



УСТРОЙСТВО
ЧИСЛОВОГО ПРОГРАММНОГО УПРАВЛЕНИЯ
NC-110, NC200, NC201, NC210, NC220, NC230

**Руководство создания постоянных циклов
визуального программирования.**

Руководство программиста

Санкт-Петербург
2005г

Содержание.

1.	Необходимые средства для создания циклов	3
2.	Схема создания циклов	3
3.	Создание файлов, определяющих цикл	3
3.1.	Создание файлов ууу.BMP и INDEX.BMP	3
3.1.1.	Создание файлов иконок ууу.BMP,	3
3.1.2.	Создание файлов-архивов иконок INDEX.BMP,	4
3.2.	Создание файла рисунка цикла (xxx.BMP)	5
3.2.1.	Создание рисунка цикла в формате BMP xxx_.BMP,	5
3.2.2.	Создание файлов-архивов иконок INDEX.BMP,	6
3.3.	Создание файла помощи цикла (xxx.HLP)	6
3.4.	Создание файла шаблона параметров цикла (xxx.TXT).....	7
3.5.	Создание файла помощи по каждому параметру цикла (xxx.INF).....	9
3.6.	Создание файла ТП цикла (xxx.)	10
3.6.1.	Команды черновой обработки фрезерных циклов.....	10
3.6.1.1.	Команда USL определяет величину припуска оставляемую вдоль боковой поверхности.	10
3.6.1.2.	Команда USP определяет величину припуска оставляемую по глубине.....	10
3.6.1.3.	Команда QIL отклонение от размера начала работы.....	10
3.6.1.4.	Пример ТП цикла с именем SGS:.....	10
4.	Встраивание циклов в ПО УЧПУ.....	13
4.1.	Структура каталогов и расположения файлов циклов.	13
4.1.1.	Полная структура каталогов и файлов циклов.....	13
4.1.2.	Упрощенная структура каталогов и файлов циклов.....	14
4.2.	Создание структуры каталогов и файлов циклов.	14
4.2.1.	Создание структуры каталога CYCLE.....	14
4.2.2.	Структура файла index.hlp	15

Введение.

В этом документе будет рассмотрена схема создания постоянных циклов визуального программирования (далее циклы).

1. Необходимые средства для создания циклов

Аппаратно-программный комплекс для создания циклов должен включать персональный компьютер (ПК). На ПК должна быть построена вся схема каталогов и файлов циклов, которая должна быть создана в УЧПУ в каталоге **CYCLE**. Схема каталогов и файлов показана на рис.1 и рис.2. Для разработки циклов ПК должен иметь следующие программы:

- Графический редактор, сохраняющий рисунки в формате Bit Map Picture (BMP).
- Программное обеспечение (ПО), поставляемое с УЧПУ:
 - Файл **USERBMP.EXE**, расположенный на дискете «COPYFLASH №0» в архивном файле **bmp32.rar** каталога **BMP32**.
 - Файл **CNCSYS.PAL**, расположенный в каталоге **C:\CNC32\MP0**.

Примечание.

- На ПК файлы **USERBMP.EXE** и **CNCSYS.PAL** для удобства можно разместить в корневом каталоге диска «С», обеспечив к ним доступ из любого каталога ПК.
- Для исключения потери цвета в палитре цветов в ПК необходимо использовать файл **CNCSYS.PAL**, расположенный в каталоге **C:\CNC32\MP0**.
- Если на ПК установлена операционная система WINDOWS XP, то необходимо изменить свойства программы **USERBMP.EXE**. Для этого установите маркер мыши на файл **USERBMP.EXE** и нажмите правую кнопку мыши. Откройте опцию «Свойства (Properties)»-«Память (Memory)»-«Дополнительная память (XMS)(Extended memory (XMS))». В окне «Дополнительная память (XMS)» установите объем не равным нулю, например значение: 16384.

Отладку, разработанных на ПК циклов, необходимо выполнить в УЧПУ.

2. Схема создания циклов

Схема создания циклов состоит из двух частей:

1. Создание файлов, определяющих цикл.
2. Встраивание цикла в ПО УЧПУ.

3. Создание файлов, определяющих цикл

Цикл определяют шесть обязательных файлов:

1. Иконка цикла (расширение файла **ууу.BMP**).
2. Рисунок цикла (расширение файла **xxx.BMP**).
3. Помощь по циклу (расширение файла **xxx.HLP**).
4. Шаблон параметров (расширение файла **xxx.TXT**).
5. Помощь по каждому параметру, определяющему цикл (расширение файла **xxx.INF**).
6. Технологическая программа (ТП) (файл без расширения **xxx**).

Примечание. Имена файлов «xxx», для одного цикла должны быть одинаковыми и состоять из трех букв, например: **SGS.BMP**, **SGS.HLP**, **SGS.TXT**, **SGS.INF**, **SGS**.

3.1. Создание файлов **ууу.BMP** и **INDEX.BMP**

На ПК необходимо создать структуру каталогов для размещения циклов, которая должна быть идентична структуре размещения циклов в конкретном УЧПУ (примеры структур см. на рис.1, или рис.2). В каждом созданном по структуре каталоге создаем файлы иконок **ууу.BMP** и файл-архив этих иконок **INDEX.BMP**. Для создания файла-архива иконок можно написать командный файл **INDEX.BAT**.

3.1.1. Создание файлов иконок **ууу.BMP**,

Файлы иконок **ууу.BMP**, где: **ууу** – индивидуальные имена иконок создаются с помощью графического редактора, например **PBRUSH.EXE** (штатный графический редактор OS WINDOWS).

Формат иконки в графическом редакторе: 72 x 63 пикселя x 256 цветов.

Максимальное количество иконок в каждом каталоге равно 255 шт.

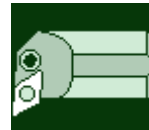
Иконка изображает элемент каталога. Элементами каталога могут являться:

1. Вложенные в него другие каталоги с соответствующими иконками, например:

- для каталога **FREZA** иконка с именем: **0FREZA.BMP**:



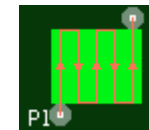
- для каталога **ТОК** иконка с именем: **0ТОК.BMP**:



- для каталога **SVERLO** иконка с именем: **0SVERLO.BMP**:



2. Размещенные в данном каталоге файлы циклов с соответствующими иконками, например для цикла с именем **SGS** иконка с именем: **0SGS.BMP**:



3. Во вложенных каталогах для возможности движения вверх по их дереву используется каталог с именем “..” с соответствующей иконкой для этого каталога, например:



Рекомендуемое имя данной иконки: **0000.BMP**

3.1.2. Создание файлов-архивов иконок **INDEX.BMP**,

Для создания файла-архива иконок **INDEX.BMP** каждого каталога в каждом из них сформируем командные файлы с именем **INDEX.BAT**.

