

SIMATIC S7-200Советы

Группа	Тема
3	Использование XMT и RCV команды для связи с интерфейсом оператора

CPU – необходимые для этого совета											
CPU 210	<input type="checkbox"/>	CPU 212	<input type="checkbox"/>	CPU 214	<input type="checkbox"/>	CPU 215	<input checked="" type="checkbox"/>	CPU 216	<input checked="" type="checkbox"/>	ДРУГИЕ	<input type="checkbox"/>

Обзор

Этот пример объяснет, как использовать XMT и RCV команды для связи с интерфейсом оператора. Программа S7-200 выполняется на S7-216, связанном с PC (выполняя программу терминала) через PPI кабель. XMT команда используется, чтобы передать меню команды и информацию об ошибке. RCV команда используется, чтобы получить команды от программы терминала. Эта программа поддерживает две команды ON и OFF. Если введен ON, включаются первые 8 выводов на S7-216. Если выполнен OFF выключаются первые 8 выводов .

Обратите внимание: При этой записи RCV команда поддерживается только на 0 порте CPU S7-215 и 0 и 1 портах CPU.S7-216

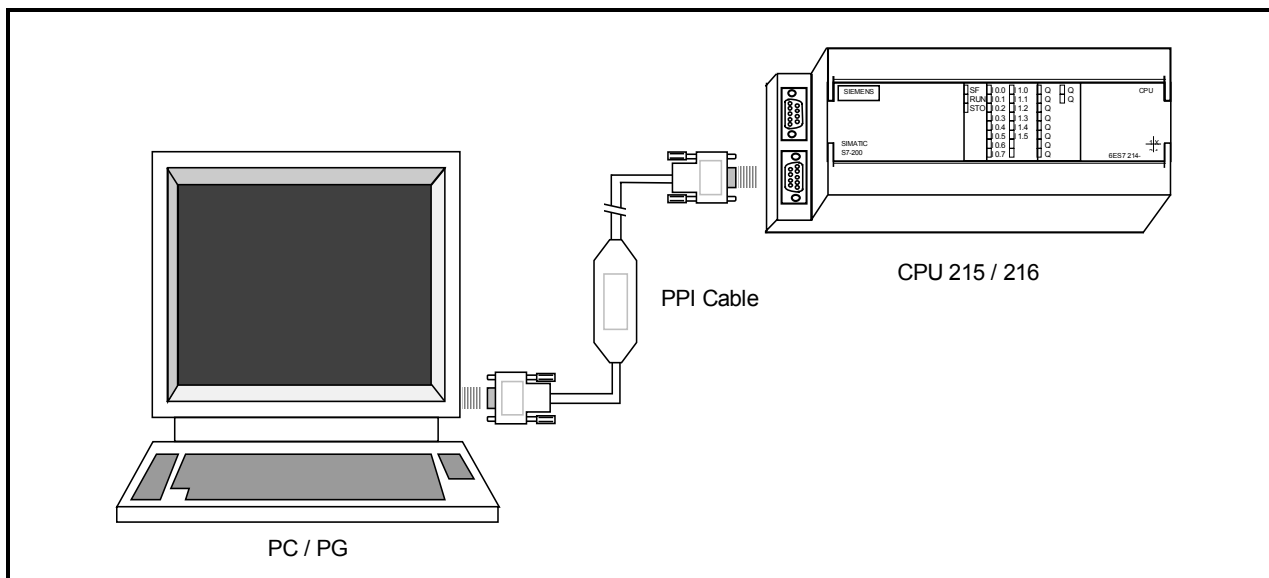


Рисунок 52.1 – Установка сети

Описание программы

Часть первая: Аппаратура

Интерфейс оператора

Эта программа использует программу терминала, которая отправлена с Windows 3.1. Программа терминала должна быть конфигурирована следующим образом:

- DEC VT100 (ANSI)
- Локальное эхо
- 9600, 7 бит, контроль четности, 1 стоповый бит
- Flow control -- none

Хост

- SIMATIC S7-200 CPU 215 или CPU 216
- PC/PPI Кабель соединенный с 0 портом S7216 и другим концом к одному из портов персонального компьютера RS-232 .
 - PC/PPI кабель должен быть конфигурирован на 9600 бод. Переключатель, устанавливающий 0100 (т.е. SW2 вверх, все другие вниз)

Часть вторая: RCV команды

Специальные биты памяти, не определенные в текущем руководстве.

SM4.5 - Порт 0: Передатчик неактивен ('0' – идет передача; '1' – нет передачи)

SM4.6 - Порт 1: Передатчик неактивен ('0' - идет передача; '1' - нет передачи)

Каждый раз при выполнении RCV команды, поле счета в обозначенном буфере получения очищается. Все другие данные в буфере будут записаны как только будут получены новые данные.

PC/PPI кабель содержит допускаемый данными RS-232 для RS-485 конвертера. Если Вы используете этот кабель для Freepoint связи, Вы должны позволить минимум 2 символьных промежутка между сообщениями противоположных направлений (получение напротив передачи). Вычисления в таблице основаны на 10 битовых символах (1 стартовый бит, 7 битов данных, 1 бит четности, 1 стоповый бит). Таблица ниже дает символьные времена для различных скоростей в бодах.

Скорость В бодах	время 1 символа Мсек	время 2 символов Мсек
38400	0.26	0.52
19200	0.52	1.04
9600	1.04	2.08
4800	2.08	4.17
2400	4.16	8.34
1200	8.34	16.67
600	16.67	33.34
300	33.34	66.67

Чтобы вычислить время сообщения в миллисекундах используют следующие формулы:

Символьное_время = $1000 \text{ ms} \cdot (\text{скорость} / \text{число_бит_на_символ})$

Время_сообщения = $\text{символьное_время} \cdot \text{число_символов}$

Если имеется изменение в конфигурации порта (т.е., четности, битов данных, или скорости в бодах), RCV команда сбросит порт и установит новую конфигурацию перед установкой получения символов.

Выполнение XMT и RCV команд взаимно исключается. Следующие комбинации команд дают ошибку если выполняются одновременно на одном и том же порте: XMT - XMT, RCV - RCV, XMT - RCV и RCV - XMT. Во всех случаях первая команда выполнится правильно а вторая команда будет терпеть неудачу с ошибкой (9).

