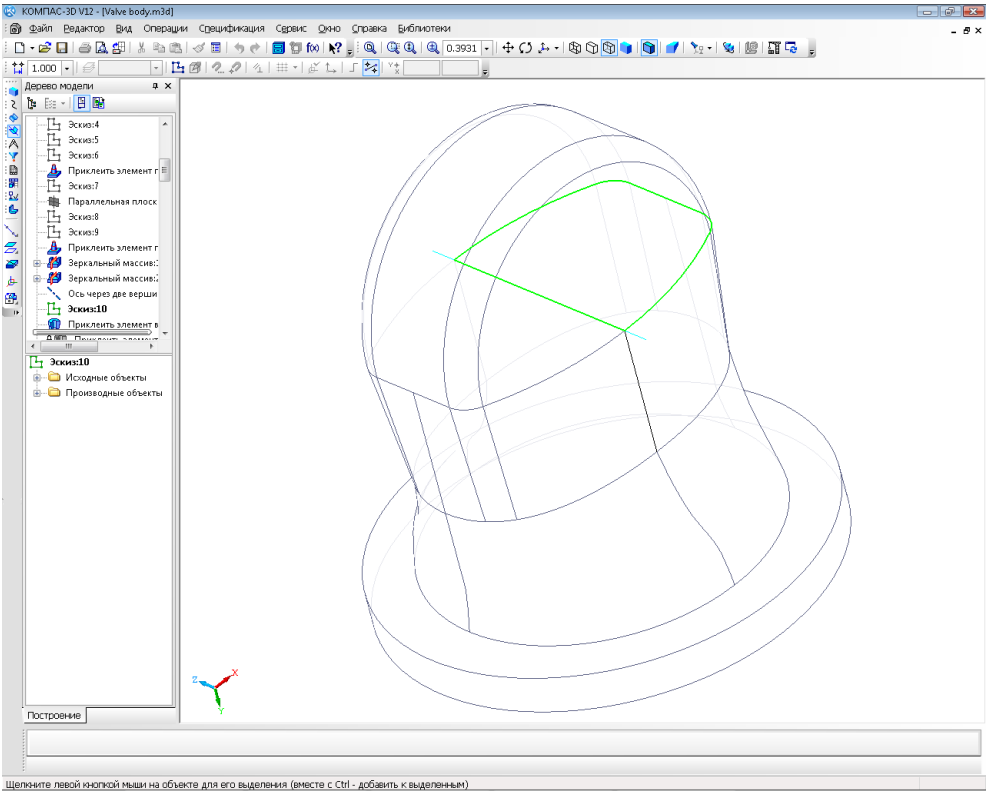



*Рис. 13.9. Результат операции вращения эскиза*

Каждый из этих элементов – не просто набор операций, собранных вместе, – это также и параметрический объект. Причем в **Библиотеке** присутствуют все стандартизованные ряды размеров и параметров.

Как мы уже упоминали выше, для создания одинаковых элементов можно использовать механизм создания массивов. В данной ситуации логично применить команду **Массив по концентрической сетке**. Для выполнения операции мы выберем в дереве ранее созданное отверстие, в параметрах зададим количество копий по окружности, а также укажем на ту цилиндрическую поверхность, ось которой можно использовать в качестве оси операции. В результате мы получим параметризованный массив отверстий (рис. 13.12), который можно отредактировать в любое время. Мало того, при необходимости мы можем удалить часть экземпляров массива или вернуть ранее удаленные.

Постепенно наращивая или удаляя материал с помощью стандартных команд твердотельного моделирования, мы получили почти готовую модель корпуса с тремя фланцами, на которых расположены массивы крепежных отверстий. Для получения большей жесткости и прочности изделия желательно как бы «подкрепить» ряд конструктивных элементов. Например, объединить ребрами жесткости

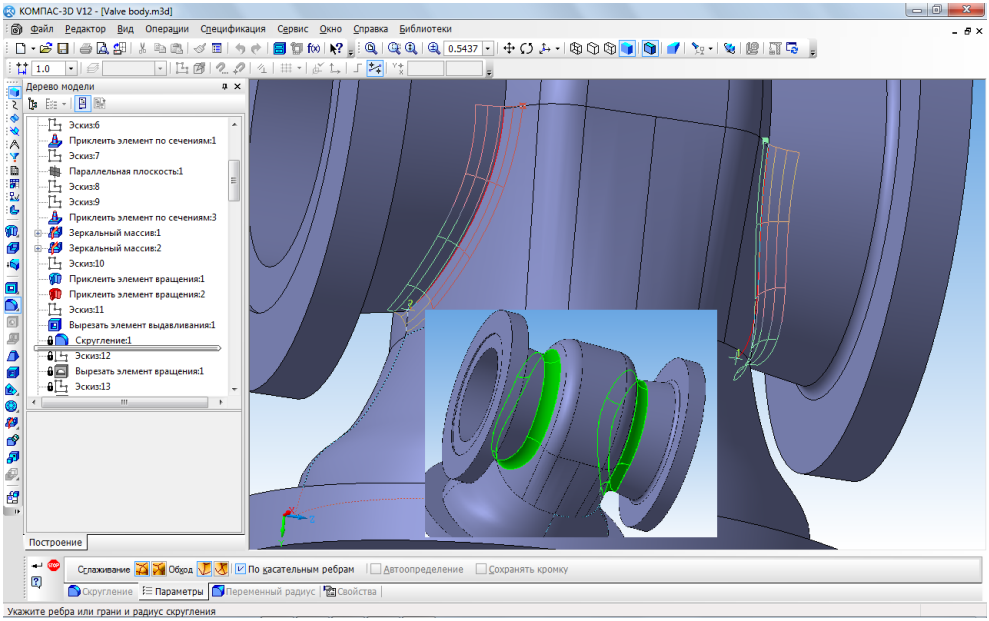



*Рис. 13.10. Применение операции скругления*

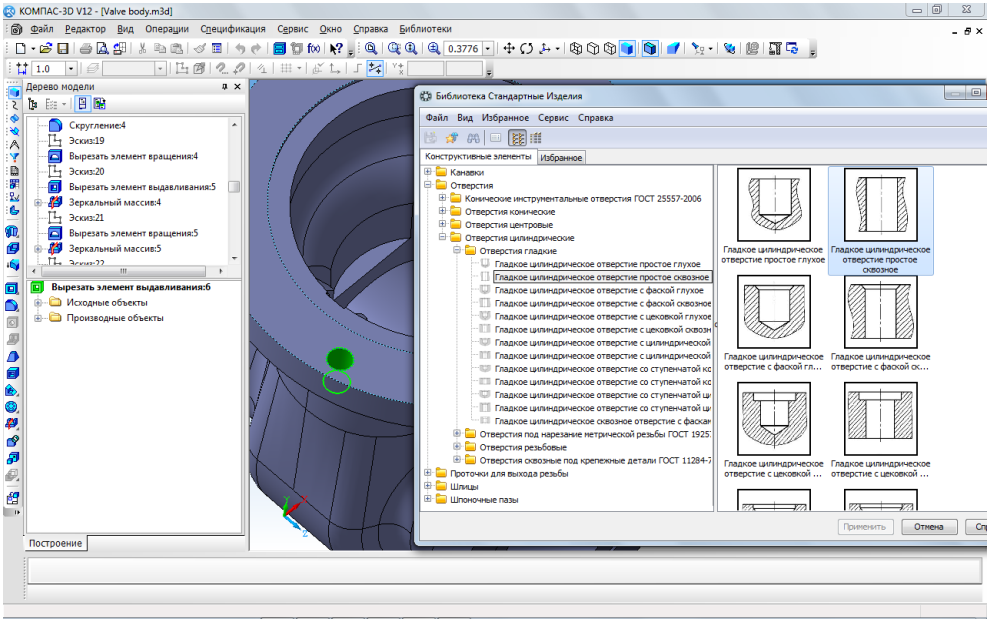



*Рис. 13.11. Редактирование параметров отверстия*

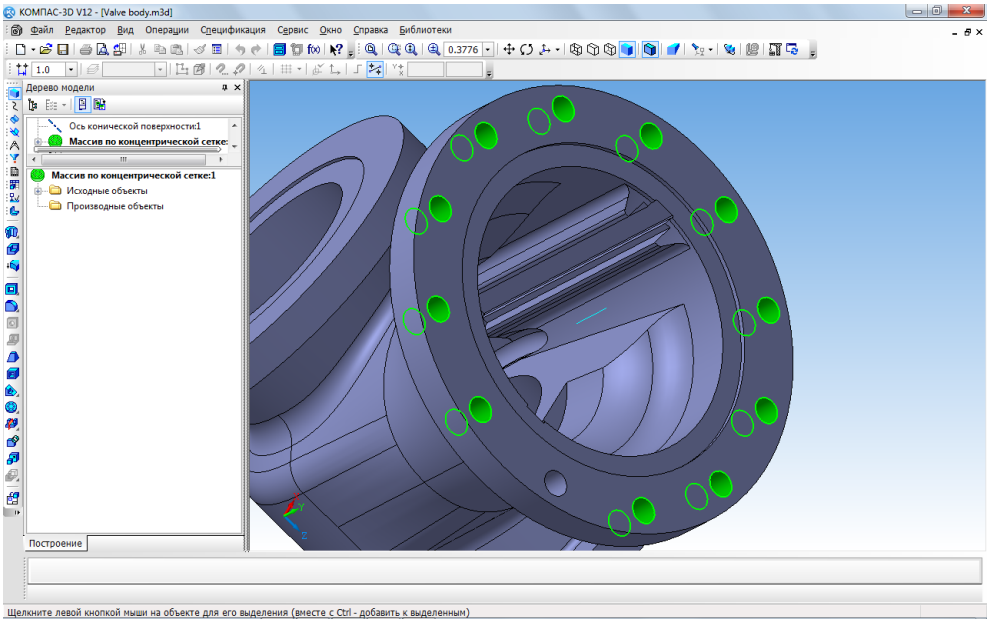



*Рис. 13.12. Массив отверстий*

переходы от основного корпуса к фланцам. Обычно одна из сторон ребра жесткости (со стороны «образующей») имеет довольно простую форму, зато примыкание ребра к граням детали происходит по сложным поверхностям. Построение ребра как набора обычных операций выдавливания – зачастую долгий и кропотливый процесс.

Поэтому воспользуемся отдельно выделенной командой построения ребер. Для начала необходимо выбрать в модели или построить вспомогательную плоскость, где мы отрисуем эскиз образующей ребра жесткости. Самое приятное состоит в том, что концевые точки эскиза необязательно привязывать к элементам детали. Достаточно, чтобы концы образующей коснулись (или немного вошли внутрь) тела детали. Сам эскиз представляет в нашем случае отрезок в плоскости Z0Y (рис. 13.13).

После того как эскиз начерчен, выполним команду **Ребро жесткости**, введем толщину ребра и при необходимости зададим углы уклонов. Система сама определит направление, в котором ребро будет построено. Если по каким-либо причинам система не может «угадать» направление, его легко указать вручную.

Осталось, как говорится, немного. С помощью уже известных операций приклеивания выдавливанием, вставки отверстия, скругления и зеркального массива добавляем в деталь бобышки (рис. 13.14). Таким образом, мы построили корпус клапана.

Всегда интересно посмотреть внутреннее устройство детали. Если ранее нам приходилось строить на чертеже разрезы и сечения, причем отрисовка внутренних

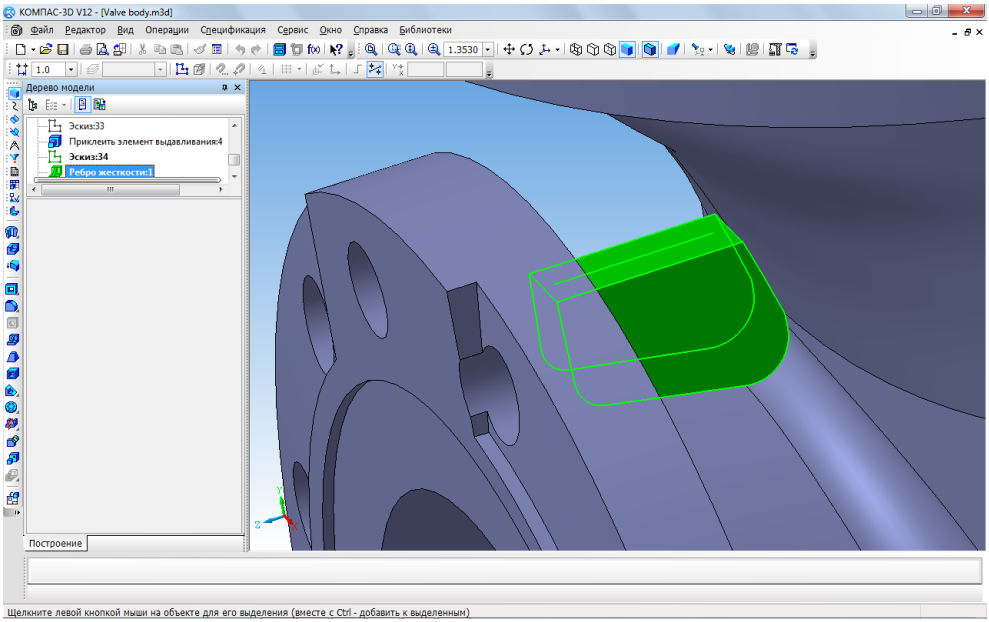



*Рис. 13.13. Построение ребра*

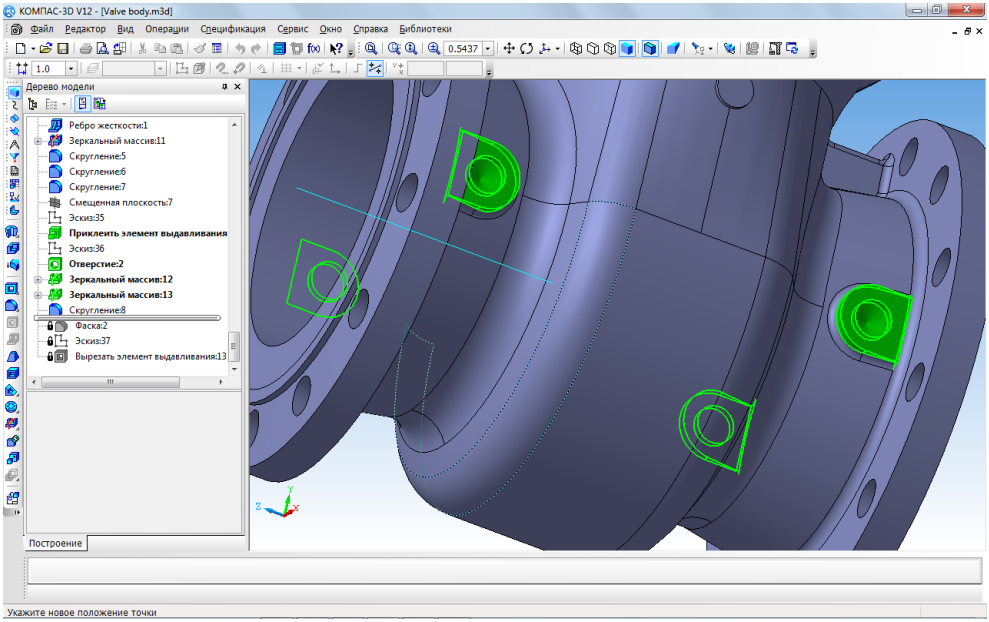



*Рис. 13.14. Добавление бобышек*

элементов и полостей была довольно затруднительна, – имея трехмерное представление детали, все аналогичные операции становятся исключительно простыми. Выберем плоскость Z0Y и выполним команду **Сечение поверхностью**. Деталь будет рассечена указанной плоскостью (рис. 13.15).

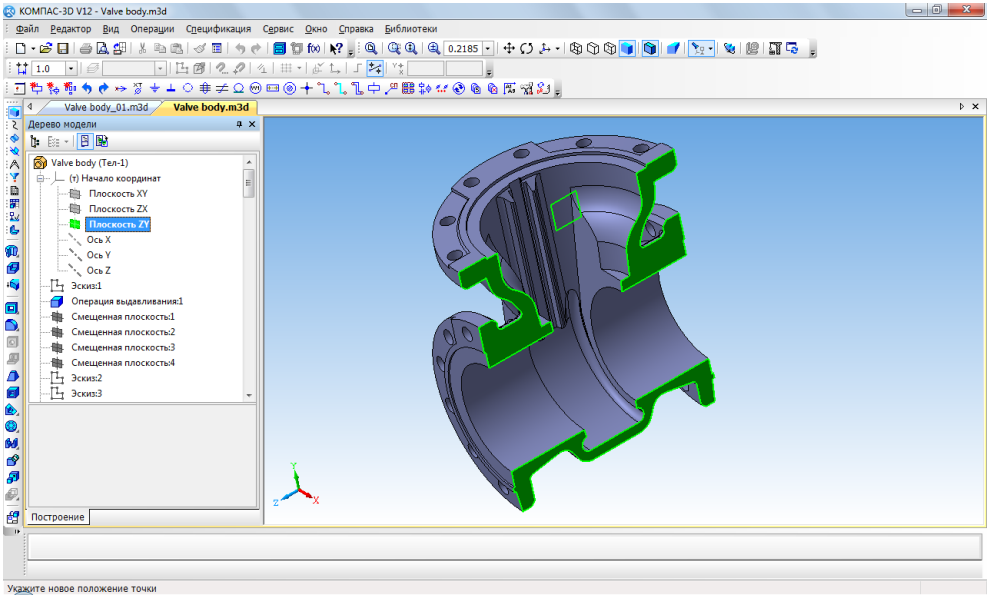



*Рис. 13.15. Сечение построенной 3D-модели*

# 13.2. Поверхностное моделирование

Как мы упомянули в начале главы, некоторые твердотельные формообразующие операции в последней версии КОМПАС-3D можно заменить операциями поверхностного моделирования. В ряде случаев – например, при создании концевых обтекателей корпуса судна, зализов в месте стыковки центроплана и фюзеляжа самолета – твердотельные операции практически неприменимы.

В этом разделе мы увидим, как поверхностное моделирование применяется при создании того же корпуса клапана, а затем используем их по «прямому назначению».

Начнем моделирование корпуса клапана точно так же, как и в 1 части, – с создания фланца (рис. 13.4, 13.5). Однако затем мы не будем создавать смещенные плоскости, не будем строить набор поперечных сечений. Вместо этого (рис. 13.16) в плоскости Z0Y мы создадим эскиз (эскиз 38), в котором вычертим одну из кривых – образующих перехода. В плоскости X0Y мы вычертим вторую образующую (эскиз 39). На основании фланца вычертим дугу (эскиз 40), которая ограничивает переход «снизу», а на единственной смещенной плоскости вычертим вторую дугу (эскиз 41), ограничивающую переход «сверху». Таким образом, мы получили сеть

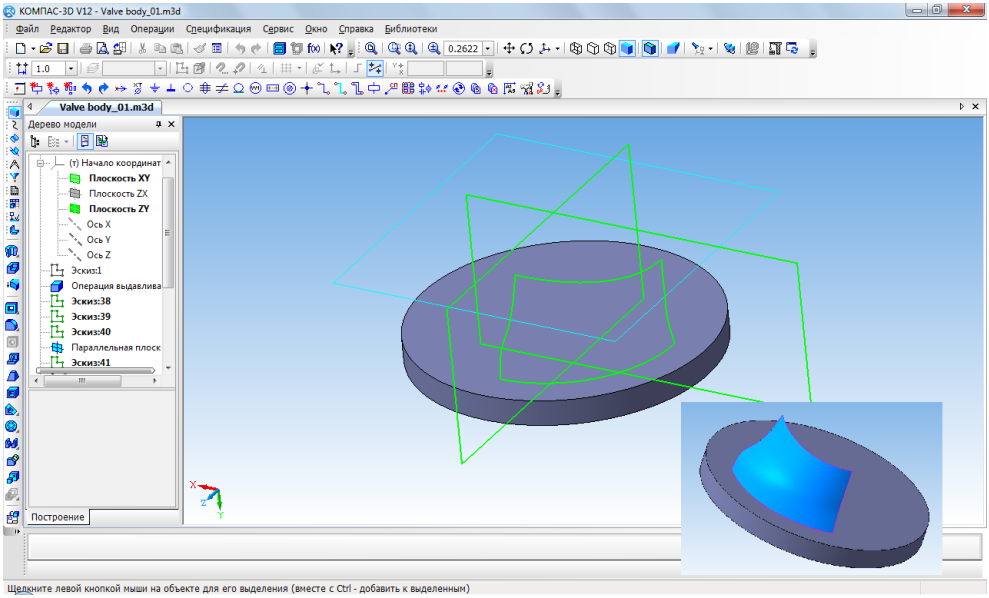



*Рис. 13.16. Создание кривых как основы для поверхностной модели*

кривых, по которым создадим поверхность, воспользовавшись командой **Поверхность по сети кривых**. Кривые в эскизах 38 и 39 указываем для направления U, кривые в эскизах 40 и 41 – в направлении V.

Теперь с помощью команды **Зеркально отразить тело или поверхность** полученную поверхность «размножим» (рис. 13.17).

Мы получили открытую полость, которую нужно «заполнить» материалом. Проще всего это можно сделать с помощью команды **Сшивка поверхностей с одновременным созданием тела**. Однако открытую полость необходимо «закрыть», наложив сверху и снизу **Заплатки**. Создадим такие заплатки, указав на ребра, ограничивающие полость. Получим теперь объем из четырех «боковых» поверхностей и двух торцевых. И вот сейчас применим команду **Сшивка поверхностей**. Чтобы убедиться в том, что объем заполнен материалом, «вырежем» в детали «четверть», построив эскиз в виде прямого угла и применив операцию **Сечение по эскизу** (рис. 13.18).

Дальнейшее построение детали можно произвести так же, как и в части 1.

А теперь построим новую деталь, формы которой сложно и трудоемко (если вообще возможно) получить с помощью твердотельного моделирования. Это модель ручки безопасной одноразовой бритвы (рис. 13.19).

Построение начинается с отрисовки ряда эскизов, определяющих внешние контуры бритвы. Форма и размеры эскизов задаются дизайнером. На рис. 13.20 приведены эскизы будущих контуров. Затем мы строим **Поверхности выдавливания**, которые частично будут использованы как внешние поверхности, но в основном мы применим их в качестве вспомогательных.

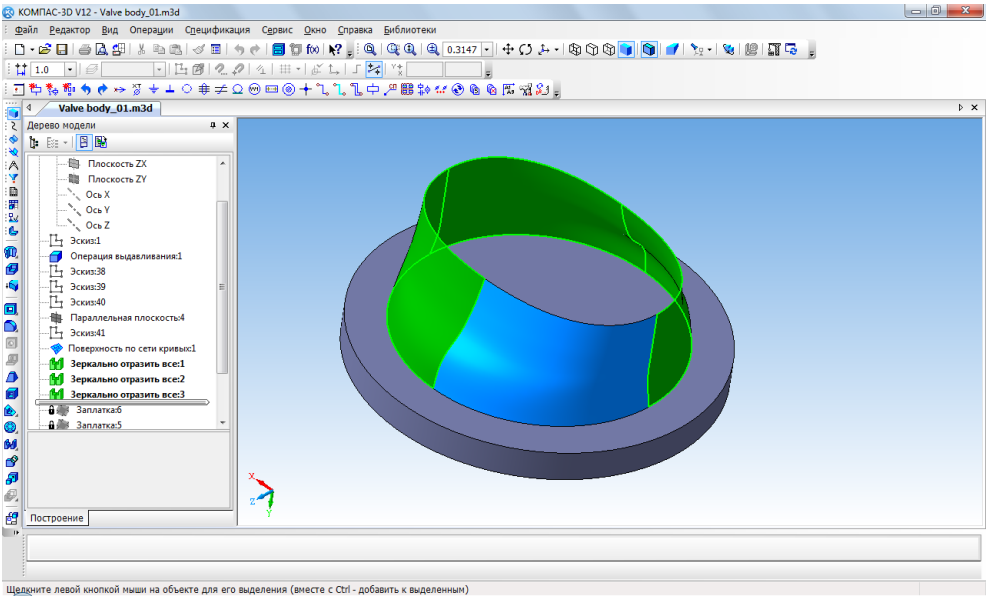



*Рис. 13.17. Размножение поверхности*

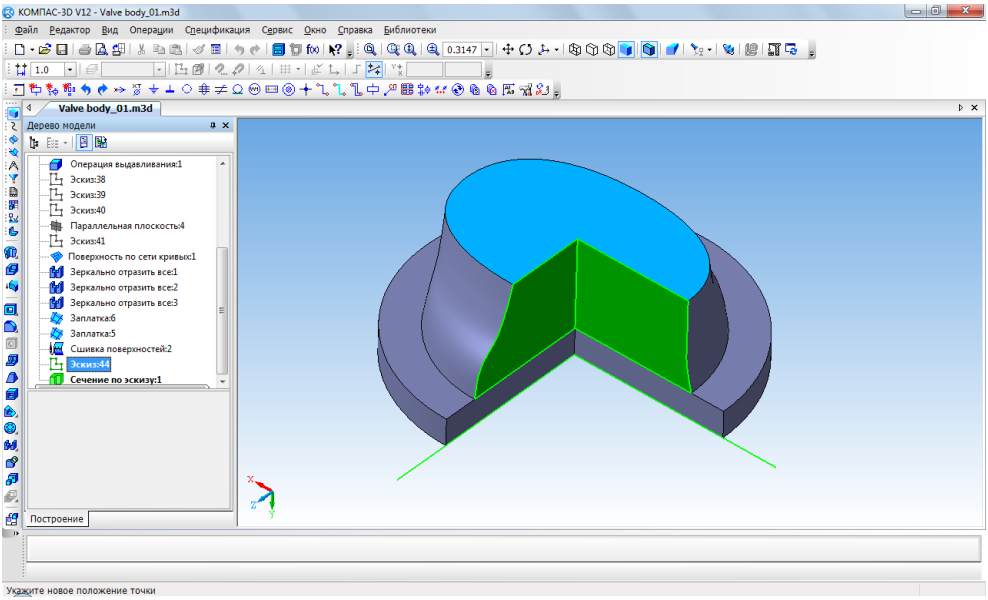



*Рис. 13.18. Применение операции* ***Сечение по эскизу***

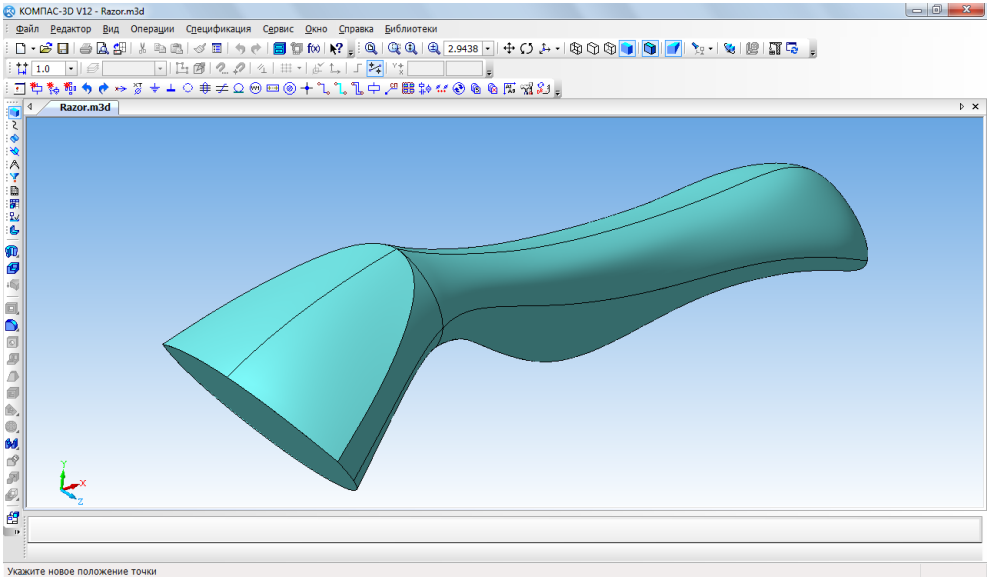



*Рис. 13.19. 3D-модель ручки безопасной одноразовой бритвы*

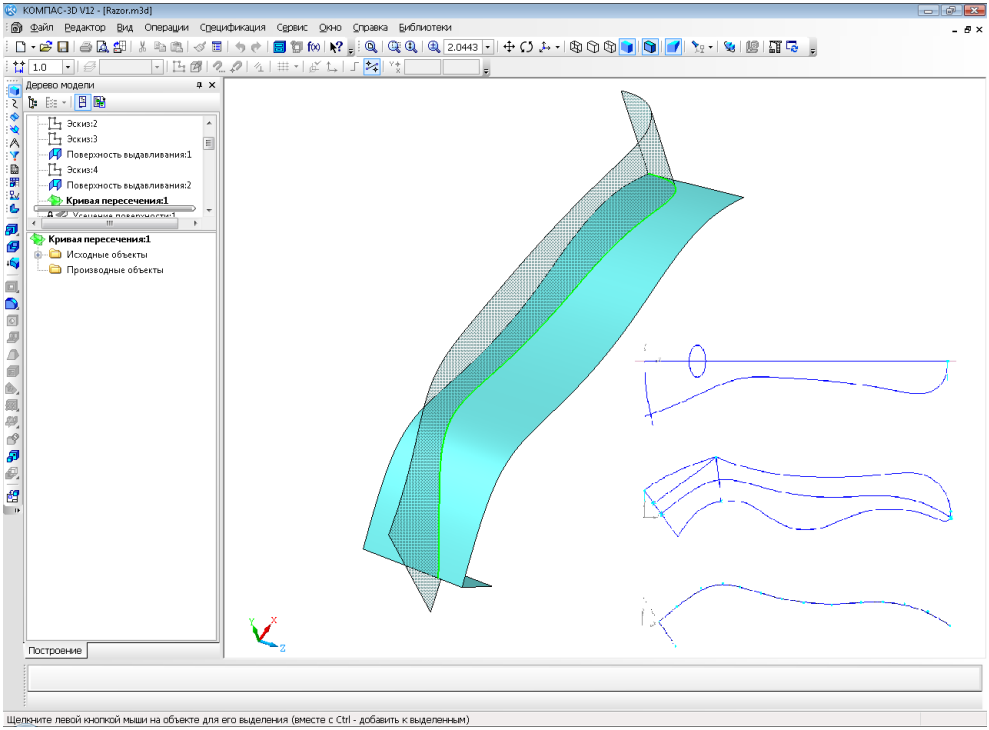



*Рис. 13.20. Форма и размеры эскизов задаются дизайнером*

Линия пересечения поверхностей выдавливания послужит основой для построения части ручки бритвы. Для получения этой пространственной кривой нужно воспользоваться новой командой **Кривая пересечения поверхностей**. Дополнительными построениями мы создадим еще несколько кривых, служащих образующими ручки. И теперь пришло время применить уже знакомую нам по первой части команду построения **Поверхности по сети кривых**. В качестве базовых кривых мы указываем линии пересечения ранее построенных поверхностей.