Командные файлы с именем **INDEX.BAT** предназначены для преобразования всех файлов иконок **ууу.BMP** текущего каталога в единый файл-архив иконок этого каталога под именем **INDEX.BMP**, для чего с помощью файла **INDEX.BAT**:

1. Копируются все исходные файлы иконок соответствующего уровня меню **0000.BMP**, **0SGS.BMP** и другие, если таковые имеются, под именами **000.BMP**, **001.BMP** ...
2. Запускается программа **USERBMP.EXE**, которая создает архив иконок с именем **CNC.USR**.
3. Удаляется уже существующий файл **INDEX.BMP**.
4. Удаляются все файлы с именами из трех цифр, начинающиеся с цифры **0??..BMP..**
5. Переименовывается архив иконок **CNC.USR** под именем **INDEX.BMP**.

ВНИМАНИЕ. Перед запуском файла **INDEX.BAT** убедитесь, что в ПК используется последний созданный файл **CNC.SYS.PAL**.

ВНИМАНИЕ. Внимательно следите за датой и временем создания файла **CNC.SYS.PAL** до и после работы командного файла **INDEX.BAT**. Если время создания файла **CNC.SYS.PAL** обновилось после работы программы **USERBMP.EXE**, используемой в командном файле **INDEX.BAT**, то файл **CNC.SYS.PAL** необходимо скопировать в УЧПУ в каталог **C:\CNC\MP0**.

Файл **INDEX.BMP** должен быть скопирован в каталог УЧПУ, где размещены файлы данного цикла, например, с именами **SGS.BMP**, **SGS.HLP**, **SGS.TXT**, **SGS.INF**, **SGS**.

Файл **INDEX.BAT** для каталога **CYCLE**, в котором элементами являются вложенные каталоги **FREZA**, **ТОК**, **SVERLO**, строится по следующей схеме:

COPY 0FREZA.BMP 000.BMP – копирование иконки каталога **0FREZA.BMP** под порядковым именем **000.BMP**.

- COPY 0ТОК.BMP 001.BMP** – копирование иконки каталога **0ТОК.BMP** под порядковым именем **001.BMP**.
- COPY 0SVERLO.BMP 002.BMP**– копирование иконки каталога **0SVERLO.BMP** под порядковым именем **002.BMP**.
-
- USERBMP.EXE** -создание архива иконок с именем **CNC.USR**.
- DEL INDEX.BMP** -удаление ранее созданного файла **INDEX.BMP**.
- DEL 0??BMP** -удаление всех файлов с порядковыми именами **000.bmp, 001.bmp, ...**
- REN CNC.USR INDEX.BMP** -переименование архива иконок с именем **CNC.USR** под именем **INDEX.BMP**.

Файл **INDEX.BAT** для каталога, в котором элементами являются иконки циклов, строится по следующей схеме:

- [COPY 0000.BMP 001.BMP]** – копирование иконки цикла **0000.BMP** под порядковым именем **001.BMP**.
- COPY 0SGS.BMP 002.BMP** – копирование иконки цикла **0SGS.BMP** под порядковым именем **002.BMP**.
- COPY 0SCA.BMP 003.BMP** – копирование иконки цикла **0SCA.BMP** под порядковым именем **003.BMP**.
- COPY 0SGA.BMP 004.BMP** - копирование иконки цикла **0SGA.BMP** под порядковым именем **004.BMP**.
-
- USERBMP.EXE** - создание архива иконок с именем **CNC.USR**.
- DEL INDEX.BMP** -удаление ранее созданного файла **INDEX.BMP**.
- DEL 0??BMP** -удаление всех файлов с порядковыми именами **000.bmp, 001.bmp, ...**
- REN CNC.USR INDEX.BMP** -переименование архива иконок с именем **CNC.USR** под именем **INDEX.BMP**.

Примечание. Запись **COPY 0000.BMP 001.BMP** - необязательная в командном файле и устанавливается, если создаваемые иконки циклов расположены в подкаталогах. Запись предназначена для возможности движения вверх по уровням меню выбора цикла в редакторе ВП.

Иконки файла **INDEX.BMP** выводятся в поле 12 видеостраницы редактора ВП, см. рис.5 при выборе опции: «F7 Добавить виз.кадр»-«F1 Визуал.прогр-ие» основного редактора ВП, описанного в «Руководстве оператора. Визуальное программирование».

3.2. Создание файла рисунка цикла (xxx.BMP)

На ПК в каталоге разрабатываемого цикла создаем рисунок цикла в формате BMP с именем **xxx_.BMP** и командный файл для его преобразования в файл-архив рисунка с именем **xxx.BMP**:

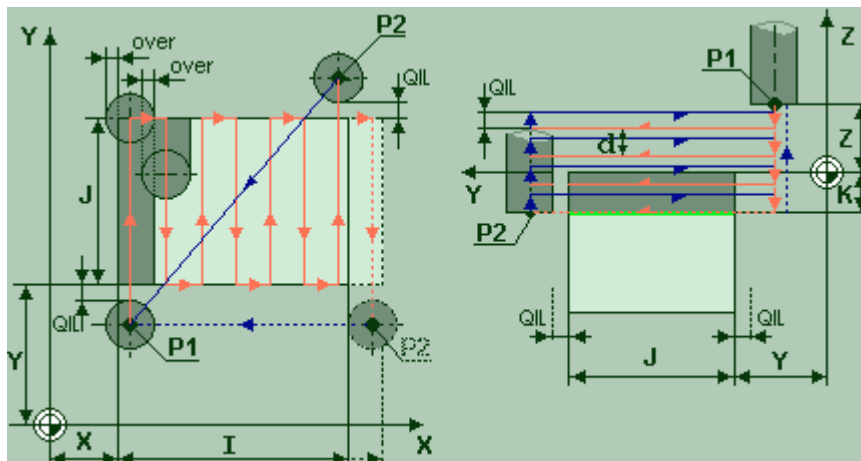
1.1.1. Создание рисунка цикла в формате BMP xxx_.BMP,

Рисунки **xxx.BMP**, где: **xxx.BMP** – индивидуальные имена рисунков циклов создаются с помощью графического редактора, например **PBRUSH.EXE** (штатный графический редактор OS WINDOWS).

Формат рисунка в графическом редакторе: 432 x 230 пикселя x 256 цветов.

Максимальное количество рисунков в каждом каталоге 255 шт.

Рисунок изображает схему цикла с указанием имен параметров, использованных в файле-шаблоне **xxx.TXT** к данному циклу, например рисунок цикла «SGS» **SGS.BMP**:



Для оператора рисунок цикла выводится в поле 2 видеостраницы редактора ВП, рис.6 при вводе значений параметров цикла.

3.2.2. Создание файлов-архивов иконок INDEX.BMP,

Для создания файла-архива рисунка каждого цикла сформируем командные файлы с именем **xxx_.BAT**.

Командные файлы с именем **xxx_.BAT** предназначены для преобразования файла-рисунка в формате BMP (**xxx_.BMP**) в файл-архив рисунка этого цикла под именем **xxx.BMP**, для чего с помощью файла **xxx_.BAT**:

1. Копируется файл-рисунок цикла **xxx_.BMP** под именем **000.BMP**.
2. Запускается программа **USERBMP.EXE**, которая создает файл-архив рисунка с именем **CNC.USR**.
3. Удаляется уже существующий файл **xxx.BMP**.
4. Удаляется файл с именем **000.BMP**.
5. Переименовывается файл-архив рисунка цикла **CNC.USR** под именем **xxx.BMP**.

