

Г. А. ДРОБУШЕВИЧ

СПРАВОЧНИК ПРОГРАММИСТА



10 EC
EC 20

Г. А. ДРОБУШЕВИЧ

СПРАВОЧНИК ПРОГРАММИСТА

Минск «Вышэйшая школа» 1978

6П5.8
Д 75
УДК 681.3(031)

Д 75 Дробушевич Г. А.
Справочник программиста.— Мн.: Выш. школа, 1978.—
192 с., ил.

В книге даны необходимые сведения по языкам программирования ЕС ЭВМ АССЕМБЛЕР, ФОРТРАН, КОБОЛ, ПЛ/1, РПГ, операционным системам ОС и ДОС, а также приводится таблица кодов ДКОИ и англо-русский словарь программиста. Книга рассчитана на студентов, преподавателей, работников ВЦ, на всех, кто использует ЭВМ.

Д $\frac{30502-164}{М 304(05)-78}$ 86—78

6П5.8

ПРЕДИСЛОВИЕ

Эффективное использование ЕС ЭВМ невозможно без хорошего знания языков программирования АСЕМБЛЕР, ФОРТРАН, КОБОЛ, ПЛ/1, РПГ, возможностей операционной системы (ОС) и дисковой операционной системы (ДОС). Для изучения языков программирования необходимы описания и руководства, которые даются в технической документации по операционным системам, а также учебники и учебные пособия. При использовании этих языков программирования необходимы справочники.

Значительное время при разработке программного обеспечения занимает отладка программ и поиск ошибок, многие из которых обнаруживаются при трансляции. К сожалению, сообщения об ошибках и весь листинг программы печатаются на английском языке. Перевод текста сообщения представляет значительную трудность даже для тех, кто неплохо владеет английским языком. Воспользоваться же переводами, приведенными в технической документации, затруднительно из-за крайне ограниченного числа экземпляров технической документации, получаемых вместе с ЭВМ.

Настоящий справочник содержит краткие сведения по указанным выше языкам программирования и операторам управления заданиями в ДОС ЕС.

Справочник состоит из шести глав. В первой главе рассматриваются операторы управления заданиями ДОС ЕС ЭВМ. В последующих пяти главах рассматриваются языки программирования ЕС ЭВМ АСЕМБЛЕР, ФОРТРАН, КОБОЛ, ПЛ/1, РПГ. В каждой из этих глав даются краткие сведения о языке, правила записи программ на бланках и список сообщений об ошибках во время трансляции (номер ошибки, английский текст и перевод). Приводится англо-русский словарь, содержащий наиболее часто встречающиеся английские слова в терминологии программиста.

При составлении справочника автор использовал материал технической документации по ДОС ЕС, а также другие материалы из опубликованных источников.

Автор приносит глубокую благодарность докт. техн. наук, проф. Р. И. Фурунжиеву, взявшему на себя нелегкий труд рецензирования рукописи, а также сотрудникам кафедры общего программирования БГУ, которые оказали помощь в подборе материала для книги и сделали свои замечания по ее содержанию.

Автор будет признателен читателям за все замечания, которые просит присылать по адресу: 220080, г. Минск, БГУ им. В. И. Ленина, факультет прикладной математики, кафедра общего программирования.

УПРАВЛЯЮЩИЕ ОПЕРАТОРЫ ДОС ЕС

ОПЕРАТОРЫ УПРАВЛЕНИЯ ЗАДАНИЯМИ

1. JOB — начать задание.

Оператор JOB указывает начало последовательности управляющих операторов нового задания.

// JOB имя задания комментарий

где имя задания — имя задания (от 1 до 8 знаков);

комментарий — пояснительный текст (например, целесообразно указать имя программиста или лица, получающего задание).

2. / & — конец задания.

Этот оператор должен быть последним в пакете операторов каждого задания.

/ & — комментарий.

Между знаком & и комментариями должно быть не менее одиннадцати пробелов.

3. EXEC — выполнить программу.

Оператор EXEC информирует программу УПРАВЛЕНИЕ ЗАДАНИЯМИ о конце управляющих операторов шага задания и указывает на необходимость начать выполнение проблемной программы. По оператору EXEC загружается первая (единственная) фаза программы и передается управление в точку входа (заданную оператором ENTRY или предполагаемую по умолчанию).

// EXEC имя фазы

где имя фазы — имя фазы из библиотеки абсолютных модулей (от 1 до 8 знаков).

Если имя фазы не указано, в основную память загружается и затем выполняется фаза, которая непосредственно перед этим была создана РЕДАКТОРОМ и временно помещена в библиотеку абсолютных модулей.

4. ASSGN — назначить логическому устройству.

Оператор ASSGN используется для назначения системным логическим устройствам или логическим устройствам программиста конкретных физических устройств.

// ASSGN SYS×××, $\left\{ \begin{matrix} X'cuu' \\ UA \\ IGN \end{matrix} \right\}, \left[\left\{ X'ss' \right\} \right]$

где SYS××× — символическое имя логического устройства.

