

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

УТВЕРЖДАЮ
Декан (директор)

_____ В.Н. Бориков
« ___ » _____ 2014 г.

С.Н. Торгаев

Микропроцессор Intel 8080

Методические указания к выполнению лабораторных работ
по курсу «Основы микропроцессорной техники» для студентов III курса,
обучающихся по направлениям 210100 «Электроника и микроэлектроника» и
210000 «Биотехнические системы и технологии»

Издательство
Томского политехнического университета
2014

УДК 000000
ББК 00000
А00

Торгаев С.Н.

Т60 Микропроцессор Intel 8080: методические указания к выполнению лабораторных работ по курсу «Основы микропроцессорной техники» для студентов III курса, обучающихся по направлениям 210100 «Электроника и наноэлектроника» и 210000 «Биотехнические системы и технологии» / В.С.Н. Торгаев; Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2014. – 52 с.

**УДК 000000
ББК 00000**

Методические указания рассмотрены и рекомендованы
к изданию методическим семинаром кафедры
промышленной и медицинской электроники ИНК
«_»_____ 2014 г.

Зав. кафедрой ПМЭ
профессор, д.т.н.

_____ *Г.С. Евтушенко*

© ФГБОУ ВПО НИ ТПУ, 2012
©Торгаев С.Н., 2014

Лабораторная работа №1

ОСНОВЫ РАБОТЫ С ЛАБОРАТОРНЫМ МАКЕТОМ МИКРОПРОЦЕССОРА

1.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Изучение процессора *Intel 8080* осуществляется с использованием отладочного комплекта Cyclone II FPGA компании Altera (рис. 1). Работа процессора реализована на базе микросхемы программируемой логической интегральной схемы (ПЛИС).

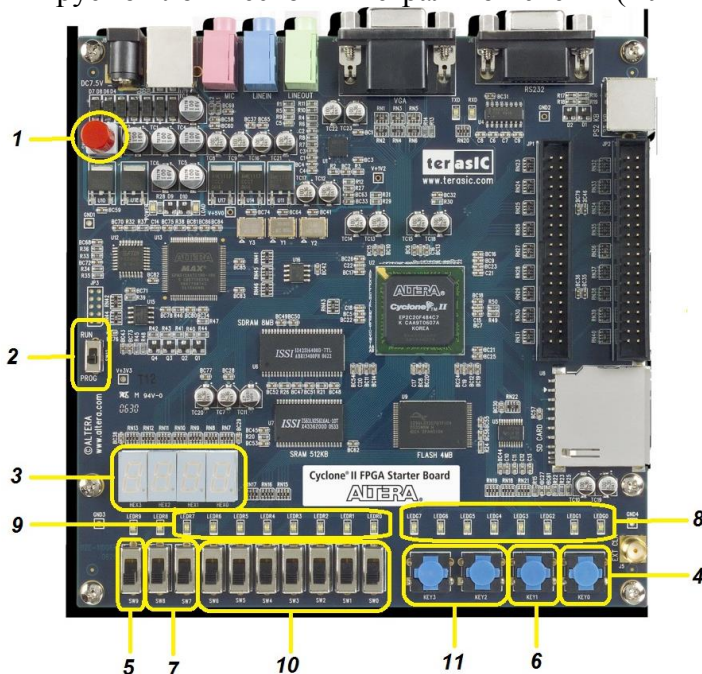


Рис. 1 - Отладочный комплект Cyclone II FPGA

Включение устройства осуществляется посредством кнопки *1* (рис. 1). При этом питание лабораторного модуля обеспечивает USB-порт персонального компьютера. Для работы модуля в режиме эмулятора микропроцессора K580BM80 (i8080) необходимо, чтобы переключатель *2* находился в положении *RUN* (верхнее положение).

Под цифрой *3* на рис. 1 расположены четыре семисегментных индикатора *HEX0-HEX3*, которые отображают содержимое программного счетчика *PC* микропроцессора. При работе процессора после нажатия кнопки *RESET* (кнопка *4* – *KEY0*) в программный счетчик записывается значение 0000h.

Тумблер *5* (*SW9*) переключает микропроцессор в режимы *СТАРТ-СТОП*. В положении *СТАРТ* (нижнее положение тумблера) микропроцессор выполняет программу заданную пользователем в автоматическом режиме. В верхнем положении тумблера (*СТОП*) микропроцессор приостанавливает выполнение текущей программы и переходит в пошаговый режим работы. В данном режиме микропроцессор выполняет команды только по нажатию кнопки *6* (*KEY1*).