ВНИМАНИЕ. Перед запуском файла **xxx_.BAT** убедитесь, что в ПК используется последний созданный файл **CNC.SYS.PAL**.

ВНИМАНИЕ. Внимательно следите за датой и временем создания файла **CNC.SYS.PAL** до и после работы командного файла **xxx_.BAT**. Если время создания файла **CNC.SYS.PAL** обновилось после работы программы **USERBMP.EXE**, используемой в командном файле **xxx_.BAT**, то файл **CNC.SYS.PAL** необходимо скопировать в УЧПУ в каталог **C:\CNC\MP0**.

Пример командного файла с именем **SGS_.BAT**:

```
;SGS
COPY SGS_.BMP 000.BMP
USERBMP.EXE 000.BMP
DEL SGS.BMP
REN CNC.USR SGS.BMP
```

Файл **SGS.BMP** должен быть скопирован в каталог УЧПУ, где размещены файлы данного цикла с именами **SGS.HLP**, **SGS.TXT**, **SGS.INF**, **SGS**, **INDEX.BMP**.

3.3. Создание файла помощи цикла (xxx.HLP)

На ПК, в каталоге разрабатываемого цикла, создаем файл помощи с именем **xxx.HLP**. Максимальная длина строки в файле **xxx.HLP** равна 51 символ. Для оператора текст файла выводится в поле 2 видеостраницы редактора ВП, рис.6 по кнопке «F1» (Помощь/Рисунок).

Пример файла помощи для цикла с именем **SGS.HLP**:

«Черновое многопроходное фрезерование по ширине и глубине прямоугольной плоскости торцевой фрезой.

Фрезерование выполняется вдоль ординаты.

Цикл выполняется при функциях G90,G40,G94,G97.

Возможно определить величины припуска и отклонения от размера начала работы путем использования следующих команд USP, QIL

При использовании данного цикла для чистовой обработки параметр d необходимо задать равным 0, в этом случае ранее заданный в данном цикле припуск не учитывается. Фрезерование контура выполняется по заданной глубине за один проход.

Параметры цикла:

X-координата, определяющая левый нижний угол прямоугольника по оси X относительно нуля детали.
Y-координата, определяющая левый нижний угол прямоугольника по оси Y относительно нуля детали.

Z-координата точки начала обработки по оси Z.
 I-длина стороны прямоугольника по оси X.
 J-длина стороны прямоугольника по оси Y.
 K-координата, определяющая глубину по оси Z.
 d-глубина одного прохода.

OVER-минимальная величина перекрытия между проходами, а также размер перекрытия между фрезой и торцами заготовки на первом и последнем проходе.

USP-определяет величину припуска, оставляемого по глубине.

QIL-отклонение от размера начала работы по оси Y, а также расстояние между инструментом и деталью по оси Z для перехода на новый проход по глубине.»

3.4. Создание файла шаблона параметров цикла (xxx.TXT)

В каталоге разрабатываемого цикла создаем файл шаблона его параметров xxx.TXT. Для оператора текст файла выводится в поле 1 видеостраницы редактора ВП, рис.6.

Структура файла **xxx.TXT**:

Формат первой строки файла **xxx.TXT**:
[num]&[kod]&[info]

где:

[num] - определяет число параметров постоянного цикла

[kod] - имя трехбуквенного кода цикла

[info] - краткая информация о цикле. Выводится в поле 4 видеостраницы редактора ВП, рис.6.

«&» - разделитель между параметрами.

Формат строк, начиная со второй:

**[тип параметра]&[имя параметра]&[значение по умолчанию]&
 [0-пустое значение недопустимо/1-допустимо]&[1-ось/0-не ось]&
 [{шаблонные условия;}|{значения фиксированного списка}]**

где:

[тип параметра] - определяется цифрой от 1 до 8:

1 - unsign char	- 8-битовое без знака
2 - int	- 16-битовое со знаком
3 - long	- 32-битовое со знаком
4 - float	- 32-битовое вещественное
5 - double	- 64-битовое вещественное
6 - фиксированный список	
7 - фиксированный список для контуров (в циклах не используется)	
8 - строковое	

[имя параметра] - имя параметра может состоять из 5 алфавитно-цифровых символов. Значащим символом является только первый.

[значение по умолчанию] - значение параметра, индицируемое при создании оператором нового кадра ВП

[0-пустое значение недопустимо/1-допустимо] если пустое значение параметра допустимо, то в соответствующую этому параметру переменную PL будет записано число 999999.

[1-ось/0-не ось] если параметр является осью, то значение данного параметра должно быть равно 1.

[{шаблонные условия;}|{значения фиксированного списка}]

1. шаблонные условия, сформированные для параметра, будут проверяться на истинность при вводе значения параметра и при вставке сформированного кадра ВП в УП. Шаблонные условия могут быть сформированы выражениями:

- с использованием логических операций:

==	равно
!=	не равно
>	больше
<	меньше
>=	больше равно

<= меньше равно

- с использованием функций:

sin()
 cos()
 tan()
 ars()
 arc()
 art()
 sqr()
 abs()
 int()
 neg()
 mod()

2. Ряд значений, сформированный для параметра типа «фиксированный список», будет индцироваться на экране при вводе его значения. Ввод значения осуществляется установкой маркера на нужное значение и последующего нажатия клавиши ENTER.

Пример:

13&F11&Цикл расточки

1&X1&0&0&1&-p1<12.34;3.24-(p1+p6)>(-p5)+p2

2&Y1&0&0&1&p2>1.23;p2-1>mod(p10,3)

3&Z1&1&0&1&p3>-100

4&X2&123.25&0&1&p4<(-1.23)

1&Y2&200&0&1&p5>0

4&Z2&100.25&0&1&abs(p6)<1

5&R1&0.2555&0&1&abs(p7)<1

4&R1&100.2&1&1&p8<1.23;p2-sin(p8)<ars(p7)

5&F1&-100.5555&0&1&p9<1.23

2&F2&-2589&0&1&p10>=1.23

6&f3&2&0&1&0;1;2;3;4;5

1&L& &1&1&p12<1.23

3&D&2&1&1&?F!=1:D>0:D==0

- фиксированный список (0;1;2;3;4;5)

Файл шаблона параметров определяет вид кадра управляющей программы (УП) с циклом, который формирует разработчик УП. При создании нового кадра УП с циклом в редакторе ВП для имени каждого параметра будет использован только первый символ из параметра [имя параметра] и его значение из соответствующего ему параметра [значение по умолчанию]. Для приведенного выше примера имена параметров и значения по умолчанию приведены в табл. 1.

Таблица заполнения параметров в ТП цикла данными из шаблона xxx.TXT.

Таблица 1.

