

Общество с Ограниченной Ответственностью
ООО "ЕВРАЗИЯ ЛИМИТЕД"
(г. Пермь, Россия)

**СИСТЕМА
АВТОМАТИЗИРОВАННОЙ ПОДГОТОВКИ
УПРАВЛЯЮЩИХ ПРОГРАММ ДЛЯ СТАНКОВ С
ЧПУ**

САПР-ЧПУ/2000 ®

ЯЗЫК ПРОГРАММИРОВАНИЯ

Редакция документации 5.01 (Ноябрь 2000 г.)

Текущая версия документации описывает возможности
системы САПР-ЧПУ/2000 версии не ниже 5.0

©1989-2000, ООО «Евразия Лимитед», Все права защищены.

САПР-ЧПУ® является зарегистрированной торговой маркой коллектива ее разработчиков и защищена Законом "Об авторском праве на программные системы" в государстве Россия на основании свидетельства РОСАПО № 940136 от 28 марта 1994г.

Рисунки встроены студентом гр МТ-45 ЯГТУ Казуовым Ю.В. по заданию доц. Калачева О.Н. исключительно для использования в учебном процессе tms.ystu.ru 2004 г.

Система САПР-ЧПУ/2000 предназначена для автоматизации проектирования управляющих программ для всех типов станков с ЧПУ и обрабатывающих центров на базе ПЭВМ.

Система САПР-ЧПУ/2000 может эксплуатироваться на персональных компьютерах IBM(и совместимых с ними) с процессором Pentium-100 (или Celeron-300) и выше. Операционная среда MS-Windows версий 95, 98, 2000, Me или NT.

Инструкция содержит описание языка, на котором составляется программа обработки деталей на оборудовании с ЧПУ.

В тексте инструкции допускается сокращение названия системы САПР-ЧПУ/2000 до САПР-ЧПУ.

Программа «САПР-ЧПУ»® является зарегистрированной торговой маркой коллектива ее разработчиков и защищена Законом "Об авторском праве на программные системы" в государстве Россия на основании свидетельства РОСАПО № 940136 от 28 марта 1994г.

Проектирование, программная реализация, тестирование и сопровождение САПР-ЧПУ/2000 выполняется в течение 1989-2000гг. авторским коллективом. Правообладателями на систему САПР-ЧПУ/2000 являются:

Трухин Н.М. - ведущий конструктор проекта САПР-ЧПУ;
Филиппович В.И. - директор ООО "Евразия Лимитед";
Филиппович К.В. - начальник группы разработки САПР-ЧПУ;
Воскобойников Ю.С. - научный сотрудник.

Авторы системы имеют монопольное право на развитие и сопровождение системы САПР-ЧПУ, т.е. сохраняют за собой право на внесение изменений и дополнений в программы и инструкции.

Лицензионные права

Вы получаете право использовать систему САПР-ЧПУ/2000 при условии, что в любой момент времени каждая копия программы будет работать только на одном компьютере, а их общее число не превышает максимальное число установок системы САПР-ЧПУ/2000, оговоренное в договоре на его приобретение.

Права собственности

Вы становитесь владельцем только носителей информации, на которых записан программный продукт и документация. Права собственности на сами программы и тексты документации остаются исключительно за автором.

Ограничение на копирование.

Лицензия не дает вам прав на копирование и тиражирование программного продукта. Для обеспечения хранения файлов программного продукта вы имеете право сделать резервные копии так, как это описано в документации. Другие способы копирования являются нарушением условий настоящей лицензии.

Ограничение на использование.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Вы не имеете прав на распространение, модификацию, адаптацию, перевод, декомпиляцию или реассемблирование, снятие защиты от копирования с поставленного продукта, а также включение в состав собственных разработок. Вы не имеете прав на распространение, модификацию, перевод входящих в комплект поставки текстовых материалов и включение их в состав собственных текстов без письменного согласия Авторов.

1. МЕТОД ОПИСАНИЯ ЯЗЫКА _____	8
2. АЛФАВИТ ЯЗЫКА _____	8
3. СТРУКТУРА ИСХОДНОЙ ПРОГРАММЫ НА ЯЗЫКЕ САПР-ЧПУ _____	8
4. КОММЕНТАРИЙ В ТЕКСТЕ ПРОГРАММЫ _____	9
5. ЗАГОЛОВОК ПРОГРАММЫ _____	9
6. СИСТЕМА КООРДИНАТ _____	9
7. РАЗДЕЛ ДАННЫХ _____	10
7.1 ИДЕНТИФИКАТОР ДАННОГО _____	10
7.2. ЧИСЛО _____	11
7.3 УГОЛ _____	11
7.4 АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ _____	12
7.4.1 ОПЕРАНДЫ _____	12
7.4.2 ЗНАКИ ОПЕРАЦИЙ _____	13
7.4.3 ФУНКЦИИ _____	13
7.4.4 КРУГЛЫЕ СКОБКИ _____	14
7.4.5 ПРАВИЛА ЗАПИСИ АРИФМЕТИЧЕСКОГО ВЫРАЖЕНИЯ _____	14
7.5 ОПРЕДЕЛЕНИЕ ЧИСЛОВЫХ ДАННЫХ _____	15
7.6 УГЛЫ В АРИФМЕТИЧЕСКОМ ВЫРАЖЕНИИ _____	15
7.7 ОПРЕДЕЛЕНИЕ ГЕОМЕТРИЧЕСКИХ ЭЛЕМЕНТОВ _____	16
7.7.1 СПОСОБЫ ЗАДАНИЯ ТОЧЕК _____	17
7.7.2 СПОСОБЫ ЗАДАНИЯ ПРЯМЫХ _____	22
7.7.3 СПОСОБЫ ЗАДАНИЯ ОКРУЖНОСТЕЙ _____	26
7.7.4 ДОПОЛНИТЕЛЬНЫЕ СПОСОБЫ ЗАДАНИЯ ОКРУЖНОСТЕЙ _____	31
7.8 ЗАПИСЬ РАЗДЕЛА ДАННЫХ _____	34
8. РАЗДЕЛ ПРОЦЕДУР _____	34
8.1 НАЧАЛО ПРОЦЕДУРЫ, КОНЕЦ ПРОЦЕДУРЫ _____	35
8.2 ЧАСТОТА ВРАЩЕНИЯ ШПИНДЕЛЯ. ПОДАЧА. _____	35
8.3 СКОРОСТЬ РЕЗАНИЯ _____	36
8.4 ОПЕРАТОРЫ, ОПРЕДЕЛЯЮЩИЕ ДВИЖЕНИЕ ИНСТРУМЕНТА _____	37
8.5 ЗАПИСЬ ПРОЦЕДУРЫ _____	38
8.6 АВТОМАТИЧЕСКИЙ РАСЧЕТ ЭКВИДИСТАНТЫ. _____	41
8.6.1 НАЧАЛО АВТОМАТИЧЕСКОГО РАСЧЕТА ЭКВИДИСТАНТЫ _____	42
(ВЫХОД НА ЭКВИДИСТАНТУ). _____	42
8.6.2 КОНЕЦ АВТОМАТИЧЕСКОГО РАСЧЕТА ЭКВИДИСТАНТЫ. _____	45

8.6.3 ИНТЕРПРЕТАЦИЯ ОПЕРАТОРОВ ДОТК, ДОПР, ДОКР ПРИ _____	46
АВТОМАТИЧЕСКОМ РАСЧЕТЕ ЭКВИДИСТАНТЫ. _____	46
8.6.4 НЕПЕРЕСЕЧЕНИЕ ЭЛЕМЕНТОВ ЭКВИДИСТАНТЫ. _____	46
8.7 ИСКАЖЕНИЕ КОНТУРА ПРИ АВТОМАТИЧЕСКОМ РАСЧЕТЕ _____	47
ЭКВИДИСТАНТЫ. _____	47
8.8 ПОДАЧА ПРИ АВТОМАТИЧЕСКОМ РАСЧЕТЕ ЭКВИДИСТАНТЫ _____	48
8.9 КОРРЕКЦИЯ ЭКВИДИСТАНТЫ _____	49
8.9.1. КОРРЕКЦИЯ НА ДЛИНУ И ПОЛОЖЕНИЕ ИНСТРУМЕНТА _____	49
8.9.2 КОРРЕКЦИЯ РАДИУСА ИНСТРУМЕНТА _____	49
8.9.3 СКРУГЛЕНИЕ КОНТУРА ИЛИ ЭКВИДИСТАНТЫ _____	50
8.9.4 ВВОД И СБРОС КОРРЕКЦИИ _____	51
8.10 УЧАСТКИ _____	54
8.10.1 ПРАВИЛА ОФОРМЛЕНИЯ УЧАСТКА _____	54
8.10.2 ОПЕРАТОР ОБУ - ОБРАБОТАТЬ УЧАСТОК _____	55
8.10.3 ПРАВИЛА УПОТРЕБЛЕНИЯ ОПЕРАТОРА ОБУ _____	55
8.10.4 ВЫПОЛНЕНИЕ ПОСЛЕДОВАТЕЛЬНОСТИ УЧАСТКОВ _____	57
8.10.4.1 ОБРАБОТКА ПОСЛЕДОВАТЕЛЬНОСТИ УЧАСТКОВ С _____	57
ПЕРЕНОСОМ, НО БЕЗ ПОВОРОТА _____	57
8.10.4.2 ОБРАБОТКА ПОСЛЕДОВАТЕЛЬНОСТИ УЧАСТКОВ С ПЕРЕНОСОМ И ПОВОРОТОМ УЧАСТКОВ _____	59
НА ПЛОСКОСТИ. _____	59
8.10.4.3 ЗЕРКАЛЬНОЕ ОТОБРАЖЕНИЕ УЧАСТКА _____	61
8.10.5 ЗАМЕНА ЭЛЕМЕНТОВ (ФРАЗ) В УЧАСТКЕ _____	62
8.10.6 ПРАВИЛА ЗАМЕНЫ ЭЛЕМЕНТОВ (ФРАЗ) УЧАСТКА _____	62
8.10.7 ПОЛНЫЙ ФОРМАТ ОПЕРАТОРА ОБУ _____	63
8.11 ОПЕРАТОР "ФАСКА" _____	63
8.12 ФУНКЦИЯ "РАССТОЯНИЕ МЕЖДУ ЭЛЕМЕНТАМИ" _____	64
8.13 Функция "ГЕОМЕТРИЧЕСКАЯ ОШИБКА" _____	64
8.14 ПОВОРОТ СТОЛА, ШПИНДЕЛЬНОЙ БАБКИ И ГОЛОВКИ _____	65
8.14.1 ОПЕРАТОРЫ +ФКР и -ФКР. ДВИЖЕНИЕ ПО ОКРУЖНОСТИ _____	65
ЗА СЧЕТ ПОВОРОТА СТОЛА _____	65
8.14.2 ОПЕРАТОР ВHF-ВОЗВРАТ В 0 ПО КООРДИНАТЕ F _____	66
8.15 ОДНОВРЕМЕННОЕ ДВИЖЕНИЕ БОЛЕЕ ЧЕМ ПО ДВУМ КООРДИНАТАМ, ДОПОЛНИТЕЛЬНЫЕ _____	66
КООРДИНАТЫ А,В,С,Н _____	66
8.16 ОПЕРАТОР СК - ДВИЖЕНИЕ ПО ПРЯМОЙ ИЛИ ОКРУЖНОСТИ _____	67
С ОДНОВРЕМЕННЫМ ДВИЖЕНИЕМ ПО Z И/ ИЛИ ПОВОРОТОМ _____	67
8.17 ОПЕРАТОР W - ВЫДАЧА НЕСКОЛЬКИХ КООРДИНАТ _____	68
В ОДНОМ КАДРЕ УПРАВЛЯЮЩЕЙ ПРОГРАММЫ _____	68
8.18 СМЕНА ПЛОСКОСТИ ОБРАБОТКИ _____	68
8.18.1 ПРАВИЛА ПРОГРАММИРОВАНИЯ ПРИ СМЕНЕ _____	69
ПЛОСКОСТИ ОБРАБОТКИ _____	69

8.19	ТЕХНОЛОГИЧЕСКИЕ КОМАНДЫ УПРАВЛЕНИЯ СТАНКОМ	70
8.19.1	ЗАПИСЬ В РАЗДЕЛЕ ПРОЦЕДУР ИНФОРМАЦИИ НА ЯЗЫКЕ СИСТЕМЫ ЧПУ	71
8.19.2	ПРОПУСК КАДРОВ В УПРАВЛЯЮЩЕЙ ПРОГРАММЕ	72
8.19.3	ГЛАВНЫЙ КАДР	72
8.19.4	ОПЕРАТОР "КАДР"	73
8.19.5	СМЕНА ИНСТРУМЕНТА.	74
8.19.7	ДВА ТИПА РАСЧЕТА X,Y,Z,F,Q ПРИ ФОРМИРОВАНИИ УПРАВЛЯЮЩЕЙ ПРОГРАММЫ	78
8.19.7.1	РАСЧЕТ ПЕРЕМЕЩЕНИЙ В ПРИРАЩЕНИЯХ	78
8.19.7.2	РАСЧЕТ ПЕРЕМЕЩЕНИЙ В АБСОЛЮТНЫХ КООРДИНАТАХ	78
8.19.7.3	СМЕНА ТИПА РАСЧЕТА ПЕРЕМЕЩЕНИЙ В ПРОЦЕДУРЕ	80
8.20	ВЫБОРКА МЕТАЛЛА В ЗОНЕ	81
8.20.1	ПРАВИЛА, ОБЩИЕ ДЛЯ ВСЕХ ВЫБОРОК	81
8.20.2	ВЫБОРКА ПО СХЕМЕ "ПЕТЛЯ"	82
8.20.2.1	ТРЕБОВАНИЯ К УЧАСТКУ.	83
8.20.2.2	СХЕМА ДВИЖЕНИЯ ИНСТРУМЕНТА	83
8.20.3	ВЫБОРКА ПО СХЕМЕ "ЗИГЗАГ"	84
8.20.3.1	ТРЕБОВАНИЯ К УЧАСТКУ	84
8.20.3.2	СХЕМА ДВИЖЕНИЯ ИНСТРУМЕНТА	85
8.20.3.3	ОБРАБОТКА "ГЛУБОКИХ" ЗОН	86
8.20.3.4	ОПИСАНИЕ КОНТУРА ПРЕПЯТСТВИЯ И ПОРЯДОК ВКЛЮЧЕНИЯ ОПИСАНИЯ ПРЕПЯТСТВИЯ В ОПИСАНИЕ КОНТУРА ВЫБОРКИ	87
8.20.4	ВЫБОРКА ПО СХЕМЕ "ЭКВИДИСТАНТА"	87
8.20.4.1	СХЕМА ДВИЖЕНИЯ ИНСТРУМЕНТА В ЭКВИДИСТАНТНОЙ ВЫБОРКЕ	88
8.20.4.2	СХЕМА РАБОЧИХ ПРОХОДОВ.	88
8.21	НАРЕЗАНИЕ РЕЗЬБЫ РЕЗЦОМ	89
8.22	ЦИКЛЫ ОБРАБОТКИ ОТВЕРСТИЙ	90
8.23	ДВИЖЕНИЕ ПО ТОЧЕЧНОЗАДАННОЙ КРИВОЙ	92
8.23.1	ДВИЖЕНИЕ ПО ТОЧЕЧНО ЗАДАННОЙ КРИВОЙ (ЛИНЕЙНО-КРУГОВАЯ ИНТЕРПОЛЯЦИЯ)	92
8.23.2	Способы задания параметров в #ТКР	93
8.23.3	Пример использования #ТКР	94
8.23.4	Плавный заход и выход на/с #ТКР	95
8.24	ОПЕРАТОР ПРИСВАИВАНИЯ	96
8.24.1	Арифметический оператор присваивания	96
8.24.2	Геометрический оператор присваивания	96
8.24.3	ПРИМЕНЕНИЕ ОПЕРАТОРА ПРИСВАИВАНИЯ	96
8.24.4	ДОПОЛНИТЕЛЬНЫЙ ФОРМАТ ОПЕРАТОРОВ ТК,ПР,+КР,-КР	97

8.25 УПРАВЛЕНИЕ ПОСЛЕДОВАТЕЛЬНОСТЬЮ ВЫПОЛНЕНИЯ	97
ОПЕРАТОРОВ РАЗДЕЛА ПРОЦЕДУР	97
8.25.1 МЕТКИ	97
8.25.2 ОПЕРАТОР УСЛОВНОГО ПЕРЕХОДА	97
8.25.3 ОПЕРАТОР БЕЗУСЛОВНОГО ПЕРЕХОДА	99
8.25.4 ОПЕРАТОР ВЫБОРА	99
8.25.5 ЦИКЛЫ	101
8.26 ОПЕРАТОРЫ "ВЫХОД", "ПОВТОР"	104
8.27 ОПЕРАТОР "ДИАЛОГ"	104
8.28 МАССИВ ДАННЫХ, ИНДЕКСИРОВАНИЕ ЭЛЕМЕНТОВ МАССИВА	105
8.29 ИНДЕКСИРОВАНИЕ МЕТКИ В ОПЕРАТОРАХ ПЕРЕХОДА,	106
УЧАСТКА В ОПЕРАТОРЕ ОБУ, ИНСТРУМЕНТА В	106
ОПЕРАТОРЕ СМЕНЫ ИНСТРУМЕНТА	106
8.30 ПОДСТАНОВКА СИМВОЛЬНОЙ СТРОКИ	107
8.30.1 Задание символьной строки	107
8.30.2 Подстановка символьной строки	107
8.30.3. Массив символьных строк, индексирование	107
символьных строк	107
8.30.4 Перевод целых положительных чисел в символьные	108
строки и подстановка	108
8.30.5 Присваивание символьных строк в разделе процедур	108
8.30.6 Сравнение символьных строк	108
8.31 ОПЕРАТОР ПЕЧАТИ	109
8.32 МАКРОПРОЦЕДУРЫ	110
8.32.1 ВЫЗОВ МАКРОПРОЦЕДУРЫ	111
8.32.2 СОСТАВЛЕНИЕ МАКРОПРОЦЕДУРЫ	113
8.32.3 ОБЯЗАТЕЛЬНЫЕ И НЕОБЯЗАТЕЛЬНЫЕ ПАРАМЕТРЫ МАКРО	114
8.32.4 ПРОВЕРКА НА ЗАДАНИЕ ПАРАМЕТРА В ВЫЗОВЕ	114
8.32.5 ГРАВИРОВАНИЕ ТЕКСТА ПО ГОСТ 2930-62	115
8.32.6 СВЕРЛЕНИЕ ГРУППЫ ТОЧЕК, ОБРАЗУЮЩИХ МАТРИЦУ.	117
8.32.7 НАРЕЗАНИЕ РЕЗЬБЫ НА ТОКАРНЫХ СТАНКАХ	119
8.33 ФАЙЛЫ ДАННЫХ НА МАГНИТНЫХ ДИСКАХ	120
8.33.1 ЗАПИСЬ - ЭЛЕМЕНТ ФАЙЛА	121
8.33.2 ИМЯ ФАЙЛА ДАННЫХ	121
8.33.3 ТИПЫ ФАЙЛОВ ДАННЫХ	121
8.33.4 СТРУКТУРА ФАЙЛА В СИМВОЛЬНОМ ФОРМАТЕ	121
8.33.5 ФАЙЛ ВО ВНУТРЕННЕМ ФОРМАТЕ ЭВМ	122
8.33.6 ОПЕРАТОРЫ ОБРАБОТКИ ФАЙЛОВ	122
8.33.6.1 ОТКРЫТЬ ВХОДНОЙ ФАЙЛ ДЛЯ ЧТЕНИЯ	122
8.33.6.2 ЧИТАТЬ ДАННЫЕ ИЗ ФАЙЛА В СИМВОЛЬНОМ ФОРМАТЕ	122

8.33.6.3 ЧИТАТЬ ДАННЫЕ ИЗ ФАЙЛА ВО ВНУТРЕННЕМ ФОРМАТЕ _____	123
8.33.6.5 ЗАПИСАТЬ ДАННЫЕ В ФАЙЛ ВО ВНУТРЕННЕМ ФОРМАТЕ _____	123
8.33.6.6 ЗАКРЫТЬ ВХОДНОЙ/ВЫХОДНОЙ ФАЙЛ _____	124
8.33.6.7 ОТКРЫТЬ СУЩЕСТВУЮЩИЙ ФАЙЛ ДЛЯ ДОПОЛНЕНИЯ _____	124
8.33.6.8 ОБРАБОТКА НЕСКОЛЬКИХ ФАЙЛОВ _____	124
8.34 Оператор "ВЫЗОВ СП" _____	127
8.35 Оператор ВЫЗОВ exe(com) файлов _____	128
8.35.1 Вызов задачи _____	129

1. МЕТОД ОПИСАНИЯ ЯЗЫКА

Исходная программа на языке САПР-ЧПУ состоит из операторов. Оператор - это определение какого-то данного или команда, задающая какое-то действие. Совокупность правил, по которым записываются операторы, называется синтаксисом языка.

Синтаксис языка САПР-ЧПУ описывается в данной инструкции с помощью специальных форматов (шаблонов). Элементами формата могут быть прописные и строчные буквы, символы разделители ",", ";", "/", "(", ")" и квадратные скобки "[", "]". Прописные буквы и символы-разделители переносятся из формата в оператор без изменений, строчные буквы заменяются конкретной информацией пользователя. Квадратные скобки содержат информацию, которая может быть опущена или использована. Если несколько конструкций в формате записаны в нескольких строках друг под другом, то из этих конструкций необходимо выбрать какую-то одну.

Пример: Формат задания точки пересечением или касанием известных прямой и окружности:

ТКномер = [BX] ПРномер, КРномер
 МХ
 БУ
 МУ

Примеры конкретных операторов, соответствующих данному формату:

ТК1=БУПР2,КР10;
 ТК2=ПР3,КР6;

2. АЛФАВИТ ЯЗЫКА

Алфавит языка - это совокупность символов, которые разрешается использовать в исходной программе:

русские и латинские прописные буквы, цифры и специальные символы: + - * : / . , ; = # , . ' () [] !

Между русскими и латинскими буквами одинакового написания (А, В, Е, К, М, Н, О, Р, С, Т) отличия нет, то есть для САПР-ЧПУ они равнозначны. Также равнозначны для САПР-ЧПУ русская буква У и латинская буква Y, апостроф (') и кавычка (").

Строчные буквы (а,б,в, ... а,b,c, ...) можно использовать только в комментариях и в операторе печати(ПЧ), во всех остальных операторах используются только прописные буквы (А,В,В, ... А,В,С...).

3. СТРУКТУРА ИСХОДНОЙ ПРОГРАММЫ НА ЯЗЫКЕ САПР-ЧПУ

заголовок
 раздел данных
 !
 раздел процедур
 !

В заголовке записывается имя исходной программы, а также указывается для какого комплекса "станок - система ЧПУ" составлена программа.

В разделе данных описывается геометрия обрабатываемого контура, могут задаваться технологические параметры, раздел оканчивается символом "!".

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

В разделе процедур описывается процесс обработки детали: движение инструмента, технологические режимы обработки, команды управления станком.

Исходная программа представляет собой последовательность операторов. В заголовке и в разделе данных каждый оператор оканчивается символом ";". В разделе процедур большинство операторов оканчивается также символом ";", некоторые операторы оканчиваются группой символов ");".

В тексте программы для большей читабельности разрешается использовать пробелы и символ "горизонтальная табуляция". Эти символы могут использоваться в любом количестве и в любом месте как между операторами, так и внутри оператора. В одной строке текста можно записывать любое количество операторов или часть одного оператора, специального символа переноса нет, он не нужен.

4. КОММЕНТАРИЙ В ТЕКСТЕ ПРОГРАММЫ

В тексте исходной программы: в заголовке, разделе данных и в разделе процедур между операторами можно использовать комментирующую информацию. Комментарий начинается с символа "*" и оканчивается символом ";". Комментарий не должен содержать символа "!". Число символов в одном комментирующем предложении - не более 2400. Комментарий при трансляции исходной программы игнорируется.

5. ЗАГОЛОВОК ПРОГРАММЫ

Заголовок состоит из двух обязательных операторов:

```
ПРОГРАММА=имя программы;
СТАНОК=имя паспорта;
```

Имя программы - это последовательность латинских букв и цифр, количество символов не больше восьми. Каждому комплексу "станок - система ЧПУ" в САПР-ЧПУ должен соответствовать паспорт - файл на магнитных дисках.

Имя паспорта состоит из латинских букв и цифр, количество символов не больше восьми.

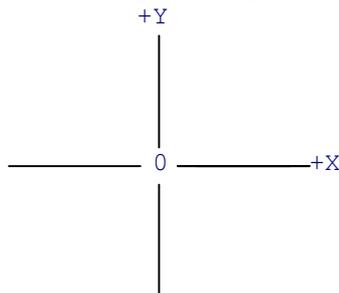
Пример заголовка:

```
ПРОГРАММА=120;
СТАНОК=00050;
```

Операторы ПРОГРАММА и СТАНОК записываются начиная с первой колонки в текстовых файлах, т.е. табуляция недопустима.

6. СИСТЕМА КООРДИНАТ

Описание геометрии детали в САПР-ЧПУ независимо от системы координат станка осуществляется в правосторонней прямоугольной системе координат.



Ось Z в этой системе направлена вверх перпендикулярно плоскости XY. Соотношение между данной системой и системой координат станка устанавливается в паспорте станка (см. инструкцию по составлению паспортов, п. 6.4).

7. РАЗДЕЛ ДАННЫХ

В исходной программе используются следующие данные:

геометрические элементы (точки, прямые, окружности);
числовые данные;
символьные строки.

Раздел данных состоит из операторов. Каждый оператор определяет геометрический элемент, числовое данное или строку символов.

Примеры:

```

ТК1=10,100;      * точка с координатами x=10, y=100;
X0=0;           * число 0;
СТР='ПЕРМЬ';    * строка символов ПЕРМЬ;

```

7.1 ИДЕНТИФИКАТОР ДАННОГО

Геометрическим элементам, числовым данным, символьным строкам в программе присваиваются имена - идентификаторы.

Примеры:

```

ТК10=10,200;    * точка ТК10;
ПР399=ТК10,ТК20; * прямая ПР399;
КР150=0,0,100;  * окружность КР150;
R1=250;         * числовое данное R1;

```

В общем случае идентификатор данного состоит из символического обозначения типа данного и порядкового номера 0, 1, 2,

тип [номер]

тип -последовательность русских и латинских букв, число букв не больше шести.
номер -число от 0 до 399, от одной до трех цифр (номер может отсутствовать).

В САПР-ЧПУ используются следующие обозначения типов:

ТК -точка;
ПР -прямая;
КР -окружность;
X -координата X, смещение по X;
Y -координата Y, смещение по Y;
R -радиус, расстояние;
B -угол в градусной мере;
Z -величина подъема или опускания инструмента;
S -подача в мм/мин;
SN -подача в мм/об;
V -скорость резания в м/мин;
N -частота вращения шпинделя в минуту об/мин;
P -расстояние от центра инструмента до контура;
ИН -диаметр инструмента.

Примечание: в следующих главах будут рассмотрены и другие типы данных.

На месте типа в идентификаторе числового данного или символьной строки можно указать любую последовательность букв, но не больше шести.

Примеры: D0, D1, D2, ШАГ1, ШАГ2.

Данные одного типа в программе не обязательно должны иметь последовательную нумерацию с нуля 0, 1, 2,
Некоторые номера могут быть пропущены.

Пользователь может присваивать числовым данным и символьным строкам идентификаторы, не содержащие номер. Если номера нет, считается, что он равен нулю. Поэтому, например, идентификаторы A и A0 относятся к одному данному.

Примеры неверных идентификаторов:

A/B1 "/" - не буква и не цифра;
1A буквы должны быть в начале идентификатора;
B1C2 буквы и цифры должны быть отделены:
 буквы - в начале, цифры - в конце;
ABSDEFG число букв больше шести;
C400 номер больше 399;
TK идентификатор геометрического элемента должен иметь номер.

7.2. ЧИСЛО

Числа в САПР-ЧПУ записываются в естественном виде, только вместо десятичной запятой используется точка.

Примеры: 25 -1046.78321 0.000659532

7.3 УГОЛ

Угловые величины в градусной мере записываются в исходной программе двумя способами.

Вариант 1:

градусы минуты секунды

градусы -от 0 до 999 (одна - три цифры);
минуты -от 00 до 59 (две цифры);
секунды -от 00 до 59 (две цифры).

Первые незначащие нули в числе можно опустить.

Примеры: 1200000 120 градусов;
 303000 30 градусов, 30 минут;
 102040 10 градусов, 20 минут 40 секунд;
 120 1 минута, 20 секунд;
 30 30 секунд.

Вариант 2:

целое число градусов . доли градуса

Если задается целое число градусов, то ноль после точек можно опустить, но точку опускать нельзя.

Примеры: 30. 30 градусов (без точки - 30 секунд);
 45.5 45 градусов + 0.5 градуса;
 .01 одна сотая доля градуса,
 то есть 36 секунд.

В зависимости от направления отсчета угол может быть положительным или отрицательным. Если угол отсчитывается в направлении, противоположном движению часовой стрелки, то считается положительным, в противном случае - отрицательным. Пример: (см. Рисунок 5). Угол прямой PP_1 с положительным направлением оси X равен +45 градусов или -135 градусов.

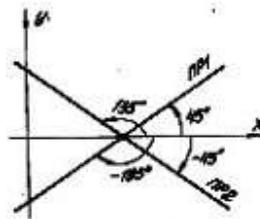


Рисунок 5

Все угловые величины в САПР-ЧПУ переводятся из градусной меры в радианную. Чтобы отличать угловые величины от любых других неугловых величин, для угловых используются специальные идентификаторы:

В номер, F номер, Q номер.

Примеры: B1=100.; данному B1 будет присвоено значение 1.74532925 (100 градусов в радианах). X1=100; данному X1 будет присвоено значение 100.

Угловые величины можно задать и в радианах, если для этих величин использовать идентификаторы, отличные от В номер, F номер, Q номер. Пример: X1=1; A=SIN(X1); * X1 - угол в 1 радиан;

7.4 АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ

Арифметическое выражение представляет собой последовательность операндов, функций, знаков операций и круглых скобок.

7.4.1 ОПЕРАНДЫ

Под операндами понимаются числовые данные, над которыми производятся математические операции.

Пример: A+10 A, 10 -операнды.

Операнды в арифметическом выражении могут быть представлены числами, идентификаторами числовых данных, а также идентификаторами параметров геометрических элементов:

ХТКномер -координата X точки ТКномер;
 УТКномер -координата Y точки ТКномер;
 ХКРномер -координата X центра окружности КРномер;
 УКРномер -координата Y центра окружности КРномер;
 РКРномер -радиус окружности КРномер;

СПРномер - косинус угла F для прямой ПРномер (см. рис. 4),

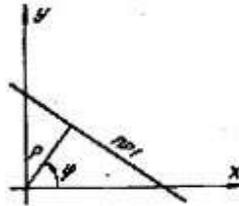


Рисунок 4

F - угол от положительного направления оси X до перпендикуляра к прямой, направление отсчета - против часовой стрелки;

СПРномер - синус угла F для прямой ПРномер;

РПРномер - расстояние от точки $(0,0)$ до прямой ПРномер, если прямая проходит через начало координат, то РПРномер = 0.

Пример: Пусть точка ТК2 имеет следующие координаты: $x=1$, $y=2$. Тогда выражение ХТК2+3 имеет значение 4.

7.4.2 ЗНАКИ ОПЕРАЦИЙ

- + сложение;
- вычитание или знак минус перед операндом, скобкой или функцией;
- * умножение;
- : деление;
- ** возведение в степень.

7.4.3 ФУНКЦИИ

SIGN (x)	SIGN(x) = 1, если $x>1$; SIGN(x) = -1, если $x<1$; SIGN(x) = 0, если $x=0$;
ABS (x)	абсолютное значение x ;
SQRT (x)	корень квадратный из x ;
SIN (x)	синус x ;
COS (x)	косинус x ;
TG (x)	тангенс x ;
ASIN (x)	арксинус x ;
ACOS (x)	арккосинус x ;
ATG (x)	арктангенс x ;
LN (x)	натуральный логарифм x ;
EXP (x)	экспонента e^{**x} ;
ОЦ (x)	округление x до целого по правилам округления;
ОЦБ (x)	округление x до целого в большую сторону;
L (ТКномер, ТКномер)	расстояние между точками;
L (ТКномер, ПРномер)	расстояние от точки до прямой;
L (ПРномер, ПРномер)	расстояние между параллельными прямыми;
L (ТКномер, ЦКРномер)	расстояние от точки до центра окружности;
L (ПРномер, ЦКРномер)	расстояние от центра окружности до прямой;
L (ЦКРномер, ЦКРномер)	расстояние между центрами окружностей;
ERR (ТКномер, ПРномер, КРномер)	определение принадлежности точки прямой или кривой;
ERR (КРномер, КРномер)	определение ошибки пересечения кривых;
ERR (КРномер, ПРномер)	определение ошибки пересечения кривой прямой;
ERR (ПРномер, ПРномер)	определение ошибки пересечения прямой с прямой.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

5. Аргумент функции – произвольное арифметическое выражение, записывается в круглых скобках.

```
SQRT (X1**2+Y1**2)
ASIN (SQRT (2) :2)
```

6. При вычислении значения выражения сначала вычисляется содержимое скобок, значения функций. Внутри скобок и при их отсутствии операции выполняются слева направо в порядке старшинства операций: в первую очередь выполняется возведение в степень; затем выполняется умножение, деление и изменение знака операнда, перед которым записан знак "минус"; наконец выполняется сложение и вычитание.

1. **
2. * : - (изменение знака)
3. + - (вычитание)

Пример: $X1 = -2^{**}2$; Сначала выполняется возведение в степень ($2^{**}2=4$), потом изменение знака, в результате $X1 = -4$.

7.5 ОПРЕДЕЛЕНИЕ ЧИСЛОВЫХ ДАННЫХ

Формат оператора для определения числового данного:

идентификатор = арифметическое выражение ;
--

Слева от символа "=" записывается идентификатор данного, которому присваивается значение арифметического выражения. Арифметическое выражение в частном случае может состоять из одного операнда, например, из одного числа.

Примеры: $A=4$; $X1=A$; $Y1=A-1$; $R1=\text{SQRT}(X1^{**}2+Y1^{**}2)$;

В результате данные получают следующие значения: $X1=4$; $Y1=3$; $R1=5$.

7.6 УГЛЫ В АРИФМЕТИЧЕСКОМ ВЫРАЖЕНИИ

Рассмотрим особенности определения угловых величин:

Вномер
 Fномер = арифметическое выражение;
 Qномер

В таких операторах при вычислении значения арифметического выражения число, записанное в самом начале выражения, будет переводиться из градусной меры в радианную. Остальные числовые константы в выражении считаются обычными числами, не углами.

Примеры:

$B0=30$. 30. -угол 30 градусов;
 $B2=180.:12$; 180. -угол 180 градусов, 12 -обычное число;
 $B3=90.+B1$; 90. -угол 90 градусов,
 сравните: $B3=B1+90$. -ошибка, число 90 не будет переведено в радианы;
 $B4=B1*2$; 2 -обычное число,
 сравните: $B4=2*B1$ -ошибка, число 2 будет переведено в радианы, как угол 2 секунды.

Независимо от типа определяемого данного можно в любом месте арифметического выражения записать угол в градусах, указав его следующим образом:

B/число F/число Q/число

Примеры:

$X1=\text{SIN}(F/45.+2*F0)$; 45 - угол 45 градусов, 2-обычное число
 $Y1=\text{TG}(B/3.)$; 3 - угол 3 градуса
 $Y2=\text{TG}(3)$; 3 - угол 3 радиана

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

7.7 ОПРЕДЕЛЕНИЕ ГЕОМЕТРИЧЕСКИХ ЭЛЕМЕНТОВ

Точки, прямые, окружности задаются с помощью следующих операторов:

ТКномер
 ПРномер = способ задания
 КРномер

Способ задания - информация, однозначно определяющая геометрический элемент. В САПР-ЧПУ имеются 58 способов задания точек, прямых, окружностей. Каждому способу задания соответствует оператор определенной структуры.

Геометрические элементы определяются указанием расположения элемента относительно других элементов и заданием числовой информации типа X, Y, R, B.

Расположение определяемого элемента относительно других элементов указывается с помощью специальных признаков:

/ - признак перпендикулярности;

// - признак параллельности;

I - признак симметрии;

+ - признак внешнего касания окружностей, когда центры окружностей расположены по разные стороны от точки касания (см. рис. 45, окружности КР0, КР1);

- - признак внутреннего касания окружностей, когда центры окружностей расположены по одну сторону от точки касания (см. рис. 45, окружности КР0, КР3);

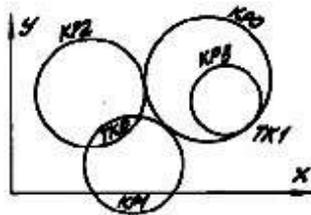


Рисунок 45

Ц - центр окружности или центр другой системы координат;

БХ - больше X (справа);

МХ - меньше X (слева);

БУ - больше Y (сверху);

МУ - меньше Y (снизу).

Примеры:

ПР1=/ПР0,ТК0; прямая ПР1 проходит через точку ТК1 перпендикулярно прямой ПР0 (см. рис. 28);

ПР1>//ПР0,ТК1; прямая ПР1 проходит через точку ТК1 параллельно прямой ПР0 (см. рис. 27);

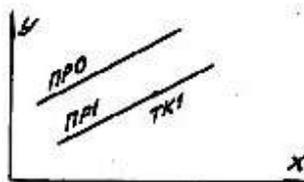


Рисунок 27

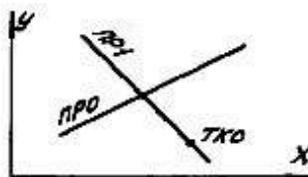


Рисунок 28

$KP1=IKP2, PPO$; окружность $KP1$ симметрична $KP2$ относительно прямой PPO (см. рис. 54).

Числовые данные X, Y, R, B задаются при определении точек, прямых и окружностей тремя способами:

1. Xномер Yномер Rномер Bномер

Естественно, что соответствующие числовые данные должны быть определены в разделе данных.

2. X/арифметическое выражение
Y/арифметическое выражение
R/арифметическое выражение
B/арифметическое выражение

3. Только для некоторых способов задания разрешается указывать X, Y, R просто числами.

Примеры: $TK1=X1, Y1$; $TK2=X/A:2, Y/UTK0+B$; $TK3=10, 20$;

При задании угла арифметическим выражением помните, что число в выражении считается углом в градусах, если оно записано самым первым в выражении, или перед числом записаны символы "B/" или "F/" или "Q/" (см. п. 7.6).

Примеры:

$TK1=R1, B/100.+B1$; правильно, число 100 будет переведено в радианы;
 $TK1=R1, B/B1+100.$; неправильно, число 100 не будет переведено в радианы;
 $TK2=R2, B/B1*2$; правильно, 2 - обычное число;
 $TK2=R2, B/2*B1$; неправильно, число 2 будет переведено в радианы (2 секунды);
 $TK3=R1, B/(B1+B/45.)*2$; правильно, число 45 будет переведено в радианы;
 $TK3=R1, B/(B1+45.)*2$; неправильно, число 45 не будет переведено в радианы.

Все способы задания точек, прямых, окружностей будут описываться с помощью форматов. В этих форматах строчные латинские буквы i, j, k, n соответствуют номерам 0, 1, 2, ..., 399. Сокращение "а.в." расшифровывается как "арифметическое выражение".

7.7.1 СПОСОБЫ ЗАДАНИЯ ТОЧЕК

1. Точка задана координатами X, Y .

$$TKn = Xi \quad , \quad Yj \quad ; \\ X/a.v. \quad Y/a.v. \\ \text{число} \quad \text{число}$$

Пример: (см. рис. 6) $TK1=4, 6$;

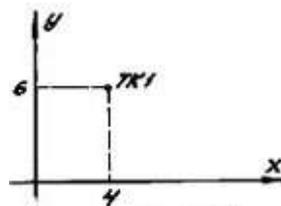


Рисунок 6

2. Точка задана пересечением двух прямых.

$$TKn = PRi \quad , \quad PRj ;$$

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Пример: (см. рис. 7) $TK1=PR1, PR2$;

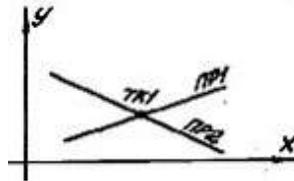


Рисунок 7

3. Точка задана пересечением или касанием прямой и окружности.

$$TKn = \begin{matrix} BX & PRi & , & KPj; \\ BU \\ MX \\ MU \end{matrix}$$

Сравнивая координаты двух точек пересечения, определяется признак BX, BU, MX, MU. При касании признак не указывается.

Примеры: (см. рис. 8) $TK1=MXPR1, KP0$; $TK2=BUPR1, KP0$; $TK3=PR2, KP0$;

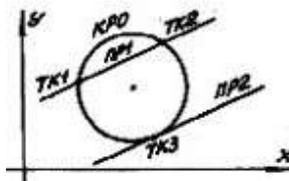


Рисунок 8

4. Точка задана пересечением или касанием двух окружностей.

$$TKn = \begin{matrix} BX & KPi & , & KPj; \\ BU \\ MX \\ MU \end{matrix}$$

Сравнивая координаты двух точек пересечения, определяется признак BX, BU, MX, MU. При касании признак не указывается.