Таблица 1. Частоты работы микропроцессора

Положение тумблеров		Частота
<i>SW8</i>	<i>SW7</i>	
↓	↓	25 Гц
↓	↑	2.5 кГц
↑	↓	250 кГц
↑	↑	25 МГц

Тумблеры 7 (*SW7-SW8*) осуществляют переключение частоты работы микропроцессора. В таблице 1 представлены частоты работы микропроцессора в зависимости от положения тумблеров *SW7-SW8*.

На светодиодах 8 (*LEDG0-LEDG7*) отображаются данные поступающие на внутреннюю шину данных микропроцессора.

Светодиоды 9 (*LEDR0-LEDR7*) подключены к порту ввода/вывода микропроцессора и отображают выводимые данные. Адрес порта вывода – 10h, вывод в порт (т.е. на светодиоды) осуществляется посредством команды *OUT*.

Переключатели 10 (*SW0 – SW6*) подключены к порту ввода микропроцессора (адрес порта 12h). При нижнем положении переключателя на порт поступает логический 0, а при верхнем положении – логическая 1. Считывание данных с порта осуществляется посредством команды *IN*. Кнопки 11 (*KEY2 – KEY3*) подключены к младшим разрядам порта ввода с адресом 11h. При нажатии кнопки на порт ввода поступает логическая 1.

Окно программы эмулятора микропроцессора K580BM80 (i8080) представлено на рис. 2. При подключении лабораторного модуля к персональному компьютеру и открытии программы эмулятора необходимо подключить модель к программе посредством кнопки «Открыть устройство». После подключения устройство будут активированы кнопки «Очистить», «Открыть файл» и «Сохранить файл».

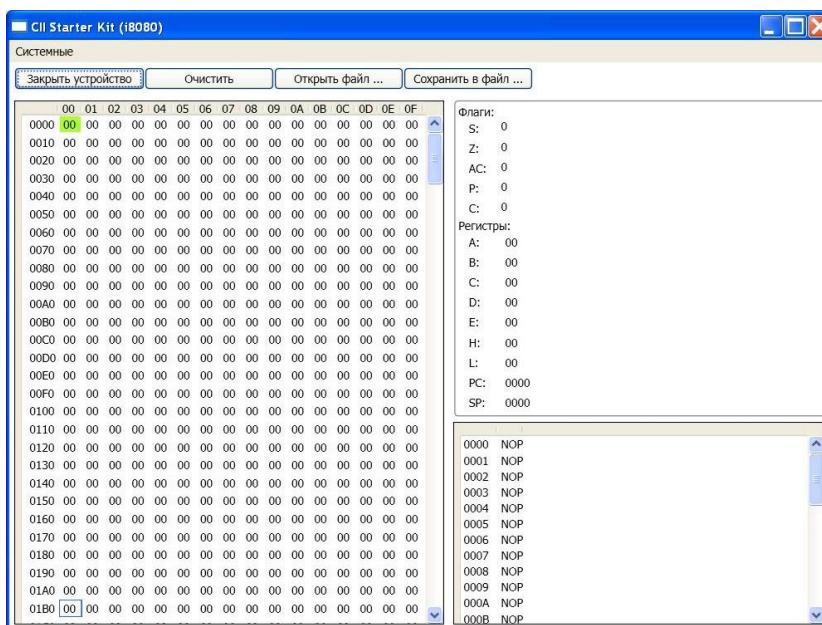


Рис. 2 – Окно программы эмулятора микропроцессора K580BM80

В левом окне программы эмулятора показано адресное пространство памяти. Выполняемая программа записывается в виде шестнадцатеричных кодов операций и операндов в ячейки памяти. В ходе занесения кодов программы в правом нижнем окне можно наблюдать мнемоническое обозначение программы.

В окне «*Флаги*» показаны биты флагов регистра состояния: *C* – перенос; *AC* – вспомогательный перенос; *S* – знак; *Z* – ноль; *P* – четность. Доступ к флагам возможен только программным путем.

В окне «*Регистры*» показано содержимое следующих регистров:

- 16-разрядный счетчик команд (*PC*);
- 16-разрядный регистр указатель стека (*SP*);
- 8-разрядный регистр-накопитель (*A*);
- шесть 8 –разрядных регистров общего назначения: *B,C,D,E,H,L*.