Таким образом, мы получили поверхностную модель половины ручки. Теперь зеркально отразим все получившиеся поверхности, чтобы получить полную модель. Затем на открытый торец наложим **Заплатку** (рис. 13.21).

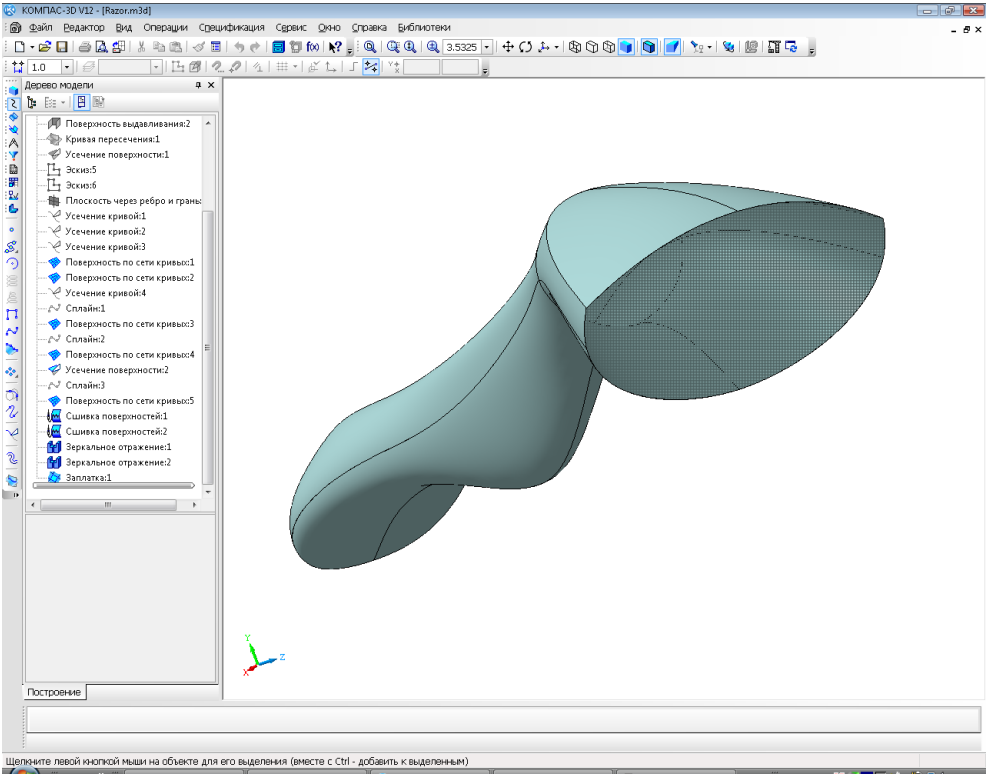



*Рис. 13.21. Наложение поверхности-заплатки*

Поверхностную модель можно при необходимости «превратить» в твердое тело, произведя **Сшивку поверхностей детали с образованием тела**. Настраивая свойства модели, мы можем выбрать материал для нее как из текстового файла плотностей GRAPHIC.DNS, так и из библиотеки **Материалы и сортаменты** (рис. 13.22). Библиотека содержит обширный перечень материалов и сортамен-

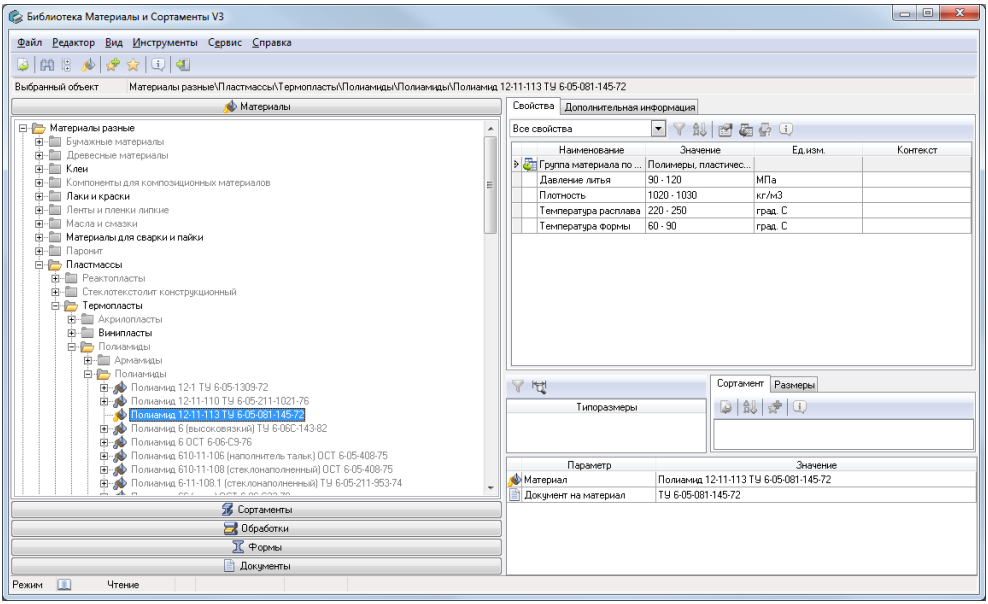



*Рис. 13.22. Библиотека* ***Материалы и сортаменты***

тов, информацию о свойствах материалов, назначении и области применения, заменителях и условиях замены, информацию по нормативным документам и т. д. Она предоставляет конструктору следующую информацию:

- обозначения и документы на поставку черных и цветных металлов, их сплавов и неметаллических материалов (в том числе строительных);
- физико-механические, технологические свойства конструкционных материалов, их назначение и области применения;
- виды сортамента (фасонного, листового, профильного и т. п.), изготавливаемого из этих материалов, включая перечни типоразмеров, выпускаемых промышленностью;
- марки смазочных материалов с данными по их свойствам и областям применения;
- марки лакокрасочных, металлических и неметаллических покрытий, включая характеристики и условия эксплуатации покрытий;
- таблицы соответствия российских и зарубежных марок сталей.

Общее количество материалов в библиотеке составляет более 8500 наименований. В том числе:

- 1300 отечественных марок сталей и сплавов;
- 2120 зарубежных марок сталей;
- 145 марок чугунов;
- 725 марок цветных металлов и сплавов;
- 397 марок масел и смазок;

- 171 марка лаков и красок;
- 418 марок пластмасс;
- 464 марки клеев;
- 1200 наименований сварочных материалов;
- 567 наименований проводов и кабелей;
- 131 вид гальванических покрытий;
- 46 наименований композиционных материалов.

Кроме уже описанных команд поверхностного моделирования, необходимо упомянуть и о некоторых других, очень важных формообразующих операциях.

**Поверхность по сети точек.** Данная операция позволяет построить поверхностную модель по пространственным точкам, которые образуют условную пространственную сеть. Массив этих точек можно получить:

- построив точки на имеющейся грани детали командой **Группа точек по поверхности**;
- загрузив из текстового файла или из файла электронных таблиц.

Пример поверхности, созданной по сети точек, показан на рис. 13.23 (справа).

**Поверхность по пласту точек.** Команда позволяет построить поверхностную модель по пространственным точкам, которые представляют неупорядоченный массив. Такой массив можно загрузить из текстового файла или из файла электронных таблиц. Сначала КОМПАС анализирует координаты точек массива и на их основе выстраивает **Сеть точек**, а затем автоматически строится поверхность по получившейся сети. Пример поверхности, созданной по пласту точек, также показан на рис. 13.23 (слева).

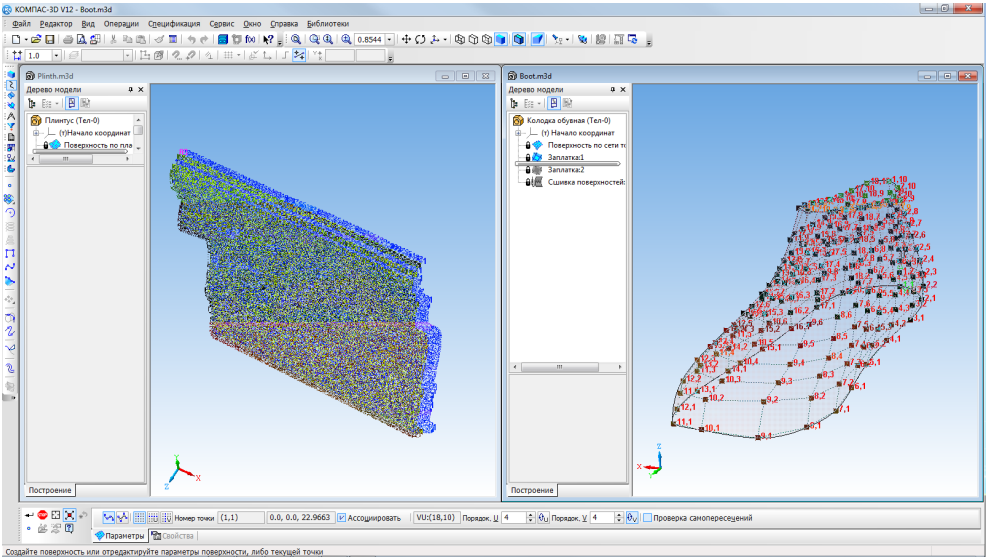



*Рис. 13.23. Поверхности по точкам*

# 13.3. Моделирование деталей из листового материала

В этой части главы мы познакомимся с построением приборного корпуса (рис. 13.24), который изготавливается из листового металла толщиной 1 мм.

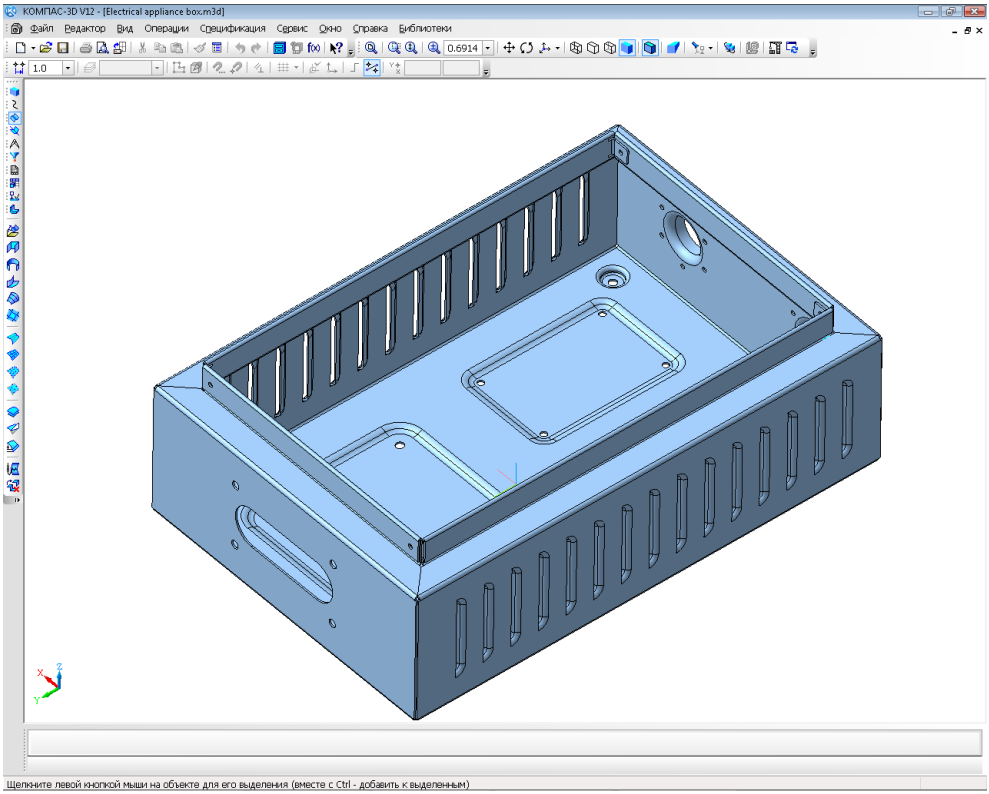



*Рис. 13.24. Корпус прибора из листового материала*

В качестве эскиза основания корпуса вычертим обычный прямоугольник размерами 200 × 300 мм. Проставим управляющие размеры, с помощью которых впоследствии сможем менять габариты будущей детали. После окончания работы с эскизом применим базовую команду работы с листовым материалом – **Листовое тело**. Зададим толщину листового материала, выберем на будущее способ определения длины развертки (**Коэффициент нейтрального слоя**, **Величина сгиба**, **Уменьшение сгиба**, **Таблица сгибов**) и создадим деталь (рис. 13.25).

Теперь добавим к основанию стенки. Стенки формируются как **Сгибы**, с использованием соответствующей операции. Необходимо указать ребро основания, к которому добавляется сгиб, а также определить параметры операции – **Угол сги-**

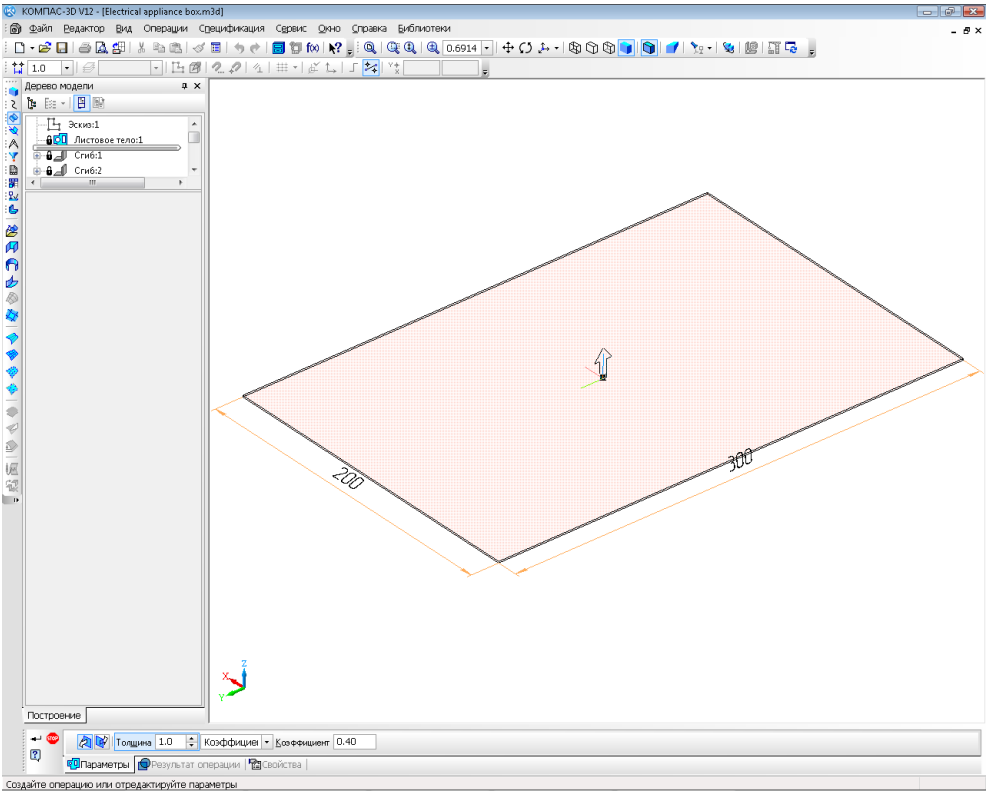



*Рис. 13.25. Эскиз основания корпуса*

**ба**, **Радиус сгиба**, **Длина сгиба**, **Уклоны боковых ребер сгиба** и т. п. Сгиб можно построить как вдоль всего ребра основания, так и на определенной его части. Выбранные параметры запоминаются системой. В результате выполнения операции мы получим что-то вроде обычного металлического ящика или короба (рис. 13.26).

Постепенно добавляя сгибы, наращиваем «верхний» пояс жесткости, «прорезаем» различные отверстия и пазы. Нужно обратить внимание, что при создании самых первых сгибов у нас образовались зазоры между вертикальными стенками. Эти зазоры необходимо «удалить». Вообще говоря, сделать это можно было сразу после создания первых сгибов, но мы «упустили» момент. Однако ничего страшного не произошло, «зашьем» зазоры сейчас. Для этого воспользуемся командой **Замыкание углов**. Выбираем углы на модели, определяем способ замыкания (встык, с перекрытием, плотное) и тип обработки углов (по кромке или по хорде, с зазором или без). После выполнения этой командой зазоры в углах исчезают (рис. 13.27).

В системе КОМПАС-3D к операциям работы с листовым материалом принято относить и операции штамповки, хотя в «чистом» виде штамповка имеет свои особенности. Однако при небольших деформациях можно пренебречь утонением материала.

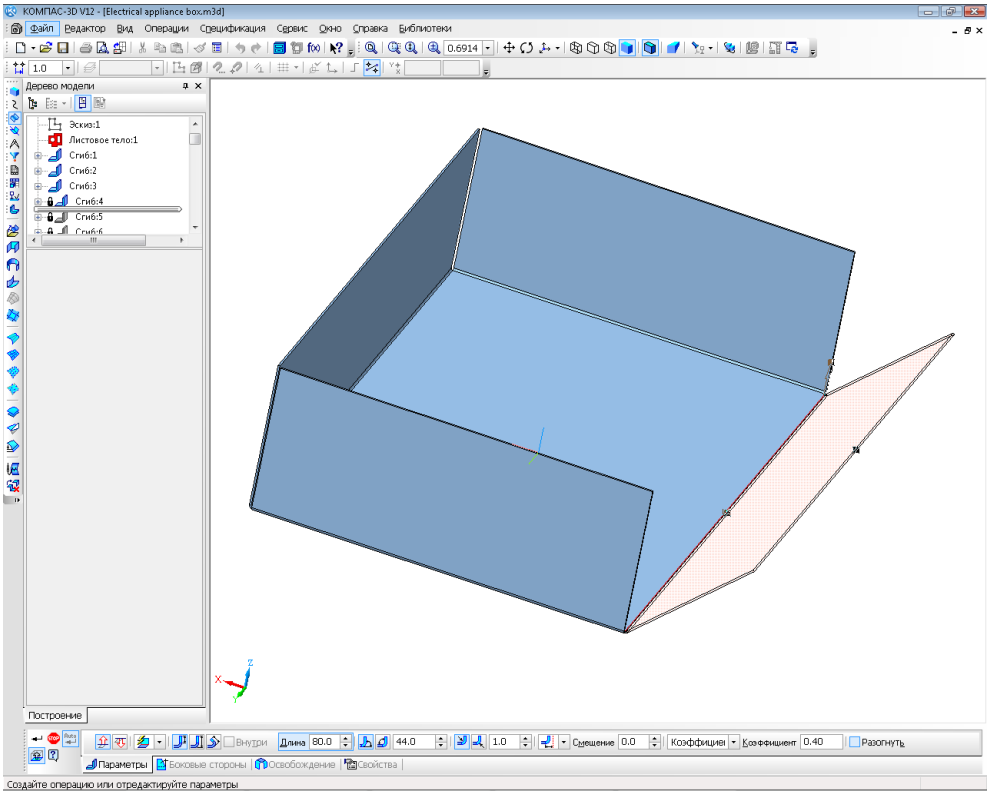



*Рис. 13. 26. Добавление стенок к основанию*

Сформируем жалюзи системы охлаждения на боковых гранях корпуса. Сначала вычертим на грани набор параллельных отрезков, которые определяют положение прорезей под жалюзи. Затем выбираем команду **Жалюзи**. Вводим тип жалюзи (вытяжкой или подрезкой), определяем параметры высоты жалюзи, радиус скругления и т. п. (рис. 13.28).

Для создания опорных ножек и углублений в днище корпуса можно воспользоваться командой **Закрытая штамповка**. На нижней грани нужно начертить окружность (для моделирования опоры) или прямоугольник (для моделирования углубления). Теперь обратимся к команде **Закрытая штамповка** и настроим параметры – зададим высоту выдавливания, уклон боковых граней, радиусы скруглений в местах перегибов листа, а также определим направление формирования элемента. Закончив операцию, мы увидим на дне корпуса созданные нами элементы (рис. 13.29).

На свободных боковых гранях нам необходимо создать отверстия для вывода пучков проводов и кабелей. Здесь как нельзя лучше подойдет операция **Открытая штамповка**. Она очень похожа по набору действий и параметров на предыдущую операцию закрытой штамповки, но в результате мы получаем не углубление, а отверстие с отбортовками (рис. 13.30).

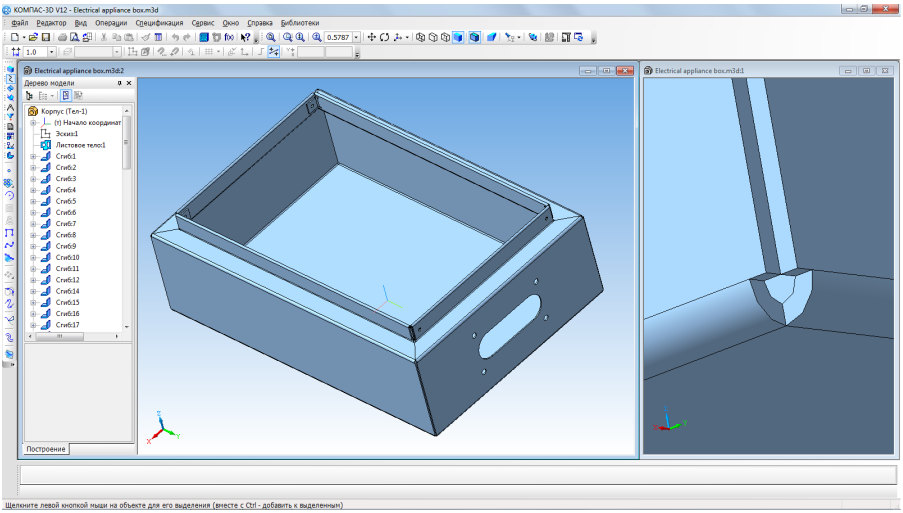



*Рис. 13.27. Применение операции **Замыкание углов***

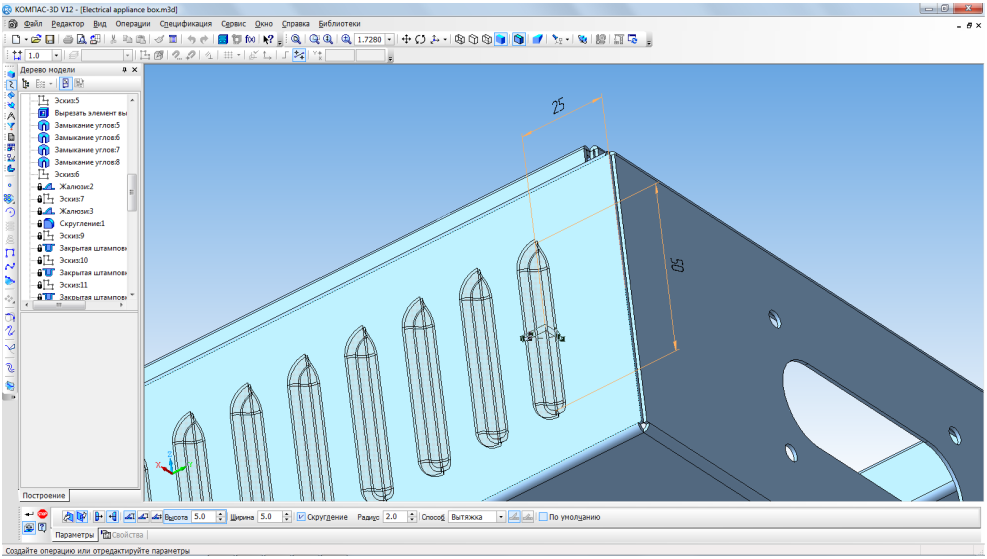



*Рис. 13.28. Формирование жалюзей*

Собственно говоря, на этом построение детали можно считать законченным. Осталось теперь подготовить развертку, контур которой мы потом передадим в САМ-систему для подготовки управляющей программы. Настроим параметры развертки, указав грань, которая будет базовой (это в нашем случае дно корпуса),

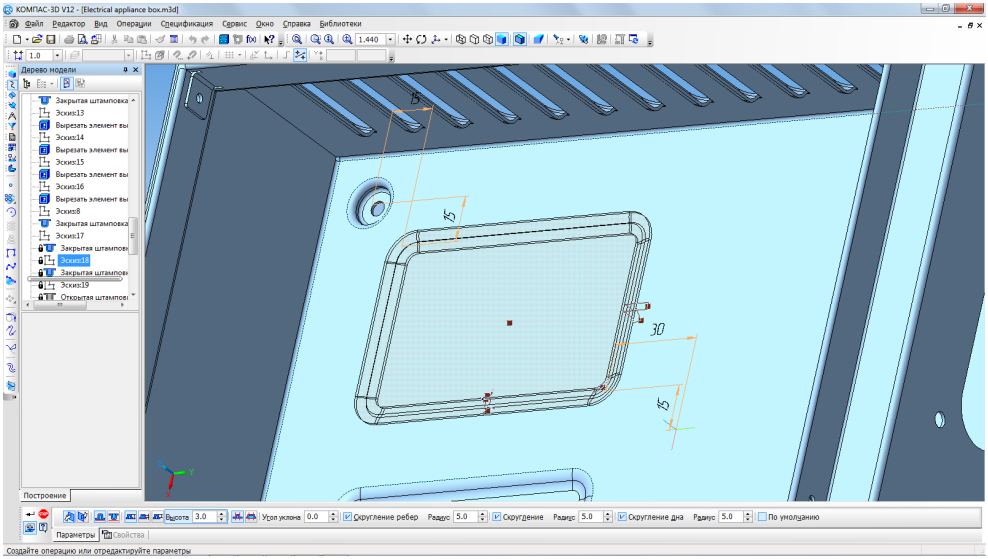



*Рис. 13.29. Создание опорных ножек и углублений в днище корпуса*

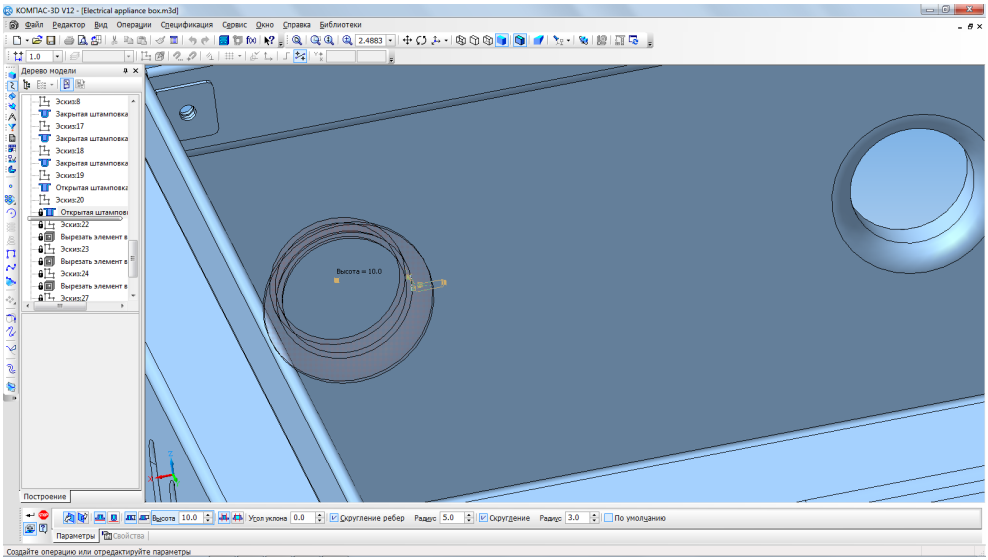



*Рис. 13.30. Отверстие с отбортовками*

и выбрав все сгибы. Теперь достаточно всего лишь нажать на команду **Развертка**, чтобы получить ее. В ассоциативном чертеже мы также можем на одном из видов показать наш корпус в развернутом состоянии и образмерить его (рис. 13.31).