Номер параметра	Имя параметра из файла шаблона	Имя параметра, используемое в кадре УП	Имя переменной «\$PSx» для использования в ТП, где «x» -порядковый номер параметра	Значение параметра по умолчанию	Имя переменной «PLx», для использования в ТП, где «x» -порядковый номер параметра
1	X1	X	\$PS1 (\$PS1="X")	0	PL1=0
2	Y1	Y	\$PS2 (\$PS2="Y")	0	PL2=0
3	Z1	Z	\$PS3 (\$PS3="Z")	1	PL3=1
4	X2	X	\$PS4 (\$PS4="X")	123.25	PL4=123.25
5	Y2	Y	\$PS5 (\$PS5="Y")	200	PL5=200
6	Z2	Z	\$PS6 (\$PS6="Z")	100.25	PL6=100.25
7	R1	R	\$PS7 (\$PS7="R")	0.2555	PL7=0.2555
8	R1	R	\$PS8 (\$PS8="R")	100.2	PL8=100.2
9	F1	F	\$PS9 (\$PS9="F")	-100.5555	PL9=-100.5555
10	F2	F	\$PS10 (\$PS10="F")	-2589	PL10=-2589
11	f3	f	\$PS11 (\$PS11="f")	2	PL11=2
12	L	L	\$PS12 (\$PS12="L")	пробел	PL12=пробел
13	D	D	\$PS13 (\$PS13="D")	2	PL13=2
					PL0=значение радиуса инструмента, из активного корректора уууу, заданного с функцией Тхххх.ууууМб в кадре предшествующем кадру цикла

Пример файла-шаблона для цикла с именем **SGS.TXT**:

```
10&SGS&ФРЕЗЕРОВАНИЕ
5&X&20&0&0
5&Y&20&0&0
5&Z&6&0&0
5&I&50&0&1&p4>0
5&J&33&0&1&p5>0
5&K&-6&0&1&p6<p3
5&d&3&0&1&p7>=0
4&OVER&0&0&1&p8>=0
5&USP&0&0&1&p9>=0
5&QIL&0&1&1&p10>=0
```

Кадры УП с именем PROBA/MP2, использующую цикла с именем **SGS** приведены в примере 1. В файле корректоров вводим диаметр фрезы: 1,Z0,K20 – диаметр фрезы 20 мм

Пример 1 УП «PROBA/MP2»

```
T1.1M6
G97G94S400M3F150
(SGS,X20,Y20,Z6,I50,J33,K-6,d3,O0,U,Q0)
```

Данные из кадра (**SGS,X20,Y20,Z6,I50,J33,K-6,d3,O0,U,Q0**), а также радиус фрезы из корректора 1 используются в ТП цикла с именем **SGS**. Для цикла **SGS** имена параметров и значения по умолчанию приведены в табл.2.

Таблица заполнения данных в примере, который использует цикл с именем **SGS**.

Таблица 2.

Номер параметра из файла шаблона SGS	Имя параметра из файла шаблона	Имя параметра, используемое в кадре УП	Имя переменной «\$PSx» для использования в ТП, где «x» - порядковый номер параметра	Значение параметра	Имя переменной «PLx», для использования в ТП, где «x» - порядковый номер параметра
1	X	X	\$PS1 (\$PS1="X")	20	PL1=20
2	Y	Y	\$PS2 (\$PS2="Y")	20	PL2=20
3	Z	Z	\$PS3 (\$PS3="Z")	6	PL3=6
4	I	I	\$PS4 (\$PS4="I")	50	PL4=50
5	J	J	\$PS5 (\$PS5="J")	33	PL5=33
6	K	K	\$PS6 (\$PS6="K")	-6	PL6=-6
7	d	d	\$PS7 (\$PS7="d")	3	PL7=3
8	OVER	O	\$PS8 (\$PS8="O")	0	PL8=0
9	USP	U	\$PS9 (\$PS9="U")	0	PL9=0
10	QIL	Q	\$PS10 (\$PS10="Q")	0	PL10=0
Значение в переменной PL0 содержит радиус инструмента из активного корректора					PL0=10

3.5. Создание файла помощи по каждому параметру цикла (xxx.INF)

В каталоге разрабатываемого цикла создаем файл помощи по каждому параметру с именем **xxx.INF**, которые определены в файле-шаблоне **xxx.TXT**. Количество строк этого файла должно соответствовать числу, указанному в поле [num] в файле-шаблоне **xxx.TXT**. Для оператора текст файла выводится в поле 3 видеостраницы редактора ВП, рис.6, по кнопке «F2» (Редакт. парам.).

Пример файла-помощи по каждому параметру с именем **SGS.INF**:

```
X Координата, определяющая левый нижний угол прямоугольника по оси X.
Y Координата, определяющая левый нижний угол прямоугольника по оси Y.
Z Координата точки начала обработки по оси Z.
I Длина стороны прямоугольника по оси X.
J Длина стороны прямоугольника по оси Y.
K Координата, определяющая глубину по оси Z.
d Глубина одного прохода.
OVER Минимальная величина перекрытия между проходами.
USP припуск оставляемый по глубине.
QIL Отклонение от размера начала работы.
```

3.6. Создание файла ТП цикла (xxx.)

В каталоге разрабатываемого цикла создаем файл ТП с именем xxx. (без расширения).

ТП может включать любые функции (G, M, S, T и т.д.) и макрокоманды (FIL, TGL, DFP, SPA, SPF и др.), используемые для составления УП и представленные в руководстве программиста MC/TC, в том числе, ТП широко использует параметрическое программирование.

3.6.1. Команды черновой обработки фрезерных циклов

Для реализации фрезерных циклов введены новые команды:

USL

USP

QIL

3.6.1.1. Команда USL определяет величину припуска оставляемую вдоль боковой поверхности.

Команда USL обычно программируется в ТП при черновой обработке, но может быть задана с клавиатуры.

Формат команды:

USL=величина

Где:

Величина - значение припуска оставляемого вдоль боковой поверхности. Значение может быть задано явным или неявным способом параметром E формата LR и выражено в единицах измерения размеров. По умолчанию значение USL = 0

3.6.1.2. Команда USP определяет величину припуска оставляемую по глубине.

Команда USP обычно программируется в ТП при черновой обработке, но может быть задана с клавиатуры.

Формат команды:

USP=величина

Где:

Величина - значение припуска оставляемого по глубине. Значение может быть задано явным или неявным способом параметром E формата LR и выражено в единицах измерения размеров. По умолчанию значение USP = 0

3.6.1.3. Команда QIL отклонение от размера начала работы.

Команда QIL обычно программируется в ТП при черновой обработке, но может быть задана с клавиатуры.

