



**сообщения
объединенного
института
ядерных
исследований
дубна**

430/83

17/1-83

11-82-738

Е.Ю.Мазепа

**ПРОТОКОЛ ОБМЕНА ИНФОРМАЦИЕЙ
С ГРУППОВЫМИ ТЕРМИНАЛАМИ ЭВМ CDC-6500
И ЕГО РЕАЛИЗАЦИЯ
В СИСТЕМЕ КОЛЛЕКТИВНОГО ПОЛЬЗОВАНИЯ
НА БАЗЕ ЭВМ БЭСМ-6, CDC-6500
И КОНЦЕНТРАТОРА ТЕРМИНАЛОВ**

1982

С вводом в эксплуатацию концентратора терминалов (ЕС-1010) для ЭВМ БЭСМ-6/1/ пользователи ЦВК ОИЯИ получили возможность проводить с 16 удаленных терминалов следующие работы:

- независимое от ЭВМ БЭСМ-6 редактирование и накопление текстовой информации;
- запуск на счет задач для ЭВМ БЭСМ-6 и получение результатов счета на терминалы либо на внешние устройства ЭВМ БЭСМ-6 (как в пакетном, так и в интерактивном режимах).

В ноябре 1981 года была выполнена работа по подсоединению концентратора терминалов к ЭВМ CDC-6500 в качестве удаленной групповой станции (терминала), имеющей в своем составе индивидуальные терминалы. Это позволило пользователям концентратора терминалов (по их желанию) находиться либо в сеансе с собственной программной системой концентратора терминалов, либо в сеансе с подсистемой ИНТЕРКОМ на ЭВМ CDC-6500. Работа пользователей ЭВМ CDC-6500 возможна с любого из 16 терминалов концентратора, если только общее число терминалов концентратора, одновременно работающих с CDC-6500, не превышает 12.

Протокол обмена моды 4А между групповым терминалом и ЭВМ CDC-6500

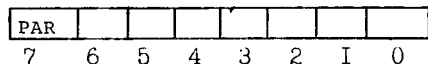
Опишем здесь только ту часть протокола обмена, которая позволяет обслуживать экранные пульта группового терминала. Желая ознакомиться с другими возможностями протокола обмена моды 4А (запуск задач с читающего устройства перфокарт группового терминала на ЭВМ CDC-6500, распечатка листингов на АЦПУ группового терминала) можно рекомендовать литературу^{2,3/}.

Центральная ЭВМ^{*}) и групповой терминал обмениваются между собой сообщениями, представляющими собой последовательность байтов.

^{*}) Т.к. протокол моды 4А был разработан для ЭВМ CDC серии 6000 и может, в принципе, быть реализован на любой ЭВМ, в дальнейшем вместо CDC-6500 будем употреблять термин "центральная ЭВМ".

Формат байта

Каждый байт имеет вид:



где в битах с 0 по 6 содержится один из 128 возможных кодов, допускаемых при передаче сообщения. Седьмой бит - это бит для контроля правильности кода. Значение этого бита определяется следующей булевой функцией:

$$b_7 = f(b_0, b_1, \dots, b_6) = b_0 \oplus b_1 \oplus \dots \oplus b_6,$$

где b_i - значение i -го разряда байта, \oplus - сложение по модулю 2. Т.е. 7-й разряд равен 1, если количество единиц в остальных разрядах четно, и равен 0 в противном случае.

Такой контроль называется поперечным контролем с дополнением до нечетности.

Формат сообщения

Каждое принимаемое/ посылаемое сообщение имеет следующий формат:

SYNC	SYNC	SON	SITE	STATION	CC	DATA	ESCAPE	E-код	ETX	MP
------	------	-----	------	---------	----	------	--------	-------	-----	----

где

SYNC=16*^{*)} - синхросимвол: специальный символ, синхронизирующий аппаратуру приема/передачи. Синхросимволы следуют в начале каждого сообщения (не менее четырех) и могут вставляться внутри сообщения до ETX при потере синхронизации. Синхросимволы игнорируются при подсчете продольной нечетности (см.ниже).

SON=1 - START OF HEADER. Признак начала сообщения.

SITE=70-7F - номер индивидуального терминала (их может быть несколько на одной линии, в описываемом случае 12, номера от 70 до 7F). В дальнейшем мы будем называть их сайтами.

STATION - номер экранного пульта индивидуального терминала (их в принципе тоже может быть несколько для разных типов терминалов и протоколов, но в описываемом протоколе - 1). Кроме этого, 4-ый разряд этого кода используется для контроля правильности посылок (см. ниже).