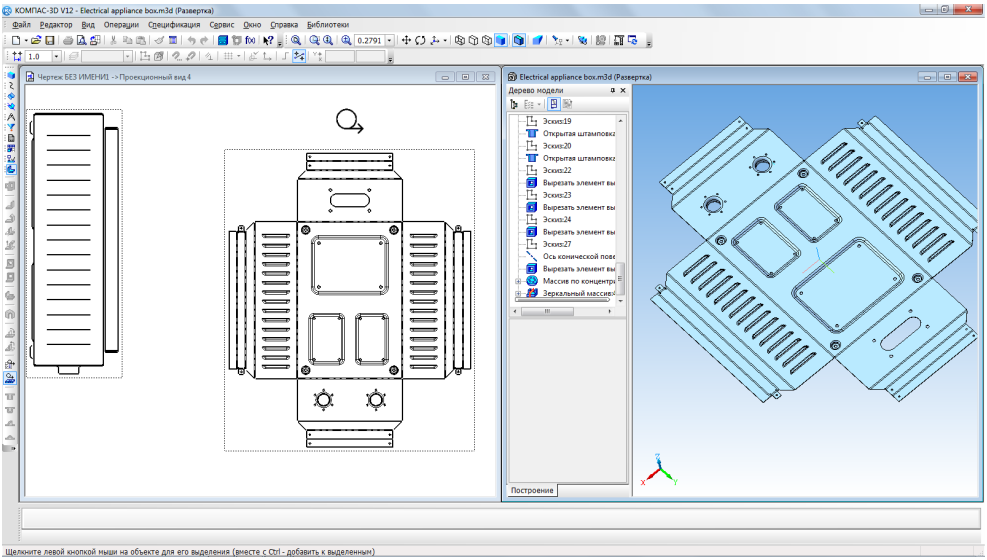



*Рис. 13.31. Результат применения операции **Развертка***

# 13.4. Экспорт геометрии

Большинство систем для подготовки управляющих программ способны принимать геометрические данные в универсальных обменных форматах. Среди них наиболее распространенными являются:

- IGES (*.igs) – Digital Representation for Communication of Product Definition Data (Цифровое представление для коммуникации данных определения продукта);
- STL (*.stl) – формат файла, используемый для хранения трехмерных моделей объектов для использования в технологиях быстрого прототипирования. Информация об объекте представляет собой список треугольных граней, которые описывают его поверхность. STL-файл может быть текстовым (ASCII) или двоичным;
- ACIS (*.sat) – формат описания данных коммерческого ядра геометрического моделирования, разрабатываемого и поддерживаемого компанией Spatial Corp.;
- STEP (*.stp) – самый современный универсальный формат описания данных, как геометрических, так и атрибутивных. Формат постоянно развивается и совершенствуется.

Система КОМПАС-3D поддерживает сохранение данных о трехмерных деталях во всех этих форматах. Кроме того, можно сохранять данные и в формате ядра Parasolid (*.x_t, *.x_b). Все эти конвертеры данных встроены в базовую конфигурацию системы при ее поставке и инсталляции. Экспорт данных представляет

собой вариант стандартной команды **Сохранить как…** и может дополнительно настраиваться в отдельном окне (рис. 13.32).

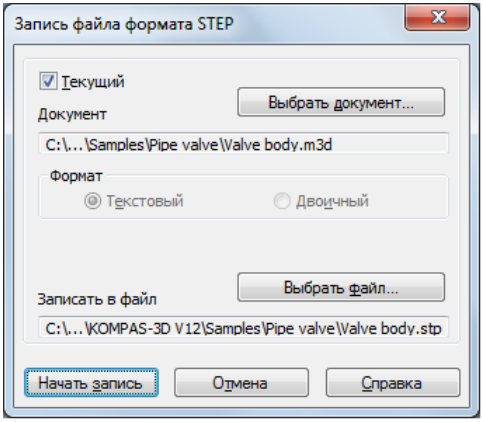



*Рис. 13.32. КОМПАС-3D умеет экспортировать геометрию в различные форматы*

Таким образом, мы рассмотрели различные варианты создания деталей в системе КОМПАС-3D. Функционал системы сегодня способен помогать разработчику изделий моделировать детали самых сложных форм и конфигурации.

# Глава 14
# ОСНОВЫ РАБОТЫ В CAM-СИСТЕМЕ ESPRIT

## 14.1. Общие сведения

В этой главе вы познакомитесь с одной из самых популярных и мощных CAD/CAM-систем в мире – ESPRIT, а точнее с самой последней ее версией 2012. Программное обеспечение ESPRIT – разработка известной американской компании DP Technology. Функциональность ESPRIT включает программирование фрезерной обработки от 2 до 5 осей, токарной обработки от 2 до 22 осей, электроэрозионной обработки от 2 до 5 осей, многозадачных токарно-фрезерных станков с синхронизацией, станков с осью B. Помимо возможности генерации УП для станков с ЧПУ, ESPRIT 2012 обладает и средствами построения 2D/3D-каркасной геометрии, создания поверхностных и твердотельных моделей.

ESPRIT имеет модульную структуру, позволяющую оптимально подобрать функции системы в зависимости от производственных потребностей предприятия. Конечно же в рамках одной главы невозможно рассказать обо всех возможностях системы, поэтому данное описание ESPRIT пригодится тем, кто хочет быстро понять и освоить только главные «заповеди» системы и получить быстрый результат, а также тем, кто впервые приступает к работе с CAD/CAM.

***Таблица 14.1. Краткая характеристика основных модулей ESPRIT***

| Название модуля (программного продукта) | Назначение и краткая характеристика |
|---|---|
| ESPRIT Commercial License | Базовая лицензия включает следующие компоненты ESPRIT: 2D/3D-каркасное построение, твердотельное моделирование Parasolid, NURBS-поверхностное моделирование, адаптивное распознавание конструктивных элементов детали, база знаний инструментов//режимов резания, универсальный постпроцессор, твердотельная верификация и симуляция станка, интерфейс для создания приложений на Visual Basic и ESPRIT API; трансляторы CAD-форматов ACIS, DXF, IGES, Inventor (*.ipt, *.iam), Parasolid (X_T, X_B), Solid Edge (*.par, *.psm, *.asm), SolidWorks (*.sldprt, *.sldasm), VDA, КОМПАС-3D (*.m3d, *.a3d); ассоциативная связь с Inventor, Solid Edge и SolidWorks |
| SolidMill Traditional | $2^1/_2$-осевое фрезерование |
| SolidMill Production | $2^1/_2$-осевое фрезерование, 4- и 5-осевое индексирование, обработка на 4-ой оси вращения, включает SolidMill Traditional |
| SolidTurn Traditional | 2-осевое точение |

***Таблица 14.1. Краткая характеристика основных модулей ESPRIT (окончание)***

| Название модуля (программного продукта) | Назначение и краткая характеристика |
|---|---|
| SolidTurn Production | 2-осевое точение, синхронизация обработки нескольких шпинделей и головок, включает SolidTurn Traditional |
| SolidMillTurn Traditional | Индексация оси C и поворотное фрезерование |
| SolidMillTurn Advanced | Индексация оси C и поворотное фрезерование, индексация фрезерной оси Y, включает SolidMillTurn Traditional |
| SolidMillTurn Production | Индексация оси C и поворотное фрезерование, индексация фрезерной оси Y, индексация фрезерной оси B, включает SolidMillTurn Traditional и Advanced |
| FreeForm 3-axis | Одновременная 3-осевая многоповерхностная/твердотельная 3D-фрезерная обработка как дополнение к SolidMill или SolidMillTurn |
| FreeForm 5-axis | Одновременная 3-осевая многоповерхностная/твердотельная 3D-фрезерная обработка, одновременная 5-осевая многоповерхностная/твердотельная 3D-фрезерная обработка как дополнение к SolidMill или SolidMillTurn |
| SolidWire Gold | 2-осевая и наклонная электроэрозионная обработка |
| SolidWire Platinum | 2-осевая и наклонная электроэрозионная обработка, 4- и 5-осевая электроэрозионная обработка, включает SolidWire Gold |
| KnowledgeBase | База знаний и технологий |
| Cutdata | База режимов резания |
| Report Generator | Генератор отчетов и карт наладки |

# 14.2. Системные требования

Для эффективного функционирования ESPRIT необходим компьютер, удовлетворяющий следующим системным требованиям:

- 1.0 GHz процессор;
- ОС Windows 7 или Window Vista;
- 4 Gb RAM, 3 Гб свободного места на диске;
- 256 MB OpenGL-видеокарта;
- 1024 г 768 разрешение монитора;
- Internet Explorer 6 или новее.

# 14.3. Активация лицензии и запуск программы

Активация лицензии осуществляется при помощи утилиты Security Manager, которую можно вызвать из меню программ Windows: **Все программы** ⇒ **DP Technology** ⇒ **Security**. Для того чтобы запустить программу, щелкните по ярлыку **ESPRIT** на рабочем столе или воспользуйтесь соответствующим пунктом меню Windows. Первое, что вы увидите после успешного запуска, – главное окно ES-

PRIT. Система готова к работе с файлами, созданию геометрии и приему (импорту) моделей из других CAD'ов.

## 14.4. Интерфейс программы

Главное окно ESPRIT условно можно разделить на 7 функциональных областей:

- графическое окно – здесь происходит основная работа и взаимодействие с 2D-геометрией и 3D-моделями;
- менеджер проекта – содержит структурированные данные о технологических операциях, режущих инструментах и элементах обработки;
- главное меню – представляет собой текстовое выпадающее меню на русском языке в верхней части главного окна;
- инструментальные панели – набор кнопок и пиктограмм для быстрого доступа к различным функциям системы;
- окно свойств – умеет отображать свойства геометрических и логических элементов;

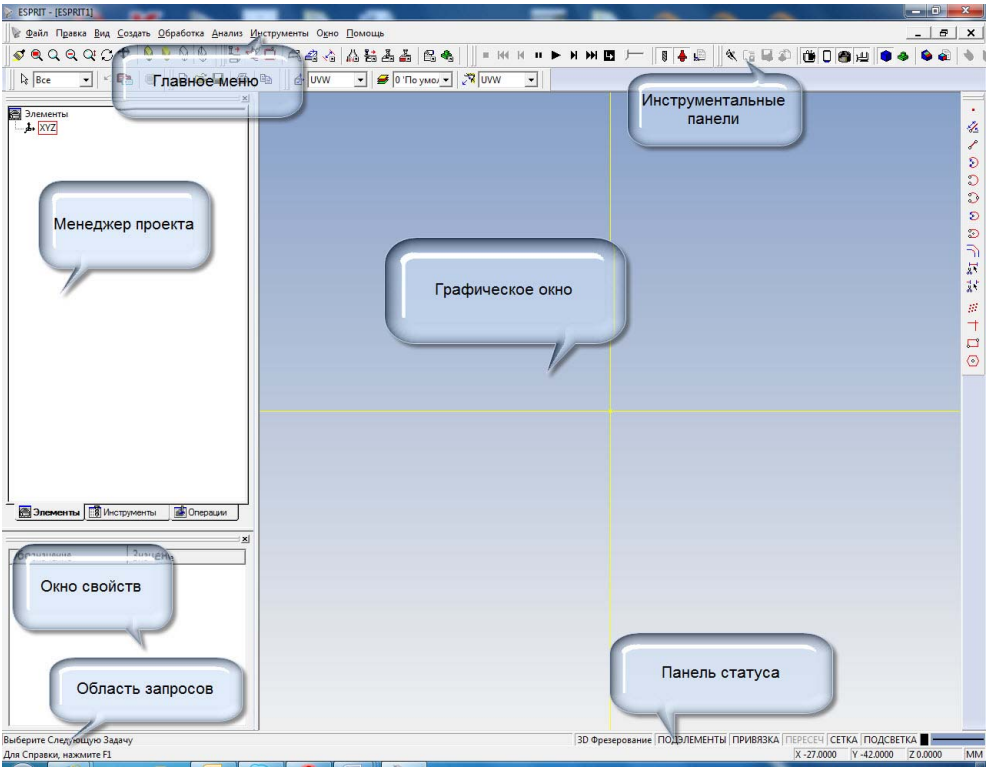



*Рис. 14.1. Интерфейс ESPRIT 2012*

- область запросов – выводит подсказки о требуемых от пользователя действиях;
- панель статуса – настройки, определяющие привязку и способ выделения геометрических элементов в графической области.

Прежде чем начать работу в ESPRIT, убедитесь, что активна метрическая, а не дюймовая система координат. Для этого в главном меню выберите пункт **Инструменты**, далее **Единицы системы**, **ММ**. В этом же пункте главного меню **Язык** отвечает за выбор языка (русский, английский) для общения с системой.

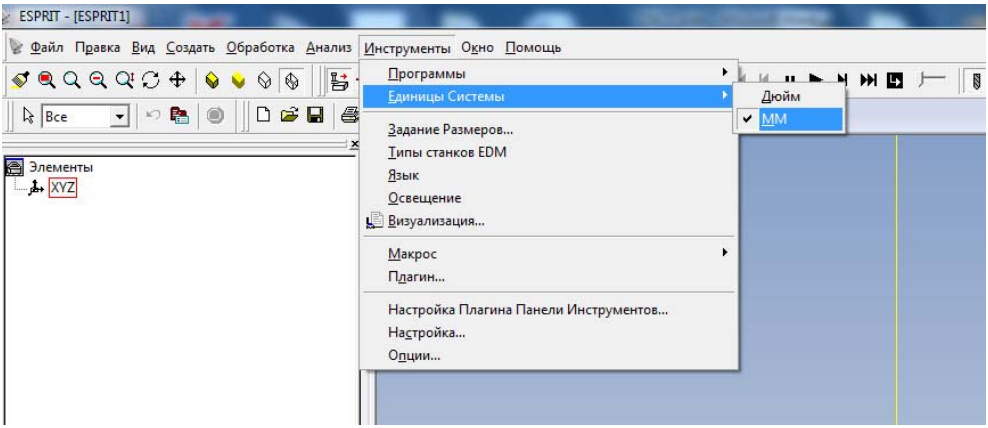



*Рис. 14.2. Выбор метрической системы координат*

Пользователю предоставляется возможность создавать собственные или модифицировать существующие кнопочные панели инструментов и наполнять их функциями по собственному усмотрению. Самый простой способ настроить ESPRIT «под себя» – обратиться к пункту **Опции** меню **Инструменты**.

***Таблица 14.2. Основные функции главного меню***

| Пункт главного меню | Основные функции и операции |
|---|---|
| Файл | Работа с файлами: открытие, сохранение, импорт/экспорт |
| Правка | Редактирование геометрии и траектории; операции копирования, вставки; откат назад |
| Вид | Управление графическим видом, осями, слоями, окнами |
| Создать | Операции по созданию 2D- и 3D-геометрических элементов |
| Обработка | Выбор операции (стратегии) обработки |
| Анализ | Анализ качества и нормалей поверхностных моделей |
| Инструменты | Глобальные настройки системы |
| Окно | Управление окнами графической области |
| Помощь | Вызов контекстной справки, обновление системы |

Меню правой кнопки мыши обеспечивает управление видами и отвечает за трансформацию геометрических элементов: копирование, вращение, проецирование и т. д.

## 14.5. Порядок работы в программе

Работа в программе строится по четкому алгоритму. На первом этапе пользователь создает или открывает (импортирует) геометрические элементы. Благодаря широкой поддержке различных форматов вы можете открыть родной файл практически любой CAD-системы. Особое преимущество ESPRIT заключается в способности работать напрямую с файлами программы КОМПАС-3D, причем с поддержкой дерева построения.

Далее пользователь выбирает определенные геометрические элементы, подлежащие обработке, и на их основе создает логические элементы различного типа: цепочки, поверхностные элементы. Эти элементы отображаются в соответствующей закладке **Менеджера проекта**.

На третьем этапе на элементы назначаются операции (стратегии) обработки, пользователь определяет настройки, технологические параметры и режущий инструмент. Система производит расчет траектории обработки по заданным параметрам и выводит готовые операции в соответствующую закладку **Менеджера проекта**.

Следующий шаг заключается в проверке созданных траекторий. Для этого пользователь включает визуализацию или симуляцию процесса обработки. Финальный этап работы в ESPRIT – постпроцессирование. Выбрав постпроцессор на требуемый станок, пользователь получает готовую управляющую программу, которая передается на станок для выполнения.

## 14.6. Создание операций фрезерной обработки

На DVD-приложении к книге содержатся демонстрационные файлы ESPRIT. Скопируйте все файлы из папки **Espritsamples** на жесткий диск ПК. Откройте файл Sample1.esp, который мы используем для создания операций фрезерной обработки. Убедитесь, что включен режим **3D-фрезерование**.

Если **Менеджер проекта** и **Окно свойств** не отображаются в левой части экрана, выберите пункт **Вид** главного меню и активируйте соответствующие пункты. Учтите, что эти панели и окна могут быть перемещены в любое удобное для пользователя место главного окна ESPRIT. Для включения координатных осей используйте пункты **Ось XYZ** и **Ось UVW**. Большинство инструментальных панелей – контекстно-зависимые и могут появляться и скрываться в зависимости от текущего режима работы.

Для поворота, масштабирования, перемещения и закрашивания 3D-модели используйте инструментальные панели **Вид** и **Слои и плоскости**. Убедитесь, что настройки панели статуса установлены аналогично рис. 14.8.

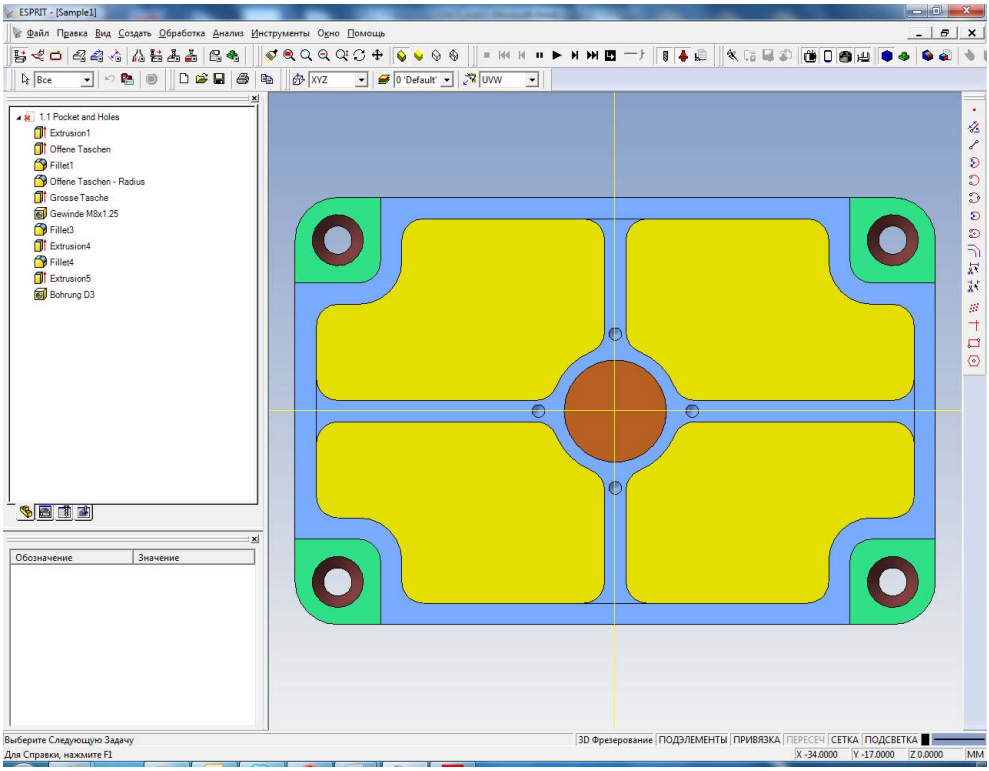



*Рис. 14.3. 3D-модель корпусной детали для фрезерной обработки*

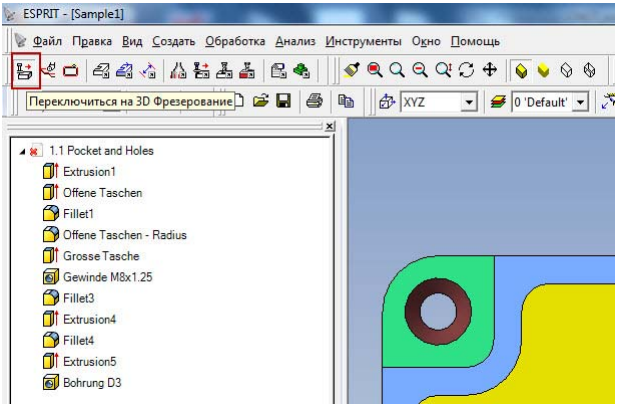



*Рис. 14.4. Режим фрезерования включен*

Выберите окном всю 3D-модель детали, перейдите в закладку **Элементы** менеджера проекта и нажмите кнопку **Создание элементов** ⇒ **Редактирование элементов**. В результате этих действий ESPRIT распознает конструктивные элемен-

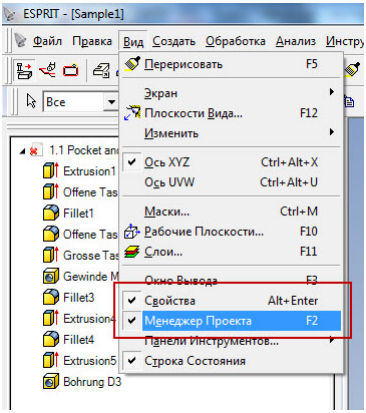



Рис. 14.5. Включение **Менеджера проекта** и **Окна свойств**

Рис. 14.6. Панель **Вид**

Рис. 14.7. Панель **Слои и плоскости**

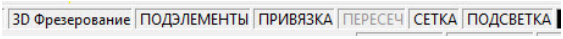



Рис. 14.8. Панель статуса в нижней части главного окна ESPRIT

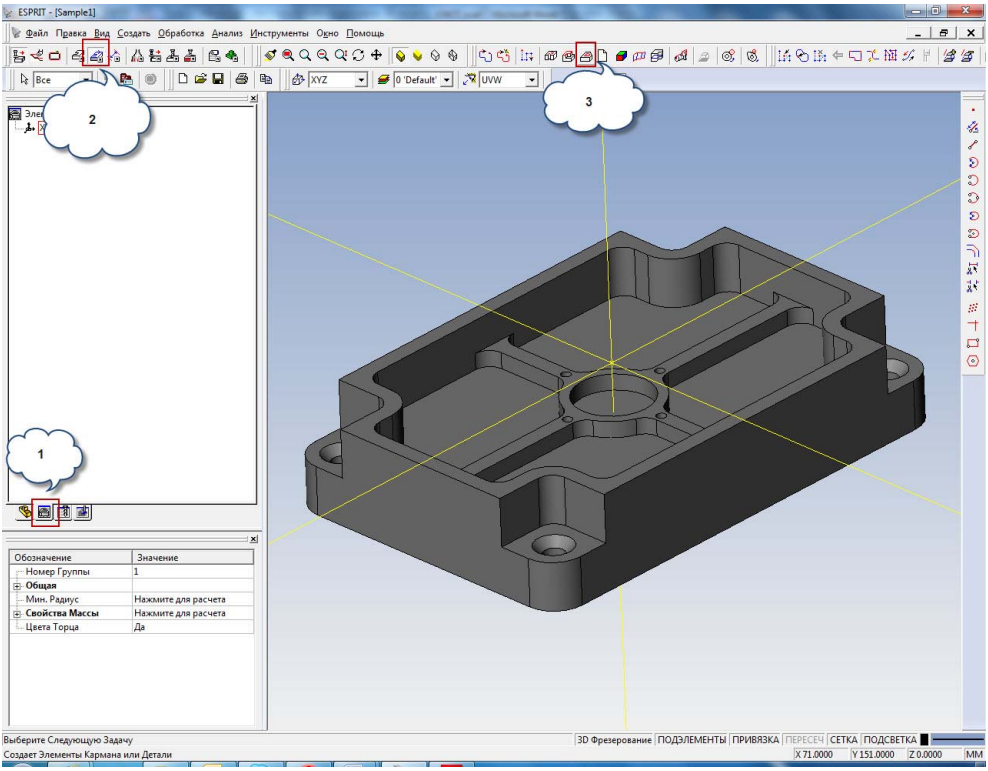



Рис. 14.9. Действия пользователя для распознавания конструктивных элементов типа карман в 3D-модели детали

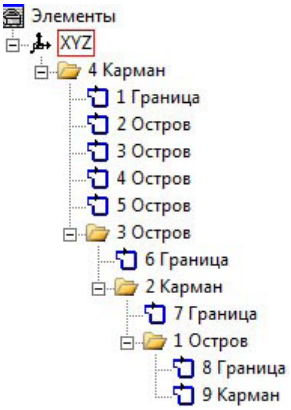



*Рис. 14.10. Дерево конструктивных элементов в закладке **Элементы***

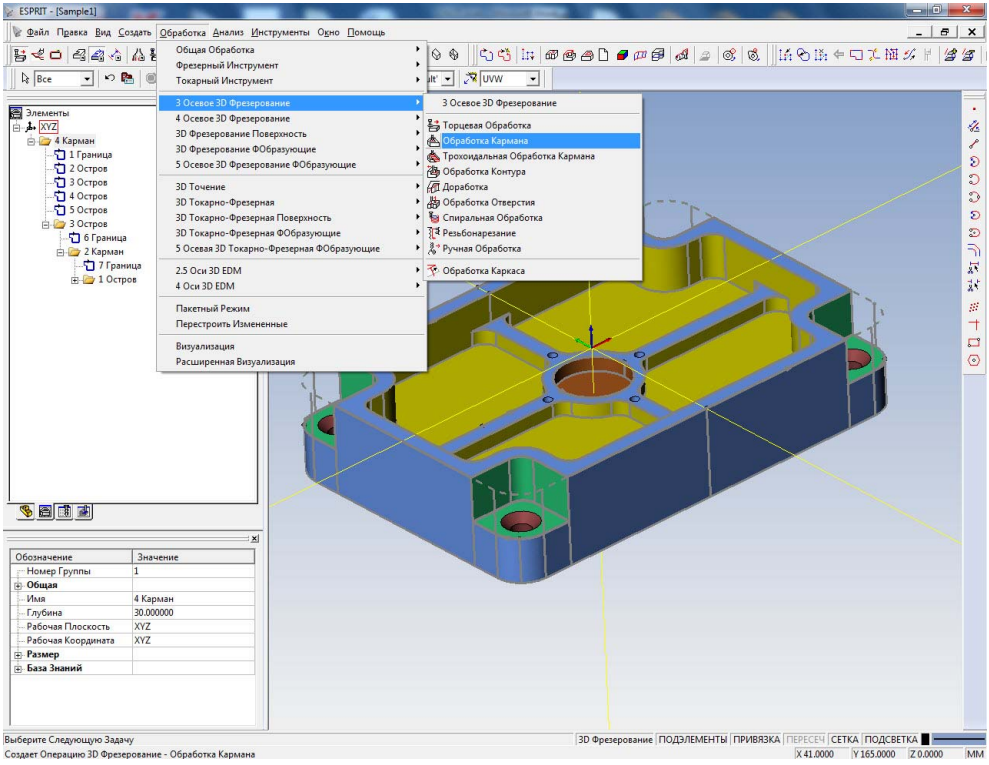



*Рис. 14.11. Выбор стратегии обработки*

ты выбранного типа в твердотельной модели и отобразит их списком (карманы, границы, острова).