Системные логические устройства

Имя логического устройства	Имя системного файла	Назначение	Тип физического устройства
1	2	3	4
SYSRDR	IJSYSIN	Для ввода управляющих операторов задания	Перфокарточное устройство ввода, накопитель на магнитной ленте, накопитель на магнитных дисках
SYSIPT	IJSYSIN	Для ввода потока данных (системный ввод)	Перфокарточное устройство ввода, накопитель на магнитной ленте, накопитель

1	2	3	4
SYSPCH	IJSYSPH	Для вывода информации на перфокарты (системный перфоратор)	на магнитных дисках Перфокарточное устройство вывода, накопитель на магнитной ленте, накопитель на магнитных дисках
SYSLST	IJSYSLS	Для вывода информации на печать (системная печать)	Печатающее устройство, накопитель на магнитной ленте, накопитель на магнитных дисках
SYSLOG		Для ввода директив и вывода сообщений (связь с оператором)	Пультная пишущая машинка, печатающее устройство
SYSLNK	IJSYSLN	Для ввода данных для РЕДАКТОРА	Накопитель на магнитных дисках
SYSCLB	IJSYSCL	Личная библиотека абсолютных модулей	Накопитель на магнитных дисках
SYSRLB	IJSYSRL	Личная библиотека объектных модулей	Накопитель на магнитных дисках
SYSSLB	IJSYSSL	Личная библиотека исходных модулей	Накопитель на магнитных дисках
SYSREC	IJSYSRC	Регистрационный файл, предназначенный для сбора статистики о машинных ошибках, ошибках каналов, интерфейса и устройств ввода-вывода	Накопитель на магнитных дисках

X'suu' — адрес физического устройства в шестнадцатеричном коде, где с — номер канала 0—6; uu — номер устройства в канале 00—FE;

UA — назначение для логического устройства отменяется. Попытка выполнить операцию на логическом устройстве, назначение для которого отменено по директиве с параметром UA, вызовет снятие задания;

IGN — назначение для логического устройства отменяется. Параметр IGN не допускается для логических устройств SYSRDR, SYSCLB, SYSIN и SYSIPT;

X'ss' — модификатор плотности записи на магнитную ленту: X'C8' — 32 байта/мм, X'CO' — 63 байта/мм, X'F8' — 8 байт/мм. Если операнд опущен, предполагается X'CO';

ALT — назначение сменной магнитной ленты, которая будет использоваться тогда, когда емкость основной магнитной ленты, назначенной для этого же логического устройства, будет исчерпана. Параметр ALT не допускается для логических устройств

SYSRDR, SYSIPT и SYSIN.

5. OPTION — установить режим.

Оператор OPTION устанавливает режимы работы программы УПРАВЛЕНИЕ ЗАДАНИЯМИ и некоторых обрабатывающих программ (при генерации системы устанавливаются стандартные режимы работы).

// OPTION режим, режим, режим ...

где режим — один из следующих параметров:

LOG Устанавливает режим LOG для SYSLST, при котором все директивы и операторы УПРАВЛЕНИЕ ЗАДАНИЯМИ выводятся на SYSLST.

NOLOG Отменяет режим LOG для SYSLST.

DUMP Устанавливает режим DUMP, при котором в случае ненормального завершения проблемной программы на SYSLST (если для него есть назначение) выводится содержимое общих регистров и областей основной памяти, используемых управляющей программой и разделом, в котором выполнялась проблемная программа.

NODUMP Отменяет режим **DUMP**.

LINK Режим **LINK** для трансляторов указывает, что объектный модуль должен быть помещен на **SYSLNK**. Режим **LINK** для РЕДАКТОРА и ОТЛАДКИ указывает, что отредактированная программа должна быть помещена на временное хранение в библиотеку абсолютных модулей. Режим **LINK** не допускается для разделов переднего плана, если во время генерации системы не была указана возможность работы с личной библиотекой абсолютных модулей.

NOLINK Отменяет режим **LINK**. Трансляторы могут также отменять режим **LINK**, если проблемная программа содержит ошибку, которая препятствует успешному выполнению программы.

DECK Если указан режим **DECK**, то трансляторы осуществляют вывод объектных модулей на **SYSPCH**. Если указан режим **LINK**, то режим **DECK** игнорируется трансляторами РПГ и БАЗИСНЫЙ ФОРТРАН.

NODECK Отменяет режим **DECK**.

LIST При указании этого режима трансляторы осуществляют распечатку исходного модуля на **SYSLSL**. АССЕМБЛЕР выводит на **SYSLSL** также распечатку объектного модуля в шестнадцатеричном виде и перечень всех ошибок в исходной программе. Трансляторы ФОРТРАНа дополнительно выводят на **SYSLSL** распечатку распределения памяти и сообщения об ошибках в исходной программе.

NOLIST Отменяет режим **LIST**.

LIST X Вызывает печать трансляторами ФОРТРАНа IV и ПЛ/1 объектного модуля на **SYSLSL**. Транслятор КОБОЛ при указании этого режима осуществляет вывод на **SYSLSL** объектного модуля, а также глобальных таблиц литералов и распределение регистров.

NOLIST X Отменяет режим **LIST X**.

