

Выполнение команд в ЭВМ типа IBM PC

Цель работы: изучение порядка функционирования ЭВМ типа IBM PC при выполнении линейных операций типа сложения и вычитания.

Введение

Микропроцессор *Intel-8086* (K1810BM80) имеет двухадресную систему команд [8,10]. Ее особенностью является отсутствие команд, использующих оба операнда из оперативной памяти. Исключения составляют лишь команды пересылки и сравнения цепочек байт или слов, которые в данной работе рассматриваться не будут. Таким образом, в командах допустимы следующие сочетания операндов: *RR, RS, RI, SI*. Здесь *R* обозначает операнд, находящийся в одном из регистров регистровой памяти микропроцессора, *S* – операнд, находящийся в оперативной памяти, адрес которого формируется по одному из допустимых способов адресации, *I* – непосредственный операнд, закодированный в адресном поле самой команды. Формат команды определяется способом адресации операнда, находящегося в оперативной памяти, длиной используемого непосредственного операнда, а также наличием и длиной смещения, используемого при относительных режимах адресации.

Микропроцессор имеет все классические режимы адресации, но с определенными особенностями.

Непосредственная адресация предполагает, что операнд занимает одно из полей команды и, следовательно, выбирается из оперативной памяти одновременно с ней. В зависимости от форматов обрабатываемых процессором данных непосредственный операнд может иметь длину 8 или 16 бит, что в дальнейшем будем обозначать *data8* и *data16* соответственно.

Механизмы адресации операндов, находящихся в регистровой памяти и в оперативной памяти, существенно различаются. К регистровой памяти допускается лишь *прямая регистровая адресация*. При этом в команде указывается номер регистра, содержащего операнд. 16-разрядный операнд может находиться в регистрах *AX, BX, CX, DX, DI, SI, SP, BP*, а 8-разрядный – в регистрах *AL, AH, BL, BH, CL, CH, DL, DH*, представляющих собой младшие (*xL*) и старшие (*xH*) байты регистров *AX, BX, CX, DX*.

Адресация оперативной памяти имеет свои особенности, связанные с её разбиением на сегменты и использованием сегментной группы регистров для указания начального адреса сегмента. 16-разрядный адрес, получаемый в блоке формирования адреса операнда на основе указанного режима адресации, называется эффективным адресом (ЭА). Иногда эффективный адрес обозначается как *EA* (*effective address*). 20-разрядный адрес, который получается сложением эффективного адреса и увеличенного в 16 раз значения соответствующего сегментного регистра, называется физическим адресом (ФА).

Именно физический адрес передается из микропроцессора по 20-ти адресным линиям, входящим в состав системной шины, в оперативную память и используется при обращении к её ячейке на физическом уровне. При получении эффективного адреса могут использоваться все классические режимы адресации, а также некоторые их комбинации.

Прямая адресация предполагает, что эффективный адрес является частью команды. Так как ЭА состоит из 16 разрядов, то и соответствующее поле команды должно иметь такую же длину.

При *регистровой косвенной адресации* эффективный адрес операнда находится в базовом регистре *BX* или одном из индексных регистров *DI* либо *SI*:

$$\text{ЭА} = \left\{ \begin{array}{c} [BX] \\ [DI] \\ [SI] \end{array} \right\}$$

Обозначение имени регистра в квадратных скобках указывает на содержимое соответствующего регистра. Фигурные скобки – символ выбора одной из нескольких возможных альтернатив.

При *регистровой относительной адресации* эффективный адрес равен сумме содержимого базового или индексного регистра и смещения:

$$\text{ЭА} = \left\{ \begin{array}{c} [BX] \\ [BP] \\ [DI] \\ [SI] \end{array} \right\} + \left\{ \begin{array}{c} \text{disp8} \\ \text{disp16} \end{array} \right\}$$

Обозначения *disp8* и *disp16* здесь и далее указывают на 8- или 16-разрядное смещение соответственно.

