

инв. № 2003
экз. 2

СИСТЕМА ЭЛЕКТРОННЫХ ВЫЧИСЛИТЕЛЬНЫХ МАШИН

ОПЕРАЦИОННАЯ СИСТЕМА

Средства программирования

Микропрограммное обеспечение алфавитно-цифровых дисплеев ЕС-7906

Руководство программиста

ЦБТ. 804. 003. Д56

1980

ЕДИНАЯ СИСТЕМА ЭЛЕКТРОННЫХ ВЫЧИСЛИТЕЛЬНЫХ МАШИН
ОПЕРАЦИОННАЯ СИСТЕМА
Средства программирования
для комплекса алфавитно-цифровых дисплеев ЕС-7906
Руководство программиста
Ц51.804.003 Д56

1980

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
I. УПРАВЛЕНИЕ ВВОДОМ-ВЫВОДОМ	II
I.1. Общие сведения	II
I.2. Макрокоманды ввода-вывода	13
I.2.1. Макрокоманда блока управления данными - DCB ..	15
I.2.2. Макрокоманда открытия блока управления данными - OPEN	19
I.2.3. Макрокоманда считывания знаковых данных - GREAD	22
I.2.4. Макрокоманда записи знаковых данных - GWRITE	31
I.2.5. Макрокоманда стирания буфера индикатора - GCNTRL	39
I.2.6. Макрокоманда закрытия блока управления данными - CLOSE	42
I.3. Модификация DCB во время выполнения	44
2. БАЗИСНЫЙ МЕТОД ОБРАБОТКИ ВНИМАНИЯ	46
2.1. Функции базисного метода обработки сигналов внимания	47
2.2. Макрокоманды базисного метода обработки сигналов внимания	49
2.2.1. Макрокоманда определения условий асинхрон- ного входа - BAEC	49

Ц51.804.003 Д56

ЕС ЭВМ ОПЕРАЦИОННАЯ СИСТЕМА Средства программирования для комплекса алфавитно- цифровых дисплеев ЕС-7906 Руководство программиста		
Лит	Лист	Листов
Ф А	2	94
Ф	(4)	

2.2.2. Макрокоманда включения программы внимания - SPAR	53
2.2.3. Макрокоманда выключения программы внимания - DAR	56
2.2.4. Макрокоманда запроса внимания - ATTNINQ	58
2.3. Пользовательские программы обработки внимания	61
2.3.1. Включение пользовательской программы внимания	61
2.3.2. Обработка внимания	62
2.3.3. Синхронизация пользовательских программ	64
2.4. Использование средств базисной обработки внимания при MVT	64
3. СПЕЦИАЛЬНЫЙ МЕТОД ОБРАБОТКИ ВНИМАНИЯ	66
3.1. Общие сведения	66
3.2. Макрокоманды специальной обработки сигналов внимания	67
3.2.1. Макрокоманда анализа графического внимания - ANALYZ	67
3.2.2. Макрокоманда обслуживания графического внимания - GSERV	74
ПРИЛОЖЕНИЕ 1	76
ПРИЛОЖЕНИЕ 2	81
ПРИЛОЖЕНИЕ 3	88

ВВЕДЕНИЕ

Данный документ содержит инструкции по программированию на языке ассемблера для комплекса ЕС-7906. В нем детально описаны макрокоманды и стандартные управляющие программы графического метода доступа операционной системы ОС ЕС в применении к ЕС-7066. Для иллюстрации использования макрокоманд включены примеры с соответствующими пояснениями.

Для более полного понимания работы операционной системы и, в частности, программного обеспечения ЕС-7066 читатель должен быть знаком с содержанием [1], [2], [3].

Комплекс ЕС-7906 представляет собой систему индикаторов на электроннолучевой трубке (ЭЛТ) ЕС-7066, подключенных к общему устройству управления ЕС-7566 и предназначенных для ввода-вывода алфавитно-цифровых данных на экран.

Блок сопряжения подключает комплекс к селекторному или мультиплексному каналу ЕС ЭВМ.

Индикаторы ЕС-7066 подключаются к устройству управления ЕС-7566 через панель расширения максимум четырьмя ветвями. Общее количество индикаторов - до 16. Распределение по ветвям произвольное.

Для отображения информации на экране могут быть использованы четыре формата, перечисленные в табл. I.

№ здел	165-4566/64	гкб	УУ.У
Нзм	Лист	№ докум.	Полл.

Таблица I

Форматы отображения информации на экране

Код формата	Количество строк в кадре	Количество символов в строке
00	12	80
01	6	80
10	12	40
11	6	40

В зависимости от формата к устройству управления могут подключаться четыре индикатора ЕС-7066 формата 00, или восемь индикаторов формата 01 или 10, или шестнадцать индикаторов формата 11. Формат задается органами ручного управления индикатора.

Максимальный объем данных, изображаемых на всех экранах, определяется размером буферного ЗУ и не превышает 3840 знаков.

Индикаторы ЕС-7066 могут удаляться от устройства управления на расстояние до 600 м.

В состав комплекса алфавитно-цифровых дисплеев ЕС-7906 входит печатающее устройство (типа "консул"), которое может использоваться для документирования обрабатываемой на дисплеях информации. Печатающее устройство занимает место индикатора с адресом 15. Изображение на экране любого индикатора данные при нажатии соответствующей функциональной клавиши могут быть пересланы в буферную память печатающего устройства и отпечатаны. Кроме того, данные могут выводиться на печатающее устройство прямо из ЭВМ.

Каждый индикатор имеет в своем составе клавиатуру для ввода данных, при помощи которой оператор может записать, стереть, за-

Изм	Лист	№ докум.	Подп.	Дат

Ц51.804.003 л56

Лист
5

просить необходимые данные из ЭВМ, откорректировать и отослать данные к печатающему устройству или в ЭВМ.

В режиме ввода оператор набирает на клавиатуре индикатора необходимые данные, которые автоматически записываются в буферную память этого индикатора и отображаются на экране. По окончании ввода данных оператор посыпает в ЭВМ сигнал внимания (нажимая соответствующую функциональную клавишу), по которому ЭВМ обращается к буферной памяти и считывает записанные в ней данные.

В режиме вывода данных из ЭВМ данные записываются в буферную память индикатора или печатающего устройства и отображаются на экране или печатаются. Для отображения данных на экране устройством управления производится циклический опрос буферной памяти (перезапись или регенерация), которая выдает коды символов поочередно для всех индикаторов. Каждый индикатор воспринимает только свои данные.

Средства программирования операционной системы для индикатора ЕС-7066 включают макрокоманды и управляющие стандартные программы. Они используются для двух ключевых функций: управления вводом-выводом и обработки сигнала внимания. Макрокоманды функционируют как часть обрабатывающей программы; управляющие стандартные программы функционируют как часть управляющей программы. Эти средства программирования составляют графический метод доступа для ЕС-7066.

Макрокоманды управления вводом-выводом используются для передачи данных или функций управления, для создания блоков управления данными (ДСВ) и для установления и устранения связи между обрабатывающей программой и дисплеями. Стандартные программы управления вводом-выводом создают слова команды канала (ССВ), формируют канальные программы и управляют передачей данных между основной памятью процессора и буферной памятью дисплея.

Сигналы внимания - асинхронные (незапланированные) прерывания

Задачи	Из 15 4566/б/1	Лист 1/9
Лист	№ докум.	Полл.

Ц51.804.003 Д56

Лист
6

ввода-вывода процессора. Они вызываются нажатием специальной клавиши клавиатуры индикатора ЕС-7066.

Средства базисного метода обработки сигналов внимания автоматически обнаруживают сигналы и направляют их к соответствующим программам пользователя обработки внимания. Макрокомандами, предусмотренными средствами графического программирования (*SAEC*, *SPAR*, *ATTNINQ* и *DAK*), пользователь ЭМ (пользователь) определяет свои программы обработки внимания и делает их доступными для управляющей программы.

Средства специальной обработки внимания позволяют пользователю проверять наличие сигналов внимания от ЕС-7066 при помощи макрокоманд *ANALYZ* и *GSERV*. Обнаружение сигналов внимания не автоматическое; обрабатывающая программа должна периодически проверять их наличие, с тем чтобы сигналы внимания не терялись. Когда обнаруживаются сигналы внимания, управление передается к определенным программам, предусмотренным пользователем.

Синхронные ошибки (т.е. ошибки ввода-вывода, которые встречаются при операциях в каналах ЭМ) обрабатываются стандартной программой обработки типичных ошибок ЕС-7906. Эта программа анализирует информацию о синхронной ошибке в байтах состояния и уточненного состояния и обеспечивает соответствующие процедуры исправления ошибок для типичных ситуаций.

Стандартная программа обработки ошибок сообщает оператору о том, что требуется его вмешательство, или что произошла ошибка в данных, или ошибка на "ШИН-К", или сбой в оборудовании (см. [4]).

Пользователь ЕС-7906 определяет конфигурацию комплекса для операционной системы, которую он генерирует, используя соответствующие макрокоманды *SYSGEN*. Требования генерации операционной системы описаны в [5].

Возможности мультипрограммного режима с переменным числом задач обеспечиваются средствами графического программирования,

З. зам	Ц5-4566761	Ркн	17876	Ц51.804.003 Л56	Лист
Лист	№ покум	Номп.	Лата		7

описанными в данном документе. Если при использовании этих средств в режиме MVT применяются специальные предположения, то они оговариваются в конце каждого раздела.

Каждая описанная в данном документе макрокоманда иллюстрируется форматом кодирования.

Общий вид формата кодирования:

Название	Операция	Операнд	Комментарий
метка или пробел	макронименование	либо отсутствует, либо один или несколько операндов, разделенных запятыми	любой текст в коде ДКОИ

При иллюстрировании формата кодирования макрокоманд применяются следующие соглашения:

- прописные (заглавные) буквы, цифры и пунктуационные знаки должны кодироваться программистом точно так же, как показано. Исключение составляют квадратные скобки [], фигурные скобки { } и многоеточия ..., которые никогда не кодируются;
- строчные (малые) буквы и слова являются переменными, вместо которых при кодировании программист должен подставлять указанную информацию или указанные значения;
- члены или группы членов в квадратных скобках [] не являются обязательными. Они могут быть опущены по усмотрению программиста;
- фигурные скобки { } определяют выбор одного операнда из группы;
- подчеркивание определяет выбор по умолчанию, т.е. если операнды не указаны, то подразумевается подчеркнутый операнд;

Л. № 145-4366/16/ План 7870
Лист № докум. Позл. Лат.

Ц51.804.003 д56

Лист 8

- вертикальная черта (|) обозначает исключающее ИЛИ.

Например, A|B обозначает, что программист может писать A или B.

Альтернативные операнды в некоторых случаях также указываются в вертикальном столбце внутри фигурных скобок;

- многоточие ... обозначает, что предшествующая синтаксическая единица (заключенная в квадратные или фигурные скобки) может быть повторена любое число раз;

- операнды отделяются запятыми;

- все операнды записываются в одну строку бланка кодирования (кроме случая, когда необходима карта продолжения), даже в том случае, когда в иллюстрации формата они могут занимать одну или несколько строк;

- за последним операндом всегда следует пробел;

- позиционные операнды должны кодироваться в последовательности указанной в соответствующей иллюстрации формата. Ключевые операнды могут записываться в любом порядке, но должны следовать за всеми позиционными операндами. На местах опущенных позиционных операндов, предшествующих последнему определенному позиционному операнду, должны стоять запятые.

Для определения способа кодирования операндов в приложении I дается обзор операндов. Приложение I содержит все макрокоманды и указывает допустимые способы записи операндов. Обзор составлен таким образом, что он может быть использован при изучении описания любой макрокоманды.

Если указывается L-форма макрокоманды или пропущен операнд формы (MF), обозначения регистров и индексов использоваться не могут; могут использоваться только перемещаемые выражения, абсолютные выражения или коды. При использовании L-формы макрокоманды поле наименования должно содержать символ. Если не указано противное, цифровые величины, используемые в качестве операндов макрокоманды,

должны быть целыми десятичными числами без предшествующих нулей.
Если нет особого указания, то пропуск операнда формы (MF) вызывает генерацию списка параметров из всех указанных operandов и выполнение макрокоманды, использующей этот список в исполнительное время, т.е. в время выполнения программы.

Соглашения связи, описанные в [6] применяются и к описанным здесь средствам программирования. Информация относительно назначения регистров для указанных функций связи содержится в данном документе.

В данном документе "буферная память индикатора" употребляется в смысле буферной памяти, размещенной физически в устройстве управления индикаторов (ЕС-7566), а неопределенный термин "регистр" обозначает "общий регистр" ЕС ЭВМ.

Лот	Ч5-7566/76/77	РКод	7.8.7
Зам	Лве	№ докум.	Подп.

Ч51.804.003 Д56

Уист
10

1. УПРАВЛЕНИЕ ВВОДОМ-ВЫВОДОМ

1.1. Общие сведения

Управление вводом-выводом осуществляется макрокомандами и соответствующими управляющими стандартными программами, которые сообща инициируют операции ввода-вывода, связанные с воспроизведением данных. Макрокоманды ввода-вывода и их функции для дисплея ЕС-7066 перечислены в табл. 2.

Примечание. Адрес области сохранения проблемной программы должен быть загружен в регистр I3 перед тем, как будет выдана любая макрокоманда ввода-вывода.

Для лучшего понимания последующего рассмотрения операций ввода-вывода кратко описаны функциональные соотношения между различными управляющими блоками системы, связанными с операциями ввода-вывода. Дополнительная информация содержится в [?].

Каждый блок управления данными *DCB* обычно создается макрокомандой *DCB* и открывается выполнением макрокоманды *OPEN*. *DCB*, содержащий ссылку к оператору *DD*, может быть связан с одним или несколькими индикаторами, подключенными к одному устройству управления ЕС-7566.

Изм	Лист	№ докум.	Подп.	Дата

Ц51.804.003 Д56

Лист
II

Таблица 2

Макрокоманды ввода-вывода

Мнемоника	Наименование макрокоманды	Функция
DCB	Блок управления данными	Создает блок управления данными DCB
OPEN	Открыть блок управления данными	Инициализирует один или несколько блоков управления данными
GREAD	Считать знаковые данные	Передает данные из буферной памяти индикатора в основную память
GWRITE	Записать знаковые данные	Передает данные из основной памяти в буфер индикатора или к печатающему устройству
GCNTRL	Стереть буфер индикатора	Стирает данные из буфера индикатора
CLOSE	Закрыть блок управления данными	Отключает один или несколько блоков управления данными

Несколько DCB могут открываться одной макрокомандой OPEN. В пользовательской области главной (основной) памяти макрокоманда OPEN создает блок расширения данных (DEV). В DEV включается таблица указателей адресов блоков управления устройствами (UCB), связанными с блоком управления данными. Для каждого индикатора и каждого печатающего устройства имеется по одному UCB, которое используется для описания соответствующих устройств.

Несколько макрокоманд ввода-вывода содержат операнд "устройство", который указывает конкретное устройство в группе индикаторов, связанной с данным DCB. Этот операнд используется как

Лист	№ докум.	Подп., Лат.	Ц51.804.003 Д56	Лист
1	12			

индекс для указания нужного элемента в списке указателей *UCB* блока расширения данных *DEA*. Значение операнда "устройство", выраженное целым десятичным числом, не должно превышать числа устройств, указанных в карте для набора данных.

Оператор *LL* может быть указан для каждого отдельного индикатора или печатающего устройства или для любой группы индикаторов (до 16), подключенных к одному устройству управления. Оператор определения данных (*LL*) описан в [8].

Значение операнда "устройство", уменьшенное на 1, помещается в блок управления событием данных (*DFCB*) при образовании его во время макрорасширения ("устройство"-1= "индекс *UCB*"). Во время выполнения стандартная программа управления вводом-выводом помещает это значение в блок ввода-вывода *I/OB*. Этим завершается связь между макрокомандой ввода-вывода и выбранным *UCB*, который, в свою очередь, указывает устройство, соответствующее макрокоманде.

1.2. Макрокоманды ввода-вывода

Макрокоманды данной группы выполняют функции ввода-вывода для ЕС-7066, которые включают инициализацию управляющих блоков и управление передачей данных между основной памятью и буфером индикатора. Они обеспечивают связь со стандартной программой управления вводом-выводом, которая инициализирует *I/OB*, создает канальные программы и выдает макрокоманду *EXCR*, обращающуюся к супервизору ввода-вывода для выполнения канальных программ.

Все макрокоманды ввода-вывода, за исключением макрокоманды *DCB*, могут использовать *L*- и *E*-формы макроопределения. Если записаны все операнды, за исключением ключевых *MF=L* и *MF=E*, то функции ввода-вывода выполняются при использовании указанных операндов. Если любой из требуемых операндов пропущен или указан неправильно во время выполнения, макрокоманда не выполняется. Деталь-

Изм	Лист	№ докум.	Полл.	Дат

ное описание форм макрокоманд и их использования содержится в
[9].