Примеры: (см. рис. 9) $TK1=KP1, KP2$; $TK2=MUKP2, KP3$; $TK3=BUKP2, KP3$;

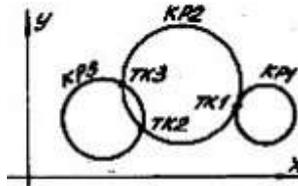


Рисунок 9

5. Точка задана смещением по X, Y от другой точки.

$$TKn = \begin{matrix} TKi & , & Xj & , & Yk & ; \\ & & X/a.v. & & Y/a.v. & \\ & & \text{число} & & \text{число} & \end{matrix}$$

Пример: (см. рис. 10) $TK1=TK0,10,5;$

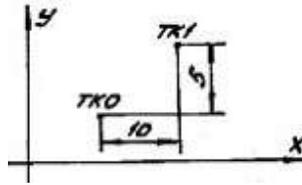


Рисунок 10

6. Точка задана как центр окружности.

$$TKn = ЦКРi;$$

Пример: (см. рис. 11) $TK0=ЦКР0;$

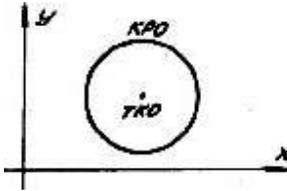


Рисунок 11

7. Точка симметрична другой точке относительно прямой.

$$TKn = I TKi , ПРj$$

Пример: (см. рис. 12) $TK1=ITK2,ПР0;$

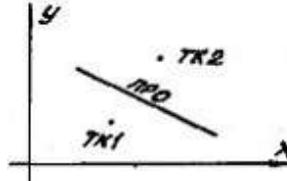


Рисунок 12

8. Точка принадлежит прямой и находится от другой точки этой прямой на заданном расстоянии.

$$TKn = BX TKi , ПРj , Rk ;$$

$$БУ \quad R/a.v.$$

$$MX$$

$$МУ$$

Сравнивая координаты двух возможных точек, определяется признак БХ, БУ, МХ, МУ.

Примеры: (см. рис. 13) $TK1=БУТК0,ПР0,R/10;$ $TK2=МХТК0,ПР0,R/5;$

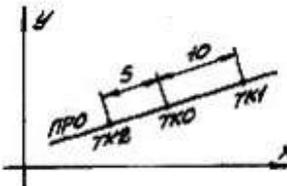


Рисунок 13

9. Точка задана полярными координатами: радиусом-вектором и углом радиуса-вектора с осью X.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

$$TK_n = B_i \quad , \quad R_j \quad ; \\ B/a.v. \quad R/a.v.$$

Угол отсчитывается от положительного направления оси X.

Примеры: (см. рис. 14) $TK_1=B/30.,R/10$; $TK_2=B/-20.,R/5$;

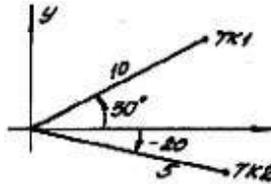


Рисунок 14

10. Точка задана полярными координатами в "местной" системе координат.

$$TK_n = \text{цTK}_i \quad , \quad B_j \quad , \quad R_k \quad ; \\ B/a.v. \quad R/a.v.$$

Примеры: (см. рис. 15) $TK_2=\text{цTK}_1,B/60.,R/20$; $TK_3=\text{цTK}_1,B/-30.,R/10$;

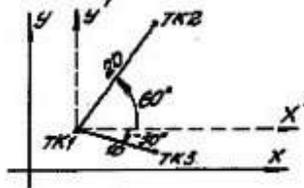


Рисунок 15

11. Точка задана поворотом радиуса-вектора.

$$TK_n = TK_i \quad , \quad B_j \quad ; \\ B/a.v.$$

Примеры: (см. рис. 16) $TK_1=TK_0,B/20.$; $TK_2=TK_0,B/-20.$;

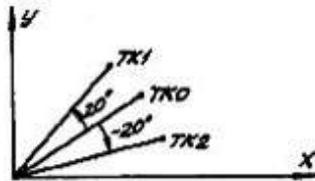


Рисунок 16

12. Точка задана поворотом радиуса-вектора в "местной" системе координат.

$$TK_n = \text{цTK}_i \quad , \quad TK_j \quad , \quad B_k \quad ; \\ B/a.v.$$

Примеры: (см. рис. 17) $TK_1=\text{цTK}_0,B/30.$; $TK_2=\text{цTK}_3,TK_0,B/-60.$;

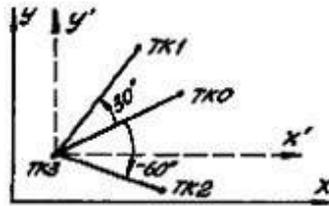


Рисунок 17

13. Точка задана увеличением или уменьшением радиуса-вектора.

$$TK_n = TK_i, R_j ; \\ R/a.v.$$

Примеры: (см. рис. 18) $TK_1 = TK_0, R/10$; $TK_2 = TK_0, R/-5$;

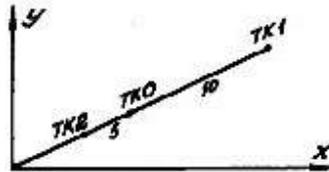


Рисунок 18

14. Точка задана увеличением или уменьшением радиуса-вектора в "местной" системе координат.

$$TK_n = \zeta TK_i, TK_j, R_k ; \\ R/a.v.$$

Примеры: (см. рис. 19) $TK_1 = \zeta TK_3, TK_0, R/2$; $TK_2 = \zeta TK_3, TK_0, R/-4$;

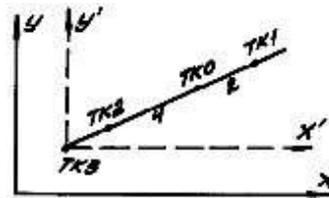


Рисунок 19

15. Точка симметрична другой точке относительно заданной точки.

$$TK_n = I TK_i, TK_j ;$$

TK_j - центр симметрии;

Пример: (см. рис. 20) $TK_1 = I TK_2, TK_0$;

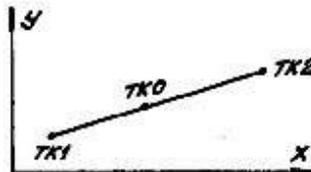


Рисунок 20

16. Точка симметрична другой точке относительно прямой.

$$TK_n = I TK_i, PR_j ;$$

Пример: $TK_1 = I TK_2, PR_0$;

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

7.7.2 СПОСОБЫ ЗАДАНИЯ ПРЯМЫХ

1. Прямая отсекает заданные отрезки на осях координат.

$$\text{ПРn} = X_i \quad , \quad Y_k \quad ; \\ X/a.v. \quad Y/a.v.$$

Примеры: (см. рис. 21) ПР1=X/150, Y/150; ПР0=X/15, Y/-5;

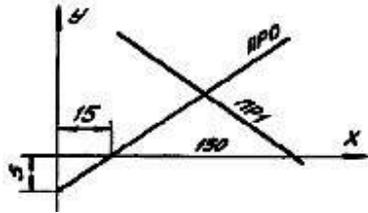


Рисунок 21

2. Прямая отсекает заданные отрезки на осях местной системы координат.

$$\text{ПРn} = ЦТК_i \quad , \quad X_i \quad , \quad Y_j \quad ; \\ X/a.v. \quad Y/a.v.$$

Пример: (см. рис. 22) ПР0=ЦТК1, X/20, Y/15;

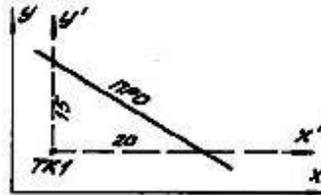


Рисунок 22

3. Прямая проходит через две точки.

$$\text{ПРn} = ТК_i \quad , \quad ТК_j \quad ;$$

Пример: (см. рис. 23) ПР1=ТК1, ТК2;

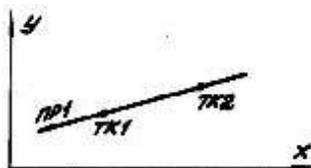


Рисунок 23

4. Прямая параллельна оси Y.

$$\text{ПРn} = X_i \quad ; \\ X/a.v.$$

Пример: (см. рис. 24) ПР1=X/25;

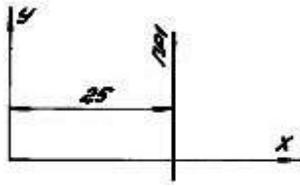


Рисунок 24

5. Прямая параллельна оси X.

$$\text{ПРn} = Y_i \quad ; \\ Y/a.v.$$

Пример: (см. рис. 25) $\text{ПР1} = Y/10$;

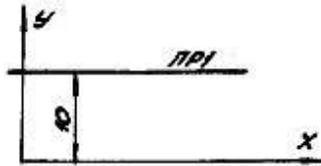


Рисунок 25

6. Прямая параллельна другой прямой и отстоит от нее на заданном расстоянии.

$$\text{ПРn} = \text{BX} // \text{ПРi}, \text{Rj} \quad ; \\ \text{MX} \quad \text{R/a.v.} \\ \text{BY} \\ \text{MY}$$

BX -ПРn справа от ПРi,
 MX -ПРn слева от ПРi,
 BY -ПРn сверху от ПРi,
 MY -ПРn снизу от ПРi.

Примеры: (см. рис. 26) $\text{ПР1} = \text{BY} // \text{ПР0}, \text{R}/5$; $\text{ПР2} = \text{BX} // \text{ПР0}, \text{R}/10$;

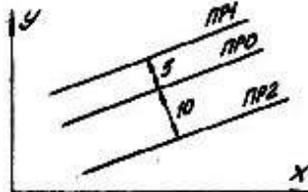


Рисунок 26

7. Прямая проходит через точку параллельно другой прямой.

$$\text{ПРn} = // \text{ПРi}, \text{ТКj};$$

Пример: (см. рис. 27) $\text{ПР1} = // \text{ПР0}, \text{ТК1}$;

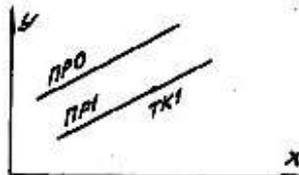


Рисунок 27

8. Прямая проходит через точку перпендикулярно заданной прямой.

$$\text{ПРn} = / \text{ПРi}, \text{ТКj};$$

Пример: (см. рис. 28) $\text{ПР1} = / \text{ПР0}, \text{ТК0};$

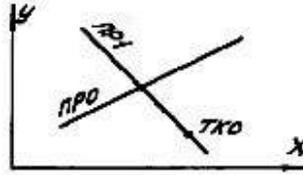


Рисунок 28

9. Прямая проходит через точку, образуя заданный угол с осью X.

$$\text{ПРn} = \text{ТКи}, \text{Бj};$$

Б/а.в.

Угол отсчитывается от положительного направления оси X.

Примеры: (см. рис. 29) $\text{ПР1} = \text{ТК0}, \text{Б}/45.;$ $\text{ПР2} = \text{ТК0}, \text{Б}/-45.;$

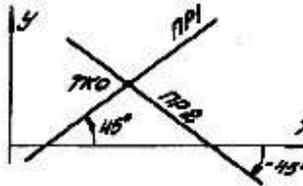


Рисунок 29

10. Прямая проходит через точку, образуя заданный угол с ругой прямой.

$$\text{ПРn} = \text{ТКи}, \text{ПРj}, \text{Бк};$$

Б/а.в.

Угол отсчитывается от заданной прямой ПРj к определяемой ПРn .

Примеры: (см. рис. 30) $\text{ПР2} = \text{ТК1}, \text{ПР1}, \text{Б}/60.;$ $\text{ПР2} = \text{ТК1}, \text{ПР1}, \text{Б}/-120.;$



Рисунок 30

11. Прямая касается окружности, образуя заданный угол с осью X.

$$\text{ПРn} = \text{БХ КРi}, \text{Бj};$$

МХ
БУ
МУ
Б/а.в.

Угол отсчитывается от положительного направления оси X. Сравнивая координаты двух точек касания (возможны две касательные), определяется признак БХ, МХ, БУ, МУ.

Примеры: (см. рис. 31) $\text{ПР1} = \text{БХ КР0}, \text{Б}/135.;$ $\text{ПР2} = \text{МХ КР0}, \text{Б}/-45.;$

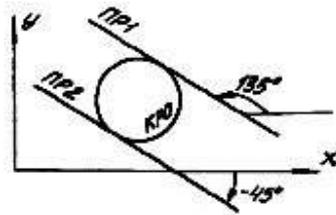


Рисунок 31

12. Прямая касается окружности, образуя заданный угол с другой прямой.

БУ
$PP_n = BX \text{ } KPi, Ppj, Bk$;
МХ
МУ
В/а.в.

Угол отсчитывается от заданной прямой PP_j к определяемой PP_n . Сравнивая координаты двух точек касания (возможны две касательные), определяется признак БХ, МХ, БУ, МУ.

Примеры: (см. рис. 32) $PP_1 = MXKp0, Pp0, B/50.$; $PP_2 = BXKp0, Pp0, B/-130.$;

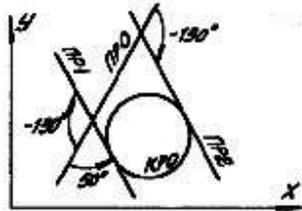


Рисунок 32

13. Прямая проходит через точку и касается окружности.

$PP_n = BX \text{ } KPi, TKj$;
МХ
БУ
МУ

Сравнивая координаты двух точек касания (возможны две касательные), определяется признак БХ, БУ, МХ, МУ. Если точка TK_j является точкой касания, признак не указывается.

Примеры: (см. рис. 33) $PP_1 = BUKp0, TK1$; $PP_2 = MUKp0, TK1$; $PP_3 = Kp0, TK2$;

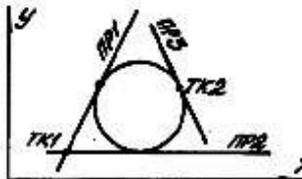


Рисунок 33

14. Прямая касается двух окружностей.

$PP_n = BX \text{ } KPi, BX \text{ } Kpj$;	
МХ	МХ
БУ	БУ
МУ	МУ

Для определения признаков ВХ, ВУ, МХ, МУ фиксируем прямую в точке касания с первой окружностью и определяем признак для второй окружности (см. способ 13), затем фиксируем прямую в точке касания со второй окружностью и определяем признак для первой окружности.

Примеры: (см. рис. 34) $ПР1=МУКР1, МУКР2$; $ПР3=МУКР1, ВУКР2$;

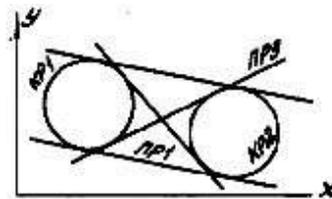


Рисунок 34

15. Прямая симметрична другой прямой относительно заданной прямой.

$$ПРn = I ПРi , ПРj ;$$

$ПРj$ - ось симметрии.

Пример: (см. рис. 35) $ПР2=IПР1, ПР0$;

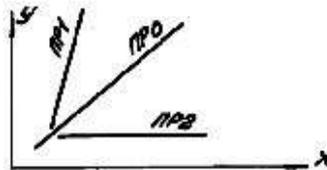


Рисунок 35

16. Прямая симметрична другой прямой относительно точки.

$$ПРn = I ПРi , ТКj ;$$

Пример: (см. рис. 36) $ПР1=IПР2, ТК1$;

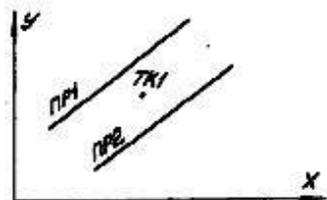


Рисунок 36

7.7. 3 СПОСОБЫ ЗАДАНИЯ ОКРУЖНОСТЕЙ

1. Окружность задана координатами центра и радиусом.

$$КРn = Xi , Yj , Rk ;$$

X/а.в.	Y/а.в.	R/а.в.
число	число	число

Пример: (см. рис. 37) $КР0=20, 40, 10$;

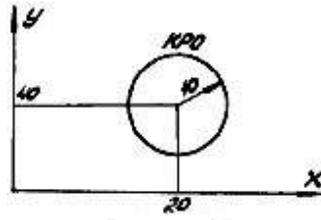


Рисунок 37

2. Окружность задана центром и радиусом.

$$K_{Pn} = ЦТК_i, R_j ;$$

R/а.в.

Пример: (см. рис. 38) $KP0=ЦТК0, R/10;$

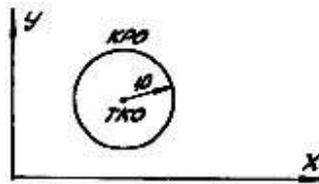


Рисунок 38

3. Окружность задана смещением центра другой окружности.

$$K_{Pn} = K_{Pi}, X_j, Y_k ;$$

X/а.в. Y/а.в.
число число

Пример: (см. рис. 39) $KP1=KP0, 30, 10;$

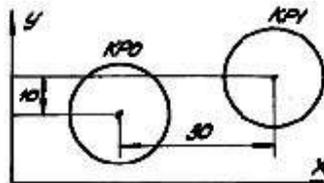


Рисунок 39

4. Окружность задана центром и точкой на окружности.

$$K_{Pn} = ЦТК_i, ТК_j ;$$

Пример: (см. рис. 40) $KP0=ЦТК0, ТК1;$

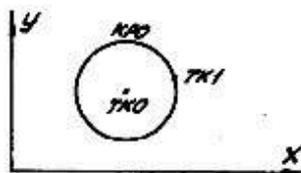


Рисунок 40

5. Окружность с заданным центром касается прямой.

$$K_{Pn} = ЦТК_i, ПР_j ;$$

Пример: (см. рис. 41) $KP0 = \text{ЦТК}0, \text{ПР}0;$

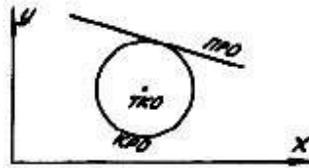


Рисунок 41

6. Окружность данного радиуса проходит через точку и касается прямой.

$$KP_n = \text{БХ } \text{ПР}_i, \text{ТК}_j, R_k; \\ \text{МХ} \quad \text{R/a.v.} \\ \text{БУ} \\ \text{МУ}$$

Сравнивая координаты центров двух возможных окружностей, определяется признак БХ, МХ, БУ, МУ. Признак не указывается, если расстояние от точки до прямой равно диаметру окружности.

Примеры: (см. рис. 42, 43)

$$KP1 = \text{БХПР}1, \text{ТК}1, R/15; \quad KP0 = \text{МХПР}1, \text{ТК}1, R/15; \\ KP2 = \text{ПР}2, \text{ТК}2, R/5;$$

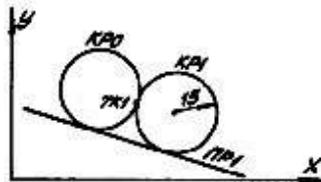


Рисунок 42

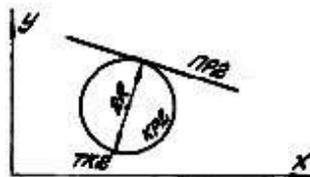


Рисунок 43

7. Окружность с заданным центром касается другой окружности.

$$KP_n = \text{ЦТК}_i, + KP_j; \\ -$$

Правила определения знака "+", "-" перед KP_j :

1. Если расстояние между центрами окружностей больше радиуса KP_j , то при внешнем касании ставится "+" и при внутреннем касании "-".

2. Если расстояние между центрами окружностей меньше радиуса KP_j , то из двух возможных окружностей "+" ставится у окружности с большим радиусом, "-" у окружности с меньшим радиусом.

3. Если центр KP_n расположен на окружности KP_j , ставится знак "+".

Примеры: (см. рис. 44) $KP0 = \text{ЦТК}0, +KP1;$ $KP2 = \text{ЦТК}1, +KP1;$

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

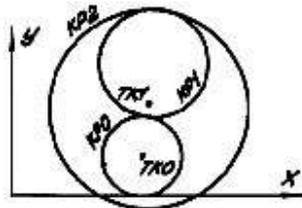


Рисунок 44

8. Окружность заданного радиуса проходит через точку и касается другой окружности.

$$K_{Pn} = BX + K_{Pi}, TK_j, R_k ;$$

$$MX - \quad \quad \quad R/a.v.$$

$$BY$$

$$MY$$

Сначала определяется знак касания "+" или "-". Затем, сравнивая координаты центров двух возможных окружностей, определяется признак BX, MX, BY, MY. Если точка TK_j - точка касания, признак не указывается.

Примеры: (см. рис. 45)

$K_{P1} = MY + K_{P0}, TK_0, R/10;$ $K_{P2} = BY + K_{P0}, TK_0, R/10;$

$K_{P3} = -K_{P0}, TK_1, R/5;$

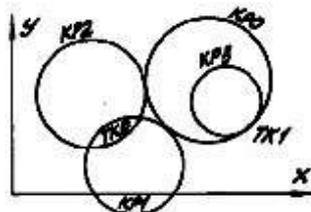


Рисунок 45

9. Окружность заданного радиуса проходит через две точки.

$$K_{Pi} = BX TK_i, TK_j, R_k ;$$

$$MX \quad \quad \quad R/a.v.$$

$$BY$$

$$MY$$

Сравнивая координаты центров двух возможных окружностей, определяется признак BX, MX, BY, MY. Если расстояние между точками равно диаметру определяемой окружности, признак не указывается.

Примеры: (см. рис. 46)

$K_{P1} = MX TK_0, TK_1, R/15;$ $K_{P2} = BX TK_0, TK_1, R/15;$

$K_{P1} = TK_2, TK_1, R/15;$

10. Окружность заданного радиуса касается двух других окружностей.

$$K_{Pn} = BX + K_{Pi}, + K_{Pj}, R_k ;$$

$$MX - \quad - \quad \quad R/a.v.$$

$$BY$$

$$MY$$

Сначала определяются знаки касания "+" или "-" окружности K_{Pn} с окружностями K_{Pi}, K_{Pj}. Затем, сравнивая координаты центров двух возможных

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

окружностей, определяется признак БХ, МХ, БУ, МУ. Если центры всех окружностей находятся на одной прямой, признак не указывается.

Примеры: (см. рис. 47) $KP0=BU+KP2, -KP1, R2;$
 $KP3=MU+KP2, -KP1, R2;$ $KP5=MU+KP2, +KP4, R3;$

11. Окружность заданного радиуса касается прямой и окружности.

$KP_n = BX + KP_i$, БХ	PR_j	, R_k	;
МХ	-	МХ		$R/a.v.$
БУ		БУ		
МУ		МУ		

Сначала определяется знак касания "+" или "-". Затем, сравнивая координаты центра определяемой окружности с координатами точки касания KP_n с PR_j , определяется признак БХ, МХ, БУ, МУ для PR_j . После этого, сравнивая координаты центров двух возможных окружностей, определяется признак БХ, МХ, БУ, МУ для KP_i . Признак БХ, МХ, БУ, МУ перед PR_j не указывается, если PR_j не пересекается с KP_i . Признак БХ, МХ, БУ, МУ перед KP_i не указывается, если центры окружностей лежат на одной прямой, перпендикулярной PR_j .

Примеры: (см. рис. 48, 49) $KP0=BU-KP1, МУPR0, R/5;$
 $KP2=BU+KP1, БУPR0, R/5;$ $KP6=BX+KP1, БУPR0, R/5;$
 $KP3=BU+KP4, PR1, R/5;$ $KP5=-KP4, БУPR2, R/2;$

12. Окружность заданного радиуса касается двух пересекающихся прямых.

$KP_n = BX$	PR_i	, БХ	PR_j	, R_k	;
МХ		МХ		$R/a.v.$	
БУ		БУ			
МУ		МУ			

Сравнивая координаты центра определяемой окружности с координатами точки касания (для каждой прямой), определяется признак БХ, МХ, БУ, МУ.

Примеры: (см. рис. 50) $KP1=МУPR1, МУPR2, R/5;$ $KP2=МУPR1, БУPR2, R/5;$

13. Окружность касается трех прямых.

$KP_n = BX$	PR_i	, БХ	PR_j	, БХ	PR_k	;
МХ		МХ		МХ		
БУ		БУ		БУ		
МУ		МУ		МУ		

Признак БХ, МХ, БУ, МУ для каждой прямой определяется из сравнения координат центра определяемой окружности с координатами точки касания.

Пример: (см. рис. 51) $KP0=BXPR1, БУPR3, МХPR2;$

14. Окружность проходит через три точки.

$KP_n = TK_i$, TK_j	, TK_k	;
---------------	----------	----------	---

Пример: (см. рис. 52) $KP0=TK1, TK2, TK3;$

15. Окружность симметрична другой окружности относительно точки.

$KP_n = I$	KP_i	, TK_j	;
------------	--------	----------	---

Пример: (см. рис. 53) $KP1=IKP2,TK0;$

16. Окружность симметрична другой окружности относительно прямой.

$$KP_n = I KP_i, PR_j;$$

Пример: (см. рис. 54) $KP1=IKP2,PR0;$

17. Центр окружности заданного радиуса совпадает с центром другой окружности (концентрические окружности).

$$KP_n = KP_i, R_j; \\ R/a.v.$$

7.7.4 ДОПОЛНИТЕЛЬНЫЕ СПОСОБЫ ЗАДАНИЯ ОКРУЖНОСТЕЙ

Дополнительные способы задания окружностей реализованы только в системе Препроцессора САПР-ЧПУ/4.1 и значительно расширяют средства кодирования геометрии детали.

1. Окружность заданного радиуса проходит через точку, центр лежит на заданной окружности.

$$KP_n = BX KP_i, TK_j, R_k; \\ MX \quad R/a.v.$$

Пример: (см. рис. 135) $KP2=BXKP1,TK1,R/5;$

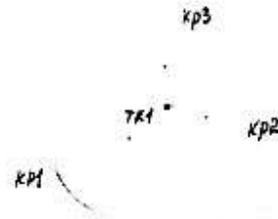


Рисунок 135

2. Окружность касается трех заданных окружностей.

$$KP_n = BU + KP_i, - KP_j, + KP_k;$$

Пример: (см. рис. 137) $KP4=BX-KP1,-KP2,-KP3;$



Рисунок 137

Правило определения знаков и признаков. При внешнем касании ставится "+", при внутреннем касании "-". Сначала определяется знак касания "+" или "-". Затем, сравнивая координаты центров двух возможных окружностей, определяется признак BX, MX, BU, MU.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

3. Окружность, проходящая через точку, касательно двух заданных окружностей.

$$K_{Pn} = BУ + K_{Pi} , + K_{Pj} , TKk;$$

Пример: (см. рис. 138) $K_{P4} = BХ + K_{P2} , -K_{P3} , TK1;$



Рисунок 138

4. Окружность, проходящая через две точки, касательно заданной окружности.

$$K_{Pn} = BХ + K_{Pi} , TKj , TKk;$$

Пример: (см. рис. 139) $K_{P2} = BХ - K_{P1} , TK1 , TK2;$



Рисунок 139

5. Окружность касается двух заданных окружностей и прямой.

$$K_{Pn} = BХ - K_{Pi} , + K_{Pj} , BХ ПРk;$$

Пример: (см. рис. 140) $K_{P3} = MХ + K_{P1} , + K_{P2} , BУ ПР1;$

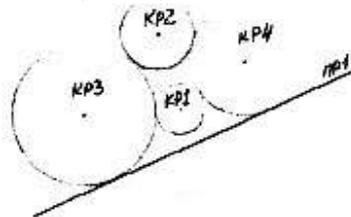


Рисунок 140

Правило определения знака { "+", "-" } перед K_{Pi} : при внешнем касании ставится "+", при внутреннем касании "-". Правило построения окружности. Сначала определяется знак касания "+" или "-". Затем, с помощью модификаторов { BХ, MХ, BУ, MУ } задаваемых перед ПРk определяется половина плоскости в которой лежит центр строящейся окружности, сравнивая координаты центров двух возможных окружностей, определяется признак { BХ, MХ, BУ, MУ } перед K_{Pi} .

6. Окружность, проходящая через точку, касается заданных окружности и прямой.

$$K_{Pn} = BХ + K_{Pi} , ПРj , TKk;$$

Пример: (см. рис. 141) $K_{P3} = MХ + K_{P1} , ПР1 , TK0;$

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

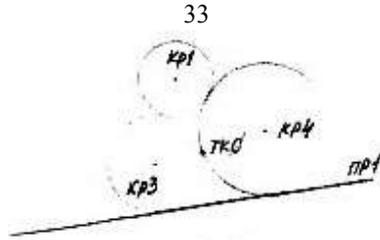


Рисунок 141

7. Окружность, проходящая через две точки, касается заданной прямой.

$$Kp_n = BX \text{ Pp}_i, TK_j, TK_k;$$

Пример: (см. рис. 136) $Kp_2 = MX \text{ Pp}_1, TK_1, TK_2;$

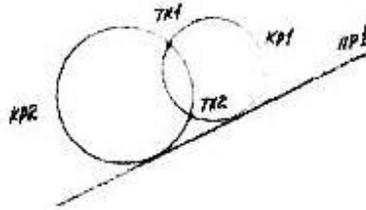


Рисунок 136

Сравнивая координаты центров двух возможных окружностей, определяется признак BX, MX, BU, МУ.

8. Окружность касается заданной окружности и двух прямых.

$$Kp_n = BX + Kp_i, BX \text{ Pp}_j, BX \text{ Pp}_k;$$

Пример: (см. рис. 142) $Kp_3 = BX + Kp_1, BU \text{ Pp}_1, МУ \text{ Pp}_4;$

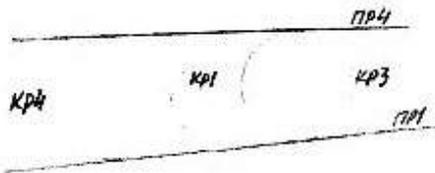


Рисунок 142

Использовать правила определения знаков и признаков из п.5 настоящего раздела.

9. Окружность проходящая через точку и касающаяся двух прямых.

$$Kp_n = BX \text{ Pp}_i, \text{ Pp}_j, TK_k;$$

Пример: (см. рис. 143) $Kp_3 = BU \text{ Pp}_1, МУ \text{ Pp}_4, TK_1;$

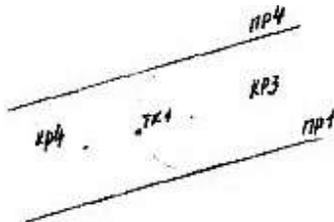


Рисунок 143

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Сравнивая координаты центров двух возможных окружностей определяется признак БХ, МХ, ВУ, МУ.

7.8 ЗАПИСЬ РАЗДЕЛА ДАННЫХ

1. Выбирается начало системы координат.
2. Геометрическим элементам на чертеже присваиваются идентификаторы: ТКномер, ПРномер, КРномер.
3. Размерам на чертеже рекомендуется также присвоить идентификаторы.
4. Используйте арифметические выражения, не производите сами вычисления на бумаге или калькуляторе.
5. Если геометрический элемент нельзя определить сразу (нет соответствующего способа задания), то необходимо задать один или несколько дополнительных элементов. Для их задания, возможно, нужны будут еще дополнительные элементы. Вложенность задания, когда один элемент определяется через другой, тот еще через другой, не должна быть больше 20.
6. Задание геометрического элемента через другие элементы предпочтительнее явного задания через X, Y, R, B. Особенно это важно, если значения X, Y, R, B задаются округленными.

Пример: КР1=X1, Y1, R1; КР2=X2, Y2, R2; ТК1=КР1, КР2; Здесь точка ТК1 задана как точка касания окружностей КР1, КР2. Если какие-то значения (X1, Y1, R1, X2, Y2, R2) заданы округленными, то касания окружностей может не получиться. Предпочтительнее, например, такая запись:

КР1=X1, Y1, R1; ТК2=X2, Y2; КР2=ЦТК2, +КР1;
ТК1=КР1, КР2;

7. Последовательность слов в операторе справа от "=" может быть любой (слово - это последовательность символов, ограниченная символом "," или ";"). Признак БХ, МХ, ВУ, МУ, если он один в операторе, можно записать в любом слове перед идентификатором точки, прямой или окружности. Например, следующие операторы эквивалентны:

ТК1=БХПР1, КР2; ТК1=БХКР2, ПР1; ТК1=ПР1, БХКР2; ТК1=КР2, БХКР2;

8. Для некоторых способов задания указаны ситуации, когда признак БХ, МХ, ВУ, МУ не нужен. Если он все-таки будет указан, то это не будет ошибкой.
9. Один и тот же элемент нельзя определять дважды.

10. Число операторов в разделе - не больше 600. Операторы в разделе записываются в любом порядке.

11. Число символов в разделе - любое.

8. РАЗДЕЛ ПРОЦЕДУР

Раздел процедур состоит из одной или нескольких процедур. Каждой процедуре соответствует одна управляющая программа к станку. Число процедур - не больше двадцати пяти.

Операторы, из которых состоит процедура, можно разделить на следующие группы:

1. Операторы начала и конца управляющей программы.
2. Операторы, задающие подачу, частоту вращения шпинделя, скорость резания.
3. Операторы, задающие движение инструмента.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

4. Операторы автоматического расчета эквидистанты.
5. Операторы коррекции эквидистанты.
6. Операторы технологических циклов (сверление, нарезание резьбы и т.д.).
7. Операторы автоматической выборки металла в зоне, ограниченной заданным контуром.
8. Технологические операторы, управляющие работой различных устройств станка.
9. Операторы начала и конца участка.
10. Оператор обработки участка.
11. Операторы присваивания, условного и безусловного перехода, выбора, циклов.
12. Оператор вызова макропроцедуры.
13. Операторы обработки файлов.
14. Оператор вызова станочных подпрограмм и автоциклов.
15. Оператор вызова пользовательских задач.

8.1 НАЧАЛО ПРОЦЕДУРЫ, КОНЕЦ ПРОЦЕДУРЫ

Каждой процедуре присваивается номер от 0 до 24. Процедура начинается с фразы

НПномер;

и заканчивается фразой

КПномер;

Пример:

```

ПРОГРАММА=CNF;
СТАНОК=090;
ТКО=0,0;
...
!
НП0; ... КП0;
НП1; ... КП1;
... ..
НП24; ... КП24;!

```

В одной исходной программе можно использовать не более 25 процедур. В паспорте станка записываются специальные операторы, указывающие какая информация автоматически выводится постпроцессором в начале и в конце управляющей программы (см. инструкцию по составлению паспортов, п. 11.1, 11.2).

8.2 ЧАСТОТА ВРАЩЕНИЯ ШПИНДЕЛЯ. ПОДАЧА.

Число оборотов шпинделя в минуту задается оператором N:

N/число оборотов[,номер диапазона];
Nномер

"число оборотов" - арифметическое выражение.

Данное "Nномер" должно быть определено в разделе данных: Nномер=число оборотов;. Знак числа оборотов определяет направление вращения шпинделя.

Примеры: N1; N/300; N/-300; N/300,2;

Информация о кодировании в управляющей программе частоты вращения шпинделя задается в паспорте станка и в специальном файле - таблице кодирования оборотов шпинделя (см. инструкцию по составлению паспортов, п. 8). Если таблица кодирования оборотов шпинделя содержит информацию о кодировании оборотов шпинделя не в одном, а в нескольких диапазонах, то номер диапазона задается вторым параметром в операторе N: "номер диапазона" - арифметическое выражение.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Номер диапазона действителен для всех последующих операторов "N". Если номер диапазона не указан, то по умолчанию предполагается диапазон 1.

Пример: ПРОГРАММА=ДИАПАЗОН;
 СТАНОК=112;!
 НПО; N/100,1;* Выбор диапазона 1;
 S/100; ТК=0,0; ТК=10,0;
 N/200,2;* Выбор диапазона 2;
 S/50; X/100;
 КПО;!

Пример новой структуры таблицы кодирования оборотов шпинделя. Если таблица состоит из одного диапазона, то фраза ДИАПАЗОН 1; не нужна.

```
ТАБЛИЦА=TS24;
ДИАПАЗОН 1;
11=12.5;
. . .
23=100;
. . .
49=2000;
ДИАПАЗОН 2;
61=12.5;
. . .
75=200;
. . .
89=2000;!
```

Оператор "S" задает подачу в миллиметрах в минуту. Оператор "SN" задает подачу в миллиметрах на оборот:

SN/подача в мм/об ; SNномер S/подача в мм/мин; Sномер
--

"подача в мм/об", "подача в мм/мин" - арифметические выражения. Данные "SNномер", "Sномер" должны быть определены в разделе данных: SNномер = подача в мм/об;
Sномер = подача в мм/мин;

Примеры: S1; S/500; SN2; SN/0.1;

Первому оператору "SN" должен предшествовать оператор "N". В языке САПР-ЧПУ нет специального оператора "ускоренная подача". Для быстрого перемещения нужно задать подачу, заведомо больше максимальной рабочей подачи. Сведения о кодировании подачи в управляющей программе записаны в паспорте станка (см. инструкцию по составлению паспортов, п. 4.1, 5).

8.3 СКОРОСТЬ РЕЗАНИЯ

Оператор "V" задает скорость резания в метрах в минуту. Оператор используется для токарных станков, обеспечивающих автоматическое поддержание постоянной скорости резания.

V/скорость резания; Vномер

"скорость резания" - арифметическое выражение. Данное "Vномер" должно быть определено в разделе данных:

Vномер = скорость резания;

Примеры: V1; V/35;

Если в программе используется оператор "V", то при изменении подачи S или изменении диаметра в управляющей программе автоматически кодируется частота вращения шпинделя:

$$N = (\text{скорость резания} * 1000) : (3.14... * \text{диаметр}).$$

При смене размерности подачи с мм/мин на мм/об в управляющую программу автоматически записывается информация о скорости резания из паспорта (см. инструкцию по составлению паспортов, п. 8).

Оператор "V" должен записываться раньше операторов "S" и "SN".

8.4 ОПЕРАТОРЫ, ОПРЕДЕЛЯЮЩИЕ ДВИЖЕНИЕ ИНСТРУМЕНТА

1. Оператор ОТ ТК - движение от точки.

ОТ ТКномер;

2. Оператор ДО ТК - движение до точки.

ДО ТКномер;

3. Оператор ОТ ПР - движение от точки пересечения прямой с элементом, определяемым в следующем операторе движения.

ОТ ПРномер;

4. Оператор ОТ КР - движение от точки пересечения окружности с элементом, определяемым в следующем операторе движения.

ОТ КРномер;

5. Оператор ПР - движение по прямой.

ПРномер;

6. Оператор +КР - движение по окружности против часовой стрелки.

+КРномер;

7. Оператор -КР - движение по окружности по часовой стрелке.

-КРномер;

8. Оператор ДО ПР - движение до точки пересечения прямой с элементом, заданным в предыдущем операторе движения.

ДО ПРномер;

9. Оператор ДО КР - движение до точки пересечения окружности с элементом, заданным в предыдущем операторе движения.

ДО КРномер;

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

10. Оператор X - перемещение, параллельное оси X в положительном ($X>0$) или отрицательном ($X<0$) направлении (смещение по X).

Xномер;
X/a.в.

11. Оператор Y - перемещение, параллельное оси Y в положительном ($Y>0$) или отрицательном ($Y<0$) направлении (смещение по Y).

Yномер;
Y/a.в.

12. Оператор R - скругление пересекающихся элементов заданным радиусом.

Rномер;
R/a.в.

13. Оператор Z - подъем ($Z>0$) или опускание ($Z<0$) инструмента перпендикулярно плоскости обработки (приращение по Z).

Zномер;
Z/a.в.

14. Оператор ZA - перемещение инструмента перпендикулярно плоскости обработки в указанную абсолютную координату Z.

ZAномер;
ZA/a.в.

15. Оператор J - точность аппроксимации окружности ломаной линией.

Jномер;
J/a.в.

Необходимость линейной аппроксимации окружностей указывается в паспорте станка (см. инструкцию по составлению паспортов, п. 6.2). Оператор J задает расстояние между дугой окружности и стягивающей хордой. Оператор J действует на все последующие операторы +KP, -KP. Если оператор J отсутствует, то по умолчанию принимается $J=0.02$ мм.

8.5 ЗАПИСЬ ПРОЦЕДУРЫ

1. После операторов OT TK, DO TK разрешается использовать операторы: DO TK, PR, +KP, -KP, X, Y (см. рис. 55а, 55б, 55в, 55д).

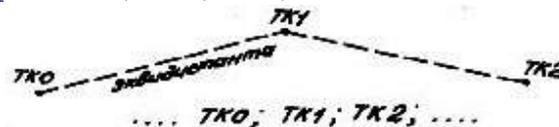


Рисунок 55а

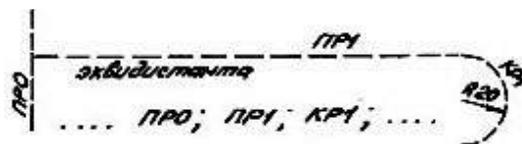


Рисунок 55б

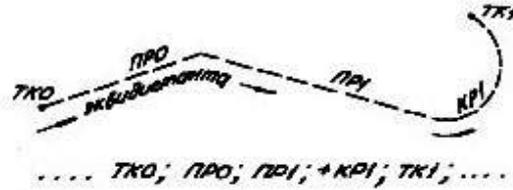


Рисунок 55в

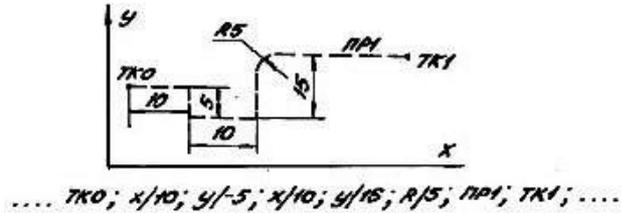


Рисунок 55д

2. После операторов ОТ ПР и ОТ КР разрешается использовать операторы: ПР, +КР, -КР. При пересечении прямой с окружностью или двух окружностей с помощью признака пересечения из двух возможных точек пересечения выбирается нужная:

- БХ; -правая точка (больше X);
- МХ; -левая точка (меньше X);
- БУ; -верхняя точка (больше Y);
- МУ; -нижняя точка (меньше Y).