Изменение содержимого данных регистров возможно, как посредством команд микропроцессора, так и в ручном режиме, путем записи в них необходимых данных с клавиатуры в шестнадцатеричном коде.

Пример реализации программы записи данных в регистры *A* и *B* представлен на рис. 3.

MVI A, 0AAh
MVI B, 11h

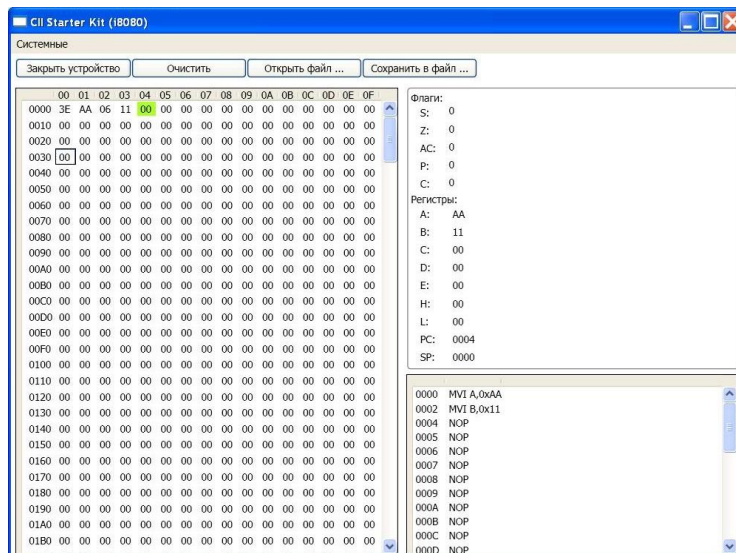


Рис. 2 – Окно программы эмулятора микропроцессора K580BM80

1.2. ПРОГРАММА РАБОТЫ

1. Подключить отладочный комплект Cyclone II FPGA к компьютеру и включить его питание.
2. Запустить программу эмулятора микропроцессора *Intel 8080*:
D:\CII_Project\CII_Project v1.1\CII Starter Kit. exe
3. Подключить отладочный макет к эмулятору посредством кнопки «Открыть устройство».
4. Запишите в память, начиная с адреса *0000h*, следующую последовательность команд:

Адрес	Команда	Машинный код	Комментарий
0000	<i>MVI A,F0h</i>	3E F0	загрузка регистра A←F0h
0002	<i>MVI B,0Fh</i>	06 0F	загрузка регистра B←0Fh

5. Переведите лабораторный макет в пошаговый режим работы и осуществите сброс микропроцессора (***RESET***). При этом значение программного счетчика микропроцессора должно установиться равным *0000h*, а на внутреннюю шину данных будет выставлен код операции команды ***MVI A,F0h*** равный *3E*. Выставьте минимальную частоту работы макета.
6. Наблюдая за внутренней шиной данных выполните одну команду. В ходе выполнения первой команды, после считывания кода операции, на шину данных на короткое время будет выставлено число *F0h*. После считывания операнда на шине данных будет выставлен код операции следующей команды - ***MVI B,0Fh***.

Лабораторная работа №2

КОМАНДЫ ЗАГРУЗКИ РЕГИСТРОВ. КОМАНДЫ ПЕРЕСЫЛКИ

2.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

В микропроцессоре K580BM80A (i8080) для программирования доступны следующие регистры:

- 16-разрядный счетчик команд (*PC*) содержит адрес команды, которая подлежит выполнению;
- 16-разрядный регистр указатель стека (*SP*), определяет адрес специализированной области ОЗУ-стека;
- 8-разрядный регистр-накопитель (*A*), используется для хранения и накопления результата в арифметических, логических операциях, а также в операциях ввода-вывода и сдвига. Кроме того, он может быть использован в качестве регистра общего назначения для хранения данных;
- шесть 8-разрядных регистров общего назначения (РОН): *B,C,D,E,H,L*;
- 8-разрядный регистр признаков *PSW (F)* содержит биты условий: *C* – перенос, *AC* – вспомогательный перенос, *S* – знак, *Z* – ноль, *P* – четность. Данные биты устанавливаются в зависимости от результата операции при выполнении арифметических, логических команд, команд сдвига и сравнения. Распределение условий в байте признаков следующее:

7 разряд	6 разряд	5 разряд	4 разряд	3 разряд	2 разряд	1 разряд	0 разряд
<i>S</i>	<i>Z</i>	<i>0</i>	<i>AC</i>	<i>0</i>	<i>P</i>	<i>I</i>	<i>C</i>

Регистры общего назначения могут использоваться для манипуляции 16-разрядными данными. Для этого регистры объединяются в пары следующим образом: *BC*, *DE*, *HL*, - соответственно, первый регистр используется для хранения старшего байта (например, *B*), а второй – для хранения младшего байта (*C*), а также *PSW (A + C)* слово состояния программы.