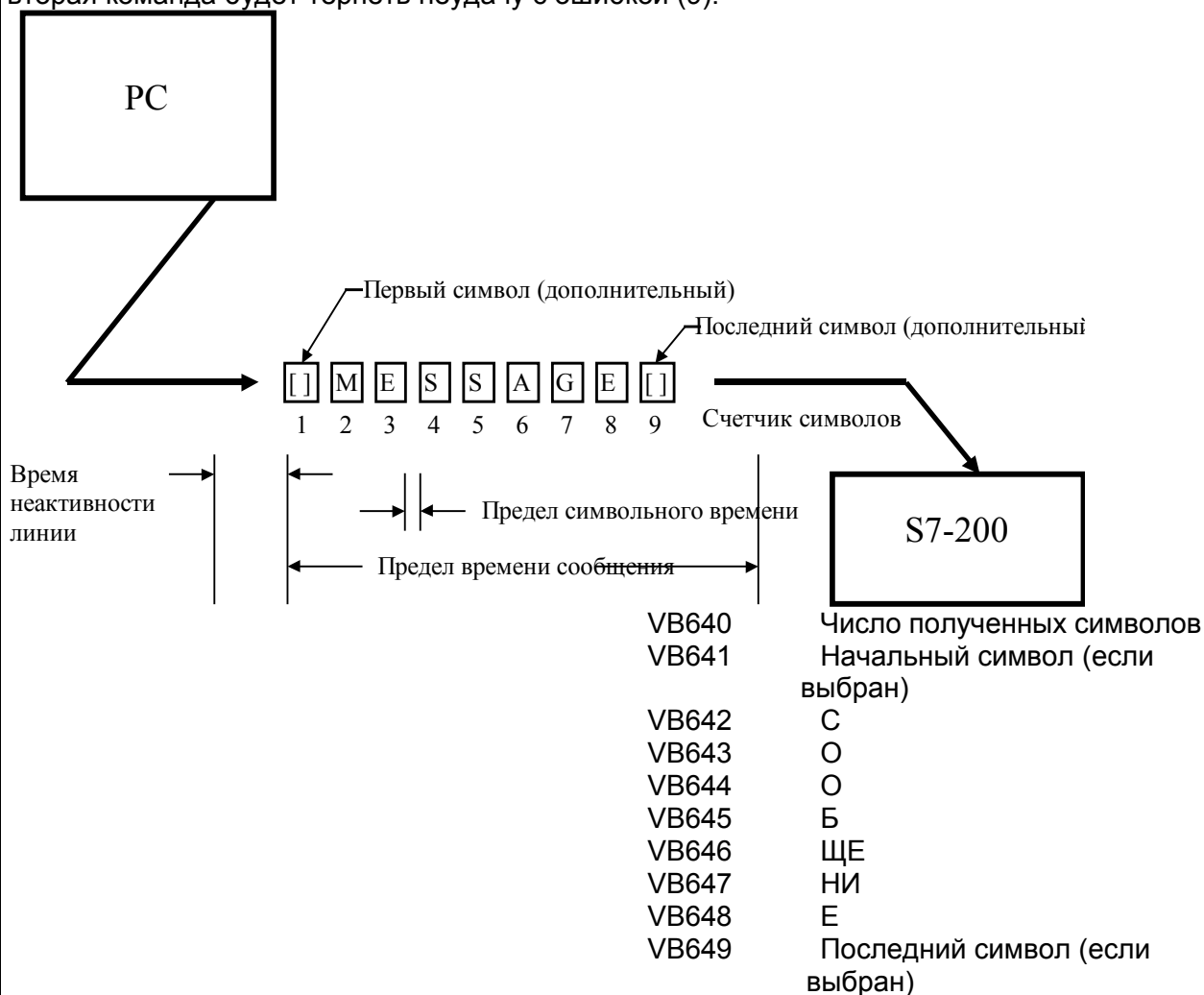


Рисунок 52.2 - Условия сообщения

Должно быть по крайней мере одно условие начала и одно условие останова, определенное для выполнения RCV команды. Как только условие начала сообщения было выполнено, завершение сообщения должно произойти прежде, чем может быть получено другое сообщение. Условие рестарта не поддерживается.

Начальное и конечное условие:

- Начало_сообщения: Перерыв_неактивности_линии_и_начальный_символ
- Конец_сообщения: Последний_символ ИЛИ Перерыв_сообщения/символа ИЛИ максимум_счетчика_символов ИЛИ ошибка_приоритета ИЛИ невозможная_команда

Чтобы прервать выполняющуюся RCV команду, очищается бит n (7) в байте управления для получения сообщения (SMB87 или SMB187, порт 0, порт, 1 соответственно) и затем выполняется RCV команда для рассматриваемого порта. Предыдущая RCV команда будет прервана, и будет установлен бит n (7) " Получение сообщения, завершено пользователем командой отключения " в байте получаемого сообщения (SMB86 или SMB186, порт 0, порт, 1 соответственно). Чтобы получить другое сообщение от рассматриваемого порта, будет необходимо повторно допустить функцию RCV для того порта, устанавливая бит n (7) в байте управления получаемого сообщения и выполняя RCV команду для определенного порта.

ПОРТ 0 ПОРТ 1	ИМЯ	ИСПОЛЬЗОВАНИЕ												
SM4.5 SM4.6	Порт 0 нет передачи Порт 1 нет передачи	Используется для определения, является ли передатчик занятым. Используйте Этот бит используется для предотвращения выполнения двух XMT команд одновременно на одном и том же порте. Если две XMT команды выполняются одновременно на одном и том же порте, происходит ошибка.												
SMB30 SMB130	Регистр управления режимом Freeport	Set up protocol and port communication parameters. XMT and RCV can only be used with the Freeport protocol.												
SMB86 SMB186	Байт состояния получаемого сообщения	Use to determine which termination condition terminated the RCV instruction. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">MSB</td> <td style="text-align: center;">LSB</td> </tr> <tr> <td style="text-align: center;">7</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">n</td> <td style="text-align: center;">o</td> </tr> <tr> <td style="text-align: center;">e</td> <td style="text-align: center;">t</td> </tr> <tr> <td style="text-align: center;">o</td> <td style="text-align: center;">c</td> </tr> <tr> <td style="text-align: center;">o</td> <td style="text-align: center;">p</td> </tr> </table>	MSB	LSB	7	0	n	o	e	t	o	c	o	p
MSB	LSB													
7	0													
n	o													
e	t													
o	c													
o	p													
SMB87 SMB187	Байт состояния получаемого сообщения	Used to configure the start and stop conditions for receiving a message. Also used to abort a message. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">MSB</td> <td style="text-align: center;">LSB</td> </tr> <tr> <td style="text-align: center;">7</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">n</td> <td style="text-align: center;">x</td> </tr> <tr> <td style="text-align: center;">y</td> <td style="text-align: center;">z</td> </tr> <tr> <td style="text-align: center;">m</td> <td style="text-align: center;">t</td> </tr> <tr> <td style="text-align: center;">o</td> <td style="text-align: center;">o</td> </tr> </table>	MSB	LSB	7	0	n	x	y	z	m	t	o	o
MSB	LSB													
7	0													
n	x													
y	z													
m	t													
o	o													
SM87.7 SM187.7	Допуск / отключение функции получения	The RCV instruction can only operate if it is enable. If the RCV instruction is in operation and this bit is cleared (Disabled) the execution of the next RCV instruction will terminate (abort) the Receive function. This termination will occur even if the message is in the process of being received.												
SM87.6 SM187.6	Обнаружение первого символа сообщения (Условие старта)	Use the Value in SMB88/SM188 to detect start of message. This could be a character used to detect start of message or possibly the address of your node in a network (i.e., if this CPU is node 5, load a 5 into SMB88). Any message that does not start with a 5 will not be received.												
SM87.5 SM187.5	Обнаружение последнего символа сообщения (Завершение сообщения)	Use the Value in SMB89/SMB189 to detect end of message. In an application where an individual was entering data the <carriage return character> might make a good end of message character. Make sure to use another form of termination such as inter-character timeout or end of message timeout just in case the end of message character isn't received.												
SM87.4 SM187.4	Обнаружение условия неактивной линии (Условие старта)	Use the Value in SMW90 to detect an idle line condition. The first character received after idle line time has expired is the start of message. To further refine the start of message one can also specify a start of message character.												
SM87.3 SM187.3	Печать таймера	Indicates timer type: Timer can either be a Inter-character timer or an End-of-Message timer. Typically shorter timeout values can be used if the Inter-Character-Timer is used.												
SM87.2 SM187.2	Запуск таймера (Завершение сообщения)	Terminate receive if the time period in SMW92 or SMW192 is exceeded.												
SMB94 SMB194	Максимальный размер буфера (Завершение сообщения)	Maximum number of buffers to be received. At least a count of 1 byte must be used. If a zero appears in this field the RCV instruction performs as a NOP.												
Для дополнительной информации см. Приложение D, Биты Специальной Памяти (SM), в														

Руководстве Программируемого Контроллера Системы Simatic S7-200

Часть третья: Программа

Эта программа связывается с интерфейсом оператора, используя XMT и RCV команды. Машина состояний была выполнена для осуществления связи. Машина состояний состоит из 5 состояний связи (Сост. Связи):

- НЕАКТИВНАЯ (0)
- RCV (1)
- XMT (2)
- ОЖИДАНИЕ (3)
- ОШИБКА (4)

Обратитесь к рисунку 52-3 для графического представления как работает машина состояний.

При первом проходе инициализируются параметры порта, событие 'XMT совершена' соотносится с прерыванием 0, событие 'RCV совершена' соотносится с прерыванием 1, прерывания допускаются, машина инициализирована к ПРОСТОЮ, и меню пользователя передается к интерфейсу оператора.

Когда передача завершена происходит прерывание выполняемой RCV команды, если машина состояний не находится в состоянии ошибки. Если машина состояний находится в ошибке, происходит возврат из прерывания. Так как XMT и RCV команды не могут быть выполнены одновременно, RCV команда должна быть защищена от выполнения, пока меню команды не будет передано. Только сообщения об ошибках передается в состоянии ошибки. Однако, сообщение об ошибках будет всегда сопровождаться командным меню. Если RCV команда должна была быть выполнена, в то время как машина состояний находится в ошибке, меню команды не может быть передано, так как XMT и RCV команды не могут выполняться одновременно. RCV команда должна победить, так как она будет выполнена сначала и меню команды никогда не будет передано.

