

НАУЧНО-ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ "СЕМИКО"

40 1340

Клавишные электронно-вычислительные машины

ЭЛЕКТРОНИКА МК

ОРГАНИЗАЦИЯ РАБОТЫ С ФУНКЦИЯМИ,
АДРЕСУЕМЫМИ ЧЕРЕЗ РЕГИСТРЫ ПАМЯТИ

ГРАФИЧЕСКИЙ ИНТЕРФЕЙС ПОЛЬЗОВАТЕЛЯ,
РАБОТА С ТАБЛИЦАМИ

часть 4 НПКД.401348.001 Д1.3 изм. 22

Новосибирск

2017

Содержание

16. ГРАФИЧЕСКИЙ ИНТЕРФЕЙС ПОЛЬЗОВАТЕЛЯ.....	4
16.1. Описание.....	4
16.2. Структура интерфейса.....	4
16.2.1. Главная таблица меню.....	4
16.2.2. Структура экранной формы.....	5
16.3. Регистры интерфейса.....	8
16.3.1. Регистр разрешения интерфейса (R9200).....	8
16.3.2. Адрес главной таблицы меню (R9201).....	8
16.3.3. Указатели экранной формы (R9202, R9203).....	9
16.3.4. Регистр действий экранной формы (R9204).....	9
16.3.5. Период автоматического обновления экрана (R9206).....	9
16.3.6. Адрес регистра для ввода чисел (R9207).....	10
16.3.7. Регистр команд интерфейса (R9208).....	10
16.3.8. Регистр результата выполнения операции (R9209).....	10
16.4. Работа интерфейса и действие команд.....	11
16.4.1. Работа интерфейса.....	11
16.4.2. Команды ВЫБОР.....	12
16.4.3. Команда ВВОД.....	13
16.4.4. Команда ВЫХОД.....	14
16.4.5. Переход из одной экранной формы в другую.....	14
16.5. Команды языка интерфейса.....	14
16.5.1. Описание.....	14
16.5.2. Перечень команд.....	15
16.5.3. Установка начальной позиции вывода (TAB).....	16
16.5.4. Загрузка начальной позиции вывода по X из памяти десятичных данных (LDX).....	16
16.5.5. Загрузка начальной позиции вывода по Y из памяти десятичных данных (LDY).....	17
16.5.6. Установка номера шрифта (TYPE).....	17
16.5.7. Загрузка адреса загружаемого шрифта (ATYPE).....	17
16.5.8. Загрузка адреса регистра для ввода числа (LDADR).....	17
16.5.9. Вывод прямоугольника (SQU).....	18
16.5.10. Вывод рамки (FRAME).....	18
16.5.11. Вывод графического образа из памяти программ (DRAWP).....	18
16.5.12. Вывод графического из памяти двоичных данных (DRAWB).....	18
16.5.13. Вывод строки символов из памяти программ (TXTP).....	18
16.5.14. Вывод строки символов из памяти двоичных данных (TXTB).....	19
16.5.15. Вывод строки символов с косвенной адресацией (TXTK).....	19
16.5.16. Вывод числа с естественной или плавающей запятой (PREXP)....	19
16.5.17. Вывод числа в форматированном виде (PRFORM).....	20
16.5.18. Вывод числа в шестнадцатичном виде (PRHEX).....	20

16.5.19. Переход к подпрограмме в текущей форме (MCALL).....	20
16.5.20. Возврат из подпрограммы в текущей форме (RETUM).....	20
16.5.21. Переход к подпрограмме в памяти программ (LCALL).....	21
16.5.22. Возврат из подпрограммы в памяти программ (RETUL).....	21
16.5.23. Переход если число не равно 0 (JNEZ).....	21
16.5.24. Безусловный переход (JM).....	21
16.5.25. Окончание программы (END).....	21
16.6. Экранные формы встроенного программного обеспечения ЭВМ.....	22
16.6.1. Экранная форма ввода десятичного числа.....	22
16.6.2. Экранная форма ввода шестнадцатиричного числа.....	22
17. РАБОТА С ТАБЛИЦАМИ.....	23
17.1. Описание.....	23
17.2. Функции для работы с таблицами.....	24
17.2.1. Чтение значения по номеру (R9210).....	24
17.2.2. Чтение двоично-десятичного значения по номеру (R9211).....	24
17.2.3. Поиск значения по индексу (R9212).....	25
17.2.4. Поиск двоично-десятичного значения по индексу (R9213).....	26

16. ГРАФИЧЕСКИЙ ИНТЕРФЕЙС ПОЛЬЗОВАТЕЛЯ

16.1. Описание

В ЭВМ реализована возможность организации пользователем графического интерфейса, состоящего из связанных экранных форм, представляющих собой систему меню. Каждая экранная форма может содержать графические объекты (изображения, прямоугольники, рамки), строки текста и числа, расположенные на графическом экране в произвольных местах. Интерфейс обеспечивает выбор активного элемента в текущей экранной форме, переход из одной экранной формы в другую, вывод из памяти ЭВМ текстовых строк и числовых значений, а также ввод в память данных ЭВМ десятичных и шестнадцатиричных чисел.