^{)} Коды всех символов приводятся в шестнадцатеричной системе счисления, без учета значения седьмого бита.

- CC - CONTROL code . Код, определяющий тип сообщения (см.ниже).
- DATA - 8-разрядные символы передаваемого текста. В описываемом случае кодировка ASCII . Количество символов от 0 до 1040.
- ESCAPE=1B - код-признак. Означает, что непосредственно следующий за ним код следует рассматривать как управляющий (например, LF- или E-код). Этот ESCAPE-код может встречаться в передаваемом тексте неоднократно, по мере надобности.
- E-код - один из управляющих кодов в сообщениях типа READ и WRITE . E - код = 22 или 42 означает, что центральная ЭВМ обменивается с экранными пультами. (E-код=20 в сообщениях типа WRITE означает, что данные должны быть распечатаны на АЦПУ терминала, E- код=21 в сообщении READ означает, что данные передаются в центральную ЭВМ с устройства ввода перфокарт. Эту часть протокола обмена мы не рассматриваем).
- ETX=03 - END OF TEXT , конец сообщения (предпоследний байт в сообщении).
- MP - MESSAGE PARITY , код продольной нечетности (последний байт в сообщении). Значение каждого бита (кроме седьмого) определяется следующей булевой функцией:
- $$f(b_1^i, b_2^i, \dots, b_n^i) = b_1^i \oplus b_2^i \oplus \dots \oplus b_n^i, \quad 0 \leq i \leq 6$$
- где b_j^i - значение i -го бита j -го кода, считая от кода SON до ETX включительно и не учитывая синхрокодов. Значение седьмого бита определяется, как и в любом другом байте. Такой контроль называется продольным контролем с дополнением до нечетного числа единиц.
- Использование продольно-поперечного контроля обеспечивает обнаружение ошибки в каждом сообщении (если, конечно, оно принято) с нечетным количеством ошибок, во всех сообщениях с двумя ошибками и в некоторых сообщениях с четным числом ошибок. Существуют и более эффективные методы обнаружения ошибок^{4/}.

Типы сообщений

На рисунке 1 приведены типы сообщений (их символические обозначения), их 16-ричные коды и направление передачи.

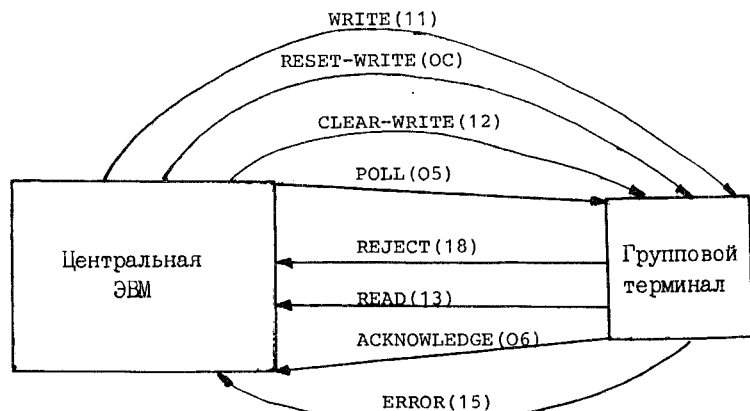


Рис. 1

Теперь рассмотрим каждое сообщение более детально.

POLL -сообщение

Как мы уже отмечали, на одной линии находятся до 12 одновременно обслуживаемых терминалов, 12 "сайтов". POLL -сообщение, посылаемое центральной ЭВМ, сканирует эти сайты в поисках READ -сообщения, т.е. готовых к передаче в ЭВМ данных (центральная ЭВМ использует соответствующий код сайта). Если READ -сообщение готово на каком-нибудь сайте, то оно отправляется в ответ на POLL -сообщение, в противном случае отправляется REJECT -сообщение. В этом случае в POLL -сообщении используются STATION -код 60 либо 70, причем 4-й разряд STATION -кода будет противоположным 4-му разряду STATION -кода последнего правильно принятого на этот сайт сообщения типа WRITE, содержащего выводимые на терминал данные (см. ниже).

Ниже мы опишем еще один случай использования POLL -сообщения.

На рисунке 2 приведен формат POLL -сообщения. Здесь код сайта равен 71, а код STATION равен 60.