Пример: ОТ ПР1; БУ; -КР2; ... Движение от верхней точки пересечения ПР1 с КР2 по окружности КР2.

Признак пересечения действует на все последующие пересечения до его замены другим признаком БХ, МХ, БУ, МУ.

3. После операторов ПР, +КР, -КР разрешается использовать операторы: ПР, +КР, -КР, ДО ТК, ДО ПР, ДО КР. При необходимости задается признак пересечения БХ, МХ, БУ, МУ (см. рис. 55б, 55в, 55г).

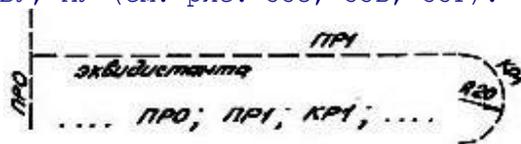


Рисунок 55б



Рисунок 55в

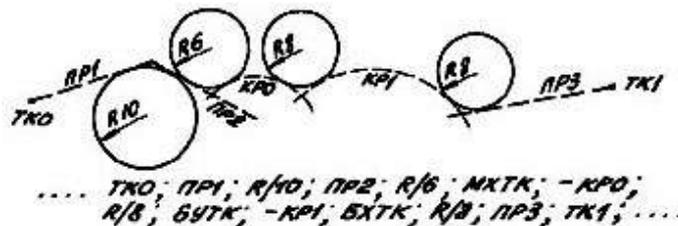


Рисунок 55г

4. После операторов ДО ПР, ДО КР разрешается использовать операторы: ДО ТК, ПР, +КР, -КР, X, Y. При необходимости задается признак пересечения.

5. После операторов X, Y разрешается использовать операторы: X, Y, ДО ТК, ПР, +КР, -КР (см. рис. 55д).

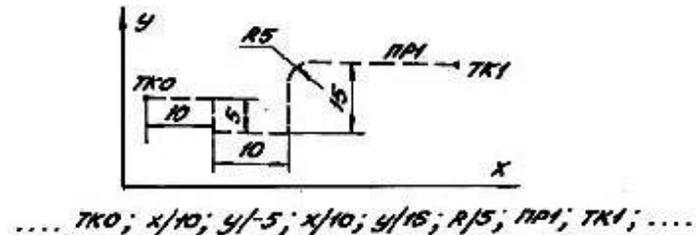


Рисунок 55д

6. Оператор R записывается между двумя операторами, определяющими движение по пересекающимся элементам. При необходимости задается признак пересечения. Оператор R локального действия, на последующие пары пересекающихся элементов не действует (см. рис. 55г, 55д).

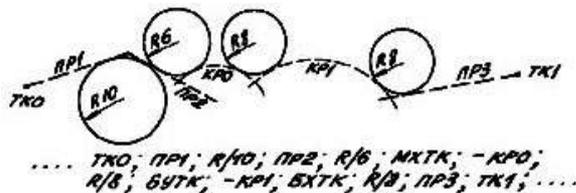


Рисунок 55г

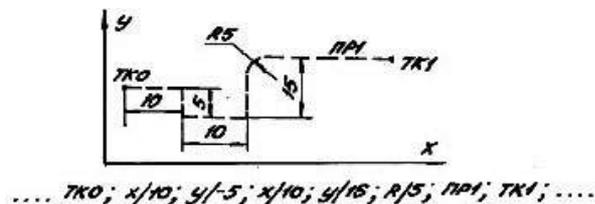


Рисунок 55д

7. Перед самым первым перемещением должна быть задана подача. Если используется оператор SN, то обязательным является и оператор N.

8. В начале процедуры задается начальная точка движения одним из операторов: ОТ ТК, ОТ ПР, ОТ КР.

9. В конце процедуры задается конечная точка движения одним из операторов: ДО ТК, ДО ПР, ДО КР.

10. Последовательность операторов ПР, +КР, -КР, ДО ТК, X, Y задает порядок и направление движения инструмента.

11. Смежные элементы контура должны пересекаться или касаться.

Примеры: ... ПР1; ДО ПР2; ПР2; ... - не правильно
 ... ПР1; ПР2; ... - правильно
 ... ПР1; ДО ПР2; ПР1; ... - правильно

13. В операторах ОТ ТК, ДО ТК предлоги ОТ, ДО можно опускать, в операторах ОТ ПР, ДО ПР, ОТ КР, ДО КР предлоги ОТ, ДО опускать нельзя.

14. Между операторами движения в соответствии с условиями обработки записываются другие операторы языка, которые будут рассмотрены в следующих параграфах.

Пример: НР0; S/8000; N/300; ТК0; ТК1; Z/-20; SN/0.15; ПР1; -КР1; ПР2; ДО ПР1; S/8000; ТК0; Z/20; КР0; !

8.6 АВТОМАТИЧЕСКИЙ РАСЧЕТ ЭКВИДИСТАНТЫ.

Под эквидистантой понимается траектория движения центра инструмента. Центр инструмента - это центр фрезы, центр скругления вершины резца, центр сечения проволочного электрода и т.п. Сам инструмент можно представить окружностью, центр которой - это центр инструмента, а радиус - это половина наружного диаметра фрезы, радиус скругления вершины резца и т.п.

В программе можно определить элементы эквидистанты (точки, прямые, окружности) и задать движение центра инструмента по эквидистанте (см. рис. 58) или определить элементы обрабатываемого контура и задать движение инструмента вдоль контура. В последнем случае задаются расстояние от центра инструмента до контура (оператор Р) и положение инструмента относительно контура - справа или слева по ходу движения (операторы ФР+, ФР-).

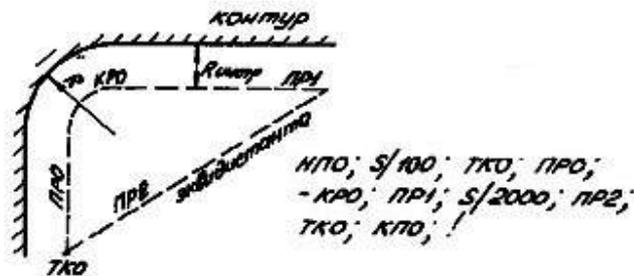


Рисунок 58

Оператор ФР+, движение центра инструмента справа от контура:

ФР+;

Все последующие операторы движения ТК, ПР, +КР, -КР, X, Y, R определяют движение центра инструмента справа от контура по ходу движения на расстоянии, заданном оператором Р (см. рис. 60).

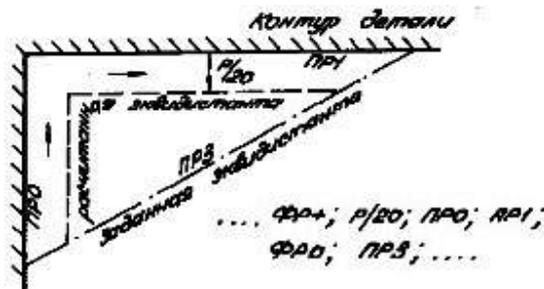


Рисунок 60

Оператор ФР-, движение центра инструмента слева от контура:

ФР-

Все последующие операторы движения ТК, ПР, +КР, -КР, X, Y, R определяют движение центра инструмента слева походу движения от указанных элементов контура на расстоянии, заданном оператором R (см.рис. 61).

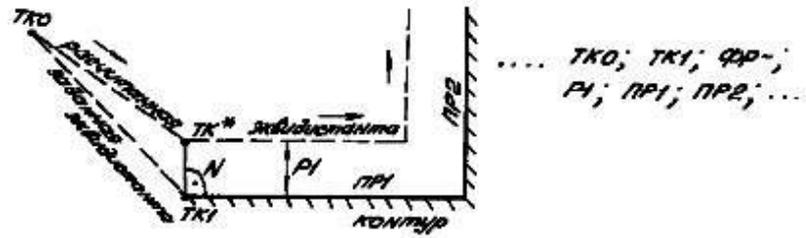


Рисунок 61

Оператор R, расстояние от центра инструмента до контура:

```

R / < арифметическое выражение >;
R < номер >;
    
```

Оператор R действует на все последующие операторы движения.

Числовые значения X, Y при расчете эквидистанты соответствуют размерам на контуре, а не величинам смещения центра инструмента (см.рис. 67).

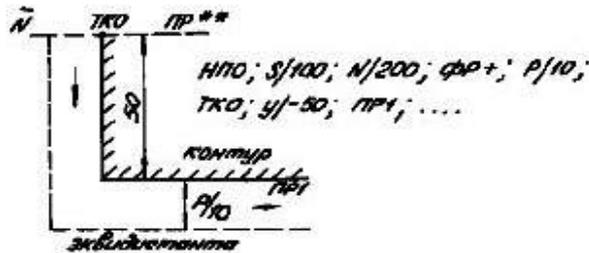


Рисунок 67

Оператор ФR0, отмена автоматического расчета эквидистанты:

```

ФR0;
    
```

Все последующие операторы ТК, ПР, +КР, -КР, X, Y, R определяют движение по указанным элементам, а не справа или слева от них.

В начале процедуры до операторов R, ФR+, ФR- и при отсутствии предполагается по умолчанию действие оператора ФR0.

В процедуре разрешается неоднократно вводить и отменять автоматический расчет эквидистанты, менять положение центра инструмента относительно контура (справа или слева), изменять расстояние от центра до контура.

8.6.1 НАЧАЛО АВТОМАТИЧЕСКОГО РАСЧЕТА ЭКВИДИСТАНТЫ (ВЫХОД НА ЭКВИДИСТАНТУ).

Имеются три варианта начала автоматического расчета эквидистанты.

Вариант 1.

```

          ТК          ПР
...ДО  ПР<J>; ФР+; R; +КР<K>; ...
          КР          ФР-   -КР
    
```

Для компактности формата символ P заменяет запись P/< арифметическое выражение > или P < номер >.

Оператор ДО ТК<J>, ДО ПР<J>, ДО КР<J> определяет точку на элементе K, в которой САПР - ЧПУ строит нормаль к элементу K. Центр инструмента из предыдущего положения перемещается по прямой в точку пересечения нормали с эквидистантой к элементу K, затем движение продолжается по эквидистанте (см.рис. 61,62,63). Элементом выхода на эквидистанту в этом варианте будет всегда прямая, рассчитанная САПР - ЧПУ, даже если в исходной программе задана окружность (см.рис. 64).

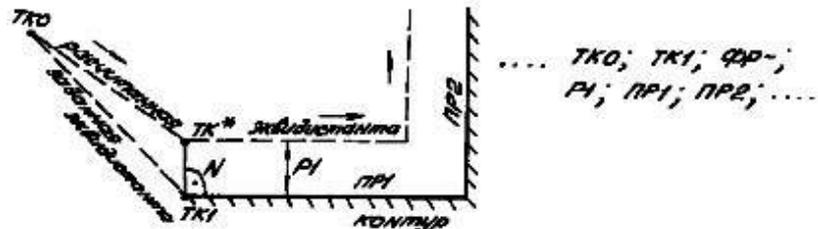


Рисунок 61

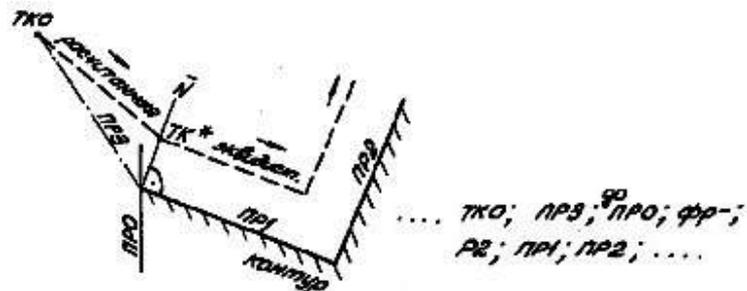


Рисунок 62

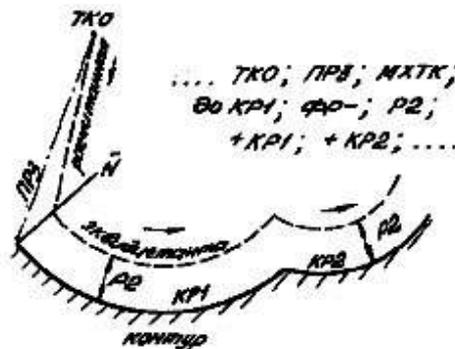


Рисунок 63

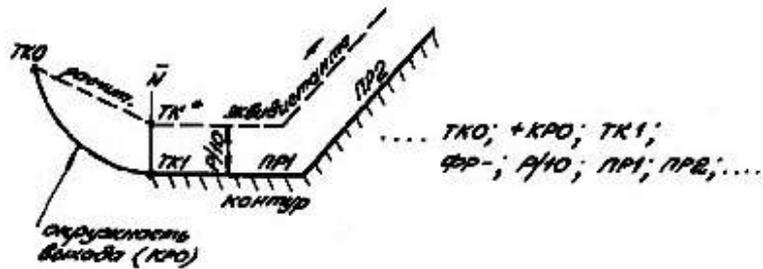


Рисунок 64

Вариант 2.

	ПР	ПР
...	+КР<I>; ФР+; Р;	+КР<K>; ...
	-КР	ФР- -КР

Центр инструмента перемещается по элементу I до точки пересечения с эквидистантой к элементу K, затем движение продолжается по эквидистанте (рис. 65, 66)

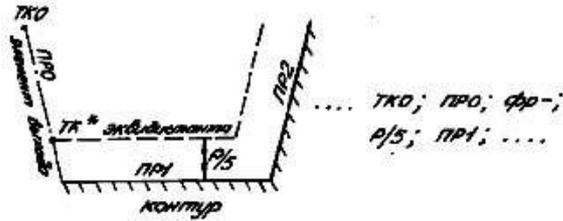


Рисунок 65

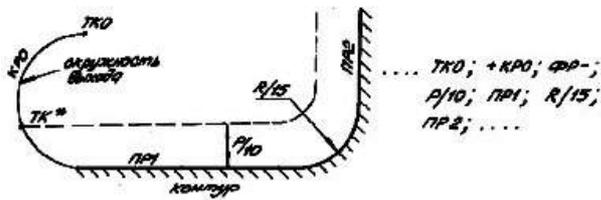


Рисунок 66

Вариант 3.

	ТК	ПР
НП<N>; ФР+; Р; ОТ	ПР<J>; +КР<K>; ...	
СМ	ФР-	КР -КР

Данный вариант можно использовать только в начале процедуры и после операторов СМ (смена инструмента), ПЛ ХУ, ПЛХЗ, ПЛ YZ (смена плоскости обработки), ПЕРЕХОД (разрыв контура). Центр инструмента из начальной точки перемещается по нормали к элементу K до точки пересечения с эквидистантой, затем движение продолжается по эквидистанте (см.рис. 67, 68).

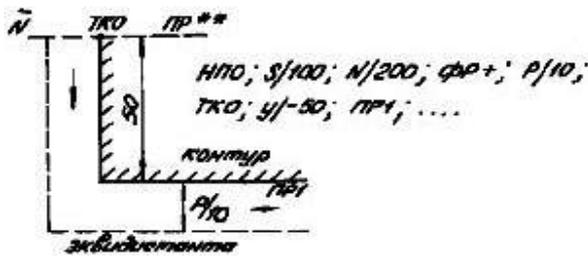


Рисунок 67

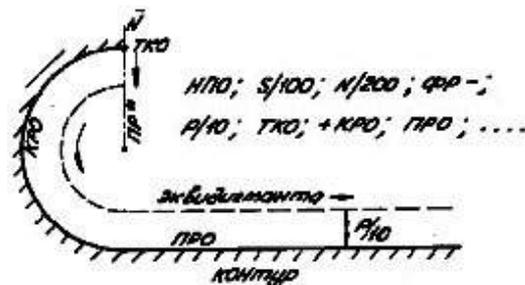


Рисунок 68

Итак, автоматический расчет эквидистанты начинается с элемента, идентификатор которого указан в первом операторе движения после ФР+, ФР-, а вариант расчета определяется предшествующим оператором:

ДО ТК, ДО ПР, ДО КР - вариант 1;
 ПР, +КР, -КР - вариант 2;
 НП, СМ - вариант 3.

8.6.2 КОНЕЦ АВТОМАТИЧЕСКОГО РАСЧЕТА ЭКВИДИСТАНТЫ.

Аналогично вводу возможны три варианта отмены автоматического расчета эквидистанты.

Вариант 1.

ТК
... ДО ПР<J>; ФР0; ДО ТК<K>; ...
КР

Оператор ДО ТК, ДО ПР, ДО КР определяет точку на контурном элементе, в которой строится нормаль.

Центр инструмента из точки пересечения эквидистанты с этой нормалью перемещается по прямой до ТК<K> (см.рис.70,71,72). Сход с эквидистанты всегда осуществляется по прямой, рассчитанной САПР - ЧПУ, даже если в исходной программе задан сход по окружности (см.рис. 73).

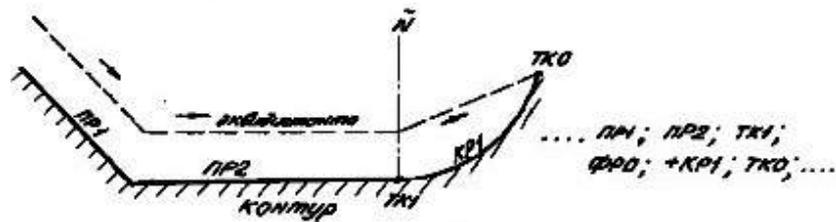


Рисунок 73

Вариант 2.

ПР	ПР
... +КР<I>; ФР0; +КР<J>; ...	
-КР	-КР

Отмена автоматического расчета эквидистанты осуществляется в точке пересечения эквидистанты с элементом J, затем движение центра инструмента продолжается по этому элементу (см.рис. 74,75).

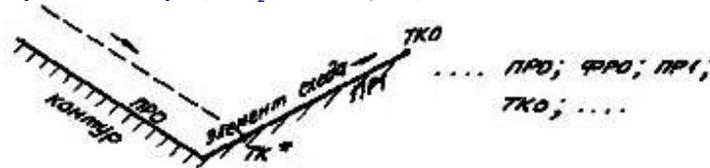


Рисунок 74

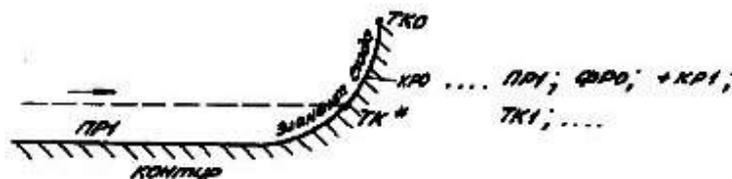


Рисунок 75

Вариант 3.

Если в конце процедуры нет оператора ФР0, то выход в конечную точку назначается САПР - ЧПУ автоматически по нормали к последнему элементу (см.рис. 76,77).

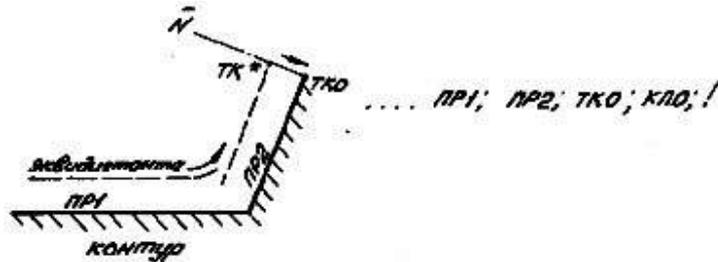


Рисунок 76

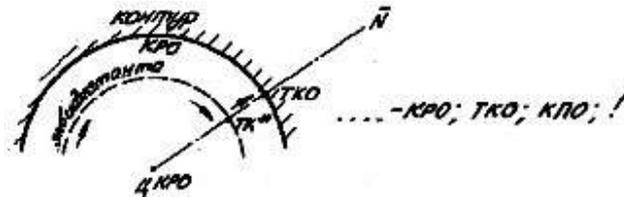


Рисунок 77

8.6.3 ИНТЕРПРЕТАЦИЯ ОПЕРАТОРОВ ДОТК, ДОПР, ДОКР ПРИ АВТОМАТИЧЕСКОМ РАСЧЕТЕ ЭКВИДИСТАНТЫ.

Если при движении по прямой или окружности необходимо в некоторой точке выполнить определенные операции (сменить подачу, опуститься или подняться по Z , ввести техкоманду и т.п.), а затем продолжить движение по тому же элементу, то используются операторы ДОТК, ДОПР, ДОКР:

ПР	ТК	S	ПР
... +КР<I>;	ДО ПР<J>;	Z;	+КР<I>; ...
-КР	КР	...	-КР

Пример:

...ФР-;...ПР1;ДО ПР2;S/1000;Z/20;ПР1;...

Операторы ДО ТК, ДО ПР, ДО КР определяют точку на контурном элементе, в которой строится нормаль. Пересечение нормали с эквидистантой определяет точку, в которой будут выполняться указанные операции.

8.6.4 НЕПЕРЕСЕЧЕНИЕ ЭЛЕМЕНТОВ ЭКВИДИСТАНТЫ.

Смежные элементы контура всегда должны пересекаться или касаться, соответствующие элементы эквидистанты могут не пересекаться. В этом случае САПР - ЧПУ назначает перемещение с одного элемента эквидистанты на другой:

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

а) по прямой, соединяющей центры окружностей, если смежные элементы - окружности (см.рис. 80);

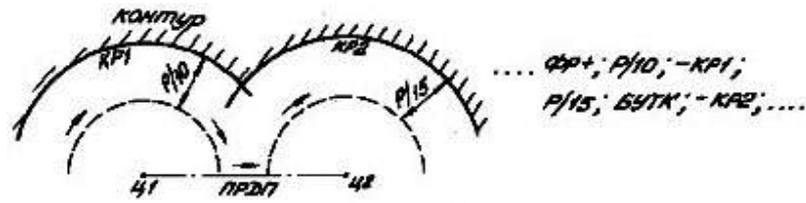


Рисунок 80

б) по перпендикуляру, восстановленному к прямой из центра окружности, если смежные элементы - прямая и окружность (см.рис. 81)

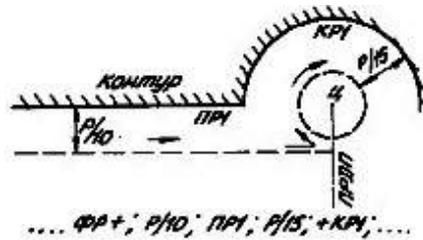


Рисунок 81

Чтобы избежать в таких ситуациях дополнительных перемещений, можно использовать оператор скругления R с минимальным радиусом 0.1 мм. При этом расстояние от центра инструмента до обоих элементов должно быть одинаковым (см.рис. 82,83).

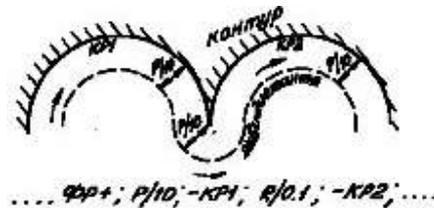


Рисунок 82



Рисунок 83

8.7 ИСКАЖЕНИЕ КОНТУРА ПРИ АВТОМАТИЧЕСКОМ РАСЧЕТЕ ЭКВИДИСТАНТЫ.

При автоматическом расчете эквидистанты могут возникнуть ситуации, когда САПР - ЧПУ "корректирует" контур, описанный в программе, чтобы избежать зареза инструмента в деталь:

1. Радиус инструмента больше радиуса окружности на контуре. В этом случае окружность заменяется прямой. (см.рис. 84).

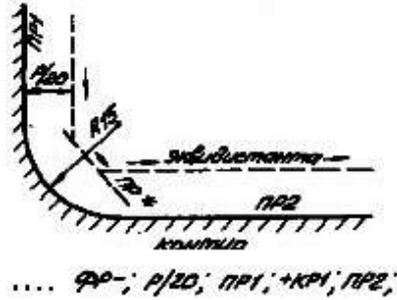


Рисунок 84

2. Направление движения по элементу на эквидистанте противоположно направлению движения по соответствующему элементу на контуре. В этом случае геометрический элемент "выбрасывается" (см.рис. 85,86). Операторы S, Z, F, Q и техкоманды, записанные перед оператором движения по этому элементу, игнорируются.

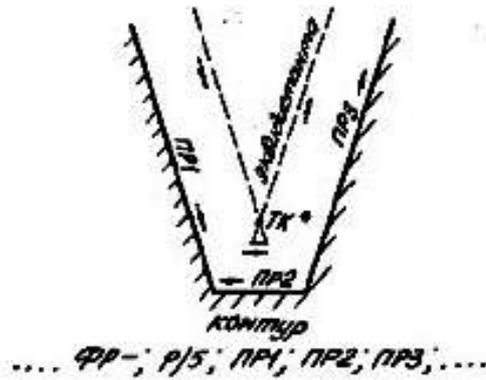


Рисунок 85

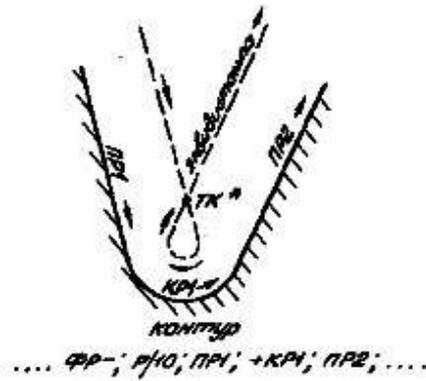


Рисунок 86

8.8 ПОДАЧА ПРИ АВТОМАТИЧЕСКОМ РАСЧЕТЕ ЭКВИДИСТАНТЫ

При автоматическом расчете эквидистанты технолог должен задавать подачу для эквидистанты, а не для контура, САПР - ЧПУ подачу автоматически не пересчитывает.

8.9 КОРРЕКЦИЯ ЭКВИДИСТАНТЫ

Существует два вида коррекции:

- а) коррекция на длину и положение инструмента;
- б) коррекция на радиус инструмента.

Коррекция осуществляется системой управления станком.

8.9.1. КОРРЕКЦИЯ НА ДЛИНУ И ПОЛОЖЕНИЕ ИНСТРУМЕНТА

Операторы ввода коррекции:

$+KX<I>$, $+KY<I>$, $+KZ<I>$, $+KXY<I>$, $+KXZ<I>$, $+KYZ<I>$, $+KXYZ<I>$
--

Операторы отмены коррекции:

$-KX<I>$, $-KY<I>$, $-KZ<I>$, $-KXY<I>$, $-KXZ<I>$, $KYZ<I>$, $-KXYZ<I>$

"+"	- признак ввода коррекции,
"-"	- признак отмены коррекции,
X, Y, Z	- коррекция по одной координате,
XY, XZ, YZ	- коррекция по двум координатам,
XYZ	- коррекция по трем координатам,
I	- номер корректора.

Каждый оператор коррекции является техкомандой, описанной в паспорте станка (техкоманды рассматриваются в п. 7.12 данной инструкции). Оператор коррекции вызывает запись в управляющую программу информации из соответствующей техкоманды. На последующие операторы в процедуре операторы коррекции никакого влияния не оказывают, то есть оператор коррекции – операторы одноразового действия.

Операторы ввода и отмены коррекции записываются перед соответствующим перемещением, то есть $+KX$, $-KX$ перед перемещением по X и т.д.

Число цифр в номере корректора (от одной до шести) указывается в паспорте.

Пример: (см.рис. 87) $+KY01$; ... $-KY01$; ... Коррекция по Y, величина коррекции устанавливается на корректоре 01.

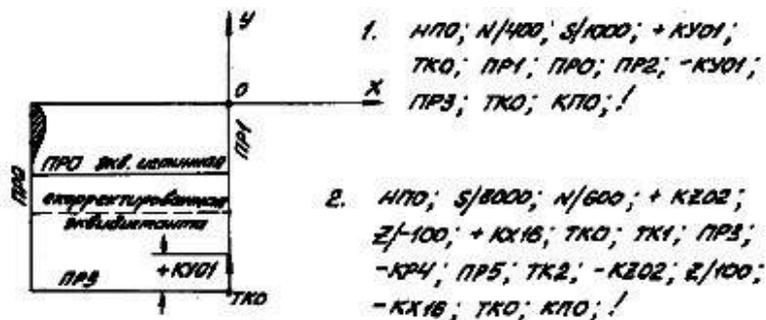


Рисунок 87

8.9.2 КОРРЕКЦИЯ РАДИУСА ИНСТРУМЕНТА

Операторы ввода и отмены коррекции радиуса инструмента:

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

- +НК [I] - ввод коррекции при движении центра инструмента справа от контура по ходу движения (рис. 88),
- НК [I] - ввод коррекции при движении центра инструмента слева от контура по ходу движения (рис.89),
- КК [I] - конец коррекции.

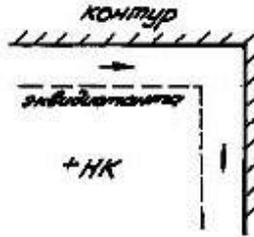


Рисунок 88

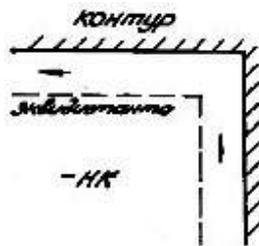


Рисунок 89

I - номер корректора, число цифр в номере указано в паспорте, для некоторых систем ЧПУ номер корректора не указывается.

Операторы +НК, -НК действуют на последующие операторы движения до оператора КК.

В САПР - ЧПУ имеются три типа коррекции радиуса инструмента: осепараллельная (позиционная), эквидистантная и контурная. Тип коррекции указан в паспорте станка (см.инструкцию по паспортам, п.7.1.).

При осепараллельной коррекции коррекция траектории осуществляется только параллельно осям координат (позиционные системы ЧПУ).

При эквидистантной коррекции система ЧПУ корректирует рассчитанную ЭВМ эквидистанту. В исходной программе обязательно используются операторы ФР+, ФР-. Положение инструмента (+НК - справа, -НК-слева) указывается относительно эквидистанты.

При контурной коррекции система ЧПУ сама рассчитывает эквидистанту к контуру в зависимости от величины радиуса, установленной на корректоре, операторы ФР+, ФР- не используются.

8.9.3 СКРУГЛЕНИЕ КОНТУРА ИЛИ ЭКВИДИСТАНТЫ

В паспорте станка может использоваться оператор ОСК (общие сведения о коррекции) :

ОСК=ТИП, Re, Rmin, Mn;

Если в паспорте Rmin не равен 0, то САПР-ЧПУ обеспечивает автоматическое скругление контура при контурной коррекции (ТИП=1) или эквидистанты при эквидистантной коррекции (ТИП=2) .

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Если в операторе ОСК заданы:
 ТИП=1, Rmin#0 или ТИП=2, Rmin#0, Re=0
 ,то углы скругляются минимальным радиусом скругления Rmin или радиусом R инструмента +Rmin (см.рис. 90). Поэтому радиус инструмента R нужно указывать не только при эквидистантной коррекции, но и при контурной.

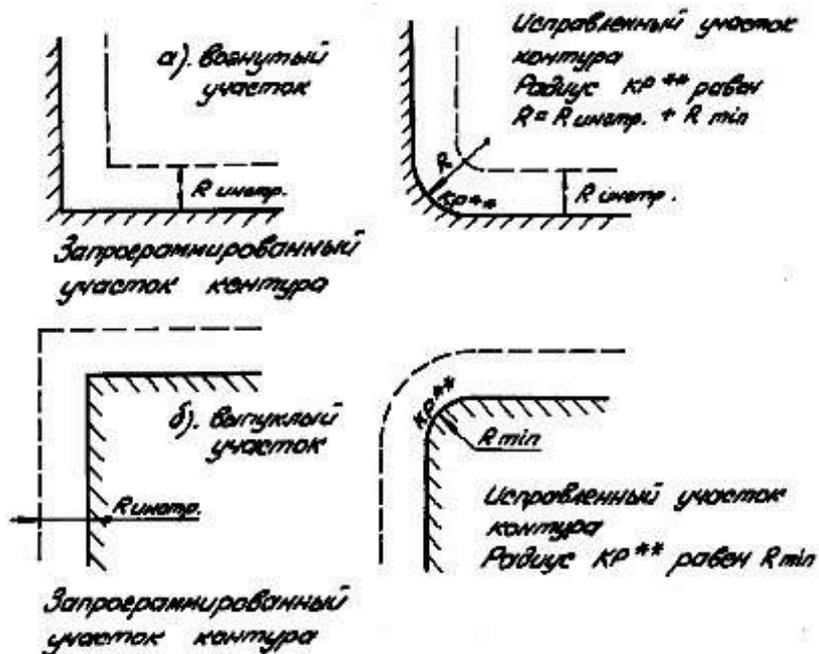


Рисунок 90

Если же ТИП=2, Rmin#0 и Re#0, то внутренние и внешние углы скругляются радиусом Re (т.е. Rmin и радиус R не используются).

Если в паспорте минимальный радиус скругления R min указан равным нулю, то автоматическое скругление не осуществляется. В этом случае, если система ЧПУ требует гладкого контура при коррекции, технолог должен сам скруглять контур с помощью оператора R.

При автоматическом скруглении контура (эквидистанты) технологические команды и операторы позиционирования Z, F, Q записываются в управляющую программу после вставляемой окружности. Если их необходимо выдать перед окружностью, необходимо скруглить элементы с помощью оператора R, а техкоманды и операторы позиционирования записать перед ним.

8.9.4 ВВОД И СБРОС КОРРЕКЦИИ

В САПР - ЧПУ имеются три варианта ввода и сброса коррекции. Какой вариант применить в программе зависит от конкретной системы ЧПУ. Информация о кодировании ввода и сброса коррекции в управляющей программе записывается в паспорте станка (см.инструкцию по паспортам, п. 7).

Вариант 1.

Коррекция вводится на нормали к прямой (рис.91,92). Точка ввода коррекции выбирается до врезания инструмента в металл, длина отрезка L должна быть больше радиуса инструмента не менее, чем на 2 мм.

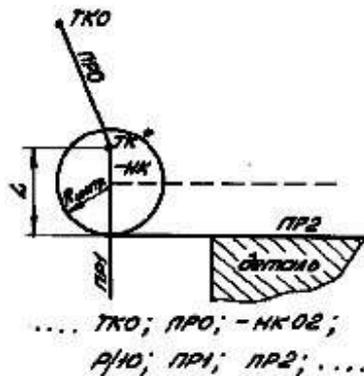


Рисунок 91

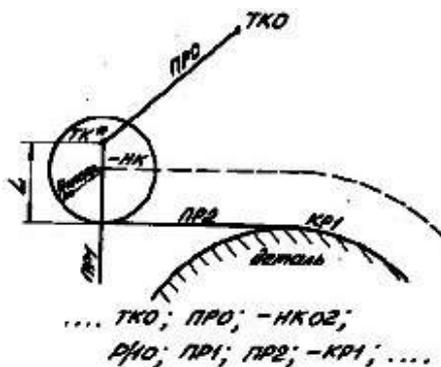


Рисунок 92

Вариант 2.

Ввод коррекции на прямой, касательной к окружности (см.рис. 93,94). Радиус окружности должен быть больше радиуса фрезы не менее, чем на 2 мм.

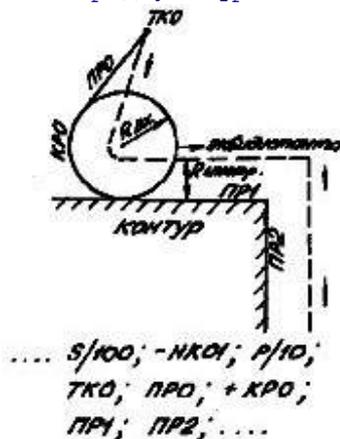


Рисунок 93

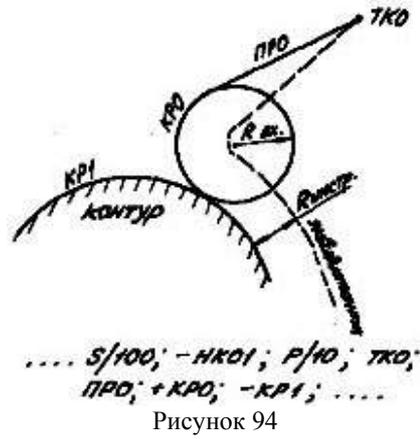


Рисунок 94

Вариант 3.

Ввод коррекции на прямой, проходящей через центр окружности (см.рис. 96).

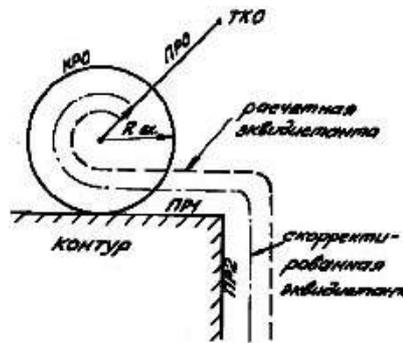


Рисунок 96

Сброс коррекции осуществляется перед последним перемещением аналогично вводу (см.рис. 97,98).

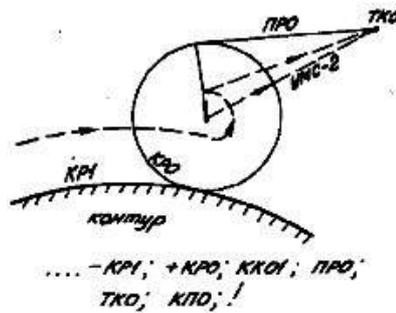
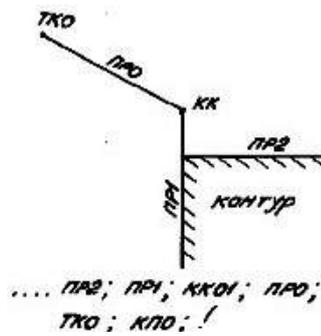


Рисунок 98

При вводе коррекции элемент, на котором производится ввод коррекции, с последующим элементом автоматически не скругляется. При сбросе коррекции элемент, на котором осуществляется сброс коррекции, с предыдущим элементом также автоматически не скругляется.

Пример:

ТК1; +НК1; ПР1; ПР2;...ПР3; КК1; ПР4; ТК2; ...
Прямые ПР1 с ПР2 и ПР3 с ПР4 не скругляются.

8.10 УЧАСТКИ

Если обрабатываемый контур содержит несколько идентичных по конфигурации и технологии обработки участков, то в исходной программе можно описать только один участок. В разделе процедур каждый раз, когда необходимо обработать такой участок, достаточно записать оператор ОБУ (см.рис. 99).

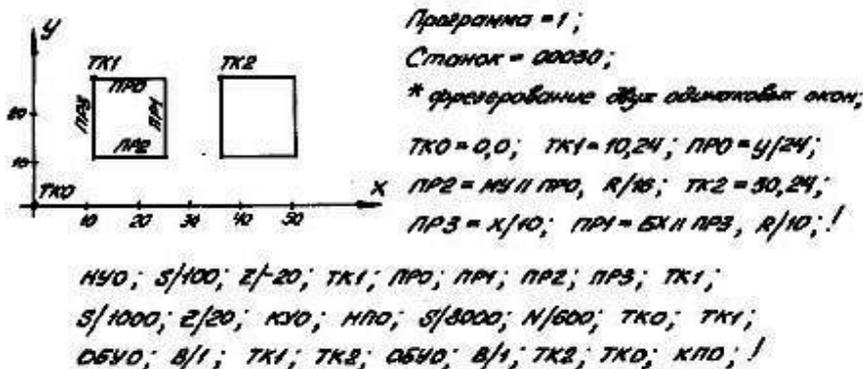


Рисунок 99

Участок в САПР - ЧПУ не обязательно соответствует части контура, он может быть последовательностью любых операторов языка.

Пример: NY0; S/200; Z/-50; S/1000; Z/50; KY0;

В версии САПР-ЧПУ/4 разрешено использование до 500 участков.

8.10.1 ПРАВИЛА ОФОРМЛЕНИЯ УЧАСТКА

1. Формат участка:

NY<I>; ФРАЗА1; ФРАЗА2; ... KY<I>;

, где: $0 \leq i \leq 499$ - номер участка.

Примечание: В зависимости от контекста под участком в данной инструкции понимается часть обрабатываемого контура или последовательность фраз, записанных между NY<I> и KY<I>. 2. Участок записывается вне процедуры, т.е. перед процедурой или после процедуры.

3. Если участок соответствует части контура, то он может быть незамкнутым, но должен иметь начальную и конечную точки, то есть не может начинаться и оканчиваться опера торами ПР, +КР, -КР:

TK	TK
NY<I>; OT ПР<J>; ... ДО ПР<K>; KY<I>;	
KP	KP

Если в участке нет операторов ПР, +КР, -КР, то начальную и конечную точки не задают.

8.10.2 ОПЕРАТОР ОБУ – ОБРАБОТАТЬ УЧАСТОК

Запись участка вне процедуры не задает самого выполнения участка. Для этого необходимо в процедуре записать оператор ОБУ. Самый простой формат оператора:

ОБУ<I>; В/1;

Выполняются операторы между фразами НУ<I> и КУ<I>, а затем операторы, записанные после фразы В/1.

8.10.3 ПРАВИЛА УПОТРЕБЛЕНИЯ ОПЕРАТОРА ОБУ

1. Перед оператором ОБУ движение по прямой или окружности должно быть ограничено одним из операторов: ДО ТК, ДО ПР, ДО КР. Эти операторы определяют точку на контуре, которая должна совпадать с начальной точкой участка (см.рис. 99).