Экранные формы расположены в памяти программ ЭВМ и выполнены в виде компактных структур. Все экранные формы пронумерованы. Связь между структурами экранных форм осуществляет главная таблица меню, в которой номеру экранной формы сопоставлен адрес начала соответствующей структуры в памяти программ.

Функции графического интерфейса выполняются только во время выполнения программы пользователя и не работают в режиме пошагового прохода.

16.2. Структура интерфейса

16.2.1. Главная таблица меню

Главная таблица меню размещается в памяти программ ЭВМ и состоит из последовательных записей длиной по 3 байта. Первый байт записи - номер экранной формы от 0 до 252 (0FCh), два других байта - адрес начала таблицы экранной формы в BCD формате (от 00h, 00h до 99h, 99h). Заканчивается главная таблица меню байтом с кодом 0FFh.

Система меню может содержать до 253 запрограммированных пользователем экранных форм. Экранные формы с номерами 253 (0FDh) и 254

(0FEh) используются интерфейсом для ввода чисел в шестнадцатичном и десятичном виде соответственно. Они включены во встроенное программное обеспечение ЭВМ и не требуют отдельного программирования пользователем.

16.2.2. Структура экранной формы

16.2.2.1. Структура экранной формы состоит из заголовка, таблицы флагов действий, таблицы переходов по команде ВВОД, таблицы переходов по команде ВЫХОД, таблицы переходов по команде ВЫБОР ВВЕРХ, таблицы переходов по команде ВЫБОР ВНИЗ, таблицы адресов программ элементов, основной программы экранной формы и программ элементов.

Адреса таблиц в заголовке, адреса программ в таблицах и адреса переходов в командах языка интерфейса занимают один байт и являются смещением относительно начала структуры экранной формы.

16.2.2.2. Заголовок структуры содержит 8 байт, расположенных в следующей последовательности:

- J - максимальный номер элемента в экранной форме;
- DE - адрес таблицы флагов действий;
- PI - адрес таблицы переходов по команде ВВОД;
- PO - адрес таблицы переходов по команде ВЫХОД;
- PL - адрес таблицы переходов по команде ВЫБОР ВЛЕВО;
- PR - адрес таблицы переходов по команде ВЫБОР ВПРАВО;
- RI - адрес основной программы экранной формы;
- KM - адрес таблицы адресов программ элементов.

Максимальный номер элемента J принимает значение на единицу меньшее общего числа элементов, которые могут быть активными в данной экранной форме, поскольку нумерация элементов начинается с нулевого номера.

Расположение таблиц и программ внутри структуры экранной формы определяется указанными в заголовке адресами и может быть произвольным.

16.2.2.3. Таблица флагов действий (DE) имеет формат:

FLDE0, FLDE1, FLDE2, ... FLDEJ

Таблица состоит из строки байт по числу элементов и ставит в соответствие каждому элементу экранной формы байт флагов действий.

Назначение битов:

бит 0 - разрешение команд ВЫБОР;

бит 1 - признак выбора;

бит 2 - разрешение команды ВВОД;

бит 3 - признак ввода;

бит 4 - разрешение команд ВЫХОД;

бит 5 - признак выхода;

бит 6 - разрешение периодического обновления экрана;

бит 7 - запрет очистки экрана.

Если в байте флагов действий активного элемента установлено разрешение команды, то интерфейс воспринимает соответствующую команду, если не установлено - игнорирует.

Установленный бит разрешения периодического обновления вызывает автоматическую перерисовку экранной формы с периодом устанавливаемым пользователем (см. п. 16.3.5) когда соответствующий элемент является активным.

Установленный бит запрета очистки экрана в байте активного элемента запрещает производить инициализацию графического экрана перед перерисовкой экранной формы.

16.2.2.4. Таблицы переходов по командам ВВОД и ВЫХОД имеют одинаковый формат:

MEOUT0, ELOUT0, MEOUT1, ELOUT1, ... MEOUTJ, ELOUTJ

Каждая таблица ставит в соответствие каждому элементу загруженной экранной формы два следующих подряд байта - номер новой экранной формы и номер активного элемента в новой экранной форме.

В форму с установленным номером и активным элементом произойдёт переход при получении команд ВВОД или ВЫХОД, когда соответствующий элемент является активным.

16.2.2.5. Таблицы переходов по командам ВЫБОР ВЛЕВО и ВЫБОР ВПРАВО имеют одинаковый формат:

ELR0, ELR1, ... ELRJ

Каждая таблица ставит в соответствие каждому элементу загруженной экранной формы один байт - номер нового активного элемента загруженной формы.

