



УСТРОЙСТВО
ЧИСЛОВОГО ПРОГРАММНОГО УПРАВЛЕНИЯ
серии NC-110, NC200

**РУКОВОДСТВО ПРОГРАММИСТА МС/ТС.
ПРОГРАММИРОВАНИЕ В ПРОЦЕССАХ**

Санкт-Петербург
2002г.

СОДЕРЖАНИЕ

1. ПАРАЛЛЕЛЬНАЯ СИНХРОННАЯ РАБОТА С НЕСКОЛЬКИМИ ПРОЦЕССАМИ	3
2. РЕЖИМЫ СИНХРОНИЗАЦИИ МЕЖДУ ПРОЦЕССАМИ	6
2.1. УСЛОВНОЕ ОЖИДАНИЕ ПРОЦЕССА	6
2.2. ВЗАИМНОЕ ОЖИДАНИЕ ПРОЦЕССА	7
3. СХЕМА СИНХРОНИЗАЦИИ ДЛЯ ТРЕХ ПАРАЛЛЕЛЬНЫХ ПРОЦЕССОВ	8
4 ПРИМЕРЫ ПРОГРАММИРОВАНИЯ	10

1. Параллельная синхронная работа с несколькими процессами

Для параллельного управления несколькими процессами (до пяти) вводятся следующие трехбуквенные коды: *EHE*, *REL*, *SND*, *WAI*.

1.1. Код *EHE* загружает и запускает выполнение указанной программы под управлением ранее выбранного процесса.

Формат:

(EHE, n, ИМЯ ПРОГРАММЫ/MPx) ,

где:

- **n** - номер процесса: цифровая константа или параметр типа *VU*;
- **/MPx** - имя запоминающего устройства (x - 0-3), если оно отличается от имени запоминающего устройства заявленного по умолчанию в файле *PGCFIL*;
- **ИМЯ ПРОГРАММЫ** - наименование программы.

Пример:

(EHE, 2, Prog1)

или:

E4=2
(EHE, E4, Prog1/MP2)

1.2. Код *REL* выгружает выполняемую программу, загруженную ранее с кодом *SPG* или *EHE*. Действие кода *REL* из управляющей программы аналогично действию трехбуквенного кода *REL*, выполненного с клавиатуры.

Формат:

(REL)

1.3. Код *WAI* может иметь две функции.

1) Прекращает выполнение программы в процессе и заставляет данный процесс ждать команды повторного старта, подаваемой другим процессом.

Формат:

(WAI, n) ,

где:

- **n** - номер процесса, дающего команду повторного старта; цифровая константа или параметр типа BY.

Пример:

(WAI, 3)

или

E1=3
(WAI, E1)

2) Устанавливает процесс, который выполняет команду ожидания до тех пор, пока специфическая переменная не примет требуемого значения.

Формат:

(WAI, Var=ЗНАЧЕНИЕ) ,

где:

- **Var** - определяет наименование переменной системы типа BL, BY или IN;
- **ЗНАЧЕНИЕ** - определяет ожидаемые значения (цифра или E-параметр).

Пример:

(WAI, SA10.BY=2) ,

или

(WAI, SK2.IN=E11) ,

или

(WAI, SYVAR2=E5) .

Примечания

1. Если переменные типа BY или IN, то блок WAI может включать следующие операторы:

- > больше чем;
- < меньше чем;
- # отличный от.

Пример:

(WAI, SYVAR>5)
(WAI, SK2>E8)
(WAI, SA10.BY#2)

2. Переменные SYVAR, SA, и SK являются общими для всех конфигурируемых процессов.

3. Код WAI может быть запрограммирован в любом процессе.

1.4. Код SND дает команду повторного старта процессу, находящемуся в состоянии ожидания.

Формат:

(SND, n) ,

где:

- **n** - номер процесса, которому посылается команда повторного старта; цифровая константа или параметр типа ВУ.

Коды *SND* могут быть запрограммированы в любом процессе, при этом система проверяет, находится ли процесс в состоянии ожидания *WAI*, а также, соответствуют ли друг другу данный процесс и процесс, дающий команду повторного старта.

Пример:

(SND, 4)

или

E5=4

(SND, E5)

Примечание - Для того, чтобы синхронизировать *SND* с движением осей, блок должен включать #.