2. Если перед оператором ОБУ нет операторов ПР, +КР, -КР, то операторы ДО ТК, ДО ПР, ДО КР перед ОБУ не нужны.

Пример:

НПО; S/200; ОТ ТК1; ОБУ0; В/1; X/100; ОБУ2; В/1;...

3. Геометрические элементы участка не обязательно описывать в системе координат детали, их можно описать в другой системе координат, исходя из удобства описания участка. САПР – ЧПУ автоматически устанавливает связь между системой координат участка и общей системой координат детали (см.рис. 100). Поэтому точка перед оператором ОБУ и точка после фразы НУ задаются в разделе данных как разные, хотя в действительности это одна точка на плоскости (см.рис. 99, второй оператор ОБУ и рис. 100).

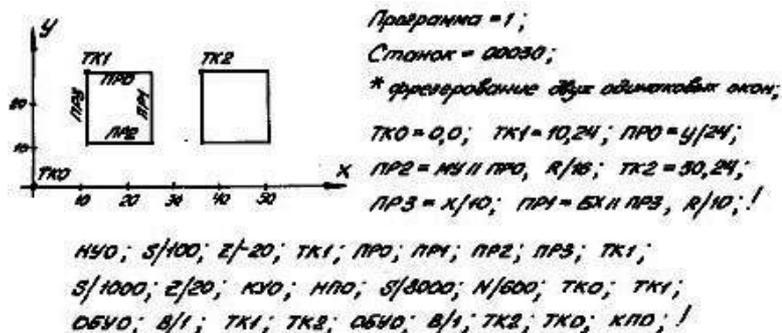


Рисунок 99

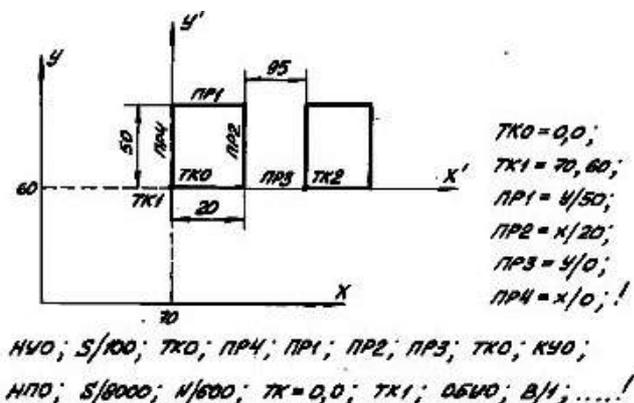
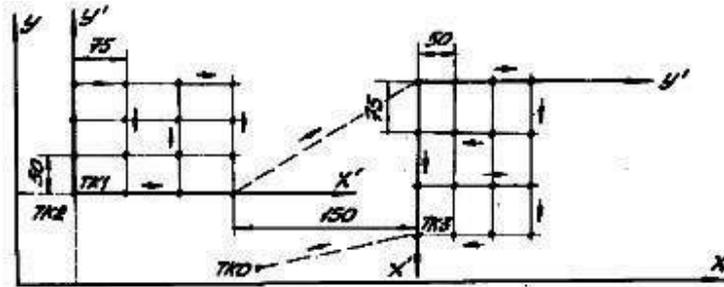


Рисунок 100

4. После выполнения участка можно использовать любой оператор движения: ДО ТК, ПР, +КР, -КР, X, Y (возможно исключение из правила, см.п.6).

Пример: ОБУ0; В/1; ПР1; ...

5. Оператор ОБУ можно использовать не только в процедуре, но и в участке, то есть участки могут быть "вложенными" друг в друга. (см.рис. 109). Глубина вложенности - не больше восьми.



TK1=0,0; TK2=10;50; TK0=250;10;
 NY0; S/40; Z/-18; S/8000; Z/12; TK1; KY0;
 NY1; ОБУ0; CY * Y/30; B/4; X/75; ОБУ0; CY * Y/-50; B/4; KY1;
 NY2; ОБУ1; CX * X/75; B/2; KY2;
 NY0; S/8000; N/400; TK0; TK2; ОБУ2; PF * P/0; P/-300000; CX * X/150;
 CY * Y/150; B/2; TK3; TK0; КПО; !

Рисунок 109

Пример максимальной вложенности:

NY0; ... ОБУ0; В/1; ... КПО;
 NY1; ... ОБУ1; В/1; ... KY0;
 NY2; ... ОБУ2; В/1; ... KY1;
 NY3; ... ОБУ3; В/1; ... KY2;
 NY4; ... ОБУ4; В/1; ... KY3;
 NY5; ... ОБУ5; В/1; ... KY4;
 NY6; ... ОБУ6; В/1; ... KY5;
 NY7; ... ОБУ7; В/1; ... KY6;
 NY8; ... ОБУ8; В/1; ... KY7;
 NY8; ... KY8;

Участок 8 уже не может содержать оператор ОБУ. Каждый участок можно описать в своей системе координат.

6. Если процедура, макропроцедура или участок начинаются с оператора ОБУ, то перед ним должен записываться оператор ОТ ТК:

НМ; НП<НОМЕР>; ОТ ТК<НОМЕР>; ОБУ<НОМЕР>; ... NY<НОМЕР>;

Примеры:

NY0; ОБУ1; В/1; ПР1; ...

НЕПРАВИЛЬНО

NY0; ОТ ТК0; ОБУ1; В/1; ПР1; ...

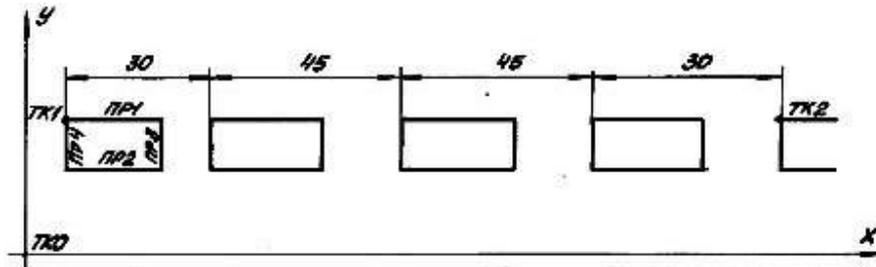
ПРАВИЛЬНО

7. Если до выполнения участка был установлен режим автоматического расчета эквидистанты (ФР+, ФР-), то он сохраняется и в участке, а также после него, если не отменялся.

8. Если до выполнения участка введена коррекция радиуса инструмента (+НК, -НК), то она сохраняется и в участке, а также после него, если не было сброса коррекции.

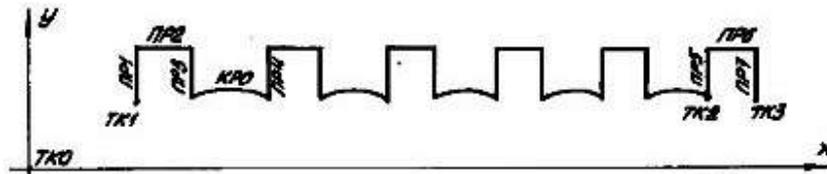
8.10.4 ВЫПОЛНЕНИЕ ПОСЛЕДОВАТЕЛЬНОСТИ УЧАСТКОВ

Часть контура может состоять из последовательности идентичных участков, в который каждый участок, начиная со второго, получается "переносом" предыдущего участка с поворотом на определенный угол или без поворота. В этом случае используется оператор ОБУ с указанием числа участков (см.рис. 101, 102, 103).



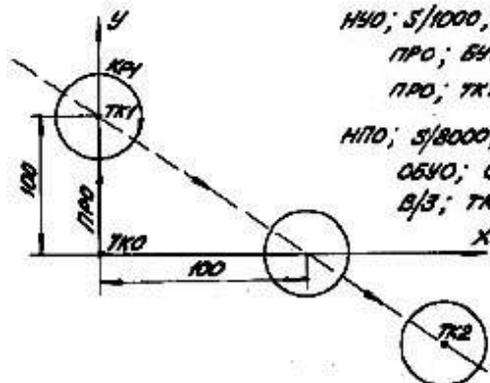
*НЧО; S/100; Z/-10; ТК1; ПР1; ПР3; ПР2; ПР4; ТК1; Z/10;
S/8000; КУО;
НПО; S/8000; N/350; ТК0; ТК1; ОБУО; СК = X/30; X/45(2);
X/30; B/5; ТК2; ТК0; КПО; !*

Рисунок 101



*НЧО; S/80; ТК1; ПР1; ПР2; ПР3; БУТК; -КРО; БУТК; ПР4; КУО;
НПО; S/8000; N/400; ТК0; ТК1; Z/-100; -НКО1; ОБУО; B/5; ТК2;
ПР5; ПР6; ПР7; ТК3; ТК0; КПО; !*

Рисунок 102



*НЧО; S/1000; Z/-10; S/60; ТК1;
ПРО; БУТК; -КР1; МУТК;
ПРО; ТК1; S/1000; Z/10; КУО;
НПО; S/8000; N/250; ТК0; ТК1;
ОБУО; СК = X/100; СУ = Y/-100;
B/3; ТК2; ТК0; КПО; !*

Рисунок 103

8.10.4.1 ОБРАБОТКА ПОСЛЕДОВАТЕЛЬНОСТИ УЧАСТКОВ С ПЕРЕНОСОМ, НО БЕЗ ПОВОРОТА

Вариант 1.

Если в последовательности участков конечная точка одного участка является начальной точкой следующего участка, а каждый участок, начиная со второго,

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

получается переносом без поворота предыдущего участка (см.рис. 102), то используется следующий оператор:

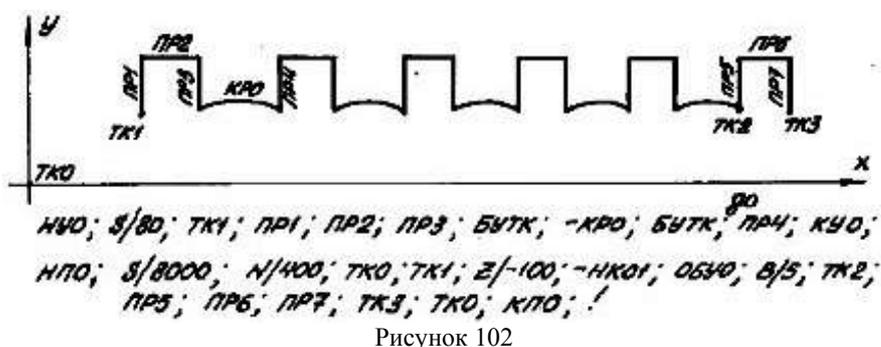
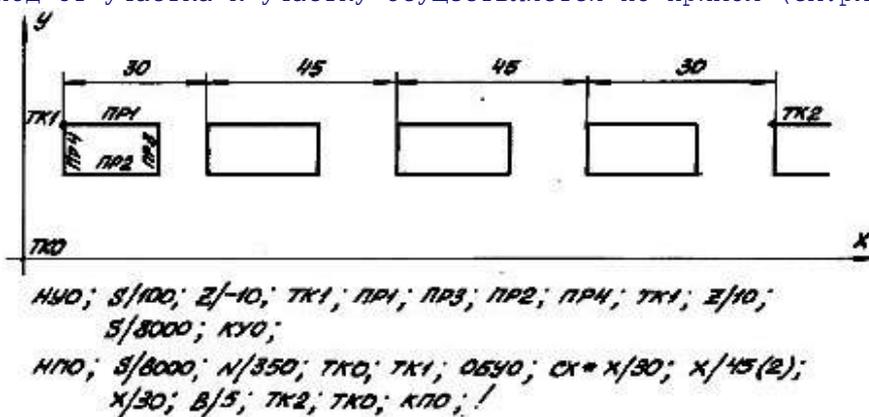
ОБУ<Т>; В/<АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ>;
В<J>

Значение арифметического выражения или данного В<J> определяет число выполнений участка I .

Часто участки не соединяются друг с другом. В этом случае можно включить в участок в качестве последнего геометрический элемент, соединяющий конечную точку участка с начальной точкой следующего участка, но для последнего участка это перемещение может быть лишним (см.рис. 102). Если перемещение от участка к участку осуществляется по прямой, удобнее использовать второй вариант.

Вариант 2.

Участки не соединяются друг с другом. Для всех участков, кроме последнего задаются смещения по X и Y от конечной точки до начальной точки следующего участка. Переход от участка к участку осуществляется по прямой (см.рис. 101).



ОБУ<I>; СХ*Х1; Х2; ... СУ*У1; У2; ... В/<А.В.>;
В<J>

X1.X2,...Y1,Y2... - числа или идентификаторы, X1.Y1 - смещение от конечной точки первого участка до начальной точки второго участка; X2.Y2 - смещение от конечной точки второго участка до начальной точки третьего участка и т.д.

Число смещений должно быть на единицу меньше числа выполнений участка, так как после последнего выполнения участка инструмент не смещается из конечной точки.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Можно указать смещения только по X или только по Y, если указаны обе конструкции CX и CY, то смещение будет по наклонной прямой, если после какого-нибудь участка нужно сместиться по одной координате, нужно другую координату задать нулевой.

Пример: ОВУ0; CX*10; 20; 30; CY*10; 0; 0; В/4;

Вместо записи нескольких одинаковых значений подряд можно записать одно значение и коэффициент повторения в круглых скобках. Для последнего X или Y коэффициент повторения можно не указывать, он подсчитывается автоматически.

Пример: Следующие записи эквивалентны:

ОВУ0; CX*45; 45; 50; В/5; ОВУ0; CX*45(2); 50; В/5;

8.10.4.2 ОБРАБОТКА ПОСЛЕДОВАТЕЛЬНОСТИ УЧАСТКОВ С ПЕРЕНОСОМ И ПОВОРОТОМ УЧАСТКОВ НА ПЛОСКОСТИ.

Вариант 1.

Участки соединяются друг с другом, то есть конечная точка каждого участка, кроме последнего, является начальной точкой следующего участка. Участки могут быть повернуты относительно друг друга на разные углы. (см.рис. 104).

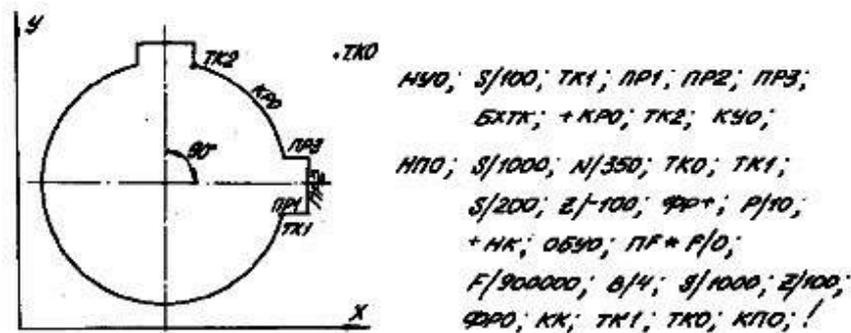


Рисунок 104

ОВУ<I>; PF*F1; F2; ... В/<A.B.>;
В<J>

F1, F2 ... - идентификатор или конструкции вида F/ЧИСЛО, число углов равно числу участков.

F1 - угол поворота для первого участка,

F2, F3, ... - угол поворота для второго, третьего участка и т.д.

Первый участок в последовательности получается, если в начальную точку перенести участок, описанный в исходной программе и повернуть его с центром поворота в начальной точке на угол F1. Если поворота нет, F1 задается равным нулю.

Каждый участок, начиная со второго, получается переносом предыдущего участка в начальную точку и его поворотом с центром поворота в начальной точке на соответствующий угол.

Угол положительный, если поворот против часовой стрелки, в противном случае угол отрицательный.

При задании углов можно использовать коэффициент повторения, у последнего угла коэффициент повторения можно опустить.

Углы поворота участка задаются приращениями, а не абсолютными значениями.

Если угол нельзя задать точно без округления, то нужно использовать арифметическое выражение.

Пример: $F1 = F/360. : 11; \dots$ ОБУ0; ПФ*F/0; F1; В/11; \dots

Вариант 2.

Участки не соединяются друг с другом. Для каждого участка, кроме последнего, задаются смещения по X и Y от конечной точки до начальной точки следующего участка. Участки могут образовывать друг с другом разные углы. Переход от участка к участку осуществляется по прямой (см. рис. 107).

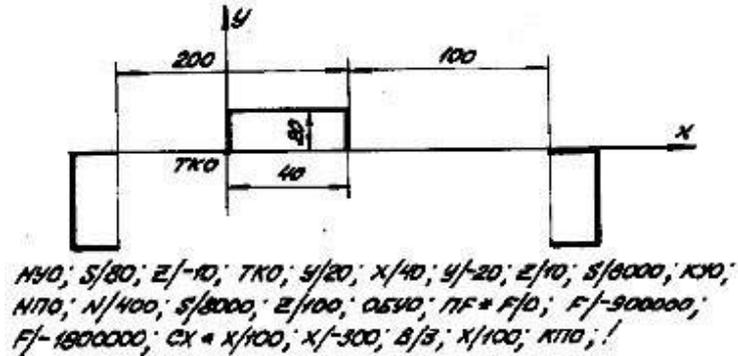


Рисунок 107

ОБУ<I>; CX*X1; X2... CY*Y1; Y2... ПФ*F1; F2... В/<A.B.>;
В<J>

Конструкции CX, CY, ПФ рассмотрены ранее. Следует только обратить внимание, что поворот участка осуществляется перед его выполнением, а смещение по X и по Y после выполнения участка.

Вариант 3.

Участки могут соединяться или не соединяться друг с другом. Задан общий для всех участков центр поворота. Участки могут поворачиваться на разные углы.

Если участки не соединяются, то переход от конечной точки участка к начальной точке следующего участка осуществляется автоматически по прямой. (см. рис. 105, 106, 108).

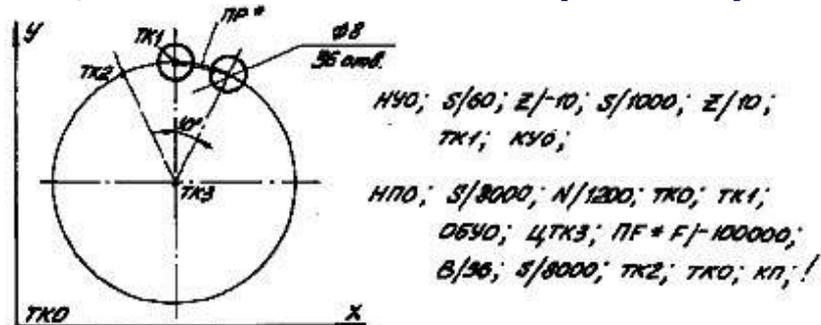


Рисунок 105

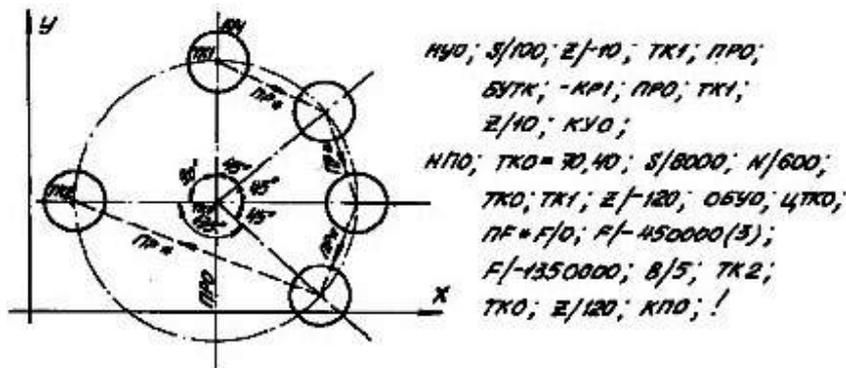


Рисунок 106

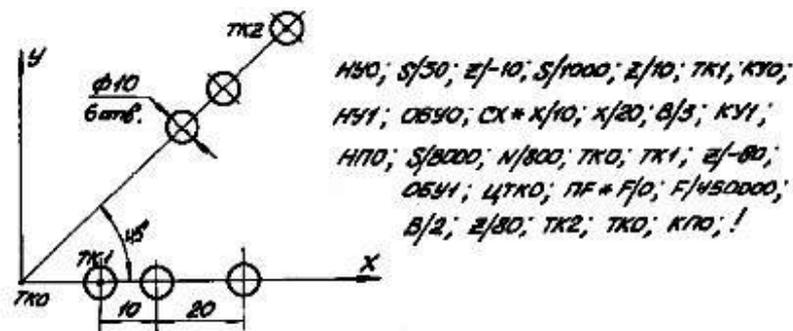


Рисунок 108

```

ОБУ<I>; ЦТК<J>; ПФ*F1; F2;...B/<A.B.>;
B<J>

```

ТК <J> - центр поворота.

F1 - угол поворота для первого участка, определяется как в предыдущих вариантах.

F2, F3, ... - угол поворота для второго, третьего участка и т.д. каждый участок, начиная со второго, получается, если предыдущий участок повернуть на соответствующий угол с центром поворота в точке ТК<J>.

8.10.4.3 ЗЕРКАЛЬНОЕ ОТОБРАЖЕНИЕ УЧАСТКА

Если геометрия участка имеет явно выраженную симметрию, то можно сократить в два раза его описание. Закодируйте половину участка N, задайте в разделе данных центр симметрии ПР1, и используйте в операторе ОБУ параметр 30/ПР1. Препроцессор САПР-ЧПУ/4 автоматически достроит вторую половину участка.

```

ОБУ<N>;30/ПР1;B/<A.B.>;
B<J>

```

Следует иметь в виду следующие правила:

1. Внутри участка можно использовать любые операторы, задающие движение инструмента, в том числе и операторы Z/ и ZA/, при этом гарантируется зеркальное отображение геометрии.

2. Внутри участка недопустимы технологические команды - ввод/сброс коррекции или фрезы, подача, обороты шпинделя должны находиться вне описания. Пример:

```

...HУ4;S/80;N/400;ОБУ7;30/ПР9;B/1;S/8000;КУ4;...

```

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

3. Прямая - центр симметрии может быть повернута на любой угол относительно оси ОХ.4. Участок может быть и замкнутым. В этом случае создается его симметричная копия. 5. Дополнительно к участку можно применить операции смещения и поворота на месте. 6. Участок можно использовать для выборки "Зиг-Заг". 7. Для исключения ошибок при зеркальной обработке участков необходимо заканчивать описание участка одним из операторов ДОРi, ДОКРi, ДОТКi.

8.10.5 ЗАМЕНА ЭЛЕМЕНТОВ (ФРАЗ) В УЧАСТКЕ

Иногда обрабатываемый контур включает в себя участки, которые не являются буквально одинаковыми, а различаются отдельными геометрическими элементами, технологическими параметрами, техкомандами. В этом случае достаточно описать участок один раз, а для обработки похожего участка использовать оператор ОБУ с заменой отдельных элементов (фраз) участка.

$\text{ОБУ}\langle I \rangle; C1 * N11; N12; \dots C2 * N21; N22; \dots B / \langle A.B \rangle;$ $B \langle J \rangle$

$C1, C2, \dots$ - идентификаторы геометрических элементов или фразы участка, которые необходимо заменить (старые).

$N11, N12, \dots, N21, N22, \dots$ - новые идентификаторы или новые фразы, заменяющие старые (разрешается использовать коэффициент повторения, который можно не указывать для последних элементов).

При первом выполнении участка старые элементы (фразы) $C1, C2, \dots$ заменяются на новые $N11, N21$. При втором выполнении участка произойдет замена тех же элементов (фраз) участка на $N12, N22, \dots$ и т.д.

Пример: $NY0; \dots P/10; \dots S/300; \dots KY0;$
 $ОБУ0; P/10 * P/10; P/8; P/6; S/300 * S/300; S/200; S/100; B/4;$

При первом выполнении участок 0 выполняется без изменений, при втором выполнении фраза $P/10$ заменяется на фразу $P/8$, фраза $S/300$ на фразу $S/200$, при третьем и четвертом выполнении фраза $P/10$ заменяется на $P/6$, фраза $S/300$ на фразу $S/100$.

8.10.6 ПРАВИЛА ЗАМЕНЫ ЭЛЕМЕНТОВ (ФРАЗ) УЧАСТКА

1. Сколько раз встречается заменяемая фраза в участке, столько раз осуществляется замена.

2. Фразы замены, записанные после $ОБУ\langle I \rangle$, действуют только на один участок I и не действуют на вложенные в него участки.

3. После выполнения участка все геометрические элементы и фразы, которые заменялись, принимают первоначальный вид бывший до первой замены.

4. Если заменяется геометрический элемент, то предлоги ОТ, ДО и признаки направления движения по окружности +, - не указываются.

Пример: $ОБУ0; KP0 * KP1; B/1;$

Вместо операторов ОТ $KP0$, $+KP0$, $-KP0$, ДО $KP0$ будут выполняться операторы ОТ $KP1$, $+KP1$, $-KP1$, ДО $KP1$.

5. Заменяющие (новые) геометрические элементы должны задаваться в системе координат участка, то есть должны "привязываться" к участку, описанному в исходной программе (см.рис. 110).

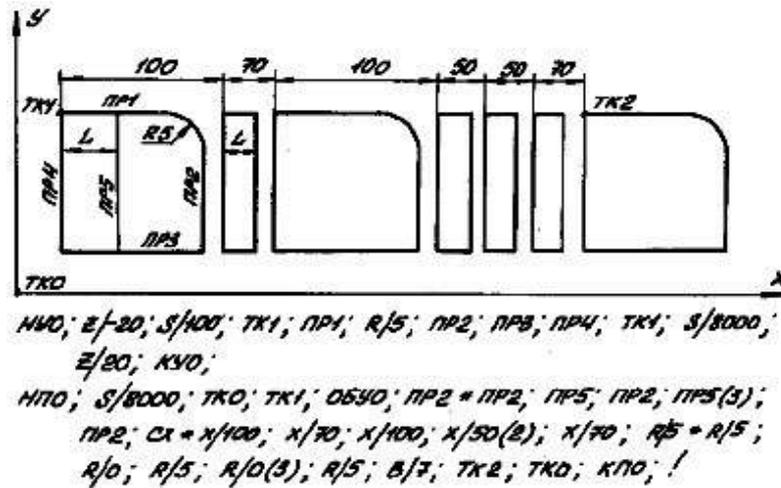


Рисунок 110

6. Нельзя заменить часть фразы, только всю фразу (исключение - замена идентификаторов в операторах ОТ ПР, ОТ КР, +КР, -КР, ДО ПР, ДО КР, ОТ ТК, ДО ТК).

Пример: NY0;... X1;...X2=X1+1;...TK=X1,X2;...KY0;OBY0;X1*X3;B/1;

Фраза X1 заменится на X3, но в арифметическом выражении $X1 + 1$ и в операторе $TK=X1, X2$ замены X1 на X3 не будет.

8.10.7 ПОЛНЫЙ ФОРМАТ ОПЕРАТОРА ОБУ

ОБУ<I>; ЦТК<J>; ПФ*F1...CX*X1...CY*Y1...C1*N11...B/<A.B.>;
B<K>

Фраза ЦТК<J>, если она есть, записывается сразу после ОБУ. Остальные конструкции записываются в любом порядке.

При повороте с заданным центром поворота (ЦТК) конструкции CX, CY не указываются.

В конструкциях CX, CY, ПФ и при замене нельзя использовать арифметические выражения.

При вложенности каждая последовательность участков может иметь свой центр поворота, свои углы поворота, свои смещения по X и по Y, свои замены. Каждый участок в программе можно описать в своей системе координат. При любом варианте смещения или поворота участок может быть незамкнутым.

8.11 ОПЕРАТОР "ФАСКА"

Оператор предназначен для построения фаски между двумя прямолинейными перемещениями.

Формат оператора: ФС (A,B); (1-й формат)
ФС (A,B); (2-й формат)
ФС (B,B); (3-й формат)

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Где А,В - идентификаторы или конструкции с арифметическим выражением
 Описание параметров: В первом формате оператора А и В задают величины отрезков, отсекаемых фаской, причем: А - первый отрезок по ходу движения, В - второй отрезок по ходу движения. Во втором и третьем формате задается отрезок и угол В (от этого отрезка к фаске).

Примеры:

```
ПРОГРАММА=TEST4;
СТАНОК=210;
А1=1; ПР0=У/0; ПР1=Х/10; ПР2=У/10; ПР3=Х/20;!
НПО; ТК=5,0; ПР0; ФС(А/1,В/2); ПР1; ФС(А1,В/-45.);
ПР2; ФС(В/3,В/10.); ПР3; ДО ПР0;КПО;!

```

8.12 ФУНКЦИЯ "РАССТОЯНИЕ МЕЖДУ ЭЛЕМЕНТАМИ"

Функция L предназначена для определения расстояния между геометрическими элементами - точками, прямыми кривыми.

Формат оператора:

```
L (TKi, TKj ); -определение расстояния от точки до точки;
L (TKi, ПРj ); -определение расстояния от точки до прямой;
L (ПРi, ПРj); определение расстояния между двумя параллельными прямыми;
L (TKi, ЦКРj); -определение расстояния от точки до центра кривой;
L (ЦКРi, ЦКРj); -определение расстояния между центрами кривых;
L (ЦКРi, ПРj ); -определение расстояния между центром кривой и прямой.

```

Функция записывает значение расстояния в идентификатор, указанный слева от знака = в следующем выражении:

```
идентификатор = L (ЦКРi,ЦКРj);

```

Примеры:

```
ПРОГРАММА=KADR;
СТАНОК=100;
ТК0=0,0; КР0=1,0,1; КР1=-1,0,1; ПР1=Х/-1; А0=0; !
НПО; S/100; ТК=0,0; Х/100; Х/100;
А0=L(ТК0,ЦКР0); ПЧ('L м/у ТК0 и центром КР0=',А0);
А0=L(ЦКР1,ЦКР0);ПЧ('L м/у центром КР1 и центром КР0=',А0);
А0=L(ЦКР0,ПР1); ПЧ('L м/у центром КР0 и прямой ПР1=',А0);
Х/100; Х/100;
КПО; !

```

8.13 Функция "ГЕОМЕТРИЧЕСКАЯ ОШИБКА"

Функция ERR предназначена для определения ошибки при непересечении геометрических элементов - прямых, кривых, либо когда точка не принадлежит прямой или окружности. Функция применяется при составлении макропроцедур для алгоритмического изменения траектории движения в случае непересечения элементов траектории.

Формат функции:

```
ERR (TKi, ПРj); - определение принадлежности точки TKi прямой ПРj;
ERR (TKi, КРj); - определение ошибки принадлежности точки TKi кривой КРj;
ERR (КРi, КРj); - определение возможности пересечения двух кривых;
ERR (КРi, ПРj); - определение возможности пересечения кривой и прямой;
ERR (ПРi, ПРj); - определение возможности пересечения двух прямых.

```

Порядок написания аргументов функции ERR – произвольный.

Функция может возвращать значение 1 (в случае ошибки) или 0 (при ее отсутствии) в идентификатор, указанный слева от знака = в следующем выражении:
идентификатор = ERR(КРi, КРj);

```
Пример:
ПРОГРАММА=P16;
СТАНОК=205;
ТК0=0,0;ТК4=10,0;ТК20=0,0; ТК21=1,0;
ПР1=X/10;ПР2=X/0;КР1=0,0,5; КР0=10,0,3;
У1=ERR(ТК0,ПР1); У2=ERR(КР1,ТК0); У3=ERR(ПР2,ПР1);
У4=ERR(ПР1,КР1); У5=ERR(КР0,КР1); У6=ERR(КР1,ПР2); !
НПО;
  S/100;ЕСЛИ ERR(ТК4,ПР1); ТО ТК20;ТК21;
      ИНАЧЕ ТК20;ТК4;
  КЕ;
КПО;
!
```

В данном примере ТК4 лежит на ПР1, а потому выполняется переход на ветвь ИНАЧЕ.

8.14 ПОВОРОТ СТОЛА, ШПИНДЕЛЬНОЙ БАБКИ И ГОЛОВКИ

Поворот стола задается следующим оператором:

F/< АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ >; F< НОМЕР >

Поворот шпиндельной бабки или головки задается следующим оператором:

Q/< АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ >; Q< НОМЕР >

Угол считается положительным при повороте против часовой стрелки и отрицательным в противном случае. Углы задаются в приращениях, а не абсолютными значениями.

Операторы ФР+, ФР- не оказывают никакого действия на операторы F, Q.

САПР – ЧПУ не учитывает значение угла поворота при расчете последующих перемещений по X, Y. Поворот должен учитываться самим технологом.

Информация о кодировании в управляющей программе углов поворота содержится в паспорте станка. Если дискретность угла поворота – несколько градусов, то возможные углы поворота задаются в паспорте техкомандами, в исходной программе используются операторы F, Q,. Если дискретность угла поворота – доли градуса, например, 0.001 градуса, то координата F описывается в паспорте аналогично X, Y, Z (см. инструкцию по паспортам п.6.4) и в исходной программе используются операторы +ФКР, -ФКР.

8.14.1 ОПЕРАТОРЫ +ФКР и -ФКР. ДВИЖЕНИЕ ПО ОКРУЖНОСТИ ЗА СЧЕТ ПОВОРОТА СТОЛА

+ ФКР <I>; - ФКР <I>;

Операторы +FKP, -FKP используются точно так же, как и операторы +KP, -KP, но обработка дуги окружности осуществляется за счет поворота стола, координаты центра инструмента в системе координат станка не изменяются.

Значение угла поворота стола рассчитывается автоматически - это угол между двумя векторами: вектором, соединяющим начальную точку эквидистанты с началом координат и вектором, соединяющим конечную точку эквидистанты с началом координат.

Все последующие перемещения по X, Y будут автоматически скорректированы с учетом угла поворота. В этом основное отличие операторов +FKP, -FKP от оператора F. При использовании операторов +FKP, -FKP исходная программа составляется обычным образом, как и без использования поворотного стола, то есть технолог должен "забыть" о поворотном столе и запрограммировать движение центра инструмента вдоль контура.

Пример программирования с использованием операторов +FKP, -FKP приведен на рис.113. В этом примере центр инструмента за всю обработку не смещается с ПР1 (если рассматривать систему координат станка). Прямые ПР1 и ПР2 могут не проходить через начало координат. В этом случае при движении по ПР2 в первом пазе центр инструмента сместится с ПР1 (имеется ввиду система координат станка), при переходе от одного паза к другому смещение будет расти, но это смещение не нужно учитывать в исходной программе, контур будет обработан правильно.

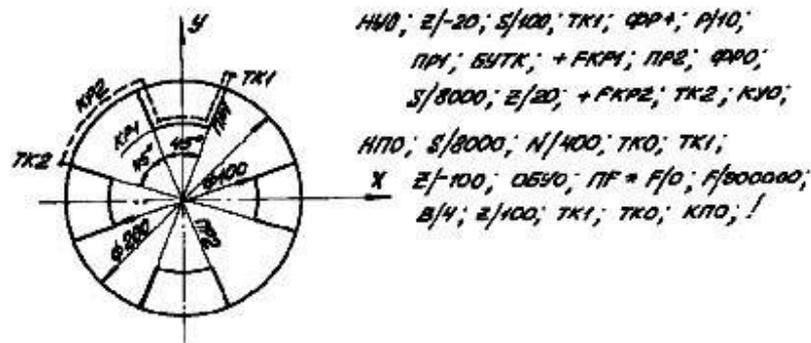


Рисунок 113

8.14.2 ОПЕРАТОР ВNF-ВОЗВРАТ В 0 ПО КООРДИНАТЕ F

ВNF;

Оператор используется для поворота стола в начальное положение. Поворот осуществляется в том же направлении, что и предшествующий поворот, если суммирующий угол поворота больше или равен 180 градусов, в противном случае поворот стола осуществляется в противоположном направлении.

Пример: НПО; ...+FKP1; +FKP2; ... ВNF; ... КПО;

Замечание: Не следует путать оператор F (поворот стола) с фразой F в конструкции ПФ (поворот участка).

8.15 ОДНОВРЕМЕННОЕ ДВИЖЕНИЕ БОЛЕЕ ЧЕМ ПО ДВУМ КООРДИНАТАМ, ДОПОЛНИТЕЛЬНЫЕ КООРДИНАТЫ A, B, C, H

Кроме ранее рассмотренных координат X, Y, Z, F, Q исходной программе можно использовать дополнительно координаты A, B, C, H которые также должны быть описаны в

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

паспорте станка (под какими адресами и в какой форме записывать в управляющую программу).

Технологический смысл этих координат для САПР-ЧПУ безразличен. Значения дополнительных координат задаются сомошью следующих операторов:

A < НОМЕР >, A/< АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ >,
 B < НОМЕР >, B/< АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ >,
 C < НОМЕР >, C/< АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ >,
 H < НОМЕР >, H/< АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ >.

Пример: ...S/100; A/200; ...C1; ...

8.16 ОПЕРАТОР СК – ДВИЖЕНИЕ ПО ПРЯМОЙ ИЛИ ОКРУЖНОСТИ С ОДНОВРЕМЕННЫМ ДВИЖЕНИЕМ ПО Z И/ ИЛИ ПОВОРОТОМ

СК; [Z/<A.B.>];	[F/<A.B.>];	[Q/<A.B.>];		
Z<I>	F<J>	Q<K>		
ZA/<A.B.>				
ZA<I>				
[A/<A.B.>];	[B/<A.B.>];	[C/<A.B.>];	[H/<A.B.>];	TK<L>;
A<M>	B<N>	C<P>	H<R>	PR<L>
				+KP<L>
				-KP<L>
				X/<A.B.>
				X<L>
				Y/<A.B.>
				Y<L>

ZA – абсолютное значение, которое достигается в конечной точке отрезка прямой или окружности. Z, F, Q, A, B, C, H приращения соответствующих величин, которые могут указываться в любой последовательности. Таким образом общее число координат, выдаваемых совместно с перемещением увеличено до 7.

Значения X, Y задаются в приращениях.

При движении до точки TK<L> или по прямой PR<L> в управляющей программе формируется один кадр, содержащий перемещения по X, Y совместно с перемещением по Z и/или поворотом на угол F, Q и/или значениями координат под адресами A, B, C, H.

При движении по окружности KP<L> окружность аппроксимируется ломаной линией, формируется N кадров (N – число хорд). Каждый кадр содержит перемещения по X, Y совместно с перемещением по Z и/или поворотом на угол. Приращения Z, F, Q разбиваются на N равных частей по числу хорд.

При автоматическом расчете эквидистанты эквидистанта считается только к прямой или окружности, к перемещению по Z эквидистанта не считается. Поворот на угол F, Q также никак не связан с автоматическим расчетом эквидистанты.

Примеры:

1. НПО; S/2000; ТК0; ТК1; СК; S/100; Z/-15; ТК2; S/50;
 СК; Z/5; -KP1; S/2000; PR0; ТК3; СК; Z/15; A/10; B/10;
 F/180.; Q/90.;

При движении от ТК1 до ТК2 инструмент одновременно опускается на 15мм, при движении по KP1 одновременно осуществляется подъем на 5мм.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

2. ... СК; F1; X2; ... СК; F/30.; Y/10; ...

8.17 ОПЕРАТОР W - ВЫДАЧА НЕСКОЛЬКИХ КООРДИНАТ В ОДНОМ КАДРЕ УПРАВЛЯЮЩЕЙ ПРОГРАММЫ

W; X; Y; Z; F; Q; A; B; C; H; КЦ; ZA;
--

X, Y, Z, ZA, F, Q, A, B, C, H - идентификаторы или конструкции с арифметическим выражением. КЦ - конец списка координат.

Пример: W; ZA/200; A/100; B2; C3; КЦ;

X, Y, Z, F, Q - приращения;
 ZA - абсолютная координата Z;
 A, B, C, H записываются в кадр без изменений.

Количество координат - от двух до восьми, координаты записываются в любом порядке. Все координаты должны быть описаны в паспорте станка.

Оператор W обрабатывается постпроцессором САПР - ЧПУ, а в процессоре при расчете траектории инструмента значения X, Y, Z, F, Q не учитываются.

Кадр кроме координат может содержать также функцию подачи, установленную перед оператором W, и техкоманды, записанные перед оператором, если они выдаются совместно с перемещением.

Если перемещения кодируются в управляющей программе в приращения и в операторе W закодировано движение по X, Y, то перед первым после оператора W перемещением по прямой или окружности необходимо записать фразу "переход;" и один из операторов ОТ ТК, ОТ ПР, ОТ КР.

Пример: ... ДО ТК1; W; X1; Y2; Z5; КЦ; ПЕРЕХОД; ОТ ТК2; ...

Оператор W может использоваться в программах для станков с двумя суппортами, при программировании объемной обработки, автоциклов сверления и т.д.

8.18 СМЕНА ПЛОСКОСТИ ОБРАБОТКИ

В исходной программе можно запрограммировать движение инструмента в любой координатной плоскости: XY, XZ, YZ. В программе можно неоднократно менять плоскость обработки с помощью следующих операторов:

ПЛ XY - плоскость XY, ПЛ XZ - плоскость XZ, ПЛ YZ - плоскость YZ.