Таблица 3

Формат *DEC8*

Смещение от на- чала <i>DEC8</i> (в байтах)	Поле	Соответствующий операнд макрокоман- ды ввода-вывода
+00	Блок управления собы- тием <i>ECB</i>	
+04	Код типа	Тип
+08	Адрес блока управления данными <i>DCB</i>	Адрес <i>dcb</i>
+12	Адрес области	Адрес области
+16	Код неправильной длины или постоянной ошибки считывания	
+17	В случае неправильной длины счет остатка из <i>CSW</i> (в байтах)	
+20	Длина (в байтах)	Длина
+24	Резервируется	
+28	Индекс для указания элемента в <i>DEC</i> -табли- це указателей <i>UCB</i> , связанных с <i>DCB</i>	Устройство
+29	Единица для типов операций с восстанов- лением клавиатуры, нуль в противном случае.	
+30	Нули	
+31	Нули	

32-байтный *DCB* формируется во время макрорасширения; информация, полученная при анализе операндов макрокоманды, включается в соответствующие позиции *DCB*. Эти данные используются стандартной программой ввода-вывода для образования *ССМ*. Формат *DCB* для ЕС-7066 приведен в табл.3.

Примечание. Если операция ввода-вывода или некоторое другое событие зависит от завершения предыдущей операции ввода-вывода, то непосредственно за ней должна быть выдана макрокоманда *WAIT* (см. 9). Коды *FCB*, отмечающие состояние завершения и результат (успешный или с ошибкой) операции ввода-вывода, описаны в [7].

1.2.1. Макрокоманда блока управления данными - *DCB*

Эта макрокоманда резервирует память для блока управления данными и инициализирует его (см. табл. 4). В дополнение к основной информации, требующейся в *DCB*, память резервируется также для адресов стандартной программы управления вводом-выводом (*GIOCR*) и блоков ввода-вывода (*JOB*), образуемых во время открытия. Один *DCB* обслуживает максимум 16 индикаторов или 15 индикаторов и одно печатающее устройство, подключенных к одному и тому же устройству управления.

Таблица 4

Элементы блока управления данными (*DCB*)

Смещение от начала <i>DCB</i> (в байтах)	Поле <i>DCB</i>
+0-13	Резервируется
+14	<i>GTYPE</i> - указывает основные или специальные процедуры обработки внимания
+15-25	Резервируется

Продолжение табл. 4

Смещение от начала JCB (в байтах)	Поле DCS
+26-27	DSORG идентифицирует графический блок управления данными
+28-31	Резервируется
+32	GNCR - максимальное число макрокоманд ввода-вывода, выдаваемых перед макроКомандой WAIT
+33-35	POLST - адрес списка опроса, используется при специальной обработке внимания
+36	Резервируется
+37-39	EXLST - адрес списка пользовательских программ
+40-47	DDNAME - соответствует наименованию оператора DD
+48-49	Резервируется
+50-51	MACRF - типы макрокоманд ввода-вывода, используемых в проблемной программе

Формат макрокоманды DCS

Название	Операция	Операнд
[метка]	DOB	<p>DSORG = GS, MACRF =</p> <p>, GTYPE = [BASIC/EXPRESS]]</p> <p>, DDNAME = имя DD]</p> <p>[, EXLST = адрес-выходного- список</p> <p>[, GNCR = число-блоков]</p> <p>[, POLST = адрес-списка- опроса]</p>

н. пам.	чз 456676	Улаган
н. пам.	н. засум.	Подп. Дот.

Ц51.804.003 д56

Лист

15

DSORG = 65 – указывает, что определен графический блок управления данными. Этот операнд является обязательным.

MACRF – указывает тип или типы макрокоманд ввода-вывода, которые будут использоваться. *R* указывает *READ*, *W* указывает *WRITE* и *C* указывает *CTRL*. Ниже приводятся восемь допустимых кодов для операнда *MACRF*:

R
W
RC
WO
(R,W)
(RC,W)
(R,WO)
(RC,WC)

STYPE – указывает метод обработки внимания – базисный или специальный. Если этот операнд пропущен, то предполагается специальная обработка внимания.

DDNAME – указывает имя оператора *ДД*, описывающего набор данных, который подлежит обработке. *DDNAME* точно соответствует содержимому поля *майменование оператора ДД* из языка управления заданиями. Оно может иметь длину до 8 знаков, первый из которых должен быть буквой. *DDNAME* осуществляет логическую связь между оператором *ДД* и блоком управления данными проблемной программы пользователя.

EXLST – указывает адрес списка пользовательских программ, получающих управление от *OPEN*. Формат этого списка описан в [9.10]. Если *EXLST* не указан, то предполагается значение нуль.

GNCP – указывает максимальное число макрокоманд ввода-вывода, которые могут быть выданы перед макрокомандой *WAIT* (т.е. число блоков ввода-вывода, построенных во время открытия).

В исполнительное время значение *БНСР* должно быть от 1 до 99. Если значение *БНСР* не указывается, то предполагается значение 1. Программист может указать во время компиляции значение 0, но во время выполнения он должен вновь указать значение в пределах допустимого диапазона. Если во время открытия значение равно нуль, то происходит ненормальное окончание задачи.

POLST - адрес области основной памяти, зарезервированной пользователем, где с помощью макрокоманды *OPEN* составляется список адресов *DCB*. Этот список адресов *DCB*, называемый списком опроса, используется средствами специальной обработки внимания, макрокомандами *OPEN*, *CLOSE*, *GSERV* и *ANALYZ*.

Первая запись списка опроса содержит адрес последней записи (первоначально она должна быть установлена в нуль). Если открываются дополнительные устройства, то адреса *DCB* добавляются к списку опроса. Длина области, необходимой для построения списка, равна $I + n$ подных слов, где n - число устройств, для которых открыт *DCB*. Если потребитель желает открыть несколько *DCB*, используя одну и ту же макрокоманду *OPEN*, то он должен использовать один и тот же операнд *POLST*, с каждой макрокомандой *DCB*. Выдавая другие макрокоманды *OPEN*, он может использовать один и тот же или другие операнды *POLST*. При построении нескольких списков опроса для каждого из них требуется отдельная макрокоманда *ANALYZ*. Если *POLST* не указан, то предполагается значение нуль. Этот operand используется только при специальной обработке внимания.

Пользователь может построить *DCB* во время компиляции, используя для этой цели макрокоманду *DCB*. Информация в операторе определения данных (*ДД*) задается пользователем во время выполнения. Операнд *БНСР* может быть указан во время выполнения с помощью оператора *ДД* при условии, что во время компиляции *БНСР* устанавливается в нуль.

Изм.	Лист	№ докум.	Подп.	Дат.	Лист
					18

Используя рабочую область, программист может создать свой собственный *DCB* в исполнительное время. В этом случае он может указать в карте *ДД* операнды *DSORG* и *GMCR* при условии, что построенный *DCB* содержит ту же самую информацию в тех же самых ячейках, что и *DCB*, построенный с помощью макрорасширения *DCB* (см. табл. 4). Таким способом пользователь может эффективно строить *DCB* и писать реинтегрируемые программы.

ДД – имя в карте *ДД* должно быть идентично имени, указанному в этой рабочей области (в байтах от +40 до +70). Информация относительно кодирования карты *ДД* содержится в [8].

Блок управления данными, скомпилированный из макрокоманды *DCB*, не будет иметь наименований составляющих полей. Для указания и получения доступа к полям в блоке управления данными пользователь может писать макрокоманду *DCBL* следующим образом:

DCBL DSORG = 65

Поле наименования должно содержать пробел; указывается только операнд *DSORG*.

Макрокоманда *DCBL* образует фиктивную секцию *DSECT*, задающую символические наименования полей блока управления данными. Каждое поле определяется так, что при правильной инициализации базовых регистров пользователь может указать любое или все поля одного или нескольких блоков управления данными, пользуясь их символическими именами. Дополнительная информация относительно макрокоманды *DCBL* содержится в [9].

I.2.2. Макрокоманда открытия блока управления данными – *OPEN*

Эта макрокоманда для ЕС-7066 инициализирует один или несколько блоков управления данными, устанавливающих связь между проблемной программой и индикатором.

Изм	Лист	№ докум.	Подп.	Дат.

OPEN выполняет следующие функции:

- с помощью установки соответствующего бита в *DCB* указывает что индикатор "открыт";
- заполняет и инициализирует соответствующие поля в *DCB*;
- запрашивает основную память для инициализации цепочки *IOB* и соответствующих канальных программ;
- в *IOB* обеспечивается связь к списку *CCW* и следующему *IOB* в цепочке. Число построенных *IOB* на единицу больше числа, указанного операндом *GNCP*, если *GNCP* не равно 99. (Для каждого *DCB* может быть построено не более 99 *IOB*.) Так как *IOB* не строятся динамически, то для того, чтобы всегда гарантировать наличие *IOB*, строится дополнительный блок *IOB*. В *DCB* помещается указатель начала цепочки *IOB*:
 - загружает стандартную программу управления графическим вводом-выводом (*GIOCR*) и помещает ее адрес в *DCB*;
 - выполняет функции инициализации, требуемые средствами обработки внимания. Если в макрокоманде *DCB* был указан операнд *POLST*, то строится список адресов *DCB* (список опроса). Список опроса содержит одну запись для каждого индикатора, который будет опрашиваться пользователем.

Формат макрокоманды *OPEN*:

Название	Операция	Операнд
[метка]	<i>OPEN</i>	(адрес <i>dcb</i> [, адрес <i>dcb</i>]...) [<i>MF</i> = { <i>L</i> (<i>E</i> , имя- списка-параметров)} (<i>E</i> , (<i>I</i>))]]

адрес *dcб* - указывает адреса *DCB*, которые должны быть инициализированы для нужного графического устройства (устройств). Может быть указано любое число адресов. Это позволяет параллельно открывать *DCB* и связанные с ними наборы данных. Если указано несколько адресов *DCB*, то операнды "адрес *dcб*" должны отделяться двумя запятыми. Вторая запятая указывает пропуск необязательных operandов, не относящихся к средствам графического программирования.

MF=L - указывает, что расширение макрокоманды составляет только список параметров. Наименование списка параметров - это символьическое наименование, указанное в поле наименований этой макрокоманды *OPEN*.

MF = (E, имя-списка-параметров) - указывает, что макрокоманда должна выполнить функцию инициализации, используя список параметров, составленный другой макрокомандой *OPEN*. Функция инициализации выполняется для каждого операнда "адрес *dcб*" в списке. Операнд "имя-списка-параметров" должен точно соответствовать наименованию, указанному в поле наименований макрокоманды *OPEN*, используемой для построения списка параметров.

MF = (E, (I)) - выполняет ту же самую функцию, что и *(E, имя-списка-параметров)*. Если указывается регистр *(I)*, значит, программист должен загрузить адрес списка параметров в регистр *I* до выполнения макрокоманды.

Если operand указывает уже открытый *DCB*, то никакие действия не выполняются. Если operand указывает недействительный адрес *DCB* при открытии, то задание заканчивается ненормально. Если не соответствующего оператора *DD*, то открытие не вызывает указания ошибки; однако попытка использовать набор данных приведет к ненормальному окончанию задачи.

Во время выполнения стандартная программа *OPEN* проверяет опе-

Изм	Лис	№ до	у	Подп.	Дат
-----	-----	------	---	-------	-----

ранд *DNCR* макрокоманды *OPEN*. Если значение *DNCR* есть нуль или превышает 99, то задача заканчивается ненормально.

Если указанное в карте *ДЛ* устройство не является графическим, то соответствующий *DCB* является недействительным и задача заканчивается ненормально.

В примере I в операторах с именами *EXAMPLE2* и *EXAMPLE3 DCB1* и *DCB2* инициализируются с использованием списка параметров, составленного оператором с именем *EXAMPLE1*. В операторе *EXAMPLE3* адрес списка параметров загружен в регистр I.

Пример I.

<i>EXAMPLE1</i>	<i>OPEN (DCB1,,DCB2), MF=L</i>
<i>EXAMPLE2</i>	<i>OPEN MF=(E,EXAMPLE1)</i>
<i>EXAMPLE3</i>	<i>OPEN MF=(E,(1))</i>

I.2.3. Макрокоманда считывания знаковых данных — *GREAD*

Эта макрокоманда передает данные из буфера индикатора в основную память. Существует шесть возможных операций считывания, выполняемых макрокомандой *GREAD* в зависимости от указания операнда "тип".

Формат макрокоманды *GREAD*:

Название	Операция	Операнд
[метка]	<i>GREAD</i>	адрес <i>dесв</i> ,тип ,адрес <i>dесв</i> ,[длина] ,адрео-области [устройство] [,MF={L/E}]

адрес *dесв* —
ДЕСВ, о
 операнд
 тип — указыва
 правиль
 операнд
 операнд
DSM
DSM1
 клави
SM1
 индик
SM
 без
DS
D
 адрес *d*
 гра
 длина —
 адрес-с
 в
 устрои

адрес *dcсb* - имя, присваиваемое блоку управления событием данных *DECb*, образованному в макрорасширении, или его адрес. Этот операнд необходим для макрорасширения.

тип - указывает тип операции считывания. Этот операнд необходим для правильного макрорасширения. Ниже приведены допустимые формы операнда (каждый "тип" описан после рассмотрения всех других операндов, относящихся к макрокоманде *OLEAL*):

DSM - считать сообщение ручного ввода индикатора EC-7066;

DSMW - считать сообщение ручного ввода без восстановления клавиатуры индикатора;

SMI - неполная операция считывания сообщения ручного ввода индикатора;

SMIW - неполная операция считывания сообщения ручного ввода без восстановления клавиатуры индикатора EC-7066;

DSB - считать буфер индикатора;

DSEW - считать буфер без восстановления клавиатуры индикатора.

адрес *dcсb* - адрес блока управления данными *DCB*, связанного с графическим устройством.

длина - число байтов данных, которые должны быть считаны.

адрес-области - адрес ячейки основной памяти, в которую будет считываться первый байт данных.

устройство - десятичное целое число; используется для вычисления индекса нужного элемента в таблице *DEC* (указатели адреса для *UCB*, связанных с *DCB*). Указанный *UCB* содержит адрес нужного устройства. Значение "устройство" должно содержаться в диапазоне от 1 до *N*, где *N* - число устройств, указанных в карте определения данных для этого набора данных. *N* не может превышать 16 (максимальная конфигурация - это 15 индикаторов и одно печатающее устройство или 16 индикаторов для каждого устройства управления). При *N* < 1 и *N* > 16 устанавливается *N* = 1.

Изм	Лист	№ докум.	Подп.	Дата

Ц51.804.003 Д56

Лист
23

При использовании обозначения регистра содержимое регистра во время выполнения должно быть на единицу меньше величины операнда "устройство", описанного выше.

Пропуск операнда "устройство" означает задание значения для этого операнда, указывающего первый элемент в таблице ДЕСВ. Примечание. Устройство, указанное в операнде "устройство", не может быть печатающим устройством.

$MF=L$ - указывает, что расширение макрокоманды должно составлять только список параметров. Наименование, данное списку параметров, совпадает с наименованием, указанным операндом "адрес дескт" в L-форме этой макрокоманды.

$MF=E$ - указывает, что макрокоманда должна выполнить операцию считывания, используя список параметров, составленный другой макрокомандой $BRFAD$. Первый операнд в E-форме макрокоманды должен точно соответствовать наименованию списка параметров. Список параметров может быть изменен во время выполнения указанием новых значений любых операндов, предшествующих именевому операнду $MF = E$.

Операнд "тип", указанный в L-форме макрокоманды, может быть изменен с помощью E-формы соответствующей макрокоманды. Другие операнды, необходимые для выполнения и не указанные в L-форме макрокоманды, должны быть указаны в E-форме соответствующей макрокоманды.

Примечание. Каждому определенному списку параметров должно даваться уникальное наименование. Использование одного и того же наименования для нескольких блоков ДЕСВ приводит к ошибке многократного определения во время компиляции.

Если ключевые операнды $MF=L$ и $MF=E$ не указаны, то для получения макрорасширения должны быть закодированы все необходимые операнды. Пропуск операнда "адрес дескт" прекращает макрорасширение;

Лист № документа	Поле 1	Поле 2	Поле 3

пропуск других операндов вызывает сообщение об ошибке в листинге компилятора, но не заканчивает макрорасширение.

Существует шесть типов операций считывания EC-7066.

1. *DSM* - считать сообщение ручного ввода индикатора. Этот операнд "тип" вызывает передачу данных из буфера выбранного индикатора в основную память с помощью устройства управления. Данные, начиная с первой воспроизводимой позиции, следующей за стартовым символом, последовательно восстанавливаются и передаются до тех пор, пока не будет обнаружен символ "Конец сообщения" или счетчик байтов не станет равным нулю. Затем стартовый символ стирается. Данные, расположенные в одной строке и справа от символа новой строки, не передаются, символ новой строки передается. Если встречается сбойный символ, то он передается. Символ конца сообщения не передается в основную память, а воспроизведенные данные на ЭЛТ не изменяются и не разрушаются при этой операции чтения.

С помощью *GREAD* типа *DSM* образуется одно слово команды канала: *CCW* считывания сообщения ручного ввода индикатора, которое инициирует операцию считывания.

2. *DSMI* - считать сообщение ручного ввода без восстановления клавиатуры индикатора. Операция, вызываемая этим операндом "тип", идентична *DSM*, за исключением того, что клавиатура остается блокированной при завершении операции.

С помощью *GREAD* типа *DSMI* образуется одно слово команды канала: *CCW* считывания сообщения ручного ввода без восстановления клавиатуры индикатора, которое инициирует операцию считывания.