Символическое обозначение	Семибитовый код	Двоичный вид							
		7	6	5	4	3	2	1	0
SYNC	16	0	0	0	1	0	1	1	0
SYNC	16	0	0	0	1	0	1	1	0
SYNC	16	0	0	0	1	0	1	1	0
SYNC	16	0	0	0	1	0	1	1	0
SOH	01	0	0	0	0	0	0	0	1
SITE	71	1	1	1	1	0	0	0	1
STATION	60	1	1	1	0	0	0	0	0
POLL	05	1	0	0	0	0	1	0	1
ETX	03	1	0	0	0	0	0	1	1
MP	69	1	1	1	0	1	0	0	1

Рис. 2

REJECT -сообщение

REJECT -сообщение используется:

- в ответ на POLL-сообщение. Сайт сообщает, что у него нет READ -сообщения. В этом случае используется код STATION - 60 либо 70. 4-й разряд в STATION -коде будет таким же, как 4-й разряд в STATION -коде последнего правильно принятого на этот сайт сообщения типа WRITE;

- в ответ на любое сообщение, если сайт не может это сообщение принять (например, занят буфер для этого сообщения). В таком случае используется тот код STATION, который был в принимаемом сообщении. На рисунке 3 приведен формат REJECT -сообщения. Здесь код сайта равен 77, а код STATION равен 61.

Символическое обозначение	Семибитовый код	Двоичный вид							
		7	6	5	4	3	2	1	0
SYNC	16	0	0	0	1	3	1	1	0
SYNC	16	0	0	0	1	0	1	1	0
SYNC	16	0	0	0	1	0	1	1	0
SYNC	16	0	0	0	1	0	1	1	0
SOH	01	0	0	0	0	0	0	0	1
SITE	77	0	1	1	1	0	1	1	1
STATION	61	0	1	1	0	0	0	0	1
REJECT	18	1	0	0	0	1	0	0	0
ETX	03	1	0	0	0	0	0	1	1
MP	73	0	1	1	1	0	0	1	1

Рис. 3

ACKNOWLEDGE -сообщение

Если сайт принял сообщение WRITE, CLEAR-WRITE или RESET - WRITE и в принятом сообщении не было обнаружено ошибок, то сайт обязан ответить ACKNOWLEDGE -сообщением, сигнализируя этим центральной ЭЕМ, что информация получена без ошибок. Код STATION в ACKNOWLEDGE -сообщении должен быть таким же, как и в принятом сообщении WRITE.

На рисунке 4 приведен формат ACKNOWLEDGE -сообщения. Здесь код сайта равен 7I, а код STATION равен 6I.

Символическое название	Семибитовый код	Двоичный вид							
		7	6	5	4	3	2	1	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SON	0I	0	0	0	0	0	0	0	I
SITE	7I	I	I	I	I	0	0	0	I
STATION	6I	0	I	I	0	0	0	0	I
ACKNOWLEDGE	06	I	0	0	0	0	I	I	0
ETX	03	I	0	0	0	0	0	I	I
MP	6B	0	I	I	0	I	0	I	I

Рис. 4

READ -сообщение

Если на клавиатуре дисплея сайта набрана строка, то ее содержимое передается в центральную ЭЕМ READ -сообщением в ответ на ROLL -сообщение. В этом случае код STATION должен быть таким же, как у последнего правильно принятого на этот сайт сообщения типа WRITE.

На рисунке 5 приведен формат READ -сообщения. Предполагается, что была набрана строка EDITOR, код сайта равен 7I, а код STATION равен 6I.

Рис. 5.

Символическое обозначение	Семибитовый код	Двоичный вид							
		7	6	5	4	3	2	1	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SON	0I	0	0	0	0	0	0	0	I

Рис.5 (продолжение)

Символическое обозначение	Семибитовый код	Двоичный вид							
		7	6	5	4	3	2	1	0
SITE	7I	I	I	I	I	0	0	0	I
STATION	6I	0	I	I	0	0	0	0	I
READ	I3	0	0	0	I	0	0	I	I
E	45	0	I	0	0	0	I	0	I
D	44	I	I	0	0	0	I	0	0
I	49	0	I	0	0	I	0	0	I
T	54	0	I	0	I	0	I	0	0
O	4F	0	I	0	0	I	I	I	I
R	52	0	I	0	I	0	0	I	0
ESCAPE	IB	I	0	0	I	I	0	I	I
E-код	42	I	I	0	0	0	0	I	0
ETX	03	I	0	0	0	0	0	I	I
MP	24	0	0	I	0	0	I	0	0

ERROR -сообщение

ERROR -сообщение высылается сайтом в ответ на любое сообщение центральной ЭЕМ в следующих случаях:

- неопознан тип присланного сообщения,
- неверен E-код в сообщении типа WRITE,
- неверен STATION -код,
- обнаружены ошибки при продольно-поперечном контроле.