SYM При указании этого режима АССЕМБЛЕР выводит таблицу символических имен на **SYSPCH**, трансляторы ПЛ/1 и КОБОЛ — на **SYSLSL**.

NOSYM Отменяет режим **SYM**.

XREF При указании этого режима трансляторы ПЛ/1, АССЕМБЛЕР и КОБОЛ выводят таблицу перекрестных ссылок на **SYSLSL**.

NOXREF Отменяет режим **XREF**.

ERRS При указании этого режима трансляторы БАЗИСНЫЙ ФОРТРАН, КОБОЛ и ПЛ/1 выводят список всех ошибок в исходной программе на **SYSLSL**.

NOERRS Отменяет режим **ERRS**.

CATAL Указывает, что отредактированная РЕДАКТОРОМ программа должна быть каталогизирована в библиотеку абсолютных модулей. Указание режима **CATAL** автоматически вызывает установку режима **LINK** для трансляторов. Режим **CATAL** не допускается для разделов переднего плана, если во время генерации системы не была указана возможность работы с личной библиотекой абсолютных модулей.

STDLABEL При указании этого режима вся последующая информация о метках записывается на цилиндр меток в область стандартной информации о метках файлов. Информация доступна любой программе в любом разделе до тех пор, пока не будет введен новый оператор **// OPTION STDLABEL**. Режим **STDLABEL** отменяется по концу задания или шага задания (восстанавливается режим **USRLABEL**) и при указании режима **PARSTD** или **USRLABEL**.

USRLABEL При указании этого режима вся последующая информация о метках записывается на цилиндр меток в область временной информации о метках раздела.

PARSTD При указании этого режима вся последующая информация о метках записывается на цилиндр меток в область постоянной информации о метках раздела. Информация доступна любой программе в данном разделе до тех пор, пока не будет введен новый оператор **// OPTION PARSTD**. Режим **PARSTD** отменяется по концу задания или шага задания (восстанавливается режим **USRLABEL**) и при указании режима **USRLABEL** или **STDLABEL**. Поиск информации о метках файла во время открытия файла производится в следующем порядке: область **USRLABEL**, область **PARSTD**, область **STDLABEL**.

48C Определяет 48-символьный алфавит на **SYSIPT** (для ПЛ/1).

60C Определяет 60-символьный алфавит на **SYSIPT** (для ПЛ/1).

SYSPARM='x x x x x x x x' Задаёт строку алфавитно-цифровых символов, определяющих значение параметра **& SYSPARM** для транслятора АССЕМБЛЕР. Длина строки не более восьми символов. Апостроф внутри строки представляет-

ся двумя апострофами. Операнд не допускается, если во время генерации системы не была указана эта возможность.

6. PAUSE — организовать паузу.

Этот оператор организует паузу немедленно, сразу после его ввода. Возобновляется обработка потока управляющих операторов по директиве УПРАВЛЕНИЕ ЗАДАНИЯМИ «конец текста».

// PAUSE [любые комментарии]

7. RESET — восстановить назначения.

Оператор RESET восстанавливает стандартные назначения устройств ввода-вывода

// RESET $\left\{ \begin{array}{l} \text{SYS} \\ \text{PROG} \\ \text{ALL} \\ \text{SYS} \times \times \times \end{array} \right\}$

где SYS — восстанавливает стандартные или постоянные назначения всех системных логических устройств;

PROG — восстанавливает стандартные или постоянные назначения всех логических устройств программиста;

ALL — восстанавливает стандартные или постоянные назначения всех логических устройств.

SYS × × × — восстанавливает стандартное или постоянное назначение указанного логического устройства.

8. DLBL — запомнить информацию о метках файла на дисках.

Оператор DLBL сообщает информацию, необходимую для создания или проверки меток файла на дисках.

// DLBL имя файла, ['идентификатор файла'], [дата], [код]

где имя файла — тождественно имени файла, заданному в операторе DECLARE;

идентификатор файла — идентификатор, который присваивается файлу, содержит от 1 до 44 алфавитно-цифровых символов;

дата — определяет срок хранения файла и может задаваться в одной из двух форм: в виде числа, содержащего от 1 до 4 цифр и указывающего количество дней хранения файла (от 0 до 9999); в виде абсолютной даты истечения срока хранения файла, содержащей последние две цифры года и номер дня в году (ГГ/ДДД); если операнд опущен, срок хранения принимается равным 7 дням;

код — двух- или трехзнаковое поле, идентифицирующее тип организации файла;

SD — последовательная организация;

DA — произвольная организация;

ISC — создание индексно-последовательного файла;

ISE — обработка индексно-последовательного файла (расширение, добавление, извлечение);

Если параметр «код» опущен, то принимается код SD.

9. EXTENT — запомнить информацию об участке файла на дисках.

Этот оператор сообщает информацию об участке, занимаемом файлом на дисках. Один или несколько операторов EXTENT должны следовать непосредственно за оператором DLBL.