3. *SMI* - неполная операция считывания сообщения ручного ввода индикатора. Операция, вызываемая этим операндом "тип", идентична *DSM* во всем, за исключением того, что при обнаружении конца сообщения или при равенстве счетчика байтов канала нулю стартовый символ не стирается. Следовательно, время блокирования клавиатуры для операнда

"тип" *SMI* существенно меньше, чем для *DSM*. Это важно для применений, использующих высокоскоростной ввод данных и присоединение к одному *EC-7566* нескольких *EC-7066*.

С помощью *GREAD* типа *SMI* образуется одно слово команды канала: *CCW* неполной операции считывания сообщения ручного ввода индикатора. Это *CCW* инициирует операцию считывания.

4. *SMIW* - неполная операция считывания сообщения ручного ввода без восстановления клавиатуры индикатора. Операция, вызываемая этим операндом "тип", идентична *SMI* во всем, за исключением того, что при завершении операции клавиатура остается блокированной.

С помощью *GREAD* типа *SMIW* образуется одно слово команды канала: *CCW* неполной операции считывания сообщения ручного ввода без восстановления клавиатуры индикатора. Это *CCW* инициирует операцию считывания.

5. *DSB* - считать буфер индикатора. Этот операнд "тип" вызывает передачу максимум 240, 480 или 960 байтов данных, в зависимости от размера буфера индикатора. Первый передаваемый символ - это первый символ в левом верхнем углу ЭЛТ, а последний - последний символ в правом нижнем углу ЭЛТ. Если в воспроизводимой позиции не обнаруживается никакой знак, то передается пробел. Позиция курсора указывается расположением символа конца сообщения. Если в буфере встречаются служебные символы, то они передаются.

Ниже приведены коды служебных символов:

Бит	0	1	2	3	4	5	6	7
Стартовый символ	0	0	0	0	1	0	1	0
Конец сообщения	0	0	0	0	1	0	0	1
Новая строка	0	0	0	1	0	1	0	1
Сбойный символ	0	0	1	1	1	1	1	1

После операции ЭЛТ очищается и в первую позицию (левый верхний угол) на ЭЛТ помещается курсор.

С помощью *GREAD* типа *DSB* образуется несколько слов команды

канала. Первое, *CCW* считывания буфера индикатора, выдается при иницировании операции считывания. Второе, *CCW* стирания буфера индикатора, устраниет все данные из буфера индикатора и пересыпает курсор в первую воспроизводимую позицию ЭЛТ.

6. *L5BW* - считать буфер индикатора без восстановления клавиатуры. Операция, визуемая этим типом операций, идентична *DSB*, за исключением того, что клавиатура остается блокированной после завершения операции.

С помощью *GREAD* типа *L5BW* образуется несколько слов команды канала. Первое, *CCW* считывания буфера индикатора без восстановления клавиатуры, инициирует обмен данными. Второе, *CCW* стирания буфера индикатора без восстановления клавиатуры, устраниет все знаковые данные из буфера индикатора и устанавливает курсор в первую воспроизводимую позицию ЭЛТ.

Если счетчик байтов в *CCW* становится равным нулю, каналом образуется сигнал останова. Получение сигнала останова во время операции считывания визуирует окончание передачи данных от устройства управления в основную память. Устройство управления продолжает считывание данных из буфера индикатора, но не передает их в основную память. Обнаружение кода конца сообщения при использовании *L5M* заставляет устройство управления восстанавливать клавиатуру выбранного индикатора и стирать стартовый символ.

При использовании *SMI* клавиатура восстанавливается, а стартовый символ не стирается. При использовании *L5B* при окончании операции изображение стирается и курсор восстанавливается. Если при выполнении *DSB* обнаруживается ошибка (например, неправильная длина), то операция стирания не выполняется. Поэтому если программист требует выполнения операции стирания, то он должен гарантировать указание правильной длины.

Если operand "длина" не указан или равен нулю, то в образован-

Ізм	Лист		

№ докум. Подп. Дата

ном *CSW* счетчик байтов указывается равным 960. Это гарантирует отсутствие сигнала останова при операциях считывания. Если опорный "длина" указан неправильно, то в листинге компиллятора появится сообщение об ошибке.

Если при выполнении указание длины оказывается неправильным, то счетчик байтов из *CSW* помещается в соответствующую ячейку *DEC8* (с 17-го по 19-й байты), в 16-й байт *DEC8* помещается шестнадцатеричный код 43. Так как этот случай не считается ошибкой, то в старший байт *DEC8* помещается шестнадцатеричный код 7F. При постоянной ошибке считывания экран стирается и воспроизводится сообщение *ERROR ON READ*, код ошибки 41 помещается в левый байт (байт 0) *ECB* и шестнадцатеричный код 45 помещают в 16-й байт *DEC8* (*DEC8* + 16). Автоматический повтор операции не предусмотрен, поэтому ответственность за обработку ошибки полностью возложена на пользователя.

Перед обработкой данных, полученных при операции считывания, программа должна проверить левый байт *ECB* для определения условия завершения операции. Затем программа должна проверить байты с 16-го по 19-й *DEC8* для получения дополнительной информации об условии завершения. Если *DEC8* должен быть использован для нескольких операций ввода-вывода, то пользователь должен обнулить 16-й байт *DEC8* перед выполнением следующей операции ввода-вывода. В табл. 5 показаны индикация и условия завершения для используемых операций чтения.

После выполнения макрокоманды программа должна проверить шестнадцатеричный код возврата в регистре I5 для определения условия завершения. Коды возврата и их значения следующие:

00 - нормальное окончание;

10 - в *DEC8* отсутствует

"адрес *dcb*";

14 - "адрес *dcb*" в *DEC8*

недействителен;

Изм	Лист	№ докум.	Подп.	Дат.
-----	------	----------	-------	------

Лист

151.804.003 л56

1С - в DFCB отсутствует

код "тип";

24 - в DFCB отсутствует

"адрес-области".

Таблица 5

Индикация и условия завершения

Индикация	Условие	Операция считывания
ECB=7F DECB+I6=0 DECB+I7=не используется	Нормальное завершение	DSM, DSMW, SMI, SMIW, DSB, DSBW
ECB=41 DECB+I6=45 DECB+I7=не используется	Постоянная ошибка	DSM, DSMW, SMI, SMIW, DSB, DSBW
ECB=7F DECB+I6=43 DECB+I7=0 счетчик CSW =0	Запись большой длины: -счетчик CSW становится равным 0 до обнаружения символа конца сообщения; -счетчик CSW становится равным 0 до того, как считан полный буфер	DSM, DSMW, SMI или SMIW DSB или DSBW
ECB=7F DECB+I6=43 DECB+I7=счетчик счетчик CSW не равен начальному значению и не нуль	Запись короткой длины: -символ конца сообщения встретился до того, как счетчик CSW стал равным 0; -полный буфер был считан до того, как счетчик CSW достиг нуля	DSM, DSMW SMI или SMIW DSB или DSBW
ECB=7F DECB+I6=43 DECB+I7=пер- воначальному счетчику	Клавиша ВВОД была нажата при отсутствии стартового символа на экране или DSM была подана, когда клавиша ВВОД не была нажата	DSM, DSMW, SMI или SMIW

Лист

29

Ц51.804.003 Д56

В примере 2 *DECБ* с названием *DECБ1*, должен быть образован как часть макрорасширения. Операнд "тип" указывает, что операция для будет выполняться для набора данных, связанного с блоком управляния данными *INDCB*. В область основной памяти, названную *INAREA*, будут считано 100 байтов данных. Значение операнда "устройство" равно 1. В операторе *EXAMPLE1* указана *L*-форма, образующая список параметров. В операторе *EXAMPLE2* приведен код, в котором пропущены операнды в *L*-форме *GREAD*. Оператор *EXAMPLE3* дает *E*-форму *GREAD* с длиной, измененой до 150 байтов.

Пример 2.

<i>EXAMPLE1 GREAD</i>	<i>DECБ1,DSM,INDCB,100,</i> <i>INAREA,1,MF=L</i>
<i>EXAMPLE2 GREAD</i>	<i>DECБ1,DSM,MF=L</i>
<i>EXAMPLE3 GREAD</i>	<i>DECБ1,,,150,MF=E</i>

Программист может применить регистровую форму operandов. В этом случае адреса operandов должны загружаться в регистры (2-12) перед выполнением макрокоманды. Адрес *DECБ* может также загружаться в регистр I (см. пример 3).

Пример 3.

<i>EXAMPLE1 LA</i>	<i>4,DECБ</i>
<i>GREAD</i>	<i>(4),MF=E</i>
<i>EXAMPLE2 LA</i>	<i>1,DECБ1</i>
<i>LA</i>	<i>2,INDCB</i>
<i>LA</i>	<i>3,INAREA</i>
<i>GREAD</i>	<i>(1),,(2),,(3),MF=E</i>

Когда operand "устройство" выражается номером регистра, то значение, загруженное в регистр, должно совпадать с индексом *УСБ* ("устройство"-1) и помещаться в левый байт регистра.

I.2.4. Макрокоманда записи знаковых данных - *GWRITE*

Эта макрокоманда передает данные из основной памяти к буферу индикатора или к печатающему устройству. Существует восемь возможных операций записи, выполняемых макрокомандой *GWRITE* в зависимости от указания операнда "тип".

Формат макрокоманды *GWRITE*:

Название	Операция	Операнд
[метка]	<i>GWRITE</i>	адрес <i>dec\$</i> , тип , адрес <i>dec\$</i> , [длина] , адрес-области [, устройство] [, <i>MF</i> = {L E}]

адрес *dec\$* – имя, присваиваемое блоку управления событием данных (*DEC\$*), образованному в макрорасширении, или его адрес. Этот операнд необходим для макрорасширения.

тип – указывает тип операции записи. Этот операнд требуется для правильного макрорасширения. Допустимые формы операнда (каждый "тип" описывается после рассмотрения всех операндов, относящихся к макрокоманде *GWRITE*) следующие:

DSB – записать в буфер индикатора ЕС-7066 или записать в буфер печатающего устройства;

DSBW – записать в буфер индикатора ЕС-7066 без восстановления клавиатуры;

EBW – стереть и записать в буфер индикатора ЕС-7066;

-32-

EWW - стереть и записать в буфер индикатора EC-7066 без восстановления клавиатуры;
LNE - записать по адресу строки индикатора EC-7066;
LNEW - записать по адресу строки индикатора EC-7066 без восстановления клавиатуры;
EWL - стереть и записать по адресу строки индикатора EC-7066;
EWLW - стереть и записать по адресу строки индикатора EC-7066 без восстановления клавиатуры.
адрес *dcб* - адрес блока управления данными, связанного с графическим устройством.

длина - число байтов данных, которые должны быть записаны. Сюда включаются знак новой строки и байт управления адресом строки.

адрес-области - адрес ячейки основной памяти, из которой должен

записываться первый байт данных.

устройство - десятичное целое число; оно используется для определения индекса нужного элемента в таблице *DFB*, указывающей адреса *UCB*, связанных с *DSB*. Указанный *UCB* содержит адрес нужного устройства. Значение "устройство" должно быть в диапазоне от 1 до *N*, где *N* - число устройств, указанных в карте определения данных (*DD*) для этого набора данных. *N* не может превышать 16 (максимальная конфигурация содержит 15 индикаторов и одно печатающее устройство для каждого устройства управления или 16 индикаторов).

При использовании регистровой формы содержимое регистра во время выполнения должно быть на 1 меньше величины операнда "устройство", описанного выше.

Пропуск операнда "устройство" приводит к заданию значения 1 для этого операнда, что указывает первый элемент в таблице *DFB*.

Примечание. Печатающее устройство может указываться операндом "устройство", только когда операнд "тип" записан как *DSB*.

ЛЭМ	Лист №	докум.	Подп. дата

Лист
32

$MF = L$ – указывает, что расширение макрокоманды должно составлять только список параметров. Наименование, данное списку параметров, указывается операндом "адрес *decб*" в *L*-форме данной макрокоманды *GWRITE*.

$MF = E$ – указывает, что макрокоманда должна выполнить операцию записи, используя список параметров, составленный другой макрокомандой *GWRITE*. Первый операнд в *E*-форме макрокоманды должен точно соответствовать наименованию списка параметров. Список параметров может быть изменен во время выполнения с помощью указания новых значений любых operandов, предшествующих операнду $MF = E$.

Операнд "тип", указанный в *L*-форме макрокоманды, может быть изменен с помощью *E*-формы макрокоманды. Другие необходимые для выполнения operandы, не указанные в *L*-форме макрокоманды, должны быть указаны в *E*-форме соответствующей макрокоманды.

Примечание. Каждому определенному списку параметров должно быть дано уникальное наименование. Использование одного наименования *DSB* для нескольких *DSB* приводит к ошибке многократного определения во время компиляции.

Если ключевые operandы $MF = L$ или $MF = E$ не указаны, то для получения макрорасширения должны быть закодированы все необходимые operandы. Пропуск operand'a "адрес *decб*" прекращает макрорасширение; пропуск других operandов вызывает сообщение об ошибке в листинге компиляции, но не прерывает макрорасширение.

Существует восемь типов операций записи EC-7066.

I. *DSB* – запись в буфер индикатора. Этот operand "тип" вызывает передачу последовательных байтов знаковых данных к устройству управления в одном из двух режимов в зависимости от указания адреса индикатора или печатающего устройства.

При опознании управляемым устройством адреса индикатора данные

размещаются в буфере индикатора, начиная с позиции курсора. Курсор автоматически передвигается в следующую позицию. Если знак вводится в последнюю позицию ЭЛТ, то курсор помещается в первую воспроизводимую позицию ЭЛТ (левый верхний угол).

Если управляющее устройство опознает адрес печатающего устройства, то поступившие знаки размещаются в последовательные позиции буфера печатающего устройства и операция печати начинается.

Если при выполнении операции *GWRITE* типа *DSB* встречается ошибка записи, то перед началом повторного ввода процедура исправления ошибок системы полностью стирает изображение на экране.

При помощи *GWRITE* типа *DSBW* образуется одно слово команды канала: *CCW* записи в буфер индикатора, которое инициирует операцию записи.

2. *DSBW* - запись в буфер индикатора без восстановления клавиатуры. Операция, вызываемая этим операндом "тип", идентична *DSB* во всем, за исключением того, что при завершении операции клавиатура остается блокированной.

При помощи *GWRITE* типа *DSBW* образуется одно слово команды канала: *CCW* записи в буфер индикатора без восстановления клавиатуры, которое инициирует операцию записи.

3. *EW* - стереть и записать в буфер индикатора. Этот операнд "тип" вызывает стирание устройством управления всех данных из буфера индикатора и пересылку курсора в первую воспроизводимую позицию (левый верхний угол) ЭЛТ.

Затем данные помещаются в буфер индикатора, начиная с позиции, в которой был воспроизведен курсор.

При помощи *GWRITE* типа *EW* образуются два слова команды канала. Первое, *CCW* стирания буфера индикатора, выдается для стирания всех данных из буфера индикатора и пересылки курсора в первую позицию ЭЛТ. Второе, *CCW* записи в буфер индикатора, инициирует операцию записи.

- 35 -
4. *EBWW* - стереть и записать в буфер индикатора без восстановления клавиатуры. Операция, вызываемая этим операндом "тип", идентична *EW* во всем, за исключением того, что при завершении операции клавиатура остается блокированной.

При помощи *GWRITE* типа *EBWW* образуются два слова команды канала. Первое, *CCW* стирания буфера индикатора без восстановления клавиатуры, выдается для стирания всех данных из буфера индикатора и пересылки курсора в первую позицию ЭЛТ. Второе, *CCW* записи в буфер индикатора без восстановления клавиатуры, инициирует операцию записи.

5. *LNE* - запись по адресу строки индикатора. Этот операнд "тип" вызывает передачу устройством управления последующих байтов знаковых данных в буфер индикатора под управлением счетчика байтов канала. Начальное положение в буфере индикатора, указанное в первом байте данных, интерпретируется как байт управления. Этот байт указывает строку, в которую будут передаваться данные (см. табл. 6). Курсор пересыпается в первую позицию указанной строки, а данные размещаются в буфере индикатора, начиная с позиции курсора. Курсор автоматически продвигается к следующей позиции.

LNE, в сущности, выполняет ту же операцию, что и *DSB*. Основное различие состоит в том, что информация первого байта данных (шестнадцатеричное значение в диапазоне от *F0* до *FB*), задаваемая пользователем, указывает строку, на которую будут передаваться данные.

При помощи *GWRITE* типа *LNE* образуется одно слово команды канала: *CCW* записи по адресу строки индикатора, инициирующее операцию записи.

Изд.	Лн.	К. докум.	Подп.	Лист

Таблица 6

Формат первого байта данных для
операций с адресацией строки

Позиции битов 0 1 2 3 4 5 6 7	Шестнадцатеричный эквивалент	Выбранная строка
I I I I 0 0 0 0	#0	Первая
I I I I 0 0 0 1	#1	Вторая
I I I I 0 0 1 0	#2	Третья
I I I I 0 0 1 1	#3	Четвертая
I I I I 0 1 0 0	#4	Пятая
I I I I 0 1 0 1	#5	Шестая
I I I I 0 1 1 0	#6	Седьмая
I I I I 0 1 1 1	#7	Восьмая
I I I I 1 0 0 0	#8	Девятая
I I I I 1 0 0 1	#9	Десятая
I I I I 1 0 1 0	#A	Одинарная
I I I I 1 0 1 1	#B	Дважды

6. *LNEW* – записать по адресу строки индикатора без восстановления клавиатуры. Операция, вызываемая этим операндом "тип", идентична *LNE* во всем, за исключением того, что при завершении операции клавиатура остается блокированной.