При получении команд ВЫБОР ВЛЕВО или ВЫБОР ВПРАВО произойдёт соответствующее переключение активного элемента в загруженной экранной форме.

16.2.2.6. Основная программа экранной формы представляет собой программу написанную на языке графического интерфейса пользователя (см. п. 16.5). Основная программа всегда выполняется при загрузке и каждой перерисовке экранной формы, в том числе с заданной пользователем частотой при установленном разрешении периодического обновления экрана.

16.2.2.7. Таблица адресов программ элементов имеет формат:

ADRK0, ADRK1, ... ADRKJ

Таблица ставит в соответствие каждому элементу загруженной экранной формы один байт - адрес программы элемента. Адресом является смещение относительно начала структуры.

При загрузке и каждой перерисовке экранной формы после выполнения основной программы будет выполнена программа, соответствующая активному элементу.

16.2.2.8. Программы элементов являются частью структуры экранной формы. При загрузке и каждой перерисовке формы выполняется программа, соответствующая активному элементу. Адрес программы выбирается из таблицы адресов (см. п. 16.2.2.7).

16.3. Регистры интерфейса

16.3.1. Регистр разрешения интерфейса (R9200)

Регистр доступен при обращении к регистру памяти с адресом 9200. При записи числа 0 функции интерфейса запрещены, при записи числа 1 - разрешены. Запись регистра возможна только при выполнении программы пользователя, при остановке программы разрешение автоматически сбрасывается.

В режиме автоматической работы ЭВМ регистр разрешения интерфейса всегда содержит число 0.

16.3.2. Адрес главной таблицы меню (R9201)

Регистр памяти с адресом 9201 содержит число от 0 до 9999, соответствующее адресу начала главной таблицы меню в памяти программ. Регистр доступен по записи при разрешении функций интерфейса, по чтению - во всех режимах.

16.3.3. Указатели экранной формы (R9202, R9203)

В качестве указателей экранной формы используется два числа - это номер экранной формы, записываемый в регистр памяти с адресом 9202 и номер активного элемента в экранной форме, записываемый в регистр памяти с адресом 9203. Указатели могут принимать целые значения от 0 до числа, не превышающего 255. Они доступны по чтению и записи при разрешении функций интерфейса, в автоматическом режиме оба числа имеют значение 0.

При выполнении команд интерфейса в регистры указателей автоматически записываются номер загруженной экранной формы и номер активного элемента этой экранной формы.

16.3.4. Регистр действий экранной формы (R9204)

Значение регистра памяти с адресом 9204 записывается из таблицы экранной формы при её загрузке и доступно только по чтению. Регистр действий представляет собой двоичное число разрядностью 1 байт, битовые поля которого определяют вид загруженной экранной формы и правила выполнения команд.

В режиме автоматической работы значение регистра действий не определено.

16.3.5. Период автоматического обновления экрана (R9206)

Регистр доступен при обращении к регистру памяти с адресом 9206 и может принимать целые значения от 0 до 255. Записанное число определяет период автоматического обновления экрана с дискретностью 10 мс. При записи числа 0 период составляет 10 мс, при записи числа 255 - 2,56 с.

При считывании содержимого регистра 9206 в регистр стека X записывается нулевое значение, если обновление экрана выполнялось с момента предыдущего считывания, и ненулевое значение в противном случае.

Автоматическое обновление используется, если во флаге действия активного элемента загруженной формы установлен бит разрешения периодического обновления экрана (см. п. 16.2.2.3).

16.3.6. Адрес регистра для ввода чисел (R9207)

Регистр памяти с адресом 9207 содержит адрес регистра данных для записи числа при вводе его из интерфейса пользователя. Возможные значения от 0 до 999. Доступен по записи при разрешении функций интерфейса. По чтению доступен во всех режимах.

Кроме этого значение адреса загружается в регистр адреса автоматически из таблицы экранной формы при выполнении соответствующих команд языка графического интерфейса.

16.3.7. Регистр команд интерфейса (R9208)

Команды интерфейса выполняются путем выполнения команды записи в регистр памяти с адресом 9208. Требуемый код команды предварительно заносится в регистр X ЭВМ. При чтении из указанного регистра, считывается код последней поданной команды.

Регистр команд доступен по записи при разрешении функций интерфейса, по чтению - во всех режимах.

16.3.8. Регистр результата выполнения операции (R9209)

Регистр доступен только по чтению при обращении к регистру памяти с адресом 9209 и содержит результат последней выполненной команды интерфейса или код ошибки.

Регистр может принимать значения:

- 0 - команда закончена успешно;
- 1 - требуется продолжить выполнение команды;
- 2 - несуществующий номер экранной формы;
- 3 - несуществующий номер активного элемента;

- 4 - несуществующий код команды экранной формы;
- 5 - несуществующий номер шрифта;
- 6 - несуществующий код символа в установленном шрифте;
- 7 - окончание ввода числа.