Формат команды:

QIL=величина

Где:

Величина - значение может быть задано явным или неявным способом параметром E формата LR и выражено в единицах измерения размеров. По умолчанию значение USP = 0

3.6.1.4. Пример ТП цикла с именем SGS:

```

;----- SGS -----
;ЧЕРНОВОЕ ФРЕЗЕРОВАНИЕ ПРЯМОУГОЛЬНОЙ ПЛОСКОСТИ
G94 G97 G40 G90
;(X) Координата определяющая левый нижний угол прямоугольника по оси X. $PS1="X"
E30=PL1
;(Y) Координата определяющая левый нижний угол прямоугольника по оси Y. $PS2="Y"
E31=PL2
;(Z) Координата точки начала отработки по оси Z. $PS3="Z"
E32=PL3
;(I) Размер прямоугольника вдоль X
E33=PL4
;(J) Размер прямоугольника вдоль Y
E34=PL5
;(K) Координата определяющая глубину по оси Z.
E35=PL6
;(d) Глубина одного прохода
E39=PL7
;(O) Величина перекрытия между двумя соседними проходами
E37=PL8
;(R) Радиус фрезы

```

```

E38=PL0
;(USP) Припуск по глубине
E49=PL9
;(QIL) Отклонение от размера начала работы
E48=PL10
(BNE,$PS11," ",9)
E48=1
"9"
; -----
; ----- Чистовая обработка d=0 -----
(BEQ,E39,0,16)
(BNC,17)
"16"

E1=1
E49=0
"17"
;
; ----расчет количества проходов-----
;E57-Глубина по Z
E57=ABS(E32-E35)-E49
(BEQ,E39,0,6)
;E1-Расчет целого количества проходов
E1=INT(E57/E39)
;E41-Расчет остатка, мм
E58=E57-(E1*E39)
;E59-10% глубины одного прохода
E59=E39*0.1
;Если остаток <=10% глубины одного прохода переход к "6", если > прибавляю 1 проход
(BLE,E58,E59,6)

E1=E1+1
"6"
;выбор плоскости интерполяции $PS1="X", $PS2="Y"
(DPI,$PS1,$PS2)
; ----- ФРЕЗЕРОВАНИЕ ВДОЛЬ Y -----
; ----- РАСЧЕТ -----
E40=E33+(E37*2)
E41=E34
E77=E38+E48
E50=NEG(E77)
E51=E38-E37
E42=(E38*2)-E37
E43=E41+E38
E44=E40-(E38*2)
(BLE,E44,0,V3)
E2=INT(E44/E42)+1
E45=E44/E2
(BNC,V4)
"V3"
E2=0
E45=0
"V4"
E46=(E32-E35-E49)/E1
E47=E32
E43=E43+E48
;E2=E2-1
; ----- ДВИЖЕНИЕ -----
;смещение нуля детали
(UIO,$PS1E30,$PS2E31)
G0 $PS1E51 $PS2E50
$PS3E32
(RPT,E1)
G0 G90 $PS1E51 $PS2E50
E47=E47-E46
G1 $PS3E47
E41=ABS(E41)
E43=ABS(E43)
G1 G91 $PS2E77

```

(BEQ,E2,0,111)
(RPT,E2)
\$PS2E41
\$PS1E45
E43=NEG(E43)
E41=NEG(E41)
(ERP)
"111"
;E43=NEG(E43)
;E41=NEG(E41)
;\$PS1E45
\$PS2E43
E47=E47+E48
G0 G90 \$PS3E47
E47=E47-E48
(ERP)
G0 G90 \$PS3E32
\$PS1E51 \$PS2E50
"END"
E30=NEG(E30)
E31=NEG(E31)
;возврат к нулю детали
(UIO,\$PS1E30,\$PS2E31)

4. Встраивание циклов в ПО УЧПУ.

4.1. Структура каталогов и расположения файлов циклов.

4.1.1. Полная структура каталогов и файлов циклов.

Полная структура каталогов и файлов циклов выполняется по схеме представленной на рис.1.

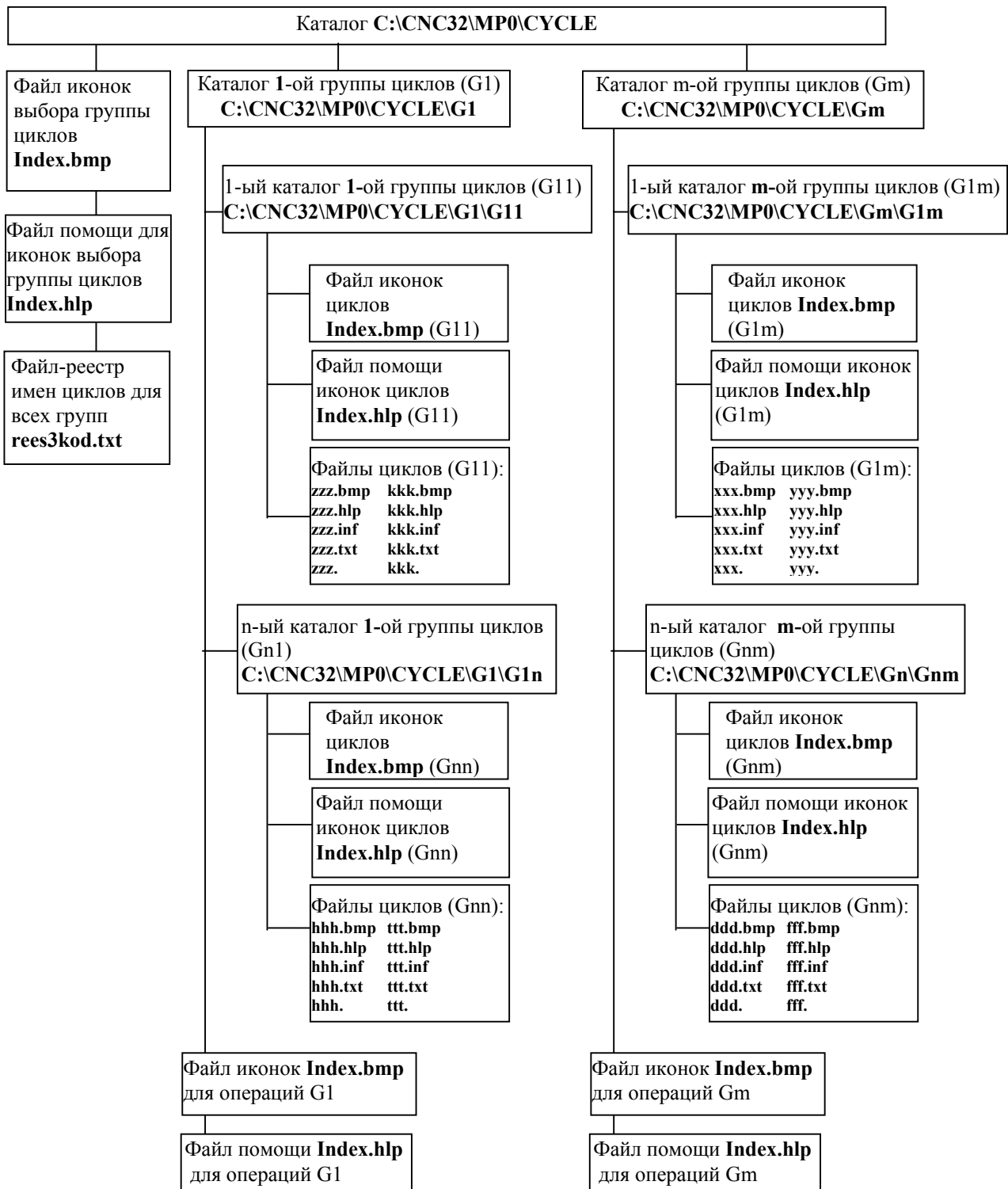


Рис.1

4.1.2. Упрощенная структура каталогов и файлов циклов.