2.2. ПРОГРАММА РАБОТЫ

2.2.1. Команды загрузки регистров общего назначения.

Общий вид команды: ***MVI R, d8***,

где *R* – идентификатор регистра: *A, B, C, D, E, H, L*, а *d8* – непосредственный операнд (байтовое число).

Пример: Запишите в память, начиная с адреса *0000h*, следующую последовательность команд:

Адрес	Команда	Машинный код	Комментарий
0000	<i>MVI A,00</i>	3E 00	загрузка регистра <i>A←00h</i>
0002	<i>MVI B,01</i>	06 01	загрузка регистра <i>B←01h</i>
0004	<i>MVI C,02</i>	0E 02	загрузка регистра <i>C←02h</i>
0006	<i>MVI D,03</i>	16 03	загрузка регистра <i>D←03h</i>
0008	<i>MVI E,04</i>	1E 04	загрузка регистра <i>E←04h</i>
000A	<i>MVI H,05</i>	26 05	загрузка регистра <i>H←05h</i>
000C	<i>MVI L,06</i>	2E 06	загрузка регистра <i>L←06h</i>

Выполните эту последовательность команд в пошаговом режиме и наблюдайте, как изменяется содержимое регистров общего назначения *A, B, C, D, E, H, L*. Значения регистров должны быть следующими:

$$A = 00h; B = 01h; C = 02h; D = 03h; E = 04h; H = 05h; L = 06h.$$

2.2.2. Команды загрузки регистров 16-разрядными данными.

Общий вид команды: **LXI R, <dh dl>**

где **R** – идентификатор пары регистров: **B, D, H**;

dh – старший байт 16-разрядного операнда;

dl – младший байт 16-разрядного операнда.

Пример: Запишите в память, начиная с адреса **0000h**, коды следующей последовательности команд:

Адрес	Команда	Машинный код	Комментарий
0000	LXI B, 3132h	01 32 31	Загрузка регистровой пары BC числом 3132h . Младший байт данных загружается в регистр C , а старший байт – в регистр B .
0003	LXI D, 3334h	11 34 33	Загрузка регистровой пары DE числом 3334h . Младший байт данных загружается в регистр E , а старший байт – в регистр D .
0006	LXI H, 3536h	21 36 35	Загрузка регистровой пары HL числом 3536h . Младший байт данных загружается в регистр L , а старший байт – в регистр H .

Примечание: в памяти располагается сначала младший байт операнда, затем – старший.

Выполните эту последовательность команд в пошаговом режиме и наблюдайте, как изменяется содержимое регистров **B, C, D, E, H, L**. Значения регистров должны быть следующими:

$$B = 31h; C = 32h; D = 33h; E = 34h; H = 35h; L = 36h.$$

2.2.3. Команды загрузки регистра указателя стека.

Команда непосредственной загрузки регистра указатель стека имеет вид: **LXI SP, <dh dl>**,

где **dh** – старший байт 16-разрядного операнда;

dl – младший байт 16-разрядного операнда.

Пример: Запишите в память, начиная с адреса **0000h**, коды следующих команд:

Адрес	Команда	Машинный код	Комментарий
0000	LXI SP, 0B01h	31 01 0B	загрузка указателя стека: SP←0B01h
0003	LXI SP, 0100h	31 00 01	загрузка указателя стека: SP←0100h

Выполните эту последовательность команд в пошаговом режиме и просмотрите содержание регистра указателя стека после каждого шага выполнения программы.

После первого шага, т.е. после выполнения микропроцессором первой команды, содержимое регистра указателя стека должно быть равным **SP = 0B01h**.

После второго шага, т.е. после выполнения микропроцессором второй команды, содержимое регистра указателя стека должно быть равным **SP = 0100h**.