// EXTENT [SYS × × ×], [регистрационный номер], [тип],

[номер участка], [адрес участка],

[количество дорожек], [дорожка разделения цилиндра]

где SYS × × × — символическое имя логического устройства, которому назначен накопитель на магнитных дисках. Если этот параметр опущен, то берется имя, указанное в предшествующем операторе EXTENT;

регистрационный номер — регистрационный номер пакета дисков (тома), на котором находится описываемый в EXTENT участок (от 1 до 6 знаков). Если этот параметр опущен, то используется номер, указанный в предшествующем операторе EXTENT. Если этот параметр опущен во всех операторах EXTENT, то во время открытия файла не производится контроль на правильность установки нужного тома и потребитель сам несет ответственность за использование неверного носителя;

тип — тип участка. Этот параметр записывается в виде цифр 1, 2, 4 или 8, где 1 — определяет участок, предназначенный для размещения данных на неразделенных цилиндрах; 2 — определяет участок, предназначенный для области переполнения (для индексно-последовательного файла); 4 — определяет участок, предназначенный для размещения индекса (для индексно-последовательного файла); 8 — определяет участок, предназначенный для размещения данных на разделенных цилиндрах (последовательный файл).

Если этот параметр опущен, то предполагается 1;

адрес участка — от 1 до 5 цифр, определяющих относительный номер первой дорожки участка (относительно нулевой дорожки на нулевом цилиндре). Нумерация ведется относительно дорожки 0 на цилиндре 0. Для индексно-последовательного и произвольного файлов указание адреса участка обязательно. Для вводного файла последовательного доступа этот операнд может быть опущен (в этом случае адрес участка берется из метки файла). Формула для подсчета адреса участка $K \cdot n + h$

где $K = \begin{cases} 10 & \text{для дисков типа EC-5052;} \\ 20 & \text{для дисков типа EC-5061;} \end{cases}$

n — номер цилиндра;

h — номер головки;

количество — количество дорожек, занимаемых участком (от 1 до 5 цифр). Для последовательного вводного файла этот параметр можно опустить. Для файлов, разделяющих цилиндры с другими файлами, количество дорожек представляет произведение количества цилиндров, отведенных файлу, на количество дорожек, занимаемых этим файлом на каждом из цилиндров;

дорожка разделения цилиндра — номер последней из дорожек (1—2 десятичные цифры), занимаемых участком на каждом из цилиндров в случае разделения цилиндров. Номер первой дорожки этого участка берется из адреса участка.

10. TLBL — напомнить информацию о метках файла на магнитной ленте.

Оператор TLBL сообщает информацию, которая необходима для создания или проверки меток файлов на магнитной ленте.

// TLBL имя файла, ['идентификатор файла'], [дата],

[регистрационный номер файла], [порядковый номер тома], [порядковый номер файла], [номер поколения], [номер версии поколения]

где имя файла — идентично имени файла, объявленному в операторе DECLARE;

идентификатор файла — идентификатор, который связан с именем файла на носителе данных и содержит от 1 до 17 алфавитно-цифровых символов. Если этот операнд опущен, то для выводного файла в качестве идентификатора используется имя файла, для вводного файла проверка идентификатора не производится. Для вводных файлов идентификатор сравнивается с информацией в метке ленты;

дата — для выводного файла указывается срок хранения файла в днях (0—9999, а для вводного — дата создания в форме ГГ/ДДД (ГГ — год, ДДД — день). Если этот параметр опущен, срок хранения принимается равным нулю. Для вводных файлов в метку заносится текущая дата, полученная во время первоначальной загрузки. Если параметр опущен, проверка даты создания файла не производится;

регистрационный номер файла — архивный номер первой катушки файла, расположенного на нескольких катушках (от 1 до 6 знаков). Если этот параметр опущен для выводного файла, будет использован регистрационный номер первой катушки файла. Если параметр опущен для вводного файла, контроль регистрационного номера не производится;

порядковый номер тома — порядковый номер тома многотомного файла (от 1 до 4 двоичных цифр). При переключении на следующую ленту этот номер автоматически увеличивается. Если параметр опущен для выводного файла, его значение принимается равным 0001, если параметр опущен для вводного файла, контроль номера тома не производится;

порядковый номер файла — порядковый номер файла на многофайловом томе (от 1 до 4 двоичных цифр). В случае выводных файлов он увеличивается автоматически при выполнении процедур OPEN/CLOSE. Если параметр опущен для выводного файла, значение принимается равным 0001 (двоичное). Если параметр опущен для вводного файла, контроль порядкового номера не производится;

номер поколения — определяет дополнительную информацию к идентификатору файла (от 1 до 4 цифр). Если этот параметр опущен для выводного файла, его значение принимается равным 0001 (двоичное). Если параметр опущен для вводного файла, контроль номера поколения не производится;

номер версии поколения — определяет дополнительную информацию к номеру поколения (от 1 до 2 цифр). Если параметр опущен для выводного файла, его значение принимается равным 01 (двоичное). Если параметр опущен для вводного файла, контроль номера версии поколения не производится.