Нулевое значение указывает, что отрисовка экранной формы выполнена успешно и дополнительных действий не требуется.

Значение 1 устанавливается, если последняя выполненная команда разрешена для активного элемента, кроме того во флагах действия был установлен соответствующий признак действия. В таком случае стандартная обработка команды интерфейса, включающая изменение номера загруженной формы и/или активного элемента, не производится. Обработка команды должна полностью выполняться программой пользователя.

Значения от 2 до 6 не должны возникать в процессе работы отлаженной программы пользователя.

Значение 7 устанавливается при окончании ввода числа при помощи форм с номерами 253 и 254 (см. п. 16.6).

16.4. Работа интерфейса и действие команд

16.4.1. Работа интерфейса

В качестве команд управления графическим интерфейсом используются коды кнопок ЭВМ (см. приложение Б) ВЫХОД (32), ВВОД (33), ВЫБОР ВПРАВО (34), ВЫБОР ВЛЕВО (35), ВЫБОР ВВЕРХ (36), ВЫБОР ВНИЗ (37), а также специальный код 64, по которому производится безусловная перерисовка экранной формы. Все команды выполняются по правилам, определенным флагами действий активного элемента в загруженной экранной форме.

Перед началом работы интерфейса программе пользователя необходимо включить функции интерфейса (записать 1 в регистр памяти с адресом 9200), запрограммировать адрес главной таблицы меню (R9201), загрузить указатели, соответствующие начальной экранной форме (R9202, R9203) и подать команду

перерисовки экранной формы (записать 64 в регистр памяти с адресом 9208), после чего автоматически выполняются следующие операции.

По номеру экранной формы из главной таблицы меню находится адрес таблицы экранной формы и, по номеру активного элемента, загружается регистр действий. Далее в область графического экрана последовательно выводятся графические объекты, строки текста и числа, запрограммированные в основной программе загруженной формы, а затем объекты, строки и числа, запрограммированные в программе для активного элемента. После этого содержимое графической памяти выводится на индикатор (аналогично действию команды К ГРФ), в регистр результата заносится число 0.

Дальнейшая работа программы пользователя должна происходить в цикле, в котором содержимое буфера клавиатуры (R9029) передаётся в регистр команд интерфейса (R9208).

Обработка нажатия любой клавиши, код которой не является командой интерфейса, должен выполняться программой пользователя, если это необходимо.

Если в регистр результата выполнения операции (R9209) после отрисовки формы заносится код 1, то программа пользователя должна выполнить некоторые действия, исходя из считываемых значений указателей экранной формы (R9202, R9203) и последней поданной команды интерфейса (R9208).

Если во время отрисовки экранной формы возникает ошибка (код от 2 до 6), то содержимое графической памяти на индикатор не выводится, в регистр результата выполнения операции (R9209) заносится код ошибки.

16.4.2. Команды ВЫБОР

Команды ВЫБОР используются в экранных формах, имеющих несколько элементов, например выполненных в виде списка. Команды предназначены для выбора нужного элемента, т.е. для назначения этого элемента активным.

Все команды выбора выполняются, если в регистре действий (см. п. 16.3.4) установлен флаг разрешения кнопок ВЫБОР.

По команде ВЫБОР ВНИЗ указатель номера активного элемента увеличивается на единицу и экранная форма перерисовывается. При достижении максимального значения, определенного экранной формой, номер сбрасывается в ноль. Таким образом обеспечивается последовательный обход всех элементов экранной формы.

По команде ВЫБОР ВВЕРХ производятся те же действия, только указатель номера каждый раз уменьшается на единицу и, после достижения нуля, принимает максимальное значение. Таким образом обеспечивается последовательный обход всех элементов экранной формы в обратную сторону.

По командам ВЫБОР ВЛЕВО и ВЫБОР ВПРАВО указатель номера активного элемента загружается из соответствующей таблицы переходов. Такая функция удобна для работы с экранными формами, выполненными в виде списка с несколькими столбцами и обеспечивает переход из одного столбца в другой.

Если в регистре действий (R9204) кроме флага разрешения кнопок ВЫБОР установлен флаг признака выбора, то при выполнении каждой команды выбора действие не выполняется и в регистре результата выполнения операции (R9209) устанавливается значение 1. Это дает возможность организовать любое нестандартное выполнение команд.

16.4.3. Команда ВВОД

Команда ВВОД выполняется, если в регистре действий установлен соответствующий флаг разрешения.

При подаче команды указатели экранной формы загружаются значениями из таблицы переходов для кнопки ВВОД, соответствующими номеру активного элемента загруженной экранной формы, после чего экранная форма перерисовывается. На экран выводится новая экранная форма и таким образом происходит переход из одной экранной формы в другую.