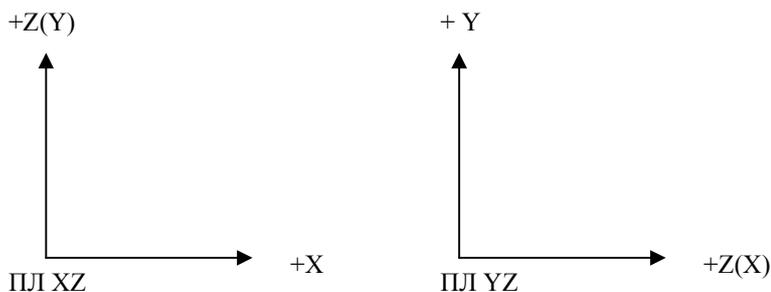
В начале процедуры до первого оператора смены плоскости, а также при отсутствии такого оператора предполагается по умолчанию плоскость XY. В исходной программе независимо от плоскости обработки программирование осуществляется в плоскости XY и только при формировании управляющей программы производится замена координат:

X<- ->Z (плоскость YZ) или Y<- ->Z (плоскость XZ).

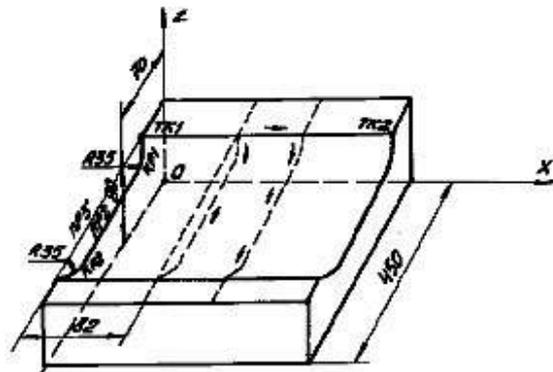
В управляющей программе автоматически записываются технологические команды из паспорта, характеризующие плоскость обработки, если такие техкоманды, указаны в паспорте (см.инструкцию по паспортам, п.6.4).

8.18.1 ПРАВИЛА ПРОГРАММИРОВАНИЯ ПРИ СМЕНЕ ПЛОСКОСТИ ОБРАБОТКИ

1. В любой плоскости можно использовать автоматический расчет эквидистанты.
 2. Оператор Z всегда означает перемещение, перпендикулярное плоскости обработки.
 3. Перед сменой плоскости движение должно быть ограничено оператором ДО ТК, ДО ПР, ДО КР, автоматический расчет эквидистанты должен быть отменен.
 4. После смены плоскости первым оператором движения должен быть оператор ОТ ТК, ОТ ПР, ОТ КР. Можно ввести автоматический расчет эквидистанты и коррекцию радиуса инструмента.
 5. Смена плоскости осуществляется в некоторой точке пространства с координатами (X, Y, Z), в разделе процедур перед сменой и после смены указываются две разные точки, у которых X и Y – это проекции точки в пространстве на координатные плоскости.
- Пример:
- Пусть в точке с координатами (100, 200, 300) осуществляется смена плоскости с XY на YZ. ТК1=100, 200; проекция точки на плоскость XY ТК2=300, 200; проекция точки на плоскость YZ ... ДО ТК1; ПЛ YZ; ОТ ТК2; ...
6. Смена плоскости производится только в основной процедуре, в макропроцедуре и в участке операторы ПЛ XY, ПЛ XZ, ПЛ YZ использовать нельзя.
 7. Чтобы правильно задавать знаки числовых величин, признаки пересечений БХТК, МХТК, БУТК, МУТК, направление движения по окружности (+КР, -КР), нужно правильно представлять направления осей координат в плоскостях XZ и YZ:



Пример программирования с использованием операторов смены плоскости приведен на рис. 114.



y Программа = ТЕСТ ПЛ0Б;
 Станок = 00030;
 * Обработка матрицы;
 ТК1 = 80,35; ТК2 = 328,35;
 КР1 = 80,70,35; КР2 = 80,380,35;
 ПР2 = X/45; ПР3 = X/80; !
 НУ0; S/100; ПЛ0Б; ТК1; -КР1; ПР2; -КР2; S/8000; БУТК;
 ПР3; ТК1; ПЛ0Б; X/82; КУ0;
 НЛ0; N/380; S/80000; ТК=0,0; ТК=0,35; Z/100; ОБУ0;
 В/4; ТК2; ТК=0,0; Z/100; КЛ0; !

Рисунок 114

8.19 ТЕХНОЛОГИЧЕСКИЕ КОМАНДЫ УПРАВЛЕНИЯ СТАНКОМ

Технологическая команда – это фраза на языке САПР – ЧПУ, которой соответствует одна или несколько команд на языке системы ЧПУ (включение охлаждения, смена инструмента и т.д.).

Техкоманда на языке САПР – ЧПУ может состоять из любых символов языка, число символов – любое, но число цифр – не более шести.

Каждая техкоманда описывается в паспорте станка (см.инструкцию по паспортам, п.12). Для каждой техкоманды в паспорте указывается запись на языке САПР – ЧПУ эквивалент на языке системы ЧПУ (последовательность символов или кадров) и режим выдачи (например, выдает совместно с перемещением или отдельным кадром).

В разных паспортах будут разные наборы техкоманд, пользователь может вводить в паспорт любые техкоманды, расширяя тем самым язык САПР – ЧПУ. Прежде чем приступить к составлению программы необходимо познакомиться с паспортом станка.

Цифровая часть техкоманды обычно бывает переменной. В паспорте в записи техкоманды на языке САПР – ЧПУ вместо цифр записываются пробелы или символ ".", число этих символов должно соответствовать числу цифр в техкоманде из исходной программы.

Если в разделе процедур встречается техкоманда, которая не описана в паспорте, выдается сообщение об ошибке.

Примеры техкоманд на языке САПР – ЧПУ:

ВКЛОХ – включить охлаждение,
 ВЫКЛОХ – выключить охлаждение,

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

ТОРМШП - останов шпинделя,
 ТЕХОСТ - технологический останов,
 СТОП<Т> - выдержка времени (Т- число секунд),
 СМ<N> - смена инструмента (N- номер инструмента),
 СМН<N> - начальная смена инструмента (самая первая),
 СМК<N> - конечная смена инструмента (самая последняя),
 +НК<N> - ввод коррекции на радиус инструмента справа,
 -НК<N> - ввод коррекции на радиус инструмента слева,
 КК<N> - отмена коррекции на радиус инструмента;

Примечание: Операторы коррекции на длину и положение инструмента (+КХ<I>, - КХ<I>, ...) также являются техкомандами и должны быть описаны в паспорте.

8.19.1 ЗАПИСЬ В РАЗДЕЛЕ ПРОЦЕДУР ИНФОРМАЦИИ НА ЯЗЫКЕ СИСТЕМЫ ЧПУ

В разделе процедур можно записать последовательность символов или кадров на языке системы ЧПУ, заключив эту последовательность в квадратные скобки. Это может быть, например, техкоманда, которая не описана в паспорте.

[СИМВОЛЫ] ИЛИ [КАДР1 КК/КАДР2 КК/ ... КАДРН]
--

КК/ - разделитель между кадрами, в конце последнего кадра не записывается.

Если в паспорте станка указано, что кадры выдаются с номером и/или контрольным числом, то номер кадра и/или контрольное число приформировываются автоматически, к каждому кадру приформировывается конец кадра.

Пример: ... НР0; S/1000; [S103M103КК/М108]; ТК1; ...

Вместо записи одного символа N раз подряд можно записать его один раз и указать затем в круглых скобках число N:

СИМВОЛ (N)

Если в кадр нужно выдать сам символ (, то вместо него записывается символ Л, вместо) записывается П.

Число символов в квадратных скобках - не больше 120.

Информация в квадратных скобках выдается в управляющую программу отдельно от перемещения, задаваемого операторами движения, но в самих скобках можно закодировать перемещение инструмента.

Траектория, закодированная в скобках, может быть незамкнутой по X, Y. В этом случае, если перемещения кодируются в управляющей программе в приращениях, после скобок перед первым движением по прямой или окружности необходимо записать фразу "переход;" и один из операторов ОТ ТК, ОТ ПР, ОТ КР.

Пример: ... ДО ТК1; [...]; ПЕРЕХОД; ОТ ТК2; ...

В квадратных скобках вместо конкретного значения числа можно записывать идентификаторы числовых данных, например:

ХХ90=100;УУ90=200; ... [X<ХХ90>Y<УУ89>Z100F4610]; ...

Запись <имя> означает, что при трансляции программы необходимо заменить символы <имя> числовым значением данного. Примечание: скобки <> идентичны @@ и являются частью формата.

Пример:

```
ZZ67=56;... FT2=4510;... [G00Z<ZZ67>F@FT2@]; ...
```

Правила по которым система САПР-ЧПУ/3 осуществляет подстановку чисел в оператор КВАДРАТНЫЕ СКОБКИ:

1. Число может быть целым и дробным, положительным и отрицательным и равняться нулю.
2. Максимальное число цифр в целой части - 6, первые нули в целой части опускаются.
3. Если модуль числа меньше 1, то ноль целых не опускается.
4. Максимальное число цифр в дробной части - 3, то есть число округляется с точностью 0.001.
5. Целая часть числа отделяется от дробной символом '.'.
6. Последние нули в дробной части опускаются. Если число целое, то вместе с дробной частью опускается точка.
7. Положительные числа записываются со знаком '+', если перед идентификатором записан символ '+'. В противном случае знак '+' опускается. Знак '-' всегда выдается с отрицательными числами.

Пример: ...P0=5.43; P1=-35; P4=35.43; ... Следующие фразы эквивалентны: [X<+P0>Y<+P1>Z<+P4>]; или [X+5.43Y-35Z+35.43];

В версии САПР-ЧПУ/4 введен дополнительный формат оператора "КВАДРАТНЫЕ СКОБКИ": [[...]]; Такой формат обеспечивает формирование кадра, к которому не присоединяется ни номер кадра, ни знак конца кадра. Сравните кадры УП в следующем примере:

Исходная программа	Кадры УП
НП9; ... [G04F0000]; ... КП9;!	N123G04F0000LF
НП9; ... [[G04F0000]]; ... КП9;!	G04F0000

8.19.2 ПРОПУСК КАДРОВ В УПРАВЛЯЮЩЕЙ ПРОГРАММЕ

В исходной программе (в процедуре) можно указать место, начиная с которого в управляющей программе перед каждым кадром будет выводиться символ /. В нужном месте процедуры можно отменить вывод этого символа. Для реализации пропуска кадра служат 2 оператора:

ВПК; - ВЫВОД СИМВОЛА / (ПРОПУСК КАДРА),
ОПК; - ОТМЕНИТЬ ВЫВОД СИМВОЛА /.

В паспорте не нужно описывать техкоманды ВПК, ОПК. Операторы можно использовать только для станков, в которых реализована команда "пропуск кадра".

8.19.3 ГЛАВНЫЙ КАДР

ГКАДР [<НОМЕР>];

В управляющей программе адрес кадра (символ N) заменяется на символ : . Если указан номер, то изменяется и номер кадра, но общая нумерация кадров в управляющей программе не изменяется.

```
ПРОГРАММА=GKADR;  
СТАНОК=100; !
```

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

```
НПО; S/100; ТК=0,0; X/100; X/100; ГКАДР 200; X/100;
X/100; КПО; !
```

Кадры управляющей программы:

```
%kad
N2 LF*
N4G01Z100000F0100 LF*
N6Z100000 LF*
:200Z100000 LF*
N8Z100000 LF*
...
```

8.19.4 ОПЕРАТОР "КАДР"

Оператор предназначен для изменения нумерации кадров в управляющей программе.

Формат оператора : КАДР N;

Описание параметров : N - число и задает новый номер кадра. Все последующие кадры будут иметь номера больше чем N. Пример:

```
ПРОГРАММА=KADR;
СТАНОК=100;!
НПО; S/100; ТК=0,0; X/100; X/100; КАДР 200; X/100;
X/100; КПО; !
```

Кадры управляющей программы:

```
%kad
N2 LF*
N4G01Z100000F0100 LF*
N6Z100000 LF*
N200Z100000 LF*
N202Z100000 LF*
...
```

8.19.5 СМЕНА ИНСТРУМЕНТА.

Поворот резцедержателя токарного станка, смена инструмента в многоинструментальном станке осуществляется с помощью оператора СМ.

По оператору СМ для разных станков будут выполняться разные функции в зависимости от описания смены инструмента в паспорте, например, поворот revolverной головки в указанную позицию, возврат инструмента из revolverной головки в указанное гнездо инструментального магазина в revolverную головку.

Вариант 1.

СМН [N];	СМ [N];	СМК [N];
----------	---------	----------

Оператор является техкомандой, описанной в паспорте станка. N - одна или несколько цифр (не больше 6), для некоторых станков может отсутствовать. N может быть номером позиции резца в резцедержателе, номером позиции инструмента в revolverной головке и/или номером гнезда в инструментальном магазине.

Для некоторых станков самая первая и самая последняя смены кодируются в управляющей программе не так, как остальные смены. В этом случае в паспорте станка, кроме техкоманды СМ<N> описываются техкоманды:

СМН<N> - начальная смена,
СМК<N> - конечная смена.

Перед сменой инструмента движение должно быть ограничено оператором ДО ТК, ДО ПР, ДО КР, автоматический расчет эквидистанты должен быть отменен. Если это не сделает технолог, САПР - ЧПУ отменит автоматический расчет эквидистанты автоматически.

После смены инструмента первым оператором движения должен быть оператор ОТ ТК, ОТ ПР, ОТ КР, можно ввести автоматический расчет эквидистанты и коррекцию.

ТК	ТК
... ДО ПР<I>; СМ<N>; ОТ ПР<J>; ...	
КР	КР

При фрезерной, сверлильной обработке точка перед командой СМ и после этой команды будет одна и та же - точка смены инструмента. Пример: ... ТК1;СМ02;ТК1; ...

При токарной обработке эти точки могут быть разными, то есть возможен разрыв траектории. Пример: ... ТК1;СМ1;ТК2;

Обычно первый вариант смены инструмента используется при фрезерной и сверлильной обработке, при токарной обработке может использоваться второй или третий варианты.

Вариант 2.

При всех сменах инструмента центр резцедержателя находится в одной и той же точке, координаты которой (ХН, УН) задаются в разделе данных по фразе НТК (см.рис.2а)

НТК=<ХН>, <УН>;

ХН, УН - числа.

Пример: НТК=390, -180;

В разделе данных с помощью фраз ИН описываются все инструменты (см.рис. 3А).

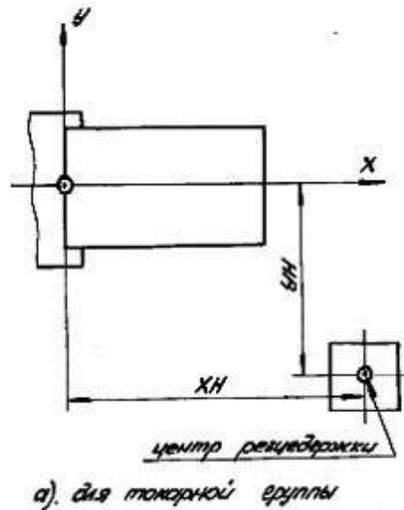


Рисунок 2а

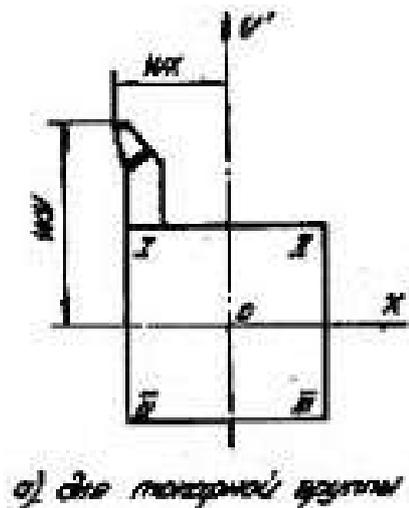


Рисунок 3а

ИН<I>=<N>, <WX>, <WY>;

I - порядковый номер инструмента, $0 \leq I \leq 35$;

N - номер позиции резца в резцедержателе, он используется для формирования техкоманды СМ<N>, число цифр - от одной до шести, определяется описанием техкоманды СМ в паспорте станка;

WX, WY - вылеты инструмента относительно центра резцедержателя (числа со знаками).

Фразы НТК, ИН можно записывать в любом месте раздела данных в любой последовательности. В разделе процедур эти фразы записывать нельзя.

При втором варианте смена инструмента задается оператором СМ<I>, где I - порядковый номер инструмента, которому соответствует фраза ИН<I> из раздела данных. По оператору СМ<I> будет сформулирована техкоманда СМ<N>, в которой цифровой код N - это код, указанный в фразе ИН<I>. Техкоманда СМ<N> должна быть описана в паспорте станка.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Пример: НТК=200,200;
 ИН1=10,-60,102;
 ИН2=20,-50,-150;
 ...
 НПО;...СМ1;...СМ2;...КПО; !

В данном примере оператору СМ1 соответствует техкоманда СМ10. Сразу после смены центр инструмента будет находиться в точке $(XН+WХ, YН+WУ)$, т. е. $(200-60, 200+102) = (140, 302)$. Где: $XН, YН$ - координаты центра резцедержателя из фразы НТК, $WХ, WУ$ - вылеты инструмента, указанные во фразе ИН1. Во втором варианте после СМ задается точка, в которую должен прийти центр инструмента, это начальная точка обработки (СМ<I>; ДО ТК...). Перемещение из точки $(XН+WХ, YН+WУ)$ в начальную точку обработки назначается САПР - ЧПУ автоматически. Перед следующей сменой автоматически назначается перемещение из конечной точки обработки (той, которая указана перед следующим оператором СМ) в точку $(XН+WХ, YН+WУ)$, то есть траектория центра инструмента замкнется, центр резцедержателя возвратится в начальную точку $(XН, YН)$.

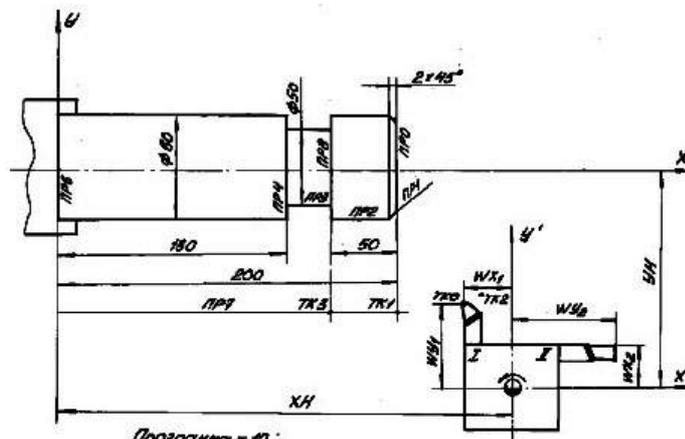
При данном варианте смены в начале процедуры до первого перемещения должен быть записан оператор СМ. Это может быть оператор СМН. В конце процедуры, чтобы замкнуть траекторию движения последнего инструмента, необходимо записать оператор СМ, это может быть оператор СМК. Для техкоманды СМК, описанной в паспорте, может отсутствовать эквивалент на языке системы ЧПУ. Пример:

НТК=360,190;
 ИН1=10,-60,110;
 ...
 НПО; S/8000; N/500; СМ1; ТК1;...ТК2; СМ2;...

По оператору СМ1 будет сформулирована техкоманда СМ10. После смены инструмента автоматически будет назначено перемещение по прямой из точки $(300, 300)$ в начальную точку обработки ТК1. После прихода в конечную точку обработки ТК2 автоматически назначается перемещение по прямой в точку $(300, 300)$.

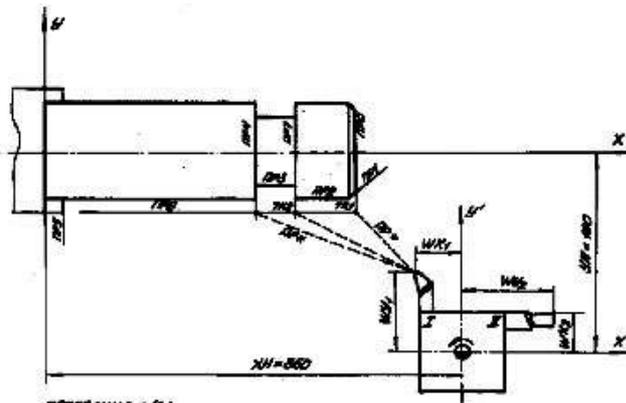
$XН+WХ=360-60=300, YН+WУ=190+110=300$.

Примеры программирования с использованием операторов смены инструмента приведены на рис. 115,116.



Программа = 10;
 Станок = 00008;
 * обработка вала - отвалом;
 ТК0 = X/200+43; Y/-132;
 ТК2 = X/200+37; Y/-124;
 ...
 ПР3 = Y/-40; !
 НПО; S/8000; N/500; СМ01; ТК0; ТК1; S/100; P/4;
 ФР-; ПР0; ПР4; ПР2; P/5; ПР6; S/8000; ФР0;
 ПР7; ТК1; ТК0; СМ02; N/320; ТК2; ТК3; S/40;
 ФР-; P/4; ПР8; P/1; ПР3; P/2; ПР4; ФР0;
 S/8000; ПР7; ТК3; ТК2; КПО; !

Рисунок 115



```

ПРОГРАММА = 9;
Станок = 20000;
* обработка вкл * способ 2;
НТК = 300, 100, 0;
НН1 = 10, -90, 10, 0, 1;
НН2 = 11, -30, 87, 0, 1;
ТК1 = 200, -85;
.....
ПРС = 5/-35; !

НП0; 5/3000; 1/300; СМН1; ТК1; ФР0; П/1; ПР0; ПР1;
ПР2; П/5; ПР3; 5/3000; ФР0; ПР0; ТК1; СМ2;
1/300; ТК2; ФР0; ФР0; П/4; ПР3; П/1; ПР3;
П/2; ПР4; ФР0; ПР5; СМКУ; КП0; !

```

Рисунок 116

Вариант 3

После каждой смены инструмента центр резцедержателя находится в новой точке, координаты которой описываются в расширенном формате оператора ИН. Оператор НТК в этом случае не используется.

```
ИН<I>=<N>, <WX>, <WY>, <XN>, <YN>;
```

I - порядковый номер инструмента, $0 \leq I \leq 35$;
 N - номер позиции резцедержателя, используется для формирования техкоманды СМ <N>;
 WX, WY - вылеты инструмента относительно центра резцедержателя, координаты которого для 1-го инструмента задаются параметрами XN, YN.
 XN, YN - координаты центра резцедержателя для 1-ой смены инструмента.

8.19.6 ОПЕРАТОР "ПЕРЕХОД"

```

ТК ТК
... ДО ПР<I>; ПЕРЕХОД; ОТ ПР<J>; ...
КР КР

```

Оператор используется при токарной обработке, если необходимо в исходной программе запрограммировать перемещение сначала для одной вершины резца, а потом для второй вершины. В управляющую программу будут записываться координаты для первой вершины. Пример:

В исходной программе:	В УП (абс. координаты):
ДО ТК=100, -40;	X 100 Y -40
ПЕРЕХОД;	
ОТ ТК=110, -20;	
ДО ТК=50, -10;	X 40 Y -30

8.19.7 ДВА ТИПА РАСЧЕТА X, Y, Z, F, Q ПРИ ФОРМИРОВАНИИ УПРАВЛЯЮЩЕЙ ПРОГРАММЫ

В управляющую программу могут записываться приращения X, Y, Z, F, Q или их абсолютные значения в системе координат станка. Тип расчета перемещений задан в паспорте станка (см.инструкцию по паспортам, п.6.1).

8.19.7.1 РАСЧЕТ ПЕРЕМЕЩЕНИЙ В ПРИРАЩЕНИЯХ

Значения Z, F, Q, заданные в исходной программе, записываются в управляющую программу без изменений. Для оператора ZA определяется соответствующее приращение по Z. При перемещении центра инструмента из точки ТК<I> в точку ТК<J> в управляющую программу записываются приращения по X и по Y: $X=X_{ТК<J>} - X_{ТК<I>}$, $Y=Y_{ТК<J>} - Y_{ТК<I>}$. Начало системы координат, в которой описывается геометрия детали, выбирается технологом произвольно.

8.19.7.2 РАСЧЕТ ПЕРЕМЕЩЕНИЙ В АБСОЛЮТНЫХ КООРДИНАТАХ

Самые первые операторы Z, F, Q в процедуре и первый оператор Z после каждой смены инструмента задают абсолютные значения и записываются в управляющую программу без изменений. Все последующие операторы Z, F, Q определяют приращения, которые автоматически суммируются с соответствующими предыдущими значениями и в управляющую программу записываются абсолютные координаты.

В исходной программе для станка с абсолютной системой координат рекомендуется вместо оператора Z использовать оператор ZA. Значение ZA записывается в кадр без изменений.

В САПР - ЧПУ имеются 4 варианта кодирования абсолютных координат X, Y.

Вариант 1. При смене инструмента фразы НТК, ИН не используются, центр системы координат - точка (0,0).

		СМ<I>	
ТК	ТК	ПЕРЕХОД;	ТК
НП<I>;	ОТ ПР ... ДО ПР	ПЛ XY;	ОТ ПР ...
КР	КР	ПЛ XZ;	КР
		ПЛ YZ;	

Координаты самой первой точки после фраз НП, СМ, ПЕРЕХОД, ПЛ XY, ПЛ XZ, ПЛ YZ в управляющую программу не записываются (координаты этих точек могут быть любыми). Координаты всех остальных точек записываются в управляющую программу без изменений.

Пример:

```

ТК0=50,50; ТК1=100,100; ТК2=150,100; ТК3=400,400;
ТК4=500,500; ... !
НП0;S/800; ОТ ТК0;ДО ТК1;ДО ТК2;...ДО ТК3;
СМ2;ОТ ТК3;ДО ТК4;

```

Значения X, Y в управляющей программе:

```

X=100, Y=100 (ДО ТК1)
X=150          (ДО ТК2)
...
X=400, Y=400 (ДО ТК3)
СМЕНА ИНСТРУМЕНТА
X=500, Y=500 (ДО ТК4)

```

Вариант 2. При смене инструмента фразы НТК, ИН не используются, центр системы координат - самая первая точка в процедуре.

```

                                CM<I>;
                                ТК      ТК      ПЕРЕХОД;      ТК
НП<I>; ПСК; ОТ ПР ... ДО ПР  ПЛ XY;      ОТ ПР ...
                                КР      КР      ПЛ XZ;      КР
                                ПЛ YZ;

```

Фраза ПСК; (перенести систему координат) означает, что при расчете абсолютных координат в управляющей программе центром системы считается не точка с координатами (0,0), а самая первая точка в процедуре при перемещении центра инструмента в точку ТК<I> в управляющую программу записываются следующие координаты X, Y.

$X = X_{ТК<I>} - X_C$, $Y = Y_{ТК<I>} - Y_C$,
где (X_C, Y_C) - центр новой системы координат (первая точка в процедуре).

Пример:

```

...ТК0=50,50; ТК1=100,100; ТК2=150,100;
ТК3=400,400; ТК4=500,500; ...!
НПО; S/800; ПСК; ОТТК0; ДОТК1; ДОТК2;...ДОТК3;
СМ2;О ТТК3; ДОТК4;...

```

Значения X, Y в управляющей программе:

```

X=50, Y=50 (ДО ТК1)
X=100 (ДО ТК2)
...
X=350, Y=350 (ДО ТК3)
СМЕНА ИНСТРУМЕНТА
X=450, Y=450 (ДО ТК4)

```

Вариант 3. При смене инструмента используются фразы НТК, ИН, в паспорте есть техкоманды СДН, ОСДН.

Техкоманды СДН (сдвиг нуля станка), ОСДН (отмена сдвига нуля станка) не обычные техкоманды, они не записываются технологом в разделе процедур, а выдаются в управляющую программу автоматически при смене инструмента вместе со следующими значениями:

$X_H + W_X$, $Y_H + W_Y$

X_H, Y_H - координаты начальной точки (НТК); W_X, W_Y - вылеты инструмента (ИН). (см.инструкцию по паспортам, п. 11.4).

В управляющую программу координаты точек, рассчитанные САПР - ЧПУ, записываются без изменений.

Пример:

```

НТК=400,-200; ИН1=10,-60,100; ИН2=11,-50,90;
...
ТК1=200,-50; ТК2=150,-50; ТК3=150,-100; ...!
НПО;S/800;СМН1;ДО ТК1;ДО ТК2;...СМ2;ДО ТК3;...

```

Значения X, Y в управляющей программе (ММ):

```

X=340, Y=-100 (кадр выдается вместе с СДН)
X=200, Y=-50 (ДО ТК1);

```

```

X=150          (ДО ТК2);
...
X=340, Y=-100 (кадр перед СМ2, выдается с ОСДН).
Смена инструмента
X=350, Y=-110 (кадр выдается вместе с СДН)
X=150, Y=-100 (ДО ТК3)

```

Вариант 4. При смене инструмента используются фразы НТК, ИН, в паспорте нет техкоманд СДН, ОСДН.

При перемещении центра инструмента в точку ТК<I> в управляющую программу записываются координаты центра резцедержателя :

```

X=XТК<I> - WX,   Y=YТК<I> - WY
WX, WY - вылеты инструмента.

```

Пример:

```

НТК=400,-200; ИН1=10,-60,100; ИН2=11,-50,90;
...
ТК1=200,-50; ТК2=150,-50; ТК3=150,-100; ... !
НПО;S/800;СМ1;ДО ТК1;ДО ТК2;...СМ2;ДО ТК3;...

```

Значения X, Y в управляющей программе (ММ) :

```

X=260, Y=-150 (ДО ТК1)
X=210          (ДО ТК2)
...
X=400, Y=-200 (кадр перед СМ2)
СМЕНА ИНСТРУМЕНТА
X=200, Y=-190; (ДО ТК3)

```

18.9.7.3 СМЕНА ТИПА РАСЧЕТА ПЕРЕМЕЩЕНИЙ В ПРОЦЕДУРЕ

Если система ЧПУ работает в приращениях, а также в абсолютных координатах, то в процедуре технолог может неоднократно менять тип расчета перемещений, указанный в паспорте. Для этого имеются два оператора:

<pre> ПРЩ; - расчет перемещений в приращениях, АБС; - расчет перемещений в абсолютных координатах. </pre>

Эти операторы одновременно являются техкомандами и должны быть описаны в паспорте станка.

```

Пример: программа: НПО;...ПРЩ;...АБС;...ПРЩ;...АБС;...КПО;
           паспорт: ПАСПОРТ=112;
                   ТРП=АБС;
                   ...
                   ТЕХКОМАНДЫ;
                   АБС=ЗН,G90;
                   ПРЩ=ЗН,G91;
                   !

```

Тип расчета перемещений на окружностях должен соответствовать типу круговой интерполяции, указанному в паспорте станка.

Примечание: Для изменения типа расчета перемещений в тексте исходной программы необходимо указать в паспорте, в команде ТРП, абсолютный тип расчета перемещений. Иначе команды АБС; и ПРЩ; не будут выполнены.

8.20 ВЫБОРКА МЕТАЛЛА В ЗОНЕ

В САПР-ЧПУ возможно автоматическое построение траектории инструмента для выборки металла в зоне, ограниченной заданным контуром. В данной версии системы имеются 3 оператора-выборки. Контур, ограничивающий зону выборки, описывается как участок: $NY<I>$; . . . $KY<I>$; Формат оператора выборки:

ВЫБП<I>; Н; Т; S; Р; R; SP; SX;ТТ; КЦ; З Э
--

ВЫБП - выборка по схеме "петля";
 ВЫБЗ - выборка по схеме "зигзаг";
 ВЫБЭ - выборка по схеме "эквидистанта";

I - номер участка, ограничивающего зону выборки.

Н, Т, S, Р, R, SP, SX, ТТ - параметры выборки (идентификаторы или конструкции с арифметическим выражением, например: Т1; Н/3;).

КЦ - фраза, ограничивающая список параметров.

Для некоторых выборок используются не все параметры. Последовательность параметров в списке - произвольная.

В зависимости от типа, параметров выборки и конфигурации контура, ограничивающего зону выборки, автоматически рассчитывается траектория инструмента внутри зоны.

8.20.1 ПРАВИЛА, ОБЩИЕ ДЛЯ ВСЕХ ВЫБОРОК

1. Перед выполнением выборки инструмент должен находиться в начальной точке участка.
2. Движение перед выборкой должно быть ограничено одним из операторов: ТК, ДО ТК, ДО ПР, ДО КР.
3. Участок должен быть замкнутым.
4. Участок должен содержать фразу ФР+ или ФР-, задающую положение центра инструмента при движении вдоль контура, но оператор Р (расстояние от центра инструмента до контура в участке не указывается. Для "эквидистантной" выборки допускается не указывать операторы ФР+;ФР-;.
5. При черновой обработке припуск для чистового прохода оставляется только для элементов участка, движение вдоль которых осуществляется с ФР+ или ФР-. При движении по элементу с ФР0 припуск не оставляется.
6. После выполнения выборки инструмент будет находиться в начальной точке участка.
7. После выполнения выборки можно использовать любой оператор движения, в том числе ДО ТК, ПР, +КР, -КР.
8. Выборки можно выполнить с поворотом на угол, если оператор выборки заключить в участок. При выполнении такого участка можно заменять параметры выборки.

Пример:

$NY1; <КОНТУР, ОГРАНИЧИВАЮЩИЙ ЗОНУ ВЫБОРКИ> KY1;$
 $NY0; ВЫБЗ1; Н/10; R/10; КЦ; KY0;$
 $НП0; \dots ОВУ0; ПГ*F/30.; R/10*R/15; В/1; \dots$

9. Если в операторе выборки одновременно не заданы (или равны нулю) параметры Н, Т, S, то выборка не выполняется.

10. Нулевое значение параметра означает, что параметр не задан.

8.20.2 ВЫБОРКА ПО СХЕМЕ " ПЕТЛЯ "

Выборка по схеме "петля" используется при обтачивании наружных и растачивании внутренних поверхностей деталей на токарных станках, а также в ряде случаев при обработке открытых пазов на фрезерных станках.

ВЫБП<I>; [Н;] [Т;] [S;] [R;] [SP;] [ТТ;] КЦ;

Н - величина с'ема металла при каждом черновом проходе (если параметр не задан, черновые проходы не осуществляются);

Т - припуск под чистовой проход (если параметр не задан, припуск не оставляется);

S - коэффициент увеличения (или уменьшения) всех подач в участке при чистовом проходе (если параметр не задан, чистового прохода не будет);

R - радиус закругления вершины резца (или радиус фрезы);

ТТ- порядковый номер геометрического элемента контура, который определяет направление проходов в выборке. Если он не задан то направление проходов в выборке определяется по последнему геометрическому элементу контура выборки;

SP- рабочая подача на черновых проходах (при незаданном параметре за рабочую принимается подача, установленная до выборки);

Подача холостого хода задается в участке перед последним перемещением.

Примеры:

1. (См.рис. 117)
 НУ0;S/100;ТК1;ПР0;ФР-;ПР3;S/60;+КР0;МХТК;ПР4;ПР5;
 ПР6;ПР1;ФР0;S/8000;ПР2;ТК1;
 КУ0;
 НПО;S/8000;N/600;+КХУ01;ТК0;ТК1;
 ВЫБП0;Н/3;Т/1;S/1.2;R/0.5;SP/60;КЦ;ТК0;
 КПО;!

Выполняются черновые проходы и чистовой проход.

2. ВЫБП0;Н/3;Т/1;R/0.5;КЦ;

Выполняются только черновые проходы, чистовой проход не выполняется, но припуск 1 мм под чистовую обработку оставляется.

3. ВЫБП0;Т/1;S/1.5;R/0.5;КЦ;

Черновые проходы не выполняются, сразу выполняется чистовой проход.

4. ВЫБП0;Н/3;R/0.5;КЦ;

Выполняются только черновые проходы, припуск под чистовую обработку не оставляется.

8.20.2.1 ТРЕБОВАНИЯ К УЧАСТКУ.

1. Последний элемент участка - прямая, задающая направление проходов инструмента при выборке зоны.
2. Перемещение по первому и последнему элементам участка осуществляется с ФР0.
3. Внутри участка операторы коррекции и оператор N (частота вращения шпинделя) действуют только на чистовом проходе.

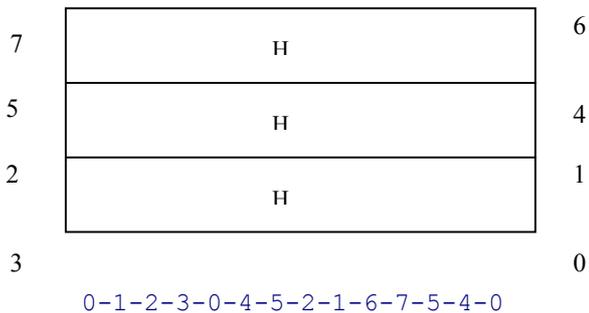
8.20.2.2 СХЕМА ДВИЖЕНИЯ ИНСТРУМЕНТА

Пример: Контур, ограничивающий зону выборки:



НУ0; ТК0; ПР1; ФР-; ПР2; ПР3; ФР0; ПР4; ТК0; КУ0;
НПО; ... ТК0; ВЫБП0; Н/3; Т/1; S/1; R/0.5; КЦ; ...

СХЕМА РАБОЧИХ ПРОХОДОВ:



1. Выборка зоны осуществляется по прямым, параллельным последнему элементу участка и отстоящим от него на величину H , $2H$, $3H$ и т.д. Будем называть эти прямые прямыми прохода. Рабочая подача на прямых прохода - SP , подача ускоренного хода - это подача в участке перед последним перемещением.

2. Переход с одной прямой прохода на другую осуществляется по контуру (ФР0) или по эквидистанте на расстоянии $T+R$ от контура (ФР+, ФР-) с подачами, заданными в участке.

3. При рабочих проходах, ближайших к элементам контура, с'ем металла может быть меньше H .

4. Перед числовым проходом инструмент возвращается в начальную точку участка по кратчайшему расстоянию, если при этом он не пересекает контур детали. В случае неизбежности соударения инструмента с деталью на данном пути возврата в начальную точку контура выборки, переход в эту точку осуществляется вдоль контура выборки. Чистовой проход осуществляется на расстоянии R от контура с подачами, заданными в участке и умноженными на коэффициент S .

8.20.3 ВЫБОРКА ПО СХЕМЕ "ЗИГЗАГ"

Выборка по схеме "зигзаг" (см.рис. 120) используется при фрезеровании карманов различной конфигурации, в том числе областей содержащих препятствия типа бобышек.

ВЫБЗ<I>; [H;] [T;] [S;] R; [P;] [SP;] [SX;] [TT]КЦ;

H - величина съема металла за один черновой проход (если параметр не указан, черновые проходы не осуществляются);

T - припуск под чистовую обработку (если параметр не указан, припуск не оставляется);

S - коэффициент увеличения (или уменьшения) подач на чистовом проходе (если параметр не задан, чистового прохода не будет);

R - радиус фрезы;

SP- рабочая подача на черновых проходах (при незаданном параметре за рабочую принимается подача, установленная до выборки);

P - величина под'ема и опускания по Z после черновой обработки при переходе в начальную точку участка (при незаданном параметре переход осуществляется без поднятия инструмента);

SX- подача ускоренного под'ема и опускания по Z при переходе в начальную точку участка после рабочих проходов (задается, если задан P);

TT- параметр задающий необходимость осуществления получистового прохода.

Пример: (см.рис. 120)

```

NY0;S/100;TK1;+HK09;P/10;PR0;DO KP0;FP+;-KP0;PR2;PR3;PR4;-KP1;
  PR1;PR2;FR0;-KP2;KK09;S/1000;PR5;TK1;
KY0;
NPO;
  S/8000;N/60;
    TK0;TK1;Z/-120;
      ВЫБЗ0;H/10;R/10;S/1.4;T/3;КЦ;...
KP0;

```

8.20.3.1 ТРЕБОВАНИЯ К УЧАСТКУ

1. Первые два элемента в участке - элементы подхода к контуру выборки, последние два элемента - элементы схода с контура. Это могут быть две прямые или прямая и окружность.

2. Третий элемент в участке - задает направление проходов инструмента в выборке. Если это окружность, то касательная в начальной точке будет определять направление проходов.

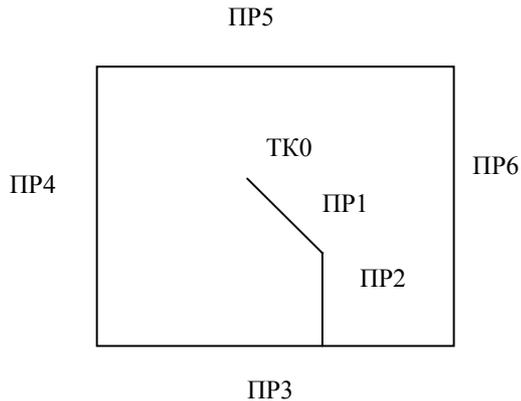
3. Врезание в металл по Z перед выборкой и под'ем по Z после выборки выполняются соответственно перед оператором "ВЫБЗ" и после этого оператора, а не в участке.

4. Частота вращения шпинделя N задается перед оператором ВЫБЗ, а не в участке.

5. В участке можно использовать операторы коррекции радиуса инструмента. Ввод и сброс коррекции задают по общим правилам. При контурной коррекции для автоматического скругления контура необходимо указать радиус инструмента с помощью оператора R. Коррекция вводится на элементах подхода к контуру, отменяется на элементах схода с контура. Операторы коррекции игнорируются на черновых и получистовом проходах и выполняются только на чистовом проходе.