Нормальная работа машины состояний простая и прямая:

- командное меню передается наружу,
- при полной передаче происходит прерывание,
- допускается функция получения,
- Переход машины состояний к НЕАКТИВНОМУ состоянию, ждет команду
- Команда получена,
- Получение завершено происходит прерывание
- Сообщение проверяется на ошибки получения
- Переход машины состояний в состояние получения
- Сообщение проверяется на команду поддержки
- Команда выполняется
- Переход машины состояний в состояние передачи
- передается меню команды

Обратите внимание: Состояние Связи 3 "ожидание 2 символов" необходим, только если используется PPI кабель. Ожидание 2 символов необходимо для того, чтобы направить связь вокруг. Если используется связь напрямую через RS485 кабель, состояние 3, может быть опущено.

Для дополнительной справки о XMT и RCV командах, см. Руководство Программируемого Контроллера Системы S7-200. Размер программы - 430 байтов.

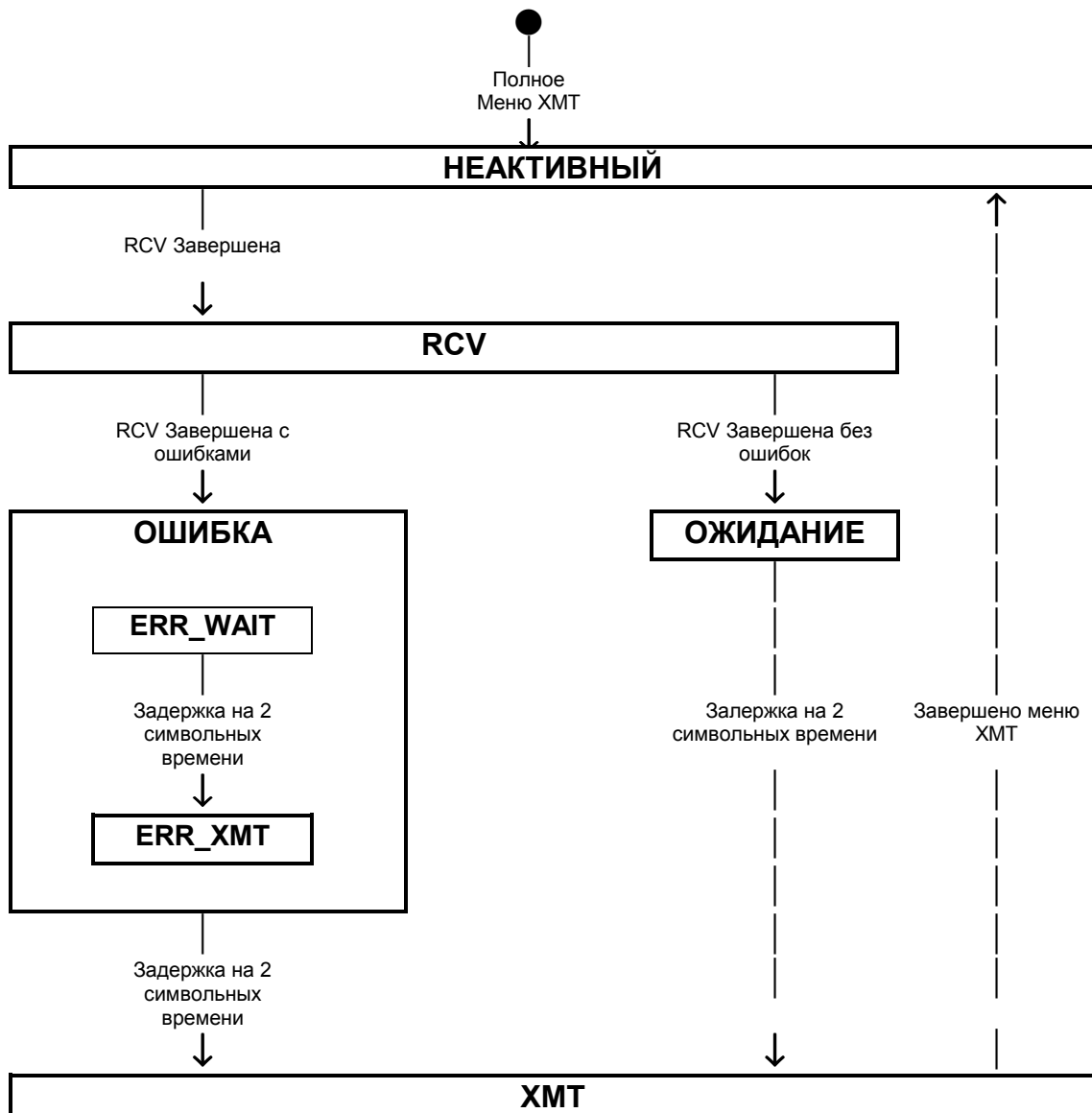
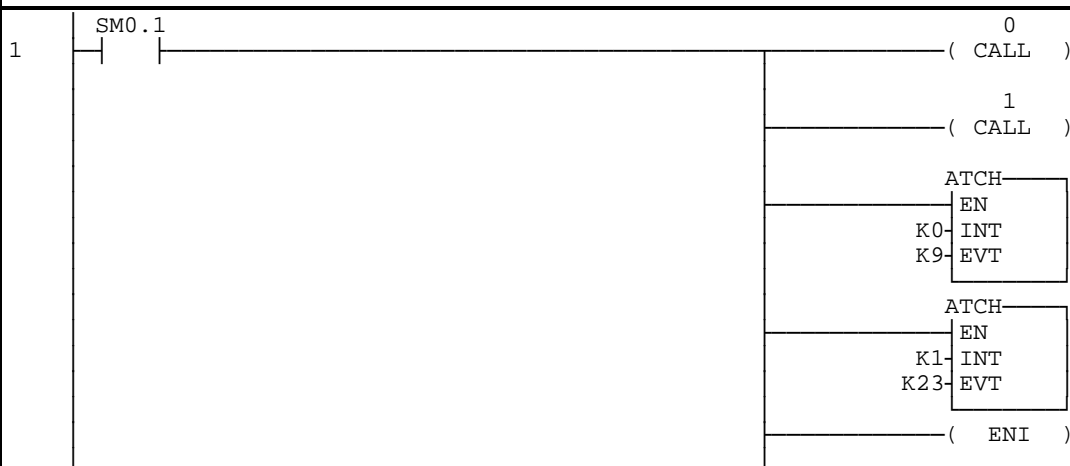
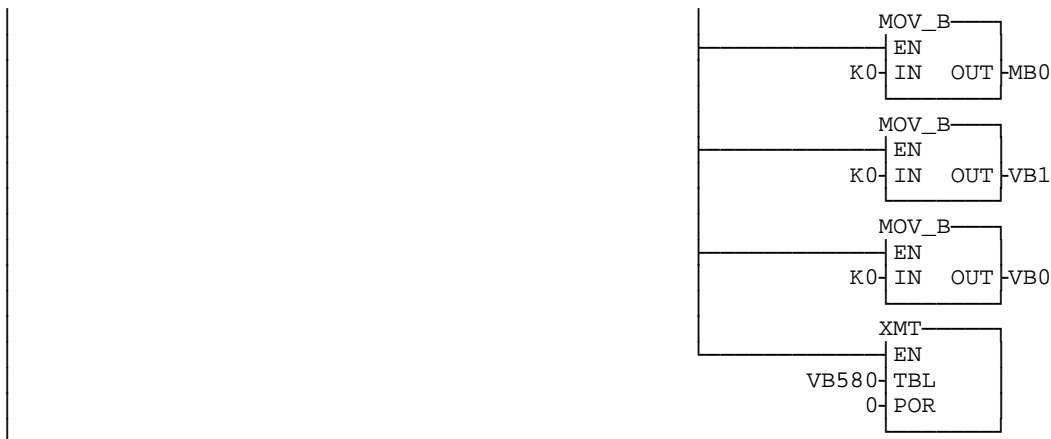