Если в регистре действий (R9204) кроме флага разрешения кнопки ВВОД установлен флаг признака ввода, то при выполнении команды ввод действие не

выполняется и в регистре результата выполнения операции (R9209) устанавливается значение 1.

16.4.4. Команда ВЫХОД

Действие команды ВЫХОД аналогично действию команды ВВОД, при этом указатели новой экранной формы загружаются из таблицы переходов для кнопки ВЫХОД.

16.4.5. Переход из одной экранной формы в другую

Переход выполняется автоматически при обработке команд ВВОД и ВЫХОД.

Переход в новую экранную форму можно выполнить непосредственно загрузив регистры указателей экранной формы (R9203, R9204) необходимыми значениями. После подачи команды перерисовки экранной формы (записи кода 64 в R9208), требуемая экранная форма будет выведена на экран.

16.5. Команды языка интерфейса

16.5.1. Описание

Программы для структуры экранной формы записываются на языке интерфейса пользователя. Команды языка подразделяются на команды графического вывода и команды управления.

Команды графического вывода по назначению аналогичны функциям графического экрана (см. п. 2). Номера регистров памяти и адреса в памяти программ в командах записываются в BCD формате и занимают два байта, старший байт - первый.

Команды управления определяют порядок выполнения программы. Адреса переходов в командах являются смещением относительно адреса начала структуры экранной формы, занимают один байт и могут принимать значения от 0 до 255.

Все программы написанные на языке экранной формы должны заканчиваться командой окончания END с кодом 255 (0FFh).

16.5.2. Перечень команд

Команды языка графического интерфейса пользователя перечислены в табл. 7.

Таблица 7

Команда	Обозна- чение	Параметры	Код	
			DEC	HEX
Команды графического вывода				
Установка начальной позиции вывода	TAB	X0, Y0	96	60h
Загрузка начальной позиции вывода по X из памяти десятичных данных	LDX	addr	104	68h
Загрузка начальной позиции вывода по Y из памяти десятичных данных	LDY	addr	105	69h
Установка номера шрифта	TYPE	N	101	65h
Загрузка адреса загружаемого шрифта	ATYPE	addr	78	4Eh
Загрузка адреса регистра для ввода числа	LDADR	addr	77	4Dh
Вывод прямоугольника	SQU	atr, X, Y	89	59h
Вывод рамки	FRAME	atr, X, Y	91	5Bh
Вывод графического образа из памяти программ	DRAWP	atr, addr	84	54h
Вывод графического образа из памяти двоичных данных	DRAWB	atr, addr	86	56h
Вывод строки символов из памяти программ	TXTP	atr, addr	83	53h
Вывод строки символов из памяти двоичных данных	TXTB	atr, addr	85	55h
Вывод строки символов с косвенной адресацией	TXTK	atr, addr	87	57h
Вывод числа с естественной или плавающей запятой	PREXP	atr, addr	92	5Ch
Вывод числа в форматированном виде	PRFORM	atr, addr, M, N	93	5Dh
Вывод числа в шестнадцатиричном виде	PRHEX	atr, addr	94	5Eh

Продолжение таблицы 7

Команда	Обозна- чение	Параметры	Код	
			DEC	HEX
Команды управления				
Переход к подпрограмме в текущей форме	MCALL	rem	102	66h
Возврат из подпрограммы в текущей форме	RETUM		103	67h
Переход к подпрограмме в памяти программ	LCALL	addr	112	70h
Возврат из подпрограммы в памяти программ	RETUL		113	71h
Переход, если число не равно 0	JNEZ	addr, rem	1	01h
Безусловный переход	JM	rem	56	38h
Окончание программы	END		255	0FFh
Примечание:				
X0,Y0 - координаты графического объекта, 1 байт;				
X,Y - размеры графического объекта, 1 байт;				
atr - атрибуты графического объекта, 1 байт;				
M, N - число, 1 байт;				
addr - адрес в памяти в формате BCD, 2 байта;				
rem - адрес (смещение) относительно начала текущей формы, 1 байт.				

16.5.3. Установка начальной позиции вывода (TAB)

Код команды: 96 (60h).

Длина команды: 3 байта.

Формат записи: TAB, X0, Y0.

Где X0 (от 0 до 127), Y0 (от 0 до 63) - начальные координаты для вывода объекта на экран.

16.5.4. Загрузка начальной позиции вывода по X из памяти десятичных данных (LDX)

Код команды: 104 (68h).

Длина команды: 3 байта.

Формат записи: LDX, addr.

Где addr - 2 байта, адрес регистра памяти в BCD формате (от 00h, 00h до 09h, 99h).

16.5.5. Загрузка начальной позиции вывода по Y из памяти десятичных данных (LDY)

Код команды: 105 (69h).

Длина команды: 3 байта.