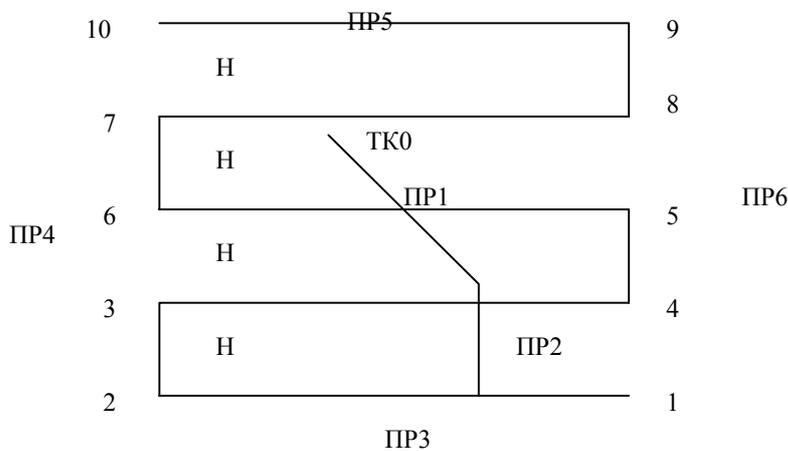
8.20.3.2 СХЕМА ДВИЖЕНИЯ ИНСТРУМЕНТА

Пример: Контур, ограничивающий выборку.



НУ0; ТК0; ПР1; ПР2; ФР+; ПР3; ПР4; ПР5; ПР6; ПР3; ФР0; ПР2;
 ПР1; ТК0; КУ0;
 НПО; ... ТК0; ВЫВ30; Н/10; R/10; КЦ; ...

СХЕМА РАБОЧИХ ПРОХОДОВ.



0-1-2-3-4-5-6-7-8-9-10-0

1. Выборка зоны осуществляется по прямым, параллельным третьему элементу участка. Первая прямая отстоит от третьего элемента на расстоянии T+R, остальные прямые отстоят друг от друга на расстоянии Н. Подача на прямых прохода - SP.

2. Переход с одной прямой на другую осуществляется по контуру (ФР0) или по эквидистанте на расстоянии T+R от контура (ФР+, ФР-) с подачами, заданными в участке.

3. В отличие от выборки по схеме "петля" холостых ходов нет, направление фрезерования меняется.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

4. При проходах, ближайший к элементам контура, с'ем металла может быть меньше Н.

5. После завершения черновых проходов выполняется под'ем инструмента на величину, задаваемую параметром Р (если параметр Р не задан, под'ем не осуществляется). Сначала под'ем производится на 1 мм с подачей SP (или 100 мм/мин, если параметр SP не задан). Затем под'ем продолжается с ускоренной подачей SX (или 1000 мм/мин, если параметр SX не задан).

6. После под'ема инструмент возвращается в начальную точку участка с подачей SX, выполняется опускание на величину Р сначала на ускоренной подаче SX (или 1000 мм/мин), потом на подаче SP (или 100 мм/мин).

7. Выполняется получистовой проход вдоль контура на расстоянии T+R от него с подачами, заданными в участке. Проход оканчивается в начальной точке участка.

8. Если параметр S задан, то выполняется чистовой проход вдоль контура на расстоянии R от него с подачами, заданными в участке и умноженными на S. На элементах подхода вводится коррекция радиуса инструмента (если она есть в участке).

9. Необходимость выполнения получистового прохода управляет параметр TT, который может принимать следующие значения:

0-получистовой проход не делать (отсутствие параметра в обращении эквивалентно 0 значению);

1-получистовой проход делать в направлении описания участка;

2-получистовой проход делать в направлении противоположном описанию участка;

8.20.3.3 ОБРАБОТКА "ГЛУБОКИХ" ЗОН

При обработке "глубоких" зон можно заключить оператор выборки в участок и выполнить его несколько раз с опусканием по Z перед каждой выборкой.

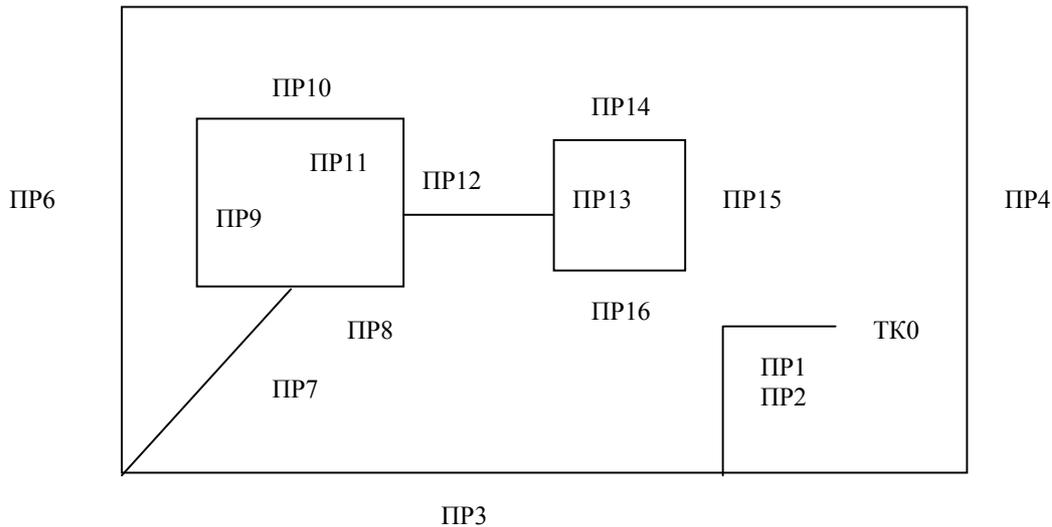
Пример: (см.рис. 122)

Выборка зоны выполняется за два раза: после первого выполнения инструмент опускается на 8 мм.

8.20.3.4 ОПИСАНИЕ КОНТУРА ПРЕПЯТСТВИЯ И ПОРЯДОК ВКЛЮЧЕНИЯ ОПИСАНИЯ ПРЕПЯТСТВИЯ В ОПИСАНИЕ КОНТУРА ВЫБОРКИ

Каждое препятствие описывается замкнутым контуром или участком. Для совместимости с предыдущими версиями системы в обращении к выборке указывается только один участок описывающий внешний контур детали. Контур препятствия можно включить в описание этого участка с помощью "ниточки", которая соединяет начальную точку контура препятствия с выбранной точкой внешнего контура детали. Например:

ПР5



НУ0;

*подход к обрабатываемому контуру;	ТК0; ПР1; ПР2; ФР-;
*внешний контур;	ПР3; ПР4; ПР5; ПР6;
*первая ниточка;	ФР0; ПР7; ФР-;
*начало описания первого препятствия;	ПР8; ПР9; ПР10; ПР11;
*вторая ниточка;	ФР0; ПР12; ФР-;
*описание второго препятствия;	ПР13; ПР14; ПР15; ПР16; ПР13;
*вторая ниточка;	ФР0; ПР12; ФР-;
*конец описания первого препятствия;	ПР11; ПР8;
*первая ниточка;	ФР0; ПР7; ФР-;
*внешний контур;	ПР3;
*отход от обрабатываемого контура;	ФР0; ПР2; ПР1; ТК0;

КУ0;

Задание направления контура препятствий следует выбирать таким, чтобы фреза была с той же стороны, что и на внешнем контуре. Соединение препятствия "ниточкой" с основным контуром не должно осуществляться через точку излома контура (препятствия). Соединяемые контура в точке касания "ниточки" должны быть гладкими.

8.20.4 ВЫБОРКА ПО СХЕМЕ "ЭКВИДИСТАНТА"

Выборка по схеме "эквидистанта" используется при фрезеровании карманов различной конфигурации, в том числе областей, содержащих препятствия типа бобышек, а также при торцевании поверхностей.

ВЫВЭ<I>; [H;] [T;] [S;] R; [P;] [SP;] [SX;] КЦ;

H - величина с'ема металла за один черновой проход-виток (если параметр не указан, черновые проходы не осуществляются);
T - припуск под чистовую обработку (если параметр не указан, припуск не оставляется);

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

7-8-9-10-11-7- второй виток
 12-13-14-15-12- третий виток
 2-1-0 возврат к двум элементам отхода

Первый рабочий проход выборки-"виток", строится эквидистантно к внешнему контуру участка (участок без элементов подхода и отхода) внутрь обрабатываемой зоны. Расстояние от первого витка до контура равно $T+N$, где T -припуск на чистовую обработку, N -величина съема металла за один черновой проход.

Каждый последующий "виток" строится внутри предыдущего. Расстояние между двумя очередными витками равно H .

Переход с "витка" на "виток" осуществляется по прямой соединяющей начальную точку контура наружного "витка" к начальной точке контура внутреннего "витка".

После выполнения самого внутреннего (последнего) "витка" инструмент переводится в начальную точку контура отхода инструмента, для возврата по ним в начальную точку контура участка. Переход осуществляется на высоте плоскости безопасности задаваемой параметром P . При отказе от чистовой обработки переход осуществляется сразу в начальную точку участка. При образовании нескольких внутренних "витков" внутри внешнего сначала обрабатывается первый из них, а затем осуществляется переход в плоскости безопасности к началу следующего. Причем, как и в "зигзаге", опускание из плоскости безопасности осуществляется на свободное от металла место с последующим подходом к началу "витка" на рабочей подаче.

8.21 НАРЕЗАНИЕ РЕЗЬБЫ РЕЗЦОМ

Для нарезания цилиндрических резьб на внутренних и наружных поверхностях на токарных станках используется оператор "нарез" (см. рис. 125, 126).

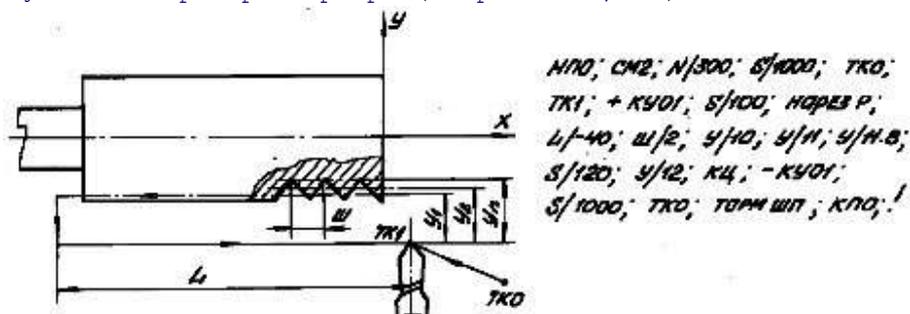


Рисунок 125

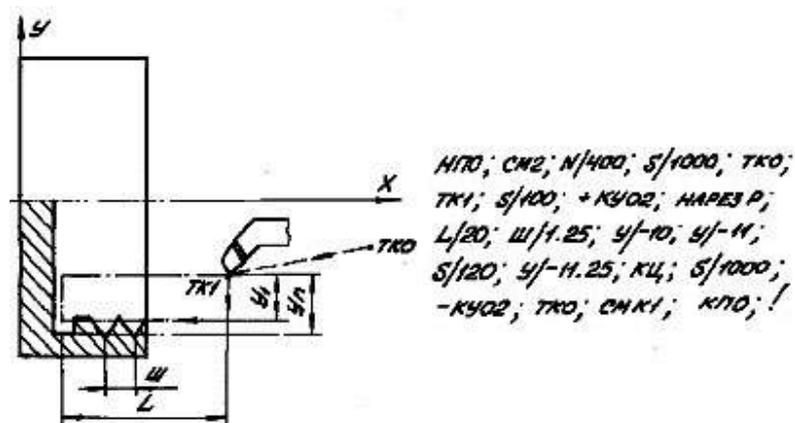


Рисунок 126

НАРЕЗР; L; Ш; [S1;] Y1; [S2;] Y2; ... КЦ;

L, Ш, S1, Y1, ... - параметры цикла, идентификаторы или конструкции с арифметическими выражениями, например, L1; Y/10.
 L - длина резьбы с учетом выхода резца (положительное или отрицательное число);
 Ш - шаг резьбы (положительное число);
 Y1, Y2, ... - величина первого, второго (и т.д.) прохода (положительные или отрицательные числа);
 S1, S2, ... - подача первого, второго (и т.д.) прохода. Параметры S могут отсутствовать или записываться не перед всеми Y;
 КЦ - конец списка параметров.

Коррекция и частота вращения шпинделя назначаются перед циклом резбонарезания.

В паспорте станка должны быть записаны сведения о кодировании цикла резбонарезания в управляющей программе (см. инструкцию по паспортам, п.9).

8.22 ЦИКЛЫ ОБРАБОТКИ ОТВЕРСТИЙ

Под циклом обработки отверстий понимается одна из следующих операций: центрование, сверление, зенкерование, развертывание, растачивание, нарезание резьбы метчиком. В САПР - ЧПУ имеются шесть операторов, реализующих данные операции.

СВЕРЛ (L [;ZA] [;S] [;A] [;B] [;C] [;H]); Z ЦСВЕРЛ ЗЕНКЕР РАСТОЧ НАРЕЗМ РАЗВЕРТ

СВЕРЛ - цикл сверления;
 ЦСВЕРЛ - цикл центрования;
 ЗЕНКЕР - цикл зенкерования;
 РАСТОЧ - цикл растачивания;
 НАРЕЗМ - цикл нарезания резьбы метчиком;
 РАЗВЕРТ - цикл развертывания.

L, ZA (или Z), S, A, B, C, H - параметры цикла, идентификаторы или конструкции с арифметическими выражениями. Параметры записываются в любом порядке, обязательным параметром является только L, остальные могут быть опущены.

В паспорте станка должно быть описано кодирование автоциклов обработки отверстий (см. инструкцию по паспортам, п.9.3), причем обязательно использование оператора ЦОО.

Параметры автоциклов:

ZA или Z - быстрый подход инструмента к рабочей поверхности детали с учетом недохода, задается абсолютной координатой (ZA) или приращением (Z) со знаком -. Параметр не указывается, если инструмент уже находится в рабочей позиции или в паспорте станка вообще нет информации о кодировании быстрого подхода.

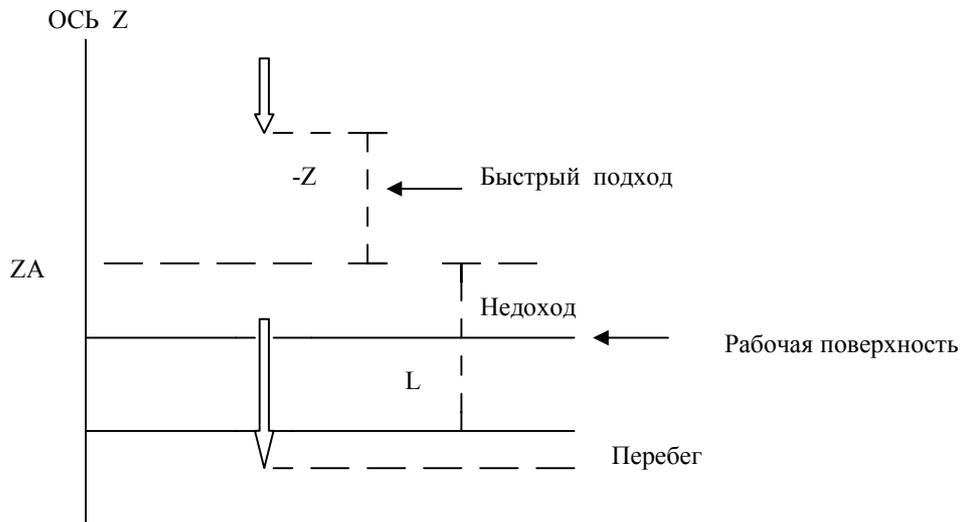
L - величина рабочего хода (L= глубина отверстия + недоход + перебег).

S - рабочая подача, если параметр не указан, действительна подача, указанная ранее.

A, B, C, H - технологические параметры автоцикла (выдержка времени, глубина сверления за один проход и количество проходов при глубоком сверлении, координата выхода после сверления на безопасное расстояние и т.д.). В паспорте станка должны

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

быть записаны операторы ПА,ПВ,ПС,Н, в шаблоне кадра должны быть описаны координаты А,В,С,Н (см.инструкцию по паспортам, п.6.4, 10.4).



В паспорте станка должны быть описаны техкоманды СВЕРЛ, ЦСВЕРЛ, ЗЕНКЕР, и т.д., задающие функции автоциклов, например, G81, G82 и т.д.

Если есть функция "конец цикла" (G80), то в паспорте описывается соответствующая техкоманда, например, КЦ. Если окончания разных автоциклов кодируются различными функциями G, то в паспорте описываются несколько команд, например, КЦС - конец цикла сверления, КЦР - конец цикла растачивания и т.д.

Цикл может обрабатываться в одной точке или на группе точек. В первом случае команда "конец цикла" записывается сразу после оператора цикла, во втором случае после оператора автоцикла записываются операторы обхода группы точек, а затем команда "конец цикла".

Примеры:

1. Сверление в одной точке, быстрого подхода нет:

```
...ТК1; СВЕРЛ(L/20;S/200); КЦ; ...
```

2. Сверление в 10 точках, расположенных на прямой, параллельной оси X, расстояние между точками 10мм. К первой точке осуществляется быстрый подход:

```
СВЕРЛ(ZA/-100;L/15);ЦИКЛ I1=0,1,10;X/10;КОНЕЦ;КЦ;
```

Переход от точки к точке в группе точек можно задать с помощью любых операторов САПР - ЧПУ, реализующих такое перемещение, например, X, Y, ДО ТК; ОБУ, МАКРО и другие.

```
Пример: ...ТК1;СВЕРЛ(...);МАКРО/<ИМЯ>(...);КЦ;...
```

В САПР - ЧПУ с помощью одного оператора (вызов макропроцедуры SWRM) можно запрограммировать сверление группы точек, принадлежащих одной прямой, образующих матрицу (решетку) с исключением из обработки отдельных точек(см.п. 7.29 данной инструкции).

Если оператор обработки отверстия используется в участке, то в соответствующем операторе ОБУ можно указать замену параметров L, Z (ZA), S, A, B, C, H.

Если в системе ЧПУ станка возможно использование подпрограмм, то обращение на подпрограмму сверления можно записать с помощью оператора "Вызов станочной подпрограммы" (см. п. 8.35 данной инструкции).

8.23 ДВИЖЕНИЕ ПО ТОЧЕЧНОЗАДАННОЙ КРИВОЙ

Движение по кривой, заданной группой точек, определяется оператором ТКР (см. рис. 132).

ТКР (J; X1, Y1; X2, Y2; ... XN, YN);

J - точность аппроксимации кривой ломаной линией, J<НОМЕР> или J/<А.В.>;
 X1, Y1 - координаты первой точки;
 X2, Y2 - координаты второй точки и т.д.
 X, Y - числа, идентификаторы или конструкции ХТК<НОМЕР>, УТК<НОМЕР> (число точек - не менее четырех).

САПР - ЧПУ строит кривую, плавно соединяющую заданные точки. Первая точка со второй и предпоследняя точка с последней соединяются параболой, остальные точки соединяются кривыми, которые описываются уравнениями третьей степени (кубические полиномы). Полученные кривые аппроксимируются ломаной линией с заданной точностью (J).

При движении по точечнозаданной кривой можно использовать автоматический расчет эквидистанты и коррекцию радиуса инструмента. Перемещение по прямой или окружности на контуре перед точечнозаданной кривой должно быть ограничено первой точкой кривой.

Пример: ...ДО ТК=10,10;ТКР (J/0.05;10,10;30,40;...

После оператора ТКР можно использовать любой оператор движения.

8.23.1 ДВИЖЕНИЕ ПО ТОЧЕЧНО ЗАДАННОЙ КРИВОЙ (ЛИНЕЙНО-КРУГОВАЯ ИНТЕРПОЛЯЦИЯ)

Данная возможность реализована в системе САПР-ЧПУ/4 при помощи дополнительного модуля LKI.EXE и определяет движение по кривой, заданной группой точек, в виде автоматически генерируемого множества гладко сопряженных кругов и линий. При этом достигается значительное сокращение длины УП и повышается чистота обработки. Модуль LKI.EXE является дополнительным и не входит в базовый комплект САПР-ЧПУ.

Движение по точечно заданной кривой с линейно-круговой интерполяцией осуществляется при помощи следующего оператора: #ТКР (параметры);

Параметры используемые при вызове оператора:

1. Обязательные параметры:

K - количество точек;

один из вариантов задания точек :

- | | | | |
|----|---------|---|-----------------|
| а) | X1-X199 | вариант задания точек перечислением Y1-Y199 | координат X и Y |
| б) | X1,N | вариант задания точек массивом из N Y1,N | координат X и Y |
| в) | TK1,N | вариант задания точек массивом из N точек | |
| г) | R1-R199 | вариант задания точек перечислением | |
| | B1-B199 | полярных координат R и угла B | |

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

- д) R1,N вариант задания точек массивом из N
 B1,N полярных координат R и угла B

2. Необязательные параметры :

RI - радиус инструмента со знаком:

>0 - фреза справа от контура (ФР+);

<0 - фреза слева от контура (ФР-);

отсутствие RI (по умолчанию) предполагает движение прямо по контуру;

СК - тип системы координат: 1-полярная, 2-декартовая (по умолчанию);

RMAX - максимальный обрабатываемый радиус на данном станке (умолчание RMAX=10000 мм);

J - стрелка прогиба - допустимое отклонение автоматически генерируемого контура от заданного (по умолчанию J=0.01);

NI - начальный индекс элемента точки ТКР (умолчание NI=1);

ПС - Указание сгладить все вогнутые "ямки" на генерируемом контуре радиусом RI.

Значения:

ПС=0 - не сглаживать (по умолчанию),

ПС=1 - сгладить "ямки";

Параметр RI является необязательным, он включает контроль возможности выполнения дуг контура указанным радиусом инструмента. Этот параметр не вызывает каких-либо изменений в формируемом контуре. Если такой контроль не нужен то этот параметр не указывается.

Параметр ПС указывается в том случае, если требуется сгладить радиусом не менее RI все вогнутые круги ("ямки") на генерируемом контуре. Последние являются продуктом алгоритма модуля LKI при попытках плавно соединить два круга большого радиуса, выдержать поля допуска генерируемого контура, а также избежать появления линейного перемещения (G01). С другой стороны появление "ямок" пагубно влияет на траекторию движения фрезы при ее радиусе, близком или большем радиуса "ямки". В этом случае станки сбоят или намертво встают, обнаружив нулевой радиус дуги. Как избавиться от этого?

Задайте RI - максимальный радиус инструмента, которым можно обработать такую деталь в цехе, а также ПС=1. Все ямки будут иметь гарантированный радиус RI. Однако, следует понимать, что предискажение траектории обработки "ямок" может привести к отклонению построенного контура от поля допуска, что потребует принятия какого-то компромисса.

8.23.2 Способы задания параметров в #ТКР

Возможны пять вариантов обращения к оператору с соответствующим набором параметров.

1 вариант: Используется декартовая система координат. В обращении к оператору задаются координаты N-точек последовательным перечислением (x, y).

```
#ТКР ( RMAX=<A.B.>;
      J=<A.B.>;
      СК=2; *2-декартовая система;
      RI=<A.B.>; K=<A.B.>;
      X1=<A.B.>; Y1=<A.B.>;
      ...
      XN=<A.B.>; YN=<A.B.>;
```

2 вариант: Используется декартовая система координат. В обращении к оператору задаются координаты N точек. Указывается начальный элемент массива, содержащего абсциссы точек (X1) и число точек (N). Аналогичной фразой описываются координаты точек (Y1, N;)

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

```
#ТКР( RMAX=<A.B.>;
      J=<A.B.>;
      СК=2;           *2-декартова система;
      RI=<A.B.>;
      K=<A.B.>;
      X1,N;           *массив абсцисс точек;
      Y1,N;           *массив ординат точек;
```

3 вариант: Используется декартова система координат. В обращении к оператору задаются N точек. Указывается начальный элемент массива, содержащего точки (TK1) и число точек (N).

```
#ТКР( RMAX=<A.B.>;
      J=<A.B.>;
      СК=2;           *2-декартова система;
      RI=<A.B.>;
      K=<A.B.>;
      TK1,N);
```

4 вариант: Используется полярная система координат. В обращении к оператору задаются координаты N точек последовательным перечислением (R_i, B_i) , где R_i – радиус – вектор очередной точки, B_i – угол, образованный радиус – вектором и положительным направлением оси X.

```
#ТКР( RMAX=<A.B.>;
      J=<A.B.>;
      RI=<A.B.>;
      K=<A.B.>;
      СК=1;           *1-полярная система;
      R1=<A.B.>;       B1=<A.B.>;
      R2=<A.B.>;       B2=<A.B.>;
      ...
      RN=<A.B.>;       BN=<A.B.>);
```

5 вариант: Используется полярная система координат. В обращении к оператору задаются координаты N точек. Указывается начальный элемент массива, содержащего расстояния до точек (R1) и число точек (N). Аналогичной фразой описываются углы, образованные радиус – векторами точек с центром системы координат $(Y1, N)$.

```
#ТКР( RMAX=<A.B.>;
      J=<A.B.>;
      RI=<A.B.>;
      K=<A.B.>;
      СК=1;           *1-полярная система ;
      R1,N;           *расстояния до точек ;
      Y1,N);         *углы, образованные тк;
```

8.23.3 Пример использования #ТКР

САПР – ЧПУ строит контур, плавно соединяющий заданные точки. Первая и предпоследняя из заданных точек гарантированно совпадают с начальной и конечной точкой контура, остальные лежат в пределах заданного поля допуска.

При движении по точечно-заданной кривой с линейно-круговой интерполяцией можно использовать автоматический расчет эквидистанты и контурную коррекцию на радиус инструмента.

Пример программы.

```

ПРОГРАММА=КУЛАК7;          СТАНОК=401;
R0=120.0;                   B0=270.0;    R1=120.0;           B1=225.0;
R2=120.0;                   B2=160.0;    R3=119.675;        B3=153.0;
R4=117.570;                 B4=146.0;    R5=112.570;        B5=139.0;
R6=104.675;                 B6=132.0;    R7=95.0;           B7=125.0;
R8=85.325;                  B8=118.0;    R9=77.430;         B9=111.0;
R10=72.430;                 B10=104.0;   R11=70.325;        B11=97.0;
R12=70.0;                   B12=90.0;    R13=70.325;        B13=83.0;
R14=72.430;                 B14=76.0;    R15=77.430;        B15=69.0;
R16=85.325;                 B16=62.0;    R17=95.0;          B17=55.0;
R18=104.675;                B18=48.0;    R19=112.570;       B19=41.0;
R20=117.570;                B20=34.0;    R21=119.675;       B21=27.0;

R22=120.0;                  B22=20.0;    R23=120.0;         B23=10.0;
R24=120.0;                  B24=270.0;

ТК0=0,0;
!
НПО;S/1500; ТК=-130,-130;ТК=-130,130;ТК=130,130;
ТК=130,-130;ТК=0,-130;ТК=0,-120.0;S/140;
#ТКР (
    RMAX=10000.0; *Максимальный обраб-мый радиус на данном
                  станке;
    J=0.01;       *Стрелка прогиба;
    СК=1;         *1-полярная система координат;
                  *2-декартовая система координат;
    RI=-30.0;     *Радиус инструмента;
                  *Фреза слева при обходе контура;
    K=25;         *Число точек;
    NI=0;         *начальный индекс первого элемента ТКР;
    R0,25;        *расстояния до точек контура в полярной
                  сист.коор.;
    B0,25);       *углы опред.точки на контуре в полярной
                  сист.коор.;
ТК=0,-120.0; S/1500;ТК=0.0,-130;ТК=130,-130;
КПО;
!
```

8.23.4 Плавный заход и выход на/с #ТКР

Перемещение по прямой или окружности на контуре перед то- чечной заданной кривой должно быть ограничено первой точкой кривой, а при необходимости скругления контура с #ТКР после точки указывается оператор DR. В этой ситуации оператор R/ не поможет, так как скругляет только два текущих элемента. Оператор DR скругляет заданным радиусом элемент подвода с автоматически построенными кругами. Такая возможность недоступна технологу какими либо другими средствами.

Пример: ...ДО ТК=10,10; DRF/5;
#ТКР(СК=2; K=30; X1=10.0; Y1=10.0;...

В операторе #ТКР движение по контуру ограничивается оператором движения в точку, поэтому после оператора #ТКР можно использовать любой оператор движения допустимый после оператора ТК=..., а при необходимости скругления контура на выходе с #ТКР перед оператором движения по контуру указывается оператор DR или оператор R/.

8.24 ОПЕРАТОР ПРИСВАИВАНИЯ

8.24.1. Арифметический оператор присваивания

<ИДЕНТИФИКАТОР>=<АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ>;

Данному, идентификатор которого записан слева от '=' присваивается значение арифметического выражения, записанного справа от '='. Примеры: Z1=100; P1=D:2; Запись Z1=100; не означает под'ема инструмента на 100мм, для этого необходимо записать оператор Z1 или Z/100. Также запись P1=D:2; не устанавливает новое расстояние от центра инструмента до контура, для этого нужно записать оператор P1. Эти операторы устанавливают новые числовые значения данных Z1 и P1. Знак = (присвоить) не является математическим знаком равенства.

Например, запись I1=I1+1; означает, что значение данного I1 увеличивается на 1.

С помощью арифметического оператора присваивания можно изменять отдельные координаты точек, прямых, окружностей, записывая слева от = следующие идентификаторы: ХТК<I>, УТК<I>, ХКР<I>, УКР<I>, РКР<I>, СПР<I>, РПР<I> (I - номер геометрического элемента). Примеры: РКР2=0; ХТК1=ХТК1+10;

8.24.2. Геометрический оператор присваивания

ТК ПР<НОМЕР>=<СПОСОБ ЗАДАНИЯ>; КР

С помощью этого оператора определяется геометрический элемент, но не движение по прямой или окружности. Примеры: ТК1=10,20; ПР2=ТК1,ТК2; Данное, идентификатор которого записан слева от =, не обязательно должно быть определено в разделе данных. Достаточно чтобы в разделе данных было задано данное того же типа с номером, который больше номера определяемого данного. Пример: пусть в разделе данных определена только одна точка ТК99=0,0; тогда в разделе процедурно определить любую точку с номером от 0 до 99 включительно.

Таким образом, с помощью оператора присваивания можно многократно изменять значение одного и того же данного (переопределять геометрические элементы не заданные в разделе данных).

8.24.3 ПРИМЕНЕНИЕ ОПЕРАТОРА ПРИСВАИВАНИЯ

Оператор присваивания может использоваться вместо замены данных при обработке участков (оператор ОБУ), когда замену использовать нельзя.

Пример: В разделе данных: Z1=10; X1=30; B1=10;
В разделе процедур: НУ0;S/100;Z/-Z1;S/1000;КУ0;
НУ1;ОБУ0;СХ*Х1;В1;...КУ1;
НП0; ...ОБУ1;В/1;...Z1=20;X1=-50;B1=15;ОБУ1;В/1; ...

При первом выполнении участка 1 имеем сверление 10 отверстий на глубину 10 мм на расстоянии 30 мм друг от друга. При втором выполнении участка 1 имеем сверление 15 отверстий глубиной 20 мм, расположенных на расстоянии 50 мм друг от друга. Примечание: Без особой необходимости не следует использовать в разделе процедур геометрический оператор присваивания, так как это увеличивает время трансляции программы на ЭВМ.

8.24.4 ДОПОЛНИТЕЛЬНЫЙ ФОРМАТ ОПЕРАТОРОВ ТК, ПР, +КР, -КР

ТК ПР=< СПОСОБ ЗАДАНИЯ >; +КР -КР
--

Данные операторы определяют движение от точки, до точки, по прямой, по окружности. Эти операторы отличаются от геометрических операторов присваивания тем, что идентификатор геометрического элемента в операторе присваивания должен иметь номер (номер 0 можно опускать только в идентификаторах числовых данных).

Пример: НПО; ТК=10, 10; ПР1; ПР=X/10; +КР=ВХПР2, МУПР3, R/100;

Сравните с операторами присваивания:

ТК0=10, 10; ПР0=X/10; КР0=ВХПР2, МУПР3, R/100;

Дополнительный формат операторов движения используется обычно в макропроцедурах, при программировании об'емной обработки.

Примечание: Не рекомендуется пользоваться данным форматом без особой необходимости, так как использование такого формата увеличивает время трансляции на ЭВМ.

8.25 УПРАВЛЕНИЕ ПОСЛЕДОВАТЕЛЬНОСТЬЮ ВЫПОЛНЕНИЯ
ОПЕРАТОРОВ РАЗДЕЛА ПРОЦЕДУР

Операторы в процедуре обычно выполняются последовательно друг за другом. Из рассмотренных операторов только ОБУ прерывает прямую последовательность выполнения операторов.

В данном параграфе приводятся операторы, управляющие последовательностью выполнения операторов. Это операторы условного и безусловного перехода, операторы выбора, операторы циклов. Данные операторы применяются в макропроцедурах, при программировании об'емной обработки.

Эти операторы не вызывают записи в управляющую программу какойлибо информации, они задают условия, при которых какая-то группа операторов будет выполняться или не выполняться, какая-то группа операторов будет выполняться несколько раз подряд.

8.25.1 МЕТКИ

Любой оператор в процедуре может быть помечен меткой, которая записывается перед оператором.

М<НОМЕР>;

Номер - число от 0 до 499.

Метки нужны при использовании операторов условного и безусловного перехода.
Пример: НПО; ...M1; ТК1; ...

8.25.2 ОПЕРАТОР УСЛОВНОГО ПЕРЕХОДА

ЕСЛИ <УСЛОВИЕ>; ИДИ М<НОМЕР>;

При выполнении оператора проверяется выполняется или нет условие. Если условие выполняется, то обработка будет продолжена с оператора, перед которым

записана указанная метка, в противном случае обработка продолжается с оператора, записанного после фразы ИДИ... .

Пример: ЕСЛИ R0=0; ИДИМ1;

В инструкции используются пробелы для наглядности, в распечатки исходных программ пробелы вставляются автоматически.

Условие в операторе ЕСЛИ может быть простым или составным. Простое условие состоит из двух арифметических выражений, между которыми записан знак сравнения:

> - больше,
 < - меньше,
 = - равно,
 # - не равно,
 >= - больше или равно,
 <= - меньше или равно.

Из-за ограничения точности представления чисел в ЭВМ (в САПР - ЧПУ - не более 9 десятичных цифр) не всегда можно сравнивать числа на строгое равенство, например, значение выражения (1:3)*3 не равно 1. Поэтому, если возможно, нужно использовать сравнение "больше или равно", "меньше или равно".

Составное условие состоит из нескольких простых условий, соединенных союзами И, ИЛИ, которые заключаются в апострофы или кавычки (пробелы не разрешаются).

Пример: ЕСЛИ A>B 'И' C<D; ИДИ M1;

Если в условии есть оба союза одновременно, то союз И имеет приоритет перед союзом ИЛИ, то есть сначала проверяются условия, соединяемые союзом И, а потом условия, соединяемые союзом ИЛИ.

Для изменения последовательности сравнений и в сомнительных случаях в составных условиях используются круглые скобки.

Пример: Следующие две записи эквивалентны:

1. ЕСЛИ D1=D2 'ИЛИ' ХТК2<ХТК1 'И' УТК2<УТК1; ИДИ M3;
2. ЕСЛИ (D1=D2) 'ИЛИ' (ХТК2<ХТК1 'И' УТК2<УТК1); ИДИ M3;

Сравните с такой записью:

ЕСЛИ (D1=D2) 'ИЛИ' ХТК2<ХТК1) 'И' (УТК2<УТК1); ИДИ M3; Это будет уже другое условие.

Пример программирования - движение по спирали Архимеда (см.рис. 130).

Параметрические уравнения спирали:

$$X(F) = (R_H + \frac{R_K - R_H}{F_K - F_H} * (F - F_H)) * \cos(F),$$

$$Y(F) = (R_H + \frac{R_K - R_H}{F_K - F_H} * (F - F_H)) * \sin(F)$$

R_H - начальный радиус;

F_H - начальный угол.

R_H, F_H - полярные координаты начальной точки спирали;

R_K - конечный радиус;

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

FK - конечный угол,
 RK, FK - полярные координаты конечной точки спирали;
 F - параметр кривой - угол, который изменяется от FH до FK с некоторым заданным шагом.

Для каждого угла F рассчитываются координаты X, Y очередной точки спирали, спираль аппроксимируется ломаной линией.

```

ПРИМЕР = PRIMER1;
СТАНОК = 205;
*СПИРАЛЬ АРХИМЕДА;
TK1=R/60.,B/30.; KP1=0,0,60; TK2=R/70,B/-120.; KP2=0,0,70;
RH=60;          FH=F/30.;  RK=70;          FK=F/-120.;
K=(RK-RH):(FK-FH);    A=0;          F0=29.;
!
НПО; ...-KP1;TK1;
M0; A=RH+K*(F0-FH);
ХТК1=A*COS(F0);УТК1=A*SIN(F0);TK1;
F0=F0-F/1.;ЕСЛИ F0>FK;ИДИ M0;
TK2;-KP2;
...
КПО;!
  
```

Спираль аппроксимируется с шагом -1 градус (угол убывает от 30 градусов до -120 градусов). Последовательность операторов $A=RH+K*(F-FH)$; $XTK1=A*\cos(F)$; $YTK1=A*\sin(F)$; $TK1$; $F=F-F/1.$; будет выполняться до тех пор, пока $F > -120$ градусов, каждый раз осуществляется перемещение до точки $TK1$ с новыми координатами.

Так как углы в радианах - не целые числа, то точное равенство $F=FK$ может не выполняться, то есть приращение угла F при перемещении от предпоследней точки к последней может быть меньше 1 градуса. Поэтому перемещение в последнюю точку спирали ($TK2$) осуществляется вне цикла.

Данные K и A введены, чтобы упростить вид уравнений и уменьшить время трансляции программы. Вместо одного оператора $TK=X/A*\cos(F)$, $Y/A*\sin(F)$; используются три оператора $XTK1=A*\cos(F)$; $YTK1=A*\sin(F)$; $TK1$; также, чтобы уменьшить время трансляции на ЭВМ.

8.25.3 ОПЕРАТОР БЕЗУСЛОВНОГО ПЕРЕХОДА

ИЛИ M<НОМЕР>;

Оператор вызывает прерывание прямой последовательности операторов без всякого условия, обработка продолжается с оператора, помеченного меткой.

Последовательность операторов, начинающихся с метки, может находиться не только в процедуре или в участке, но и вне их, но в этом случае такая последовательность должна оканчиваться безусловным переходом обратно в процедуру или в участок.

Пример:

```
НПО; ... ИДИ M1; ... M1; ... КПО;
```

8.25.4 ОПЕРАТОР ВЫБОРА

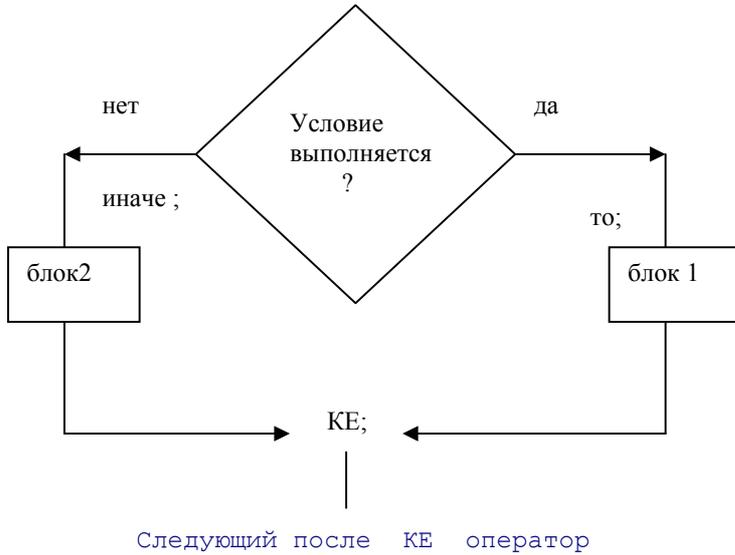
ЕСЛИ <УСЛОВИЕ>; ТО [;] <БЛОК1>; ИНАЧЕ [;] <БЛОК2>; КЕ;

БЛОК1, БЛОК2 - последовательности любых операторов (блок может состоять из одного оператора).

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Если условие выполняется, то выполняются операторы, составляющие БЛОК 1, затем операторы, записанные после фразы КЕ, БЛОК 2 пропускается. Если условие не выполняется, то БЛОК 1 пропускается, выполняются операторы, составляющие БЛОК 2, а затем обработка продолжается с оператора, записанного после КЕ. Символ ';' (точка с запятой) после фраз 'ТО' и 'ИНАЧЕ' необязателен.

СХЕМА РАБОТЫ ОПЕРАТОРА:



Пример:

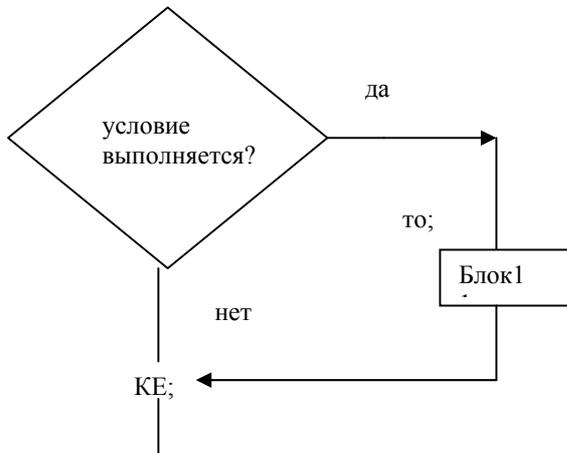
ЕСЛИ R1#0; ТО; +КР1; ИНАЧЕ ПР1; КЕ;
 При R1#0 осуществляется движение по КР1,
 при R1=0 осуществляется движение по ПР1.