Выделите элемент верхнего уровня **4 Pocket**, щелкнув по нему левой кнопкой мыши. В графической области возникнет каркас, обозначающий конструктивные элементы модели. Далее выберите: **Обработка** ⇒ **3-осевое 3D-фрезерование** ⇒ **Обработка кармана**. В закладке **Операции** менеджера операций появятся страницы параметров создаваемой операции.

Не закрывая окна, перейдите в закладку **Инструменты**. Щелкните в пустой области правой кнопкой мыши, далее **Новый** ⇒ **Фрезерный инструмент** ⇒ **Концевая фреза**. В нижней закладке установите диаметр режущей части фрезы равным 8 мм и нажмите **ОК**.

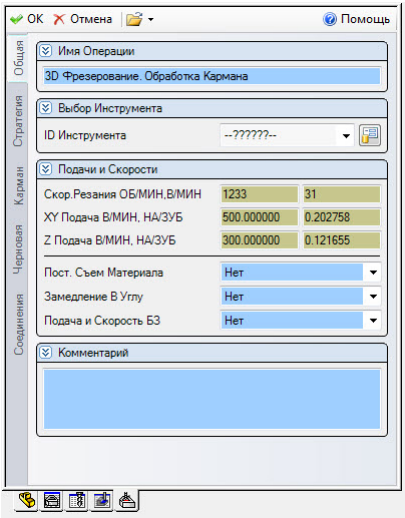



*Рис. 14.12. Выбор инструмента и режимов обработки*

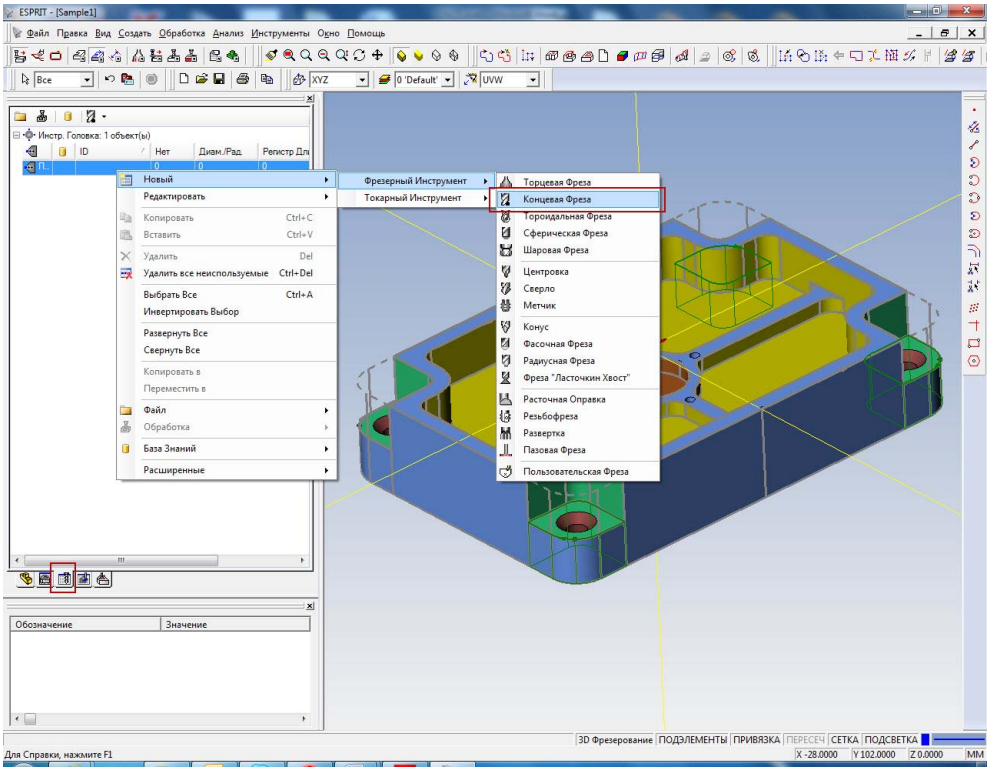



*Рис. 14.13. Создание нового инструмента*

Вернитесь в закладку **Операции** и выберите в поле **ID инструмента** название созданного инструмента из списка. Далее установите параметры обработки в оставшихся вкладках операции, как показано на рис. 14.15–14.18, и нажмите **OK** в верхней части окна.

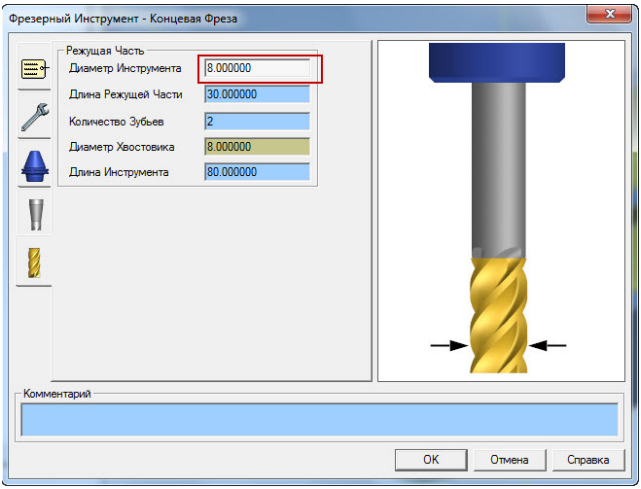



*Рис. 14.14. Определение параметров инструмента*

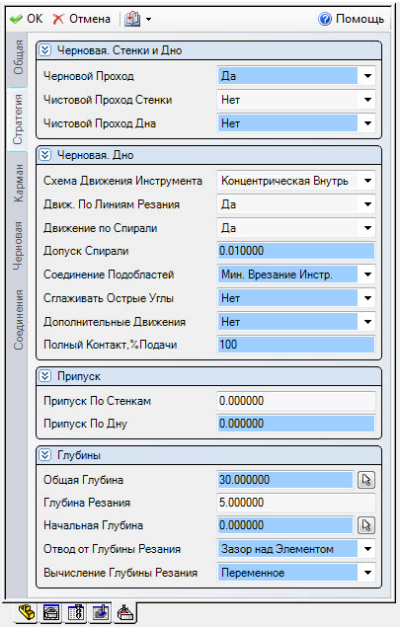



*Рис. 14.15. Параметры стратегии обработки*

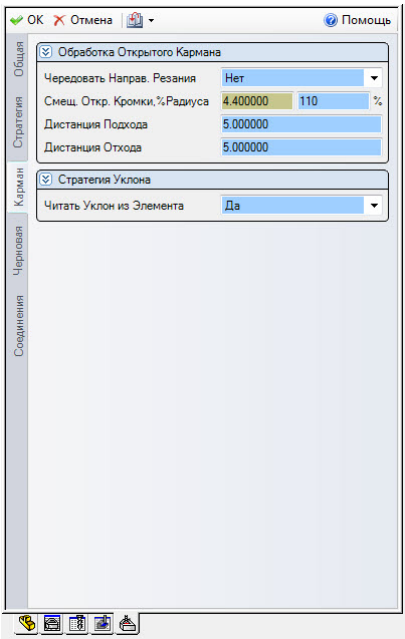



*Рис. 14.16. Параметры вкладки* ***Карман***

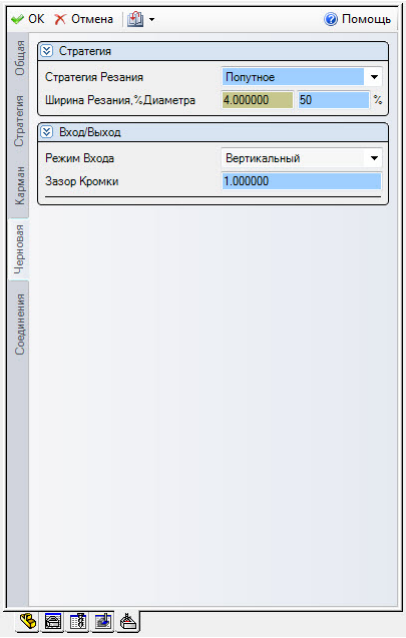



*Рис. 14.17. Параметры вкладки* ***Черновая***

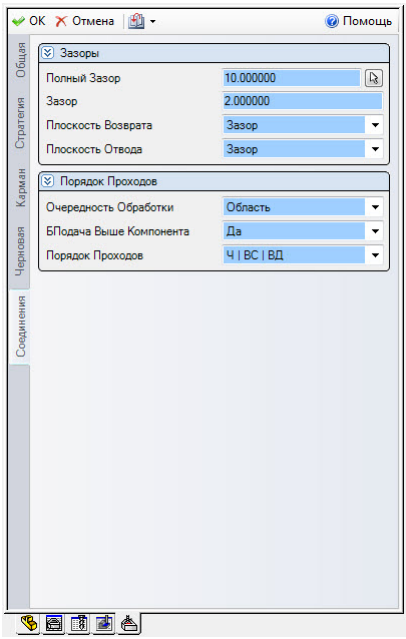



*Рис. 14.18. Определение плоскостей отвода и очередности обработки*

ESPRIT произведет расчет в строгом соответствии с параметрами операции, в графической области появится схематичное изображение новой траектории, а в закладках **Элементы** и **Операции** отобразится название операции – **3D-фрезерование. Обработка кармана**. Если вдруг вы захотите что-то исправить и изменить параметры операции – просто дважды кликните левой кнопкой мыши по ней.

Для технолога-программиста может представлять значительный интерес информация о времени выполнения созданной операции. Вы можете включить требуемый столбец, для этого кликните правой кнопкой мыши по заголовку закладки **Операции** и активируйте пункт **Выбор поля**.

В представленной 3D-модели детали вы можете заметить две группы отверстий. Давайте произведем их центровку.

Для распознавания отверстий нажмите уже известную вам кнопку **Создание элементов** ⇒ **Редактирование элементов**, далее – кнопка **Отверстия** в соответствующей инструментальной панели и **OK**.

В закладке **Элементы** менеджера операций выберите появившиеся элементы. Для совместного выбора используйте кнопку **Ctrl** или **Shift**. Далее выберите: **Обработка** ⇒ **3-осевое 3D-фрезерование** ⇒ **Обработка отверстия**. В закладке **Операции** менеджера операций появятся страницы параметров создаваемой операции.

*Рис. 14.19. Траектория перемещения инструмента*

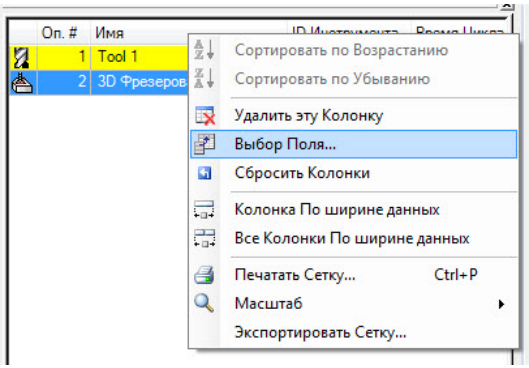
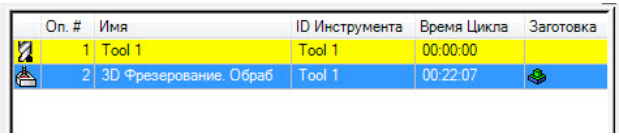



*Рис. 14.20. Настройка структуры информации об операции*



| Оп. # | Имя | ID Инструмента | Время Цикла | Заготовка |
|---|---|---|---|---|
| 1 | Tool 1 | Tool 1 | 00:00:00 | |
| 2 | 3D Фрезерование. Обраб | Tool 1 | 00:22:07 | |

*Рис. 14.21. Операция и время ее цикла*

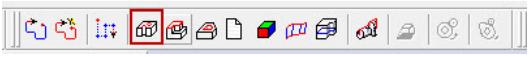


*Рис. 14.22. Кнопка распознавания отверстий*

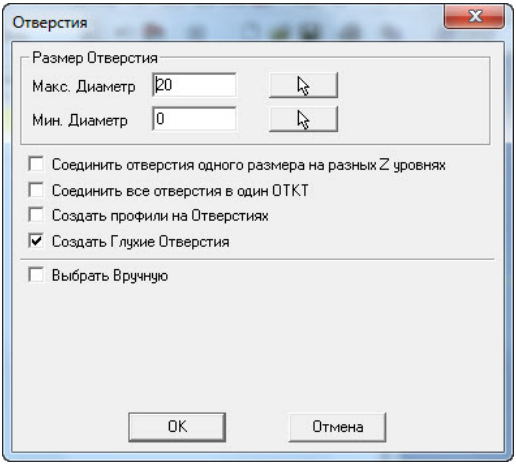



*Рис. 14.23. Диалог распознавания отверстий*

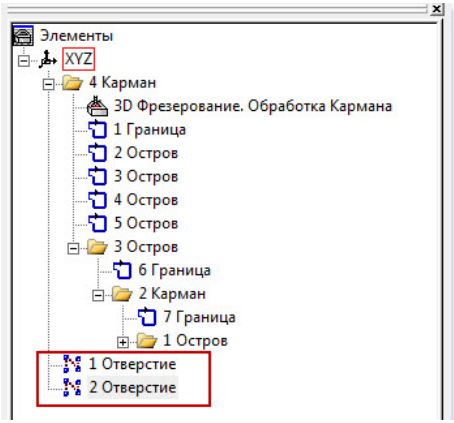



*Рис. 14.24. Новые элементы – две группы отверстий*

Последовательно выполните следующие действия:

- создайте новый инструмент – **Центровка**, диаметром 5 мм;
- далее установите параметры обработки в оставшихся вкладках операции, как показано на рис. 14.27 и 14.28, и нажмите **OK** в верхней части окна. Обратите внимание, что **Общая глубина** равна 1 мм.

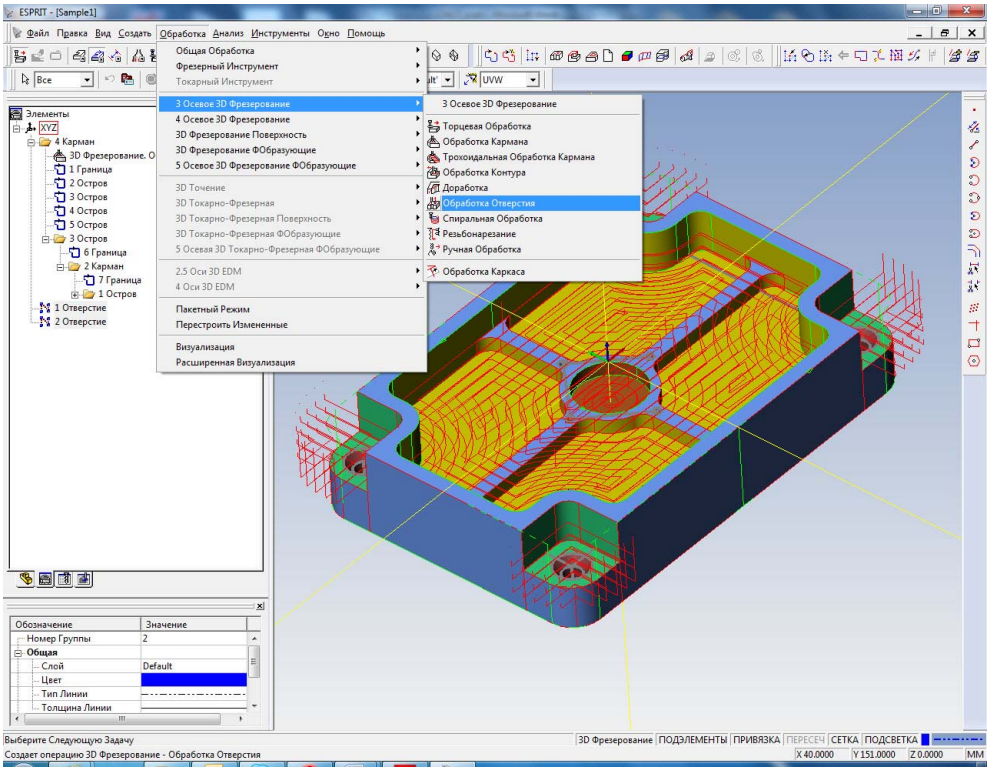



*Рис. 14.25. Выбор операции сверления*

ESPRIT позволяет произвести наглядную и точную проверку созданных траекторий. Для этого нажмите на кнопку **Визуализация** ⇒ **Расширенная визуализация**, и вы увидите панель с кнопками запуска и останова процесса симуляции обработки. Нажмите кнопку **Запустить** для запуска динамической симуляции. Вы можете вращать и масштабировать модель станочной среды для удобного восприятия и точной оценки результата.

По окончании процесса нажмите кнопку **Остановить**, и вы вернетесь в привычную графическую область ESPRIT.

Финальный этап нашей работы – вывод управляющей программы. Для этого выберите операции, нажмите правую кнопку мыши, далее – **УП**. В появившемся окне укажите файл постпроцессора HAASVF-4.asc из папки Espritsamples и дважды нажмите **ОК**. Готовая управляющая программа для станка с ЧПУ будет открыта в редакторе ESPRIT NC.

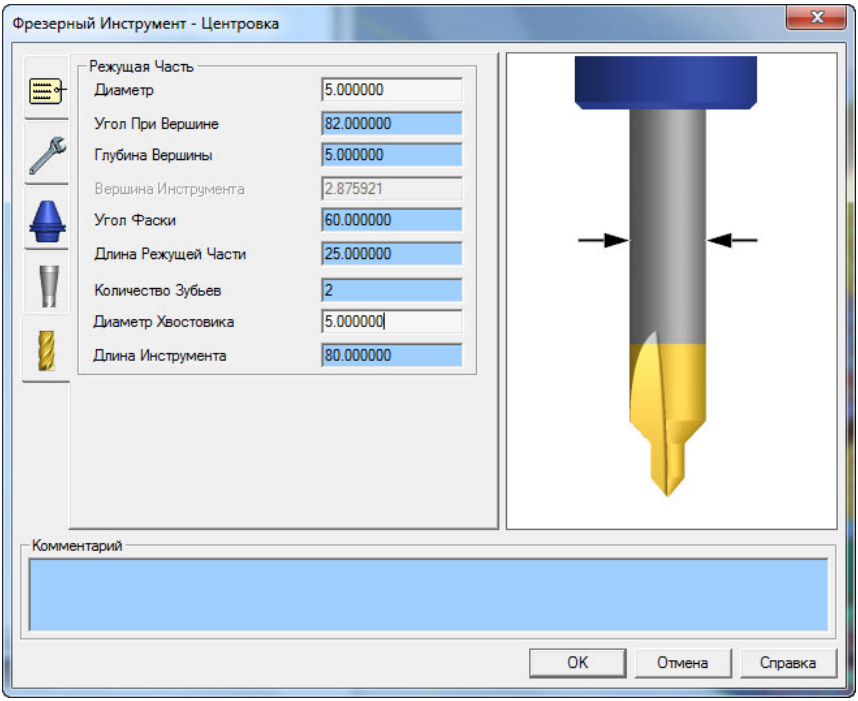



*Рис. 14.26. Создание центровки*

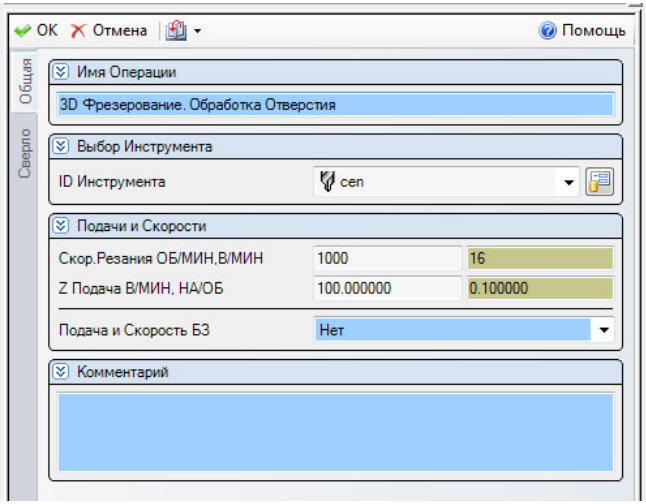



*Рис. 14.27. Выбор инструмента и режимов обработки*

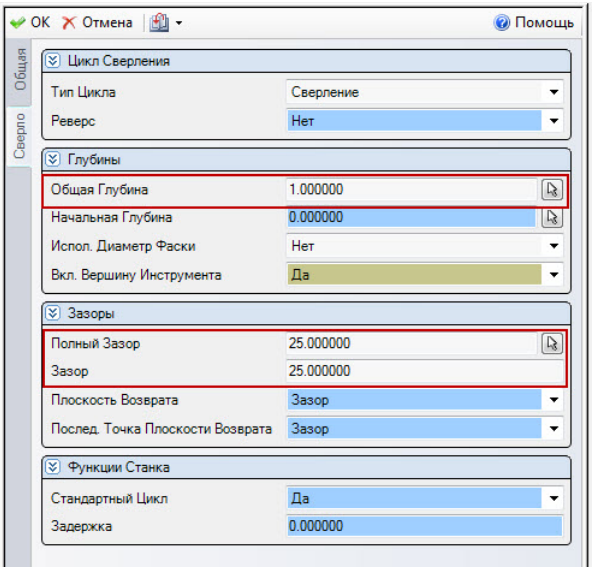



*Рис. 14.28. Выбор цикла и глубины сверления*

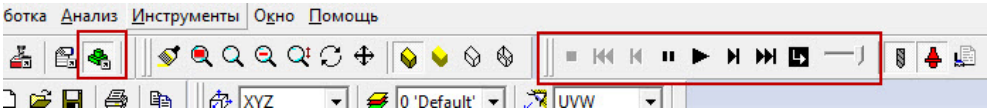



*Рис. 14.29. Запуск симуляции обработки*

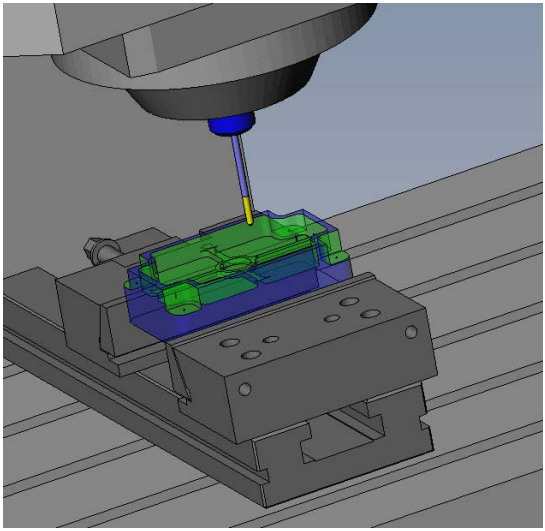


*Рис. 14.30. Динамическая симуляция траектории обработки*

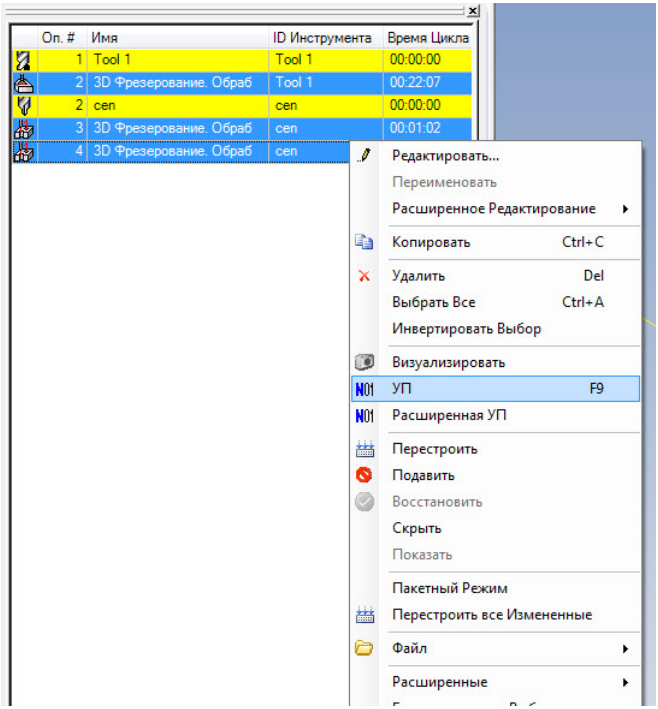



*Рис. 14.31. Выбор операций для постпроцессирования*

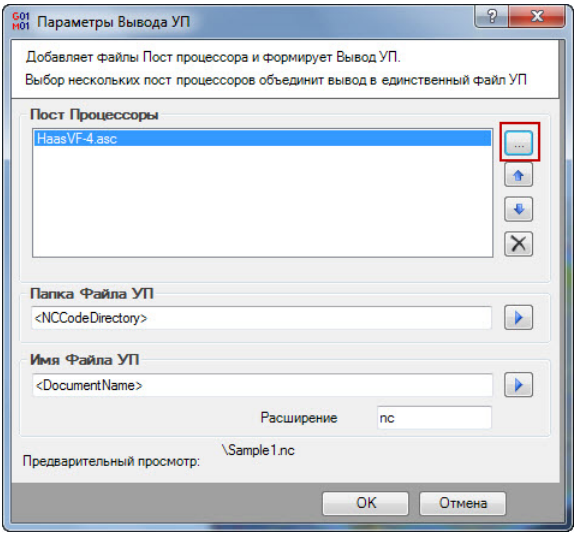



*Рис. 14.32. Диалог выбора постпроцессора*

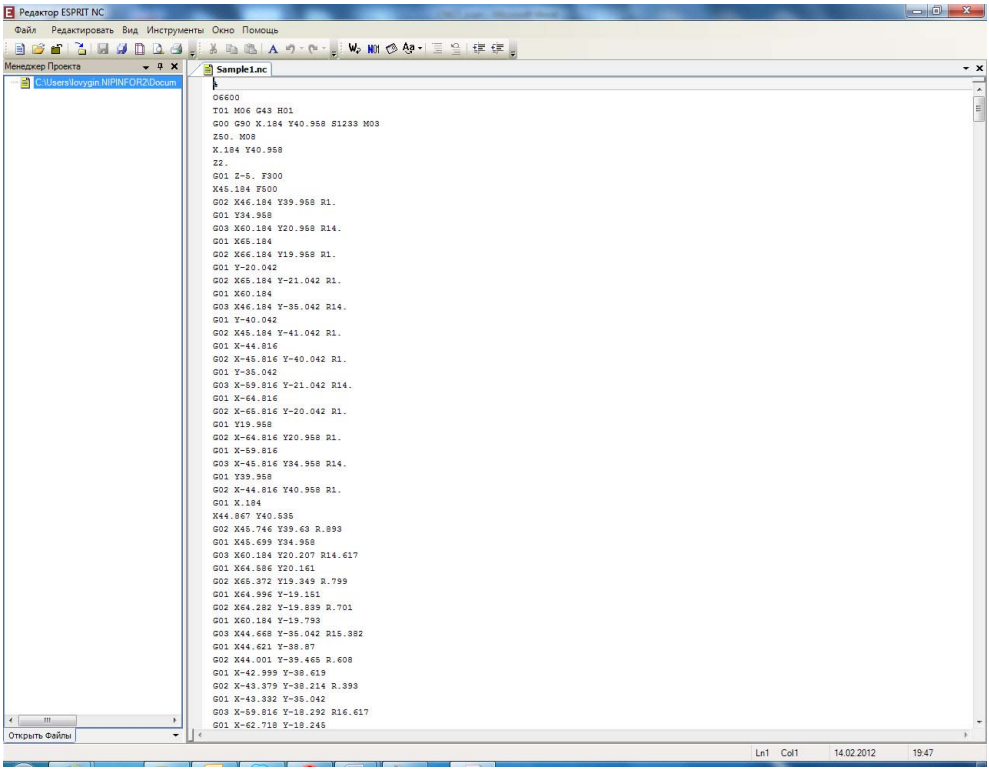



*Рис. 14.33. Готовая УП в фирменном редакторе ESPRIT NC*

# Глава 15
# УПРАВЛЕНИЕ СТАНКОМ С ЧПУ

## 15.1. Органы управления

Большинство органов управления современного станка с ЧПУ сосредоточены на передней панели стойки ЧПУ. К органам управления относятся различные переключатели и клавиши, а также дисплей, позволяющий оператору «общаться» со станком. Как правило, системы ЧПУ имеют монохромный или цветной электронно-лучевой дисплей, хотя самые современные станки могут быть оснащены жидкокристаллическим дисплеем. Любая стойка ЧПУ имеет клавиатуру: либо полноразмерную, аналогичную клавиатуре обычного персонального компьютера, либо ограниченную, которая позволяет вводить только основные символы и знаки программирования.