Формат записи: LDY, addr.

Где addr - 2 байта, адрес регистра памяти в BCD формате (от 00h, 00h до 09h, 99h).

16.5.6. Установка номера шрифта (TYPE)

Код команды: 101 (65h).

Длина команды: 2 байта.

Формат записи: TYPE, N.

Где N - 1 байт, номер шрифта (от 0 до 2 или 254).

16.5.7. Загрузка адреса загружаемого шрифта (ATYPE)

Код команды: 78 (4Eh).

Длина команды: 3 байта.

Формат записи: ATYPE, addr.

Где addr - 2 байта, адрес начала структуры загружаемого шрифта в памяти программ в BCD формате (от 00h, 00h до 99h, 99h).

16.5.8. Загрузка адреса регистра для ввода числа (LDADR)

Код команды: 77 (4Dh).

Длина команды: 3 байта.

Формат записи: LDADR, addr.

Где addr - 2 байта, адрес регистра памяти в BCD формате (от 00h, 00h до 09h, 99h).

16.5.9. Вывод прямоугольника (SQU)

Код команды 89 (59h).

Длина команды: 4 байта.

Формат записи: SQU, atr, X, Y.

Где atr - атрибуты вывода, X и Y - размеры прямоугольника.

16.5.10. Вывод рамки (FRAME)

Код команды: 91 (5Bh).

Длина команды: 4 байта.

Формат записи: FRAME, atr, X, Y.

Где atr - атрибуты вывода, X и Y - размеры рамки.

16.5.11. Вывод графического образа из памяти программ (DRAWP)

Код команды: 84 (54h).

Длина команды: 4 байта.

Формат записи: DRAWP, atr, addr.

Где atr - атрибуты вывода, addr - 2 байта, адрес рисунка в памяти программ в BCD формате (от 00h, 00h до 99h, 99h).

16.5.12. Вывод графического из памяти двоичных данных (DRAWB)

Код команды: 86 (56h).

Длина команды: 4 байта.

Формат записи: DRAWB, atr, addr.

Где atr - атрибуты вывода, addr - 2 байта, адрес рисунка в памяти двоичных данных в BCD формате (от 10h, 00h до 81h, 67h).

16.5.13. Вывод строки символов из памяти программ (TXTP)

Код команды: 83 (53h).

Длина команды: 4 байта.

Формат записи: TXTP, atr, addr.

Где atr - атрибуты вывода, addr - 2 байта, адрес текста в памяти программ в BCD формате (от 00h, 00h до 99h, 99h). Расстояние между символами записывается в старшем полубайте байта атрибутов.

16.5.14. Вывод строки символов из памяти двоичных данных (TXTB)

Код команды: 85 (55h).

Длина команды: 4 байта.

Формат записи: TXTB, atr, addr.

Где atr - атрибуты вывода, addr - 2 байта, адрес текста в памяти байтовых данных в BCD формате (от 10h, 00h до 81h, 67h). Расстояние между символами записывается в старшем полубайте байта атрибутов.

16.5.15. Вывод строки символов с косвенной адресацией (TXTK)

Код команды: 87 (57h).

Длина команды: 4 байта.

Формат записи: TXTK, atr, addr.

Где atr - атрибуты вывода, addr - 2 байта, адрес регистра десятичных данных в BCD формате (от 00h, 00h до 09h, 99h), в котором хранится адрес начала строки символов в памяти байтовых данных (от 1000 до 8167). Расстояние между символами записывается в старшем полубайте байта атрибутов. Содержимое адресного регистра не изменяется.

16.5.16. Вывод числа с естественной или плавающей запятой (PREXP)

Код команды: 92 (5Ch).

Длина команды: 4 байта.

Формат записи: PREXP, atr, addr.

Где atr - атрибуты вывода, addr - 2 байта, адрес регистра памяти в BCD формате (от 00h, 00h до 09h, 99h). Расстояние между символами записывается в старшем полубайте байта атрибутов.

16.5.17. Вывод числа в форматированном виде (PRFORM)

Код команды: 93 (5Dh).

Длина команды: 6 байт.

Формат записи: PRFORM, atr, addr, M, N.

Где atr - атрибуты вывода, addr - 2 байта, адрес регистра памяти в BCD формате (от 00h, 00h до 09h, 99h), M, N - число значащих цифр целой и дробной части числа соответственно. Расстояние между символами записывается в старшем полубайте байта атрибутов.

16.5.18. Вывод числа в шестнадцатичном виде (PRHEX)

Код команды: 94 (5Eh).

Длина команды: 4 байта.

Формат записи: PRHEX, atr, addr.

Где atr - атрибуты вывода, addr - 2 байта, адрес регистра памяти в BCD формате (от 00h, 00h до 09h, 99h). Расстояние между символами записывается в старшем полубайте байта атрибутов.