Код STATION в ERROR -сообщении должен быть таким же, как и в последнем правильно принятом на этот сайт сообщении типа WRITE. На рисунке 6 приведен формат ERROR -сообщения. Здесь код сайта равен 72, а код STATION равен 6I.

Рис. 6

Символическое обозначение	Семибитовый код	Двоичный вид							
		7	6	5	4	3	2	1	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SON	0I	0	0	0	0	0	0	0	I
SITE	72	I	I	I	I	0	0	I	0
STATION	6I	0	I	I	0	0	0	0	I
ERROR	I5	0	0	0	I	0	I	0	I
ETX	03	I	0	0	0	0	0	I	I
MP	7B	I	I	I	I	I	0	I	I

WRITE -сообщение (CLEAR-WRITE, RESET-WRITE)

Сообщения WRITE, CLEAR-WRITE и RESET-WRITE отличаются лишь тем, что данные WRITE -сообщения высвечиваются на экран, начиная с текущей позиции курсора, данные RESET - WRITE высвечиваются после предварительной установки курсора в начало экрана (без чистки), а данные CLEAR - WRITE так же, как и при RESET - WRITE, но с предварительной чисткой экрана. Необходимость различать эти сообщения возникла из-за того, что на экранных пультах группового терминала CDC - 734, для которого разрабатывался протокол моды 4А, отсутствует ROLL -режим. Для дисплеев, имеющих ROLL -режим (например, для "ВИДЕОТОН-340"), сообщения WRITE, CLEAR - WRITE и RESET - WRITE можно считать эквивалентными.

Инициатива обмена всегда принадлежит центральной машине, поэтому обмен начинается с посылки центральной машиной WRITE -сообщения, причем все последующие посылки центральной ЭВМ идут с перевернутым 4-ым разрядом в STATION -коде, включая и следующее WRITE-сообщение, после которого опять все последующие посылки идут с перевернутым 4-ым разрядом в STATION -коде. Правила формирования 4-го разряда в STATION -коде в других типах сообщений, собственно, описаны выше. Программный драйвер центральной ЭВМ^{2/} контролирует в полученных от сайта сообщениях состояние указанного разряда и, если состояние этого разряда противоположно ожидаемому, повторяет предыдущую посылку. Поясним это на примере^{*)}:

Номер сообщения	Сообщение от CDC-6500	Код STATION в сообщении	Ответ сайта	Код STATION в ответе
1	WRITE	7I	ACKNOWLEDGE	7I
2	POLL	60	REJECT	70
3	POLL	60	REJECT	70
4	WRITE	6I	ACKNOWLEDGE	6I
5	POLL	70	READ	6I
6	WRITE	7I	ERROR	6I
7	WRITE	7I	ACKNOWLEDGE	7I

*) Для простоты здесь (как и далее) приводится пример обслуживания одного сайта. Обслуживание многих сайтов производится последовательно (см. трассировку протокола обмена).

Посылка номер 7 является повторением посылки номер 6, потому что программный драйвер центральной ЭВМ обнаружил код ERROR -сообщения и смену 4-ого разряда в STATION - коде в ответном сообщении. На рисунке 7 приведен формат WRITE -сообщения. Предполагается, что для высветки на экран послан текст COMMAND-, код сайта равен 74, а код STATION равен 6I.

Символическое обозначение	Семибитовый код	Двоичный вид							
		7	6	5	4	3	2	1	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SYNC	I6	0	0	0	I	0	I	I	0
SOH	0I	0	0	0	0	0	0	0	I
SITE	74	I	I	I	I	0	I	0	0
STATION	6I	0	I	I	0	0	0	0	I
WRITE	II	I	0	0	I	0	0	0	I
C	43	0	I	0	0	0	0	I	I
O	4F	0	I	0	0	I	I	I	I
M	4D	I	I	0	0	I	I	0	I
M	4D	I	I	0	0	I	I	0	I
A	4I	I	I	0	0	0	0	0	I
N	4E	I	I	0	0	I	I	I	0
D	44	I	I	0	0	0	I	0	0
-	2D	I	0	I	0	I	I	0	I
ESCAPE	1B	0	0	0	I	I	0	I	I
E - код	42	I	I	0	0	0	0	I	0
ETX	03	I	0	0	0	0	0	I	I
MP	4A	0	I	0	0	I	0	I	0