Команда косвенной загрузки регистра указатель стека имеет вид: **SPHL**

По этой команде в указатель стека загружается содержимое регистровой пары **HL**. Поэтому, чтобы в указатель стека загрузить, например, число **0210h**, его предварительно надо загрузить в регистровую пару **HL**.

Пример: Запишите в память, начиная с адреса **0000h**, коды следующих команд:

Адрес	Команда	Машинный код	Комментарий
0000	LXI H, 0210h	21 10 02	загрузка HL←0210h
0003	SPHL	F9	загрузка SP←HL

Выполните эту последовательность команд в пошаговом режиме и просмотрите содержание регистра указателя стека после выполнения программы. После выполнения содержимое регистра указателя стека должно быть равным **SP = 0210h**.

2.2.4. Команды пересылки.

Общий вид команды: **MOV R1, R2**,

где **R1** – идентификатор регистра получателя: **A, B, C, D, E, H, L**;

R2 – идентификатор регистра источника: **A, B, C, D, E, H, L**.

Пример: Запишите в память, начиная с адреса **0000h**, коды команд:

Адрес	Команда	Машинный код	Комментарий
0000	MVI A, FFh	3E FF	загрузка регистра A←FFh
0002	MOV B, A	47	пересылка B←A
0003	MOV C, B	48	пересылка C←B
0004	MOV D, C	51	пересылка D←C
0005	MOV E, D	5A	пересылка E←D
0006	MOV H, E	63	пересылка H←E
0007	MOV L, H	6C	пересылка L←H

Выполните эту последовательность команд в пошаговом режиме и наблюдайте, как изменяется содержимое регистров **A, B, C, D, E, H, L**. Значения регистров должны быть следующими:

$$A = FFh; B = FFh; C = FFh; D = FFh; E = FFh; H = FFh; L = FFh.$$

2.2.5. Команда загрузки счетчика команд.

Общий вид команды: **PCHL**

По этой команде в счетчик команд записывается содержимое пары регистров **HL**. Т.о., для того, чтобы загрузить в счетчик команд адрес **0100h**, необходимо сначала это число загрузить в регистровую пару **HL**.

Пример: Запишите в память по адресу **0000h** коды следующих команд:

Адрес	Команда	Машинный код	Комментарий
0000	LXI H, 0100h	21 00 01	HL←0100h
0003	PCHL	E9	PC ← HL

Выполните эту последовательность команд в пошаговом режиме и наблюдайте, как изменяется содержимое регистров **H, L** и программного счетчика **PC**. Значения должны быть следующими:

$$H = 01h; L = 00h; PC = 0100h.$$

2.3. КОНТРОЛЬНЫЕ ЗАДАНИЯ

2.3.1. Напишите и выполните программу загрузки регистров:

Вариант	1	2	3	4	5	6
	<i>B:←F0h</i>	<i>B:←01h</i>	<i>B:←AEh</i>	<i>B:←21h</i>	<i>B:←23h</i>	<i>B:←BBh</i>
	<i>C:←33h</i>	<i>C:←35h</i>	<i>C:←FBh</i>	<i>C:←16h</i>	<i>C:←45h</i>	<i>C:←CCh</i>
	<i>D:←EEh</i>	<i>D:←EAh</i>	<i>D:←35h</i>	<i>D:←E1h</i>	<i>D:←10h</i>	<i>D:←D1h</i>
	<i>E:←AAh</i>	<i>E:←A1h</i>	<i>E:←26h</i>	<i>E:←D5h</i>	<i>E:←62h</i>	<i>E:←EEh</i>
	<i>H:←00h</i>	<i>H:←A1h</i>	<i>H:←16h</i>	<i>H:←01h</i>	<i>H:←A5h</i>	<i>H:←AAh</i>
	<i>L:←19h</i>	<i>L:←18h</i>	<i>L:←AAh</i>	<i>L:←20h</i>	<i>L:←97h</i>	<i>L:←FFh</i>
	<i>A:←FFh</i>	<i>A:←F5h</i>	<i>A:←FEh</i>	<i>A:←25h</i>	<i>A:←F1h</i>	<i>A:←43h</i>
Вариант	7	8	9	10	11	12
	<i>B:←23h</i>	<i>B:←23h</i>	<i>B:←0Eh</i>	<i>B:←11h</i>	<i>B:←23h</i>	<i>B:←8Bh</i>
	<i>C:←45h</i>	<i>C:←45h</i>	<i>C:←0Bh</i>	<i>C:←66h</i>	<i>C:←33h</i>	<i>C:←1Ch</i>
	<i>D:←10h</i>	<i>D:←1Ah</i>	<i>D:←05h</i>	<i>D:←11h</i>	<i>D:←15h</i>	<i>D:←11h</i>
	<i>E:←62h</i>	<i>E:←A1h</i>	<i>E:←36h</i>	<i>E:←55h</i>	<i>E:←1Dh</i>	<i>E:←E3h</i>
	<i>H:←A5h</i>	<i>H:←1Bh</i>	<i>H:←06h</i>	<i>H:←00h</i>	<i>H:←D1h</i>	<i>H:←A5h</i>
	<i>L:←97h</i>	<i>L:←19h</i>	<i>L:←0Ah</i>	<i>L:←22h</i>	<i>L:←2Dh</i>	<i>L:←F5h</i>
	<i>A:←95h</i>	<i>A:←53h</i>	<i>A:←0Eh</i>	<i>A:←66h</i>	<i>A:←F3h</i>	<i>A:←44h</i>