При помощи *WRITE* типа *LNEW* образуется одно слово команды канала: *CCW* записи по адресу строки индикатора без восстановления клавиатуры, инициирующее операцию записи.

7. *EWL* – стереть и записать по адресу строки индикатора. Этот операнд "тип" вызывает стирание из буфера индикатора всех знаковых данных и перемещение курсора в первую позицию (левый верхний угол)

ЭЛТ. Затем курсор пересыпается в первую позицию указанной строки, а данные размещаются в буфере индикатора, начиная с позиции курсора. Курсор автоматически продвигается к следующей позиции.

EWL выполняет ту же операцию, что и *LNE*, за исключением того, что сначала производится операция стирания.

При помощи *GWRITE* типа *EWL* образуются два слова команды канала. Первое, *CCW* стирания буфера индикатора, устраниет все данные из буфера индикатора и восстанавливает курсор. Второе, *CCW* записи по адресу строки индикатора, инициирует операцию записи.

8. *EWLW* - стереть и записать по адресу строки индикатора без восстановления клавиатуры. Операция, вызываемая этим операндом "тип" идентична *EWL* во всем, за исключением того, что при завершении операции клавиатура остается блокированной.

При помощи *GWRITE* типа *EWLW* образуются два слова команды канала. Первое, *CCW* стирания буфера индикатора без восстановления клавиатуры, устраниет все данные из буфера индикатора и устанавливает курсор. Второе, *CCW* записи по адресу строки индикатора без восстановления клавиатуры, инициирует операцию записи.

Нормальное завершение указывается шестнадцатеричным кодом *7F* в левом байте *FCB* (байт 0). При постоянной ошибке в этом байте устанавливается шестнадцатеричный код *41*.

После выполнения макрокоманды программа должна проверить шестнадцатеричный код возврата в регистре I5 для определения условия завершения операции. Ниже приведены коды возврата и их значения:

00 - нормальное завершение;

10 - в *DFCB* отсутствует "адрес *dcb*";

14 - "адрес *dcb*" в *DFCB* недействителен;

1C - в *DFCB* отсутствует operand "тип";

24 - в *DFCB* отсутствует "адрес области".

Изм	Лист	N.	докум.	Подп.	Дата
-----	------	----	--------	-------	------

- 30 -

В примере 4 *DECБ*, названное *DECБ 1*, должно образовываться как часть макрорасширения. Операнд "тип" определяет операцию *DSB* для набора данных, связанного с блоком управления данными *OUTDCB*. Из области главной памяти, названной *OUTAREA*, должно быть считано 100 байтов. Величина операнда "устройство" равна 1. В операторе *EXAMPLE* указана *L*-форма для образования списка параметров. В операторе *EXAMPLE2* приведено кодирование для *L*-формы макрокоманды *GWRITE* при пропуске позиционных operandов, а в операторе *EXAMPLE3* кодирование для *E*-формы макрокоманды *GWRITE* с измененной длиной до 150 байтов.

Пример 4.

<i>EXAMPLE GWRITE</i>	<i>DECБ1,DSB,OUTDCB,100,</i>
	<i>OUTAREA,1,MF=L</i>
<i>EXAMPLE2 GWRITE</i>	<i>DECБ1,DSB,MF=L</i>
<i>EXAMPLE3 GWRITE</i>	<i>DECБ1,,,150,MF=E</i>

Программист может применять регистровую форму operandов. Перед выполнением макрокоманды адреса operandов должны загружаться в регистры (2+I2), а адрес *DECБ* может загружаться также в регистр I (см. пример 5).

Пример 5.

<i>EXAMPLE LA</i>	<i>1,DECБ1</i>
<i>GWRITE</i>	<i>(1),MF=E</i>
<i>EXAMPLE2 LA</i>	<i>1,DECБ1</i>
<i>LA</i>	<i>2,OUTDCB</i>
<i>LA</i>	<i>3,OUTAREA</i>
<i>GWRITE</i>	<i>(1),,(2),,(3),MF=E</i>

При указании номера регистра для operandца "устройство" значение, загруженное в регистр, должно совпадать с индексом *UCB* ("устройство"-I) и размещаться в старшем байте регистра.

1.2.5. Макрокоманда стирания буфера индикатора - *GCNTRL*

Эта макрокоманда стирает данные в буфере индикатора. Существуют две возможные операции стирания, выполняемые макрокомандой *GCNTRL*, в зависимости от указания операнда "тип".

Формат макрокоманды *GCNTRL*:

Название	Операция	Операнд
[метка]	<i>GCNTRL</i>	адрес <i>dcsv</i> , тип , адрес <i>dcv</i> [,устройство] [,MF{LIE}]

адрес *dcsv* - имя, присваиваемое блоку управления событием данных *DCS*, или его адрес. Этот operand необходим для макрорасширения. тип - указывает тип операции управления. Этот operand необходим для макрорасширения. Ниже приведены допустимые формы операнда (каждый operand "тип" рассматривается после описания всех operandов, относящихся к макрокоманде *GCNTRL*):

ERS - стереть буфер индикатора ЕС-7066;

ERSW - стереть буфер индикатора ЕС-7066 без восстановления клавиатуры.

адрес *dcv* - адрес блока управления данными (*DCS*), связанного с графическим устройством.

устройство - целое десятичное число; используется для создания номера нужного элемента в таблице *DEF*, указывающей адреса *UCS*, связанных с *DCS*. Указанный *UCS* содержит адрес нужного устройства. Значение "устройство" должно содержаться в диапазоне от 1 до *N*, где *N* - число устройств, указанных в карте определения данных (*ДД*) для этого набора данных. *N* не может превышать 16

1	Замі	15.4566761	Ном	7876
Лам	Лист	№ докум.	Полп.	Лата

Ц51.804.003 Д56

Лист
39

(максимальная конфигурация - 15 индикаторов и одно печатающее устройство или 16 индикаторов для каждого устройства управления).

Примечание. Устройство, указанное операндом "устройство", не может быть печатающим, так как операция стирания незаконна для печатающего устройства.

При пропуске операнда "устройство" этот operand получает значение I, указывающее первый элемент в таблице ДЕВ. Если operand определен неправильно, то в листинге Ассемблера печатается сообщение об ошибке.

$MF=L$ - указывает, что расширение макрокоманды состоит только из списка параметров. Наименование, данное списку параметров, совпадает с наименованием, указанным операндом "наименование *адрес*" L-формы данной макрокоманды *GCNTRL*.

$MF=E$ - указывает, что макрокоманда должна выполнить функцию стирания при помощи списка параметров, составленного другой макрокомандой *GCNTRL*. Первый operand в E-форме макрокоманды должен точно соответствовать наименованию списка параметров. Список параметров может быть изменен во время выполнения указанием новых значений любых operandов, предшествующих ключевому.

Операнд "тип", указанный в L-форме макрокоманды, может быть модифицирован с помощью E-формы соответствующей макрокоманды. Другие необходимые для выполнения operandы, не указанные в L-форме макрокоманды, должны быть указаны в E-форме соответствующей макрокоманды.

Примечание. Если ключевые operandы $MF=L$ или $MF=E$ не записаны, то для получения макрорасширения должны быть закодированы все необходимые operandы. Пропуск операнда "адрес *адрес*" прекращает макрорасширение; пропуск других operandов вызывает в листинге Ассемблера сообщение об ошибке, но не прекращает макрорасширение.

Изм	Лист	№ докум.	Подп.	Дата
-----	------	----------	-------	------

Существуют два типа операций управления для ЕС-7066.

1. *EFS* - стереть буфер индикатора. Этот operand "тип" заставляет устройство управления стереть все данные из буфера индикатора.

Курсор помещается в первую позицию ЭЛТ (левый верхний угол), а оставшееся изображение стирается.

При помощи *GCNTRL* типа *EFS* образуется одно слово команды канала: *CCW* управления стиранием буфера индикатора, инициирующее операцию стирания.

2. *ERSW* - стереть буфер индикатора без восстановления клавиатуры. Операция, вызываемая этим operandом "тип", идентична *EFS* во всем, за исключением того, что при завершении операции клавиатура остается блокированной.

При помощи *GCNTRL* типа *ERSW* образуется одно слово команды канала: *CCW* управления стиранием буфера индикатора без восстановления клавиатуры, инициирующее операцию стирания.

Нормальное завершение указывается шестнадцатеричным кодом *7F* в старшем байте *DEC8* (байт 0). В случае постоянной ошибки в этом байте устанавливается шестнадцатеричный код *41*.

После выполнения макрокоманды программа должна проверить шестнадцатеричный код возврата в регистре I5 для определения условий завершения. Ниже приводятся коды возврата и их значения:

00 - нормальное завершение;

10 - в *DEC8* отсутствует "адрес *dcb*";

14 - "адрес *dcb*" в *DEC8* недействителен;

1C - в *DEC8* отсутствует код "тип";

24 - в *DEC8* отсутствует "адрес области".

В примере 6 *DEC8*, названный *DEC81*, образуется как макрорасширение L-формы макрокоманды. Operand "тип" указывает, что для набора данных, связанного с блоком управления данными *OUTDCB*, будет выполнена операция стирания (*EFS*). Значение операнда "устройство"

Изм	Лист	№ докум.	Подп.	Дата

Ц51.804.003 Д56

Лист
41

равно 1. В операторе EXAMP1 указана L-форма, а в операторе EXAMP2 - L-форма макрокоманды GCNTRL с пропущенными позиционными operandами. В операторе EXAMP3 приведено кодирование для E-формы макрокоманды GCNTRL с измененным позиционным operandом.

Пример 6.

EXAMP1 GCNTRL	DEC B1,EBS,OUTDOB,1, MF=L
EXAMP2 GCNTRL	DEC B1,EBS, MF=L
EXAMP3 GCNTRL	DEC B1,,,4, MF=E

Программист может применить регистровую форму operandов. В этом случае перед выполнением макрокоманды он должен загрузить значения каждого operandа регистровой формы в один из регистров (2+12). Адрес DECB может также загружаться в регистр I (см. пример 7).

Пример 7.

EXAMP1 LA	1,DEC B1
GCNTRL	(1),MF=E
EXAMP2 LA	1,DEC B1
LA	2,3
SLL	2,24
GCNTRL	(1),,,(2),MF=E

Если для operand "устройство" указан номер регистра, то значение, загруженное в регистр, совпадает с индексом UCB ("устройство"-I) и размещается в старшем байте регистра.

I.2.6. Макрокоманда закрытия блока управления данными -

CLOSE

Эта макрокоманда "закрывает" один или несколько блоков управления данными, разрушая связь между проблемной программой и

индикатором (индикаторами). *CLOSE* выполняет следующие функции:

- указывает, что *DCB* для индикатора "закрывается" при помощи установления соответствующего бита в *DCB*;
- освобождает основную память, затребованную во время открытия, включая цепь *IOP* и соответствующие канальные программы, а также память, затребованную для функций обработки внимания;
- очищает поля *DCB*, инициализированные во время открытия;
- удаляет стандартные программы внимания, загруженные с помощью функций "открытия" (только при базисной обработке внимания);
- устраняет записи из списка адресов *DCB* (списка опроса), составленного во время открытия (только при специальной обработке внимания).

Формат макрокоманды *CLOSE*:

Название	Операция	Операнд
[метка]	<i>CLOSE</i>	(адрес <i>dcб</i> [, , адрес <i>dcб</i>]...) [MF = { L (E, имя-описка- параметров) (E, (I))}]

адрес *dcб* - указывает адреса *DCB* для графических устройств, которые должны быть закрыты. Может быть указано любое число адресов. Если указано несколько адресов, то операнды "адрес *dcб*" должны отделяться двумя запятыми. Вторая запятая указывает пропуск необязательных operandов, не имеющих отношения к графическому методу доступа.

MF = L - указывает, что расширение макрокоманды должно состоять только из списка параметров. Имя, данное списку параметров,

2. БАЗИСНЫЙ МЕТОД ОБРАБОТКИ СИГНАЛОВ ВНИМАНИЯ

Оператор индикатора ЕС-7066 может сообщить информацию системе при помощи клавиатуры. Нажатие специальной клавиши (ВВОД) клавиатуры вызывает сигнал внимания (незапланированное прерывание ввода-вывода). Для того, чтобы такое прерывание было осмысленным, должны существовать стандартные программы, получающие управление при появлении сигнала внимания.

При базисной обработке внимания управляющая программа автоматически обнаруживает внимание и передает управление соответствующей пользовательской программе обработки сигнала внимания. Пользователь должен передать управляющей программе информацию относительно того, какие имеются программы для обработки сигналов внимания и от каких индикаторов обрабатываются сигналы внимания.

Макрокоманды, описанние в длинном разделе, обеспечивают необходимую связь между пользователем и супервизором ввода-вывода. Это дает программисту гибкую систему управления общим механизмом обработки внимания. Макрокоманды и их функции перечислены в табл. 7.

Таблица 7
Макрокоманды базисной обработки внимания

Монемоника	Наименование макрокоманды	Функция
ВВОД	Определить условия асинхронного входа	Определяет программу обработки внимания и указывает точные условия, которые будут вызывать вход в эту программу
Лист	№ локум.	Полл.

151.804.003 Л56

Лист

46

Продолжение табл. 7

Мнемоника	Наименование макрокоманды	Функция
BPAR	Включить программу внимания	Делает пользовательскую программу обработки внимания, доступной для использования системой
DAR	Выключить программу внимания	Делает пользовательскую программу обработки внимания, не доступной для использования системой
ATTNINQ	Запрос внимания	Разрешает связь между программой внимания и управляющей программой

4.1. Функции базисного метода обработки сигналов внимания

Базисная обработка внимания обеспечивает средства для:

- автоматического обнаружения сигнала внимания;
- направления сигналов внимания к пользовательским программам обработки сигналов внимания;
- постановки последовательных сигналов в очередь до тех пор, пока соответствующие пользовательские программы не смогут их обработать;
- ожидания сигналов внимания, совмещенного с выборкой обработкой.

Главные преимущества базисной обработки внимания следующие:

- если во время получения сигнала соответствующая программа обрабатывает ранее обнаруженные сигналы, то вновь полученные сигналы не теряются;
- обработка может совмещаться с вводом информации и (или) ответом оператора;
- организация системы упрощается, так как устраивается необходимость в постоянной проверке наличия сигналов внимания.

При нажатии клавиши ВВОД на клавиатуре ЕС-7066 управляющая программа прерывает текущую обрабатывающую программу (которая может быть и пользовательской программой обработки внимания) и определяет, какая пользовательская программа была указана для обработки внимания о этого устройства. Если пользовательская программа внимания не была указана, то данные внимания теряются и управление возвращается к прерванной программе. Если пользовательская программа внимания доступна, то данные внимания помещаются в область (СOMAREA), определенную пользователем, и управление передается к программе внимания. Для того чтобы программа внимания была доступна при получении сигнала внимания, она должна быть указана (определенна) макрокомандой ЗАКС, а соответствующий ГАСВ должен быть включен макрокомандой ЗРАД. Если программа, необходимая для обработки внимания, занята, то данные внимания сохраняются до тех пор, пока не закончится текущая обработка (обработка предыдущего сигнала внимания).

Примечание. Получение внимания ЕС-7066 вызывает блокировку клавиатуры ЕС-7066 до тех пор, пока сигнал внимания не будет обработан программой внимания. Для разблокировки клавиатуры должна использоваться макрокоманда СРЕАЛ или ГМРНГЕ.

Изм.	Лист	№ докум.	Подп.	Лот
------	------	----------	-------	-----

2.2. Макрокоманды базисного метода обработки сигналов внимания

Макрокоманды этой группы позволяют пользователю определить для системы свои возможности в обработке внимания. Пользователь может предназначить одну программу обработки внимания для обслуживания сигналов от всех графических устройств, занятых в задаче, либо он может определить для отдельных устройств отдельные программы. Макрокоманды **SAEC**, **SPAR**, **LAR** используются для определения указания состояния программ обработки внимания. Макрокоманда **ATTNIND** используется для выбора сигнала внимания от определенного индикатора.

2.2.1. Макрокоманда определения условий асинхронного входа - **SAEC**

Эта макрокоманда определяет программу обработки внимания при помощи создания списка параметров, называемого графическим управляющим блоком-внимания **GACB**. **GACB** содержит информацию, которая идентифицирует программу обработки внимания, связывает ее с ЕС-7066, указывает точку входа программы и тип или типы сигналов, вызывающих вход в эту программу. Впоследствии **GACB** может указываться символическим наименованием макрокоманды **SAEC**.

Формат макрокоманды **SAEC**:

Название	Операция	Операнд
[метка]	SAEC	[KP={адрес-точки-входа 0}] [, DCB=адрес-dcb] [, COMAREA=адрес-области-связи] [, ATTNTYPE={({A S B},7066) SAVE RESTORE}] [MF={L1(Е,адрес-gacb)}]

Л.рнг	Ц51.4566/76	Икн	17870
Изм	Лист	№ докум.	Подп.