Рис.7

Контроль, при потере ответных сообщений сайта

Контроль 4-го разряда кода STATION в ответных сообщениях сайта является хорошим средством, обеспечивающим достаточно надежно повторение сообщений, принятых ранее с ошибкой. Однако может случиться и так, что центральная ЭВМ вообще не получит за ожидаемое время ответного сообщения или не опознает его, тем не менее машина должна, во-первых, не повторять то сообщение, которое дошло до сайта правильно, несмотря на то, что центральная ЭВМ не получила подтверждение ACKNOWLEDGE -сообщением, и, во-вторых, повторить то со-

сообщение, которое принято сайтом неверно, несмотря на то, что центральная ЭВМ не получила error -сообщения. В таких случаях центральная ЭВМ высылает POLL -сообщение с кодом STATION , равным 6I или 7I, сигнализируя сайту (единицей в нулевом бите) о том, что машина не получила предыдущего ответного сообщения. Поясним это на следующих двух примерах.

Пример 1

Номер сообщения	Сообщение от CDC-6500	Код STATION в сообщении	Ответ сайта	Код STATION в ответе
1	WRITE	7I	ACKNOWLEDGE	7I
2	WRITE	6I	ACKNOWLEDGE	6I
3	WRITE	7I	ACKNOWLEDGE*	7I
4	POLL	6I или 7I	REJECT	70
5	WRITE	6I	ACKNOWLEDGE	6I

Получив сообщение № 3, сайт ответил сообщением ACKNOWLEDGE, которое программный драйвер центральной ЭВМ не получил. В этом случае центральная ЭВМ высылает POLL -сообщение с кодом STATION 6I или 7I (а не 60, как в обычном случае), сайт отвечает REJECT -сообщением с кодом STATION , равным 70 (4-ый разряд кода STATION равен 4-му разряду кода STATION последнего правильно принятого WRITE -сообщения). ЭВМ не повторяет сообщения № 3, т.е. видит (по 4-ому разряду кода STATION в REJECT -сообщении), что сообщение № 3 принято правильно.

Пример 2

Номер сообщения	Сообщение от CDC-6500	Код STATION в сообщении	Ответ сайта	Код STATION в ответе
1	WRITE	7I	ACKNOWLEDGE	7I
2	WRITE	6I	ACKNOWLEDGE	6I
3	WRITE	7I	ERROR*	6I
4	POLL	6I или 7I	REJECT	60
5	WRITE	7I	ACKNOWLEDGE	7I

Получив сообщение номер 3 и обнаружив в нем ошибки, сайт выслал сообщение ERROR , которое центральная ЭВМ не получила. В этом случае машина высылает POLL -сообщение с кодом STATION 6I или 7I, на которое сайт отвечает REJECT -сообщением с кодом STATION , равным 60 (4-ый разряд этого кода равен 4-му разряду кода STATION последнего правильно принятого WRITE -сообщения).

Центральная ЭВМ повторяет сообщение номер 3 (это сообщение номер 5), т.к. видит, что сообщение № 3 или пропало, или было принято с ошибкой.

Эмуляция протокола обмена моды 4А для системы концентратора терминалов и CDC-6500 на микропроцессорном устройстве связи

Технические характеристики и средства программирования микропроцессорного устройства связи (УС) уже достаточно подробно описывались^{5-7/}; напомним лишь, что основным блоком является микро-ЭВМ на базе микропроцессора INTEL-8085А с оперативной памятью 4К байтов (K=1024).

Программа, размещенная в ОЗУ микропроцессора, поддерживает протокол обмена моды 4А с ЭВМ CDC-6500 , буферизует принимаемые от CDC - 6500 данные и обменивается массивами информации с ЭВМ EC-1010 (EC-1010 имеет прямой доступ к ОЗУ УС, поэтому УС только общается EC-1010, можно ли передавать информацию в ОЗУ УС или нужно ли считывать информацию из ОЗУ УС).

Программное обеспечение УС

Программное обеспечение УС, поддерживающего протокол обмена, состоит из 4-х модулей:

STARTT

После инициации УС^{7/} управление передается в нулевую ячейку ОЗУ УС, начиная с которой располагается модуль STARTT . В этом модуле:

- происходит программирование (на прием байтов в синхронном режиме) универсального синхронно-асинхронного приемника-передатчика INTEL - 8251А (USART),
- вырабатывается сигнал прерывания в EC-1010 по причине I (см.ниже),
- передается управление на блок команд, обеспечивающих ожидание прерываний, поступающих от USARTа.

В модуле STARTT по фиксированному адресу содержится также команда перехода на подпрограмму обработки прерываний.