Все клавиши, переключатели и рукоятки станка можно условно разделить на несколько функциональных групп:

- **Клавиши для ввода различных символов, букв и цифр.**
  При помощи клавиатуры УЧПУ оператор станка может составить программу обработки прямо на экране, вводя G-коды, различные слова данных и специальные символы программирования (например, знак конца кадра). В случае ограниченной клавиатуры одна клавиша может отвечать за несколько символов (адресов).
- **Клавиши редактирования и курсора.**
  Клавиши редактирования позволяют оператору станка изменять содержимое управляющей программы. Курсорные клавиши предназначены для навигации по программе.
- **Программные или экранные клавиши.**
  Программные клавиши используются для выполнения различных функций в зависимости от программного обеспечения системы ЧПУ и текущего экранного режима. Обычно эти клавиши расположены прямо под дисплеем, а их текущие функции отображаются в нижней части дисплея.
- **Клавиши и переключатели режимов работы станка.**
  Станок с ЧПУ имеет несколько режимов работы. Для перехода из одного режима в другой обычно используется специальный переключатель.
- **Кнопки прямого управления осевыми перемещениями.**
  При помощи этих кнопок оператор может перемещать исполнительные органы станка в осевых направлениях на рабочей подаче или на ускоренной подаче.

- **Рукоятки управления скоростью подачи и вращения шпинделя.**
  Многие станки имеют средства для прямого (без программирования G- и M-кодов) включения/выключения шпинделя и управления скоростью его вращения. Система ЧПУ предоставляет оператору станка возможность корректировки запрограммированной скорости подачи и частоты вращения шпинделя в определенных диапазонах.
- **Клавиши и переключатели для работы со специальными функциями станка.**
  За включение и выключение освещения рабочей зоны станка, управление системой удаления стружки и другие вспомогательные действия отвечают клавиши и переключатели для работы со специальными функциями.
- **Клавиши цикла программирования.**
  За пуск управляющей программы отвечает кнопка **Старт цикла**, а за ее останов – кнопки **Останов подачи** или **Сброс**. К этой же группе относятся клавиши для активации функций выборочной остановки **M01**, пропуска кадра /, покадрового выполнения программы, пробного прогона и блокировки осевых перемещений.
- **Другие органы управления.**
  Часть органов управления может быть расположена не на самом пульте УЧПУ. Например, выключатель электропитания часто расположен на тыльной стороне корпуса станка, а клавиши управления инструментальным магазином – прямо рядом с окошком для загрузки инструментов. Большая красная кнопка **Экстренный останов** находится на самом видном и доступном месте.

Практически все станки с ЧПУ имеют маховики, которые дают возможность оператору перемещать исполнительные органы вручную. Как правило, этими маховиками оператор пользуется для выполнения точных операций, таких как поиск нулевой точки или измерение длины инструмента.

Кроме различных органов управления, станок с ЧПУ имеет набор индикаторов. Эти индикаторы (светодиоды или лампочки) могут показывать, пришли или не пришли исполнительные органы станка в нулевую точку, включена ли подача СОЖ, и сигнализируют о возникшей аварийной ситуации.

*Рис. 15.1. Кнопка для экстренной остановки станка*

# 15.2. Основные режимы работы

**Режим автоматического управления.** Этот режим является основным для станка с ЧПУ. Именно в этом режиме производится обработка детали по программе. Для запуска УП на выполнение необходимо сначала выбрать активную программу и затем нажать кнопку **Старт цикла**.

В режиме автоматического управления оператор может влиять на запрограммированную скорость подачи и частоту вращения шпинделя. Рукоятка коррекции ускоренного хода позволяет изменять скорость холостых перемещений исполнительных органов станка обычно в диапазоне от 0 до 150%.

**Режим редактирования.** В этом режиме оператор станка может вводить новую или редактировать существующую программу обработки вручную, используя клавиатуру УЧПУ.

Возможности по редактированию УП у разных стоек ЧПУ могут значительно отличаться. Простейшие системы позволяют вставлять, удалять и копировать слова данных. Самые современные СЧПУ имеют функции поиска и замены данных (аналогично текстовым редакторам на ПК), копирования, удаления и переноса определенного программного диапазона, способны редактировать УП в фоновом режиме.

**Функция фонового редактирования данных** позволяет оператору станка создавать или редактировать одну программу при одновременном выполнении другой программы. Для фонового редактирования систему управления необходимо переключить в автоматический режим.

Обычно в режиме редактирования осуществляется ввод/вывод УП с персонального компьютера или другого внешнего устройства. Здесь же можно проверить размер свободной памяти СЧПУ и количество зарегистрированных программ.

**Режим ручного ввода данных MDI.** Режим ручного ввода данных MDI позволяет оператору ввести и выполнить один или несколько кадров, не записанных в памяти СЧПУ. Обычно этот режим используется для ввода отдельных G- и M-кодов, например для смены инструмента или включения оборотов шпинделя. Введенные команды и слова данных после выполнения или сброса удаляются.

**Толчковый режим.** Толчковый (старт-стопный) режим обеспечивает ручное перемещение исполнительных органов станка при нажатии на соответствующие клавиши на панели УЧПУ.

**Режим управления ручным генератором импульсов или маховиками.** В этом режиме осуществляется перемещение исполнительных органов станка при помощи ручного генератора импульсов, который похож на пульт дистан-

*Рис. 15.2. Ручной генератор импульсов*

ционного управления или при помощи специальных маховиков на панели УЧПУ. Оператор станка может задавать шаг и направление перемещения при помощи специальных переключателей.

**Режим возврата в нулевую точку.** Возврат исполнительных органов в нулевую точку является стандартной процедурой при включении станка. В этом случае происходит синхронизация станка и системы управления.

**Режим прямого числового управления DNC.** Режим DNC позволяет выполнять программу обработки прямо из компьютера или другого внешнего устройства, не записывая ее в память системы. Обычно в этом режиме выполняются УП большого размера, которые не могут поместиться в памяти СЧПУ.

**Режим редактирования параметров.** В этом режиме производят редактирование параметров системы ЧПУ. Пользовательские параметры отвечают за настройку текущей даты и времени, работу в различных режимах и т. д. Системные параметры влияют на функционирование станка в целом. Не рекомендуется самостоятельно изменять значения системных параметров. Иногда вход в область параметров заблокирован и для редактирования требуется ввести специальный код, установленный производителем станка.

**Тестовые режимы.** У любого станка с ЧПУ есть определенное количество тестовых функций. К ним, например, относятся пробный прогон и покадровая отработка УП. Некоторые системы ЧПУ позволяют осуществлять графическую проверку траектории.

## 15.3. Индикация системы координат

Во время выполнения обработки по программе или ручного перемещения исполнительных органов станка есть возможность наблюдать за их текущим положением в различных координатных системах:

| | |
|---|---|
| ABSOLUTE | Абсолютное положение в рабочей системе координат (G54–G59) |
| MACHINE | Текущее положение относительно нуля станка |
| DISTANCE TO GO | Оставшееся расстояние перемещения в кадре |

По координатам регистров ABSOLUTE и MACHINE можно судить о правильности перемещения по программе. Координаты регистра MACHINE обычно используются оператором станка для нахождения нуля детали и установления рабочей системы координат.

## 15.4. Установление рабочей системы координат

Существуют несколько методов «привязки» координатной системы к детали. При **классическом методе** оператор станка использует плоскопараллельные концевые меры, калибры или индикатор (центроискатель). **Метод контакта** основан

на касании режущим инструментом обрабатываемой детали (заготовки). **Автоматический метод** подразумевает использование специального щупа и инфракрасных датчиков, которые устанавливаются в качестве опции на современные обрабатывающие центры.

Классический метод является универсальным и может использоваться на любом станке с ЧПУ, поэтому именно его мы рассмотрим наиболее подробно. Так как «привязка» осуществляется в ручном режиме, то ее точность невысока – порядка 0.02 мм. Метод достаточно прост и заключается в касании шпинделем или калибром плоскопараллельной концевой меры, прижатой к поверхности детали. После несложных вычислений текущие машинные координаты вручную заносятся в регистры рабочих смещений (G54–G59). Нахождение нулевой точки состоит их двух этапов: первый – по оси Z, второй – по осям X и Y.

## 15.4.1. Алгоритм нахождения нулевой точки детали по оси Z

1. Подготовить и держать поблизости плоскопараллельную концевую меру толщиной не более 25 мм.
2. Подвести торец шпинделя в толчковом режиме к поверхности детали по оси Z на расстояние не более 50 мм.
3. При помощи маховика или ручного генератора импульсов подвести торец шпинделя еще ближе к детали, так чтобы это расстояние стало меньше толщины плоскопараллельной концевой меры.
4. Положить плоскопараллельную концевую меру на поверхность детали рядом со шпинделем.

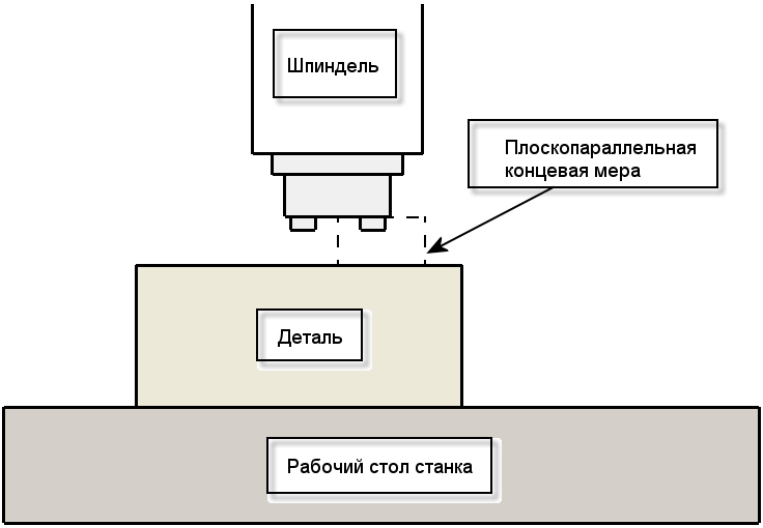



*Рис. 15.4. Установка рабочей системы координат по Z*

5. Постепенно перемещать шпиндель в положительном направлении по оси Z (вверх), непрерывно контролировать зазор между шпинделем и деталью.
6. Как только плоскопараллельная концевая мера войдет между шпинделем и деталью, остановить движение шпинделя. Шпиндель установлен правильно, если при смещении плоскопараллельной концевой меры чувствуется небольшое сопротивление.
7. Так как базовой позицией для шпинделя является точка пересечения его торца и оси вращения, то необходимо учесть толщину плоскопараллельной концевой меры.
   Пример:
   Машинная координата по Z = –400.
   Толщина плоскопараллельной концевой меры = 25 мм.
   В регистр рабочего смещения по Z заносим = –400 – 25 = –425 мм.
8. Ввести в регистр рабочего смещения по Z значение, рассчитанное в п. 7.

## 15.4.2. Алгоритм нахождения нулевой точки детали по осям X и Y

1. Вставить в шпиндель цилиндрический калибр с известным диаметром, например 20 мм.
2. В толчковом режиме подвести калибр к поверхности детали по оси X на расстояние не более 25 мм.
3. Приложить к поверхности детали по оси X плоскопараллельную концевую меру.

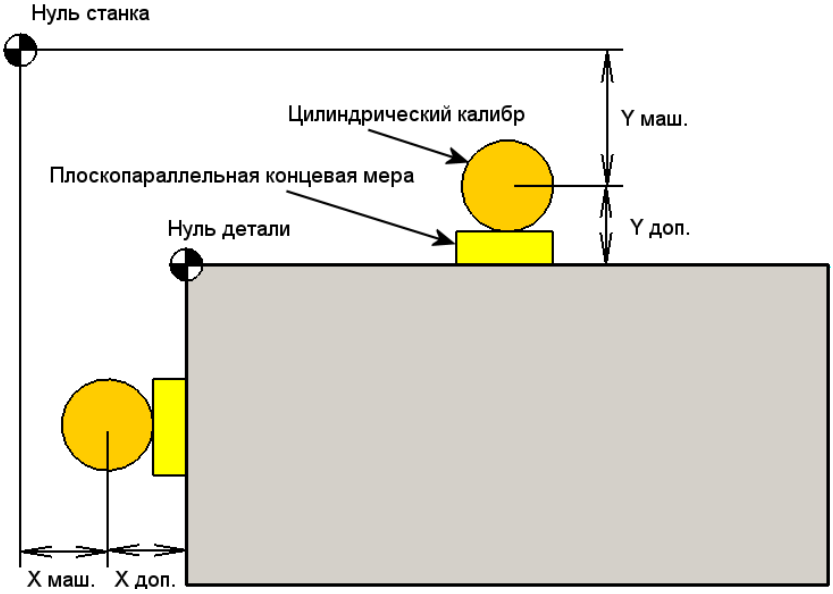



*Рис. 15.5. Установка рабочей системы координат по X и Y*

4. При помощи маховика постепенно перемещать шпиндель с калибром к детали вдоль оси X до касания с плоскопараллельной концевой мерой. Шпиндель установлен правильно, если при смещении плоскопараллельной концевой меры чувствуется небольшое сопротивление.
5. Отметить машинную позицию шпинделя, учитывая радиус цилиндрического калибра и толщину плоскопараллельной концевой меры, вычислить значение для ввода в регистр рабочего смещения по оси X.
6. Ввести в регистр рабочего смещения по X значение, рассчитанное в п. 7.
7. В толчковом режиме подвести калибр к поверхности детали по оси Y на расстояние не более 25 мм.
8. Приложить к поверхности детали по оси Y плоскопараллельную концевую меру.
9. При помощи маховика постепенно перемещать шпиндель с калибром к детали вдоль оси Y до касания с плоскопараллельной концевой мерой. Шпиндель установлен правильно, если при смещении плоскопараллельной концевой меры чувствуется небольшое сопротивление.
10. Отметить машинную позицию шпинделя, учитывая радиус цилиндрического калибра и толщину плоскопараллельной концевой меры, вычислить значение для ввода в регистр рабочего смещения по оси Y.
11. Ввести в регистр рабочего смещения по Y значение, рассчитанное в п. 10.

## 15.4.3. Алгоритм нахождения нулевой точки в центре отверстия

1. Установить в шпиндель стрелочный индикатор (центроискатель).
2. В толчковом режиме подвести индикатор как можно ближе к центру отверстия над деталью.

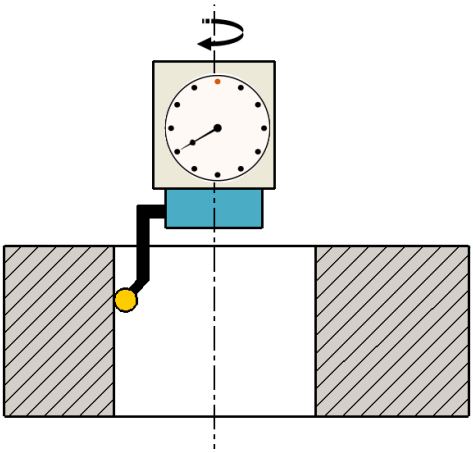


*Рис. 15.6. Установка рабочей системы координат в центре отверстия*

3. При помощи маховика осторожно вставить щуп индикатора в отверстие.
4. Прислонить щуп к стенке отверстия.
5. Используя вращательное движение, юстировать положение осей X и Y шпинделя до тех пор, пока показываемый индикатором дисбаланс не окажется в допустимых пределах.
6. Записать машинные координаты по X и Y в соответствующие регистры рабочих смещений.

# 15.5. Измерение инструмента и детали

Как вы уже знаете, для того чтобы режущий инструмент приходил в правильную позицию, необходимо выполнить компенсацию его длины, то есть произвести смещение базовой точки шпинделя на величину, записанную в регистре длины этого инструмента. В некоторых случаях нужно произвести коррекцию на радиус инструмента. Перед началом обработки оператор станка должен измерить длину и радиус каждого инструмента, использующегося в УП, и записать числовые значения в соответствующие регистры системы ЧПУ.

Существуют несколько методов для измерения длины и радиуса режущего инструмента: метод касания заготовки, измерение вне станка при помощи специального измерительного устройства и автоматическое измерение на станке.

В настоящее время многие обрабатывающие центры оснащаются специальным **датчиком касания**. Обычно он расположен в углу рабочего стола и не мешает обработке. Измерение геометрических характеристик инструмента производится автоматически по специальной программе, заложенной в СЧПУ. Требуемый инструмент устанавливается в шпиндель и по команде оператора автоматически

*Рис. 15.7. Датчик Renishaw TS-27 для измерения длины и радиуса инструмента. Преимущества автоматического измерения – высокая точность и скорость*

*Рис. 15.8. Устройство для измерения инструмента вне станка*

подводится к датчику. Сначала инструмент подводится к датчику касания сверху. Так как система знает координаты датчика и базовой точки шпинделя, то в момент касания она производит вычисление абсолютной длины инструмента. Затем инструмент касается датчика сбоку, и система определяет его радиус. **Перед началом цикла автоматического измерения оператор должен внести в регистры системы примерные значения длины и радиуса инструмента (можно замерить обычной линейкой) во избежание столкновения с датчиком.** После измерения полученные данные сразу попадают в соответствующие регистры СЧПУ.

Аналогичные системы существуют и в бесконтактном варианте, при этом измерение инструмента осуществляется при помощи лазера.

На некоторых предприятиях для измерения инструментов используют специальное электронно-механическое устройство. Инструмент устанавливают в гнездо, аналогичное конусу шпинделя станка, и подводят измерительный щуп к режущей кромке. На небольшом дисплее отображаются длина и радиус. Оператор станка должен самостоятельно занести полученные значения в соответствующие регистры системы.

*Рис. 15.9. Измерительный щуп касается поверхности детали*

Для автоматического установления рабочей системы координат и измерения размеров деталей на станки с ЧПУ устанавливается специальная система, состоящая из измерительного щупа, который крепится в шпинделе аналогично инструменту и инфракрасным датчикам, «висящим» на корпусе внутри рабочей зоны. Оператор станка может подвести щуп к детали, по определенной команде он коснется поверхности и определит ее координаты. Можно, например, автоматически измерить диаметр обработанного отверстия и найти его центр. Инфракрасные датчики работают как приемник – они получают сигнал от щупа в момент его касания с поверхностью детали и передают его в систему управления.

# Глава 16
# СПРАВОЧНИК КОДОВ И СПЕЦИАЛЬНЫХ СИМВОЛОВ ПРОГРАММИРОВАНИЯ

## 16.1. G-коды

**G00 – ускоренное перемещение.** Код G00 используется для выполнения ускоренного перемещения. Ускоренное перемещение, или позиционирование, необходимо для быстрого перемещения режущего инструмента к позиции обработки или к безопасной позиции. Ускоренное перемещение никогда не используется для выполнения обработки, так как скорость движения исполнительного органа станка очень высока и непостоянна. Код G00 отменяется при программировании следующих кодов: G01, G02, G03.

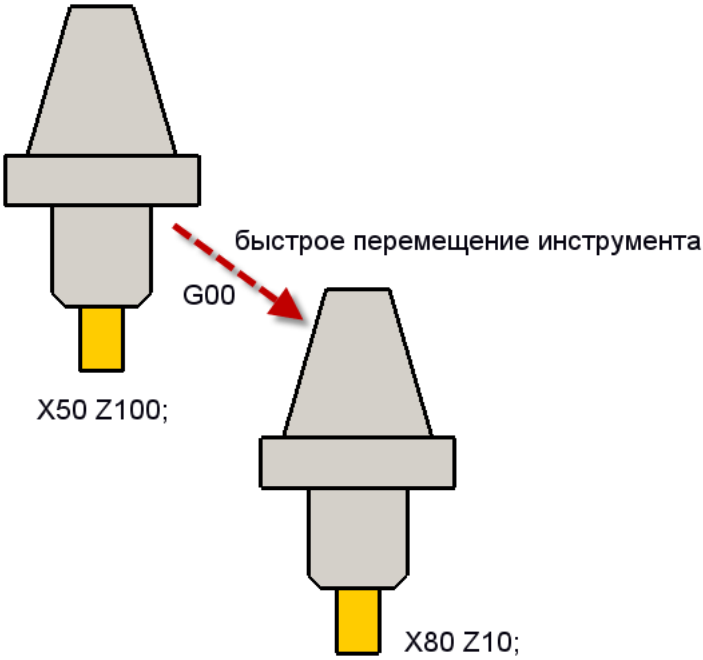



*Рис. 16.1. Пример ускоренного перемещения.*
*G00 X80 Z10 – позиционирование в точку с координатами (80; 10).*
*Более подробную информацию вы найдете в главе 6*

**G01 – линейная интерполяция.** Код G01 – команда линейной интерполяции, обеспечивающая перемещение инструмента по прямой линии с заданной скоростью. Скорость перемещения указывается F-адресом. Код G01 отменяется с помощью кодов G00, G02нG03.

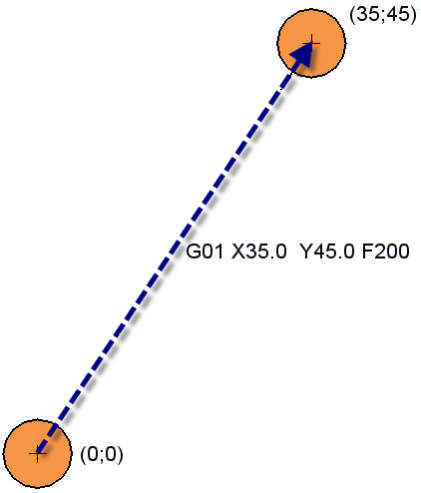



*Рис. 16.2. Пример линейной интерполяции. G01 X35 Y45 F200 – перемещение по прямой в точку с координатами (35; 45) со скоростью подачи 200 мм/мин. Более подробную информацию вы найдете в главе 6*

**G02 – круговая интерполяция** (дуга **в направлении часовой стрелки**). Код G02 предназначен для выполнения круговой интерполяции, то есть для перемещения инструмента по дуге (окружности) в направлении часовой стрелки с заданной скоростью. Скорость перемещения указывается F-адресом. Код G02 отменяется с помощью кодов G00, G01 и G03.

**G03 – круговая интерполяция** (дуга **против часовой стрелки**). Код G03 предназначен для выполнения круговой интерполяции, то есть для перемещения ин-

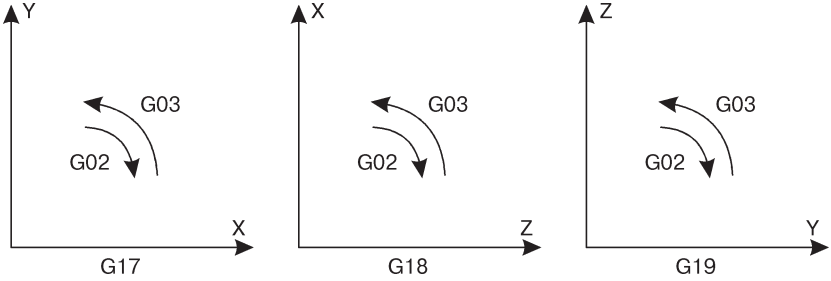



*Рис. 16.3. Круговая интерполяция в разных плоскостях*

струмента по дуге (окружности) против часовой стрелки с заданной скоростью. Скорость перемещения указывается F-адресом. Код G03 отменяется с помощью кодов G00, G01 и G02.

**G04 – выдержка.** Код G04 – команда на выполнение выдержки (паузы) с заданным временем. Этот немодальный код программируется вместе с X- или P-адресом, который указывает длительность времени выдержки. Обычно это время составляет от 0.001 до 99999.999 секунды. Код G04, X- или P-адрес программируются вместе в одном кадре, который не содержит никаких перемещений.

Если для определения времени выдержки используется P, то нельзя программировать десятичную точку. Адрес P определяет время выдержки в миллисекундах, а X – в секундах. Если команда G04 программируется без временного фактора, то она воспринимается системой ЧПУ как немодальная команда для точного останова.

Пример:

G04 X1.5 – выдержка 1.5 секунды;

G04 P2000 – выдержка 2 секунды.

**G09 – точный останов.** Из-за автоматического ускорения и замедления осевых перемещений исполнительных органов станка с ЧПУ не происходит точная обработка кромок углов при переходе от одного движения резания к другому. Эта неточность обработки выражается в закруглении или притуплении углов.

Предположим, вы обрабатываете прямоугольный контур и стремитесь получить острую кромку в углу (рис. 16.4). Если вы работаете в обычном режиме, то возможно, что при переходе от движения по оси Y к движению по оси X произойдет небольшое скругление этой самой кромки (рис. 16.5). Особенно сильно данный эффект проявляется при работе с высокими скоростями подач и на больших обрабатывающих центрах.

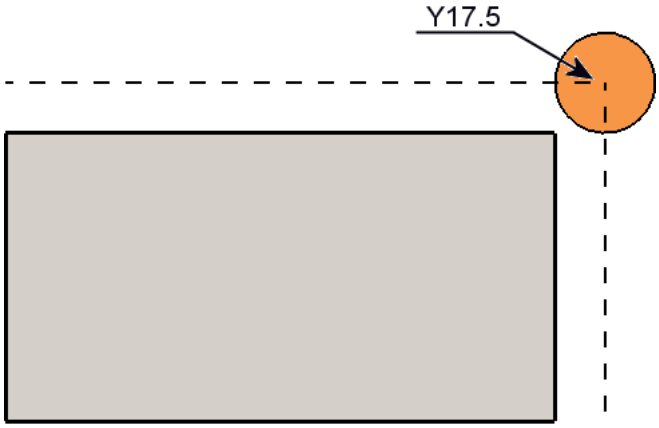



*Рис. 16.4. Необходимо получить острую кромку в правом верхнем углу контура*

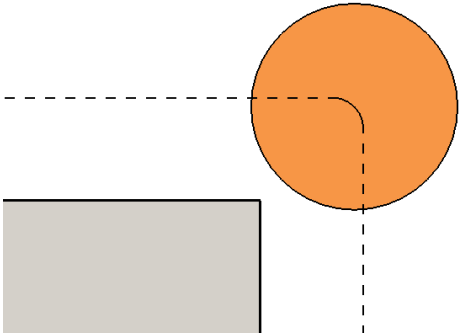


*Рис. 16.5. Скругление кромки*

Немодальный код G09 предназначен для согласования фактической траектории инструмента с запрограммированной траекторией. То есть при переходе от одного движения к другому СЧПУ обеспечит законченное и точное перемещение в указанную координату.

Код G09 обычно указывается вместе с координатой, в которой необходимо выполнить точный останов. Управляющая программа, гарантирующая получение острой кромки правого верхнего угла прямоугольного контура, будет выглядеть следующим образом:

```
%
O0005
N100 G21
N102 G0 G17 G40 G49 G80 G90
N104 T1 M6
N106 G0 G90 G54 X30. Y-22.5 S1000 M3
N108 G43 H1 Z100.
N110 Z10.
N112 G1 Z-2. F100.
N114 Y-12.5
N116 G09 Y17.5
N118 X-25.
N120 X-35.
N122 Z8.
N124 G0 Z100.
N126 M5
N132 M30
%
```

Когда инструмент приходит в координату Y17.5, то СЧПУ выполняет точный останов. Время выдержки в этой координате определяется значением специального параметра системы.

Чуть позже вы познакомитесь с кодом G61, который работает аналогично G09, но является модальным.

**G10 – включение режима ввода данных в СЧПУ.** Команда G10 позволяет устанавливать или смещать рабочую систему координат и вводить определенные значения в регистры коррекции инструмента памяти СЧПУ при помощи управляющей программы или специальной (отдельной) программы.

Если вы хотите ввести какие-либо значения в регистры коррекции при помощи УП, то они должны находиться в начале программы. Этим вы обеспечиваете согласованность значений в регистрах коррекции и самой программы обработки.

Обычно для ввода значений в регистры коррекции применяется следующий формат:

G10 L11 P_R_;

где G10 – включение режима ввода данных; L11 – настройка регистра коррекции инструмента; P – выбор регистра коррекции, который необходимо изменить; R – вводимое значение коррекции.

Если команда G10 используется одновременно с кодом G90, то значения в регистрах коррекции переписываются (заменяются новыми значениями). Когда G10 работает вместе с кодом G91, то значения в корректорах складываются (или вычитаются) с числовым значением при R. К примеру, кадр G10 G90 Lll P12 RIOO.05 заменяет существующее значение в регистре коррекции № 12 на новое значение 100.05.

Для того чтобы установить или сместить рабочую систему координат, используется следующий формат:

G10 L2 P_X_Y_Z_;

где G10 – включение режима ввода данных; L2 – определение стандартной рабочей системы координат; P – выбор рабочей системы координат; X, Y, Z – значения, определяющие новое положение рабочей системы координат.

Подготовительная функция G10 является модальной и остается активной до тех пор, пока не будет отменена кодом G11. Перед использованием G10 внимательно ознакомьтесь с документацией к станку, так как формат кадра с G10 может быть различным.

**G11 – выключение режима ввода данных в СЧПУ.** При помощи команды G11 отменяется команда G10 для включения режима ввода данных в СЧПУ.