16.5.19. Переход к подпрограмме в текущей форме (MCALL)

Код команды: 102 (66h).

Длина команды: 2 байта.

Формат записи: MCALL, rem.

Где rem - адрес (смещение) метки подпрограммы.

Возможен вложенный вызов подпрограмм.

16.5.20. Возврат из подпрограммы в текущей форме (RETUM)

Код команды: 103 (67h).

Длина команды: 1 байт.

Формат записи: RETUM.

16.5.21. Переход к подпрограмме в памяти программ (LCALL)

Код команды: 112 (70h).

Длина команды: 3 байта.

Формат записи: LCALL, addr.

Где addr - адрес подпрограммы в памяти программ в BCD формате (от 00h, 00h до 99h, 99h).

Возможен вложенный вызов подпрограмм.

16.5.22. Возврат из подпрограммы в памяти программ (RETUL)

Код команды: 113 (71h).

Длина команды: 1 байт.

Формат записи: RETUL.

16.5.23. Переход если число не равно 0 (JNEZ)

Код команды: 1 (01h).

Длина команды: 4 байта.

Формат записи: JNEZ, addr, rem.

Где addr - 2 байта, адрес регистра памяти в BCD формате (от 00h, 00h до 09h, 99h), rem - адрес (смещение) метки перехода.

16.5.24. Безусловный переход (JM)

Код команды: 56 (38h).

Длина команды: 2 байта.

Формат записи: JM, rem.

Где rem - адрес (смещение) метки перехода.

16.5.25. Окончание программы (END)

Код команды: 255 (0FFh).

Длина команды: 1 байт.

Формат записи: END.

16.6. Экранные формы встроенного программного обеспечения ЭВМ

16.6.1. Экранная форма ввода десятичного числа

Форма с номером 254 (0FEh) предназначена для ввода десятичных чисел. Структура расположена во встроенном программном обеспечении ЭВМ вне памяти программ пользователя.

При указании номера этой формы в таблице переходов по команде ВВОД вызывается форма, состоящая из одного элемента - поля ввода числа, которая выводится без очистки предыдущего изображения в нижнюю часть графического экрана.

Ввод числа производится в естественной форме или с плавающей запятой и аналогичен вводу числа в регистр стека X в режиме автоматической работы.

Для ввода применяются клавиши "0...9", "запятая", "ВП" и "/-/-".

Запись значения в регистр после набора числа производится нажатием клавиши "ВВОД", при этом выполняется возврат в экранную форму из которой было выполнено обращение к форме ввода десятичного числа.

Нажатие клавиши "Сх" при вводе позволяет обнулить неверно набранное значение.

Нажатие клавиши "ВЫХОД" выполняет возврат в предыдущую форму без изменения ранее записанного в регистре значения.

Адрес регистра для ввода числа указывается командой LDADR языка интерфейса или загрузкой нужным значением регистра памяти с номером 9207.

16.6.2. Экранная форма ввода шестнадцатичного числа

Форма с номером 253 (0FDh) предназначена для ввода однобайтовых шестнадцатичных чисел. Структура расположена во встроенном программном обеспечении ЭВМ вне памяти программ пользователя.

При указании номера этой формы в таблице переходов по команде ВВОД вызывается форма, состоящая из одного элемента - поля ввода

шестнадцатиричного числа, которая выводится без очистки предыдущего изображения в нижнюю левую часть графического экрана.

Для ввода применяются клавиши "0...9" и "A...F". При формировании шестнадцатиричного числа учитываются только последние нажатые клавиши. Неверно набранное значение можно заменить продолжив ввод.

Подтверждение ввода после набора числа производится клавишей "ВВОД", при этом выполняется возврат в форму из которой было выполнено обращение к форме ввода шестнадцатиричного числа.

Нажатие клавиши "ВЫХОД" выполняет возврат в предыдущую форму без изменения ранее записанного в регистре значения.

Адрес регистра для ввода числа указывается командой LDADR языка интерфейса или загрузкой нужным значением регистра памяти с номером 9207.

17. РАБОТА С ТАБЛИЦАМИ

17.1. Описание

В ЭВМ предусмотрена возможность работы с упорядоченными или ассоциативными таблицами, размещёнными в памяти программ.

Упорядоченные таблицы предназначены для выбора значения по его порядковому номеру. Таблицы содержат последовательно записанные двухбайтовые числа в шестнадцатиричном или упакованном BCD формате.

Ассоциативные таблицы предназначены для поиска значения по связанному с ним индексу. Таблицы содержат группы по три байта, состоящие из однобайтового ключа и ассоциированного с ним значения - двухбайтового числа в шестнадцатиричном или упакованном BCD формате. Последняя группа таблицы начинается числом 255 (0FFh), поиск по данному индексу не производится.