11. LBLTYP — резервировать память для информации о метках.

Этот оператор определяет объем основной памяти, которая должна быть отведена РЕДАКТОРОМ в области проблемной программы для обработки меток файла.

$$// \text{LBLTYP} \left\{ \begin{array}{l} \text{NSD (nn)} \\ \text{TAPE} \end{array} \right\}$$

где NSD(nn) — задается, если должна быть зарезервирована одна или несколько областей меток для индексно-последовательного файла или файла с прямым доступом; nn — максимальное количество участков, используемых для одного такого файла. В этом случае резервируется $(84 + (20 \cdot nn))$ байт. Если указан параметр TAPE, то резервируется 80 байтов.

Область меток необходима только тогда, когда в фазе обрабатываются ленточные файлы с метками, индексно-последовательные и региональные (с прямым доступом) файлы на дисках. Если такие файлы в программе не используются, оператор LBLTYP не нужен. Если оператор LBLTYP необходим, он должен непосредственно предшествовать оператору // EXEC LNKEDT. Это приводит к тому, что ад рес загрузки фазы увеличивается на длину области меток.

12. MTC — выполнять служебные операции на магнитной ленте.

С помощью оператора MTC на магнитной ленте выполняются некоторые служебные операции.

$$// \text{MTC код операции} \left\{ \begin{array}{l} \text{X'сш' } \\ \text{SYS} \times \times \times \end{array} \right\} [, nn]$$

где код операции определяет вид служебной операции, выполняемой на магнитной ленте:

BSF — шаг на файл назад, BCR — шаг на блок назад,
ERG — стереть промежуток, FSF — шаг на файл вперед,
FSR — шаг на блок вперед, RUN — перемотать и разгрузить,
REW — перемотать, WTM — записать марку.

13. DATE — указать текущую дату.

Оператор DATE указывает текущую дату, действительную только для задания, в котором встретился этот оператор.

$$// \text{DATE} \left\{ \begin{array}{l} \text{мм/дд/гг} \\ \text{дд/мм/гг} \end{array} \right\}$$

где дд — день (01—31); мм — месяц (01—12); гг — год (00—99).

14. UPSI — установить значения программных переключателей.

Оператор UPSI устанавливает значения битов в байте программных переключателей UPSI в области связи. Он имеет вид

// UPSI строка символов

где строка символов содержит от одного до восьми символов 0, 1 или X. Каждый символ соответствует одному биту в байте UPSI: первый — биту 0, второй — биту 1 и т.д. Бит, которому соответствует символ 0, устанавливается в 0; бит, которому соответствует символ 1, устанавливается в 1; бит, которому соответствует символ X, не изменяется. Если operand содержит меньше восьми символов, состояние оставшихся крайних справа бит в байте не изменяется.

15. LOG — печать управляющих операторов и директив.

// LOG

Все выводимые операторы и директивы выдаются на SYSLOG.

16. NOLOG — отменить режим LOG.

Он имеет вид

// NOLOG

17. LISTIO — распечатать таблицу назначений логических устройств.

//LISTIO операнд

где операнд может принимать значения: SYS, PROG, F1, F2, ALL, SYS×××, UNITS, DOWN, UA, X'cuu' для директивы и оператора Управления заданиями и значения: BG, F1, F2, UA, ALL для директивы Инициатора одиночных программ. Каждое из значений определяет характер печатаемой информации.

Значение	Печатаемая информация
SYS	Таблица назначений для всех системных логических устройств раздела
PROG	Таблица назначений для всех логических устройств программиста в разделе
BG	Таблица назначений для всех логических устройств фонового раздела
F1	Таблица назначений для всех логических устройств первого раздела переднего плана
F2	Таблица назначений для всех логических устройств второго раздела переднего плана
ALL	Таблица назначений для всех логических устройств всех разделов
SYS×××	Назначение для указанного операндом логического устройства в разделе
UNITS	Таблица всех физических устройств, которые назначены в данный момент логическим устройством
DOWN	Список всех физических устройств, использование которых запрещено
UA	Список всех физических устройств, которые в момент выдачи директивы или оператора LISTIO не назначены ни одному из логических устройств
X'cuu'	Список всех логических устройств, которым назначено физическое устройство с адресом

18. RSTRT — запустить программу с контрольной точки.

Имя задания, указанное в операторе JOB, должно быть идентичным имени задания, используемому при создании контрольной точки. Оператор имеет вид

//RSTRT SYS×××, nnnn [, имя файла]

где SYS××× — логическое устройство, на котором расположен файл контрольных точек;

nnnn — номер контрольной точки, с которой производится запуск программы.

Номера контрольных точек печатаются на SYSLST во время их создания;

имя файла — имя файла контрольных точек. Этот операнд указывается в том случае, если информация контрольной точки записана на диски. Имя файла должно быть идентичным имени файла, используемому в макрокоманде DTFPH, описывающей файл контрольных точек на дисках, и в макрокоманде CHKRT.