Упрощенная структура каталогов и файлов циклов выполняется по схеме представленной на рис.2.

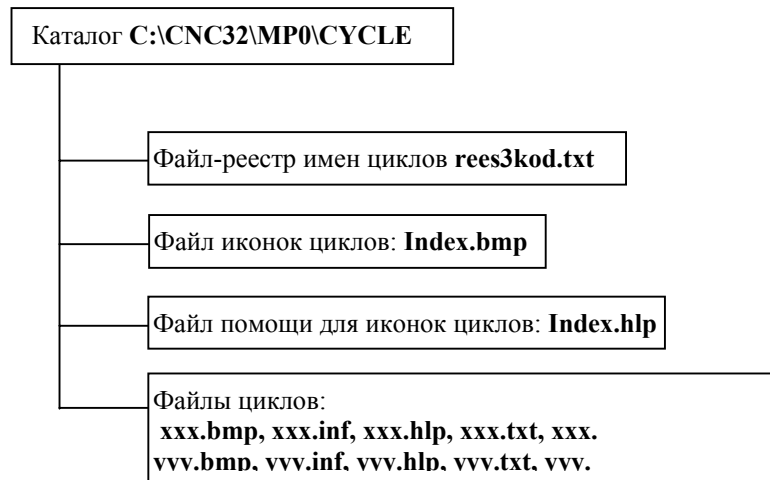


Рис.2

4.2. Создание структуры каталогов и файлов циклов.

4.2.1. Создание структуры каталога CYCLE.

1. Создаем каталог C:\CNC32\MP0\CYCLE

Если для разработки УП конкретного станка достаточно одной группы циклов, то файлы всех циклов можно расположить непосредственно в каталоге C:\CNC32\MP0\CYCLE –упрощенная схема, см. рис.2.

Если для разработки УП станка недостаточно одной группы циклов, то файлы циклов для удобства можно расположить в подкаталогах каталога C:\CNC32\MP0\CYCLE, - см. схему на рис.1, например:

- C:\CNC32\MP0\CYCLE\FREZA фрезерные циклы
- C:\CNC32\MP0\CYCLE\ТОК -токарные циклы
- C:\CNC32\MP0\CYCLE\SVERLO -сверлильные циклы

2. Создаем файл-реестр циклов с фиксированным именем rees3kod.txt. Файл должен быть расположен в каталоге C:\CNC32\MP0\CYCLE. Этот файл имеет два назначения:

- В первой строке указывается местоположение каталога GTL от каталога MP0, используемого в редакторе ВП при задании свободного профиля. Эта запись не относится к созданию цикла.

Полный путь на диске «С» к каталогу GTL: C:\CNC32\MP0\CONTUSER\GTL, поэтому обычно эта запись имеет следующий вид:

GTL\CONTUSER\GTL

- Начиная со второй строки, указываются имена циклов с указанием к ним полного пути от каталога CYCLE, например:

SGS\CYCLE\FREZA

ВНИМАНИЕ. Все буквы в файле rees3kod.txt должны быть заглавными.

3. В каталоге CYCLE и в каждом каталоге, вложенном в этот каталог, например FREZA, ТОК, SVERLO создаем файлы:

- | | |
|------------------|---|
| index.bmp | Файл является файлом иконок циклов или иконок каталогов, в которых расположены циклы, например FREZA, ТОК, SVERLO. Создание этого файла приведено в п.3.1.1, п.3.1.2. |
| index.hlp | Файл содержит список элементов, которые находятся в данном каталоге, и имеют соответствующие иконки в файле index.bmp. |

4.2.2. Структура файла **index.hlp**

Первая строка файла **index.hlp** определяет:

1. Число элементов каталога.
2. Название группы циклов или каталогов.

Структура первой строки файла **index.hlp**:

[num] %[название группы циклов или каталогов]

где:

[num] - количество элементов в данном каталоге.
 [название группы циклов] - название группы циклов из 21 символа. Выводится в поле 11 видеостраницы редактора ВП, рис.5.
 Символ «%» - является разделителем между параметрами.

Примечание. Количество элементов в файле **index.hlp**, начиная со второй строки, должно быть равно количеству иконок в файле **index.bmp**, который расположен в текущем каталоге.

Структура строк файла **index.hlp** начиная со второй:

[../имя вложенного каталога / имя цикла] %[1-каталог/2-файл] %[имя], где:

- значения первой скобки []

[..] - имя каталога для возможности движения вверх по дереву вложенных каталогов, например:
 ../%1%Возврат в предыдущее меню

или

[имя вложенного каталога] - например :
 FREZA%1%Фрезерные циклы

или

[имя цикла] - имя цикла это имя файла, например:
 SGS%2%SGS

- значения второй скобки []:

[1] - записывается, если в первом параметре указано имя каталога
 [2] - записывается, если в первом параметре указано имя файла цикла

- значения третьей скобки []

[имя] - имя каталога или цикла. Выводится в поле 10 видеостраницы редактора ВП, рис.5. Макс. длина строки подсказки 71 символ, например:
 «Фрезерные циклы»

Пример файла **index.hlp**, определяющий выбор группы циклов, содержащихся с каталогах с именами **FREZA, TOK, и SVERLO**:

```
3% ВЫБОР ЦИКЛА
TOK%1%Токарные циклы
FREZA%1%Фрезерные циклы
SVERLO%1%Циклы сверления
```

Пример файла **index.hlp**, определяющий конкретные циклы из группы циклов каталога **FREZA**

```
2% ФРЕЗЕРОВАНИЕ
../%1%Возврат в предыдущее меню
SGS%2%SGS:Фрезерование плоскости параллельно оси Y
```

Пример структуры каталогов и файлов для двух циклов приведен на рис.3.