Рисунок 52.3 – машина состояний

Программа

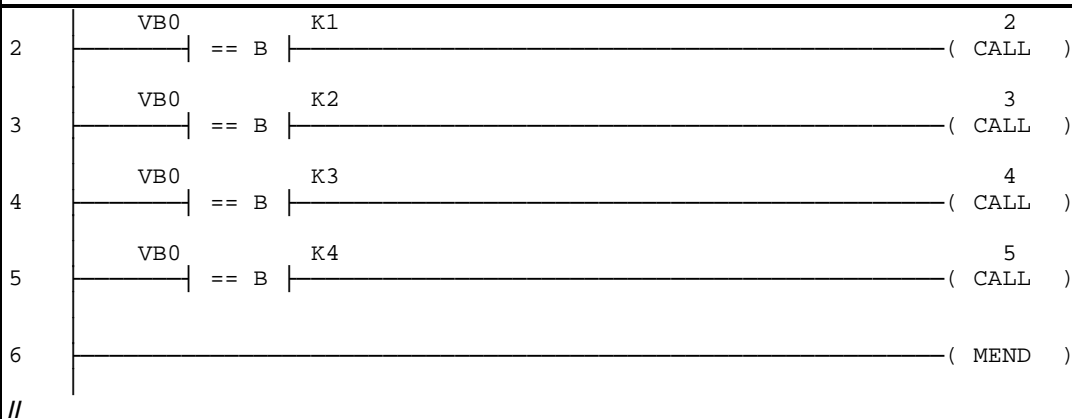
```
//
// ЗАГОЛОВОК: Командный процессор
//
// ЦЕЛЬ:      Эта программа связывается с интерфейсом оператора
//            используя передчу и получение команд.
//
// Использование Памяти:
// MB0       : Байт ошибки
// M0.7      : Большой бит ошибки
// M0.0      : Недопустимая команда
// SM0.1     : Первый бит просмотра
// SM4.5     : Бит- передатчик занят
// SMB30     : Порт 0 Freeport байт управления режимом
// SMB86     : Порт 0 Байт состояния получаемого сообщения
// SMB87     : Порт 0 Байт управления получаемого сообщения
// SMB89     : Порт 0 " конечный символ сообщения " ( <CR> )
//           : Условие останова
// SMW90     : Порт 0 Время простоя линии в миллисекундах.
//           : Условие начала.
// SMW92     : Порт 0 Значение меж-символьного перерыва (1 секунда)
//           : Условие ошибки
// VB0       : Сост.связи, 0=простой,1=получение,2=xmt,3=ожидание, 4=ошибка
// VB1       : Подсостояние ошибки, 1=ожидание, 2=xmt.
// VB100     : Сообщение об ошибках - завершение получения сообщения
//           : по команде отключения пользователем
// VB180     : Сообщение об ошибках - получен последний символ
// VB260     : Сообщение об ошибках - Получаемое сообщение завершено:
//           : таймер истек
// VB340     : Сообщение об ошибках - Получаемое сообщение завершено:
//           : счетчик достиг максимального количества символов
// VB420     : Сообщение об ошибках - Получаемое сообщение завершено:
//           : из-за ошибки четности
// VB500     : Сообщение об ошибках -недопустимая команда
// VB580     : Буфер команд - ввод команды (ON|OFF + <CR>):
// VB640     : Получение буфера
// T32       : Задержка на 2 символьных времени (5 msec)
//
```





```

LD SM0.1 // если(первый проход) {
CALL 0 // Инициализация параметров связи порта 0
CALL 1 // Инициализация параметров связи порта 0
ATCH 0,9 // Прерывание 0 --> xmt завершена
ATCH 1,23 // Прерывание 1 --> rcv завершена
ENI // допуск прерываний
MOVB 0,MB0 // Нет ошибок
MOVB 0,VB1 // инициализация подсостояния ошибок в NULL
MOVB 0,VB0 // инициализация состояния связи в IDLE
XMT VB580,0 // передача меню
// }
  
```



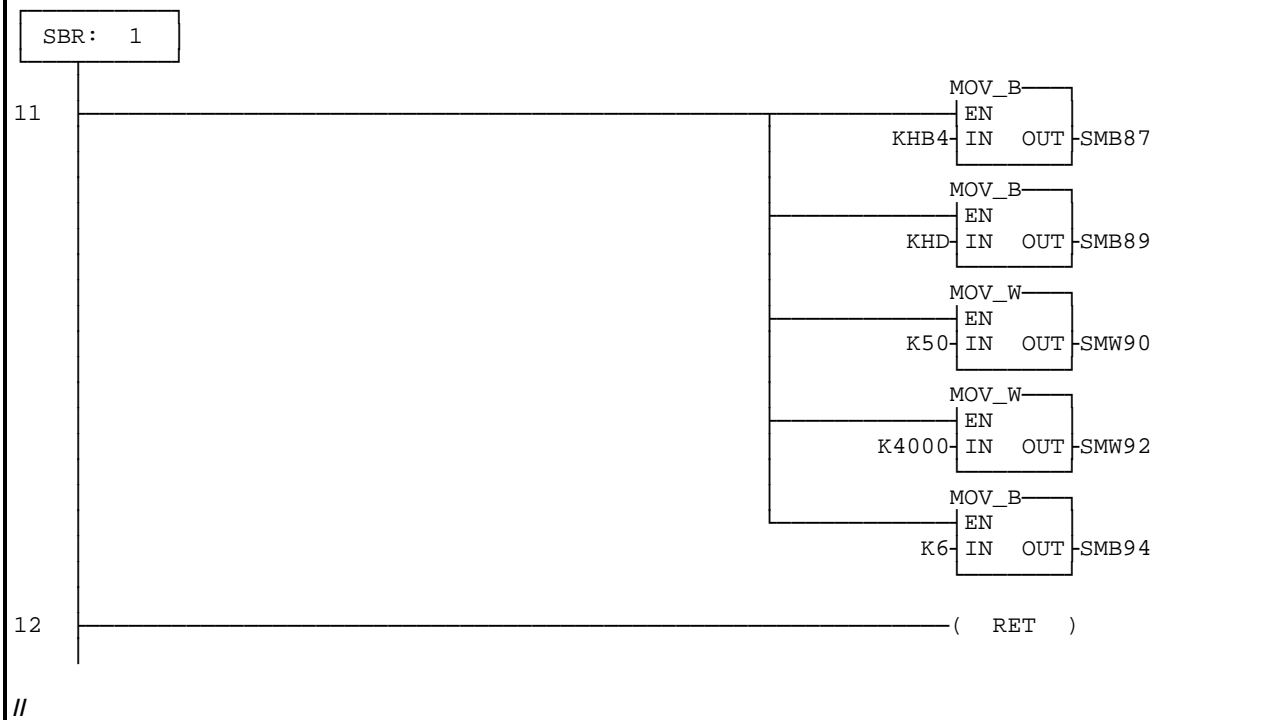
```

// Процессор машины состояний. Имеются только 5 состояний.
// Простой, Передача, Получение, Ожидание, и Ошибка.
//
LDB= VB0,1 // если ( сост.связи = получение )
CALL 2 // процесс сообщения
LDB= VB0,2 // если ( сост.связи = передача )
CALL 3 // процесс сообщения
LDB= VB0,3 // если ( сост.связи = ожидание )
CALL 4 // процесс ожидания 2 символов
LDB= VB0,4 // если ( сост.связи = ошибка )
CALL 5 // процесс ошибки
MEND
  
```



// Подпрограмма 0 – Инициализация порта 0
 //
 // Функция: Инициализация порта 0 для работы freeport на
 // 9600, 7 бит, контроль четности.

```
SBR 0
MOV_B 16#E9,SMB30      // Freeport, 9600, 7 бит, контроль четности
RET
```