Пример:

N25#(SND, 2) .

Для вывода на экран во время работы каждый процесс имеет доступ к видеокадру, выбранному функциональной клавишей «P1» или «P3», при этом на экран выводится информация, относящаяся к данному состоянию процесса.

Видеокадр #0 используется для того, чтобы вывести на экран состояния всех процессов.

Доступ к видеокадру #6 имеет только тот процесс, из которого он первично задан.

2. Режимы синхронизации между процессами

2.1. Условное ожидание процесса

Условное ожидание процесса: один из процессов ожидает выполнение части программы другим процессом.

1) Команда повторного старта дается кодом *SND*.

Пример:

<u>ПРОЦЕСС 1</u>	<u>ПРОЦЕСС 2</u>
.....
N20.....	N105
N21 (<i>SND, 2</i>)	N106 (<i>WAI, 1</i>)
M22.....	N107.....
.....

Процесс 2 закончил кадр *N106*, когда процесс 1 выполняет команду (*SND, 2*) в кадре *N21*.

2) Команда повторного старта для процесса ожидания подается сигналом *вх/вых*.

Пример:

<u>ПРОЦЕСС 1</u>	<u>ПРОЦЕСС 2</u>
.....
N130.....	N50.....
N131 (<i>WAI, SA12=1</i>)	N51.....
N132.....

Процесс 1 заканчивает кадр *N131*, когда бит 12 в пакете *SA* равен «1».

3) Команда повторного старта дается сигналом логики.

Пример:

<u>ПРОЦЕСС 1</u>	<u>ПРОЦЕСС 2</u>
.....
N146.....	N200.....
N147 (<i>WAI, SK250=8</i>)	N201.....
N148.....

Процесс 1 заканчивает кадр N147, когда байт 250 пакета «К» равен 8.

4) Команда повторного старта дается, когда принимается значение ожидаемой переменной SYVAR.

Пример:

<u>ПРОЦЕСС 1</u>	<u>ПРОЦЕСС 2</u>
.....
N150.....	N120.....
N151 E8=10.3+20.7	N121 (WAI, SYVAR2=31)
N152 SYVAR2=E8	N122.....
N153.....	N123.....
.....

Процесс 2 заканчивает кадр N121, когда переменная SYVAR2 принимает значение 31 в кадре N152 процесса 1 (E8 = 31).

2.2. Взаимное ожидание процесса

Пример:

<u>ПРОЦЕСС 1</u>	<u>ПРОЦЕСС 2</u>	<u>ПРОЦЕСС 3</u>
.....
N107.....	N230.....	N330.....
N108 (WAI, 2)	N231 (SND, 1)	N331 (SND, 1)
N109 (WAI, 3)	N232 (WAI, 1)	N332 (WAI, 1)
N110 (SND, 2)	N233.....	N333.....
N111 (SND, 3)
N112.....

Три процесса ждут друг друга взаимно до выполнения кадров N112 (процесс 1), N233 (процесс 2), N333 (процесс 3) одновременно.

3. Схема синхронизации для трех параллельных процессов

Пример:

<u>ПРОЦЕСС 1 (PROG91)</u>	<u>ПРОЦЕСС 2 (PROG92)</u>	<u>ПРОЦЕСС 3 (PROG93)</u>
"START"		
N1 (EXE, 2, PROG92)	N1T20.20M6S...F.	N1 (WAI, 1)
N2 (EXE, 3, PROG93)	N2GX...Y...Z..	N2T30.30M6S...F.
N3T1.1M6S...F...	N3G2X..Y..I..J..
N4G1...X...Y.....
.....	N15GXY	N19GXY.....
N24 (SND, 2)	N16 (WAI, 1)	N20 (WAI, SYVAR, 1)
.....
.....
N35 (SND, 3)	N80 (SND, 1)
.....	GXY	N81.....
.....	N40 (WAI, SA10=1)
.....
.....
N59E5=10
N60SYVAR1=E5	N85 (SND, 1)
.....	N86.....
N66 (WAI, 2)
N67 (WAI, 3)
N68.....
.....
N76GXY
N77 (WAI, 2)	N98GXYZ.....	N93 XYZ.....
N78 (WAI, 3)	N99 (SND, 1)	N94 (SND, 1)
N100 (BNC, START)	M100 (REL)	N95 (REL)

В данном примере программа PROG91, рассматриваемая в качестве главной программы, активизируется командой SPG в процессе 1. Программы PROG92 и PROG93 запускаются в кадрах N1 и N2 программы PROG91.