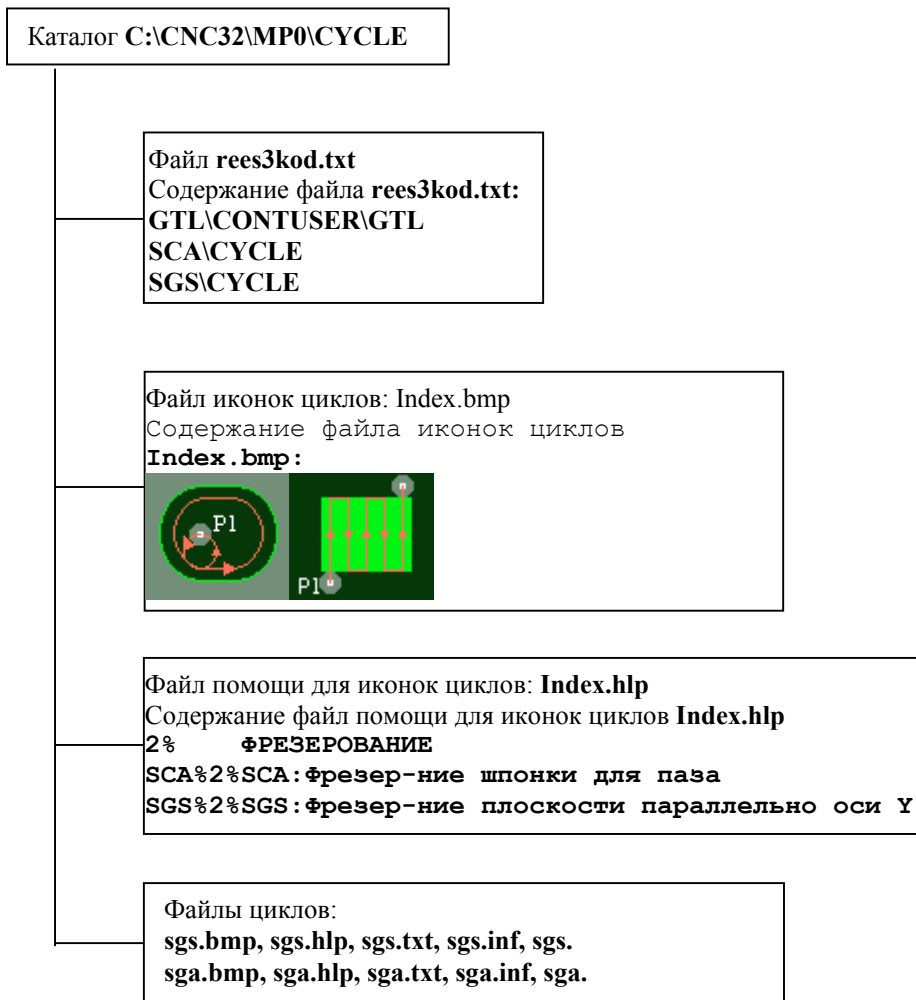


Рис.3

Пример структуры каталогов и файлов для нескольких групп циклов приведен на рис.4.