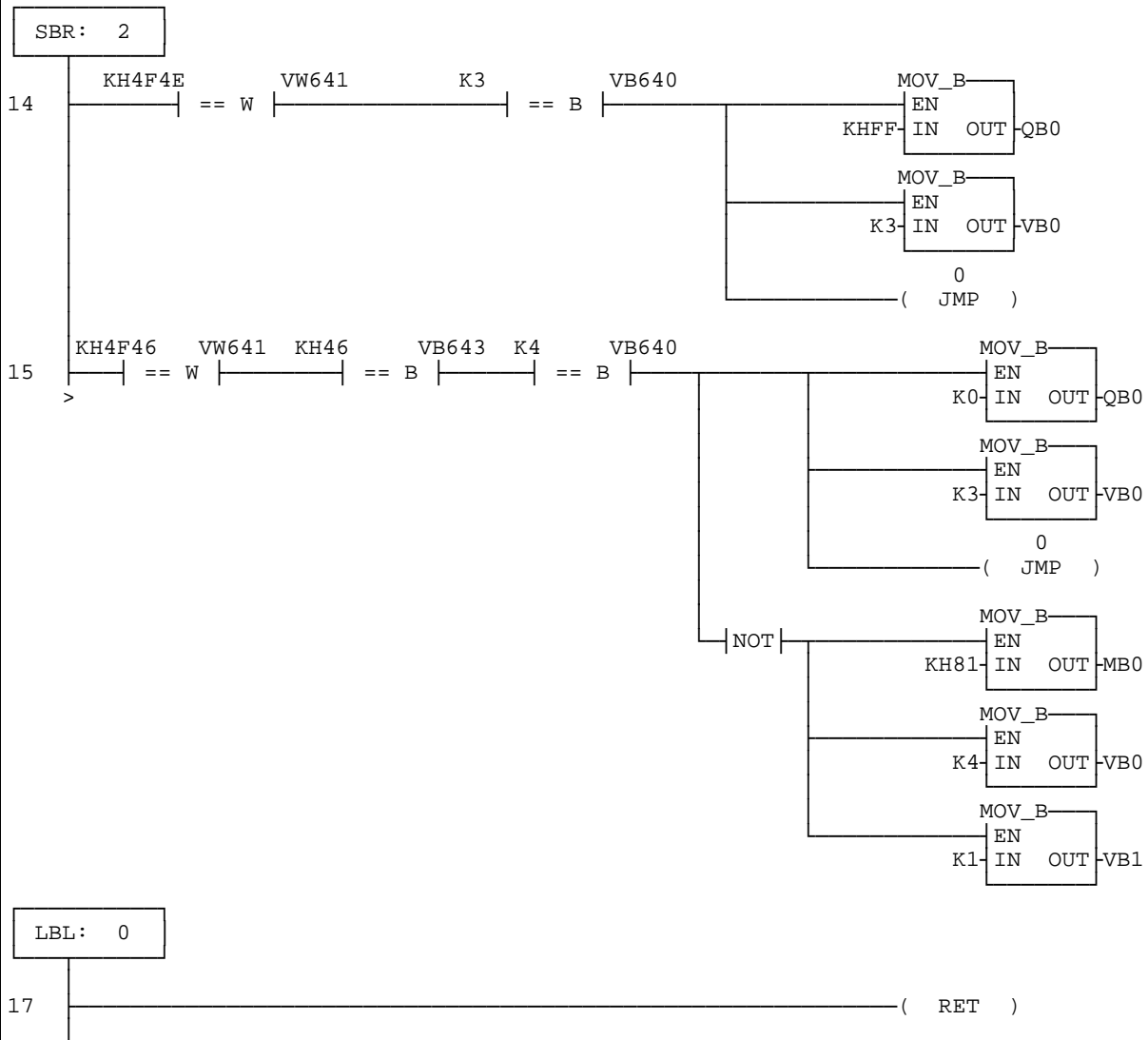
// Подпрограмма 1 -- Инициализация параметров получения(Порт 0)
 //
 // MSB LSB
 // SMB87 7 0
 // n x y z m t 0 0
 // n:(1) Допуск приемника
 // x:(0) Игнорирование SMB88
 // y:(1) Определение последнего символа сообщения (SMB89)
 // z:(1) Определение простоя линии (SMW90)
 // m:(0) Меж-символьный таймер (SMW92)
 // t:(1) Таймер допуска (m)
 //

```
SBR 1
MOV_B 16#B4,SMB87      // Инициализация управляющего бита получаемого сообщения
```

```

MOVW 16#0D,SMB89 // Символ конца сообщения <CR>
MOVW 50,SMW90 // Перерыв простоя линии = 50 мсек
MOVW 4000,SMW92 // Меж-символьный перерыв = 4000 Мсек
MOVW 6,SMB94 // максимальное число символов = 6
RET

```



```

// Подпрограмма 2 – Процесс получения сообщения – СОСТ.СВЯЗИ 1

```

```

//
// Функция: This state processes valid messages. Определяет поддерживается
// команда или нет. Поддерживаются только 2 команды.
// "ON" - включить 8 выходов
// "OFF" - выключить 8 выходов.
//

```

```

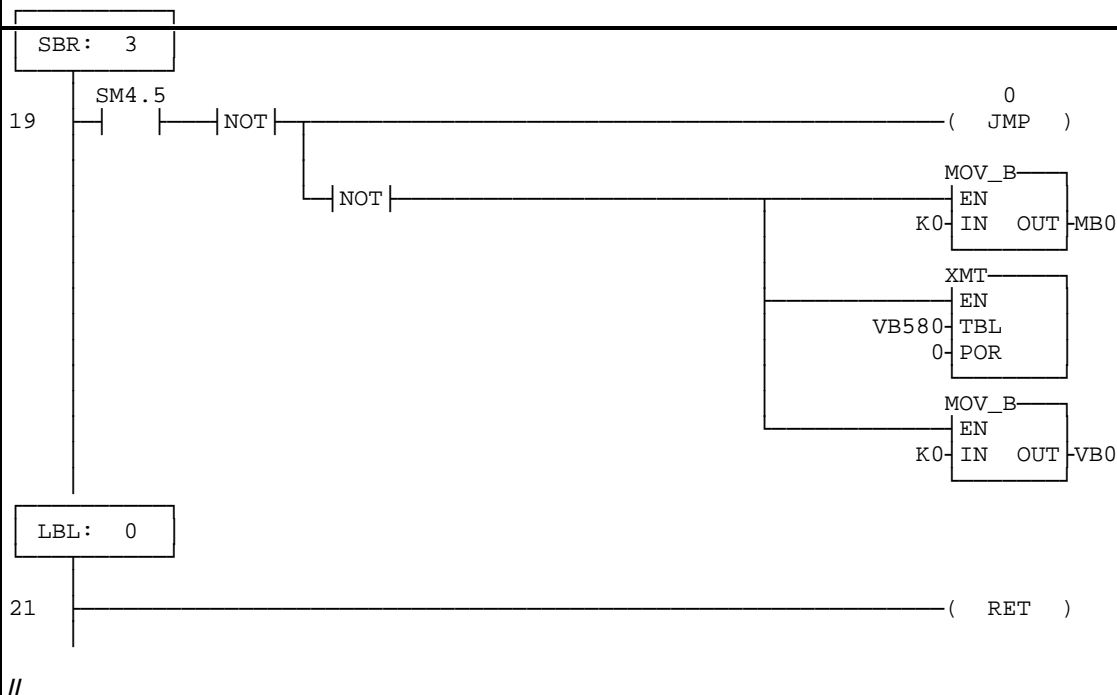
SBR 2 // В случае (КОМАНДА) {
LDW= 16#4F4E,VW641 // ON : (команда==ON) AND
AB= 3,VB640 // (счетчик символов = 3)
MOVW 16#FF,QB0 // включены 8 выходов
MOVW 3,VB0 // следующее состояние-ожидание

```

```

JMP 0 //
LDW= 16#4F46,VW641 // OFF :(команда==OFF)
AB= 16#46,VB643 // AND
AB= 4,VB640 // (счетчик символов = 4)
MOVB 0,QB0 // выключены 8 выходов
MOVB 3,VB0 // следующее состояние-ожидание
JMP 0 //
NOT // ERROR :
MOVB 16#81,MB0 // ошибка = (ошибка + недопустимая команда)
MOVB 4,VB0 // следующее состояние ошибки
MOVB 1,VB1 // следующее подсостояние ошибки-ожидание
LBL 0 // } конец
RET // возврат

```



```

//
// Подпрограмма 3 –Передача сообщения – СОСТ.СВЯЗИ 2

```

```

//
// Функция: Если передатчик не занят передает
// command menu.
//

```

```

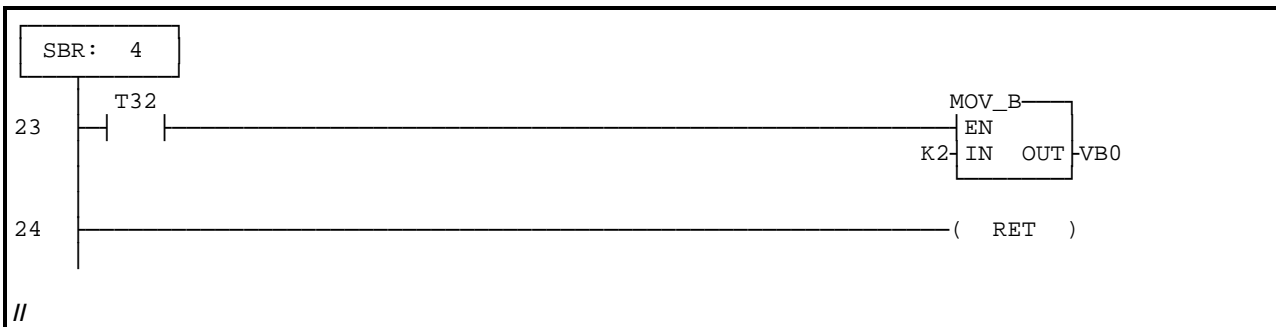
SBR 3
LD SM4.5 // если (передатчик НЕ передает)
NOT // {
JMP 0 //