USART вырабатывает прерывание микропроцессора по следующим причинам:

- с линии (из CDC-6500) поступил байт в буфер USARTа , и его можно считать в ОЗУ УС (команда IN),

- USART готов принять байт в свой буфер из ОЗУ УС и передать его в линию (команда OUT).

DATAT

Этот модуль содержит только константы, рабочие ячейки и буфера, необходимые для поддержания протокола обмена. Здесь содержится буфер, в который ЕС-1010 помещает информацию, набранную на терминалах, работающих с CDC - 6500 для передачи в CDC - 6500 (в режиме прямого доступа в ОЗУ УС). Этот буфер мы будем называть буфером А. Здесь также содержится буфер, из которого ЕС-1010 забирает (в режиме прямого доступа в ОЗУ УС) информацию, пришедшую из CDC-6500 для высвечивания на экранах терминалов. Этот буфер мы будем называть буфером Б.

Буфер А состоит из 90 байтов (80 байтов - максимальная длина строки, набранной на дисплеях системы концентратора терминалов; 10 байтов - формируемая служебная информация).

Буфер Б организован как циркулярный буфер. Длина его выбирается исходя из следующих соображений.

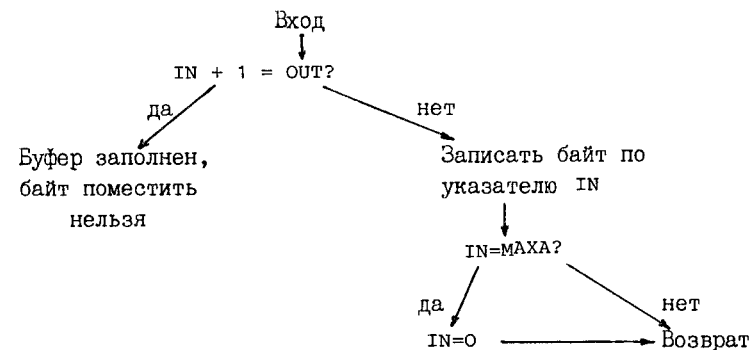
Длинные сообщения (максимальная длина равна 1040 байтам) относительно редки, и ЕС-1010 забирает информацию из буфера Б независимо от его заполнения (порциями по 80 байтов), поэтому длина буфера была выбрана равной 1350 байтов в расчете на то, что ЕС-1010 будет успевать освобождать место в буфере Б под поступающие из CDC-6500 сообщения. Эксплуатация системы показала, что такое событие, как заполнение буфера, а, следовательно, временный отказ (пока не освободится место) от поступающей из CDC-6500 информации, происходит не чаще 1-2 раз за 7 часов работы.

Напомним еще раз, что на одной линии в нашем случае находится до 12 одновременно работающих сайтов и при простой организации буферов нам потребовалось бы либо держать в ОЗУ УС буфер под каждый сайт, что заняло бы 12x1040 байтов (это намного превышает возможности УС), либо резко увеличить число отказов в приеме информации из CDC-6500.

RS30TT

В этом модуле происходит прием байта с линии по сигналу прерывания, выработанному USART'ом, распознавание байта и накопление всего сообщения. Здесь же после приема последнего байта готовится ответ центральной машине на принятое сообщение. Если в процессе приема выясняется, что это write-сообщение, то информационная его часть будет помещаться в буфер Б, в противном случае сообщение накапливается в специальных служебных буферах. На рисунке 8 приведен алгоритм записи очередного байта в буфер Б. Здесь IN - указатель

в буфере Б, по которому записывается байт (этот указатель изменяется в модуле RS30TT), OUT - указатель, по которому считывается байт из буфера Б (этот указатель меняется в ЕС-1010, и изменения сообщаются в УС), MAXA - максимальный относительный адрес в буфере Б.



Начальное состояние IN=0, OUT = MAXA

Рис.8

Для экономии места в буфере Б пробелы записываются в упакованном виде (по формуле $80_{16}+n-1$, где n - число пробелов). Если по окончании приема в сообщении обнаружилась ошибка, то указатель IN возвращается к положению, которое у него было до приема этого сообщения.

После того, как принят и обработан последний байт сообщения, в RS30TT происходит перепрограммирование USARTа на передачу байтов в синхронном режиме.

RS34TT

В этом модуле происходит передача в CDC - 6500 подготовленного сообщения. Здесь же могут быть выработаны сигналы прерывания по I5-му уровню в ЭВМ ЕС-1010.