19. CLOSE — закрыть выводные файлы на магнитной ленте.

// CLOSE SYS×××

$$\left[\begin{array}{l} , X'cuu' [,X'ss'] \\ , UA \\ , IGN \\ , ALT \end{array} \right]$$

где $SYS \times \times \times$ — логическое устройство, на котором расположены файлы, подлежащие закрытию;

$X'suu'$ — адрес физического устройства, которое после закрытия назначается для логического устройства;

$X'ss'$ — модификатор плотности записи на магнитную ленту. Он аналогичен операнду $X'ss'$ в операторе (директиве) ASSGN;

UA — указывает, что после закрытия назначение логическому устройству отменяется;

IGN — после закрытия назначение логическому устройству отменяется и все последующие обращения к этому устройству игнорируются;

ALT — после закрытия основной магнитной ленты, назначенной для логического устройства, должна быть открыта сменная магнитная лента (если она назначена логическому устройству). Этот операнд допускается только для логических устройств SYSLST, SYSPCH, SYSOUT.

А С С Е М Б Л Е Р

СИСТЕМА КОМАНД ЕС ЭВМ

Выполнить	EX	44 RX R1, D2 (X2, B2)
Execute		
Вычитание	SR	1B RR R1, R2
Subtract (c)		
Вычитание	S	5B RX R1, D2 (X2, B2)
Subtract (c)		
Вычитание без нормализации (длинное)	SWR	2F RR R1, R2
Subtract Unnormalized, Long (c)		
Вычитание без нормализации (длинное)	SW	6F RX R1, D2 (X2, B2)
Subtract Unnormalized, Long (c)		
Вычитание без нормализации (короткое)	SUR	3F RR R1, R2
Subtract Unnormalized, Short (c)		
Вычитание без нормализации (короткое)	SU	7F RX R1, D2 (X2, B2)
Subtract Unnormalized, Short (c)		
Вычитание с нормализацией (длинное)	SD	6B RX R1, D2 (X2, B2)
Subtract Normalized, Long (c)		
Вычитание с нормализацией (длинное)	SDR	2B RR R1, R2
Subtract Normalized, Long (c)		
Вычитание с нормализацией (короткое)	SER	3B RR R1, R2
Subtract Normalized, Short (c)		
Вычитание с нормализацией (короткое)	SE	7B RX R1, D2 (X2, B2)
Subtract Normalized, Short (c)		
Вычитание с нормализацией (сверхдлинное)	SXR	37 RR R1, R2
Subtract Normalized, Extended (c, x)		
Вычитание десятичное	SP	FB SS D1 (L1, B1), D2 (L2, B2)
Subtract Decimal (c, d)		
Вычитание кодов	SLR	1F RR R1, R2
Subtract Logical (c)		
Вычитание кодов	SL	5F RX R1, D2 (X2, B2)
Subtract Logical (c)		
Вычитание полуслова	SH	4B RX R1, D2 (X2, B2)
Subtract Halfword (c)		
Деление	DR	1D RR R1, R2
Divide		
Деление	D	5D RX R1, D2 (X2, B2)
Divide		
Деление десятичное	DP	FD SS D1 (L1, B1), D2 (L2, B2)
Divide Decimal (d)		
Деление (длинное)	DDR	2D RR R1, R2
Divide, Long		
Деление (длинное)	DD	6D RX R1, D2 (X2, B2)
Divide, Long		
Деление (короткое)	DER	3D RR R1, R2
Divide, Short		

Деление (короткое)	DE	7D RX R1, D2 (X2, B2)
Divide, Short		
Диагностика		83 SI
Diagnose		
Загрузка	L	58 RX R1, D2 (X2, B2)
Load		
Загрузка	LR	18 RR R1, R2
Load		
Загрузка адреса	LA	41 RX R1, D2 (X2, B2)
Load Address		
Загрузка групповая	LM	98 RS R1, R3, D2 (B2)
Load Multiple		
Загрузка групповая регистров управления	LMC	B8 RS R1, R3, D2 (B2)
Load Multiple Control (e, p)		
Загрузка действительного адреса	LRA	B1 RX R1, D2 (X2, B2)
Load Real Address (c, e, p)		
Загрузка (длинная)	LDR	28 RR R1, R2
Load, Long		
Загрузка (длинная)	LD	68 RX R1, D2 (X2, B2)
Load, Long		
Загрузка дополнения	LCR	13 RR R1, R2
Load Complement (c)		
Загрузка дополнения (длинная)	LCDR	23 RR R1, R2
Load Complement, Long (c)		
Загрузка дополнения (короткая)	LCER	33 RR R1, R2
Load Complement, Short (c)		
Загрузка и проверка	LTR	12 RR R1, R2
Load and Test (c)		
Загрузка и проверка (длинная)	LTDR	22 RR R1, R2
Load and Test, Long (c)		
Загрузка и проверка (короткая)	LTERR	32 RR R1, R2
Load and Test, Short (c)		
Загрузка (короткая)	LER	38 RR R1, R2
Load, Short		
Загрузка (короткая)	LE	78 RX R1, D2 (X2, B2)
Load, Short		
Загрузка отрицательная	LNR	11 RR R1, R2
Load Negative (c)		
Загрузка отрицательная (длинная)	LNDR	21 RR R1, R2
Load Negative, Long (c)		
Загрузка отрицательная (короткая)	LNERR	31 RR R1, R2
Load Negative, Short (c)		
Загрузка положительная	LPR	10 RR R1, R2
Load Positive (c)		
Загрузка полуслова	LH	48 RX R1, D2 (X2, B2)
Load Halfword		
Загрузка	LPSW	82 SI D1 (B1)
PSW Load PSW (n, p)		
Загрузка положительная (длинная)	LPDR	20 RR R1, R2
Load Positive, Long (c)		
Загрузка положительная (короткая)	LPER	30 RR R1, R2
Load Positive, Short (c)		
Загрузка с округлением (длинная)	LRDR	25 RR R1, R2
Load Rounded, Long (x)		
Загрузка с округлением (короткая)	LRERR	35 RR R1, R2
Load Rounded, Short (x)		
Запись в память групповая	STM	90 RS R1, R3, D2 (B2)
Store Multiple		
Запись в память	ST	50 RX R1, D2 (X2, B2)
Store		
Запись в память групповая регистров		