**G15 – выключение режима полярных координат.** При помощи команды G15 вы отменяете режим работы в полярной системе координат и возвращаетесь к программированию в прямоугольной системе координат.

**G16 – включение режима полярных координат.** Подготовительная функция G16 позволяет работать в полярной системе координат. При этом запрограммированная позиция определяется углом и расстоянием от нулевой точки рабочей системы координат или от текущей действительной позиции.

Работать в полярной системе координат можно в одной из трех плоскостей. С кодом G17 вы работаете в плоскости XY, с кодом G18 – в плоскости XZ, а с помощью кода G19 – в плоскости YZ.

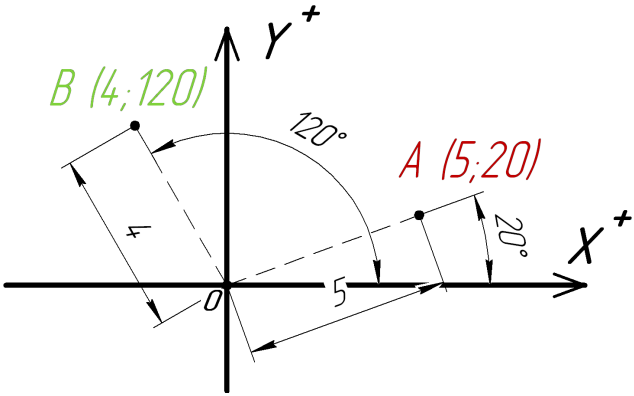



*Рис. 16.6. Полярные координаты: точка A (5;20) и точка B (4; 120)*

Если активна плоскость XY, то X-адрес определяет радиус, а Y устанавливает угол относительно оси X. Если активна плоскость XZ, то X адрес определяет радиус, а Z устанавливает угол относительно оси X. Если активна плоскость YZ, то Y-адрес определяет радиус, а Z устанавливает угол относительно оси Y. Положительным считается угол, который отсчитывается против часовой стрелки.

Полярные перемещения, которые указываются при действующей команде G90, выполняются относительно нулевой точки активной рабочей системы координат. Если же действует код G91, то полярные перемещения выполняются относительно текущей позиции. Значения угла и радиуса могут быть запрограммированы независимо как абсолютные или относительные значения. То есть полярное перемещение может быть одновременно определено углом от нулевой точки рабочей системы координат и расстоянием (радиусом) от текущей позиции.

Нередки случаи, когда на чертежах отверстия указываются при помощи полярных координат. Чтобы не пересчитывать полярные координаты в прямоугольные, можно воспользоваться подготовительной функцией G16.

```
...
G90 G17 G16
G81 G98 X4 Y30 Z-2 R0.5 F50
Y60
Y90
G15 G80
...
```

Команда G16 является модальной, поэтому остается активной до тех пор, пока ее не отменят командой G15.

**G17 – выбор плоскости XY.** Подготовительная функция G17 предназначена для выбора плоскости XY в качестве рабочей (рис. 16.7). Плоскость XY становится определяющей при использовании круговой интерполяции, вращения системы координат и постоянных циклов сверления.

**G18 – выбор плоскости XZ.** Подготовительная функция G18 предназначена для выбора плоскости XZ в качестве рабочей (рис. 16.7). Плоскость XZ становится определяющей при использовании круговой интерполяции, вращении системы координат и постоянных циклов сверления.

**G19 – выбор плоскости YZ.** Подготовительная функция G19 предназначена для выбора плоскости YZ в качестве рабочей (рис. 16.7). Плоскость YZ становится определяющей при использовании круговой интерполяции, вращения системы координат и постоянных циклов сверления.

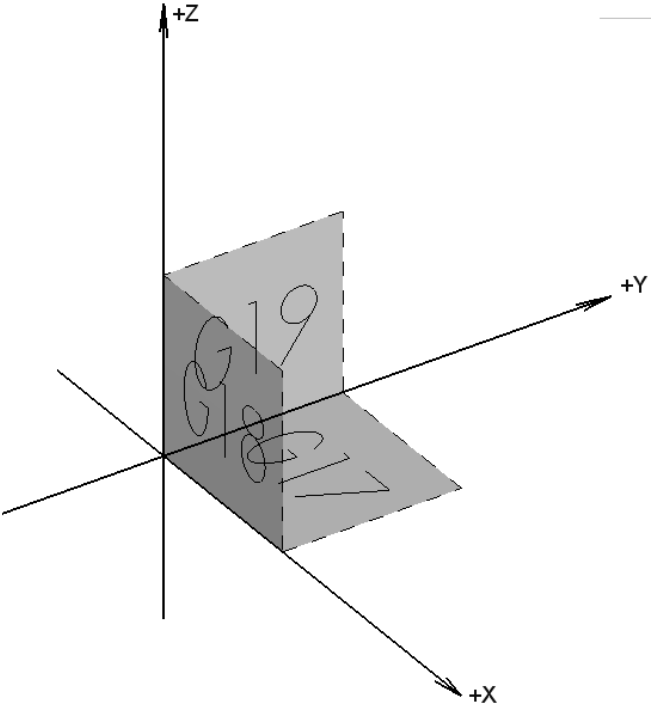



*Рис. 16.7. G17, G18, G19 используются для выбора активной плоскости системы координат*

**G20 – ввод дюймовых данных.** Код G20 активирует режим работы с дюймовыми данными. Пока действует этот режим, все вводимые данные воспринимаются как дюймовые. Рекомендуется во всех программах, которые написаны в дюймовых размерах, поставить команду G20 в начало программы (в строку безопасности), чтобы в случае, если в программе, выполняемой до этого, действовал метрический режим, обеспечить выбор корректного формата.

Пример:

N10 **G20** G40 G49 G54 G80 G90 – код G20 в строке безопасности.

Команда является модальной и действует до тех пор, пока ее не отменят командой G21.

**G21 – ввод метрических данных.** Код G21 активирует режим работы с метрическими данными. Пока действует этот режим, все вводимые данные воспринимаются как метрические. Рекомендуется во всех программах, которые написаны в метрических размерах, поставить команду G21 в начало программы (в строку безопасности), чтобы в случае, если в программе, выполняемой до этого, действовал дюймовый режим, обеспечить выбор корректного формата.

Пример:

N10 **G21** G40 G49 G54 G80 G90 – код G21 в строке безопасности.

Команда является модальной и действует до тех пор, пока ее не отменят командой G20.

**G22 – включение режима предельных перемещений.** Код G22 активирует установленный предел перемещений. В этом случае инструмент не может выйти за пределы ограничивающей области. Эта область, как правило, устанавливается с помощью параметров СЧПУ.

**G23 – выключение режима предельных перемещений.** При выполнении команды G23 установленные пределы перемещений не действуют. То есть код G23 отменяет действие кода G22 и позволяет инструменту перемещаться в любую точку рабочей зоны станка.

**G27 – проверка возврата к исходной позиции.** Код G27 работает аналогично коду G28 (см. далее). Единственная разница заключается в том, что если позиция, к которой произошло перемещение исполнительного органа, не соответствует исходной позиции, то в случае с G27 система ЧПУ станка выдает аварийное сообщение или сигнал.

Команды G27 и G28 могут использоваться в циклах и макросах автоматической смены инструмента. Перед выполнением этих G-кодов обычно отменяют коррекцию инструмента.

**G28 – автоматический возврат в исходную позицию.** Команда G28 предназначена для возврата станка в исходную позицию. Под этим понимается ускоренное перемещение исполнительных органов в нулевую точку станка. Возврат в исходную позицию предназначен прежде всего для возможности проверки размеров и качества обрабатываемой детали в середине программы обработки. Иногда код G28 ставят в конец управляющей программы, чтобы после ее завершения рабочий стол переместился в положение, удобное для съема обработанной детали.

Условный кадр для автоматического возврата в исходную позицию выглядит так:

G91G28X0.0Y0.0Z0.0

Если в кадре с G28 указываются оси X, Y и Z с нулевыми значениями, то возврат в исходную позицию происходит по этим трем осям. Однако не всегда нужно выполнять эту операцию со всеми осями. Возможно, что вам потребуется перемещение только по двум из них. Например, для возврата по осям Z и Y в программе обработки должен стоять следующий кадр:

G91 G28 Y0.0. Z0.0

Обратите особое внимание на находящийся в кадре код G91. Как вы знаете, этот код активирует работу в относительных координатах. Дело в том, что G28 позволяет запрограммировать некоторую промежуточную точку, в которую будет совершено перемещение, перед тем как станок вернется в исходную позицию. На самом деле координаты, указанные в кадре, являются координатами именно промежуточной точки. В приведенных ранее примерах мы указывали в качестве координат промежуточной точки нулевые значения. Так как в кадре стоит код относительных координат G91, то станок должен переместиться относительно текущей позиции на нуль миллиметров по каждой из осей. То есть не должен никуда двигаться. Вот поэтому при наличии в УП кадра G91 G28 X0.0 Y0.0 Z0.0 станок будет сразу возвращен в исходную позицию без «заезда» в промежуточную точку.

Если в программе обработки находится кадр G91 G28 X10.0 Z20.0, то станок сначала переместится вправо и вверх, а только затем вернется в нулевую точку. Для чего нужна эта промежуточная точка? Код G28 вызывает ускоренное перемещение, аналогичное G00, а в этом случае оно может быть непрямолинейным. То есть можно запросто что-нибудь «зацепить». Опытный программист старается сначала поднять инструмент вверх, а уже затем «отпустить» станок в нулевую точку:

G91 G28 X0.0 Y0.0 Z20.0

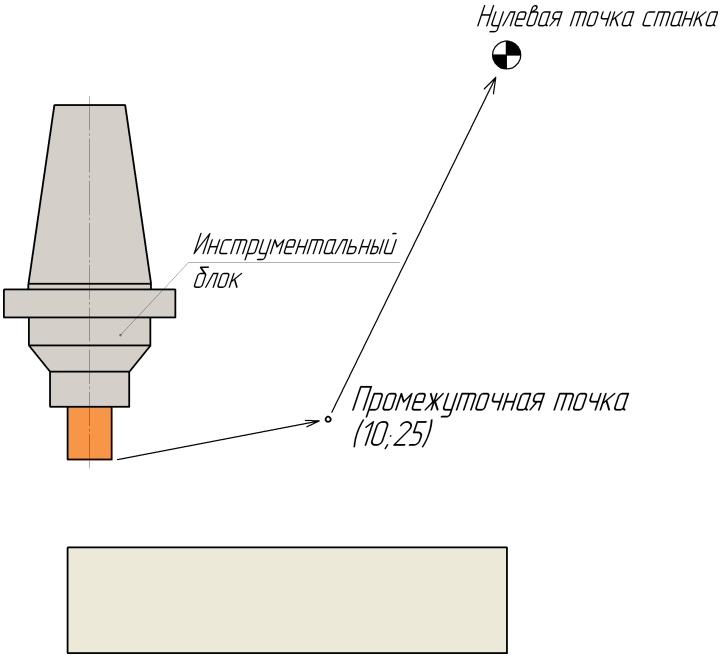



*Рис. 16.8. Если в УП присутствует команда G91 G28 X10 Y25, то инструмент сначала переместится в промежуточную точку, а затем вернется в нуль станка*

Опасайтесь указывать в кадре с G28 код абсолютных координат G90. Если в УП находится кадр G90 G28 X0.0 Y0.0 Z0.0, то очень высока вероятность столкновения режущего инструмента с частями станка или деталью.

**G30 – возврат к позиции смены инструмента.** При помощи команды G30 осуществляется автоматический возврат оси Z к позиции смены инструмента и отменяется действующая коррекция инструмента. Кадр для выполнения возврата к позиции смены инструмента должен выглядеть так:

G30 G91 Z0

Будьте внимательны: если в кадре вместо G90 находится код G91, то шпиндель будет перемещаться к поверхности рабочего стола.

**G31 – функция пропуска с реакцией на внешний сигнал.** В некоторых станках можно использовать функцию пропуска с реакцией на внешний сигнал. При помощи немодального кода G31 программист программирует линейную интерполяцию аналогично G01, но скомбинированную с возможной реакцией на внешний сигнал. Внешний сигнал подается при нажатии на определенную клавишу панели УЧПУ, например на клавишу **Старт цикла**.

Если сигнал пропуска не подавать, то программа будет выполняться таким образом, как если бы была запрограммирована команда G01. Если СЧПУ получит внешний сигнал, то выполнение программы переходит сразу же к следующему кадру данных.

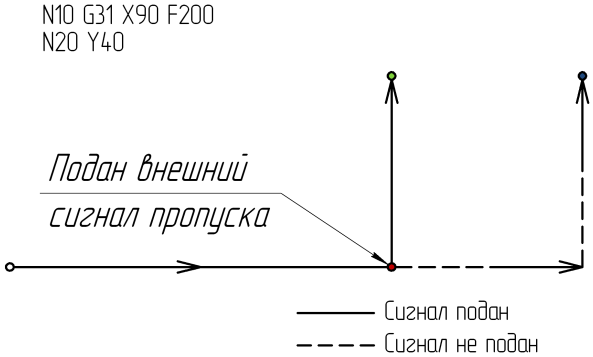



*Рис. 16.9. Функция пропуска с реакцией на внешний сигнал*

**G40 – отмена автоматической коррекции радиуса инструмента.** Автоматическая коррекция радиуса инструмента отменяется программированием команд G40 и D00. Обычно код G40 находится в кадре с командой прямолинейного холостого перемещения от контура детали.

G1 G40 X100

Некоторые станки отменяют автоматическую коррекцию радиуса инструмента при нажатии на кнопку аварийного останова или сброса, в случае возврата к нулевой точке станка и с помощью кодов окончания программы.

**G41 – коррекция на радиус, инструмент слева от детали.** Код G41 применяется для включения автоматической коррекции радиуса инструмента, находящегося слева от детали. Направление смещения определяется, если смотреть на траекторию сверху вниз, то есть со стороны «+Z» в направлении «–Z».

**G42 – коррекция на радиус, инструмент справа от детали.** Код G42 применяется для включения автоматической коррекции радиуса инструмента, находящегося справа от детали. Направление смещения определяется, если смотреть на траекторию сверху вниз, то есть со стороны «+Z» в направлении «–Z».

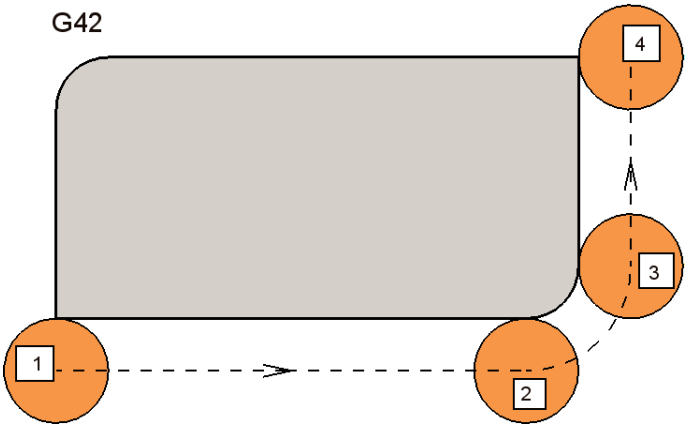



*Рис. 16.11. Коррекция справа*

**G43 – компенсация длины инструмента.** При **выполнении** УП базовая позиция шпинделя (точка пересечения торца и оси вращения) определяется запрограммированными координатами. Проблема заключается в том, что в базовой позиции шпинделя обработка резанием не осуществляется. Обработка производится кромкой режущего инструмента, которая находится на некотором расстоянии от базовой точки шпинделя. Для того чтобы в запрограммированную координату приходила именно режущая кромка, а не шпиндель, необходимо «объяснить» СЧПУ, на какую величину по оси Z нужно сместить эту базовую точку.

Компенсация длины инструмента осуществляется путем программирования команды G43 и H-слова данных. Обычно компенсация длины активируется совместно с холостым перемещением по оси Z.

Пример: G43 H01 Z100

**G49 – отмена компенсации длины инструмента.** Компенсация длины инструмента отменяется путем программирования команды G49 или H00.

**G50 – выключение режима масштабирования.** Код G50 предназначен для выключения режима масштабирования G51.

**G51 – включение режима масштабирования.** В этом режиме программист изменяет коэффициент масштаба для координатных осей станка. Режим активиру-

ется при помощи модального кода G51 и отменяется кодом G50.

Можно указать коэффициент масштаба для всех осей одновременно или отдельно для каждой оси. Если коэффициент масштаба более 1, то система координат увеличивается. Если же коэффициент масштаба менее 1, то система координат уменьшается.

Для единого изменения масштаба обычно используется следующий формат:

G51 X_Y_Z_P_

где G51 – включение режима масштабирования; X – координата по оси X для средней точки масштаба; Y – координата по оси Y для средней точки масштаба; Z – координата по оси Z для средней точки масштаба; P – коэффициент масштаба для всех осей.

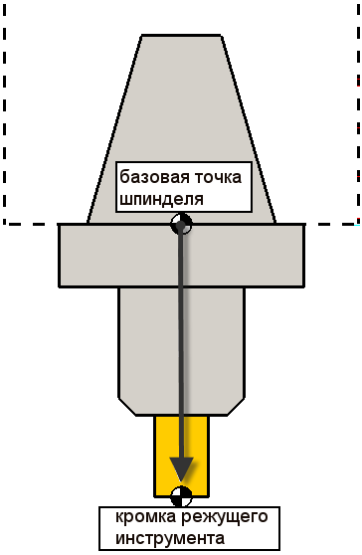



*Рис. 16.12. Команда G43H_ смещает базовую точку шпинделя к кромке режущего инструмента*

При независимом изменении масштаба возможно также зеркальное отображение с помощью отрицательных коэффициентов масштаба. Для независимого изменения масштаба обычно используется следующий формат:

G51 X Y Z I J K

где G51 – включение режима масштабирования; X – координата по оси X для средней точки масштаба; Y – координата по оси Y для средней точки масштаба; Z – координата по оси Z для средней точки масштаба; I – коэффициент масштаба для оси X; J – коэффициент масштаба для оси Y; K – коэффициент масштаба для оси Z.

В функции зеркального отображения комбинируются между собой независимое изменение масштаба и возможность зеркального отображения запрограммированных координат по одной или нескольким осям. В следующем программном примере поясняется функция зеркального отображения без изменения масштаба.

**Основная программа**

```
...
G90 G01 F100
M98 P101
G51 X5 Y5 I-1 J1 K1
M98 P101
G51 X5 Y5 I-1 J-1 K1
M98 P101
G51 X5 Y5 I1 J-1 K1
M98 P101
...
```

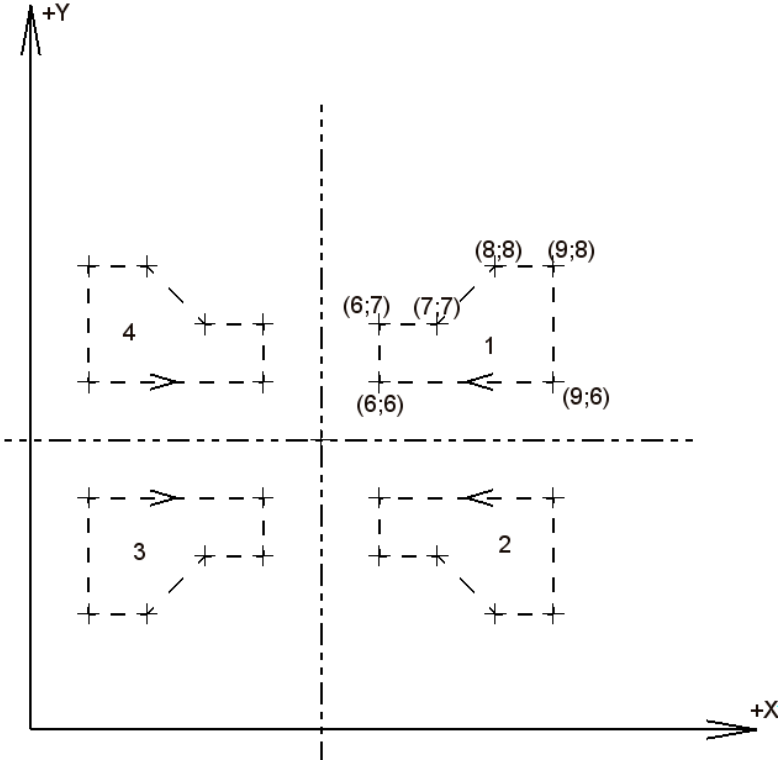



*Рис. 16.13. Зеркальное отображение траектории*

**Подпрограмма**

O0101
G90 X6 Y6
Y7
X7
X8 Y8
X9
Y6
X6
M99

**G52 – локальная система координат.** СЧПУ позволяет устанавливать, кроме стандартных рабочих систем координат, еще и локальные системы координат. Код G52 используется для определения подчиненной системы координат в пределах действующей рабочей системы (G54–G59).

Когда СЧПУ станка исполняет команду G52, то начало действующей рабочей системы координат смещается на значение, указанное при помощи слов данных X, Y и Z:

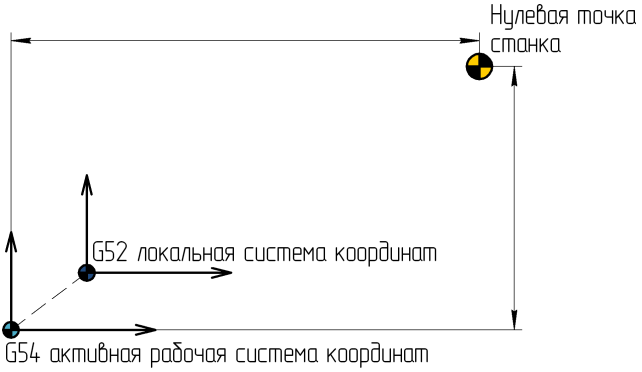



*Рис. 16.14. Локальная система координат*

G52 X_Y_Z_

Команда G52 автоматически отменяется, если программируется другая рабочая система координат G54–G59 или с помощью команды G52 X0. Y0. Z0.

**G54–G59 – стандартные рабочие системы координат.** При помощи кодов G54, G55, G56, G57, G58 и G59 определяется, в какой рабочей системе координат будет производиться обработка детали. Подробную информацию об этих кодах и о взаимосвязи рабочей системы координат с системой координат станка вы можете найти в главе 3. Путем выбора различных координатных систем программист может при помощи одной и той же программы обрабатывать различные детали. Если была выбрана одна из координатных систем G54–G59, то она действует до тех пор, пока не будет активирована другая координатная система.

**G60 – позиционирование в одном направлении.** С помощью команды G60 ко всем запрограммированным позициям по каждой оси можно перемещаться из определенного направления («+» или «–»). Благодаря этому появляется возможность исключить ошибки позиционирования, которые могут возникать из-за мертвого хода в системах сервопривода. Чаще всего направление и величина перемещения задаются параметрами СЧПУ.

**G61 – режим точного останова.** Команда G61 предназначена для включения режима точного останова. Функция точного останова подробно описана в характеристике кода G09. Единственная разница между кодами G61 и G09 заключается в том, что G09 является немодальной командой, то есть действует только в определенном кадре. Модальный код G61 остается активным, пока не будет запрограммирована команда на изменение этого режима, например с помощью кода G63 для включения режима нарезания резьбы метчиком или кода G64 режима резания.

**G63 – режим нарезания резьбы метчиком.** Режим нарезания резьбы метчиком активируется при помощи кода G63 и используется в циклах нарезания резьбы. В этом режиме невозможна корректировка скорости подачи при помощи

специальной рукоятки на панели УЧПУ станка. Режим отменяется программированием команды режима резания G64.

**G64 – режим резания.** Стандартный режим резания активируется кодом G64. С помощью этого кода отменяются другие специальные режимы – режим нарезания резьбы метчиком и режим точного останова.

**G65 – немодальный вызов макропрограммы.** Код G65 позволяет выполнить макропрограмму, находящуюся в памяти СЧПУ. Формат для немодального вызова макропрограммы выглядит следующим образом:

G65 P_L_

где G65 – команда для вызова макропрограммы; P – номер макропрограммы; L – количество выполнений макропрограммы.

Если L не указывается, то СЧПУ считает, что L = l.

**G66 – модальный вызов макропрограммы.** Команда G66 предназначена для вызова макропрограммы, как и команда G65. Единственная разница между двумя этими кодами заключается в том, что G66 является модальным кодом и макропрограмма выполняется при каждом перемещении, пока не будет запрограммирована команда G67. Формат для модального вызова макропрограммы:

G66 P_L_

где G66 – команда для вызова макропрограммы; P – номер макропрограммы; L – количество выполнений макропрограммы.

Если L не указывается, то СЧПУ считает, что L = 1.

**G67 – отмена модального вызова макропрограммы.** При помощи кода G67 отменяется режим модального вызова макропрограммы G66.

**G68 – вращение координат.** Модальная команда G68 позволяет выполнить поворот координатной системы на определенный угол. Для выполнения такого поворота требуется указать плоскость вращения, центр вращения и угол поворота.

Плоскость вращения устанавливается при помощи кодов G17 (плоскость XY), G18 (плоскость XZ) и G19 (плоскость YZ). Если желаемая плоскость вращения уже активирована, то программирование команд G17, G18 и G19 в кадре с G68 не требуется.

При действующей команде G90 центр вращения указывается абсолютными координатами относительно нулевой точки станка, если не выбрана одна из стандартных рабочих систем координат. Если выбрана одна из рабочих систем координат G54–G59, то центр вращения устанавливается относительно нулевой точки активной рабочей системы координат. В случае действующей команды G91 центр вращения указывается относительно текущей позиции. Если же координаты центра вращения не будут указаны, то в качестве центра вращения будет принята текущая позиция.

Угол вращения указывается при помощи R-слова данных. Формат для команды вращения координат обычно следующий:

G17 G68 X_Y_R_

**G69 – отмена вращения координат.** При помощи кода G68 отменяется режим вращения координат.

**G73–G89 – постоянные циклы**

| G код | Описание |
|---|---|
| G80 | Отмена постоянного цикла |
| G81 | Стандартный цикл сверления |
| G82 | Сверление с выдержкой |
| G83 | Цикл прерывистого сверления |
| G73 | Высокоскоростной цикл прерывистого сверления |
| G84 | Цикл нарезания резьбы |
| G74 | Цикл нарезания левой резьбы |
| G85 | Стандартный цикл растачивания |

О работе с постоянными циклами сверления, растачивания и нарезания резьбы вы можете узнать из главы 8.

**G90 – режим абсолютного позиционирования.** В режиме абсолютного позиционирования G90 перемещения исполнительных органов производятся относительно нулевой точки станка или относительно нулевой точки рабочей системы координат G54–G59. Код G90 является модальным и отменяется при помощи кода относительного позиционирования G91.

**G91 – режим относительного позиционирования.** При помощи кода G91 активируется режим относительного (инкрементального) позиционирования. При относительном способе отсчета за нулевое положение каждый раз принимается положение исполнительного органа, которое он занимал перед началом перемещения к следующей опорной точке. Код G91 является модальным и отменяется при помощи кода абсолютного позиционирования G90.

**G92 – смещение абсолютной системы координат.** Возникают ситуации, когда у оператора станка появляется необходимость установить определенные значения в регистрах абсолютной системы координат для перемещения нулевой точки в новое положение. Дело в том, что не все станки имеют набор из стандартных рабочих систем координат, устанавливаемых с помощью кодов G54–G59. Команда G92 применялась на станках еще до появления функции работы с несколькими стандартными системами координат.

Код G92 используют для сдвига текущего положения нулевой точки путем изменения значений в регистрах рабочих смещений. Когда СЧПУ выполнит команду G92, то значения в регистрах смещений изменятся и станут равными значениям, которые определены X-, Y- и Z-словами данных. Самое главное – учтите, что X-, Y-, Z-слова данных будут показывать текущее положение инструмента в новой координатной системе.

G92X Y Z

Рассмотрим, как работает команда G92, на конкретном примере. На рис. 16.15 изображены две заготовки. Заготовка А находится в начальной нулевой точке, ко-

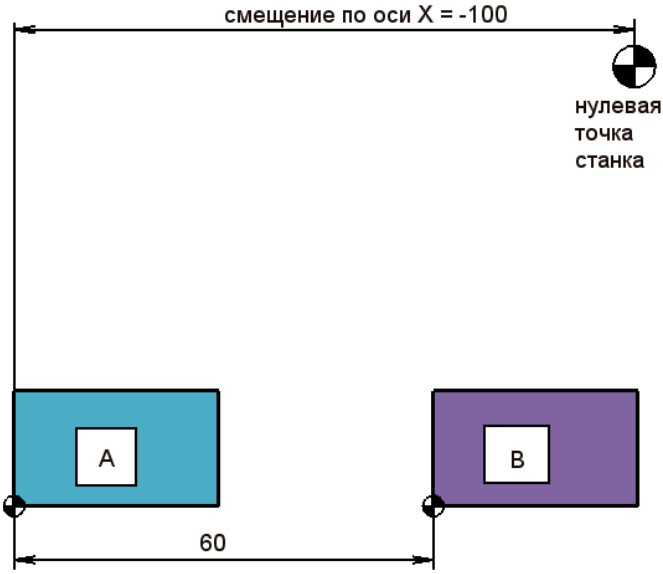



*Рис. 16.15. С помощью G92 мы заменяем регистры абсолютной позиции станка и смещаем нулевую точку*

торую установил оператор. Нам необходимо обработать заготовку B, которая расположена на 60 мм правее заготовки A, путем смещения начальной нулевой точки.