Ц51.804.003 Д56

Лист
49

EP - определяет точку входа программы внимания. Если этот операнд опущен или указан равным нулю (что эквивалентно), то любой сигнал для этой программы будет помещаться в очередь до тех пор, пока не будет указана точка входа или не проверена очередь при помощи макрокоманды *ATTNINQ*. Последующие сигналы внимания для этой программы, полученные с других устройств EC-7066, не обрабатываются до тех пор, пока не будет обработан первый сигнал.

Примечание. Управляющее устройство EC-7566 блокирует клавиатуру EC-7066 до тех пор, пока ожидающий сигнал внимания от этого устройства не будет обработан.

DCB - указывает адрес *DCB*, связанного с индикатором или группой индикаторов, сигналы внимания от которых обслуживаются.

COMAREA - указывает адрес области, определенной программистом в основной памяти, помещенной на границу полного слова и состоящей по крайней мере из 16 байтов. Эта область используется управляющей программой для передачи информации внимания пользовательской программе внимания.

ATTNTYP - указывает тип внимания, который будет вызывать вход в программу внимания.

A, S или *R* - указывает соответственно, будет ли последующая информация прибавляться, вычитаться или замещать существующее поле *ATTNTYP* в *DCB*. Если в *E*-форме макрокоманды этот operand пропущен, то принимается *R*.

7066 - указывает, что программа внимания должна обслуживать сигналы внимания, вызванные EC-7066.

SAVE, RESTORE - указывают соответственно, что содержимое поля *ATTNTYP* в *DCB* должно запоминаться либо замещаться маской, сохраненной предыдущей макрокомандой *SAC* с операндом *ATTNTYP = SAVE*.

Примечание. Коды *A*, *S*, *R*, *SAVE* и *RESTORE* предусмотрены для совместимости с другими графическими устройствами, допускающими несколько типов сигналов внимания.

MF-L - указывает, что макрокоманда должна составлять *GACB*, используя указанные операнды. Пропуск operandов в *L*-форме вызывает установку соответствующих полей в нули. Если не указаны *L*, ни *E*-формы, то принимается *L*. При использовании *L*-формы этой макрокоманды *A*, *S*, *R*, *SAVE* и *RESTORE* не указываются. Пропуск operandов *A*, *S* или *R* может не отмечаться залятыванием.

MF-E - вызывает модификацию *GACB*, построенного *L*-формой этой макрокоманды, с помощью других указанных operandов. Если некоторый operand не указан, то изменения соответствующего поля в *GACB* не происходят. Пропуск operandов *A*, *S* или *R* должен быть указан залятой.

Примечание. Значения, указанные для operandов *COMAREA*, *DCB*, *ATTNTYP* и *EP* макрокоманды *SAECS*, включаются соответственно в первое, второе, четвертое и пятое слова в *GACB*. Все используемые величины должны быть определены и *DCB* должен быть открыт до того, как выдается макрокоманда *SPALX*, указывающая этот *GACB*.

Длина построенного с помощью макрокоманды *SAECS GACB* равна 14 словам. Первые пять слов *GACB* имеют следующий формат:

<i>GACB</i>	+ 0	адрес <i>COMAREA</i>
	+ 4	адрес <i>DCB</i>
	+ 8	резервируется
	+ 12	маска <i>ATTNTYP</i>
	+ 16	адрес <i>EP</i>

Установив поле *EP* в *GACB* в нуль, программист может поместить сигналы внимания в очередь и обработать их во вспомогательной (фоновой) программе, а не в программе обработки внимания. Нужные сигналы могут быть выбраны из очереди с помощью макрокоманды *ATTNLNQ*.

Изм	Лист	№ докум.	Подп.	Дата	Лист	51
Ц51.804.003 Д56						

Поле *БР* может быть установлено в нуль во время компиляции (макрокомандой *ЗАЕС*) или в любой момент во время выполнения. Если оно установлено в нуль во время выполнения, то любые полученные до этого сигналы внимания будут обрабатываться как описано выше; любые сигналы, полученные после установки *БР* в нуль, помещаются в очередь.

Первые четыре слова в *СОМАРЕА* имеют следующий формат:

<i>СОМАРЕА</i>	+ 0	ИУ	РЕЗЕРВИРУЕТСЯ	ТИП
	+ 4		РЕЗЕРВИРУЕТСЯ	
	+ 8		РЕЗЕРВИРУЕТСЯ	
	+ 12		РЕЗЕРВИРУЕТСЯ	
			Область, определяемая пользователем	

РЕЗЕРВИРУЕТСЯ - эти поля используются управляющей программой.

ИУ - еслициальному *ДСВ* соответствует несколько ЕС-7066, то в этот байт устанавливается целое число от I до 16 для указания устройства, сигналы которого представлены в данный момент для обработки программой внимания. Значение ИУ соответствует позиции устройства в списке устройств, заданном картой *ДД* и помещенном в *ДСВ* при помощи макрокоманды *ОРЕК* (ИУ функционирует так же, как операнд "устройство" в макрокомандах *СРЕД*, *СУАП* и *ОСНTEL*). Еслициальному *ДСВ* соответствует только один ЕС-7066, то байт устанавливается в нуль (шестнадцатеричное 00).

ТИП - это всегда шестнадцатеричное 07, указывающее сигнал внимания от ЕС-7066.

Пользователь может добавить к *СОМАРЕА* другие поля (см. подраздел 3.3).

Если программа внимания должна обслуживать несколько устройств, указанных отдельными макрокомандами *ЗАЕС*, то программа должна быть реентерабельной (параллельно используемой). Адрес *ДСВ* в *САСВ*

не может быть изменен без предварительного выключения программы внимания при помощи макрокоманды **DAR**. После замены адреса **DCB** программа внимания может быть включена при помощи макрокоманды **SPAR**. Например:

```
GACBX1 SAEC EP=AR1,DCB=GRAPHD1,  
ATTNTYP=7066,  
COMAREA=AREA1
```

Созданный этой макрокомандой **GACB** имеет имя **GACBX1**. **GACB** определяет программу внимания, входная точка которой есть **AR1**. Эта программа может обслуживать сигналы внимания от индикаторов (индикаторов) ЕС-7066, связанных с указанным **DCB**, и будет получать информацию о сигнале внимания в **COMAREA**, идентифицированной символом **AREA1**. Так как макроформа (**MF**) не указана, то предполагается *L*-форма.

```
EX2 SAEC DCB=GRPMDCB2,EP=(3),  
MF=(E,GACBX1)
```

Эта макрокоманда изменяет адрес **DCB** и точку входа программы внимания в **GACB**, определенном в операторе **GACBX1**.

Примечание. Если эта программа была включена ранее (макрокомандой **SPAR**), то перед изменением адреса **DCB** она должна быть выключена (с помощью макрокоманды **DAR**).

После выдачи макрокоманды **SAEC** для изменения адреса **DCB** программа должна быть вновь включена макрокомандой **SPAR**.

2.2.2. Макрокоманда включения программы внимания - **SPAR**

Эта макрокоманда включает для использования системой ранее определенные пользовательские программы обработки внимания при помощи установления ссылок управляющей программы к **GACB**. В случае включения нескольких программ внимания она также задает приоритет.

Изм	Лист	№ докум.	Подп.	Дата

Формат макрокоманды **SPAR**:

Название	Операция	Операнд
[метка]	SPAR	(адрес-гасб [, адрес-гасб] ...) [, PRTY = целое число] [, MF = {L(Е, адрес-списка- гасб)}]

адрес-гасб - указывает адрес графического управляющего блока внимания (**GACB**) (определенного макрокомандой **SAEC**), который нужно включить в систему.

PRTY - десятичное целое число в диапазоне от 0 до 127, используемое для определения приоритета среди программ обработки внимания. Приоритет возрастет с численным значением (127 задает наивысший приоритет). Никакая программа внимания не может получить управление до тех пор, пока не закончит работу программа наивысшего приоритета для этого же устройства.

MF = L - определяет создание списка адресов **GACB**, который может быть использован Е-формой макрокоманды **SPAR** или **DAR**.

MF = E - определяет, что список **GACB**, указанный операндом "адрес-списка-гасб", должен быть модифицирован, как указано операциями "адрес-гасб", и что **GACB** из списка затем должны быть включены, т.е. предоставлены для использования системой. Список не может удлиняться или сокращаться с помощью Е-формы этой макрокоманды.

адрес-списка-гасб - указывает адрес списка **GACB**, установленный L-формой макрокоманды **SPAR**.

При программировании надо учитывать, что операнд PRTY главным образом предназначен для совместимости с другими графическими устройствами, имеющими несколько типов сигналов внимания. Для таких уст-

Лист	155	1566/161	Мод	11.11
Лист	№ докум.	Подп.	Даты	

Ц51.804.003 д56

Лист
54

ройств операнд *PRTY* имеет возможность программы, запланированной для обработки сигналов определенного типа, прерывать другую, менее важную программу, обрабатывающую сигналы другого типа от того же самого устройства.

Для этих целей программист определяет относительную "важность" программ при помощи операнда *PRTY*. Его использование для ЕС-7066, имеющего сигналы внимания единственного типа, не имеет смысла. Поэтому обычно операнд *PRTY* опускается.

При пропуске операнда *PRTY* предполагается нулевой приоритет.

Если для некоторого ЕС-7066 в данный момент представлено несколько программ внимания с одним и тем же числовым приоритетом (как, например, при пропуске операнда *PRTY*), то считается, что наивысший приоритет имеет программа, включаемая последней в последней макрокоманде *SPAR*.

Все программы обработки внимания более приоритетны, чем любая вспомогательная фоновая программа.

Если одна и та же программа определена в отдельных макрокомандах *SPAR* для нескольких индикаторов или если программа используется несколькими задачами, то программа внимания может быть прервана и может непосредственно получить управление вновь. Такая программа должна быть реентерабельной (т.е. параллельно используемой).

Если один и тот же *GACB* указывается в макрокоманде *SPAR* несколько раз, то все ссылки после первой рассматриваются как отсутствие операции. Указавшие 0 в качестве адреса *GACB* в списке адресов *GACB* также считается отсутствием операции.

Обнаруженные ошибки оканчивают обработку списка *GACB* и вызывают немедленный возврат. При возврате к программе регистр I будет содержать адрес *GACB*, связанного с ошибкой, а регистр I5 - один из кодов возврата, указывающих тип ошибки. Ниже приведены коды возврата, указывающие тип ошибки:

Изм	Лист	№ докум.	Подп.	Дата	Лист	55
					1151.804,003 Л56	

- 04 - *GACB* содержит адрес недействительного *DCB*;
- 08 - *GACB* уже включен в предыдущей макрокоманде *SPAR*;
- OC - неудовлетворен запрос основной памяти, необходимой для ссылок управляющей программы;
- 14 - (только МУТ) макрокоманда *SPAR* выдана задачей, отличной от открывшей данный *DCB*, или одной из ее подзадач.
- В примере 8 приведено использование макрокоманды *SPAR*.

Пример 8.

EX1 SPAR (GACBX2)

EX2 SPAR (GACBX1, NAME3, TAG6), MF=L

EX3 SPAR (,,GAOBX3), MF=(E, EX2)

Макрокоманда *EX1* включает программу, определяемую *GACB* с именем *GACBX2*. Макрокоманда *EX2* конструирует список, содержащий адреса трех указанных *GACB*. Макрокоманда *EX3* вызывает модификацию списка, составленного в *EX2*, заменяя третий адрес *GACB* (*ГЛ06*) на *GACBX3*. Затем устанавливаются управляющие ссылки к трем *GACB*, указанным в списке.

2.2.3. Макрокоманда выключения программы внимания - *DAR*

Эта макрокоманда противоположна макрокоманде *SPAR*. Она делает программы обработки внимания недоступными для использования системой, устранив ссылки управляющей программы к *GACB*, установленные *SPAR*.

Формат макрокоманды *DAR*:

Название	Операции	Операнд
[метка]	DAR	(адрес-гасб [, адрес-гасб] ...) [, MF={L (E, адрес-списка-гасб)}]

2	запл	15.4566/76%	ГЛ06	17.87
П	нр	покум.	Попл.	Плат.

Ц51.804.003 д56

Лист 55

адрес-дасб - указывает адрес **GACB**, определяя выключение данной программы внимания.

MF-L - указывает формирование списка адресов **GACB**, записанных в макрокоманде.

MF-G - указывает, что список **GACB**, указанный операндом "адрес-списка-дасб", должен быть модифицирован операндами "адрес-дасб" и что все указанные в списке программы внимания должны быть затем выключены, т.е. сделаны недоступными для использования системой.

адрес-списка-дасб - указывает адрес списка адресов **GACB**, составленный **L**-формой макрокоманды **SPAR** или **DAE**.

При программировании надо учитывать, что функция **CLOSE** будет выключать все определенные для соответствующего **DCB** программы внимания. Однако попытка выдать макрокоманду **CLOSE** из программы внимания приведет к ненормальному окончанию задачи. Аналогично этому попытка устранить активную программу внимания приведет к ошибке.

Обнаруженные ошибки остановят обработку списка **GACB** (или одного **GACB**) и вызовут немедленный возврат. При возврате к программе регистр I будет содержать адрес связанныго с ошибкой **GACB**, а регистр 15 - код возврата, идентифицирующий условие ошибки следующим образом:

04 - **GACB** содержит адрес недействительного **DCB**;

08 - программа не найдена (не включена макрокомандой **SPAR**);

0C - программа внимания активна;

I4 - (только для **MVT**) макрокоманда **DAE** выдается не той задачей, которая выдала **SPAR** для этого **GACB**.

Бум	Ли	№ докум.	Подп.	Лат

Ц51.804.003 Д56

Лист
57

2.2.4. Макрокоманда запроса внимания – **ATTNINQ**

Эта макрокоманда устанавливает связь между программой обработки внимания и управляющей программой, позволяя программе обработки внимания выбирать сигнал внимания от определенного индикатора или от любого индикатора. Это позволяет программе обработки внимания ожидать прихода сигнала внимания, отдавать управление фоновой программе до прихода сигнала внимания, ветвиться по указанному адресу, если ожидается определенная информация внимания, или "чистить" очередь сигналов внимания к данной программе обработки внимания. В случае **CLEAR** сигналы внимания гасятся и индикатор ЕС-7066 должен быть разблокирован вручную.

Формат макрокоманды **ATTNINQ**:

Название	Операция	Операнд
[метка]	ATTNINQ	адрес-засб, [, MODE={R W (с, адрес перехода) CLEAR} [, UNIT={целое число LLL}]] [, MP={L (E, адрес-списка- параметров)}]

адрес-засб – указывает адрес соответствующего **САСВ**.

MODE – указывает один из четырех режимов запроса следующим образом:

R – режим "отдать управление". Программа внимания отдает фоновой программе управление до тех пор, пока информация внимания от любого индикатора, обслуживаемого данной программой внимания, не будет представлена. После того как эта информация представлена, обработка продолжается с команды, следующей за макрокомандой. Если макрокоманда **ATTNINQ** выдается программой, получившей управление при помощи макрокоманды **LINK**, или если поле **EP** в соответствующем **САСВ** установлено в

Л. зм!	Ц5-4566/81	ЗКа17787
Лист	№ документа	Номер листа

нуль, то этот режим рассматривается системой как режим ожидания (*W*).

W - режим ЖДАТЬ. Задача должна быть помещена в состояние ожидания до тех пор, пока для нее не будет представлена информация от указанного операндом *UNIT* индикатора. Пока программа внимания находится в состоянии ожидания, сигналы внимания от других индикаторов не могут обрабатываться другими программами внимания. При удовлетворении условия ожидания и после возвращения управления к программе внимания обработка продолжается со следующей команды.

C, адрес-перехода - условный режим. Если сигналы от указанного операндом *UNIT* устройства в настоящий момент помещены в очередь для этой программы, то выполняется переход по адресу, указанному операндом "адрес перехода"; в противном случае продолжается обработка со следующей команды.

СБЕЛЛ - режим гашения. Все сигналы, помещенные в очередь для этой программы, должны быть удалены. Обработка продолжается со следующей команды. Клавиатура индикатора ЕС-7066 должна быть разблокирована вручную, если сигнал внимания был погашен макрокомандой *ATTNINQ*.

UNIT - целое десятичное число в диапазоне от 1 до 16, соответствующее положению устройства в списке устройств, полученному с помощью операнда *ЛСВ*, где одному *ЛСВ* соответствует несколько устройств. Это число используется в качестве индекса для выбора одного из нескольких адресов *ЛСВ* из списка в блоке расширения данных (*ЛРВ*). Значение индекса, соответствующее печатающему устройству, не должно указываться (этот operand функционирует так же, как operand "устройство" в макрокомандах *СКЛАД* и *GWRITE*).

UNIT=ALL - указывает, что принимаются сигналы от любого ЕС-7066, соответствующего данному *ЛСВ*. При пропуске операнда *UNIT* предполагается *ALL*:

Изм	Лист	№ докум.	Полл.	Дата	Лист	59
2	4М7	15-4568763	8Kb	1/78.70	Ц51.804.003 Д56	

MF-L - указывает, что расширение макрокоманды должно состоять только из списка параметров. Символ, записанный в поле наименования, присваивается списку параметров.

MF-L - вызывает модификацию списка параметров, построенного *L*-формой макрокоманды и идентифицированного операндом "адрес-списка-параметров", также вызывает выполнение запроса на внимание.

адрес-списка-параметров - адрес списка параметров, построенного *L*-формой макрокоманды **ATTNINQ**.