Перед обращением к функциям работы с таблицами адрес начала таблицы заносится в индексный регистр памяти программ R9042 (см. п. 6.1.3).

17.2. Функции для работы с таблицами

17.2.1. Чтение значения по номеру (R9210)

Операция производится при выполнении команды записи в регистр памяти данных с адресом 9210. В регистре X стека задаётся порядковый номер значения в упорядоченной таблице чисел в шестнадцатиричном формате. Преобразование содержимого регистра X стека при обращении к функции происходит аналогично модификации при косвенной адресации.

Функция считывает два последовательных байта из памяти программ. Считывание начинается с адреса, отстоящего от содержимого индексного регистра памяти программ на удвоенное содержимое регистра X стека. Имеющий меньший адрес байт интерпретируется как старший байт двухбайтового шестнадцатиричного числа. Байт имеющий больший адрес - как младший байт. Считанное из памяти программ число преобразуется в целое в диапазоне от 0 до 65536 и помещается в регистр X стека, замещая его предыдущее содержимое.

Операция чтения из регистра памяти данных по указанному адресу не изменяет стек ЭВМ.

17.2.2. Чтение двоично-десятичного значения по номеру (R9211)

Операция производится при выполнении команды записи в регистр памяти данных с адресом 9211. В регистре X стека задаётся порядковый номер значения в упорядоченной таблице двухбайтовых чисел в упакованном BCD формате. Преобразование содержимого регистра X стека при обращении к функции происходит аналогично модификации при косвенной адресации.

Функция считывает два последовательных байта из памяти программ. Считывание начинается с адреса, отстоящего от содержимого индексного регистра памяти программ на удвоенное содержимое регистра X стека. Имеющий меньший адрес байт интерпретируется как старший байт двухбайтового числа в упакованном BCD формате. Байт имеющий больший адрес - как младший байт.

Считанное из памяти программ число преобразуется в целое в диапазоне от 0 до 16665 аналогично преобразованию адреса в командах перехода и помещается в регистр X стека, замещая его предыдущее содержимое.

Операция чтения из регистра памяти данных по указанному адресу не изменяет стек ЭВМ.

17.2.3. Поиск значения по индексу (R9212)

Операция производится при выполнении команды записи в регистр памяти данных с адресом 9212. В регистре X стека задаётся индекс в диапазоне от 0 до 254. Преобразование содержимого регистра X стека при обращении к функции происходит аналогично модификации при косвенной адресации.

Функция выполняет последовательные обращения к содержимому памяти программ по адресам, отстоящим от содержимого индексного регистра памяти программ на величины, кратные трём. Считанный из памяти программ ключ очередной группы сравнивается в содержимом регистра X стека. Обращения продолжаются до совпадения значений индекса и ключа, до считывания из памяти программ признака окончания таблицы - числа 255 (0FFh) или до выхода за границы памяти программ.

При нахождении совпадения функция считывает два следующих последовательных байта из памяти программ.

Имеющий меньший адрес байт интерпретируется как старший байт двухбайтового шестнадцатиричного числа. Байт имеющий больший адрес - как младший байт. Считанное из памяти программ число преобразуется в целое в диапазоне от 0 до 65536 и помещается в регистр X стека, замещая его предыдущее содержимое.

Если совпадения не обнаружено до считывания признака окончания таблицы или до выхода за границу памяти программ, то в регистр X стека записывается число минус 1.

Операция чтения из регистра памяти данных по указанному адресу не изменяет стек ЭВМ.

17.2.4. Поиск двоично-десятичного значения по индексу (R9213)

Операция производится при выполнении команды записи в регистр памяти данных с адресом 9213. В регистре X стека задаётся индекс в диапазоне от 0 до 254. Преобразование содержимого регистра X стека при обращении к функции происходит аналогично модификации при косвенной адресации.

Функция выполняет последовательные обращения к содержимому памяти программ по адресам, отстоящим от содержимого индексного регистра памяти программ на величины, кратные трём. Считанный из памяти программ байт (ключ очередной группы) сравнивается в содержимом регистра X стека. Обращения продолжаются до совпадения значений индекса и ключа, до считывания из памяти программ признака окончания таблицы - числа 255 (0FFh) или до выхода за границы памяти программ.

При нахождении совпадения функция считывает два следующих последовательных байта из памяти программ. Имеющий меньший адрес байт интерпретируется как старший байт двухбайтового числа в упакованном BCD формате. Байт имеющий больший адрес - как младший байт.

Считанное из памяти программ число преобразуется в целое в диапазоне от 0 до 16665 аналогично преобразованию адреса в командах перехода и помещается в регистр X стека, замещая его предыдущее содержимое.

Если совпадения не обнаружено до считывания признака окончания таблицы или до выхода за границу памяти программ, то в регистр X стека записывается число минус 1.

Операция чтения из регистра памяти данных по указанному адресу не изменяет стек ЭВМ.