Сначала переместим инструмент в известную нам начальную нулевую точку, а затем используем G92:

```
...
G00 X0 Y0
G92 X-60 Y0
...
```

Кадр G92 X-60 Y0 означает, что новое текущее положение инструмента определено координатами (–60; 0), то есть на 60 мм левее требуемой нулевой точки. Таким образом, искомая нулевая точка будет находиться на 60 мм правее текущего положения инструмента.

Существует другой метод для достижения этого же результата. Можно сначала переместить инструмент в позицию, которую мы хотим сделать новой нулевой точкой, и затем выполнить команду G92 X0 Y0.

```
...
G00 X60 Y0
G92 X0 Y0
...
```

Команда G92 сама по себе не вызывает осевых перемещений. Указанное при помощи G92 смещение координатной системы на большинстве станков может быть отменено возвратом в нулевую точку или выключением станка.

**G94 – скорость подачи в дюймах/миллиметрах в минуту.** При помощи команды G94 указанная скорость подачи устанавливается в дюймах за 1 минуту или в миллиметрах за 1 минуту.

Если действует дюймовый режим G20, то скорость подачи F определяется как подача в дюймах за 1 минуту. Если же активен метрический режим G21, то скорость подачи F определяется как подача в миллиметрах за 1 минуту.

G20 F10 – скорость подачи 10 дюймов в минуту; G21 F10 – скорость подачи 10 миллиметров в минуту.

Модальный код G94 остается активным до тех пор, пока не будет запрограммирован код G95.

**G95 – скорость подачи в дюймах/миллиметрах на оборот.** При помощи команды G95 указанная скорость подачи устанавливается в дюймах на 1 оборот шпинделя или в миллиметрах на 1 оборот шпинделя. То есть скорость подачи F синхронизируется со скоростью вращения шпинделя S. При одном и том же значении F скорость подачи будет увеличиваться при увеличении числа оборотов шпинделя.

G20 F0.1 – скорость подачи равна 0.1 дюйма на оборот; G21 F0.1 – скорость подачи равна 0.1 миллиметра на оборот.

Модальный код G95 остается активным до тех пор, пока не будет запрограммирован код G94.

**G98 – возврат к исходной плоскости в цикле.** Если постоянный цикл станка работает совместно с кодом G98, то инструмент возвращается к исходной плоскости в конце каждого цикла и между всеми обрабатываемыми отверстиями. Исходная плоскость – это координата по оси Z (уровень), в которой находится инструмент перед вызовом постоянного цикла. Команда G98 отменяется при помощи команды G99.

**G99 – возврат к плоскости отвода в цикле.** Если цикл сверления работает совместно с кодом G99, то инструмент возвращается к плоскости отвода между всеми обрабатываемыми отверстиями. Плоскость отвода – это координата по оси Z (уровень), с которой начинается сверление на рабочей подаче и в которую возвращается инструмент после того, как он достиг дна обрабатываемого отверстия. Плоскость отвода обычно устанавливается в кадре цикла с помощью R-адреса. Команда G99 отменяется при помощи команды G98.

## 16.2. Адреса/слова данных

**X** является командой осевого перемещения. Как правило, за X принимают ось, вдоль которой возможно наибольшее перемещение исполнительного органа станка. При этом ось X перпендикулярна к оси Z и параллельна плоскости рабочего стола.

Положительное или отрицательное число, входящее в состав этого слова данных, определяет конечную позицию исполнительного органа станка вдоль оси X. В кадре можно запрограммировать X только один раз. Если в одном кадре будет

несколько команд X, то СЧПУ будет работать с последней из них (которая ближе к знаку конца кадра).

Пример:

G01 G90 X100 F200 – линейное перемещение в координату X = 200 со скоростью 200 мм/мин.

Когда X находится в одном кадре с кодом выдержки G04, то оно определяет время этой выдержки в секундах (паузы).

Пример:

G04 X5.0 – выполнить выдержку продолжительностью 5 секунд.

**Y** является командой осевого перемещения. Ось Y перпендикулярна осям X и Z. Положительное или отрицательное число, входящее в состав этого слова данных, определяет конечную позицию исполнительного органа станка вдоль оси Y. В кадре можно запрограммировать Y только один раз. Если в кадре будут указаны несколько команд Y, то СЧПУ будет работать с последней из них (которая ближе к знаку конца кадра).

Пример:

G01 G90 Y102 F200 – линейное перемещение в координату Y = 102 со скоростью 200 мм/мин.

**Z** является командой осевого перемещения. В качестве положительного направления оси Z принимают вертикальное направление вывода инструмента (например, сверла) из заготовки. То есть ось Z всегда связана со шпинделем станка. Положительное или отрицательное число, входящее в состав этого слова данных, определяет конечную позицию исполнительного органа станка вдоль оси Z. В кадре можно запрограммировать Z только один раз. Если в кадре будут указаны несколько команд Z, то СЧПУ будет работать с последней из них (которая ближе к знаку конца кадра).

Пример:

G01 G90 Z0.5 F200 – линейное перемещение в координату Z = 0.5 со скоростью 200 мм/мин.

**A**, **B**, **C** являются командами кругового перемещения. Под круговым перемещением понимается угловое перемещение (поворот) оси шпинделя фрезерного станка или угловое перемещение (поворот) управляемого поворотного стола (4-ая ось).

Круговые перемещения инструмента обозначают латинскими буквами – A (вокруг оси X), B (вокруг оси Y) и C (вокруг оси Z). Положительные направления вращений вокруг этих осей определяются очень просто. Если расположить большой палец по направлению оси, то другие согнутые пальцы покажут положительное направление вращения.

Пример:

G01 G90 C90 F200 – поворот стола на 90° со скоростью 200 мм/мин.

Для некоторых СЧПУ адрес C может являться командой на выполнение фаски при действующей линейной интерполяции. Числовое значение, входящее в состав C-слова данных, определяет размер фаски.

**I**, **J**, **K** применяются во время круговой интерполяции и служат для указания относительных расстояний от начальной точки дуги до ее центра. Слово данных с I относится к оси X, слово данных с J – к оси Y, а слово данных с K – к оси Z. При этом в зависимости от расположения дуги значения могут быть положительными или отрицательными.

**R.** При действующей круговой интерполяции (G02/G03) R определяет радиус, который соединяет начальную и конечную точки дуги.

Для некоторых СЧПУ адрес R может являться командой на выполнение скругления при действующей линейной интерполяции. Числовое значение, входящее в состав R-слова данных, определяет радиус скругления.

В постоянных циклах R определяет положение плоскости отвода. При работе с командой вращения координат R определяет угол поворота координатной системы.

**P** обычно используется в постоянных циклах обработки отверстий и определяет время выдержки (паузы) на дне отверстия. Числовое значение, входящее в состав P-слова данных, обычно определяет время выдержки в 1/1000 секунды.

Когда P появляется в одном кадре с кодом вызова подпрограммы M98, то оно обозначает номер вызываемой подпрограммы. В ряде случаев это же слово данных может указывать на частоту вызова подпрограммы.

Пример:

M98 P1001 – вызов подпрограммы O1001.

**Q** часто используется в циклах прерывистого сверления и определяет относительную глубину каждого рабочего хода инструмента.

В цикле растачивания Q определяет расстояние сдвига расточного инструмента от стенки обработанного отверстия для обеспечения аккуратного вывода инструмента из отверстия.

При помощи **D** выбирается значение коррекции на радиус инструмента. Коррекция радиуса инструмента активируется командами G41 и G42. При помощи команды D00 можно отменить действующую коррекцию.

При помощи **H** выбирается значение компенсации длины инструмента. Компенсация длины инструмента обычно активируется командой G43. При помощи команды H00 можно отменить действующую компенсацию длины инструмента.

Для определения скорости подачи служит **F-адрес**. Если в одном кадре будут запрограммированы несколько скоростей подач, то СЧПУ будет работать с последней из них. В случае программирования F с кодом G94 скорость подачи будет установлена в дюймах (G20) или миллиметрах (G21) в минуту (минутная подача). А в случае использования с G95 скорость подачи будет установлена в дюймах (G20) или миллиметрах (G21), наоборот. F-адрес является модальным, то есть установленная скорость подачи остается неизменной до тех пор, пока не указано новое числовое значение вместе с F или не изменен режим перемещений при помощи G00.

С помощью **S** определяется число оборотов шпинделя. S-адрес является модальным, то есть установленное число оборотов остается неизменным до тех пор, пока не указано новое числовое значение вместе с S.

При помощи **T** осуществляется управление магазином инструментов. Числовое значение с T определяет номер инструмента (ячейки), который необходимо переместить в позицию смены путем поворота инструментального магазина. Обычно T программируют в одном кадре с командой смены инструмента M06. В этом случае числовое значение при T будет определять номер инструмента, который необходимо вызвать из магазина и установить в шпиндель.

Пример:

T2 M06 – вызвать инструмент № 2.

Как правило, адрес **O** указывает системе ЧПУ на номер управляющей программы.

Пример:

O2007 – программа обработки будет зарегистрирована в памяти СЧПУ под номером 2007.

С помощью **N** производится нумерация кадров УП. При использовании номера кадра он может быть поставлен в кадре в любую позицию, но обычно его указывают в самом начале. Номер кадра не влияет на работу станка, а помогает оператору ориентироваться в содержании программы обработки.

## 16.3. M-коды

**M00 – запрограммированный останов.** Когда СЧПУ исполняет команду M00, то происходит так называемый запрограммированный останов. Все осевые перемещения останавливаются и возобновляются лишь после того, как оператор станка нажмет клавишу **Старт цикла** на панели УЧПУ. При этом шпиндель продолжает вращаться (у большинства станков) и другие функции остаются активными. Если оператор станка нажимает клавишу **Старт цикла**, то выполнение программы будет продолжено с кадра, следующего за M00.

**M01 – останов по выбору.** Код M01 предназначен для останова по выбору. Действует он аналогично коду M00, однако предоставляет выбор оператору – нужно или не нужно прерывать выполнение управляющей программы. На панели УЧПУ практически любого станка имеется клавиша (или переключатель) **M01**. Если эта клавиша нажата, то при чтении кадра с M01 происходит останов. Если же клавиша не нажата, то кадр M01 пропускается и выполнение УП не прерывается.

**M02 – конец программы.** Код M02 информирует СЧПУ о завершении программы.

**M03 – прямое вращение шпинделя.** При помощи кода M03 включается прямое (по часовой стрелке) вращение шпинделя с запрограммированным числом

оборотов (S-слово). Команда M03 остается действующей до тех пор, пока она не будет отменена с помощью M04 или M05.

**M04 – обратное вращение шпинделя.** При помощи кода M04 включается обратное (против часовой стрелки) вращение шпинделя с запрограммированным числом оборотов (S-слово). Команда M04 остается действующей до тех пор, пока она не будет отменена с помощью M03 или M05.

**M05 – останов шпинделя.** Команда M05 останавливает вращение шпинделя, но не останавливает осевые перемещения (за исключением режима G95).

**M06 – автоматическая смена инструмента.** При помощи команды M06 инструмент, закрепленный в шпинделе, меняется на инструмент, находящийся в положении готовности в магазине инструментов.

**M07 – включение подачи СОЖ в распыленном виде.** Команда M07 включает подачу СОЖ в зону обработки в распыленном виде, если станок обладает такой возможностью.

**M08 – включение подачи СОЖ.** Команда M08 включает подачу СОЖ в зону обработки в виде струи.

**M09 – выключение подачи СОЖ.** Команда M09 выключает подачу СОЖ и отменяет команды M07 и M08.

**M19 – юстировка шпинделя.** При помощи команды M19 осуществляется радиальная юстировка шпинделя (поворот в определенное положение), чтобы выставить приводной носик шпинделя на позицию смены инструмента. В этом положении шпиндель обычно зажимается и его не провернуть рукой.

**M20 – отмена юстировки шпинделя.** При помощи команды M20 отменяется команда юстировки шпинделя M19.

**M30 – конец программы.** Код M30 информирует СЧПУ о завершении программы.

**M98 – вызов подпрограммы.** Команда M98 предназначена для вызова подпрограммы. Вместе с этой командой программируется P-слово данных, которое обозначает номер вызываемой подпрограммы.

Пример:

M98 P1001 – вызвать подпрограмму O1001.

**M99 – конец подпрограммы.** При помощи команды M99 по окончании подпрограммы осуществляется возврат к главной программе, из которой была вызвана подпрограмма.

# 16.4. Специальные символы в УП

**«/» – пропуск кадра.** Пропуск кадра – полезная функция, позволяющая оператору станка выбрать определенные кадры, которые не нужно выполнять. Эта функция реализуется, если в начало кадра поставить «/» (косая черта, слэш). Для

того чтобы СЧПУ не выполнила кадр, в самом начале которого стоит «/», необходимо, чтобы специальный переключатель **Пропуск кадра** на панели УЧПУ станка находился в положении **Вкл.**.

В этом случае система пропускает данный кадр и переходит к выполнению следующего. Если же переключатель **Пропуск кадра** на панели УЧПУ станка находится в положении **Выкл.**, то кадр с кодом «/» отрабатывается как обычно. Таким образом, с помощью кода «/» и переключателя **Пропуск кадра** оператор станка может влиять на выполнение УП.

Пример:

```
...
N50 G81 X0 Y0 Z-10 R2 F50
N60 X10 Y20
N70 X10 Y30
/N80 X20 Y20
/N90 X20 Y30
N100 M05
...
```

| Положение переключателя **Пропуск кадра** | Поведение станка |
|---|---|
| Вкл. | Кадры с кодом «/» не будут выполнены |
| Выкл. | Кадры с кодом «/» будут выполнены нормально |

«;» **или** «(...)» – **комментарии в УП.** Комментарии представляют собой обычные предложения, при помощи которых программист доводит до оператора станка определенную технологическую информацию. Как правило, в комментариях содержатся следующие данные:

- дата и время создания УП;
- номер чертежа;
- материал заготовки;
- данные о рабочей системе координат;
- размеры инструмента;
- названия технологических операций.

Для того чтобы ввести комментарии в УП, необходимо использовать специальные символы (знаки) программирования. В качестве таких символов для большинства СЧПУ применяются круглые скобки или точка с запятой. Перед символами комментариев не принято ставить номера кадров, если комментарии занимают кадр полностью.

# Глава 17
# ПОЛЕЗНЫЕ ПРОГРАММЫ

## 17.1. Мониторинг ЧПУ

Сегодня большинство отечественных машиностроительных компаний обладают современным станочным парком, будь то небольшой цех, состоящий из нескольких станков с числовым программным управлением (ЧПУ), или крупное производство, оснащенное сотней обрабатывающих центров. Естественно, инвестируя в производство, администрация любого предприятия надеется получить максимум качественной продукции в как можно более сжатые сроки, чтобы быстрее «отбить» вложенные средства. Этого можно добиться лишь при условии четкой организации технологического цикла, бесперебойной работы цеховых служб и оборудования. До недавнего времени задача контроля над станками с ЧПУ и над производственным персоналом решалась довольно примитивно.

*Рис. 17.1. Электронный блок мониторинга на станке Leadwell*

Мастера участков сами писали отчеты о планах и загрузке станков, ставили «восьмерки» в табельный лист, обычно особенно не заботясь о точности полученной информации, или, наоборот, озвучивали «правильные» цифры, выгодные им самим. И до сих пор на некоторых заводах можно увидеть прикрепленные к стойке ЧПУ бумажные листочки с расчерченными таблицами, куда операторы под конец смены ставят галочки, отчитываясь о выполненной работе. Естественно, что при таком подходе и речи быть не может о достоверности информации, которую получает администрация предприятия.

Программно-аппаратный комплекс «Мониторинг ЧПУ» – это новое поколение уже известной системы «Foreman MDC», позволяющий руководству предприятия получать в виде наглядных графиков и отчетов точные данные о загрузке и причинах простоя оборудования, а цеховым службам – оперативную информацию об аварийных и нештатных ситуациях на производстве. Технологам и нормировщикам существенно помогут автоматически сохраняемые в базе данных сведения о фактическом времени обработки деталей и режимах резания. Вся полученная информация хранится на сервере и может быть использована в системе документооборота предприятия.

## Возможности

Комплекс мониторинга одинаково хорошо работает как с современными станками с ЧПУ, так и с оборудованием прошлых поколений, включая универсальные станки. Обеспечивается круглосуточный мониторинг со следующими возможностями:

- дистанционный контроль работы станков в режиме реального времени на ПК;
- автоматическое вычисление коэффициента загрузки одного станка или группы станков;
- выявление причин и продолжительности простоя производственного оборудования;
- генерация табличных, текстовых и графических отчетов для администрации предприятия;
- определение фактического машинного и вспомогательного времени обработки деталей;
- автоматическое оповещение об аварийных и нештатных ситуациях по электронной почте;
- интеллектуальная фото- и видеозапись производственного процесса;
- авторизация операторов и документации;
- архив управляющих программ с функцией редактирования и приема/передачи на станки;
- возможность оценки производственной деятельности по критерию план–факт.

## Контроль в режиме реального времени

Установив на персональный компьютер программу «Мониторинг онлайн», вы получаете возможность непосредственно со своего рабочего места в режиме реаль-

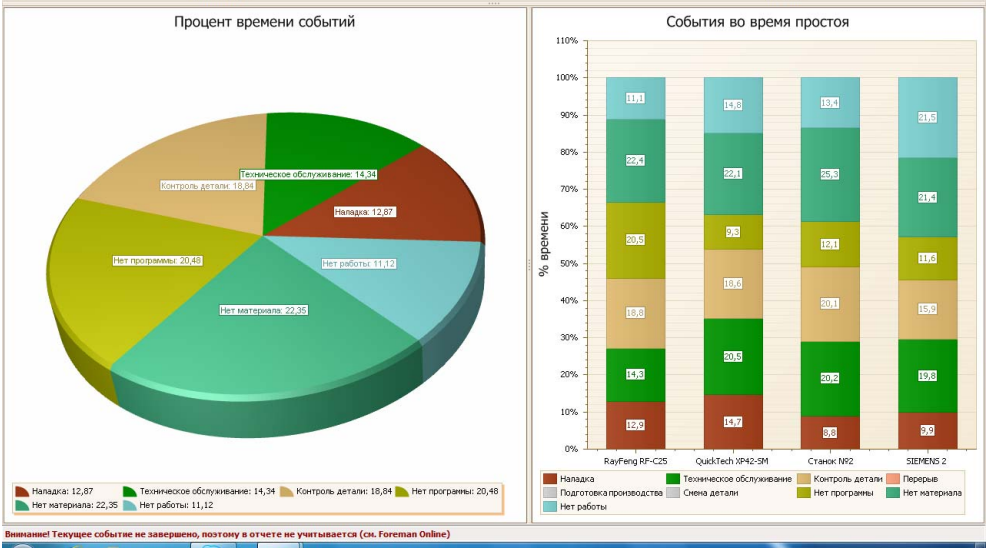
Обоснование простоев (06.09.10 08:18 - 10.09.10 01:28)

Файл Расчет Вид Экспорт

| Номер станка | Наименование станка | Время простоя |
|---|---|---|
| 1 | RayFeng RF-C25 | 23:44:19 |

| Наименование события | Время простоя | % от всего времени простоя |
|---|---|---|
| Наладка | 3:03:20 | 12,87 |
| Техническое обслуживание | 3:24:19 | 14,34 |
| Контроль детали | 4:28:20 | 18,84 |
| Нет программы | 4:51:40 | 20,48 |
| Нет материала | 5:18:20 | 22,35 |
| Нет работы | 2:38:20 | 11,12 |

| | | |
|---|---|---|
| 2 | QuickTech XP42-SM | 20:37:20 |
| 3 | Станок №2 | 16:20:00 |




Внимание! Текущее событие не завершено, поэтому в отчете не учитывается (см. Foreman Online)

*Рис. 7.2. Пример отчета о загрузке оборудования и причинах простоя*

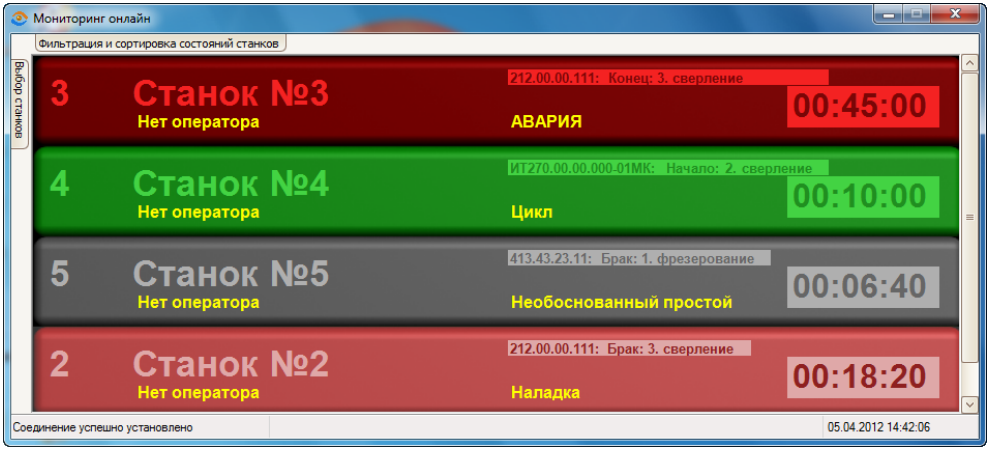



*Рис. 17.3. Окно приложения «Мониторинг онлайн»*

ного времени контролировать события, происходящие с цеховым оборудованием. Отпадает утомительная необходимость каждое утро совершать обход всего станочного парка: достаточно просто щелкнуть клавишей мыши – и вся информация о работе станков и операторов в наглядном виде отображается на мониторе. Возможен и удаленный мониторинг: чтобы в любой точке планеты получать оперативную информацию о состоянии производства, достаточно иметь лишь подключение к сети Интернет.

## Формирование отчетов и графиков

Программа «Генератор отчетов» позволяет администрации предприятия правильно оценивать различные технико-экономические параметры производства и осуществляет выпуск отчетной документации. Работа с программой очень проста и состоит из трех этапов:

- выбор объектов – следует указать названия станков или фамилии операторов для формирования отчета или графика;
- выбор временного интервала – требуется с помощью встроенного календаря задать дату/время начала и конца отчетного периода;

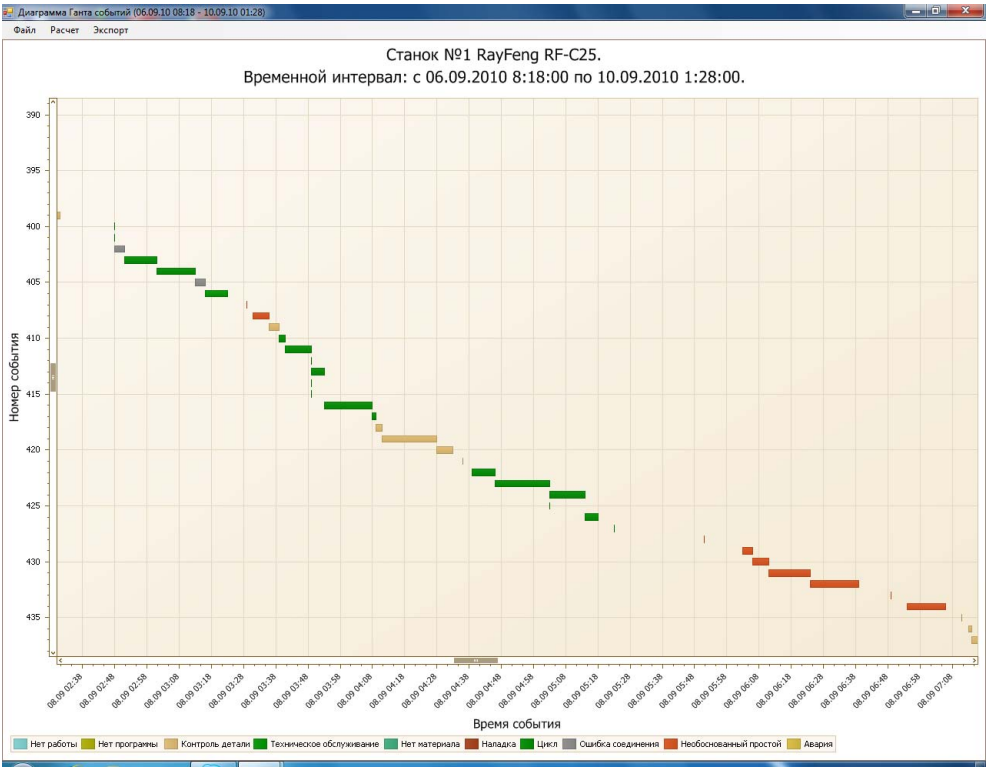



*Рис. 17.4. Диаграмма Ганта событий с конкретным станком*

- выбор типа отчета – для формирования отчета на основе информации, хранящейся в базе данных на сервере, достаточно дважды щелкнуть клавишей мыши по ярлыку.

В стандартную поставку комплекса входят несколько базовых отчетов: «Эффективность использования оборудования», «Обоснованный простой», «Работа вне графика», «Необоснованный простой», «Все события станка», «Отчет по авариям», «Эффективность работы операторов». Сформированные отчеты, содержащие красочные 3D-диаграммы и табличные данные, могут быть сохранены в различных форматах или распечатаны на принтере. Удобный механизм плагинов позволяет заказывать разработчику любые необходимые именно вашему производству типы отчетов и быстро добавлять их в систему.