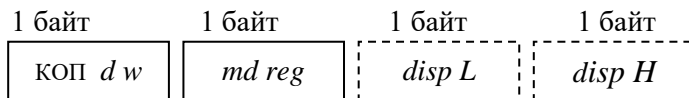
Эффективный адрес при базово-индексной адресации равен сумме содержимого базового и индексного регистров, определяемых командой:

$$\text{ЭА} = \left\{ \begin{array}{c} [(BX) \\ [BP] \end{array} \right\} + \left\{ \begin{array}{c} [(DI) \\ [SI] \end{array} \right\}$$

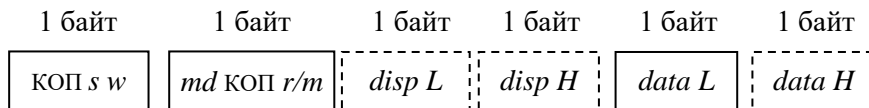
Наиболее сложен механизм относительной базово-индексной адресации. Эффективный адрес в этом случае равен сумме 8- или 16-разрядного смещения и базово-индексного адреса:

$$\text{ЭА} = \left\{ \begin{array}{c} [BX] \\ [BP] \end{array} \right\} + \left\{ \begin{array}{c} [DI] \\ [SI] \end{array} \right\} + \left\{ \begin{array}{c} \text{disp8} \\ \text{disp16} \end{array} \right\}$$

Форматы двухоперандных команд представлены на **рис.1**. Пунктиром показаны поля, которые в зависимости от режима адресации могут отсутствовать в команде.



а) формат команд типа *RR* и *RS*



б) формат команды с непосредственным операндом

Рис. 1. Форматы двухоперандных команд микропроцессора *I8086*

Поле КОП содержит код выполняемой операции. Признак w указывает на длину операндов. При $w=1$ операция проводится над словами, а при $w=0$ – над байтами.

Второй байт команды, называемый постбайтом, определяет операнды, участвующие в операции. Поле *reg* указывает регистр регистровой памяти согласно **табл. 1**.

Таблица 1

<i>reg</i>	Регистр	
	$w=1$	$w=0$
000	<i>AX</i>	<i>AL</i>
001	<i>CX</i>	<i>CL</i>
010	<i>DX</i>	<i>DL</i>
011	<i>BX</i>	<i>BL</i>
100	<i>SP</i>	<i>AH</i>
101	<i>BP</i>	<i>CH</i>
110	<i>SI</i>	<i>DH</i>
111	<i>DI</i>	<i>BH</i>

Поля *md* и *r/m* задают режим адресации второго операнда согласно табл.2.

Таблица 2

<i>r/m</i>	<i>md</i>				
	00	01	10	11	
				<i>w</i> =1	<i>w</i> =0
000	$[BX]+[SI]$ (DS)	$[BX]+[SI]+disp\ 8$ (DS)	$[BX]+[SI]+disp\ 16$ (DS)	AX	AL
001	$[BX]+[DI]$ (DS)	$[BX]+[DI]+disp\ 8$ (DS)	$[BX]+[DI]+disp\ 16$ (DS)	CX	CL
010	$[BP]+[SI]$ (SS)	$[BP]+[SI]+disp\ 8$ (SS)	$[BP]+[SI]+disp\ 16$ (SS)	DX	DL
011	$[BP]+[DI]$ (SS)	$[BP]+[DI]+disp\ 8$ (SS)	$[BP]+[DI]+disp\ 16$ (SS)	BX	BL
100	$[SI]$ (DS)	$[SI]+disp\ 8$ (DS)	$[SI]+disp\ 16$ (DS)	SP	AH
101	$[DI]$ (DS)	$[DI]+disp\ 8$ (DS)	$[DI]+disp\ 16$ (DS)	BP	CH
110	$disp\ 16$ (DS)	$[BP]+disp\ 8$ (SS)	$[BP]+disp\ 16$ (SS)	SI	DH
111	$[BX]$ (DS)	$[BX]+disp\ 8$ (DS)	$[BX]+disp\ 16$ (DS)	DI	BH

Обращение к оперативной памяти происходит по физическому адресу (ФА) ячейки, который формируется по следующей схеме:

$$\text{ФА} = (\text{сегментный регистр}) * 16 + \text{ЭА}$$

В табл. 2 помимо указания режима адресации операнда, согласно которому формируется эффективный адрес, указан также сегментный регистр, используемый по умолчанию для получения физического адреса.

В командах, использующих непосредственный операнд, признак *s* вместе с признаком *w* определяет разрядность непосредственного операнда, записываемого в команде, и разрядность выполняемой операции согласно табл.3.