При программировании следует учитывать, что макрокоманда **ATTNINQ** может выдаваться из программы внимания или из вспомогательной (фоновой) программы, если поле **EP** соответствующего **CACB** установлено в нуль. Должны соблюдаться стандартные соглашения о связях (см. [6]).

Если для соответствующего **CACB** имеется несколько устройств, то операнд **UNIT** должен использоваться вместе с режимом **W** или **C**; его использование с режимами **CLEAR** или **R** бессмысленно.

При режиме **R**, **W** или **C** информация, касающаяся сигнала внимания, удовлетворяющего запрос, находится в **COMAREA**.

Если во время выполнения макрокоманды **ATTNINQ** обнаруживаются ошибки, то происходит немедленный возврат к программе. После возврата регистр 15 содержит шестнадцатеричный код, указывающий на результат работы макрокоманды. Ниже приведены коды возврата и их значения:

00 - выполнение закончено успешно;

10 - поле адреса **CACB** в списке параметров, созданном макрокомандой **ATTNINQ**, содержит нули;

14 - макрокоманда **ATTNINQ** (режим **R**) уже была выдана для этого **CACB** из другой программы;

18 - **CACB** не был инициализирован макрокомандой **SPAR**;

Нзм	Лист	№ докум.	Подп.	Пот.

1С - инициализация GACB. макрокомандой **БРАР** закончилась неверно;

20 - поле режима в списке параметров, созданном **ATTNINQ**, содержит неправильную информацию;

24 - внутренняя ошибка. Код возврата программы исключения из очереди указывает на ошибку в ее работе;

28 - указан режим **W**, однако программа внимания уже находится в состоянии ожидания;

2C - поле адреса перехода в списке параметров, созданном **ATTNINQ**, содержит нули (выдается только в режиме С).

В примере 9 приведено использование макрокоманды **ATTNINQ**.

Пример 9.

```
EX1 ATTNINQ GACBX4, MODE=R  
EX2 ATTNINQ GACBX4, MODE=W, UNIT=5  
EX3 ATTNINQ GACBX4, MODE=(C, BRANCHA),  
    UNIT=5  
EX4 ATTNINQ GACBX4, MODE=CLEAR
```

Макрокоманда **EX1** заставляет программу отдать управление до появления сигнала внимания. Макрокоманда **EX2** помещает программу вниманий в состояние ожидания (**WAIT**) до тех пор, пока не появится информация внимания от индикатора, занимающего в списке устройств позицию 5. Макрокоманда **EX3** вызывает переход по адресу **BRANCHA**, если в очереди есть сигналы от индикатора, занимающего в списке устройств позицию 5. Макрокоманда **EX4** вызывает устранение из очереди всех сигналов внимания.

2.3. Пользовательские программы обработки внимания

Для написания программы связи оператор-машина пользователю могут быть полезны указания, описанные в п. 2.3.1 - 2.3.3.

2.3.1. Включение пользовательской программы внимания

Программа внимания определяется с помощью макрокоманды **ЗАЕС**.

Лист
Ц51.804.003 Д56
61
Изм. Лист № докум. Подп. Дат

которая создает блок управления графическим вниманием (*GACB*), содержащий заданную этой макрокомандой информацию.

В *GACB* содержится:

- адрес точки входа соответствующей программы внимания;
- адрес *DCB*, связанного с обслуживаемым устройством;
- адрес области связи *COMAREA*, которая используется для передачи информации внимания к программе внимания.

Таким образом *GACB* обеспечивает связь между индикатором или группой индикаторов и программой внимания.

Даже будучи определенной программа еще не является предоставленной для использования до тех пор, пока она не "включена" макрокомандой *SPAR*. Этим включением устанавливаются ссылки управляющей программы к ней.

Макрокоманда *DAR* может использоваться для выключения программы внимания, которую далее не желательно оставлять активной, хотя соответствующий *DCB* не закрывается. Макрокоманды *SPAR* и *DAR* могут рассматриваться как аналоги *OPEN* и *CLOSE*.

2.3.2. Обработка внимания

При входе в программу внимания регистр I содержит адрес *GACB*. В первом слове *GACB* содержится адрес области связи или *COMAREA*. Если одному *DCB* соответствует несколько устройств, то первый байт *COMAREA* указывает устройство, вызвавшее сигнал.

Использование области связи позволяет программисту передавать информацию между вспомогательной (фоновой) программой и программами обработки внимания. Каждому индикатору может соответствовать его собственная область связи, и сигналы от каждого индикатора могут записываться независимо от сигналов других индикаторов. Одним из полей области связи может быть *ESC*, где отмечается завершение обработки внимания. При завершении обработки последовательности сигналов внимания для данного индикатора программа внимания может отметить соответствующий *ESC*.

Лн	№ докум.	Подп.	Лот
151.804.003	Д56		

Используя информацию из *COMPARE* и *DCB*, программа внимания может выполнить необходимые вычисления, выдать соответствующие команды ввода-вывода и сделать все необходимое для ответа на сигналы. Если и в дальнейшем требуются сообщения от оператора индикатора, то для того, чтобы возвратить управление и ввести состояние ожидания до получения определенного сигнала или выполнения условного перехода при его наличии, может быть использована макрокоманда *ATTNINQ*.

При завершении обработки последнего сигнала из последовательности сообщений программа внимания может "почистить" очередь сигналов внимания с помощью макрокоманды *ATTNINQ* (режим *CLEAR*).

Вспомогательная (фоновая) программа выполняет следующие функции:

- открывает *DCB* для индикатора или индикаторов;
- определяет пользовательскую программу внимания при помощи макрокоманд *BLFC* и *SPAR*;
- выдает команды ввода-вывода;
- ждет указания завершения диалога индикатор-программа в блоке управления событием (*ESB*) или выполняет вспомогательную обработку, которая может и не иметь отношения к программе обработки внимания;
- закрывает *DCB*;
- завершает задачу выдачей макрокоманды *RETURN*.

Пользовательская программа внимания выполняет следующие функции:

- выполняет стандартные начальные функции (сохранение регистров, заведение базового регистра и т.д.);
- выполняет операции, основанные на информации внимания;
- выдает команды ввода-вывода;
- ожидает следующего сигнала при помощи макрокоманды *ATTNINQ*;
- после обслуживания последнего сигнала внимания указывает завершение в *ESB* и возвращает управление вспомогательной программе с помощью макрокоманд *POST* и *RETURN*.

2.3.3. Синхронизация пользовательских программ

Синхронизацию вспомогательной (фоновой) программы и программы внимания выполняет программист. Он может использовать макрокоманду **WAIT** для задержки выполнения вспомогательной программы до тех пор, пока программа внимания не отметит завершение в **ESB**. Программа внимания может отметить **ESB** с помощью макрокоманды **POST**, отыскав соответствующий **ESB** через графический управляющий блок внимания. Программа внимания передает управление назад вспомогательной программе с помощью макрокоманды **RETURN**. Когда будет отмечено завершение, вспомогательная программа выводится из состояния ожидания и продолжает свою работу. Этот цикл, т.е. передача управления программе внимания при появлении сигнала и возврат управления прерванной (вспомогательной) программе при указании завершения, продолжается до тех пор, пока определена программа внимания и появляются сигналы внимания.

2.4. Использование средств базисной обработки внимания при **MVT**

Для того чтобы пользовательские программы внимания обрабатывали сигналы внимания соответствующего устройства, они должны быть определены (макрокомандой **SAEC**) и включены (макрокомандой **SPAR**) внутри открывшей это устройство задачи или одной из ее подзадач. Макрокоманда **SPAR**, выданная несоответствующей задачей, вызовет возврат с кодом возврата - **X'14!**.

Каждый **GACB** должен выключаться (с помощью **LAR**) задачей, в которой он был включен (**SPAR**). Макрокоманда **LAR** из другой задачи будет вызывать возврат с кодом возврата - **X'14'**. Попытка одной задачи закрыть **ESB**, для которого другой задачей включен еще один или несколько **GACB**, приведет к ненормальному окончанию указанной задачи.

Макрокоманда **ATTNING** для любого **GACB** может быть выдана

Фон	У5 4566767	ЗАО	1787
Нам	Лист №	докум.	Подп. Дат.

Ц51.804.003 Д56

Лист
64

только задачей, включавшей (*зРАЗ*) этот *САСВ*; иначе задача, выдавшая макрокоманду *ATTNINQ*, ненормально заканчивается.

Программа внимания пользователя может обслуживать несколько задач. Однако для каждой комбинации задача-устройство-программа должен быть создан отдельный *САСВ*. Пользователь должен гарантировать, что задача, содержащая программу внимания, не оканчивается, в то время как другие задачи имеют активные (включенные в данный момент) *САСВ*, относящиеся к этой программе.

Изм	Лист	№ докум.	Подп.	Дат

Ц51.804.003 Д56

Лист
65

3. СПЕЦИАЛЬНЫЙ МЕТОД ОБРАБОТКИ ВНИМАНИЯ

3.1. Общие сведения

С помощью клавиатуры оператор индикатора ЕС-7066 может сообщить информацию системе. Нажатие специальной клавиши клавиатуры вызывает прерывание (сигнал внимания). Специальная обработка внимания требует проверки появления сигнала внимания со стороны пользовательской проблемной программы. При обнаружении внимания информация направляется к соответствующей программе обработки внимания.

Для обнаружения появления сигнала внимания и передачи проблемной программе информации о появлении внимания предусмотрены макрокоманды (см. табл. 8).

Таблица 8

Макрокоманды специального метода обработки внимания

Мнемоника	Наименование макрокоманды	Функции
ANALYZ	Анализ графического внимания	Определяет и анализирует появление внимания от индикатора. Перед передачей управления пользовательской программе производит сброс битов в GCB
GSEERV	Обслуживание графического внимания	Сбрасывает биты в GCB, связанном с вниманием. Эта макрокоманда используется, если пользователь желает выполнить свой собственный анализ внимания вместо использования макрокоманды ANALYZ

При появлении внимания от ЕС-7066 происходит прерывание про-

блемной программы и передача управления управляющей программе. Управляющая программа устанавливает бит в графическом байте управления (*GCB*), связанным с индикатором, от которого получен сигнал внимания. Затем управление возвращается проблемной программе в точку прерывания.

Чтобы определить, появился сигнал или нет, проблемная программа должна периодически проверять биты в *GCB*. Этот анализ наличия внимания может выполняться либо макрокомандой *ANALYZ*, либо какой-то написанной пользователем эквивалентной программой. В последнем случае для восстановления битов в *GCB* должна использоваться макрокоманда *GSERV*.

3.2. Макрокоманды специальной обработки сигналов внимания

Макрокоманды этой группы должны помочь пользователю обслуживать сигналы от EC-7066.

3.2.1. Макрокоманда анализа графического внимания - *ANALYZ*

Эта макрокоманда выполняет функцию анализа внимания, определяя появление и причину сигнала. Это делается с помощью аппарата опроса, проверяющего биты в *GCB*. Макрокоманда *ANALYZ* также собирает биты в *GCB* (вызывая макрокоманду *GSERV*), помещает в определенную пользователем область вывода закодированную информацию и передает управление соответствующей (пользовательской) программе для обработки сигналов внимания.

Для использования макрокоманды *ANALYZ* пользователь должен:

- включить в свою проблемную программу макрокоманду *ANALYZ*;
- предусмотреть непрерывный список адресов *DCB* и индексов *UCB*, описывающих устройства, от которых возможен сигнал. Если операнд *POLST* указан в макрокоманде *DCB*, то этот список создается во время открытия;
- указать адрес входа программы для обработки внимания;
- зарезервировать 8 байтов (начиная с границы полного слова)

Изм	Лист	№ докум.	Ноцн.	Плат

Ц51.804.003 Д56

Лист
67

-58-

в области своей программы, в которую могут быть помещены выходные
данные макрокоманды **ANALYZ**.
Формат макрокоманды **ANALYZ**:

Название	Операция	Операнд
[метка]	ANALYZ	адрес-списка-опроса, адрес-указателя [, <i>МР</i> = { <i>L</i> (<i>E</i> , имя-списка-па- раметров) (<i>E</i> , (<i>n</i>))}]

адрес-списка-опроса - адрес 4-байтной зарезервированной области,
за которой следует список адресов *DCB* и индексов *UCB*, указы-
вающих *UCB*, который содержит соответствующий *GCW*. Список опро-
са должен начинаться с границы полного слова. Зарезервированная
область используется **ANALYZ** для указания последнего адреса *DCB*
в списке. Если operand *POLST* указан в макрокоманде *DCB*, то
список составляется во время открытия. Формат списка следующий:

Зарезервированная область	
индекс <i>UCB</i>	адрес <i>DCB</i>
индекс <i>UCB</i>	адрес <i>DCB</i>

адрес-указателя - символьический адрес поля из пяти слов в памяти
программы, помещенного на границу полного слова
(см. табл. 9).

Таблица 9

Содержимое таблицы указателя

Слово	Содержимое
1	Не используется. Программист должен закодировать нули
2	Адрес пользовательской программы внимания
3	Не используется. Программист должен закодировать нули
4	Не используется. Программист должен закодировать нули
5	Адрес первого байта области вывода (см. ниже)

Примечание. В три из пяти слов, показанных в табл. 9, программист должен записать нули. Слово 2 таблицы указателя должно содержать действительный адрес программы внимания.

MF = L – указывает, что расширение макрокоманды должно состоять только из списка параметров. Наименование списка параметров – это символическое наименование, указанное в поле "Название" этой макрокоманды *ANALYZ*. В качестве operandов "список-опроса" и "указатель" могут указываться только перемещаемые выражения.

MF = (E, имя-списка-параметров) – указывает, что макрокоманда должна выполнить функции *ANALYZ*, используя список параметров, составленный другой макрокомандой *ANALYZ*. Операнд "имя-списка-параметров" должен точно соответствовать наименованию списка параметров.

MF = (E, (n)) – выполняет ту же самую функцию, что и *(E, имя-списка-параметров)*. *n* – номер регистра (в диапазоне от 2 до 12). Программист указывает, что он будет загружать адрес списка параметров в указанный регистр перед выполнением макрокоманды.

Изм.	Лист	№ докум.	Подп.	Лот.	Лист	69
					Ц51.804.003 Д56	

Ниже приводятся допустимые формы операнда для макрокоманды

ANALYZ :

- если ключевой operand MF опущен или указан как L , то operandы "адрес-списка-опроса" и "адрес-указателя" должны быть записаны как перемещаемые выражения;
- если указан ключевой operand $MF = E$, то operandы "адрес-списка-опроса", "адрес-указателя" и "имя-списка-параметров" могут быть записаны как перемещаемые выражения, явные или неявные, индексированные или неиндексированные адреса или в регистровой форме (см. примеры 9 и 10).

Пример 9.

```
LISTNAME ANALYZ POLST,POINTER,MP=L
          ANALYZ POLST1,MP=(E,LISTNAME)
```

Пример 10.

```
LA      2,POLST1
LA      3,LISTNAME
ANALYZ (2),MP=(E,(3))
```

Если во время компиляции пропущен operand "адрес-списка-опроса" либо "адрес-указателя", то в листинге компилятора печатается сообщение, указывающее пропуск, а макрогенерация продолжается. Если в исполнительное время отсутствует любой operand, то макрокоманда не выполняет своей функции.

Выходные данные макрокоманды **ANALYZ** размещаются в 8-байтной области вывода, адрес которой содержится в пятом слове таблицы указателя. Эта область должна начинаться на границе полного слова. Выходные данные состоят из следующих частей в указанном порядке:

- резервируемая (4 байта);
- индексный множитель адреса **UCB** (1 байт);
- адрес **DCB** (3 байта).

Если при проверке списка-опроса сигнала внимания не обнаружено, то в регистр кода возврата помещается код **x'04** и управление возвра-

Изм	Лист № докум.	Подп. Дат.

шестой следующей команде в программе пользователя. При обнаружении внимания макрокоманда определяет адрес *CSA*, восстанавливает *CSB*, запоминает выходные данные в области вывода и передает управление соответствующей пользовательской программе обработки внимания с адресом области вывода в регистре I и адресом возврата в регистре 14. Пользователь должен сохранить и восстановить регистры в его программе обработки внимания.

Следует отметить, что при завершении работы пользовательская программа обработки внимания не возвращает управление *ANALYZ*. Управление от программы обработки внимания возвращается следующей после *ANALYZ* команде с помощью макрокоманды *RETURN*.

Первоначально макрокоманда *ANALYZ* выполняет опрос устройств, начиная с первого элемента списка опроса. При обнаружении сигнала внимания, соответствующего одному из элементов списка, макрокоманда *ANALYZ* передает управление программе обработки сигнала внимания. После обработки сигнала внимания *ANALYZ* возобновляет опрос с элемента списка, следующего за элементом, который соответствует последнему обработанному сигналу внимания.

Выполняется возврат к следующей команде, если сигналы внимания не появлялись, или в слове 2 таблицы указателя (см. табл. 9) нет адреса программы внимания, или в слове 5 таблицы указателя нет адреса области вывода. Если нет программы внимания, то сигналы теряются. Регистр I5 содержит шестнадцатеричный код для указания причины возврата. Ниже приведены коды возврата и их значения:

00 - нормальный;

04 - сигналы внимания не появились;

08 - внимание обнаружено, но соответствующей программы внимания не предусмотрено;

0C - не указан либо "адрес-списка-опроса", либо "адрес-указателя";

10 - нет области вывода.