## *Ускорение работы цеховых служб*

Однако сказанное отнюдь не означает, что комплекс предназначен исключительно для мониторинга оборудования и персонального контроля над операторами. Функция автоматического оповещения позволяет существенно ускорить работу некоторых цеховых служб и оперативно влиять на возникающие проблемы. К примеру, если на станке произойдет авария, система автоматически отправит текстовое сообщение на персональный компьютер главного инженера или

Отчет по авариям (06.09.10 08:18 - 10.09.10 01:28)

Файл Расчет Экспорт

| № станка | Наименование станка | Начало аварии | Конец аварии | Длит-ть | Оператор | Спецсобытие | Состояние | Наименование события |
|---|---|---|---|---|---|---|---|---|
| 1 | RayFeng RF-C25 | 06.09 08:43:01 | 06.09 09:06:21 | 0:23:20 | Нет оператора | нет | стоп | Нет работы |
| | | 06.09 11:01:21 | 06.09 11:09:41 | 0:08:20 | | | | Контроль детали |
| | | 06.09 11:09:41 | 06.09 11:14:41 | 0:05:00 | Пушкин С.А. | | | |
| | | 06.09 11:14:41 | 06.09 11:18:01 | 0:03:20 | Нет оператора | | | |
| | | 06.09 12:39:41 | 06.09 12:43:01 | 0:03:20 | Краснов Г.А. | | старт | |
| | | 06.09 13:13:01 | 06.09 13:26:21 | 0:13:20 | | | стоп | Нет работы |
| | | 06.09 13:26:21 | 06.09 13:26:21 | 0:00:00 | Нет оператора | | | |
| | | 06.09 13:26:21 | 06.09 13:28:01 | 0:01:40 | Пушкин С.А. | | | |
| | | 06.09 13:28:01 | 06.09 13:29:41 | 0:01:40 | Нет оператора | | | |
| | | 06.09 16:26:21 | 06.09 16:28:01 | 0:01:40 | | | старт | |
| | | 06.09 16:28:01 | 06.09 16:33:01 | 0:05:00 | Пушкин С.А. | | | |
| | | 06.09 19:01:21 | 06.09 19:09:41 | 0:08:20 | Нет оператора | | стоп | Контроль детали |
| | | 06.09 19:36:21 | 06.09 19:39:41 | 0:03:20 | Пушкин С.А. | | | |
| | | 06.09 23:14:41 | 06.09 23:14:41 | 0:00:00 | Нет оператора | | | |
| | | 06.09 23:14:41 | 06.09 23:14:41 | 0:00:00 | Краснов Г.А. | | | |
| | | 07.09 00:51:21 | 07.09 00:54:41 | 0:03:20 | Савостиков А.В. | | старт | |
| | | 07.09 02:56:21 | 07.09 02:56:21 | 0:00:00 | Краснов Г.А. | | стоп | Нет материала |
| | | 07.09 03:04:41 | 07.09 03:04:41 | 0:00:00 | Нет оператора | | | |
| | | 07.09 03:24:41 | 07.09 03:24:41 | 0:00:00 | | | | Наладка |
| | | 07.09 03:28:01 | 07.09 03:28:01 | 0:00:00 | Пушкин С.А. | | | |
| | | 07.09 03:29:41 | 07.09 03:29:41 | 0:00:00 | Нет оператора | | | |
| | | 07.09 04:53:01 | 07.09 04:53:01 | 0:00:00 | Пушкин С.А. | | | Техническое обслужи... |
| | | 07.09 08:51:21 | 07.09 08:56:21 | 0:05:00 | Нет оператора | | старт | |
| | | 07.09 08:56:21 | 07.09 08:58:01 | 0:01:40 | Краснов Г.А. | | | |
| | | 07.09 12:21:21 | 07.09 12:43:01 | 0:21:40 | Пушкин С.А. | | | |
| | | 07.09 12:43:01 | 07.09 12:48:01 | 0:05:00 | Нет оператора | | | |
| | | 07.09 13:14:41 | 07.09 13:14:41 | 0:00:00 | Савостиков А.В. | | стоп | Техническое обслужи... |
| | | 07.09 13:18:01 | 07.09 13:18:01 | 0:00:00 | Нет оператора | | | |
| | | 07.09 13:26:21 | 07.09 13:29:41 | 0:03:20 | | | | |
| | | 07.09 14:19:41 | 07.09 14:26:21 | 0:06:40 | Краснов Г.А. | | старт | |
| | | 07.09 14:26:21 | 07.09 14:33:01 | 0:06:40 | Нет оператора | | | |
| | | 07.09 14:33:01 | 07.09 14:44:41 | 0:11:40 | Краснов Г.А. | | | |
| | | 07.09 14:44:41 | 07.09 14:49:41 | 0:05:00 | Нет оператора | | | |
| | | 07.09 14:49:41 | 07.09 14:51:21 | 0:01:40 | Краснов Г.А. | | | |

*Рис. 17.5. Отчет об аварийных ситуациях с оборудованием*

механика. Если закончились заготовки, то оператор может нажать кнопку **Нет материала** на блоке мониторинга –тем самым он отошлет электронное сообщение в цеховую службу, отвечающую за снабжение станка заготовками. Если же технолог-программист забудет о необходимости создания и передачи на станок управляющей программы, оператор может напомнить ему об этом, нажав кнопку **Нет программы**. Таким образом, комплекс гарантирует оперативность решения возникающих проблем, позволит быстро возобновить производственный цикл и не сорвать сроки поставки продукции.

### *Внедрение на предприятии*

Современная архитектура комплекса «Мониторинг ЧПУ» позволяет быстро внедрить его на любом предприятии. При этом не важно, какое оборудование вы используете: система позволит связать в единое информационное пространство как новые импортные обрабатывающие центры, так и устаревшие советские станки с ЧПУ. Простота монтажа объясняется, во-первых, тем, что все аппаратные средства находятся внутри блока мониторинга, который крепится непосредственно на стойке ЧПУ или рядом с ней. Аналогичные продукты других фирм устанавливаются внутрь стойки, что значительно усложняет и удорожает процесс монтажа. Во-вторых, поддержка комплексом беспроводной передачи данных позволяет избежать необходимости трудоемкой протяжки кабелей по всему цеху. Важное преимущество – разработкой комплекса занимаются отечественные инженеры, к которым всегда можно обратиться за консультацией, заказать необходимый тип отчета или связать станочную базу данных с действующей на предприятии системой документооборота. В настоящий момент комплекс внедрен и успешно работает на ведущих отечественных предприятиях машиностроения. Информация о комплексе мониторинга в Интернете: http://monitoringcnc.ru.

## 17.2. Редактор УП Cimco Edit 6

Программа Cimco Edit компании Cimco Integration предназначена для редактирования, проверки и передачи УП с компьютера на станок с ЧПУ и обратно. Этот простой и надежный редактор УП обладает многими полезными инструментами, необходимыми технологу-программисту в его ежедневной работе.

Главное меню программы состоит из нескольких пунктов: **Редактор** (работа с файлами, редактирование), **Функции УП** (специализированные инструменты для редактирования УП, статистика УП), **Прорисовка** (отображение траектории инструмента), **Сравнение файлов** (нахождение различий в двух файлах), **Передача** (отправка УП на станки), **ЧПУ-калькулятор** (графический редактор с возможностью генерации УП).

Возможности редактирования в Cimco Edit типичны для любого текстового редактора: вы можете копировать, вставлять или удалять любой фрагмент УП. Также присутствуют функции поиска определенной информации в файле и автоматической замены одних данных другими. Существует возможность выделения/удаления определенного диапазона информации и перехода к нужной строке или кадру.

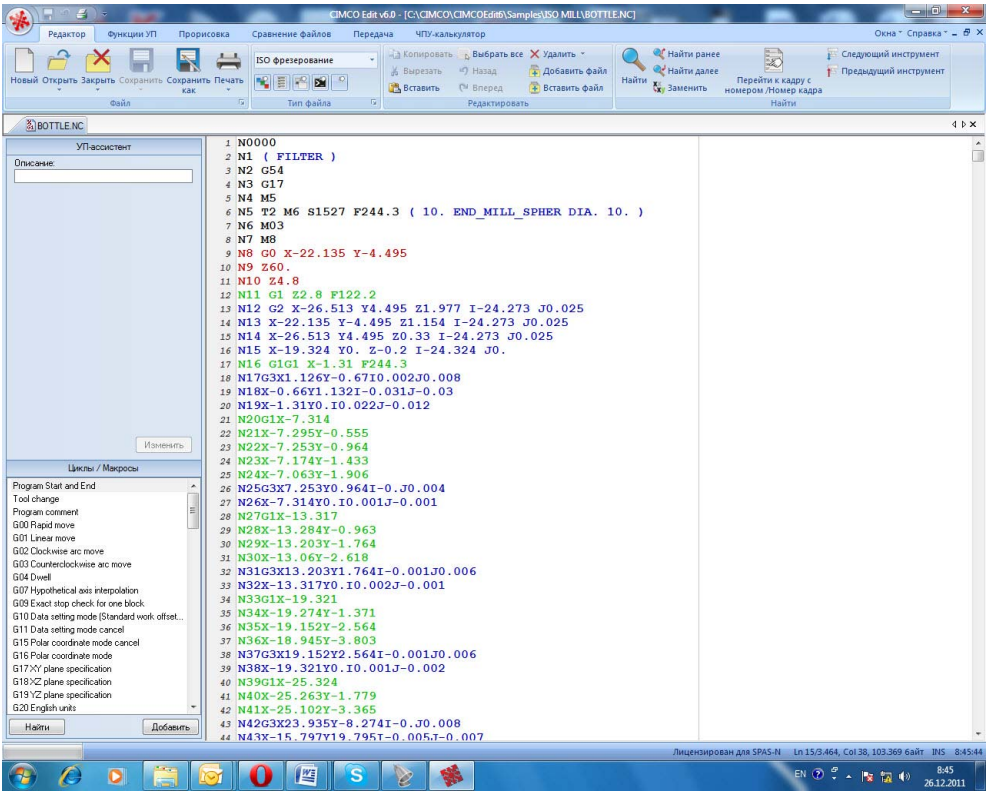



*Рис. 17.6. Интерфейс программы Cimco Edit*

Интересной, на наш взгляд, является функция «Статистика траектории», которая автоматически определяет максимальные и минимальные координаты в УП (рис. 17.7), время обработки и выводит другую полезную информацию.

Для передачи данных с персонального компьютера на станок с ЧПУ (и наоборот) воспользуйтесь пунктом **Передача УП** основного меню. Вы можете использовать Cimco Edit для связи с разными станками, нужно лишь правильно настроить параметры приема/передачи. При выборе пункта **DNC Настройка** появится окно настроек приема-передачи. В верхней части этого окна (**Станки**) вы увидите названия станков – Machine 1 и Machine 2. Здесь вы можете переименовать существующие станки, удалить их или добавить новые. В нижней части окна (**Конфигурация**) представлена информация о текущей конфигурации: протоколе передачи данных, портах и типе станка. Выберите **Machine 1** и нажмите **Настройка**. Откроется новое окно для настройки параметров приема/передачи данных выбранного станка. Закладка **Порт** настраивает порт компьютера для связи с системой ЧПУ станка. Если вы ошибетесь с настройкой параметров в этой закладке, то, скорее всего, передать программу обработки на станок не получится. Закладки

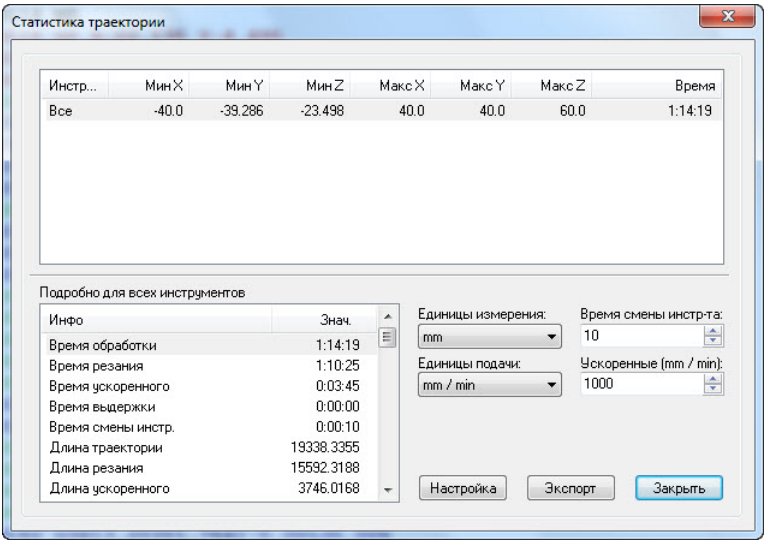



*Рис. 17.7. Статистика траектории*

**Прием** и **Передача** предназначены для настройки специальных возможностей системы, например для работы в режиме прямого числового ввода DNC.

Для передачи УП на станок выберите **Послать** – Cimco Edit начнет передавать данные, на экране появится окно состояния процесса передачи. При помощи Cimco можно передавать не только программы обработки, но и параметры станка или данные об инструменте и корректорах.

Иногда возникает необходимость сравнить два варианта одной программы или просто два текстовых файла (например, файлы параметров или постпроцессоры). Если программы большие, то найти маленькое различие визуально очень трудно. Редактор Cimco Edit поможет и в этом – пункт **Сравнение файлов** главного меню предназначен для сравнения содержимого текстовых файлов. Кнопки **Следующее различие** и **Предыдущее различие** автоматически переходят к найденным отличиям в файлах и выделяют эти места цветом (рис. 17.8).

В общем, Cimco Edit обладает массой полезных возможностей, которые могут быть дополнены и другими интересными решениями: Cimco DNC-Max, Cimco MDC, Cimco Filter. Более подробную информацию о программных продуктах Cimco вы можете найти на сайте http://cimco-software.ru.

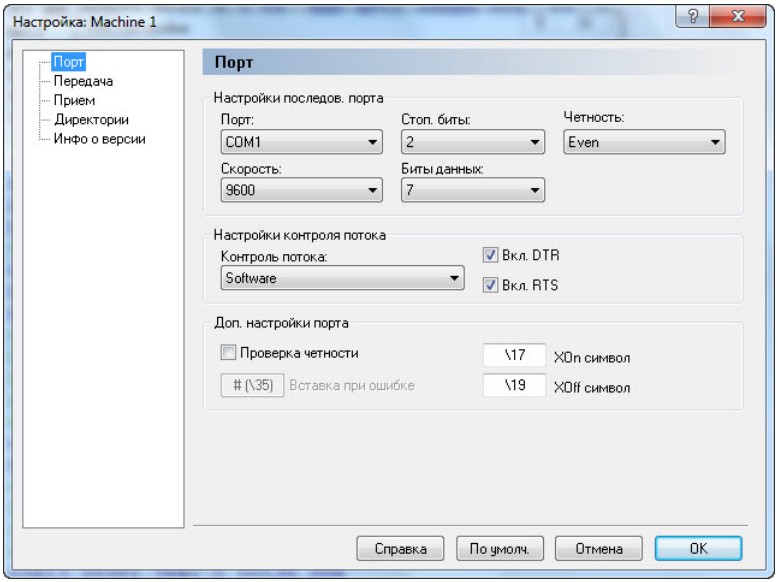



*Рис. 17.8. Окно настроек приема/передачи*

# 17.3. Техтран®

Техтран® – это семейство современных CAM-систем, объединенных общим интерфейсом и единым подходом к решению задач технологического проектирования. Простые универсальные решения и открытость данных способствовали тому, что Техтран с успехом используется предприятиями самых различных отраслей уже более тридцати лет.

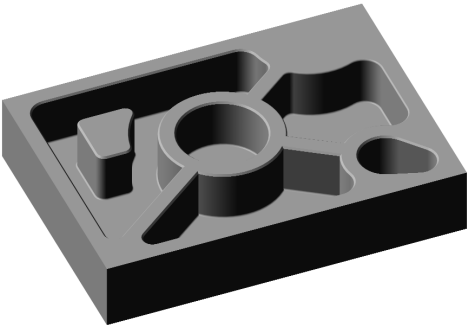
В Техтране сочетаются различные подходы к решению технологических задач. Сложные механизмы формирования десятков и сотен команд обработки для целого технологического перехода дополняются набором возможностей для программирования отдельных действий. Благодаря этому Техтран оказывается незаменим при возникновении нестандартных ситуаций и работе с нетрадиционным оборудованием.

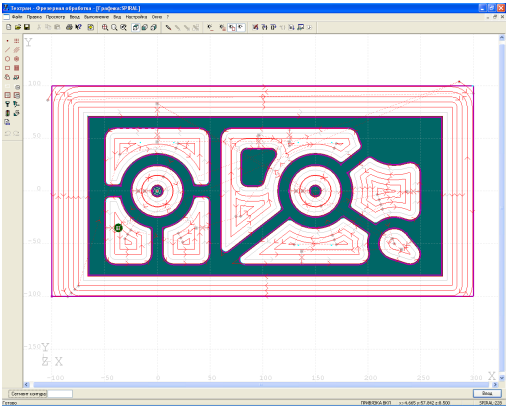
## *Фрезерная обработка*

Программа предназначена для проектирования УП 2,5-координатной обработки деталей на фрезерных, сверлильных, расточных станках с ЧПУ и обрабатывающих центрах.

- Одно- и многопроходная контурная обработка с использованием коррекции на радиус инструмента в ЧПУ.
- Выборка сплошного материала из закрытых, открытых и полуоткрытых областей.

- Послойная обработка.
- Контроль зарезания детали, учет геометрии заготовки и прижимов.
- Позиционная обработка с использованием встроенных циклов ЧПУ или формирование траектории в явном виде.
- Использование оси вращения.
- Формирование УП с использованием подпрограмм.

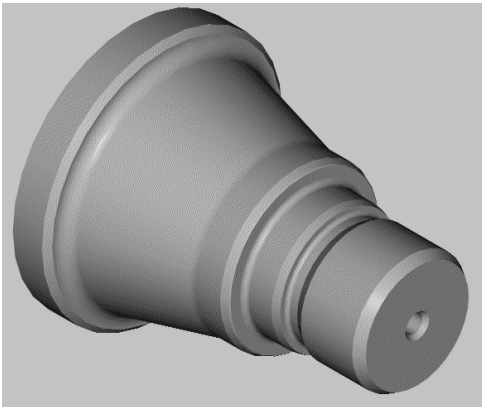


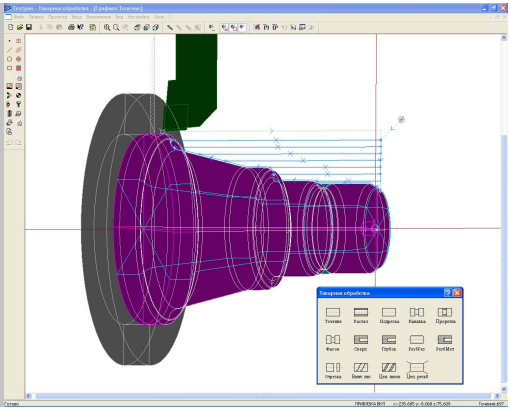


## Токарная обработка

Программа предназначена для проектирования управляющих программ обработки деталей на токарных и карусельных станках с ЧПУ и обрабатывающих центрах.

- Описание инструментов различных типов.
- Определение патрона, детали, заготовки, задней бабки, люнета.
- Попереходная обработка (точение, растачивание, отрезка, сверление и др.).
- Корректировка заготовки после каждого перемещения инструмента.
- Анализ на отсутствие столкновения инструмента с патроном, деталью и заготовкой (на ускоренных перемещениях), контроль зарезания детали.



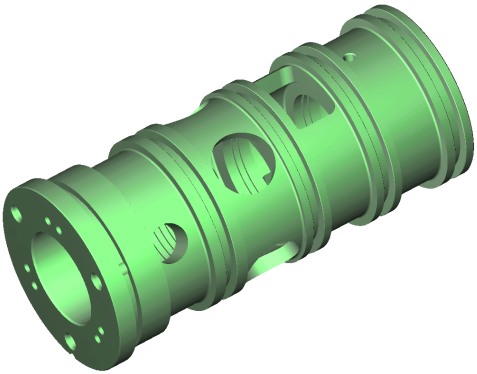


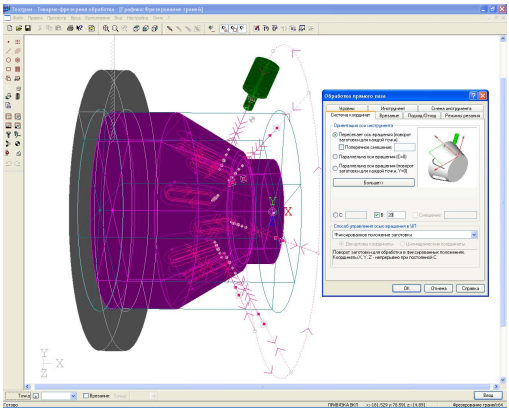
- Режим корректировки геометрии детали.
- Использование встроенных циклов ЧПУ или формирование траектории в явном виде.
- Использование двух шпинделей и двух револьверных голов.

## Токарно-фрезерная обработка

Программа предназначена для проектирования управляющих программ обработки деталей на токарно-фрезерных центрах с ЧПУ.

- Согласованное проектирование переходов токарной и фрезерной обработки.
- Совмещение геометрии токарных и фрезерных элементов.
- Фрезерная обработка с использованием оси вращения.
- Обработка радиусных и винтовых пазов за счет использования оси вращения.
- Задание способа управления осью вращения в УП.
- Использование цилиндрической и полярной интерполяции в УП.
- Определение точки смены инструмента.
- Безопасное перемещение инструмента между переходами с учетом геометрии заготовки и зажимного приспособления.

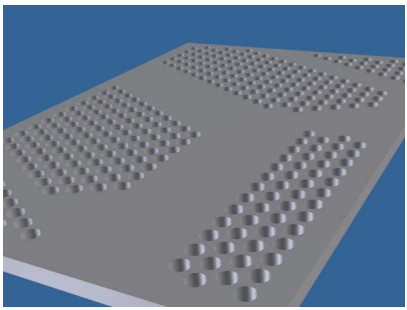




## Многошпиндельное сверление

Программа предназначена для проектирования управляющих программ обработки деталей на многошпиндельных станках с ЧПУ.

- Восстановление прямоугольной сетки отверстий по отверстиям, расположенным на границе области заполнения.

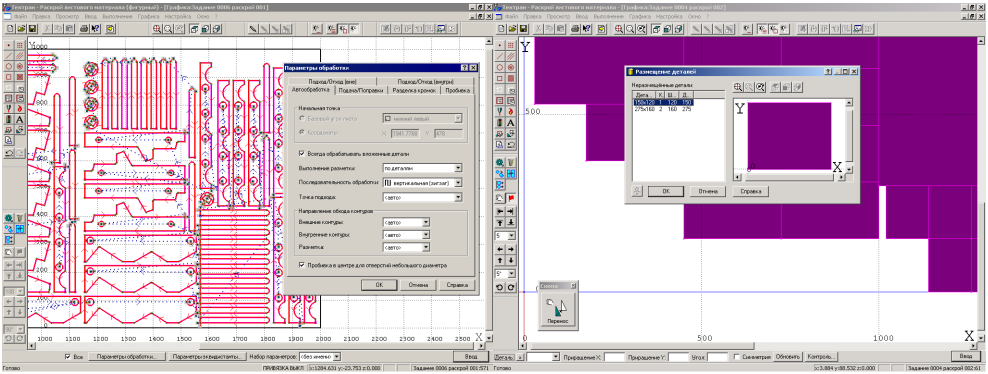


- Анализ и исправление неточностей в данных чертежа.
- Генерация различных вариантов обработки.
- Подбор наилучшего варианта обработки по нескольким критериям.
- Получение УП для наилучшего варианта.

## Раскрой листового материала

Программа предназначена для комплексного решения задачи раскроя листового материала. Она сочетает возможности системы подготовки управляющих программ с функциями организации производственного процесса.

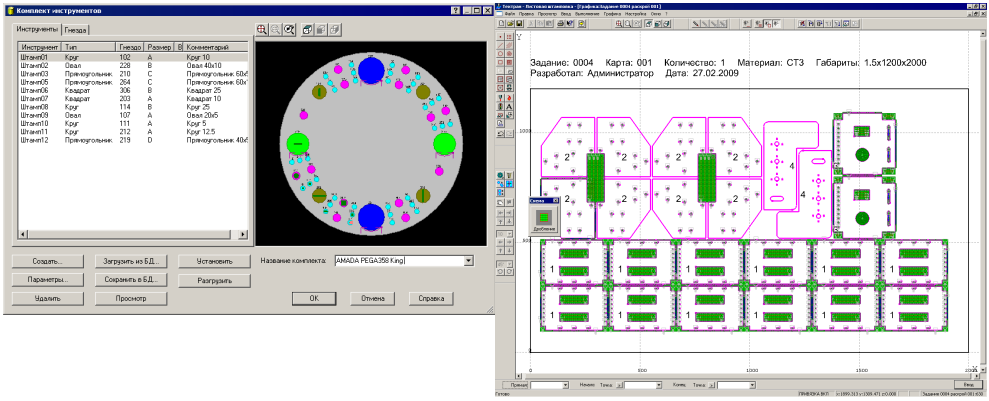
- **Фигурный раскрой.** Проектирование УП для машин термо- и гидрорезки
- **Прямоугольный раскрой.** Проектирование карт раскроя для гильотинных ножниц.
- **База данных.** Взаимодействие заданий на раскрой, деталей, листов, раскроев листов.
- **Автоматическое размещение деталей.** Автоматически укладывает детали задания на листы оптимальным образом.
- **Ручное размещение деталей**. Размещение детали на листе явным образом. Динамический контроль перемещения, перемещение до упора, копирование вплотную, выравнивание сторон.
- **Программирование обработки.** Строит траекторию инструмента с подходами и отходами. Обрабатывает мостики и перемычки. Выполняет совмещенную обработку пар деталей.
- **Разделка кромок под сварку.**
- **Использование делового отхода.**
- **Документирование**. Формирует следующие выходные документы: карту раскроя листа, спецификацию раскроя листа, спецификацию задания на раскрой, статистику.



# Листовая штамповка

Программа предназначена для комплексного решения задачи листовой штамповки.

- База данных заданий на раскрой, деталей, листов, раскроев листов.
- Автоматическое размещение деталей задания оптимальным образом.
- Ручное размещение деталей.
- Использование, как типовых инструментов, так и инструментов заданной формы.
- Формирование и использование готовых комплектов инструментов.
- Пробивка одиночных отверстий.
- Вырубка участка. Определение граничных положений за счет выравнивания относительно конца участка или смежных участков. Расстановка перемычек.
- Дробление – разрушение материала в заданной области.
- Автоматическая штамповка.
- Управление порядком обработки (оптимизация) с учетом зоны листа, инструмента, детали.

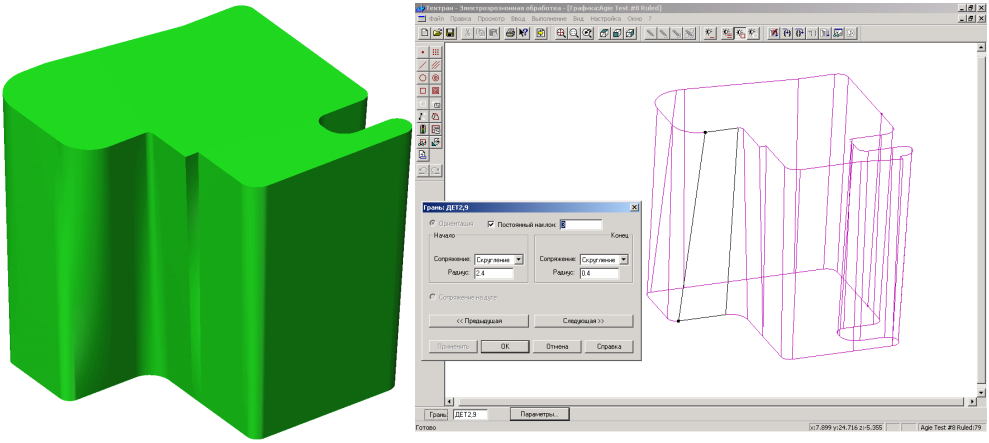



# Электроэрозионная обработка

Программа предназначена для проектирования управляющих программ 2–4-координатной обработки деталей на электроэрозионных станках с ЧПУ.

- Программирование обработки деталей с вертикальной или наклонной боковой поверхностью.
- Построение боковой поверхности детали по базовому контуру.
- Моделирование обработки с учетом параметров конусности геометрической модели.

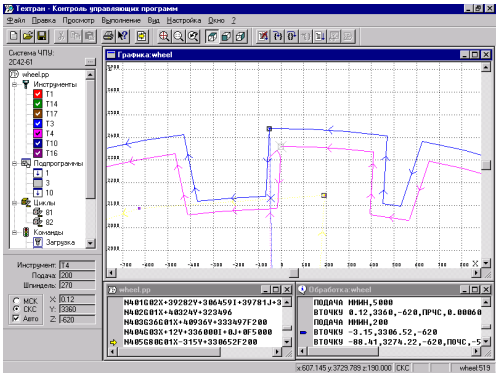
- Постоянный или переменный угол наклона проволоки.
- Произвольные контуры верхнего и нижнего профилей.
- Сопряжение контуров: автоматическое или в заданных точках.
- Контроль допустимого угла наклона проволоки



## Контроль управляющих программ

Программа предназначена для отображения, контроля и редактирования управляющих программ для станков с ЧПУ. Программа может использоваться во взаимодействии с другими программами семейства Техтран и как независимый продукт. При этом не имеет значения, как была получена УП: с использованием системы автоматизированного программирования или вручную.

- Анализ УП с учетом особенностей конкретной системы ЧПУ
- Просмотр объектов, используемых в УП: инструментов, подпрограмм, встроенных циклов, технологических команд.
- Выполнение кадров УП в различных режимах.
- Анимация выполнения.
- Преобразование УП в программу на языке Техтран, возможность использования данных УП в программах семейства Техтран.



Подробная информация о программном продукте на сайте: http://tehtran.com/

Книги издательства «ДМК Пресс» можно заказать в торгово-издательском холдинге «АЛЬЯНС БУКС» наложенным платежом, выслав открытку или письмо по почтовому адресу: 123242, Москва, а/я 20 или по электронному адресу: **orders@alians-kniga.ru**.

При оформлении заказа следует указать адрес (полностью), по которому должны быть высланы книги; фамилию, имя и отчество получателя. Желательно также указать свой телефон и электронный адрес.

Эти книги вы можете заказать и в интернет-магазине: **www.alians-kniga.ru**.

Оптовые закупки: тел. (499) 725-54-09, 725-50-27; электронный адрес books@alians-kniga.ru.

Ловыгин А. А., Теверовский Л. В.

**Современный станок с ЧПУ и CAD/CAM-система**

Главный редактор *Мовчан Д. А.*
dm@dmk-press.ru
Корректор *Синяева Г. И.*
Верстка *Чаннова А. А.*
Дизайн обложки *Мовчан А. Г.*

Подписано в печать ***2012. Формат 70×100 1/16 .
Гарнитура «Петербург». Печать офсетная.
Усл. печ. л. ***. Тираж *00 экз.
№

Веб-сайт издательства: www.dmk-press.ru