```

```

NOT //
MOVB 0,MB0 // бит ошибки = Нет ошибки
XMT VB580,0 // передача меню
MOVB 0,VB0 // следующее состояние-простой
LBL 0 // }
RET // возврат

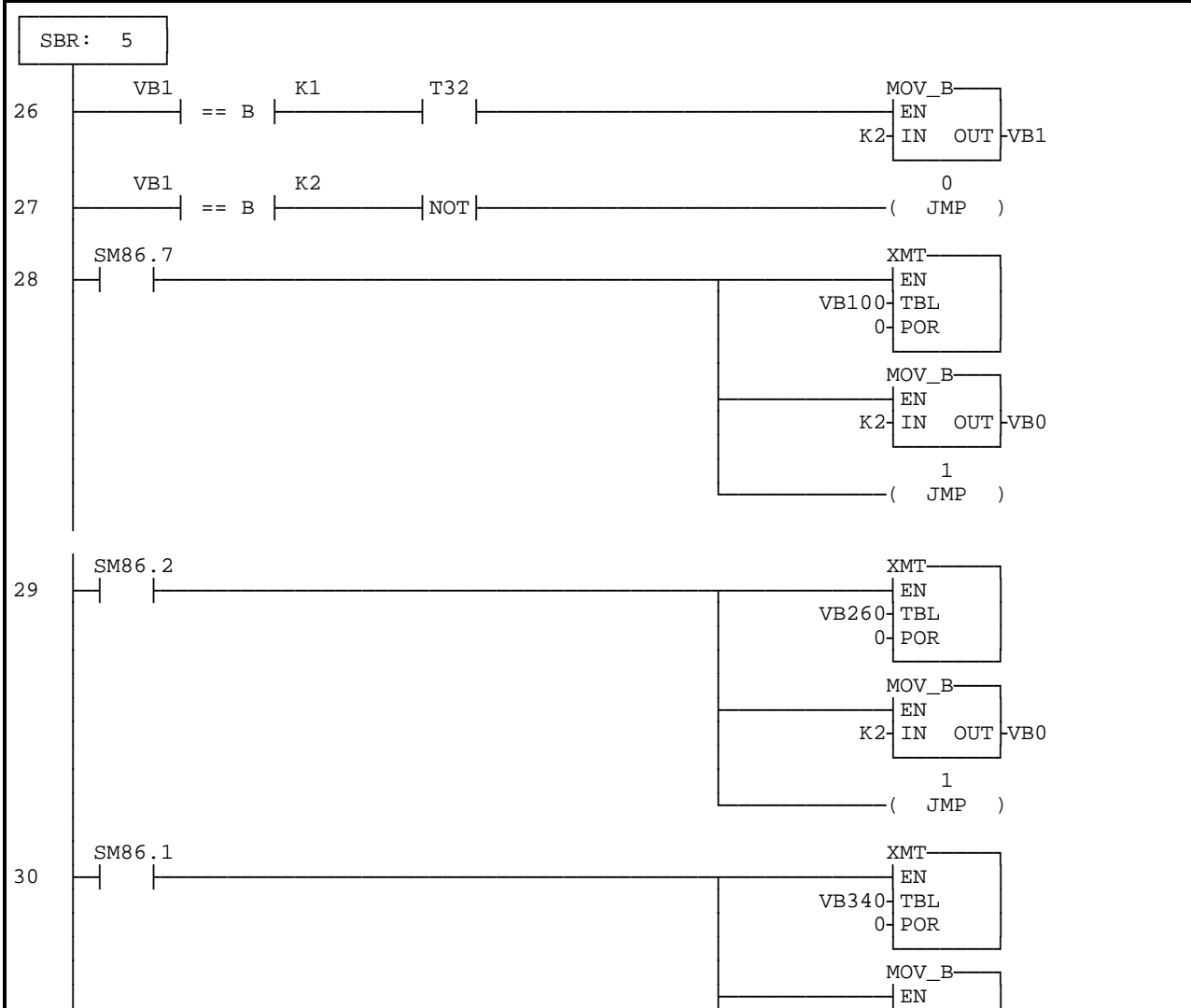
```