БЛОК 2 вместе с фразой ИНАЧЕ может отсутствовать.

```
ЕСЛИ <УСЛОВИЕ>; ТО[;] <БЛОК1>; КЕ;
```

Если условие выполняется, то операторы блока будут обработаны, в противном случае они пропускаются. Символ ';' (точка с запятой) после фразы 'ТО' необязателен.

СХЕМА РАБОТЫ ОПЕРАТОРА:



Следующий после КЕ оператор

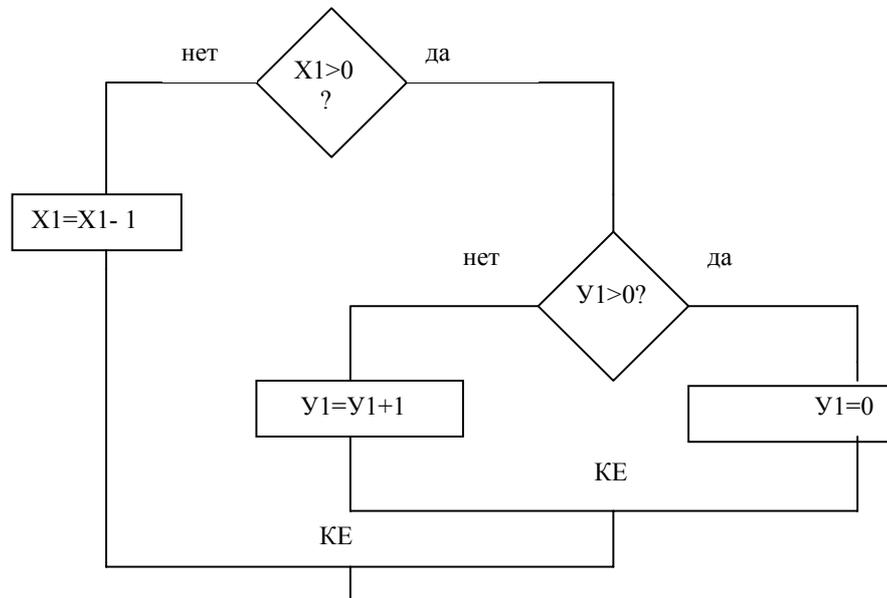
Пример: ЕСЛИ $X>Y$; ТО; $X=A$; $Y=B$; КЕ;

Если $X>Y$, то данным X , Y будут присвоены новые значения.
Если $X\leq Y$, то данные X , Y не изменяются.

Блоки 1, 2 могут содержать операторы выбора, вложенность операторов - любая. Каждой фразе ЕСЛИ должна соответствовать фраза КЕ. В случае вложенности операторов выбора каждая фраза КЕ соответствует ближайшей предшествующей "свободной" фразе ЕСЛИ. Пример:

```
ЕСЛИ  $X1>0$ ;
ТО: ЕСЛИ  $Y1>0$ ;
ТО  $Y1=0$ ;
ИНАЧЕ  $Y1=Y1+1$ ;
КЕ;
ИНАЧЕ;  $X1=X1-1$ ;
КЕ;
```

БЛОК - СХЕМА:



В отличие от системы САП-СМ4 теперь можно выйти за пределы блока с помощью операторов безусловного или условного перехода. Однако, нельзя войти в блок с помощью этих операторов, т.е. минуя оператор ЕСЛИ.

Примеры правильного программирования:

```
ЕСЛИ ...; ТО; ...ЕСЛИ  $A>0$ ;ИДИ М1; ...КЕ; ...М1;
ЕСЛИ  $A>0$ ; ИДИ М1;
```

Общее число операторов ЕСЛИ...КЕ; не должно превышать 200.

8.25.5 ЦИКЛЫ

Циклом называется последовательность операторов, которая может выполняться более одного раза подряд. В САПР-ЧПУ имеются два оператора цикла.

1. Цикл с указанным числом выполнений.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

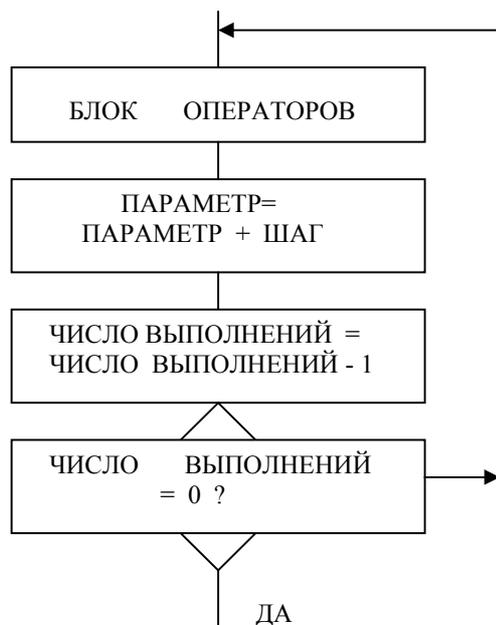
```

ЦИКЛ[;] <ИМЯ>=<НАЧ.ЗНАЧ.>,<ШАГ>,<ЧИСЛО ВЫП.>;
<БЛОК ОПЕРАТОРОВ>
КОНЕЦ;

```

ИМЯ - идентификатор числового данного - параметр цикла; НАЧ.ЗНАЧ.- начальное значение параметра цикла (число, идентификатор или арифметическое выражение); ШАГ - шаг изменения параметра (число, идентификатор или арифметическое выражение, значение может быть целым или дробным, положительным или отрицательным); ЧИСЛО ВЫП.-число выполнений цикла (число, идентификатор или арифметическое выражение, значение - целое положительное число).

Первый раз блок операторов, составляющий цикл, выполняется при параметре, равном начальному значению. После каждого выполнения цикла параметр автоматически увеличивается на шаг (или уменьшается, если шаг < 0), из числа выполнений цикла автоматически вычитается единица. Если число выполнений не равно нулю, цикл повторяется сначала. Если число выполнений равно нулю (цикл выполнен указанное число раз), то обработка продолжается с оператора, записанного после фразы "конец". Если с самого начала число выполнений цикла равно нулю или меньше нуля, то цикл вообще не выполняется. Блок операторов, составляющий цикл, состоит из любых операторов, включая вложенные операторы циклов. При вложенности циклов фраза "конец" относится к ближайшей предшествующей "свободной" фразе "цикл". Схема работы цикла:



Следующий после "конец" оператор

В отличие от системы САП-СМ4 теперь можно выйти за пределы цикла с помощью оператора условного или безусловного перехода. Однако, нельзя из процедуры войти в цикл, минуя фразу "цикл".

Второй вариант обработки спирали Архимеда.

```

ПРОГРАММА=PRIMFR2;
СТАНОК=205;

```

```
*СПИРАЛЬ АРХИМЕДА;
```

```
*НАЧАЛО РАЗДЕЛА ДАННЫХ ТАКОЕ ЖЕ, ЧТО И ДЛЯ PRIMER1;
ШАГF=(FK-FH):150; F0=FH+ШАГF;
```

```
НП0; ...-КР1;ТК1;
```

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

```

ЦИКЛ; F0=F0,ШАГF,150;

A=RH+K*(F0-FH);
ХТК1=A*COS(F0);YTK1=A*SIN(F0);
TK1;

КОНЕЦ;
-КР2; ...КП0;!
    
```

Так как углы в радианах - не целые числа, то равенство (-120 градусов - 30 градусов) : 150 = -1 градус может не выполняться, поэтому перед выполнением цикла приращение для параметра цикла F вычисляется:

ШАГF=(FK-FH):150 (-1 градус или число очень близкое). Цикл выполняется 150 раз F=29 градусов, 28 градусов и т. д.

2. Цикл с условием.

```

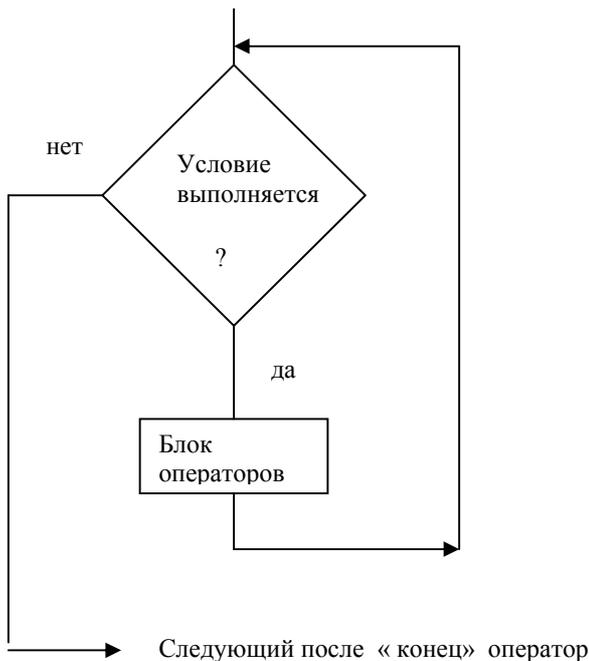
                ЦИКЛ; ПОКА
<УСЛОВИЕ>
    <БЛОК ОПЕРАТОРОВ>
                КОНЕЦ;
    
```

Условие - простое или составное условие как в операторе условного перехода или выбора.

Цикл повторяется до тех пор, пока выполняется условие. Условие проверяется перед каждым выполнением цикла, поэтому такой цикл может вообще не выполняться, если условие сразу же не выполняется.

Данные, идентификаторы которых используются в записи условия, должны изменяться при выполнении цикла так, чтобы цикл мог окончиться. В цикле с условием фраза "ЦИКЛ;" может быть опущена.

СХЕМА РАБОТЫ ЦИКЛА:



Третий вариант обработки спирали Архимеда.

```

ПРОГРАММА=PRIMER3;
СТАНОК=205;
*СПИРАЛЬ АРХИМЕДА;
*РАЗДЕЛ ДАННЫХ ТАКОЙ ЖЕ, ЧТО И ДЛЯ PRIMER1;

НПО; ...-КР1;ТК1;
ЦИКЛ; ПОКА F0>FK;

A=RH+K*(F0-FH);
ХТК1=A*COS(F0);УТК1=A*SIN(F0);
ТК1;F0-F/1.;

КОНЕЦ;
ТК2;-КР2; ...КПО;!

```

8.26 ОПЕРАТОРЫ "ВЫХОД", "ПОВТОР"

Оператор "ВЫХОД" используется для немедленного выхода из цикла. Оператор может стоять в любом месте в теле цикла. В результате следующей выполняемой конструкцией будет фраза, стоящая после оператора КЦ; или КОНЕЦ; данного цикла. Оператор "ПОВТОР" вызывает немедленный переход на оператор ЦИКЛ. Условия применения оператора аналогичны сказанному выше.

Пример:

```

ПРОГРАММА=ТЕСТСК;
СТАНОК=205;
...
НПО; ...ЦИКЛ; ПОКА L1<9;
        ЕСЛИ L1=2;ТО ТК=20,0; L1=L1+1; ПОВТОР;
        ИНАЧЕ ТК=10,0; ВЫХОД; ...
КПО;
!
```

8.27 ОПЕРАТОР "ДИАЛОГ"

Оператор предназначен для организации диалога между технологом и макропроцедурой или исходной программой. Предполагается, что из макропроцедуры задается какой-либо вопрос, а ответ на это оператор – технолог может ввести с клавиатуры одно или несколько чисел. Полученные числа, запоминаются в идентификаторах макропроцедуры и доступны для последующего использования.

Формат оператора : ВВОД ('текст',идент1,идент2,...,идентN);

Описание параметров :

текст - строка символов, которая выдается на экран терминала (обычно вопрос);
идент1 - идентификатор в который будет помещено первое число, введенное пользователем;
идент2 - идентификатор в который будет помещено второе число, введенное пользователем;
идентN - идентификатор в который будет помещено N-ое число, введенное пользователем.

Числа, введенные пользователем, должны разделяться пробелом. САПР-ЧПУ следит за тем, чтобы пользователь ввел ровно столько чисел, сколько идентификаторов указано в операторе ВВОД. Примеры: ... ВВОД('Введите X Y и угол F :',X0,Y0,F0);

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Результат работы системы САПР-ЧПУ:
 Введите два числа X Y и угол F : 1 -56.9.90.
 X= 1.000000 Y= -56.900000 Z=+90.00.00

8.28 МАССИВ ДАННЫХ, ИНДЕКСИРОВАНИЕ ЭЛЕМЕНТОВ МАССИВА

Массивом называется группа данных одного типа.

Примеры: X0, X1, X2, ... - массив X;
 TK0, TK1, TK2, ... - массив точек.

Имя данного можно рассматривать как комбинацию имени массива и порядкового номера этого данного в массиве. САПР - ЧПУ упорядочивает данные в массивы по типам и в соответствии с номерами независимо от последовательности фраз в разделе данных.

Размерность массива, то есть число элементов в массиве, на 1 больше наибольшего номера элемента этого массива, определенного в разделе данных.

Пример: если в разделе данных есть фраза TK99=0,0; и нет точки с номером, больше 99, то тем самым определен массив из 100 точек.

В разделе данных не обязательно задавать все элементы массива с нулевого номера по конечный, некоторые элементы могут быть не заданы. Эти элементы можно определить в разделе процедур.

Число различных массивов в программе вместе с числовыми данными, идентификаторы которых не содержат номер, не должно превышать 15. В версии САПР-ЧПУ/4 максимальное число элементов в одном массиве не ограничено. Пользователь сам должен следить за переполнением внутреннего ОЗУ Препроцессора. Без особой необходимости не нужно использовать массивы большой размерности, так как это может привести к прекращению обработки программы на ЭВМ из-за недостатка памяти (выдается сообщение "не хватает памяти под канонические параметры").

В разделе процедур можно ссылаться на элементы массива, не указывая конкретных номеров, в вычисляя их во время выполнения исходной программы на ЭВМ. Для этого вместо записи <ИМЯ МАССИВА><НОМЕР> используется следующая запись.

<ИМЯ МАССИВА> [НОМЕР ИНДЕКСА]

Квадратные скобки в данном случае являются частью формата.

Индекс - это числовое данное с именем I с номером 0, 1, 2, ..., 199, то есть 10, 11, 12, ..., I199. Значение индекса - целое положительное число. Индекс можно задавать и вычислять как обычное числовое данное: I1=0; I3=I3+1; I10=(N+M):2; и т.д.

Если вместо номера элемента в квадратных скобках указан номер индекса, то значение этого индекса и будет номером элемента.

Пример: Пусть I1=2, тогда запись D[1] указывает на элемент D2.

Индекс в квадратных скобках можно записывать и после идентификатора с номером:

<ИМЯ МАССИВА><НОМЕР> [НОМЕР ИНДЕКСА]

В этом случае сумма номера перед квадратными скобками и значения индекса является номером элемента.

Пример 1: Пусть I1=2, тогда запись D1[1] указывает на элемент D3.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Пример 2:

```
Программа=SWERL;
Станок=301;
*Сверление 10 отверстий, расположенных на плоскости в произвольном порядке.В
паспорте описан автоцикл сверления (функции G81, G80, координата быстрого подхода
R);
TK1=10,20; TK2=15,40; TK3=60,40; TK4=50,30; TK5=30,20; TK6=10,20; TK7=0,-10; TK8=30,-15;TK9=50,-
20;TK10=60,-20;I1=0; ! НПО;N/900;СВЕРЛ(Z/-100;L/20;S/100);ЦИКЛ;I1=0,1,10;
TK1[I];КОНЕЦ;КЦ;S/1000;TK0;КПО;!
```

После каждого выполнения цикла TK1[I]; индекс I1 (параметр цикла) автоматически получает приращение +1, цикл повторяется:

```
I1=0   TK1;   - первое выполнение,
I1=1   TK2;   - второе выполнение,
...
I1=9   TK10;  - десятое выполнение.
```

Индексирование можно использовать для замены данных при обработке участков.

```
Пример: NY0;...R=SQRT(X[1]**2+Y[1]**2);...I1=I1+1;KY0;
НПО;...I1=0;ОБУ0;B/10;...
```

При первом выполнении участка: R=SQRT(X0**2+Y0**2);

При втором выполнении участка: R=SQRT(X1**2+Y1**2);

8.29 ИНДЕКСИРОВАНИЕ МЕТКИ В ОПЕРАТОРАХ ПЕРЕХОДА, УЧАСТКА В ОПЕРАТОРЕ ОБУ, ИНСТРУМЕНТА В ОПЕРАТОРЕ СМЕНЫ ИНСТРУМЕНТА

```
ИДИ М<НОМЕР>[НОМЕР ИНДЕКСА];
ИДИ М[НОМЕР ИНДЕКСА];
```

Метка, на которую осуществляется переход, определяется во время выполнения программы на ЭВМ.

Пример: пусть I1=10, тогда оператор ИДИ М[1]; вызывает переход к оператору, записанному после метки M10.

```
ОБУ<НОМЕР УЧАСТКА> [НОМЕР ИНДЕКСА];
ОБУ[НОМЕР ИНДЕКСА];
```

Номер участка определяется во время выполнения программы на ЭВМ. Пример: Пусть I1=3, тогда оператор ОБУ[1];B/1; будет означать обработку участка 3.

```
СМ<НОМЕР>[НОМЕР ИНДЕКСА];
СМ[НОМЕР ИНДЕКСА];
```

Оператор СМ можно индексировать только в том случае, если в разделе данных есть фразы ИН (фраза НТК может отсутствовать). Пример: В разделе данных: ИН2=1020,0,0; В разделе процедур: НПО;...I1=2; ...СМ[1];... Оператор СМ[1]; ПРИ I1=2 эквивалентен техкоманде СМ1020.

Кроме смены инструмента никакие другие техкоманды, а также операторы коррекции индексировать нельзя (можно использовать подстановку символьной строки, см.п. 8.30.2).

8.30 ПОДСТАНОВКА СИМВОЛЬНОЙ СТРОКИ

Кроме геометрических элементов и числовых данных в исходной программе могут определяться символьные строки. Символьные строки используются обычно в макропроцедурах для подстановки номера корректора, номера инструмента в операторах коррекции и смены инструмента (см.п. 7.27 данной инструкции).

8.30.1 Задание символьной строки

Символьная строка задается в разделе данных или в операторе вызова макропроцедуры с помощью следующей фразы:

```
<ИДЕНТИФИКАТОР>='<СТРОКА>';
```

Примеры: НК='01'; Ф='SIN';

Следующим идентификаторам нельзя присваивать символьные строки: X, Y, Z, ZA, R, P, J, I, B, F, Q, S, N, SN, V, H, T, L, Ш, SP, SX, A, B, C, H. Этим идентификаторам могут присваиваться только числовые значения. Все остальные идентификаторы могут содержать как числовые, так и символьные данные.

Строка записывается между двух апострофов, которые строке не принадлежат. Строка может состоять из любых символов языка САПР - ЧПУ кроме двух символов: ; и ! апостроф также может принадлежать строке. Вместо знака ' (апостроф) можно использовать " (кавычки). Число символов в строке не должно превышать 120. Пример: EFG12='ПУСТО';

8.30.2 Подстановка символьной строки

В разделе процедур программы или макропроцедуры вместо всей фразы или части фразы может быть указан идентификатор символьной строки, заключенный между двух символов @.

```
@<ИДЕНТИФИКАТОР>@
```

Перед выполнением оператора, в записи которого использована данная конструкция, вместо символов @...@ происходит подстановка указанной символьной строки, только после этого оператор выполняется. Второй символ @ после идентификатора можно не записывать, если следующий символ - разделитель (запятая, точка с запятой) или знак операции, то есть явно не может принадлежать идентификатору. Примеры:

1. В разделе данных: НК1='01';
В разделе процедур: +KX@НК1;
После подстановки: +KX01;
2. В разделе данных: YC='A>B'; ПРС='X[2]=SIN(B[1])';
В разделе процедур: ЕСЛИ @YC; ТО; @ПРС; КЕ;
После подстановки: ЕСЛИ A>B; ТО; X[2]=SIN(B[1]);КЕ;

После выполнения оператора фраза принимает первоначальный вид, то есть подстановка осуществляется перед каждым выполнением оператора. Одна фраза раздела процедур может содержать несколько подстановок. В результате подстановки должна получаться только одна фраза, количество символов в которой - не больше 120.

8.30.3. Массив символьных строк, индексирование СИМВОЛЬНЫХ СТРОК

Символьные строки как и числовые данные могут образовывать массивы. Идентификаторы строк одного массива отличаются друг от друга только номерами. Строки в массиве могут иметь разную длину.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Примеры: НК1='01'; НК2='04'; ...НИ5='20';НИ6='4020';...

Один массив не может содержать одновременно и числа и символьные строки. Тип массива (числовые данные или символьные строки) определяется по заданию первого элемента массива.

Примеры: НОМ='201'; НОМ1=2; - ошибка, так как массив НОМ это массив символьных строк.

При подстановке идентификатор символьной строки можно записать с индексом:

@<ИДЕНТИФИКАТОР>[НОМЕР ИНДЕКСА]@

При неоднократном выполнении оператора и изменении индекса будут подставляться разные символьные строки, в результате на одном месте исходной программы будут получаться разные операторы.

Примеры:

- В разделе данных: НОМ1='01'; НОМ2='02';
В разделе процедур: ...+НК@НОМ[1]; ...
После подстановки: ...+НК01;... (ЕСЛИ I1=1)
...+НК02;... (ЕСЛИ I1=2)
- В разделе данных: Ф1='SIN'; Ф2='COS';
В разделе процедур: X1=@Ф[1] (B2);
После подстановки: X1=SIN (B2); (ЕСЛИ I1=1)
X2=COS (B2); (ЕСЛИ I1=2)

8.30.4 Перевод целых положительных чисел в символьные строки и подстановка

Если после символа @ указан идентификатор числа, то число перед подстановкой преобразуется в символьную константу без знака +, без нулей в начале числа и без десятичной точки. Число должно быть целым положительным или равно нулю.

Пример: ...НОМ=2;НОМ=НОМ+1;+НК@НОМ;... В результате подстановки: +НК3;

8.30.5 Присваивание символьных строк в разделе процедур

Формат : ИДЕНТ<I>=ИДЕНТ<J>; или ИДЕНТ<I>='строка'; . При этом следует учесть, что длина присваиваемой строки не должна превышать максимальной строки в массиве или длины одиночного идентификатора.

Пример: в разделе данных: STR1='123456'; STR2='66';
в разделе процедур: STR1=STR2; - правильно
STR2='123456789' - не правильно

8.30.6 Сравнение символьных строк

В разделе процедур в логических выражениях можно использовать сравнение строковых переменных с помощью операций '=' и '#'.
Пример: ЕСЛИ STR2='66'; ТО STR1='FF';
ЕСЛИ STR1#STR2; ТО ПЧ ('ОШИБКА В ДАННЫХ');

8.30.7 Оператор ПЭС

Для копирования строки (или ее элементов) в другую строку используется оператор Переслать Элементы Строки (ПЭС).

Формат: ПЭС (ИДЕНТ_i[j] [k], ИДЕНТ_m[n] [p], КОЛ);

, где:

ИДЕНТ_i—идентификатор строки в которую копируется строка; ИДЕНТ_m—идентификатор строки из которой копируется строка;

j и n —индекс в массиве строк;

k и p —индекс внутри одной строки;

КОЛ —количество пересылаемых символов (идентификатор или арифметическое выражение).

Пример: ПРОГРАММА=TESTSTR;
 СТАНОК=205;
 STR1='1000'; STR2='0200'; STR3='0030'; STR4='0004';
 IMJA=' '; I1=0; I2=0; I3=1; I4=0; I10=0;!
 НПО; S/100; ЦИКЛ I1=0, 1, 4;
 ПЭС (IMJA[4] [2], STR1[1] [1], 1); КОНЕЦ; КПО;!

В данном примере в строку IMJA копируется первый символ строки STR1, затем второй символ строки STR2, третий символ строки STR2 и четвертый символ строки STR4.

8.31 ОПЕРАТОР ПЕЧАТИ

Оператор печати предназначен для печати текстовой и числовой информации на различные устройства (принтер или экран), а также в пользовательский файл.

Замечание. По умолчанию вывод информации происходит на экран ПЭВМ в окно "Дополнительная печать". В случае вывода на принтер готовность последнего не контролируется.

ПЧ/<ИДЕНТИФИКАТОР>, <ИДЕНТИФИКАТОР>, ...; 'СИМВОЛЫ' 'СИМВОЛЫ'
--

В одном операторе можно указать распечатку нескольких символьных строк (закljučаются в апострофы или кавычки) и нескольких числовых данных.

Для распечатки координат точек, прямых и окружностей используются идентификаторы: ХТК<I>, УТК<I>, СПР<I>, SPP<I>, PPR<I>, ХКР<I>, УКР<I>, РКР<I>.

Пример 1: ПЧ/'СПИРАЛЬ АРХИМЕДА'; На печать будут выданы символы, заключенные в апострофы.

Пример 2: ПЧ/'X1=', X1, 'TK1=', ХТК1, УТК1; Пусть в момент выполнения оператора X1=20.5, TK1=100,200, тогда на печать будет выдана строка:

X1= +20.500000 TK1= +100.000000,+200.000000

При распечатке числовых данных по умолчанию под каждое число отводится 14 колонок: колонка под знак, 6 колонок под целую часть, колонка под десятичную точку, 6 колонок под дробную часть. Можно указать другое количество колонок под целую и дробную части, если после идентификатора в круглых скобках записать две цифры:

ПЧ/...<ИДЕНТИФИКАТОР>(NM), ...

N — число колонок под целую часть (N=1, 2, ... 7);

M — число колонок под дробную часть (M=0, 1, 2... 7). M, равное 0, можно не указывать. Если M не указано или равно нулю, то число печатается без знака и десятичной точки.

Пример: ПЧ/'X', I1 (2), '=', X[1] (32);

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Пусть в момент выполнения оператора $I1=30$, $X30=-290.578\dots$, тогда будет выведена строка: $X30=-290.58$

С помощью оператора ПЧ можно проверить выполнение оператора условного перехода или оператора выбора, если записать в операторе арифметического присваивания справа от = условие: <ИДЕНТИФИКАТОР>=<УСЛОВИЕ>;

Примеры: $A=X1<Y1$; $V=(X1=Y0)$;

Если условие выполняется, то данное, идентификатор которого записан слева от =, получит значение 0 (истина), в противном случае 1 (ложь). Это данное можно распечатать как обычное числовое данное.

Пример: $A=X1<Y1$; ПЧ/'X1<Y1 ',A; Пусть в момент выполнения оператора $X1=11$, $Y1=10$, тогда на печать будет выведена строка: $X1<Y1$ 1 .

Примечание: В символьных константах, выводимых на печать, могут использоваться пробелы, символы ;, ,, апострофы, кавычки, но нельзя использовать символ !, любую комбинацию из двух символов, являющуюся признаком конца константы: ', ", ', " .

По умолчанию вывод информации происходит на экран ПЭВМ, если же необходимо изменить направление вывода информации, то включите в начало программы один из следующих вариантов оператора "Назначение устройства для вывода":

ПЧ (PRN); - назначить вывод информации на принтер;
 ПЧ (имя файла); - назначить вывод информации в файл;
 ПЧ (CON); - назначить вывод информации в окно "Дополнительная печать" стандартных размеров на экран дисплея;

В последнем варианте оператора предусмотрена возможность изменения стандартных размеров окна "Дополнительная печать" при выводе информации на экран дисплея. Система САПР-ЧПУ автоматически помещает все данные, выводимые на экран в данное окно, в том числе обеспечивая прокрутку.

Пример: Первый оператор ПЧ(); не содержит изменения размеров окна "Дополнительной печати". Следовательно, все последующие операторы ПЧ будут выводить данные в окно стандартных размеров.

НПО; ... ПЧ/'I0=',I0; ...

Внимание: В случае отладочной печати все данные по оператору ПЧ/ или ПЧ() выводятся либо на экран, либо на принтер. В последнем случае размеры окна "Дополнительной печати" игнорируются.

8.32 МАКРОПРОЦЕДУРЫ

Исходная программа описывает обработку конкретной детали на конкретном станке, но часто в разных программах приходится описывать одни и те же элементы, применять одни и те же схемы обработки, иногда имеется целая группа схожих деталей. В этом случае можно составить на языке САПР - ЧПУ исходную программу обработки типового элемента (например, спирали Архимеда), типовой схемы (например, сверление группы отверстий, образующих матрицу), типовой детали. Такая программа называется макропроцедурой. Макропроцедуры хранятся на магнитных дисках.

В макропроцедуре не указываются некоторые или все размеры, технологические параметры. В исходной программе не записывается оператор вызова макропроцедуры, в котором указывается имя макропроцедуры и недостающие конкретные размеры и технологические параметры (параметры макропроцедуры). При трансляции исходной программы на ЭВМ по оператору вызова макропроцедура считывается с дисков и

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

выполняется. Макропроцедуру можно сравнить с участком. Если обрабатываемый контур содержит несколько идентичных по конфигурации и обработке участков, то в программе участок описывается один раз. Участок можно обработать только в одной программной процедуре, в которой он записан. К макропроцедуре, записанной на магнитных дисках, можно обращаться из разных исходных программ.

Каждая макропроцедура описывает определенное множество (определенный класс) идентичных по конфигурации и технологии обработки структур, начиная от простейших геометрических элементов (например, спирали Архимеда) до однотипных деталей (например, многоступенчатые валы, фланцы). Вызов одной и той же макропроцедуры, но с разными параметрами позволяет получить различные элементы (детали) из этого множества, которые могут отличаться друг от друга не только размерами, но и конфигурацией.

Для использования макропроцедуры не нужно знать как она составлена, достаточно знать вызов макропроцедуры и правила пользования макропроцедурой.

8.32.1 ВЫЗОВ МАКРОПРОЦЕДУРЫ

```
МАКРО/<ИМЯ>(<ВХОДНЫЕ ПАРАМ.>/<ВЫХОДНЫЕ ПАРАМ.>);
```

ИМЯ - имя макропроцедуры;

ВХОДНЫЕ ПАРАМ. - список параметров, которые передаются из программы в макропроцедуру;

ВЫХОДНЫЕ ПАРАМ.- список параметров, которые после выполнения макропроцедуры передаются из макропроцедуры в программу.

ПАРАМЕТРЫ это числовые данные, геометрические элементы, символьные строки. Каждый параметр имеет имя. В каждом из двух списков один параметр от другого отделяется символом ';'. Последовательность параметров - произвольная. Список выходных параметров отделяется от списка входных параметров символом /. Список выходных параметров может отсутствовать:

```
МАКРО/<ИМЯ>(<ВХОДНЫЕ ПАРАМЕТРЫ>);
```

Входной параметр можно задать тремя способами:

1. Значение параметра указывается явно.

```
<ИМЯ ПАРАМЕТРА>=<ЧИСЛО>;  
'<СТРОКА>';
```

Пример: МАКРО/SPIRAL(RN=60;...НОМ='01';...

Примечание: Если параметр - угол, то должны выполняться правила для задания угловых величин.

Пример: МАКРО/SPIRAL(FH=F/60.;...

2. Параметру присваивается значение данного из программы, имя данного не совпадает с именем параметра.

```
<ИМЯ ПАРАМЕТРА>=<ИМЯ ДАННОГО ИЗ ПРОГРАММЫ>;
```

Примеры: МАКРО/SPIRAL(RH=A;FH=B1;...
МАКРО/PRIM(TK1=TK5;ПР1=ПР10;...

3. Параметру присваивается значение данного из программы, имя данного совпадает с именем параметра.

<ИМЯ ПАРАМЕТРА>;

Пример: МАКРО/SPIRAL (RH;FH;...

4. Параметр – массив идентификаторов передается при помощи имени и числа элементов. Данная форма передачи параметров, введенная в язык программирования системы САПР-ЧПУ/4, экономит Ваше время и уменьшает длину исходного текста.

<ИМЯ ПАРАМЕТРА>, число;

Пример: МАКРО/S (RH, 5;FH;... В данном примере в макро передаются пять элементов массива – R0, R1, R2, R3, R4, R5.

Выходной параметр можно задавать двумя способами:

1. Значение выходного параметра присваивается данному из программы, имя данного не совпадает с именем параметра.

<ИМЯ ДАННОГО ИЗ ПРОГРАММЫ>=<ИМЯ ПАРАМЕТРА>;

Пример: МАКРО/PRIM (.../X1=A;TK10=TK1);

2. Значение выходного параметра присваивается данному из программы, имя данного совпадает с именем параметра.

<ИМЯ ПАРАМЕТРА>;

Пример: МАКРО/PRIM (.../A;TK1);

После выполнения макропроцедуры в программе будут изменены значения только тех данных, идентификаторы которых в списке выходных параметров, все остальные данные не будут изменены.

ПРАВИЛА ПРИМЕНЕНИЯ ОПЕРАТОРА МАКРО

1. Движение перед вызовом макропроцедуры должно быть ограничено одним из операторов: ДО ТК, ДО ПР, ДО КР. Эти операторы определяют на контуре точку, которая будет начальной точкой макропроцедуры.

2. Если перед оператором МАКРО нет операторов ПР, +КР, -КР, то операторы ДО ТК, ДО ПР, ДО КР не нужны.

3. Макропроцедура описывается в собственной системе координат, связь с системой координат программы устанавливается автоматически.

4. Если процедура, участок или макропроцедура начинается с вызова макропроцедуры, то перед оператором 'МАКРО' записывается оператор 'ОТ ТК':

```

НМ;
НП<НОМЕР>; ОТ ТК<НОМЕР>; МАКРО/...
НУ<НОМЕР>;
  
```

5. После выполнения макропроцедуры можно использовать любой оператор движения, например, ДО ТК, ПР, +КР, -КР.

6. Если до выполнения макропроцедуры был установлен режим автоматического расчета эквидистанты, то он сохраняется и в макропроцедуре.

7. Если до выполнения макропроцедуры введена коррекция радиуса инструмента, то она сохраняется и в макропроцедуре.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Пример вызова макропроцедуры.

Макропроцедура SPIRAL - движение по (вдоль) спирали Архимеда.

```
МАКРО/SPIRAL (FN;FK;RN;RK;ШАГF);
```

```
FN - начальный угол,
FK - конечный угол,
RN - начальный радиус,
RK - конечный радиус,
ШАГF- приращение угла.
```

Пример: (см.рис. 130)

```
ПРОГРАММА=PRIMER5;
СТАНОК=205;
TK1=R/60,B/30.;
KP1=0,0,60;
KP2=0,0,70;
.....
НПО; ...-KP1;TK1;
    МАКРО/SPIRAL (FN=F/60.;FK=F/-120.;RN=60;RK=70;
    ШАГF=F/-1.);-KP2;...КПО;!
```

8.32.2 СОСТАВЛЕНИЕ МАКРОПРОЦЕДУРЫ

Макропроцедура составляется независимо от какой-либо программы, которая будет использовать эту макро. Составление макропроцедуры в принципе почти не отличается от составления исходной программы.

Структура макропроцедуры:

```
МАКРО=<ИМЯ>;
[*<КОММЕНТАРИЙ>;]
<РАЗДЕЛ ДАННЫХ>
!
НМ; ... КМ;
!
```

1. Имя макропроцедуры состоит из латинских букв и цифр, количество символов - не более восьми.
2. Число комментирующих фраз - любое, могут отсутствовать.
3. Раздел данных составляется по тем же правилам, что и раздел данных программ, совпадение имен данных в программе и в макропроцедуре не имеет значения. Раздел данных может отсутствовать, в этом случае все числовые данные и геометрические элементы, с которыми работает макропроцедура, должны задаваться при вызове макропроцедуры в списке входных параметров.
4. Раздел процедур составляется по тем же правилам, что и раздел процедур программы, только вместо фраз НП<I>, КП<I> используются фразы: НМ; - начало макропроцедуры, КМ; - конец макропроцедуры;
5. В макропроцедуре могут использоваться участки, которые описываются по общим правилам вне фраз НМ и КМ, совпадение номеров участков в программе и в макро не имеет значения.

6. В макропроцедуре можно использовать все операторы языка САПР - ЧПУ, в том числе операторы вызовов других макропроцедур. Вложенность макропроцедур - любая. Каждая макропроцедура составляется независимо от внешней макропроцедуры.

7. При вложенности макропроцедур параметры внешней макропроцедуры не передаются автоматически вложенным макропроцедурам. Пример:

```
ПРОГРАММА=PRIM0;
.....
НПО; ... МАКРО/PRIM1 (X0=10;Y0=20;...
МАКРО=PRIM1;
.....
НМ; ... МАКРО/PRIM2 (X0;B=Y0;...
```

Параметры X0, Y0 из программы PRIM0 передаются макропроцедуре PRIM1, из которой они передаются вложенной макропроцедуре PRIM2. В макропроцедуре PRIM2 параметр Y0 имеет уже другое имя - B.

Примечание: без особой необходимости не следует использовать глубокую вложенность макропроцедур, так как это увеличивает время трансляции на ЭВМ.

8.32.3 ОБЯЗАТЕЛЬНЫЕ И НЕОБЯЗАТЕЛЬНЫЕ ПАРАМЕТРЫ МАКРО

Входные параметры МАКРО могут быть обязательными и необязательными. Обязательные параметры в разделе данных МАКРО не задаются, они должны быть заданы в операторе вызова МАКРО. Необязательные параметры задаются в разделе данных МАКРО (значения по умолчанию). Если в операторе вызова МАКРО необязательный параметр не указан, то действует значение по умолчанию. Если необязательный параметр в вызове указан, то значение по умолчанию игнорируется. Макропроцедура составляется так, как будто в разделе данных заданы все параметры, то есть имена обязательных параметров, не заданных в разделе данных, используются на общих основаниях. Если в макропроцедуре используются операторы нарезания резьбы, обработки отверстий, выборки, то параметры (H, T, S, P, R, SP, SX, Z, L, Ш, Y) могут быть нулевыми. Нулевые параметры игнорируются как не заданные.

8.32.4 ПРОВЕРКА НА ЗАДАНИЕ ПАРАМЕТРА В ВЫЗОВЕ

В макропроцедуре можно проверить задан или нет в вызове тот или иной параметр. Для этого используется оператор условного перехода или оператор выбора:

<pre>ЕСЛИ ЕСТЬ<ИМЯ1>'ИЛИ'ЕСТЬ<ИМЯ2>...;ИДИ M<I>; НЕТ 'И' НЕТ ТО;</pre>
--

Примеры:

1. ...ПР[1];ЕСЛИ ЕСТЬ R[1];ТО R[1];КЕ;ПР1[1];...

Пусть в момент выполнения этих операторов I1=1, тогда при заданном радиусе R1 имеет место скругление пересекающихся прямых: ...ПР1;R1;ПР2;... Если R1 не задан, то скругления не будет: ...ПР1;ПР2;...

2. НМ; ЕСЛИ НЕТ FN 'ИЛИ' НЕТ FK 'ИЛИ' НЕТ RN 'ИЛИ'
 НЕТ RK 'ИЛИ'НЕТ ШАГF;ИДИ M1;

```
.....
M2; KM;
M1;ПЧ/'ЗАДАНЫ НЕ ВСЕ ПАРАМЕТРЫ';ИДИ M2;!
```

3. ЦИКЛ;I1=0.1.N-1;ЕСЛИ НЕТ X1[1]; ТО; X1[1]=X0[1]; КЕ;КОНЕЦ;

Проверяется массив из N чисел: X0, X1, X2,...(N>=2). Если какой-то элемент массива не задан, то ему присваивается значение предыдущего элемента массива.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

В одном операторе ЕСЛИ... нельзя проверять задан или нет параметр и одновременно записывать какое-либо условие с этим же параметром.

4. Примеры:

1. Неправильно: ЕСЛИ ЕСТЬ T1 'И' T1=N;ТО;...
2. Правильно: ЕСЛИ НЕТ T1;ИДИ M1;ЕСЛИ T1=N;ТО;...
ЕСЛИ ЕСТЬ T1 'И' T1=N; ТО; ...

5. ПРИМЕР СОСТАВЛЕНИЯ МАКРОПРОЦЕДУРЫ (СПИРАЛЬ АРХИМЕДА).

```

МАКРО=SPIRAL;
*СПИРАЛЬ АРХИМЕДА;
*ПАРАМЕТРЫ: FN, FK, RN, RK, ШАГФ;
TK1=R/RN, B/FN;
A=0; F=0; N=0; K=(RK-RN):(FK-FN); !
NM; ЕСЛИ НЕТ FN 'ИЛИ' НЕТ FK 'ИЛИ' НЕТ RN 'ИЛИ'
    НЕТ RN 'ИЛИ' НЕТ ШАГФ; ИДИ M1;
    ЕСЛИ (FK-FN)*ШАГФ<0; ИДИ M2;
    N=ОЦБ((FK-FN):ШАГФ); ШАГФ=(FK-FN):N;
    ТК1;
    ЦИКЛ; F=FN+ШАГФ, ШАГФ, N;
    A=RN+K*(F-FN); ХТК1=A*COS(F); УТК1=A*SIN(F);
    ТК1;
    КОНЕЦ;
M3; KM;
M1; ПЧ/'ДЛЯ МАКРО SPIRAL ЗАДАНЫ НЕ ВСЕ ПАРАМЕТРЫ';
    ИДИ M3;
M2; ПЧ/'ДЛЯ МАКРО SPIRAL НЕВЕРНО ЗАДАНЫ ЗНАКИ УГЛОВ';
    ИДИ M3;

```

8.32.5 ГРАВИРОВАНИЕ ТЕКСТА ПО ГОСТ 2930-62

Для гравирования текста необходимо привести инструмент в точку начала текста и записать оператор вызова макропроцедуры SHPO.