Таблица 3

<i>w</i>	<i>s</i>	Операция	Непосредственный операнд
0	0	8-разрядная	8-разрядный
0	1	не используется	
1	0	16-разрядная	16-разрядный
1	1		8-разрядный, расширяемый знаком до 16-ти разрядов при выполнении операции

Символическая запись команды, в которой один операнд извлекается из оперативной памяти, а второй является непосредственным операндом в случае, когда этот операнд имеет в команде длину 1 байт, не позволяет однозначно определить, будет ли данная команда выполняться над операндами длиной 1 байт или длиной 2 байта с расширением непосредственного операнда до этой длины знаком. Для устранения этой неопределённости при символической записи команды используют специальные указатели. Указатель `byte ptr` определяет длины операндов в 1 байт, а указатель `word ptr` – в 2 байта. Например, в команде `byte ptr ADD [BX+3], 82h` будет выполняться сложение байт, а в команде `word ptr ADD [BX+3], 82h` будет выполняться сложение слов (2 байта), при этом младший байт слова будет прочитан из ОП по эффективному адресу `[BX+3]`, а старший байт – по эффективному адресу `[BX+4]`. Непосредственный операнд будет расширен знаком до 2 байт и поступит на обработку в виде `FF82h`.

Порядок выполнения работы

После запуска программы открывается окно с теоретическим материалом, после просмотра которого следует перейти к вкладке «Практика» и выбрать режим работы (с оценкой или без оценки).

Если работа выполняется без оценки, то обучаемый должен нажать кнопку «Старт». При этом ему будет выдано первое задание.

При записи информации в какой-либо регистр регистровой памяти в окне, показывающем состояние РП, необходимо выделить данный регистр. При этом в поле для ввода ответа появится идентификатор этого регистра, после чего в данное окно следует ввести информацию (рис.).

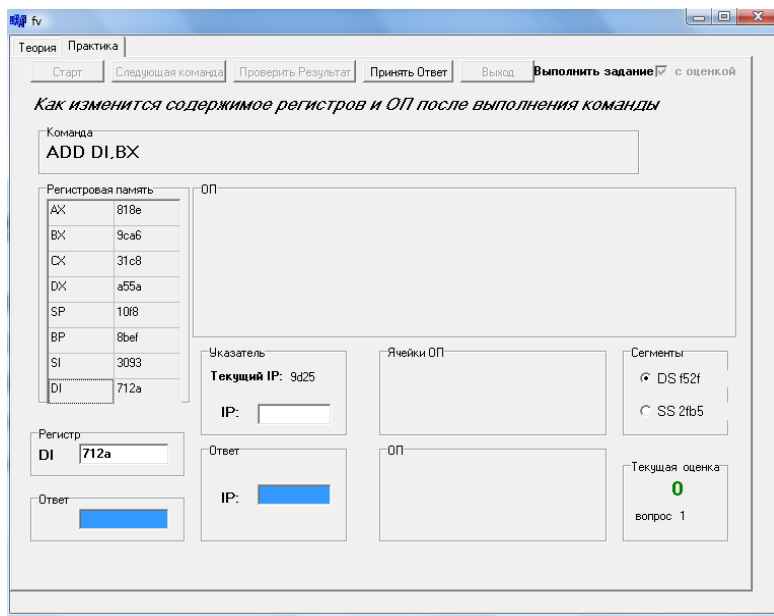


Рис. Окно тренировочного режима

Проверка правильности выполненного примера проводится при нажатии кнопки «Принять ответ». При этом программа обучающего урока выдаёт сообщение о том, правильно или неправильно вы-

полнено это задание. По кнопке «Проверить результат» выдаются правильные результаты данного задания (рис.).