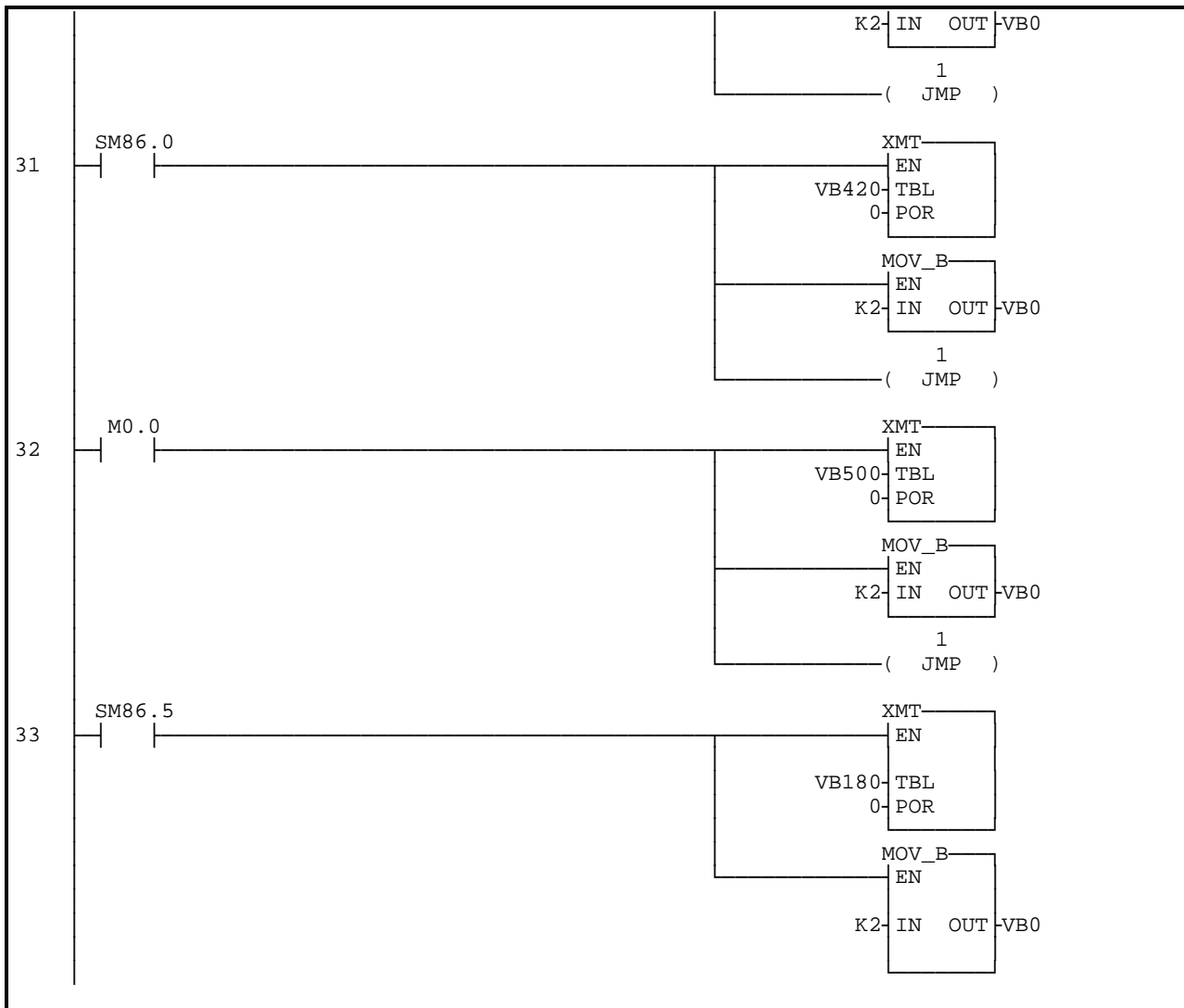


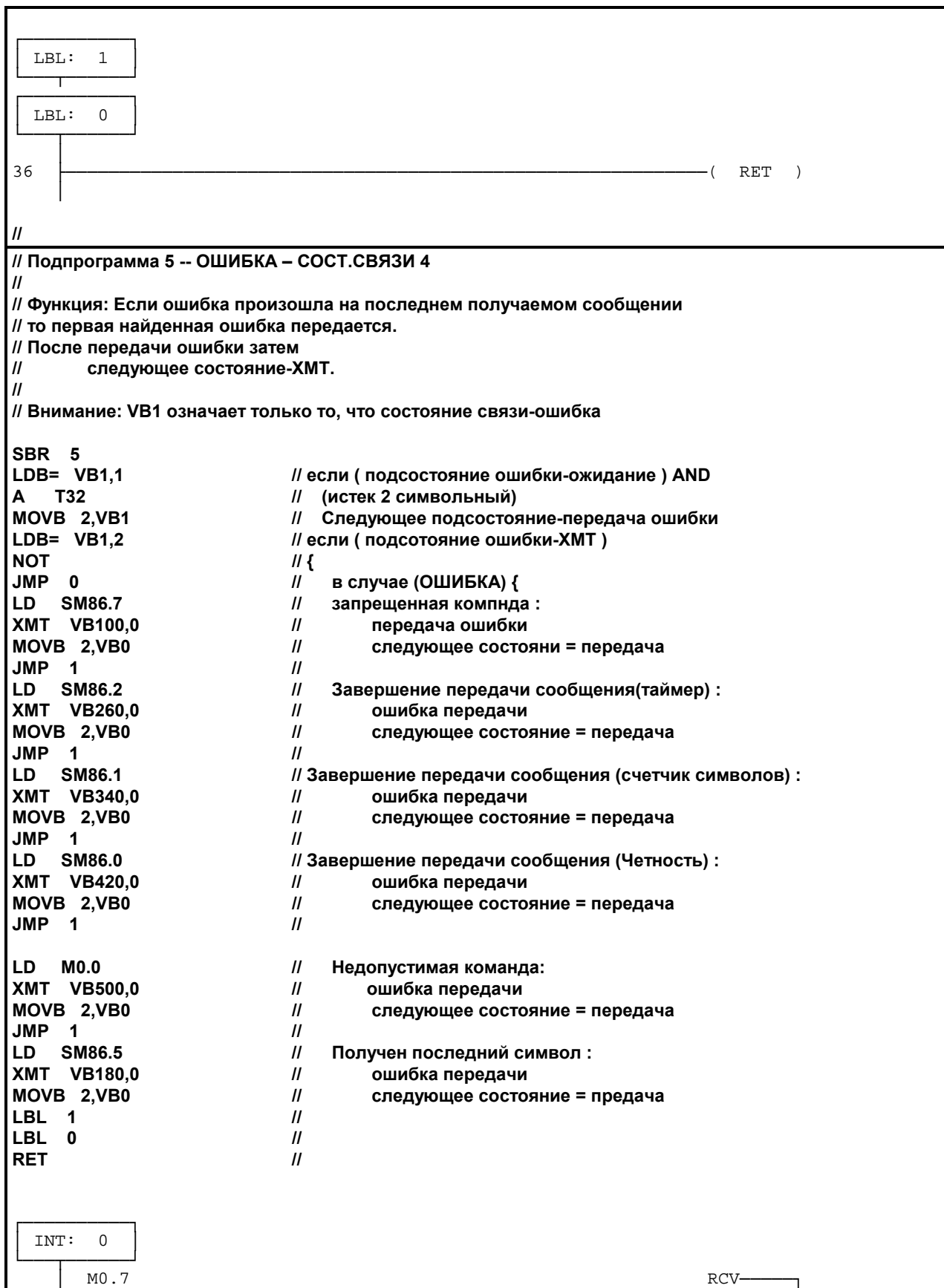
//

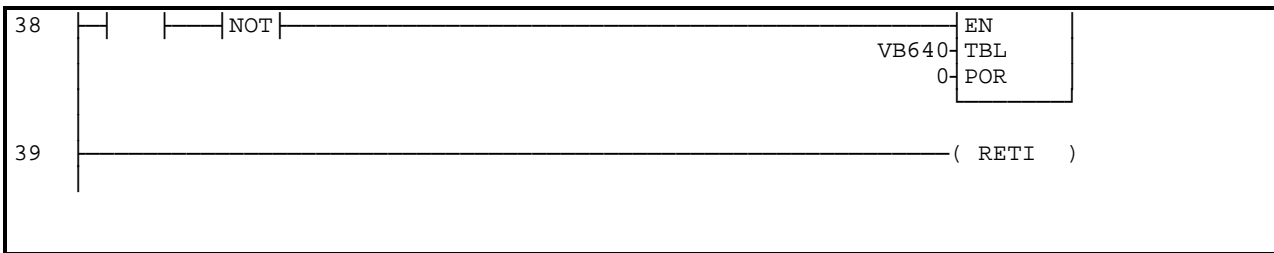
```

// Подпрограмма 4 --ОЖИДАНИЕ – СОСТ.СВЯЗИ 3
//
// Функция: Эта подпрограмма продвигают состояние связи в
// передачу если истекла задержка 2 символов
//
SBR 4
LD T32 // если (перерыв)
MOVB 2,VB0 // Следующее состояние связи-передача
RET // возврат
  