Процесс 2 выполняет кадры до N15, а затем ожидает процесс 1, чтобы дать ему команду повторного старта, т.е. выполнить кадр N24.

Процесс 3 не начинает выполнять программу, пока процесс 1 не выполнит кадр N35. Команды WAI также могут быть входом/выходом, логическим сигналом или переменными SYVAR.

В данном примере процесс 2 ожидает (на кадре N40), чтобы бит 10 в пакете SA стал равным 1 для повторного старта.

Процесс 3 ожидает на кадре N20, чтобы процесс 1 на кадре N60 дал ему повторный старт, т.к. переменная SYVAR приняла значение 10.

Процесс 1 ожидает на кадрах N66-N67 команду рестарта с других процессов.

На кадрах N68-N86-N81 три процесса синхронизируются на повторном старте.

Главный процесс 1 возвращается к началу программы (BNC, START), когда два параллельных процесса 2 и 3 закончены.

4 Примеры программирования

1) Программирование трех синхронизированных процессов, один из которых рассматривается как главный.

```
N1 (EXE, 2, PR98)
N2 (EXE, 3, PR99)
N3T1.1M6S1000F500
.....
N6G1X...Y...
N7 (SND, 3)
.....
.....
N60 (WAI, 2)
N61 (WAI, 3)
N62 (EXE, 2, PR100)
N63 (EXE, 3, PR101)
.....
N200M30
```

PR90 Главная программа
(процесс 1)

```
N1T10.10M6S500F800
N2GX...Y...
.....
.....
.....
N72 (SND, 1)
N73 (REL)
```

PR98
(процесс 2)

```
N1 (WAI, 1)
N2T14.14M6S1000F600
.....
.....
.....
.....
N68GXY
N69 (SND, 1)
N70 (REL)
```

PR99
(процесс 3)

```
N1T18.18M6S400F300  
.....  
.....  
.....  
N140 (REL)
```

```
PR100  
(процесс 2)
```

```
N1T21.21M6S2000F500  
.....  
.....  
.....  
N138 (REL)
```

```
PR101  
(процесс 3)
```

Примечания

1. Для активизации главной программы (например, PR90) из процесса (например, процесса 1) нажмите «P3», а затем введите: SPG,PR90.
2. Связь главной программы в процессе 1 с любой другой программой может быть автоматически осуществлена с помощью инструкции EXE из программы PR90.
3. Для выполнения программы используйте следующую процедуру:
 - выберите видеокادر «СИСТЕМА»;
 - установите все процессы в режим «АВТОМАТИЧЕСКИЙ» («AUTO»);
 - нажмите клавишу «ПУСК».
4. Введенный в память поиск не может быть осуществлен на программах, содержащих инструкции синхронизации (EXE, WAI, SND).

2) Программирование трех независимых (не синхронизированных) процессов приведено в таблице 1.

Таблица 1

Программа	Номер программы (процесса)	Комбинирование процесс-программы с клавиатуры
N1T1.1M6S3000 N2GXУ N3G1G41X100Y50F500 N4G2X...Y...I...J... N99G40X10Y N100GXУM30	ПРОГРАММА 10 (ПРОЦЕСС 1)	«P3» выбирает процесс 1 SPG, PROG10
N1T1.1M6S400 N2G81R2Z-50F500 N3X-20Y10 N40G80XYZM30	ПРОГРАММА 11 (ПРОЦЕСС 2)	«P3» выбирает процесс 2 SPG, PROG11
N1T1.1M6S800 N2GXУ N34GXУZ	ПРОГРАММА 12 (ПРОЦЕСС 3)	«P3» выбирает процесс 3 SPG, PROG12

Примечания

1. «ПУСК» может быть использован двумя способами:
 - 1 - видеокادر «СИСТЕМА»; если все процессы находятся в «АУТО», то выбранные программы делают автоматический старт;
 - 2 - видеокادر «ПРОЦЕССn»; каждая программа делает независимый старт в выбранном режиме.
2. Введенный в память поиск выполняется для каждого процесса независимо.