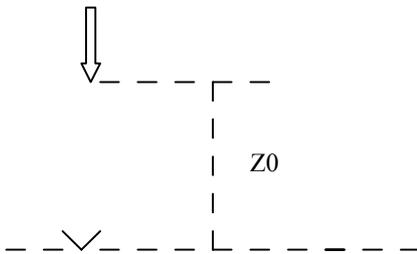
Точка начала текста - это нижняя левая точка воображаемого прямоугольника, в который можно вписать первую букву текста.

Вызов макропроцедуры SHPO:

```
...МАКРО/SHPO(N0;Z0;S1;S2;X0;Y0;R1;R2;...RN=-1);...
```

где:

N0 - номер шрифта (высота символов в миллиметрах);
Z0 - расстояние (положительное число) от поверхности текста до фрезы (если не задано, равно 1.5);



S1 - рабочая подача (если не задана, равна 80);

S2 - ускоренная подача (если не задана, равна 300);

X0, Y0 - смещения по X, Y от начальной точки текста до центра окружности, если текст располагается по дуге окружности. Текст можно получить только на верхней половине окружности. Параметры не задаются, если текст располагается на прямой.

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

R1, R2 - однозначные и двухзначные коды символов согласно таблице: NM, где N - номер столбца, M - номер строки. Например, буква Е имеет код 25, цифра 2 - код 2. Промежутки между словами получаются при пропуске очередного номера R. RN=-1 - последний обязательный параметр, означает конец строки текста.

F0 - угол от положительного направления оси X до прямой, по которой располагается текст (незаданный угол считается равным нулю).

ТАБЛИЦА КОДОВ СИМВОЛОВ.

N=	0	1	2	3	4	5	6
M=0	0	.	А	К	Ф	Ю	Р
M=1	1	-	Б	Л	Х	Я	С
M=2	2	/	В	М	Ц	Д	U
M=3	3		Г	Н	Ч	Ф	V
M=4	4		Д	О	Ш	G	W
M=5	5		Е	П	Ш	I	Y
M=6	6		Ж	Р		J	Z
M=7	7		З	С	Ы	L	
M=8	8		И	Т	Ь	N	
M=9	9		Й	У	Э	Q	

Коды непечатных символов.

Твердый знак - 46;
 Прописные греческие буквы:
 ДЕЛЬТА - 67; СИГМА - 71;
 ТЭТА - 68; ПСИ - 72;
 ЛАМБДА - 69; ОМЕГА - 73;
 КСИ - 70.

Пример:

```

ПРОГРАММА=GRAV;СТАНОК=...;
*ГРАВИРОВАНИЕ СТРОКИ "ШРИФТ ПО ГОСТ 2930-62";
*НОМЕР ШРИФТА - 5;
*СЛОВО "ШРИФТ" ЗАДАНО ПАРАМЕТРАМИ R1, R2, R3, R4, R5;
*СЛОВО "ПО" - R7, R8;
*СЛОВО "ГОСТ" - R10, R11, R12, R13;
*СЛОВО "2930-62" - R15, R16, R17, R18, R19, R20, R21;
*ПРОПУЩЕННЫЕ R6, R9, R14 - ПРОМЕЖУТКИ МЕЖДУ СЛОВАМИ;
*R22=-1 - КОНЕЦ ТЕКСТА;
!
НП0;МАКРО/SHPO(N0=5;R1=44;R2=36;R3=28;R4=40;R5=38;
R7=35;R8=34;R10=23;R11=34;R12=37;R13=38;R15=2;
R16=9;R17=3;R18=0;R19=11;R20=6;R21=2;R22=-1);
КП0;!
  
```

Для получения этого же текста с номером шрифта 3, расположенного на дуге окружности радиуса 50мм, необходимо в примере заменить параметр N0 : N0=5; и задать центр окружности: X0=35.35; Y0=-35.35; После выполнения макропроцедуры инструмент не возвращается в начальную точку текста, дальнейшее перемещение задается оператором ДО ТК.

Пример: НП0;...ДО ТК1;МАКРО/SHPO(...);ДО ТК2;...

Для гравирования строки текста, не содержащей символы греческого алфавита и специальные символы ('!', '' и т.п.), можно использовать МАКРО с именем GRAV. Параметры ее вызова совпадают с параметрами вызова МАКРО SHPO, однако не надо

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

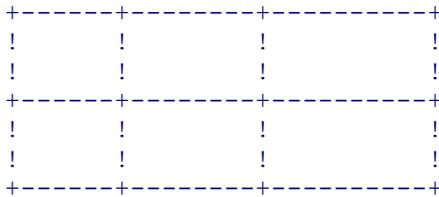
задавать коды символов. Вместо этого используйте параметр ТЕКСТ. Например: ТЕКСТ='ЭТО СТРОКА ДЛЯ ГРАВИРОВКИ\$';. Промежутки между словами кодируйте пробелом. Строку завершите символом '\$', который не гравится.

```

В нашем примере:
ПРОГРАММА=GRAV;СТАНОК=...;
*ГРАВИРОВАНИЕ СТРОКИ "ШРИФТ ПО ГОСТ 2930-62";
*МАКРО GRAV;!
НПО;
    МАКРО/GRAV(N0=5;ТЕКСТ='ШРИФТ ПО ГОСТ 2930-62$');
КПО;!
    
```

8.32.6 СВЕРЛЕНИЕ ГРУППЫ ТОЧЕК , ОБРАЗУЮЩИХ МАТРИЦУ.

Будем говорить, что группа точек образует матрицу, если эта группа образует структуру, подобную следующей:



Символ + на рисунке обозначает точки. Каждая точка получается в пересечении строки и столбца матрицы. Эти точки будем называть узлами матрицы. Сверление может осуществляться не во всех узлах матрицы. Число строк - не больше 100, число столбцов - не больше 100. Расстояние между строками и между столбцами может быть переменным.

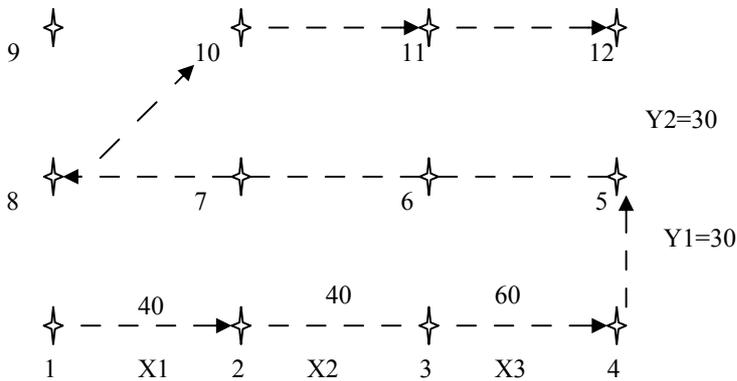
Для сверления в группе точек, образующих матрицу, используется макропроцедура SWRM:

```

МАКРО/SWRM(NX;X1;X2;...NY;Y1;Y2;...
            T1;P1;T2;P2;...L0;Z0;S0;F0;SX;A);
    
```

```

Пример:
ПРОГРАММА=SWR;
СТАНОК=205;
ТК0=0,0;ТК1=100,100;!
НПО;S/1000;N/400;ТК0;ТК1; МАКРО/SWRM( NX=4;X1=40;X3=60;NY=3;Y1=30;
T1=3;T2=6;P2=2;T3=9;Z0=-100;L0=20;S0=100);S/1000;ТК0;КПО;!
    
```



Цифрами 1, 2, ...12 обозначены узлы матрицы.

Точка 1 - точка начала обработки.

Сверление осуществляется в следующих узлах матрицы в указанном порядке: 1-2-4-5-8-10-11-12. На рисунке эти узлы обозначены символом +. В узлах 3, 6, 7, 9 сверления нет. На рисунке эти узлы обозначены символом . Пунктирная линия обозначает траекторию движения инструмента.

ПАРАМЕТРЫ МАКРОПРОЦЕДУРЫ SWRM:

NX - число столбцов в матрице, то есть число точек в строке ($NX \leq 100$).

NY - число строк в матрице, то есть число точек в столбце ($NY \leq 100$).

$X<T>$ - расстояние между столбцами с номерами $I, I+1$ ($I=1, 2, \dots, NX-1$).

Если $X<I>=X<T-1>$, то $X<I>$ можно не указывать (в примере $X2$ не указан, так как $X2=X1=40$).

За начальную точку можно выбрать любую из 4-х угловых точек матрицы. Значения $X<I>$ задаются положительными или отрицательными в зависимости от направления смещения по X от начальной точки в первой строке матрицы.

$Y<T>$ - расстояние между строками с номерами $I, I+1$. ($I=1, 2, \dots, NY-1$). Если $Y<I>=Y<I-1>$, то $Y<I>$ можно не указывать (в примере $Y2$ не указан, так как $Y2=Y1=30$). Знаки $Y<I>$ определяют направление смещения по Y от одной строки к другой.

$T<I>$ - номер узла в порядке следования, в котором нет сверления ($I=1, 2, \dots$ нумерация подряд без пропуска).

$P<T>$ - количество следующих друг за другом узлов, в которых нет сверления, начиная с узла, указанного в $T<I>$. Номер I параметра $P<I>$ должен совпадать с номером I соответствующего параметра $T<I>$.

Если пропускается одиночный узел, то $P<I>=1$ не указывается (в примере: $T1=3; T2=6; P2=2; T3=9; P1, P3$ не указаны, так как $P1=1, P3=1$).

$L0, Z0, S0$ - параметры цикла сверления:

$Z0$ - быстрый подход с учетом недохода ($Z0 < 0$) (если параметра нет, быстрого подхода не будет);

$L0$ - рабочий ход (глубина отверстия + недоход + перебег);

$S0$ - рабочая подача (если не задана, действительна подача, установленная до вызова макропроцедуры).

$F0$ - угол поворота матрицы (если не задан, считается равным нулю).

От того заданы или нет параметры $L0, SX, A$ по разному будет выполняться макропроцедура. Возможны 4 варианта:

Вариант 1. Параметр $L0$ не задан (в этом случае не задаются также $Z0, S0, SX, A$).

Макропроцедура осуществляет только обход точек: $HM; OT TK...; DO TK...; \dots KM$; Операторы СВЕРЛ и техкоманда КЦ записываются в основной программе.

Пример: $\dots СВЕРЛ(Z/-100; L/20; S/100); МАКРО/SWRM(\dots); \dots КЦ; \dots$

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Вариант 2. Параметр L0 задан, параметры SX, A не заданы.

Один автоцикл оформляется на всю группу точек: НМ;СВЕРЛ(L0;Z0;S0);ОТ ТК...;ДО ТК...;ДО ТК...;...КЦ;КМ;

Вариант 3. Параметр L0 задан, параметр SX задает подачу перехода от точки к точке, параметр A не задан.

Автоцикл оформляется в каждой точке:

НМ;СВЕРЛ(Z0;L0;S0);КЦ;ОТ ТК...;ДО ТК...;

СВЕРЛ(Z0;L0;S0);КЦ;ДО ТК...КМ;

Вариант 4. Параметры L0, SX, A заданы, A=1.

Система ЧПУ не реализует автоцикл сверления, сверление

программируется в макропроцедуре операторами Z:

НМ;S/SX;Z0;S0;Z/-L0;S/SX;Z/L0;ОТ ТК...;ДО ТК...;

S0;Z/-L0;S/SX;Z/L0;ДО ТК...;...КМ;

Перед началом обработки инструмент должен находиться в начальной точке на расстоянии !Z! от обрабатываемой поверхности (параметр Z может отсутствовать).

Макропроцедуру SWRM можно использовать для сверления группы точек на прямой. В этом случае не указываются параметры NY, Y1, Y2 и т.д., параметр NX задает количество точек на прямой.

После выполнения макропроцедуры инструмент остается в конечной точке, дальнейшее движение программируется оператором ДО ТК: ...МАКРО/SWRM(...);S/8000;ДО ТК0;...

8.32.7 НАРЕЗАНИЕ РЕЗЬБЫ НА ТОКАРНЫХ СТАНКАХ

Для нарезания резьбы на токарных станках используется макропроцедура 'REZBA':

```
МАКРО/РЕЗВА(A; TP; P; L; N; K);
```

Обязательно задаваемые параметры: A; TP; P; L; N.

Параметры МАКРО:

A - схема нарезки:

A=1 - радиальная (см.рис. 135),

A=2 - генераторная (см.рис. 136),

A=3 - комбинированная (см.рис. 137);

TP - тип резьбы:

TP=1 - метрическая (СТАНДАРТ СЭВ 180-75),

TP=2 - дюймовая (ОСТ НКТП-1260),

TP=3 - трапецеидальная (ГОСТ 9484-73),

TP=4 - упорная (ГОСТ 10177-62);

P - шаг резьбы:

P>0 - внешняя резьбы (вал),

P<0 - внутренняя резьба (гайка);

L - длина резьбы с учетом выхода резца:

L>0 - движение резца по координате X вправо от начальной точки МАКРО,

L<0 - движение резца по координате X влево от начальной точки МАКРО;

N - количество черновых проходов;

K - количество чистовых проходов (при чистовых проходах обработка ведется на полный профиль резьбы с первого чистового прохода).

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

Во всех трех схемах врезание на каждый черновой проход происходит по принципу равенства снимаемых площадей: $S_1=S_2=S_3=S_4$. (см.рис. 135, 136, 137).

Примечания:

1. При применении генераторной схемы в деталях с канавкой под выход резца длину нужно брать без учета выхода в канавку, т.е. $L=L_1+L_2$ (см.рис. 138).
2. Упорная резьба по комбинированной схеме не обрабатывается.
3. Начальная и конечная точки (TKN, см.рис. 138) макро совпадают. Расположение TKN по координате X произвольное, по координате Y на 1 мм ниже (вал) или выше (гайка) от нарезаемой поверхности.
4. Работа с макропроцедурой REZBA возможна только при описании в паспорте токарного станка цикла "нарезр".

Примеры вызова МАКРО:

```
МАКРО/РЕЗВА(ТР=1; А=1; Р=20; L=-100; N=10; К=1);
МАКРО/РЕЗВА(ТР=2; А=2; Р=-3; L=100; N=2);
МАКРО/РЕЗВА(ТР=3; А=3; Р=4; L=-100; N=5);
МАКРО/РЕЗВА(ТР=4; А=2; Р=5; L=100; N=4; К=2);
```

8.33 ФАЙЛЫ ДАННЫХ НА МАГНИТНЫХ ДИСКАХ

Файл - это совокупность числовых данных и (или) символьных строк, записанных на магнитные диски.

Файлы на дисках организуются технологом - программистом двумя способами:

1. Файл вводится с терминала и записывается на диски как обычный текстовый файл с помощью текстового редактора, например, NE.
2. Файл формируется и записывается на диски при трансляции исходных программ и макропроцедур, содержащих специальные операторы вывода.

Структура файла определяется технологом, организующим файл, и зависит от структуры данных и логики их обработки. Под обработкой файла понимается ввод (чтение) данных с дисков в память ЭВМ и их обработка или вывод (запись) данных из памяти ЭВМ на диски.

Обработка файла программируется технологом в исходных программах и макропроцедурах с помощью специальных операторов ввода-вывода.

Пример файла. Файл содержит данные о назначении оборотной подачи и частоты вращения шпинделя при обработке отверстий в зависимости от шифра материала, класса и диаметра инструмента.

Шифр матер.	Класс инструм.	Диаметр инструм.	Подача (мм/об)	Обороты (об/мин)
1	10	12	0.3	1000
1	10	40	0.5	600
...
1	20	12	0.2	1200
...
2	30	30	0.3	800
...

Примечание: В конце параграфа будет рассмотрен пример обработки данного файла.

8.33.1 ЗАПИСЬ - ЭЛЕМЕНТ ФАЙЛА

Файл состоит из записей. Запись - это логическая единица файла, то есть логически завершенная последовательность чисел или символов. Число чисел в записи или число символов в строке называется длиной записи.

В нашем примере:

1 10 12 0.3 1000 - первая запись,
1 10 40 0.5 600 - вторая запись и т.д.

Запись может состоять из одного числа или из нескольких чисел, но не более 64. В нашем примере длина записи равна 5 (материал, класс, диаметр, подача, обороты). Число записей в файле - любое. В одном файле могут быть записи равной длины и различной структуры (в нашем примере структура и длина всех записей одинаковая). Файлы могут читаться с дисков и записываться на диски только записями. Поэтому запись - это не только логическая единица файла, но и порция ввода или вывода.

8.33.2 ИМЯ ФАЙЛА ДАННЫХ

Каждому файлу при его организации технолог присваивает имя, которое используется во всех операторах обработки этого файла. Имя файла состоит не более, чем из восьми символов - латинских букв и цифр.

8.33.3 ТИПЫ ФАЙЛОВ ДАННЫХ

В зависимости от способа организации файла, он называется файлом в символьном формате (если введен с терминала с помощью текстового редактора) или файлом во внутреннем формате ЭВМ (если его вывод на диски запрограммирован в исходной программе или макропроцедуре).

В зависимости от использования файла, он называется входным (записи существующего на дисках файла считываются в память ЭВМ по специальным командам в исходных программах) или выходным (по специальным командам в исходных программах данные из памяти ЭВМ записываются в файл на дисках).

8.33.4 СТРУКТУРА ФАЙЛА В СИМВОЛЬНОМ ФОРМАТЕ

Файл состоит из строк двух видов:

1. Строки с комментарием.
2. Строки числовых данных.

Строка с комментарием начинается с символа *.

Примеры:

* шифр класс диаметр подача обороты
* матер. инструм. инструм. (мм/об) (об/мин)

При чтении файла строки с комментарием пропускаются.

Строка данных состоит из одного или нескольких чисел. Числа записываются в обычном формате, принятом в САПР - ЧПУ. Числа в строке отделяются друг от друга любым количеством пробелов или знаков табуляции.

Строка данных соответствует одной записи файла или части записи. Если все числа одной записи не помещаются в одной строке, то используется символ продолжения \$, за этим символом в строке может следовать комментарий. Вместо ввода символа \$

можно просто ввести последовательность символов ВК-ПС или в текстовом редакторе нажать на клавишу Enter.

Пример:

```
* первая запись
10 20 30
* вторая запись
100 80 -10 0 $          начало второй записи
-10.567 20.3           продолжение второй записи
50
* третья запись
30 40
Данный текст эквивалентен следующему:
10 20 30
100 80 -10 0 -10.567 20.3 50
30 40
```

При чтении файла запись, состоящая из нескольких строк, вводится полностью, как будто она состоит из одной строки.

8.33.5 ФАЙЛ ВО ВНУТРЕННЕМ ФОРМАТЕ ЭВМ

Числовые данные выводятся из памяти ЭВМ в такой файл не в символьном коде, а в том формате, в котором они представлены в памяти ЭВМ. Такие файлы нельзя распечатать или отредактировать как текстовые файлы. Файлы во внутреннем формате ЭВМ служат для обмена данными между программами, макропроцедурами.

8.33.6 ОПЕРАТОРЫ ОБРАБОТКИ ФАЙЛОВ

8.33.6.1 ОТКРЫТЬ ВХОДНОЙ ФАЙЛ ДЛЯ ЧТЕНИЯ

ОФЧ/<ИМЯ ФАЙЛА>;

Пример: ОФЧ/ТАВЛ;

Все файлы данных в САПР - ЧПУ находятся в центральном каталоге САПР-ЧПУ и имеют расширение DAT (см.инструкцию по работе на ЭВМ). Указанный файл должен существовать на томе, он может быть файлом в символьном или внутреннем формате. В дальнейшем этот файл можно только читать, изменить или дополнить его новыми данными из программы нельзя. Оператор ОФЧ должен предшествовать самому первому оператору чтения данных из файла.

8.33.6.2 ЧИТАТЬ ДАННЫЕ ИЗ ФАЙЛА В СИМВОЛЬНОМ ФОРМАТЕ

ЧДФС/<ИМЯ ФАЙЛА>,<ДЛИНА ЗАПИСИ>,<ПОЛЕ ВВОДА>;

Длина записи - идентификатор числового данного, которому после считывания записи будет присвоено значение длины записи (количество чисел в записи).

Поле ввода - идентификатор первого числа записи.

Все остальные числа из записи будут иметь идентификаторы того же типа с номерами на 1, 2, 3 и т.д. больше.

Пример: ЧДФС/ТАВЛ, L, X0;

Допустим прочитана первая запись нашего файла:

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

1 10 12 0.3 1000
 В результате: L=5; X0=1; X1=10; X2=12; X3=0.3; X4=1000;

Идентификаторы длины записи и поля ввода могут индексироваться.

Пример: ЧДФС/MAS,N,Y[1]; I1=I1+N; ...

В разделе данных необходимо зарезервировать поле ввода, рассчитанное на максимальную длину записи, например, если поле ввода начинается с Z0 и максимальная длина записи равна 10, то в разделе данных достаточно записать фразу Z9=0;.

Один и тот же оператор ЧДФС может выполняться несколько раз. В процедуре могут быть несколько операторов ЧДФС, осуществляющих чтение одного и того же файла в одно поле ввода или в разные поля.

Записи файла считываются последовательно: после первой записи читается вторая, затем третья и т.д. Нельзя вернуться назад, нельзя пропустить, не читая, несколько записей.

Если при написании программы неизвестно число записей в файле, то после каждого чтения записи необходимо проверять не равна ли нулю длина записи. Нулевая длина записи означает конец файла.

Пример:

```
...M0; ЧДФС/SPR,DL,DAN[1]; ЕСЛИ DL=0; ИДИ M1;
      <ОБРАБОТКА ЗАПИСИ>
```

ИДИ M0;

```
M1; *КОНЕЦ ФАЙЛА; ...
```

Если "поле ввода" - идентификатор символьной строки, то из файла будет считана символьная строка, количество символов в строке будет считаться длиной строки.

8.33.6.3 ЧИТАТЬ ДАННЫЕ ИЗ ФАЙЛА ВО ВНУТРЕННЕМ ФОРМАТЕ

```
ЧДФ/<ИМЯ ФАЙЛА>,<ДЛИНА ЗАПИСИ>,<ПОЛЕ ВВОДА>;
```

Оператор ЧДФ аналогичен оператору ЧДФС.

8.33.6.4 ОТКРЫТЬ ВЫХОДНОЙ ФАЙЛ ДЛЯ ЗАПИСИ

```
ОФЗ/<ИМЯ ФАЙЛА>;
```

На дисках организуется новый файл во внутреннем формате ЭВМ. Файл записывается в центральный каталог САПР-ЧПУ и имеет расширение DAT. Если уже есть файл данных с тем же именем, то он удаляется и создается новый файл. Оператор ОФЗ должен предшествовать самому первому оператору записи данных.

Файл, открытый для записи, нельзя читать до тех пор, пока он не будет закрыт, то есть, пока он не будет записан на диски полностью.

8.33.6.5 ЗАПИСАТЬ ДАННЫЕ В ФАЙЛ ВО ВНУТРЕННЕМ ФОРМАТЕ

```
ЗДФ/<ИМЯ ФАЙЛА>,<ДЛИНА ЗАПИСИ>,<ПОЛЕ ВЫВОДА>;
```

Длина записи - идентификатор числового данного, значение которого перед выполнением оператора равно числу чисел в записи (после выполнения оператора значение данного не изменяется).

Поле вывода - идентификатор первого числа в записи, идентификаторы остальных чисел имеют тот же тип и номера на 1, 2, 3 и т.д. больше.

Пример: Пусть L1=3; Y5=60.5; Y6=-10; Y7=0; Тогда после выполнения оператора ЗДФ/MAS,L1,Y5; в файл MAS будет выведена запись: 60.5 -10 0

Идентификаторы длины записи и поля вывода могут индексироваться. Записи в файл могут выводиться из одного или из разных полей вывода. Записи в файле следуют друг за другом в той последовательности, в какой они выводились. Прочитать ранее выведенные записи, изменить или удалить их нельзя.

В файл можно записать символьную строку, если поле вывода - идентификатор строки, а длина записи - число символов в строке.

8.33.6.6 ЗАКРЫТЬ ВХОДНОЙ/ВЫХОДНОЙ ФАЙЛ

ЗФ/<ИМЯ ФАЙЛА>;

Оператор означает конец чтения или записи файла. При выводе файла, оператор выполняется после вывода последней записи. После закрытия выходного файла файл можно открыть уже как входной для чтения в той же или в другой программе или макропроцедуре. Входной файл можно закрыть в любой момент, а потом вновь открыть его и начать чтение с первой записи.

8.33.6.7 ОТКРЫТЬ СУЩЕСТВУЮЩИЙ ФАЙЛ ДЛЯ ДОПОЛНЕНИЯ

ОФД/<ИМЯ ФАЙЛА>;

Оператор используется, если необходимо дополнить записанный ранее на диске файл во внутреннем формате ЭВМ новыми данными. Новые данные записываются в конец файла с помощью оператора ЗДФ, старая информация в файле сохраняется.

Оператор ОФД должен предшествовать самому первому оператору записи данных в этот файл.

После вывода последней записи файл закрывается с помощью оператора ЗФ.

Файл в символьном формате дополняется новыми данными с помощью текстового редактора.

8.33.6.8 ОБРАБОТКА НЕСКОЛЬКИХ ФАЙЛОВ

В одной программе (в основной процедуре и во вложенных макропроцедурах) возможна одновременная обработка нескольких входных и выходных файлов, но одновременно могут быть открыты не более трех файлов. Примеры обработки файлов.

Пример 1. Предположим, что макропроцедура NSN осуществляет назначение оборотов шпинделя и оборотной подачи по заданным шифру материала, классу инструмента и диаметру инструмента. В макропроцедуре в качестве входного файла используется файл в символьном формате TABL со следующей структурой записи:

M ! K ! D ! SN ! N

M - шифр материала,
K - класс инструмента,
D - диаметр инструмента,

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

SN - подача (мм/об),

N - частота вращения шпинделя (об/мин).

Записи файла упорядочены по шифру материала, классу инструмента и диаметру:

Сначала идут записи по одному материалу, потом по другому;

В группе записей для одного материала сначала идут записи по одному классу инструмента, потом по другому. В группе записей для одного материала и инструмента записи следуют по возрастанию диаметра. Предположим, что в файле есть группа записей с заданными шифром материала и классом инструмента. Если заданный диаметр совпадает с каким-нибудь диаметром из этой группы, то значения SN, N определяются явно. Если заданный диаметр меньше минимального диаметра из этой группы, то SN и N принимаются равными соответствующим значениям для минимального диаметра. Если заданный диаметр больше максимального диаметра из этой группы, то SN и N принимаются равными соответствующим значениям для максимального диаметра.

Если в файле есть две записи с $D_1 < D < D_2$:

1. M K D1 SN1 N1

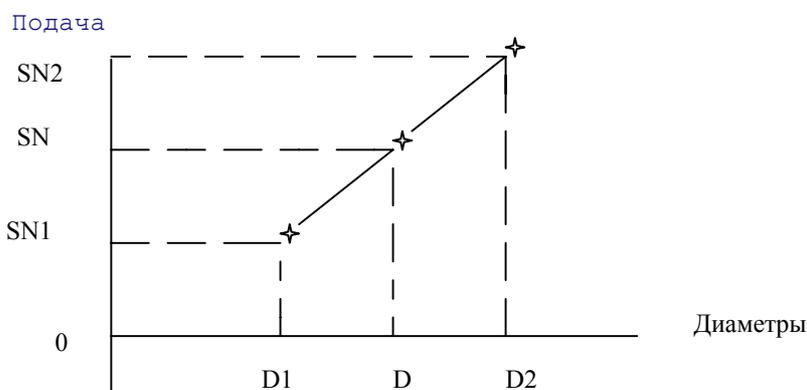
2. M K D2 SN2 N2

То SN и N рассчитываются по формулам:

$$SN = \frac{(D-D_1)(SN_2-SN_1)}{D_2-D_1} + SN_1$$

$$N = \frac{(D-D_1)(N_2-N_1)}{D_2-D_1} + N_1 \quad (N \text{ округляется до целого})$$

Уравнение для SN получается из уравнения прямой, проходящей через 2 точки: (D_1, SN_1) и (D_2, SN_2) .



$$\frac{SN-SN_1}{SN_2-SN_1} = \frac{D-D_1}{D_2-D_1}$$

Аналогично получается уравнение для N.

Пример: НПО; ...МАКРО/NSN (M=1;K=10;D=30;/SN;N);N;SN;TK1;...

Первая запись файла TABL: 1 10 12 0.3 1000;

Вторая запись: 1 10 40 0.5 600.

$$SN = \frac{(30-12) * (0.5-0.3)}{40-12} + 0.3 = 0.44$$

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

40-12

$$N = \frac{(30-12) * (600-1000)}{40-12} + 1000 = 743$$

```

МАКРО=NSN;
* Входные параметры: M,K,D;
* Выходные параметры: SN,N;
A4=0;* поле для ввода записи: A,A1,A2,A3,A4;
L=0;* длина записи;
D1=0;SN1=0;N1=0;* данные из файла;
D2=0;SN2=0;N2=0;* данные из файла;
SN=0;N=0;* выходные параметры;
!
НМ;ОФЧ/TABL;* открыть файл для чтения;
M1;ЧДФ/TABL,L,A;* читать запись из файла -->A,...,A4;
  ЕСЛИ L=0'ИЛИ'M<A;ИДИ M5;* если нет материала-ошибка;
  ЕСЛИ M>A;ИДИ M1;* искать запись с заданным матер.;
  ЕСЛИ K>A1;ИДИ M1;* искать запись с заданным INSTR.;
  ЕСЛИ K>A1;ИДИ M5;* если INSTR. нет-ошибка;
  D1=A2;SN1=A3;N1=A4;
  ЕСЛИ D<=D1;ИДИ M4;
M2;* D>D1;
  ЧДФ/TABL,L,A;* читать следующую запись-->A,...A4;
  ЕСЛИ L=0'ИЛИ'M#A'ИЛИ'K#A1;ИДИ M4;* диаметр>максим.;
  D2=A2;SN2=A3;N2=A4;
  ЕСЛИ D<=D2;ИДИ M3;*если D1<D<=D2, перейти на M3;
* D>D2;
  D1=D2;SN1=SN2;N1=N2;ИДИ M2;*читать следующую запись;
M3;SN=((D-D1)*(SN2-SN1):(D2-D1)+SN1;
N=ОЦ(((D-D1)*(N2-N1):(D2-D1)+N1);
  ЗФ/TABL;*закрывать файл TABL;
M6;KM;*конец макропроцедуры;
M4;SN=N1;N=N1;ИДИ M6;
M5;ПЧ/'ошибка в данных: M=',M(2),' K=',K(2);ИДИ M6;
!
Пример 2. Из программы PROG макропроцедуре МАК через файл
PARAM во внутреннем формате ЭВМ передаются одно значение N1,
10 значений X и 100 значений Y: N1, X0,X1,...X9, Y1,Y2,...Y100.
Программа=PROG;
.....
N1=...;X9=...;Y100=...;
K1=1;K2=10;K3=64;K4=36;
.....
НП0;...ОФЗ/PARAM;*открыть файл для записи;
  ЗДФ/PARAM,K1,N1;*первая запись: N1;
  ЗДФ/PARAM,K2,X0;*вторая запись: X0,X1,...X9;
  ЗДФ/PARAM,K3,Y1;*третья запись: Y1,Y2,...Y64;
  ЗДФ/PARAM,K4,Y65;*четвертая запись:Y65,Y66,...Y100;
  ЗФ/PARAM;*закрывать файл;
  МАКРО/МАК; ...

МАКРО=МАК;
A=0;P1=1;P2=10;P3=64;P4=36;
R9=0;Z100=0;
.....
НМ;ОФЧ/PARAM;*открыть файл для чтения;
  ЧДФ/PARAM,P1,A;*N-->A;
  ЧДФ/PARAM,P2,R;*X0,X1,...X9-->R0,R1,...R9;

```

```

ЧДФ/PARAM,P3,Z1;*Y1,Y2,...Y64-->Z1,Z2,...Z64;
ЧДФ/PARAM,P4,Z65;*Y65,Y66,...Y100-->Z65,Z66,...Z100;
ЗФ/PARAM;*закрывать файл;
.....

```

8.34 Оператор "ВЫЗОВ СП"

Для вызова станочной подпрограммы из раздела процедур используется оператор "ВЫЗОВ СП", в котором указывается условное имя СП, список параметров и их значения.

Формат оператора "ВЫЗОВ СП":

```

<имя СП> ( <пар1>=<знач>;<пар2>=<знач>;...<парN>=<знач> );
, где:

```

<имя СП> - имя, условно присвоенное станочной подпрограмме технологом (не более 8 букв латинс. алфавита и цифр);

<пар N> - имя, присвоенное N-му параметру СП (не более 6 букв латинс. алфавита и цифр);

<знач> - значение параметра (идентификатор или арифметическое выражение).

Пример: В память стойки BOCH CNC SYSTEM MICRO-8 защиты несколько СП. Одна из них - СП для внутреннего фрезерования рамки. Присвоим этой СП условное имя SUBR3, а ее параметрам имена: P, V0, ..., V6. Параметры и их значения имеют следующий смысл:

```

P - коэффициент повторения подпрограммы,
V0 - длина рамки по оси X,
V1 - длина рамки по оси Y,
V2 - радиус на углах,
V3 - припуск+расстояние безопасности,
V4 - перемещение по оси Z,
V5 - направление движения фрезы,
V6 - диаметр фрезы.

```

Тогда для вызова СП вставим в раздел процедур исходной программы следующую фразу:

```

... SUBR3(P=1;V0=150;V1=80;V2=10;V3=5;V4=-50;V5=1;V6=10);...

```

Данный оператор преобразуется ИНВАРИАНТНЫМ ПОСТПРОЦЕССОРОМ в один кадр управляющей программы, например:

```

N32(CRC,3,1,V0+150,V1+80,V2+10,V3+5,V4-50,V5+1,V6+10)LF

```

Правила использования оператора "ВЫЗОВ СП":

1. Количество операторов "ВЫЗОВ СП" используемых в разделе процедур не ограничено, т.е. можно вызывать любое количество подпрограмм.
2. Количество параметров, передаваемых в подпрограмму, не более 20.
3. Порядок перечисления параметров в операторе "ВЫЗОВ СП" произвольный.
4. Для разделения параметров в операторе "ВЫЗОВ СП" используется только символ '; '.
5. Для формирования кадра вызова СП в управляющей программе необходимо описать его характеристики в паспорте, используя оператор "ОПИСАНИЕ СП" (см. инструкцию по языку составления паспортов в системе САПР-ЧПУ, глава 13) .

8.35 Оператор ВЫЗОВ exe (com) файлов

Данный раздел предназначен для специалистов обладающих навыками программирования на любом алгоритмическом языке и работе на ЭВМ.

В системе СПР-ЧПУ/3 разработан механизм вызова задач, т.е. exe (com) файлов из любого места программы или макропроцедуры, что позволяет :

- повысить эффективность работы некоторых макропроцедур, передав часть громоздких вычислительных функций задачам;
- расширять функции процессора системы, интегрируясь с другими задачами (расчет режимов резания, выбор инструмента и т.п).

Под задачей понимается разработанный на любом алгоритмическом языке программный модуль с расширением .EXE или .COM. Объем задачи зависит от ресурса ОЗУ конкретного компьютера.

Вызываемая задача должна находиться либо в текущей директории.

В том случае если задача принимает параметры (идентификаторы из раздела данных) и возвращает параметры, необходимо учитывать интерфейсные соглашения. Интерфейсом между задачей и процессором САПР-ЧПУ являются три файла:

1. IDENT.DAT - идентификаторов входных параметров задачи; Этот файл состоит из одной записи следующей структуры:

- длина всей записи (беззнаковое целое, длиной 2 байта, без учета собственной длины);
- набор описаний входных параметров.

Каждое описание одного входного параметра содержит:

- имя идентификатора, определяющего переменную или массив, значения которых указаны в соответствующей по порядку записи файла PARAM.DAT;
- количество байт в соответствующей записи PARAM.DAT, которая определяет значение идентификатора или указанных элементов массива.

Каждое имя идентификатора завершается кодом 0.

2.PARAM.DAT - файл значений входных параметров, состоящий из K записей во внутреннем формате ЭВМ. Структура каждой из записей:

- длина записи (беззнаковое целое, длиной 2 байта, без учета собственной длины)
- значение соответствующего идентификатора или элементов массива.

Числовые значения идентификаторов указываются восьми байтовыми полями, значения строковых идентификаторов-полями переменной длины. Для числовых массивов длины записи указываются кратно восьми байтам, длина элемента строкового массива в файлах не фиксируется. Последовательность записей значений в файле определяется списком идентификаторов файла IDENT.DAT. Файлы IDENT.DAT и PARAM.DAT автоматически формируются ПРЕПРОЦЕССОРОМ САПР-ЧПУ в результате обработки оператора &.

3. POST.DAT - файл значений выходных параметров (почты), которые могут быть сформированы вызываемой задачей для последующей их обработки вызывающей макропроцедурой (программой). Структура файла и его записей аналогична PARAM.DAT. Последовательность записей значений в файле определяется списком идентификаторов выходных данных, описанных в вызове задачи.

Любой из перечисленных файлов находится в текущей директории и может быть прочитан с помощью оператора ЧДФ/<имя файла> , что бывает полезным при отладке взаимодействия макропроцедуры с задачей.

Пример: При следующем вызове задачи

```
&LKI(MOP=10000.0; ДОПУСК=0.01; СК=2; ТК30,2);
```

файл IDENT.DAT будет иметь следующее содержимое:

	1	2	3	4
Номер байта	12	3456789012	3456789012	3456789012
Знач. байта	40	MOP	8 ДОПУСК	8 СК 8 ТК30 16

8.35.1 Вызов задачи

```
&<имя задачи>[(<входные парам.>/<выходные парам.>)];
```

имя - имя вызываемой задачи (не более 8-ми и цифр и букв латинского алфавита);

входные парам. - список параметров, передаваемых задаче из макропроцедуры.

выходные парам.- список идентификаторов параметров, которым после выполнения задачи будут присвоены значения из файла POST.DAT

Параметры - это числовые данные, геометрические элементы, символьные строки, массивы. Каждый параметр имеет имя. В каждом из 2-х списков один параметр от другого отделяется символом ";". Последовательность входных параметров произвольная, а выходных определяется последовательностью данных в файле POST.DAT, сформированном вызываемой задачей. Список выходных параметров отделяется от списка входных параметров символом "/". Список выходных (входных) параметров может отсутствовать.

```
&<имя задачи>(<входные парам.>);
```

Входной параметр можно задать тремя способами:

1. Значение параметра указывается явно.

```
<имя параметра>=<а.в>;  
'<строка>';
```

пример: &ETSK(L=20;A='ЭТО ПРИМЕР';B=C*2.54-D);

2. Параметру присваивается значение данного из макропроцедуры (программы), имя данного совпадает с именем параметра

```
<имя параметра>;
```

пример: &ETSK(L;A;B;TK3);

3. Параметру присваивается значение массива данных из макропроцедуры (программы), имя данного совпадает с именем параметра.

```
<имя массива>,<n>;
```

, где n - количество элементов в массиве, любое арифметическое выражение.

Передаваемый массив записывается в файл параметров PARAM.DAT одной записью. Пример:

Инструкция по языку программирования обработки в САПР-ЧПУ/2000 (с рисунками)

```
&ETSK(L;A;B3,7;TK1[2],K1*2;KP4,3;PPO,10);
```

здесь запись B3,7 соответствует записи B3;B4;...B9;

Выходные параметры можно задать только списком идентификаторов.

Если в файле почты POST.DAT передается массив элементов, то он не может быть больше 200 элементов. В разделе данных макропроцедуры соответствующее принимаемому массиву кол-во элементов. В выходных же параметрах указывается только имя массива. Пример:

```
&VTKP(J=0.3;TK1,9/N;KP1);
```

,здесь задаче передается массив из 9-ти точек, начиная с TK1, а принимается массив из элементов окружностей, кол-во которых определяется выходным параметром N. В разделе данных макропроцедуры, вызывающей эту задачу, необходимо зарезервировать максимальное кол-во кривых KP199=0,0,0;.

В том случае, когда задаче необходимо передать макропроцедуре массив состоящий из более чем 200 элементов, то его необходимо формировать записями не более 200 элементов каждая.

Прием же параметров из почты в этом случае можно осуществлять только с помощью оператора ЧДФ/POST. Пример: предположим, что мы разработали задачу, которая на выходе формирует массив точек превышающий 200-ти элементов. Приведем фрагмент макропроцедуры, вызывающей эту задачу.

```
МАКРО=OPL;
X0=0;
X1=0;
L=2;
!
HM;
S/100; &PTK; * вызов задачи расчета;
* цикл чтения данных из файла почты;
ОФЧ/POST;ЦИКЛ;ПОКА L#0;ЧДФ/POST,L,X0;
ЕСЛИ L=0; ТО ВЫХОД;КЕ;
* перемещение по прочитанным точкам;
TK=X/X0,Y/X1;
КОНЕЦ;
ЗФ/POST;.....
KM;
!
```

В данном примере каждая запись в файле POST.DAT включает координаты только одной точки. Задаче PTK параметры не передаются.

Кроме файла почты POST.DAT обмен информацией между вызываемой задачей и макропроцедурой может осуществляется с "КОДА ВЫХОДА", формируемого на выходе из задачи.

"КОД ВЫХОДА" записывается процессором в переменную I0, при ее резервировании в разделе данных макропроцедуры и может использоваться для анализа ошибочных ситуаций, для целей построения алгоритма макропроцедуры и т.д.

Если резервирование I0 не произведено, то запись кода выхода игнорируется.

При запуске Препроцессора САПР-ЧПУ в режиме трассировки выдается на печать (в файл) сообщение о входе и выходе из задачи с соответствующим "КОДОМ ВЫХОДА".