```





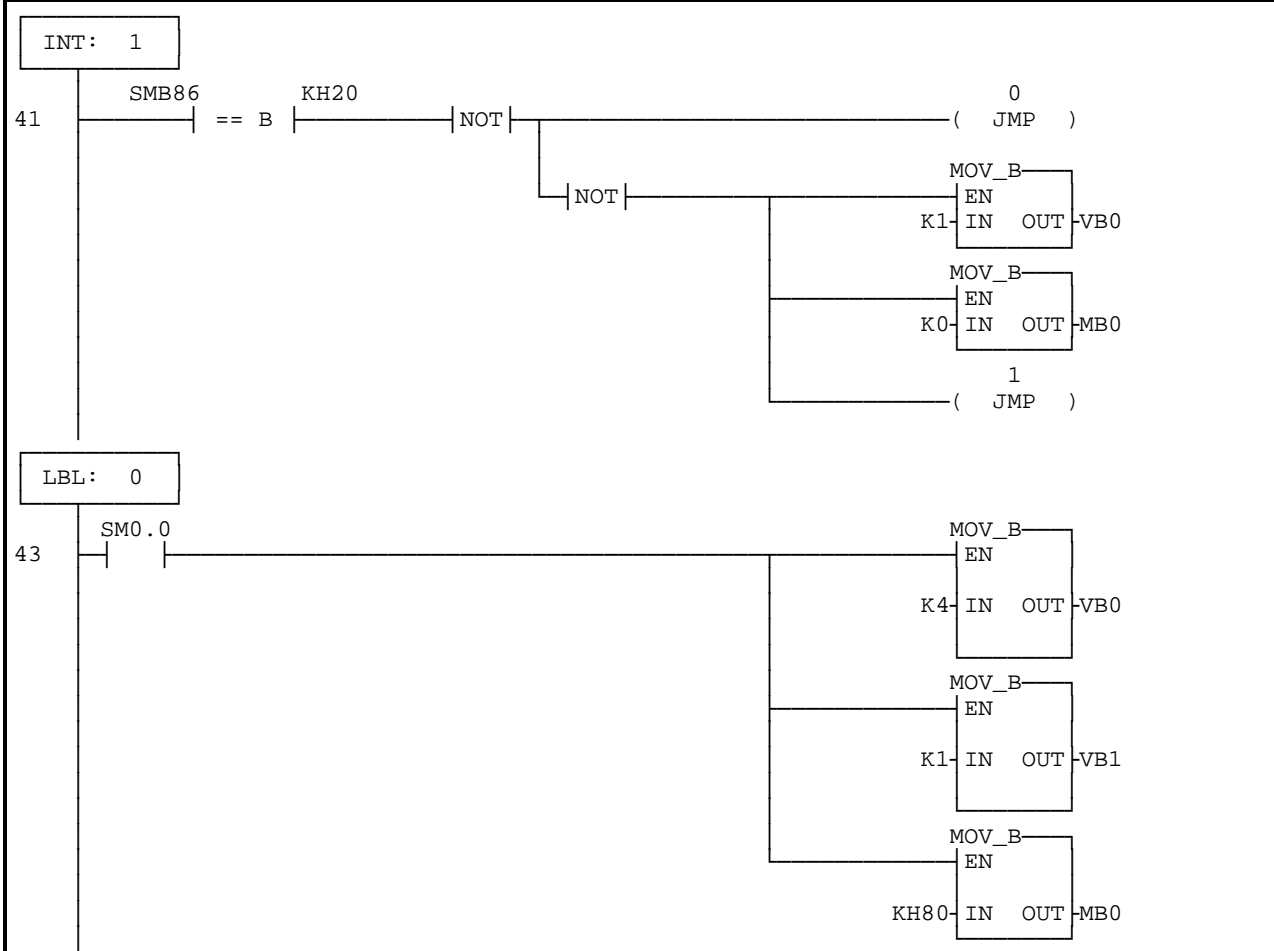


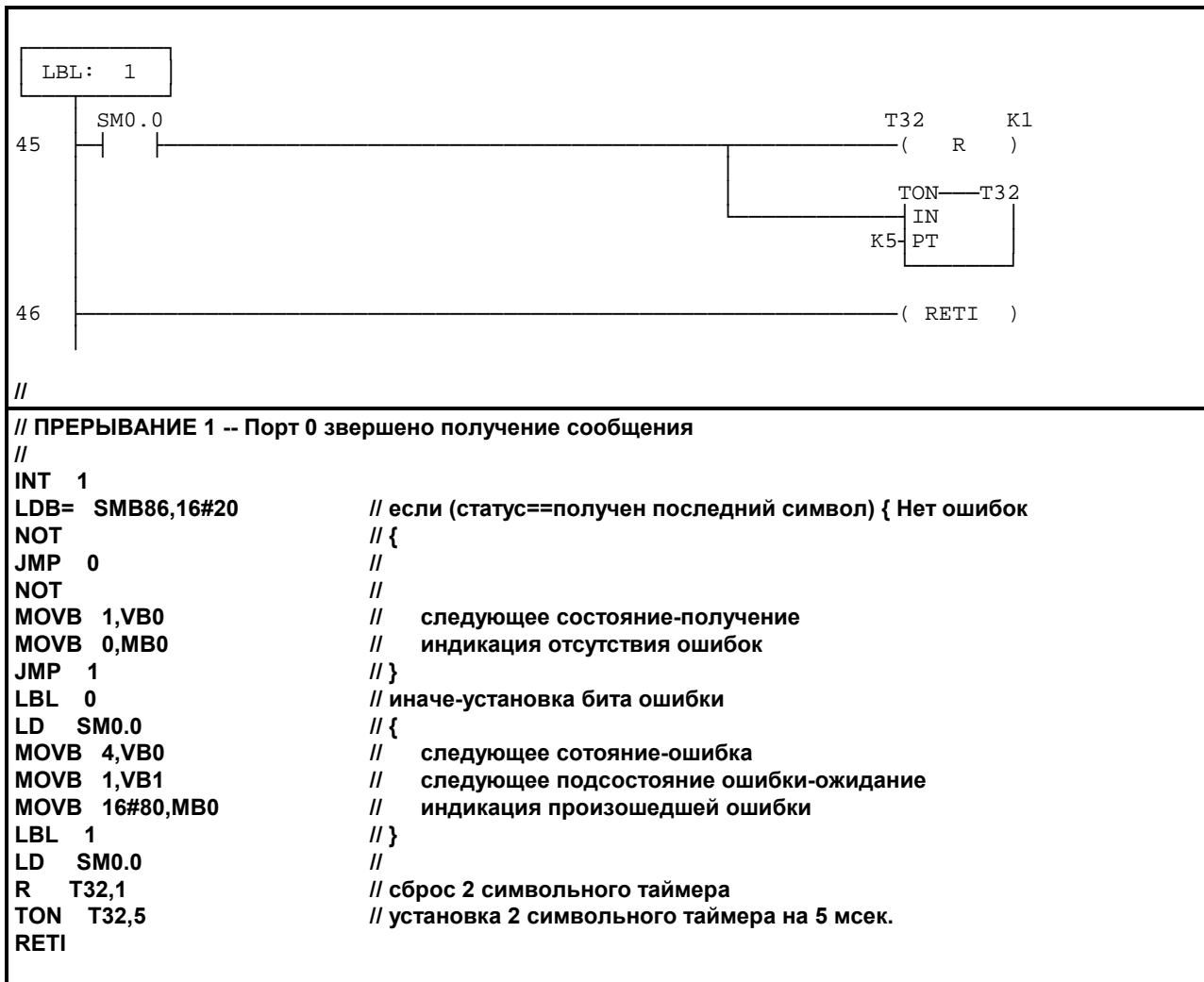


```

//
// ПЕРЕРЫВАНИЕ 0 -- Порт 0 передача завершена

INT 0
LD M0.7           // если (Нет ошибок)
NOT              // {
RCV VB640,0      // установка получения следующей команды
RETI            // }
    
```





Блок данных: DB1

```
//  
// XMT Формат сообщения для данного применения:  
//           [длина<LFCR>текст_сообщения]  
//  
VB100 51  
VW101 16#0A0D  
VB103 'Receive message terminated by user disable command'  
//  
VB180 24  
VW181 16#0A0D  
VB183 'End character received'  
//  
VB260 42  
VW261 16#0A0D  
VB263 'Receive message terminated: Timer Expired'  
//  
VB340 62  
VW341 16#0A0D  
VB343 'Receive message terminated: Maximum character count achieved'  
//  
VB420 52  
VW421 16#0A0D  
VB423 'Receive message terminated because of parity error'  
//  
VB500 17  
VW501 16#0A0D  
VB503 'Invalid Command'  
//  
VB580 32  
VW581 16#0A0D  
VB583 'Enter Command (ON|OFF + <CR>):'  
//
```

При преобразовании из IEC STL в S7-Micro/DOS STL:

- Добавлять 'K' перед всеми шестнадцатиричными числовыми константами (т.е. 4 ⇒ K4)
- Заменять '16#' на 'KH' для всех шестнадцатиричных констант (т.е. 16#FF ⇒ KHFF)
- Запятые обозначают разделение полей. Используйте стрелки или клавишу TAB, чтобы переключаться между полями.
- Чтобы преобразовывать S7-Micro/DOS программу STL в форму LAD, каждая сеть должна начинаться со слова 'СЕТЬ' и номера. Каждая сеть в Application Tip программы обозначена номером на ступенчатой диаграмме. Используйте команду INSNW под меню EDIT, чтобы ввести новую сеть Команды.MEND, RET, RETI, LBL, SBR, и INT каждая получает свою собственную сеть.
- Линия - комментарий, обозначенная '//' невозможна в S7-Micro/DOS, но Сеть - комментарий возможна.

Показанные блоки данных были созданы в Micro / Win; в Micro / DOS использование редактора V-памяти для ввода блока данных (Обратитесь к руководству пользователя Micro / DOS). Обратите внимание что в Micro / Win:

- 16#.. обозначает шестнадцатиричные значения
- 'text' представляет строковые значения

Общие замечания

SIMATIC S7-200 Советы по применению обеспечиваются для того, чтобы дать пользователям S7-200 некоторое представление относительно того, как, некоторые задачи из представления методики программирования, могут быть решены с этим контроллером. Эти инструкции не предполагают отражать все детали или разновидности оборудования, и при этом они не предусматривают любое возможное непредвиденное обстоятельство. Использование S7-200 Советов по применению свободное.

Siemens оставляет право делать изменения в спецификациях, показанных здесь или делать усовершенствования в любое время без примечания или обязательства. Это не освобождает пользователя от ответственности, при использовании звуковых методов в прикладной программе, установке, операции, и сопровождении приобретенного оборудования. Если возникает конфликт между общей информацией, содержащейся в этой публикации, содержанием рисунков и дополнительным материалом, или обоими, последние должны иметь приоритет.

Siemens не несет ответственности, по любой допустимой причине, в повреждениях или персональном ущербе, последующем из использования советов по применению.

Все права защищены. Любая форма дублирования или распространения, включая выдержки, разрешается только с конкретным разрешением SIEMENS.