Лам	Лист	№ докум.	Полл, Дата

Пример II иллюстрирует обычный способ программирования для EC-7906 с использованием макрокоманды ANALYZ.

Пример II.

	ANALYZ	POLST, POINTER
	• • •	
POLST	DC	F'0'
	DS	2F
POINTER	DC	F'0'
	DC	A(KBRTN)
	DC	2F'0'
	DC	A(OUTAREA)
OUTAREA	DC	2F'0'

POLST – символический адрес первой записи длиной в полное слово в списке адресов **DC** и индексных множителей **UCS**. Отметим, что первое слово в списке устанавливается в нуль. **POINTER** – символический адрес первой записи в таблице из пяти слов, которая содержит адрес входа пользовательской программы обработки внимания и адрес области вывода (**OUTAREA**).

Пользователь открывает графическое устройство, создавая тем самым соответствующие блоки управления. Следующая процедура – это обычно выполнение канальной программы для информирования оператора индикатора о том, что графическая программа в данный момент выполняется. Если далее должна быть обработка информации внимания, то может быть начат опрос индикаторов. Опрос проводится периодически в течение отрезка времени, когда логически могут появиться сигналы внимания от индикатора (индикаторов). Синхронизация этой функции может выполняться либо при помощи макрокоманды **ANALYZ** в соответствующие моменты выполнения проблемной программы, либо путем использования системной макрокоманды **STIMER** (установить интервальный

таймер), которая позволяет выполнять макрокоманду *ANALYZ* через указанные пользователем интервалы времени (формат и использование макрокоманды *STIMER* см. в [9]). Однако если таймер используется для установления интервала между циклами опроса, то он не может использоваться для установления других интервалов задачи.

Примечание. При желании пользователь может ожидать сигнала внимания посредством защелкивания программы опроса (т.е. циклический переход к макрокоманде *ANALYZ*). Однако необходимо отметить, что такой метод очень сильно загружает центральный процессор и тем самым нарушает одну из главных целей системы, обладающей возможностями мультипрограммирования.

Последовательность команд, приведенных в примере 12, показывает, как может быть использован интервальный таймер ЕС ЭВМ для установления интервала опроса.

Пример 12.

```
• • •  
OPEN    DCB1  
STIMER  REAL,POLADR,  
        DINTRVL=TIME
```

■ Основная программа

```
• • •  
CLOSE   DCB1  
• • •  
DS      OD  
TIME   DC      z'00000020'  
POLADR ANALYZ POIST,POINTER
```

■ Если нет внимания - возврат к основной программе,
После открытия *DCB* для устройства использование макрокоманды *STIMER* обеспечивает автоматический переход к программе опроса после указанного пользователем интервала реального времени (0,2 секунды).

Изм Лист № докум. Подп. Дата:

До истечения интервала продолжается обработка в основной программе. Действия, предпринятые стандартной программой опроса (*ANALYZ*), зависят от факта появления сигнала внимания. Если сигнал обнаружен, то управление передается пользовательской программе обработки внимания; если сигнал не обнаружен, то управление возвращается к команде, следующей за макрокомандой *ANALYZ*.

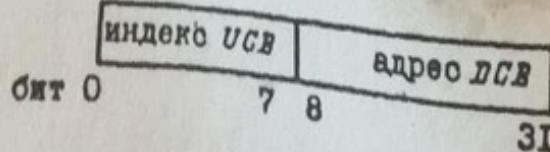
3.2.2. Макрокоманда обнуливания графического внимания - *GSERV*

Эта макрокоманда восстанавливает *DCB*, связанные с вниманием так, что никакой сигнал не обрабатывается более одного раза. Эта макрокоманда используется *ANALYZ* для сброса битов в *DCB*. Если пользователь хочет написать собственную программу опроса и анализа внимания, то для сброса битов в *DCB* (т.е. установления в нуль) должна использоваться *GSERV*.

Формат макрокоманды *GSERV*:

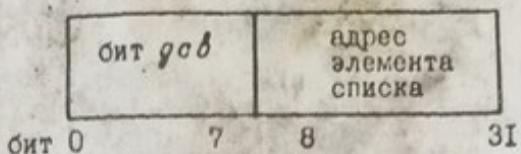
Название	Операция	Операнд
[метка]	<i>GSERV</i>	{адрес-элемента-списка, бит <i>dcb</i> (<i>I</i>)}

адрес-элемента-списка - символьический адрес 4-байтного поля, помещенного на границу полного слова, в списке опроса (т.е. в списке адресов *DCB* и индексов *UCB*). Этот список составляется во время открытия, если операнд *POLST* указан в макрокоманде *DCB*. 4-байтное поле содержит адрес *DCB* и индекс *UCB*, указывающий соответствующий устройству *UCB*, и имеет следующий формат:



бит *dev* - указывает, какой бит в *SCB* должен быть сброшен. Значение "бит *dev*" для EC-7066 должно быть 1.

Если используется регистровая форма, то перед выполнением макрокоманды параметры должны быть загружены в регистре 1. Формат загрузки информации в регистр следующий:



Для определения сбрасываемого бита программист кодирует старший байт в регистре следующим образом:

сбрасываемый бит *SCB* - 7;

код - *x'01*.

Адрес, занимающий 3 правых байта, может быть получен из списка опроса, построенного с помощью макрокоманды *OPEN*, если операнд *POLAR* был указан в макрокоманде *SCB*.

Если адрес *SCB* является недействительным или если устройство не является графическим, то задание ненормально заканчивается.

После того как *CLEAR* восстановит бит *SCB*, управление возвращается к следующей команде вызывающей программы.

Примечание. Графический байт управления (*SCB*) устанавливается в нуль, когда открывается первый *DCB* для любого индикатора.

КРАТКИЙ ОБЗОР ОПЕРАНДОВ

Краткий обзор operandов, описанный в данном приложении, содержит все макрокоманды и указывает допустимые способы записи operandов. Краткий обзор составлен так, что он может быть использован со всеми рассмотренными макрокомандами при их изучении (см. таблицу). В обзоре используются следующие обозначения:

SYM - любой символ, действительный в языке Ассемблера;

DEC INT - любые десятичные цифры вплоть до значения, указанного в описании соответствующей макрокоманды. Если *SYM* и *DEC INT* разрешены, то допускается также абсолютное выражение;

регистр - всегда кодируется в круглых скобках следующим образом:

(2-I2) - один из общих регистров от 2 до I2, в который предварительно загружается смещение, выровненное вправо значение или адрес, указанный в описании соответствующей макрокоманды. Неиспользованные старшие биты должны устанавливаться в нуль. Регистр может быть указан символьическим или абсолютным выражением.

(I) - общий регистр I, предварительно загруженный, как указано выше. Регистр может обозначаться только как (I).

Примечание: Регистровая форма может использоваться только с *E*-формой макрокоманды.

RX- тип - любой адрес, допустимый в команде типа - *RX* (например, *LA*);

Изм	Лист	№ докум.	Подп.	Дат
-----	------	----------	-------	-----

A - тип - может быть любым адресом, который можно записать
адресной константой *A*-типа.

Примечания: 1. Подчеркивание указывает выбор по умолчанию, т.е.
если операнд не закодирован, то подразумевается
подчеркнутый операнд.

2. Скобки { } указывают, что если не указан выбор
по умолчанию, то должен быть выбран один операнд
из горизонтальной или вертикальной группы.

3. Вертикальная черта (|) отделяет альтернативные
операнды в горизонтальной группе.

Изм	Лист	№ докум.	Подп.	Дата

Ц51.804.003 Д56

Лист

77

Макрокоманды и операции

Макрокоманда	Операции	SYN	URG INT	Возможное обозначение		
				Регистр *	(2-12)	(*)
ANALYZ	адрес списка опроса адрес указателя $MP = \begin{cases} L \\ (\Sigma, \\ \text{адрес списка параметров}) \end{cases}$			См. описание		X-TEN
ATTENDING	адрес басв MODE = $\begin{cases} R \\ W \\ (C, \\ \text{адрес перехода}) \\ CLEAR \end{cases}$			Кодируется, как написано		X
	UNIT=			Кодируется, как написано		X
CLOSE	$MP = \begin{cases} L \\ (\Sigma, \\ \text{адрес списка параметров}) \end{cases}$			Кодируется, как написано		X
DAR	адрес gasbv $MP = \begin{cases} L \\ (\Sigma, \\ \text{адрес списка басв}) \end{cases}$			Кодируется, как написано		X

* Регистровая форма операндов может использоваться только в Е-формой макрокоманд

Имя регистра	Номер регистра	Поле данных	Поле адреса

Устр. 304.000 276

Продолжение

Макрокоманда	Операторы	Бозничное обозначение					
		SYN	DSC INT	Регистр		III- REG	IV- REG
DS		X	(2-12)	(1)			
DS.ORG=GS							
MACR Pa		[3 7 BC EC (2,8) (BC,8) (2,EC) (BC,EC)]					
TYPE=		BASIC- EXPRESS					
DDNAME=MN DD		X					
ALIAS=адрес списка выходов							
NCPR=число блоков		X					
POLST=адрес списка яблок							
CONTROL							
ADDRESS decb		X					
TYPE		{DSC REGW}					
адрес decb			X				
УСТРОЙСТВО			X				
MP=			X				
GPB#AD							
ADDRESS decb		X		X	X		
TYPE		{DSN DSIN SMCT SMW DB NSBB}					
адрес decb			X				
ДЛЖНА			X	X			
адрес областей			X	X			
УСТРОЙСТВО			X	X			
MP=			{L S}				
RESERV							
адрес элемента списка							
ОИТ decb		X					
GRBIT#							
ADDRESS decb		X		X	X		
TYPE		{DS3 DSIN DBW SWW LAN LANW ANL SWLW}					
адрес decb			X				
ДЛИНА			X	X			
адрес областей			X	X			
УСТРОЙСТВО			X	X			
MP=			{L S}				

* Регистровая форма операндов может использоваться только с Е-формой макрокоманд

[51.804.003 E56]

79

Примечания

Напоминания	Операции	Запись в буфер					
		SYN	DSC	REP	(2-12)	(1)	RS-
OP32	SP32 dec b						
	REP _b	I	(Σ,		X		
			адрес списка параметров)				
SAC	SP32 { адрес точки входа 0}	X			X		
	DCЗапрос сб			X			
	СОЗДАНИЕ/РЕДАКТИРОВАНИЕ ОБЛАСТИ СЕТЬ				X		
	AT32IP _b	(((A S I Z), 7066) GAVS RESTOR)					
	REP _b	Z E					
SP42	SP42 dec b						
	FECI _b			X			
	REP _b	I	(Σ,		X		
			адрес списка бакб)				

• Регистрации формата операций может осуществляться только с Г-формой манипуляции

Приложение 2

ОБРАЗЕЦ ПРОГРАММЫ

Программа, приведенная в нижеследующем примере, предназначена для работы с двумя индикаторами ЕС-7066. Печатающее устройство не требуется.

Для этой программы предполагается, что первые семь знаков сообщения оператора или содержат какие-то знаки, определяющие, обрабатывает ли данная программа сообщение и запоминает результаты, или в случае отсутствия знаков оканчивает задание и воспроизводит результаты, полученные при обработке каждого индивидуального запроса.

Функции каждого элемента программы поясняются комментариями.

Перед тем, как делаются ссылки к ЛСЯ для ЕС-7066, он должен быть открыт.

Макрокоманда **READ**, показанная в примере, генерирует операцию считывания ручного ввода индикатора (**DSM**).

Макрокоманда **WRITE** выдает операцию стирания и записи в буфер индикатора (**LW**) для очистки буфера и вывода сообщения оператору.

Все определенные пользователем программы рассматриваются как подпрограммы, и поэтому для них должны соблюдаться соглашения о связях между подпрограммами. Информация из **COMMON** должна быть сохранена, если существует возможность прерывания программы внимания другой программой внимания, совместно с первой использующей **COMMON**.

Изм	Лист	№ докум.	Подп.	Дат.

Ц51.804.003 Д56

Лист
81

SAMR7066

CSECT

SAVE
(14,12)BALR
5,0USING
*,5ST
13,SAVE2+4LA
13,SAVE2OPEN
(DCB1)SPAR
(GACB1),PRTY=1LTR
15,15BNZ
ERTN1

WAIT

CLI
DECB1,X'7P'BNB
ERTN2

WAIT

CLI
DECB5,X'7P'BNB
ERTN3

WAIT

CLI
DECB5,X'7P'BNB
ERTN4

WAIT

CLI
DECB5,X'7P'BNB
ERTN5

WAIT

CLI
(GACB1)

ЗАПОЛНИТЬ РЕГИСТРЫ
ЗАРУЗКА БАЗОВОГО РЕГИСТРА
ОПРЕДЕЛЕНИЕ БАЗОВОГО РЕГИСТРА
СВИЗАТЬ ОБЛАСТИ СОХРАНЕНИЯ
АДРЕС НОВОЙ ОБЛАСТИ СОХРАНЕНИЯ
ОТКРЫТЬ 7066
ВКЛЮЧЬТЬ ПРОГРАММУ ОБРАБОТКИ ЗЕМЛЯНИИ 07 7066
ПРОВЕРИТЬ КОД ВОЗВРАТА ОТ SPAB
ПЕРЕХОД, ЕСЛИ КОД ВОЗВРАТА ≠ 0
ДЛЯ ПЕРВОГО УСТРОЙСТВА

ПРОВЕРИТЬ, ПРИЕНЫЛ ПАРАМЕТРЫ
ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОЦЕНКИ ВВОДА-ВЫВОДА
ОЦЕНКА ОКОНЧАНИЯ ОПЕРАЦИИ ВВОДА-ВЫВОДА
ПРОВЕРИТЬ, УСПЕШНО ЛИ ЗАВЕРШЕН ВВОД-ВЫВОД
ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОЦЕНКИ ВВОДА-ВЫВОДА
ДЛЯ ВТОРОГО УСТРОЙСТВА

ПРОВЕРИТЬ, ПРИЕНЫЛ ПАРАМЕТРЫ
ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОЦЕНКИ ВВОДА-ВЫВОДА
ОЦЕНКА ОКОНЧАНИЯ ОПЕРАЦИИ ВВОДА-ВЫВОДА
ПРОВЕРИТЬ, УСПЕШНО ЛИ ЗАВЕРШЕН ВВОД-ВЫВОД
ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОЦЕНКИ ВВОДА-ВЫВОДА
ДЛЯ ТРЕТЬЕГО УСТРОЙСТВА

```

    LTR      *15,15          ПРОВЕРИТЬ КОД ВОЗВРАТА ДЛЯ DAB
BNZ      ERTN1          ПЕРЕХОД. ЕСЛИ КОД ВОЗВРАТА НЕ РАВЕН НУЛЮ
    EQU      CLOSE           ЗАКРЫТЬ 7066
    ■       (DCB1)          ЗАГРУЗИТЬ АДРЕС СТАРОЙ ОСЛАСТИ СОХРАНЕНИЯ
L       13,SAVE2+4        ВОЗВРАТ. К СУЩЕРВИЗОРУ
    RETURN   (14,12)
    ■

    ■ ТОЧКА ВХОДА: ПРОГРАММА В СЛУЧАЕ ОШИБКИ В SPAR/DAB
ERTN1   B     OUT1
    ■ ТОЧКА ВХОДА: ПРОГРАММА В СЛУЧАЕ ОШИБКИ ВЪВЛА-ШЕОДА
ERTN2   B     OUT
    ■

    ■ ПРОГРАММА ОБРАБОТКА ВНЕШНЯЯ ОТ КЛАВИАТУРЫ 7066
    ■
KEYBDRTN SAVE      (14,12)          ЗАПОМЕТЬ РЕГИСТРЫ
                                5          ОСЕБОВОДИТЬ БАЗОВЫЕ РЕГИСТРЫ
DROP    6,Ø          ЗАГРУЗКА БАЗОВОГО РЕГИСТРА
BAIR    ■,6          ОПРЕДЕЛЕНИЕ БАЗОВОГО РЕГИСТРА
USING   13,SAVE3+4    СВЯЗАТЬ ОСЛАСТИ СОХРАНЕНИЯ
ST      13,SAVE3
LA      CLI          УСТАНОВКА НОВОЙ ОСЛАСТИ СОХРАНЕНИЯ
    ■
READ1   -             ПРОВЕРИТЬ, ОТ ВТОРОГО УСТРОИСТВА ПОСЛЕДОВАТЕЛЬНОЕ ВЪДЕЛАНЕ
    ■
    ■ READ2            ИЛИ НЕТ. ТО ПЕРЕХОД К ЧТЕНИЮ ВТОРОГО УСТРОИСТВА
    ■
    ■ 'READ1'          ЕСЛИ ДА, ТО ОБРАБАТЫВАЕТ ВЪДЕЛАНЕ ОТ ПЕРВОГО УСТРОИСТВА;
    ■ 'READ2'          ОБСЛУЖИВАЕТ ВТОРОЕ УСТРОИСТВО
    ■

```

Нам.	Лист	№ докум.	Подп.	Дат.