Существуют две причины прерываний по I5-му уровню в ЭВМ ЕС-1010:

1. Буфер А свободен (т.е. информация, поступившая из ЕС-1010, передана в CDC-6500).
2. В пустой буфер Б поступило сообщение от CDC-6500 (после этого

прерывания ЕС-IOIO может начинать работу по освобождению буфера Б; как легко видно, прерывание по причине 2 вырабатывается не на каждое поступившее в буфер Б сообщение).

Передав последний байт в СДС - 6500, RS34TT перепрограммирует USART на прием байтов в синхронном режиме.

Программное обеспечение концентратора терминалов для работы с CDC - 6500

TASK15

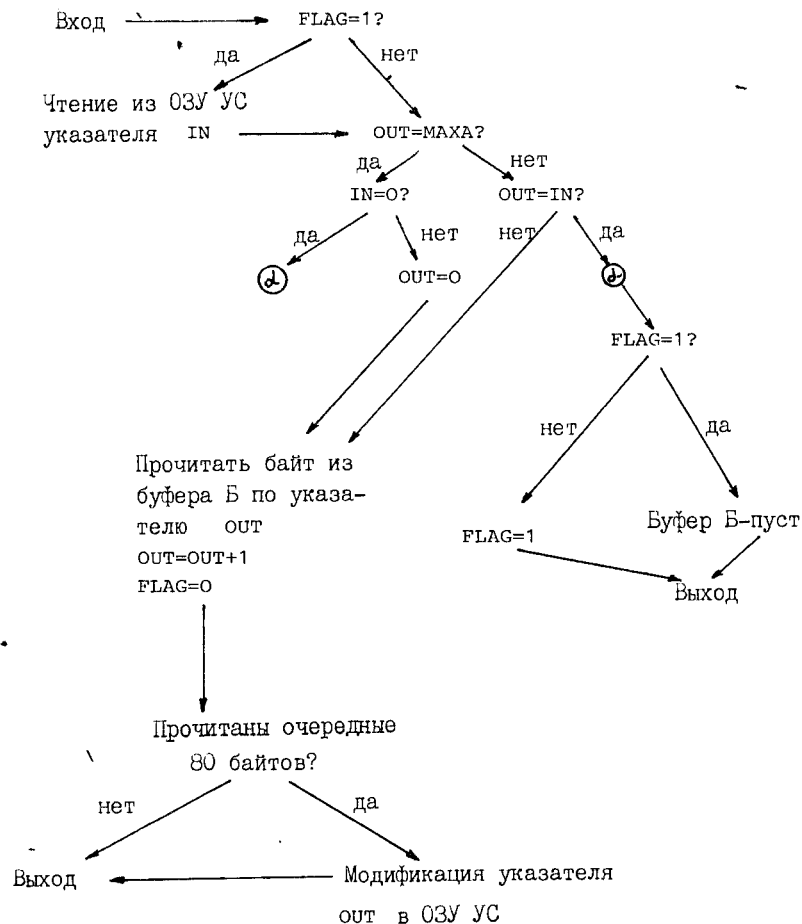
Непосредственная задача, "прикрепленная" к 15-му уровню прерываний ЭВМ ЕС-IOIO, TASK15, распознает причину прерываний и, если это причина 1, то записывает по адресу FFFF МОЗУ ЕС-IOIO единицу, если причина 2, то записывает единицу по адресу FFFF. Затем деактивирует 15-й уровень.

ININT

Планировщик системы концентратора терминалов /В/ обращается к секции ININT, если в байте по адресу FFFF записана единица и какой-нибудь из терминалов концентратора, находящийся в режиме работы с ЭВМ CDC - 6500, закончил ввод. Секция ININT определяет номер этого терминала, а также номер сайта, который логически прикреплен к данному терминалу на время его работы в режиме работы с ЭВМ CDC - 6500. После этого номер сайта и содержимое терминального буфера (набранная строка) переправляется в буфер А УС. В конце своей работы секция ININT записывает ноль по адресу FFFF.

OUTINT

Планировщик системы концентратора обращается к секции OUTINT в случае, если в байте по адресу FFFF записана единица. Секция OUTINT организует прием в терминальные буфера из буфера Б, определяя по номерам сайтов адреса соответствующих терминальных буферов (в случае надобности происходит распаковка пробелов). Здесь же организуется выдача информации на терминал очередной принятой порции (после приема очередных 80 байтов управление передается планировщику концентратора). Если в процессе приема оказывается, что вся информация из буфера Б забрана на концентратор (буфер Б пуст), то по адресу FFFF записывается ноль. На рисунке 9 приведен алгоритм чтения байта из буфера Б. Здесь in, out, маха - то же, что и на рисунке 9, FLAG - рабочая ячейка.