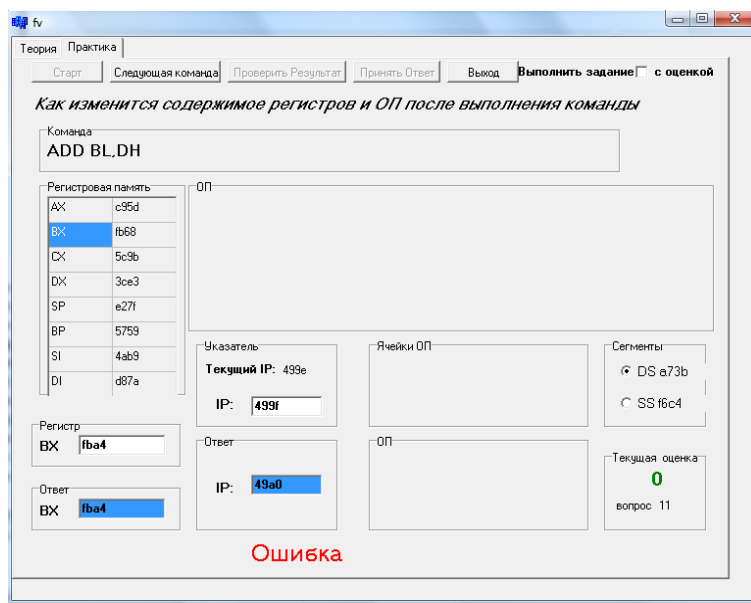


Рис. Окно с результатами проверки выполненного задания

Нажав кнопку «Следующая команда» можно получить очередное задание.

Для выполнения урока в контрольном режиме необходимо поставить флажок «Выполнить задание с оценкой» и нажать кнопку «Старт». В этом режиме становится недоступным просмотр теории. После получения очередного задания следует нажать кнопку «Принять ответ». Программные средства электронного урока проверяют правильность введённого ответа и сообщают об этом обучаемому. Если ответ введён правильно, выдаётся сообщение «Верно» и к текущей оценке добавляется 1 балл. При неправильном ответе выдаётся сообщение «Ошибка».

Кнопка «Проверить результат» в контрольном режиме недоступна. Правильный ответ выдаётся автоматически. Если студент не в состоянии самостоятельно отыскать ошибку, он должен, не

переходя к следующему заданию, пригласить для анализа результатов преподавателя.

Для перехода к следующему заданию следует нажать кнопку «Следующая команда». После выполнения 5 заданий и нажатия кнопки «Следующая команда» становится активной кнопка «Выход», по которой можно завершить контрольное выполнение урока. При этом кнопка «Принять ответ» неактивна. Новое выполнение программы в тренировочном или контрольном режимах можно осуществить её перезапуском.

Особенности работы программы

Все числа (смещение, непосредственный операнд), которые могут присутствовать в символической записи команды, представлены в 16-й системе счисления.

Необходимые вычисления результатов операции можно проводить с использованием калькулятора компьютера. При этом полученное при вычислении значение можно перенести из калькулятора в окно результата программы с помощью операций копирования и вставки (Cntr+C – Cntr+V).

При вычислении эффективного адреса следует иметь в виду, что его длина составляет 2 байта. Поэтому если при сложении его составляющих получается перенос за пределы 16-го разряда, он должен быть отброшен.

Информация в оперативную память записывается побайтно. Выбор байта, в который проходит запись, осуществляется наведением курсора на эту ячейку, при этом в окне ввода информации «Ячейки ОП» указывается адрес выбранной ячейки. Следует учитывать, что адрес слова – это адрес его младшего байта. (рис.). При формировании физического адреса следует учитывать то, какой сегментный регистр используется при данном режиме адресации (см. табл.).