управления	STMC	B0 RS R1, R3, D2 (B2)
Store Multiple Control (с, р)		
Запись в память (длинная)	STD	60 RX R1, D2 (X2, B2)
Store, Long		
Запись в память (короткая)	STE	70 RX R1, D2 (X2, B2)
Store, Short		
Запись в память полуслова	STH	40 RX R1, D2 (X2, B2)
Store Halfword		
Запись в память символа	STC	42 RX R1, D2 (X2, B2)
Store Character	NR	14 RR R1 R2
И AND (с)	N	54 RX R1, D2 (X2, B2)
И AND (с)		
И непосредственное	NI	94 SI D1 (B1), I2
And Immediate (с)	NC	D4 SS D1 (L, B1), D2 (B2)
И AND (с)	OR	16 RR R1, R2
ИЛИ Or (с)	O	56 RX R1, D2 (X2, B2)
ИЛИ Or (с)		
ИЛИ непосредственное	OI	96 SI D1 (B1), I2
Or Immediate (с)	OC	D6 SS D1 (L, B1), D2, (B2)
ИЛИ Or (с)		
Исключающее ИЛИ	XR	17 RR R1, R2
Exc usive Or (с)		
Исключающее ИЛИ	X	57 RX R1, D2 (X2, B2)
Exclusive Or (с)		
Исключающее ИЛИ непосредственное	XI	97 SI D1 (B1), I2
Exclusive Or Immediate		
Исключающее ИЛИ	XC	D7 SS D1 (L, B1), D2 (B2)
Exclusive Or (с)		
Начать ввод-вывод	SIO	9C SI D1 (B1)
Start I/O (с, р)		
Обращение к супервизору	SVC	0A RR I
Supervisor Call		
Опросить ввод-вывод	TIO	9D SI D1 (B1)
Test I/O (с, р)		
Опросить канал	TCH	9F SI D1 (B1)
Test Channel (с, р)		
Остановить ввод-вывод	HIO	9E SI D1 (B1)
Halt I/O (с, р)		
Отредактировать	ED	DE SS D1 (L, B1), D2 (B2)
Edit (с, d)		
Отредактировать и отменить	EDMK	DF SS D1 (L, B1), D2 (B2)
Edit and Mark		
Пересылка непосредственная	MVI	92 SI D1 (B1), I2
Move Immediate		
Пересылка символов	MVC	D2 SS D1 (L, B1), D2 (B2)
Move Characters		
Пересылка зон	MVZ	D3 SS D1 (L, B1), D2 (B2)
Move Zones		
Пересылка со сдвигом	MVO	F1 SS D1 (L1, B1), D2 (L2, B2)
Move with Offset		
Пересылка цифр	MVN	D1 SS D1 (L, B1), D2 (B2)
Move Numerics		
Перекодировать и проверить	TRT	DD SS D1 (L, B1), D2 (B2)
Translate and Test (с)		
Перекодировать	TR	DC SS D1 (L, B1), D2 (B2)
Translate		
Переход с возвратом	BALR	05 RR R1, R2
Branch and Link		
Переход с возвратом	BAL	45 RX R1, D2 (X2, B2)
Branch and Link		
Переход с записью в память		