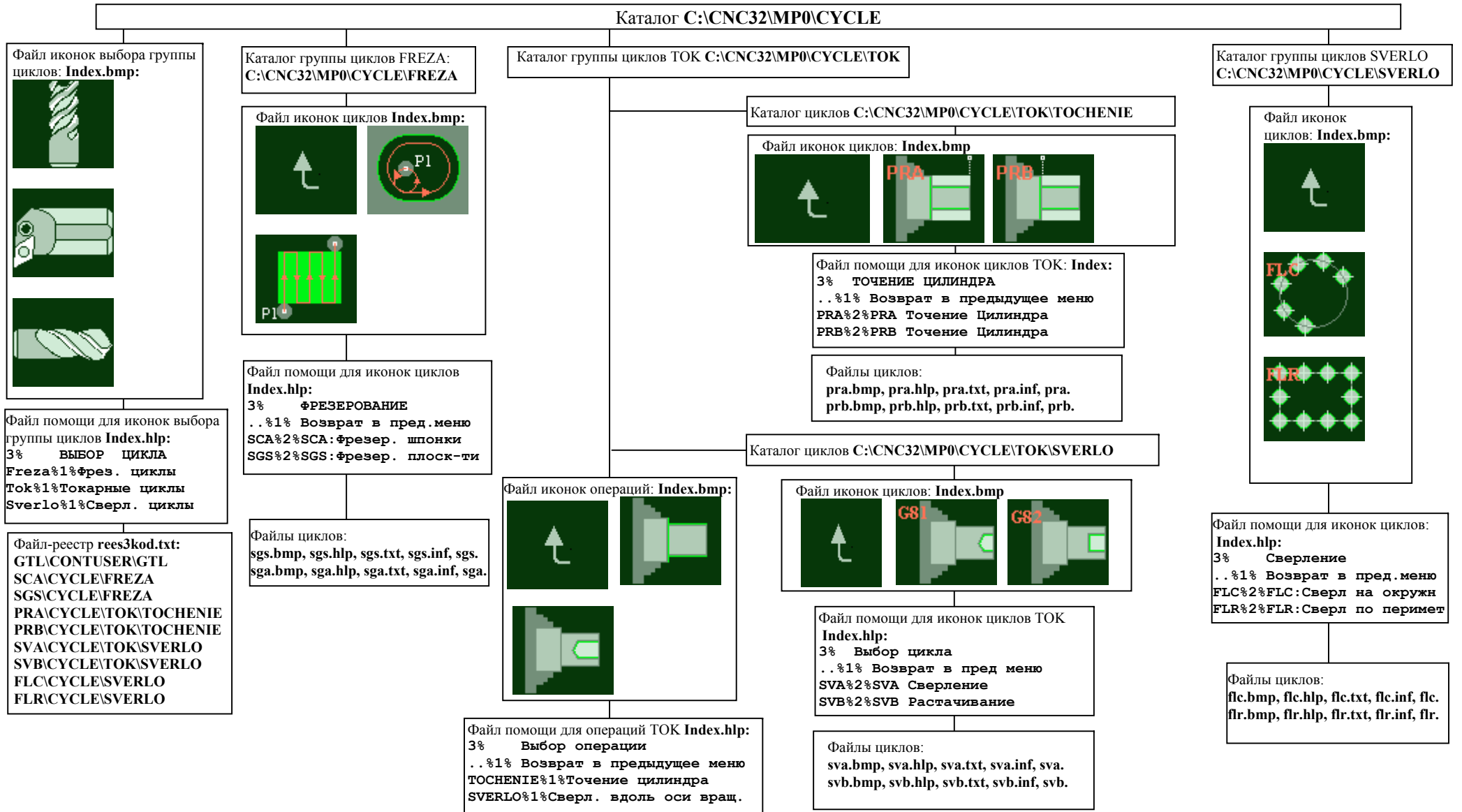


Рис.4

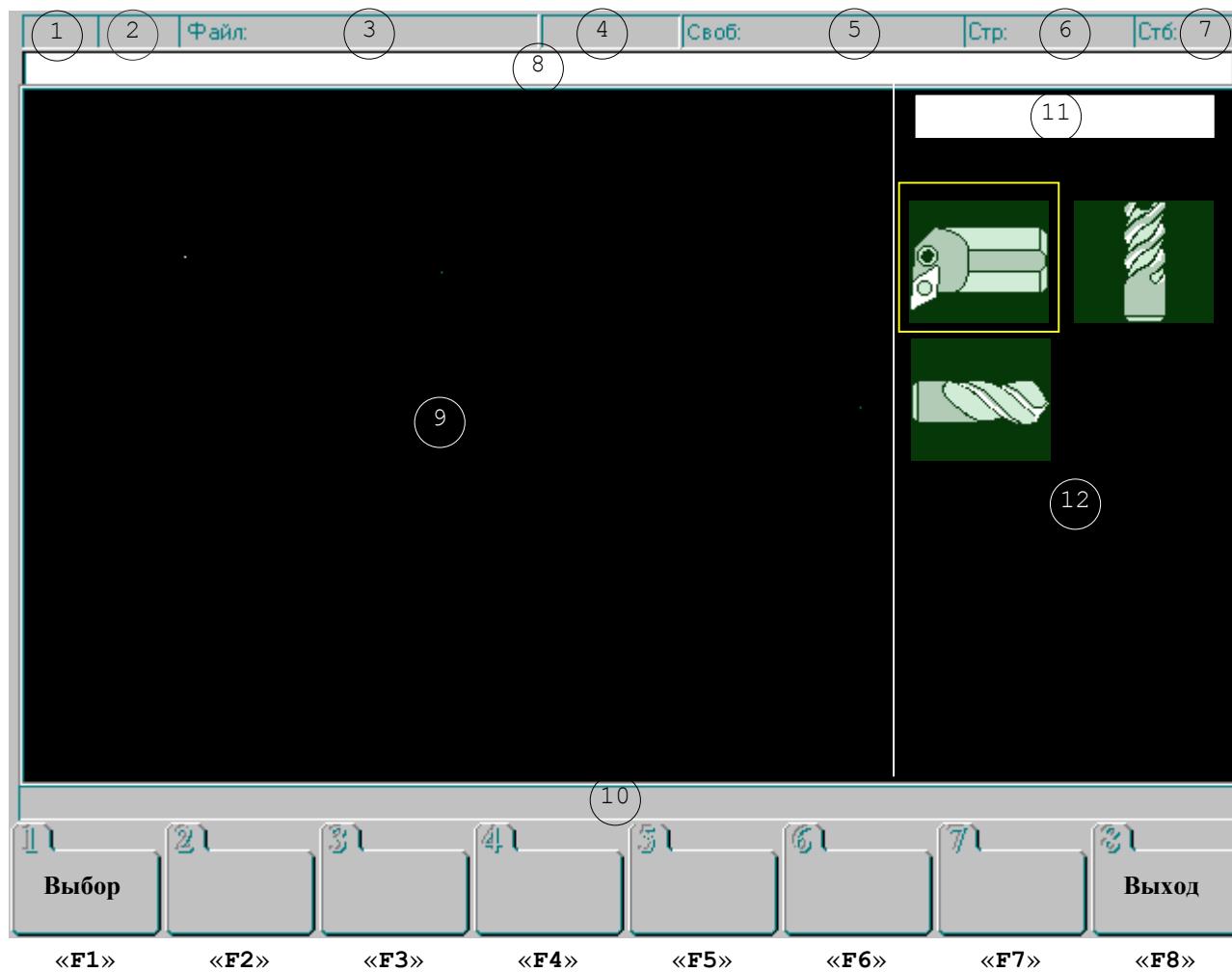


Рис.5

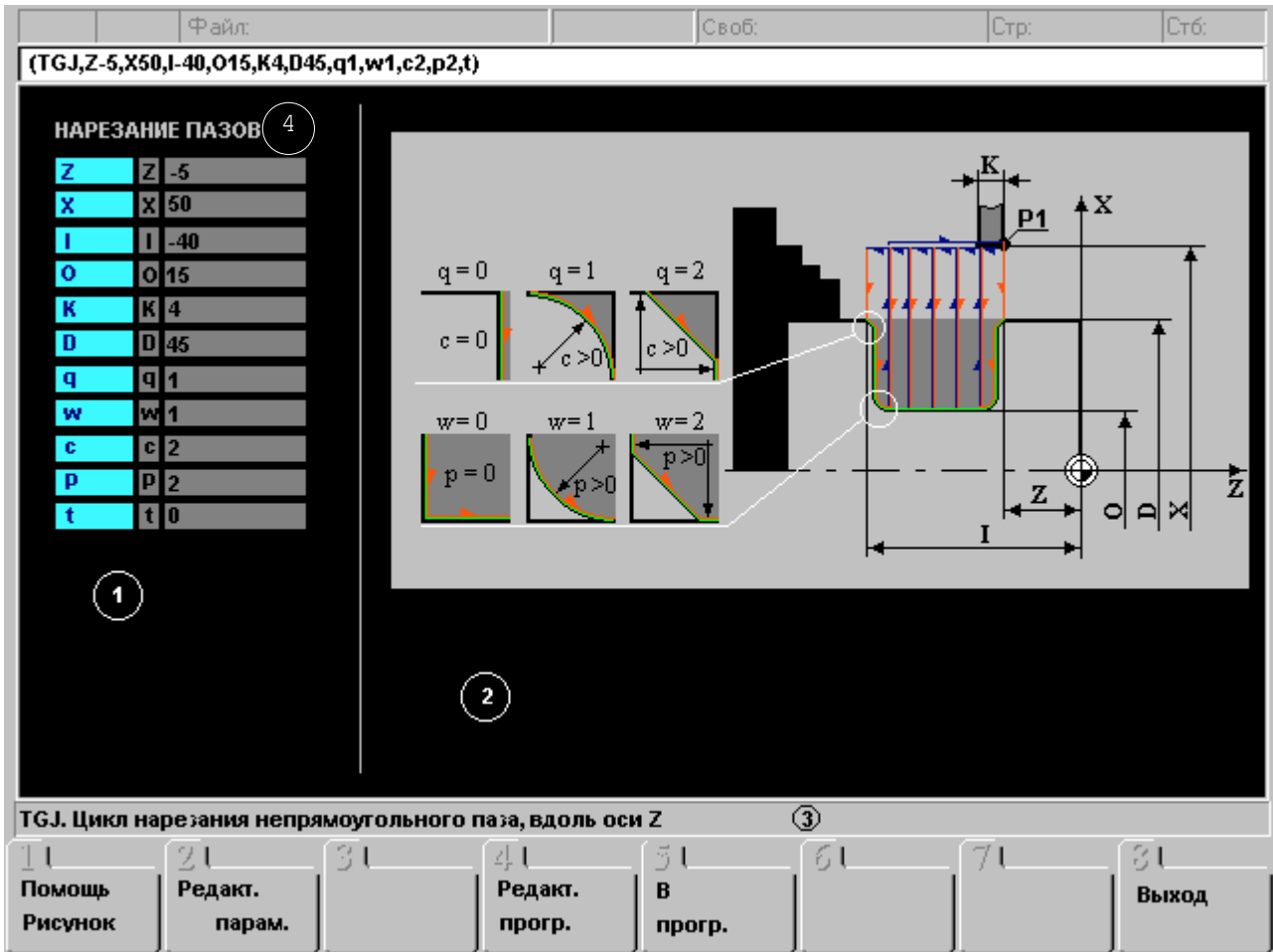


Рис.6

Сокращения

ПО	- программное обеспечение
УП	- управляющая программа
ТП	- технологическая программа
ПК	- персональный компьютер
УЧПУ	- устройство числового программного управления
ВП	- визуальное программирование

Таблицы

Таблица 1	- заполнение параметров ТП цикла данными из шаблона xxx.TXT.
Таблица 2	- заполнение данных в примере, использующий цикл с именем SGS.

Рисунки

Рисунок 1	- схема полной структуры каталогов и файлов циклов.
Рисунок 2	- схема упрощенной структуры каталогов и файлов циклов.
Рисунок3	- пример структуры каталогов и файлов для двух циклов.
Рисунок4	- пример структуры каталогов и файлов для нескольких групп циклов.