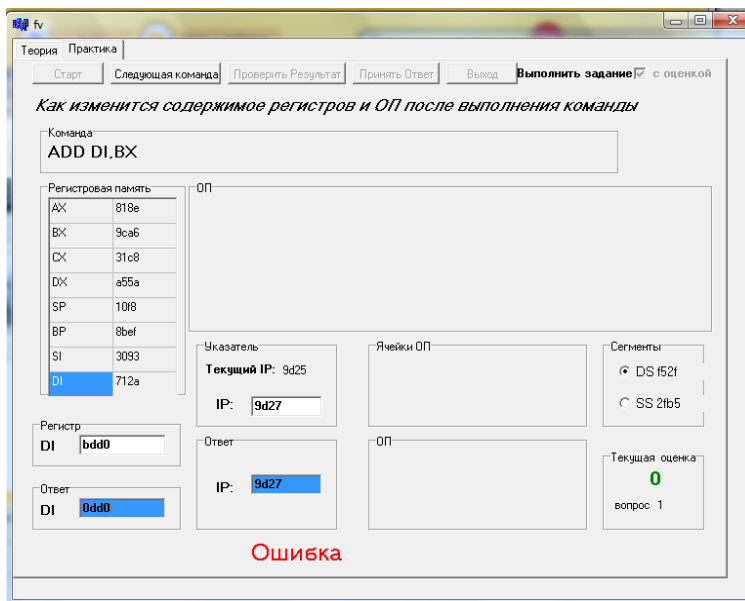


Рис. Окно урока при расположении одного из операндов в оперативной памяти

Рекомендации для преподавателя

Текущая оценка соответствует количеству правильно выполненных 5 заданий (рис.). В соответствии с ней преподаватель может выставить свою оценку за выполнение данного урока.

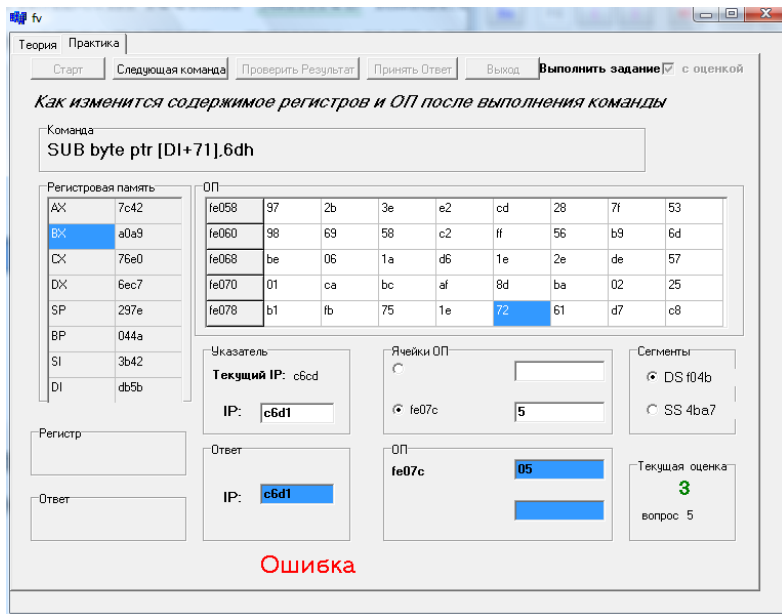


Рис. Окно урока при завершении выполнения заданий в контрольном режиме

Вопросы и задания к работе

1. Где могут располагаться операнды, используемые в командах ЭВМ типа IBM PC?
2. Какова адресность команд ЭВМ типа IBM PC?
3. Как в символической записи и машинном коде команды определяется операнд, на место которого записывается результат выполнения команды?
4. Какова длина используемых операндов, используемых в ЭВМ типа IBM PC? Каким образом определяется длина операнда в машинном представлении команды?
5. Как меняется формат команды при различных режимах адресации?

6. В каких пределах изменяется длина команды ЭВМ типа IBM PC? Приведите примеры самой короткой и самой длинной команд.

7. Как влияет длина команды на содержимое регистров ЭВМ при выполнении операции?

8. Каково назначение регистра указателя команд?

9. Как меняется регистр указателя команд во время выполнения программы? Чем отличается изменение регистра указателя команд при выполнении линейных команд и команд переходов?

10. Какую длину имеет машинное слово в ЭВМ типа IBM PC? Как она меняется в зависимости от разрядности микропроцессора?

11. Как выполняется адресация байт в пределах машинного слова?

12. Каково назначение сегментных регистров? Какая информация в них содержится? Каким образом определяется значение сегментных регистров перед началом выполнения программы?

13. Какие сегментные регистры используются при выполнении команд? Зависит ли это от используемых в команде режимов адресации операндов?

14. Что показывает физический адрес оперативной памяти? Как он формируется?

15. Что показывает эффективный адрес оперативной памяти? Как он формируется?

16. Из каких основных этапов состоит цикл выполнения команды? Чем отличается цикл выполнения команды классической ЭВМ и ЭВМ типа IBM PC? Почему?

17. Как обеспечивается автоматическое выполнение команд программы?

18. Символическая запись каких команд не позволяет однозначно определить длину используемых операндов? Какие средства применяются для того, чтобы устранить эту неоднозначность?

19. Почему в формате команды, использующей непосредственный операнд, отсутствует бит, указывающий, на место какого операнда поместить результат?