READ1

EQU

GRREAD

LTR 15,15

BNZ ERTN3

WAIT

DECB=DEC32

DECB2,X'7F'

COMMON

DEC32+16,X'45'

ERTN3

CONTIN

EQU

RRAD2

DEC32,DSM,DCB1,96#,INPUT,1
ПРОВЕРТЬ, ПРДЛЕНН И ПАРАМЕТР.
ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОПЕРАЦИИ ЗВОЛ-ВЫВОДА
ОБЫЧАЕС СКОНЧАЕНИ ОПЕРАЦИИ ЗВОЛ-ВЫВОДА
ПРОВЕРИТЬ, УСПЕШНО ЛИ ЗАВЕРШЕНА ОПЕРАЦИИ ЗВОЛ-ВЫВОДА
ЕСЛИ ДА, ТО ПРОДОЛЖАТЬ ОБРАБОТКУ
ПРОВЕРИТЬ НАЛГИЕ ПОСТОЯННОЙ ОБЛЕН ПРИ ЧТЕНИ
ЕСЛИ ЕСТЬ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОПЕРАЦИИ ВОЛД-ЗВОСДА
ЕСЛИ НЕТ, ПРОДОЛЖАТЬ ОБРАБОТКУ

GRREAD

LTR 15,15

BNZ ERTN3

WAIT

DECB=DEC36

DECB2,X'7F'

COMMON

DEC36+16,X'45'

ERTN3

INPUT(6),PRC

COMMON

CONTIN

EQU

RRAD2

ЧТЕНИЕ С УСТРОЙСТВА 2
ПРОВЕРИТЬ, ПРДЛЕНН И ПАРАМЕТР.
ЕСЛИ НЕТ, ПЕРЕХОД К ОБРАБОТКЕ ОПЕРАЦИИ ЗВОСД-ВЫВОДА
СЧИТАТЬ СЧИТЫЕ ЗВОЛ-ВЫВОДА
ПРОВЕРТЬ, УСПЕШНО ЛИ ЗАВЕРША СЧИТАТЬ ЗВОЛ-ВЫВОДА
ЕСЛИ ДА, ТО ПРОДОЛЖАТЬ ОБРАБОТКУ
ПРОВЕРИТЬ НАЛГИЕ ПОСТОЯННОЙ ОБЛЕН ПРИ ЧТЕНИ
ЕСЛИ ЕСТЬ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОПЕРАЦИИ ВОЛД-ЗВОСДА
ПРИКАЗ ПРОДОЛЖАТЬ ОБРАБОТКУ
ОПЕРАЦИЯ 'РАБОТА' В ПЕРВЫХ 6 БАЛТАХ СОСДЕДН
ДА, ПРОДОЛЖАТЬ ОБРАБОТКУ, ПЕРЕХОД К ПРОГРАММЕ ОБРА
БИ БЫЛЫХ ДАННЫХ.

GRWRITE DEC33,DSM,DCB1,96#,OUTPUT,1
СТИРТЬ СОЛНЕЧНОЕ ЗЕРКАЛО И ВЫСВЕЧИТЬ ОБРА
БИ БЫЛЫХ ДАННЫХ
ПРОВЕРИТЬ, ПРДЛЕНН И ПАРАМЕТР

Лист	№ зоокум.	Подп.	Лот
1	1	1	1

551.804.003 Д56

Лист
84

BNZ ERTN3
 WAIT ECB=DEC83
 CLI DECB3,X'7F'
 BNZ ERTN3
 GWRITE DCB87,E8H,DCB1,12,TER1SG,²

 LTR 15,15
 ERTN3
 COLSAVE+16
 13,SAVE3+4
 RETURN (14,12)

OUT3
 POST L
 CONTIN EQU CLI
 BB DEV2

GWRITE DECB4,E8H,DCB1,47,PRCLSG,¹
 LTR 15,15
 BNZ ERTN3
 WAIT ECB=DEC84
 CLI

* ПРОГРАММА ОБРАБОТИЛ РУЧНОГО ВВОДА ПО ТРЕБОВАНИЮ 7066

* ВЫВОД СООБЩЕНИЯ 'ОБРАБОТКА ЗАКОНЧЕНА' НА УСТРОЙСТВО 1

* 151.804.003 Д56

-85-

ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОШИБКИ ВВОДА-ВЫВОДА
 ОГЛАДИНИЕ ОКОНЧАНИЯ ВВОДА-ВЫВОДА
 ПРОВЕРИТЬ, УСПЕШНО ЛИ ЗАВЕРШИЛСЯ ВВОД-ВЫВОД
 ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОШИБКИ ВВОДА-ВЫВОДА
 СТЕРЕТЬ СОДЕРЖАМОЕ ЭКРАНА И ВЫСВЕТИТЬ 'СООБЩЕНИЕ
 НИЕ 'КОНЕЦ ЗАДАЧИ' НА ЭКРАН ИТОГО УСТРОЙСТВА
 ПРОВЕРИТЬ, ПРИНЯТЫ ЛИ ПАРАМЕТРЫ
 ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОШИБКИ ВВОДА-ВЫВОДА
 УСТАНОВИТЬ КОД ЗАВЕРШЕНИЯ ЦИКЛА
 ВОССТАНОВИТЬ РЕГИСТР Г3
 ВОЗВРАТ К ФОННОВОМУ ПРОГРАММЕ

ПРОВЕРИТЬ, ПОСТАВЛЕНЫ ЛИ ПАРАМЕТРЫ
 УСТРОЙСТВА ИЛИ НЕТ
 ЕСЛИ ДА, ТО ПЕРЕХОД К УЧАСТКУ 'DEV2'

ПРОВЕРИТЬ, ПОСТАВЛЕНЫ ЛИ ПАРАМЕТРЫ
 УСТРОЙСТВА ИЛИ НЕТ
 ЕСЛИ ДА, ТО ПЕРЕХОД К УЧАСТКУ 'DEV2'

ПРОВЕРИТЬ, ПОСТАВЛЕНЫ ЛИ ПАРАМЕТРЫ
 УСТРОЙСТВА ИЛИ НЕТ
 ЕСЛИ ДА, ТО ПЕРЕХОД К УЧАСТКУ 'DEV2'

ПРОВЕРИТЬ, ПОСТАВЛЕНЫ ЛИ ПАРАМЕТРЫ
 УСТРОЙСТВА ИЛИ НЕТ
 ЕСЛИ ДА, ТО ПЕРЕХОД К УЧАСТКУ 'DEV2'

ВНВ

В

СОМГ

EQU

DEV2

*

* ВЫВОД СООБЩЕНИЯ "ОБРАБОТКА ЗАКОНЧЕНА" НА УСТРОЙСТВО 2

GWRITE

DECBS, EEW, DCB1, 48, PRMSG, 2

LTR

15, 15

ERTW 3

ECB=DECB8

DECBS, X'7F'

CLT

BNB

ATTNLNQ

GACB1, MODE=(C, READ1), UNIT=1

■

ATTNLNQ

GACB1, MODE=(C, READ2), UNIT=2

■

CONSAVE+20, X'01'

TM

BC

OT

ATTNLNQ

GACB1, MODE=W, UNIT=2

READ2

■

COM2

ATTNLNQ

GACB1, MODE=E

■

HEADM

■

TOЧКА ВЫСЛА ПРОГРАММЫ ОБРАБОТКИ ОШИБКИ ВВОДА-ВЫВОДА

ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОШИБКИ ВВОДА-ВЫВОДА
ЕСЛИ ДА, ТО ПРОДОЛЖЬТЬ ОБРАБОТКУ

ПРОВЕРИТЬ, ПРИНЯТЫ ЛИ ПАРАМЕТРЫ
ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОШИБКИ ВВОДА-ВЫВОДА
ОБИДАНИЕ ОКОНЧАНИЯ ВВОДА-ВЫВОДА
ПРОВЕРИТЬ, УСПЕШНО ЛИ ЗАВЕРШИЛСЯ ВВОД-ВЫВОД
ЕСЛИ НЕТ, ТО ПЕРЕХОД К ОБРАБОТКЕ ОШИБКИ ВВОДА-ВЫВОДА
ПРОВЕРКА НАЛЕГЧИЯ СИГНАЛОВ ЕНИМАНИЯ ОТ УСТРОЙСТВА 1

ПРОВЕРКА НАЛЕГЧИЯ СИГНАЛОВ ЕНИМАНИЯ ОТ УСТРОЙСТВА 2

ПРОВЕРКА ПРИЗНАКА ИСПОЛЬЗОВАНИЯ РЕГИСТРА W
ЕСЛИ ДА, ТО ПЕРЕХОД К УЧАСТКУ "СОЛ2"
УСТАНОВКА ПРИЗНАКА РЕГИСТРА T

ОБИДАНИЕ СИГНАЛА ЕНИМАНИЯ ОТ УСТРОЙСТВА 2
ПРОГРАММЫ

ОТДАТЬ УПРАВЛЕНИЕ ФОННОВОЙ ПРОГРАММЕ

ПЕРЕХОД К НАЧАЛУ ПРОГРАММЫ ОБРАБОТКИ ВВОДА-ВЫВОДА

ERTR3 B OUT3

*** ОБЛАСТЬ СООБЩЕНИЙ ***

OPNMSG DC CL43 'ИНИЦИАТОР ГЛОВ. - ВХОД ИЗ ЗАПРОС'
 TBNMSG DC CL12 'КОНЕЦ ЗАДАЧИ'
 PRMSG DC CL47 'ОБРАБОТКА ЗАКОНЧЕНА - ВВЕДИ СЛЕДУЮЩИЙ ЗАПРОС'

*** ОБЛАСТЬ КОНСТАНТ ***

DCB1 DCB DS ORG=GS, MACRF=(R,W), DDNAME=GRAPHIC, GNCP=4, STPTE=BASIC
 GACB1 SAEC
 DS DB
 SAVE1 DC 18F'0'
 SAVE2 DC 18F'0'
 SAVE3 DC 18F'0'
 COMSAVE DC 6F'0'
 PRC DC CL6'РАБОТА'
 INPUT DC 240F'0'
 END
 SAMP7066

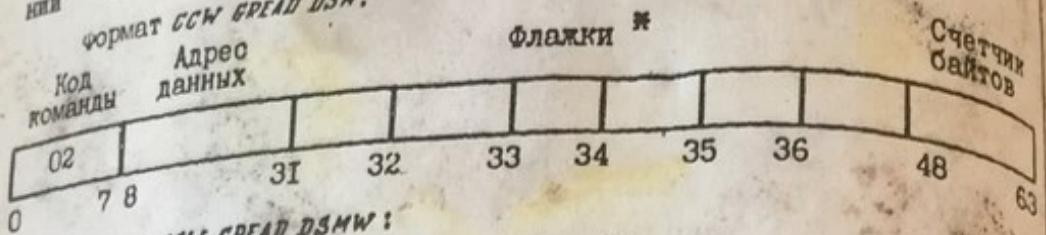
ОБЛАСТЬ СОХРАНЕНИЯ ДЛЯ GACB1
 КОД ПРИКАЗА ДЛЯ ПРОДОЛЖЕНИЯ ОБРАБОТКИ
 БУФЕР ДЛЯ СЧИТЫВАНИЯ ДАННЫХ С ЭКРАНА

Изм.	Лист	№ докум.	Подп.	Дат.

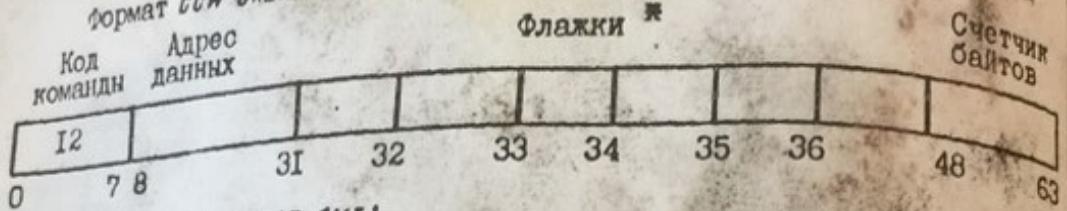
Приложение 3

ФОРМАТЫ ССW, СВЯЗАННЫЕ С УПРАВЛЕНИЕМ ВВОДОМ-ВЫВОДОМ
ниже приводятся форматы ССW, вырабатываемых при использова-
нии макрокоманд GWRITE, GREAD и GCNTRL.

формат ССW GREAD DSN:



формат ССW GREAD DSMW:



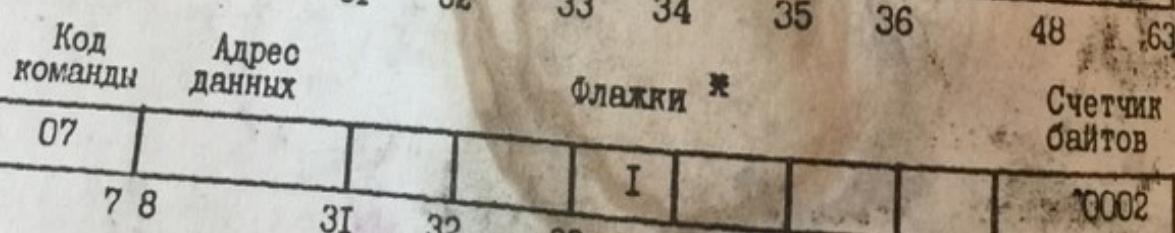
формат ССW GREAD SMI:



формат ССW GREAD SMIW:



Форматы ССW GREAD DS8:



*

Бит 33 - флагок цепочки команд; бит 34 - флагок подавления
индикации неправильной длины.

Форматы CCW GREAD DSBW:

Код команды	Адрес данных	Флажки *								Счетчик байтов
16		I								
0	78	31	32	33	34	35	36	48	63	

Код команды	Адрес данных	Флажки *								Счетчик байтов
17		I								0002
0	78	31	32	33	34	35	36	48	63	

Формат CCW GWRITE DSB:

Код команды	Адрес данных	Флажки *								Счетчик байтов
0I		I								
0	78	31	32	33	34	35	36	48	63	

Формат CCW GWRITE DSBW:

Код команды	Адрес данных	Флажки *								Счетчик байтов
II		I								
0	78	31	32	33	34	35	36	48	63	

Формат CCW GWRITE EBW:

Код команды	Адрес данных	Флажки *								Счетчик байтов
02		I	I							0002
0	78	31	32	33	34	35	36	48	63	

Код команды	Адрес данных	Флажки *								Счетчик байтов
0I		I								
0	78	31	32	33	34	35	36	48	63	

* Бит 33 – флагок цепочки команд; бит 34 – флагок подавления индикации неправильной длины.

Форматы CCW GWRITE IBWW :

Код команды		Адрес данных		Флажки *				Счетчик байтов	
I7				I	I			0002	
0		78		31	32	33	34	35	36
Код команды		Адрес данных		Флажки *				Счетчик байтов	
II				I				48	
0		78		31	32	33	34	35	36

Формат CCW GWRITE LNE :

Код команды		Адрес данных		Флажки *				Счетчик байтов	
05				I				48	
0		78		31	32	33	34	35	36

Формат CCW GWRITE LNEW:

Код команды		Адрес данных		Флажки *				Счетчик байтов	
I5				I				48	
0		78		31	32	33	34	35	36

Форматы CCW GWRITE EWL :

Код команды		Адрес данных		Флажки *				Счетчик байтов	
07				I	I			0002	
0		78		31	32	33	34	35	36

Код команды		Адрес данных		Флажки *				Счетчик байтов	
05				I				48	
0		78		31	32	33	34	35	36

* Бит 33 - флагок цепочки команд; бит 34 - флагок подавления индикации неправильной длины.

Формат CCW GWRITE EWTW:

Код команды	Адрес данных	Флажки *		Счетчик байтов	
I7		I	I		0002

Код команды	Адрес данных	Флажки *		Счетчик байтов	
I5		I			0002

Формат CCW GCNTRL ERS :

Код команды	Адрес данных	Флажки *		Счетчик байтов	
07		I			0002

Формат CCW GCNTRL ERSW:

Код команды	Адрес данных	Флажки *		Счетчик байтов	
I7		I			0002

* Бит 33 - флагок цепочки команд; бит 34 - флагок подавления индикации недравильной длины.

ПЕРЕЧЕНЬ ССЫЛОЧНЫХ ДОКУМЕНТОВ

1. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Концепции и возможности
Общее описание
Ц51.804.004 Д1
2. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Ассемблер
Описание языка
Ц51.804.004 Д16
3. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Редактор связей
Руководство программиста
Ц51.804.002 Д10
4. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Сообщения супервизора и управления данными
Руководство оператора
Ц51.804.002 Д29
5. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Генерация
Руководство системного программиста
Ц51.804.004 Д33

6. Едина
ОПЕРА
Супе
Руко
Ц51.
7. Едина
ОПЕ
Доп
Руко
Ц5
8. Едина
ОПЕ
Я
9.
10.

6. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Супервизор
Руководство программиста
Ц51.804.002 Д3
7. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Дополнительные возможности
Руководство системного программиста
Ц51.804.002 Д35
8. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Язык управления заданиями
Описание языка
Ц51.804.001-01 Д2
9. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Макрокоманды супервизора и управления данными
Руководство программиста
Ц51.804.002 Д5
10. Единая система электронных вычислительных машин
ОПЕРАЦИОННАЯ СИСТЕМА
Управление данными
Руководство программиста
Ц51.804.002 Д4

Изм.	Лист
К докум.	Подп. Дата

Ц51.804.003 Д56

Лист
93



Подписано к печати 5/02-1980г.

Заказ 005

Формат 60x84 $\frac{1}{16}$. Тираж 3025 экз. Объем 6 п.л.

Производственно-издательский комбинат ВИНИТИ

Люберцы, Октябрьский пр., 403