Branch and Store (c)	BASR	0D RR R1, R2
Переход с записью в память		
Branch and Store (c)	BAS	4D RX R1, D2 (X2, B2)
Переход по индексу больше		
Branch on Index High	BXH	86 RS R1, R3, D2 (B2)
Переход по индексу меньше или равно		
Branch on Index Low or Equal	BXLE	87 RS R1, R3, D2 (B2)
Переход по счетчику		
Branch on Count	BCTR	06 RR R1, R2
Переход по счетчику		
Branch on Count	BCT	46 RX R1, D2 (X2, B2)
Пополом (длинная)	HDR	24 RR R1, R2
Halve, Long		
Пополом (короткая)	HER	34 RR R1, R2
Halve, Short		
Преобразование в двоичную		
Convert to Binary	CVB	4F RX R1, D2 (X2, B2)
Преобразование в десятичную		
Convert to Decimal	CVD	4E RX R1, D2 (X2, B2)
Проверить по маске		
Test under Mask (c)	TM	91 SI D1 (B1), I2
Проверить и установить		
Test and Set (c)	TS	93 SI D1 (B1)
Прочитать ключ памяти		
Insert Storage Key (a, p)	ISK	09 RR R1, R2
Прочитать символ		
Insert Character	IC	43 RX R1, D2 (X2, B2)
Прямая запись		
Write Direct (b, p)	WRD	84 SI D1 (B1), I2
Прямое чтение		
Read Direct (h, p)		
Распаковать Unpack	RDD	B5 SI D1 (B1), I2
Сдвиг влево арифметический	UNPK	F3 SS D1 (L1, B1), D2 (L2, B2)
Shift Left Arithmetic		
Сдвиг влево двойной арифметический	SLA	8B RS R1, D2 (B2)
Shift Left Double Arithmetic		
Сдвиг влево двойной кода	SLDA	8F RS R1, D2 (S2)
Shift Left Double Logical		
Сдвиг влево кода	SLDL	8D RS R1, D2 (S2)
Shift Left Single Logical		
Сдвиг вправо арифметический	SLL	89 RS R1, D2 (B2)
Shift Right Single Arithmetic		
Сдвиг вправо двойной арифметический	SRA	8A RS R1, D2 (B2)
Shift Right Double Arithmetic		
Сдвиг вправо двойной кода	SRDA	8E RS R1, D2 (B2)
Shift Right Double Logical		
Сдвиг вправо кода	SRDL	8C RS R1, D2 (B2)
Shift Right Logical		
Сложение Add (c)	SRL	88 RS R1, D2 (B2)
Сложение Add (c)	AR	1A RR R1, R2
Сложение без нормализации (длинное)	A	5A RX R1, D2 (X2, B2)
Add Unnormalized, Long (c)		
Сложение без нормализации (длинное)	AWR	2E RR R1, R2
Add Unnormalized, Long (c)		
Сложение без нормализации (короткое)	AW	6E RX R1, D2 (X2, B2)
Add Unnormalized, Short (c)		
Сложение без нормализации (короткое)	AUR	3E RR R1, R2
Add Unnormalized, Short (c)		
Сложение десятичное	AU	7E RX R1, D2 (X2, B2)
Add Decimal (c, d)		
Сложение (длинное)	AP	FA SS D1 (L1, B1), D2 (L2, B2)

Add Normalized, Long (c)	AD	6A RX R1, D2 (X2, B2)
Сложение кодов		
Add Logical (c)	ALR	1E RR R1, R2
Сложение кодов		
Add Logical (c)	AL	5E RX R1, D2 (X2, B2)
Сложение полуслова		
Add Halfword	AH	4A RX R1, D2 (X2, B2)
Сложение с нормализацией (длинное)		
Add Normalized, Long (c)	ADR	2A RR R1, R2
Сложение с нормализацией (короткое)		
Add Normalized, Short (c)	AER	3A RR R1, R2
Сложение с нормализацией (короткое)		
Add Normalized, Short (c)	AE	7A RX R1, D2 (X2, B2)
Сложение с нормализацией (сверхдлинное)		
Add Normalized, Extend (c, x)	AXR	36 RR R1, R2
Сложение с очисткой		
Zero and Add	ZAP	F8 SS D1 (L1, B1), D2 (L2, B2)
Сравнение Compare (c)	CR	19 RR R1, R2
Сравнение Compare (c)	G	59 RX R1, D2 (X2, B2)
Сравнение десятичное		
Compare Decimal (c, d)	CP	F9 SS D1 (L1, B1), D2 (L2, B2)
Сравнение непосредственное		
Compare Immediate	CLI	95 SI D1 (D2), I2
Сравнение (длинное)		
Compare, Long	CD	69 RX R1, D2 (X2, B2)
Сравнение (длинное)		
Compare, Long	CDR	29 RR R1, R2
Сравнение кодов		
Compare, Logical (c)	CLR	15 RR R1, R2
Сравнение кодов		
Compare, Logical (c)	CL	55 RX R1, D2 (X2, B2)
Сравнение кодов		
Compare, Logical (c)	CLG	D5 SS D1 (L, B1), D2 (B2)
Сравнение (короткое)		
Compare, Short (c)	CER	39 RR R1, R2
Сравнение (короткое)		
Compare, Short (c)	CE	79 RX R1, D2 (X2, B2)
Сравнение полуслова		
Compare Halfword (c)	CH	49 RX R1, D2 (X2, B2)
Умножение		
Multiply	MR	1C RR R1, R2
Умножение		
Multiply	M	5C RX R1, D2 (X2, B2)
Умножение десятичное		
Multiply Decimal (d)	MP	FC SS D1 (L1, B1), D2 (L2, B2)
Умножение (длинное)		
Multiply, Long	MDR	2C RR R1, R2
Умножение (длинное)		
Multiply, Long	MD	6C RX R1, D2 (X2, B2)
Умножение длинных со сверхдлинным результатом		
Multiply to Extend (x)	MXDR	27 RR R1, R2
Умножение длинных со сверхдлинным результатом		
Multiply to Extend	MXD	67 RX R1, D2 (X2, B2)
Умножение (короткое)		
Multiply, Short	MER	3C RR R1, R2
Умножение (короткое)		
Multiply, Short	ME	7C RX R1, D2 (X2, B2)
Умножение сверхдлинное		