Начальное состояние FLAG=1

Рис.9

Трассировка протокола обмена на ЭВМ ЕС-IOIO

В программное обеспечение концентратора терминалов встроены средства трассировки протокола обмена. Трассировка может осуществляться (прекращаться) специальными действиями оператора ЭВМ ЕС-IOIO. Результат трассировки выдается на внешнее устройство ЭВМ ЕС-IOIO, закрепленное за операционной меткой M:LO. На рисунке 10 приведен

пример трассировки протокола обмена. Здесь формат каждой строки следующий:

$S_1 S_2 t_1 t_2 C_1 C_2 S_3 S_4 t_3 t_4 C_3 C_4$

где $S_1 S_2$ - код сайта в принимаемом сообщении,
 $t_1 t_2$ - код STATION в принимаемом сообщении,
 $C_1 C_2$ - control code в принимаемом сообщении,
 $S_3 S_4$ - код сайта в ответном сообщении,
 $t_3 t_4$ - код STATION в ответном сообщении,
 $C_3 C_4$ - control code в ответном сообщении.

	Продолжение	Продолжение	Продолжение
707111707106	757005756018	797005796018	707111707106
717005716018	767005766018	7A70057A6018	717005716018
727005726018	777005776018	7B70057B6018	727005726018
737005736018	787005786018	7C70057C6018	737005736018
747005746018	797005796018	7D70057D6018	747005746018
757005756018	7A70057A6018	7E70057E6018	757005756018
767005766018	7B70057B6018	7F70057F6018	767005766018
777005776018	7C70057C6018	707005706018	777005776018
787005786018	7D70057D6018	717005716113	787005786018
797005796018	7E70057E6018	727005726018	797005796018
7A70057A6018	7F70057F6018	737005736018	7A70057A6018
7B70057B6018	707005706018	747005746018	7B70057B6018
7C70057C6018	717111717106	757005756018	7C70057C6018
7D70057D6018	727005726018	767005766018	7D70057D6018
7E70057E6018	737005736018	777005776018	7E70057E6018
7F70057F6018	747005746018	787005786018	7F70057F6018
707005706018	757005756018	797005796018	707005706018
717005716018	767005766018	7A70057A6018	717005716018
727005726018	777005776018	7B70057B6018	727005726018
737005736018	787005786018	7C70057C6018	737005736018
747005746018	797005796018	7D70057D6018	747005746018

Рис.10

В заключение автор считает приятным долгом выразить искреннюю благодарность В.В.Галактионову, В.П.Ширикову и Е.Д.Федюнькину за полезные советы и обсуждения.

Литература

1. Аниховский В.Е. и др. ОИЯИ, ПИ-12809, Дубна, 1980.
2. Шириков В.П. ОИЯИ, ВП-11-12663, Дубна, 1979.
3. WATSON TERMINAL, CONTROL DATA, REFERENCE MANUAL, Minnesota 55113.
4. Сипсер Р. Архитектура связи в распределенных системах, т.1, "Мир", М., 1981.
5. Аниховский В.Е. и др. ОИЯИ, ПИ-81-853, Дубна, 1981.
6. Мазепа Е.Ю. ОИЯИ, ВП-11-82-449, Дубна, 1982.
7. Асмолов А.Г., Мазепа Е.Ю. ОИЯИ, ПИ-82-198, Дубна, 1982.
8. Галактионов В.В. и др. В кн.: Тезисы докладов Всесоюзной конференции "Диалог человек-ЭВМ". Л., 1982, с.90-92.

Рукопись поступила в издательский отдел
 19 октября 1982 года.

Мазепа Е.Ю.

11-82-738

Протокол обмена информацией с групповыми терминалами ЭВМ CDC-6500 и его реализация в системе коллективного пользования на базе ЭВМ БЭСМ-6, CDC-6500 и концентратора терминалов

Подробно описан протокол обмена информацией с групповым терминалом ЭВМ CDC-6500, а также его реализация в системе концентратора терминалов. Приведены методы контроля информации, описаны принципы работы с буферной ЭВМ, названной устройством связи.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1982

Mazepa E.Yu.

11-82-738

The Protocol of Information Exchange with CDC-6500 Computer Batch Terminal and Its Implementation in the System of Collective Sharing on the Base of the BESM-6, CDC-6500 Computers and of Terminal Concentrator

The protocol of information exchange of the CDC-6500 computer batch terminal and its implementation in the system of terminal concentrator is described. The methods of information checking are described.

The investigation has been performed at the Laboratory of Computing Techniques and Automation, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna 1982

Перевод О.С.Виноградовой.