Проверьте правильность выполнения программы.

2.3.2. Напишите и выполните программу загрузки регистровых пар:

Вариант	1	2	3
	<i>BC←FFFFh</i>	<i>BC←00FFh</i>	<i>BC←0000h</i>
	<i>DE←0123h</i>	<i>DE←0124h</i>	<i>DE←0F0Fh</i>
	<i>HL←55AAh</i>	<i>HL←5555h</i>	<i>HL←1579h</i>
Вариант	4	5	6
	<i>BC←0E0Eh</i>	<i>BC←1100h</i>	<i>BC←DDDDh</i>
	<i>DE←E0E0h</i>	<i>DE←4545h</i>	<i>DE←ABCDh</i>
	<i>HL←000Eh</i>	<i>HL←536Ah</i>	<i>HL←DCBAh</i>
Вариант	7	8	9
	<i>BC←13EFh</i>	<i>BC←1234h</i>	<i>BC←FEDCh</i>
	<i>DE←A734h</i>	<i>DE←5678h</i>	<i>DE←BA98h</i>
	<i>HL←1FA9h</i>	<i>HL←9ABCh</i>	<i>HL←7654h</i>
Вариант	10	11	12
	<i>BC←3210h</i>	<i>BC←2468h</i>	<i>BC←0000h</i>
	<i>DE←35DFh</i>	<i>DE←ACE2h</i>	<i>DE←1111h</i>
	<i>HL←5555h</i>	<i>HL←468Ah</i>	<i>HL←2222h</i>

Проверьте правильность работы программы.

2.3.3. Напишите и выполните программы загрузки регистра указателя стека с использованием команд непосредственной и косвенной загрузки.

Вариант	1	2	3
	<i>SP←0100h</i>	<i>SP←0101h</i>	<i>SP←0802h</i>
	<i>SP←0200h</i>	<i>SP←0202h</i>	<i>SP←0812h</i>
	<i>SP←0300h</i>	<i>SP←0203h</i>	<i>SP←0822h</i>
Вариант	4	5	6
	<i>SP←0200h</i>	<i>SP←0800h</i>	<i>SP←0702h</i>
	<i>SP←0201h</i>	<i>SP←0900h</i>	<i>SP←0802h</i>
	<i>SP←0202h</i>	<i>SP←0000h</i>	<i>SP←0902h</i>
Вариант	7	8	9
	<i>SP←0200h</i>	<i>SP←000Ah</i>	<i>SP←0110h</i>
	<i>SP←0222h</i>	<i>SP←000Bh</i>	<i>SP←0120h</i>
	<i>SP←0333h</i>	<i>SP←0A0Bh</i>	<i>SP←0130h</i>
Вариант	10	11	12
	<i>SP←0800h</i>	<i>SP←0110h</i>	<i>SP←0115h</i>
	<i>SP←0825h</i>	<i>SP←0111h</i>	<i>SP←0178h</i>
	<i>SP←0850h</i>	<i>SP←0800h</i>	<i>SP←0564h</i>

Проверьте правильность работы программы.

2.3.4. Напишите и выполните программу пересылки, предварительно загрузив регистры

Вариант	1	2	3
	<i>B←A (значением 00h)</i>	<i>D←H (значением 01h)</i>	<i>B←C (значением 10h)</i>
	<i>C←L (значением 0Eh)</i>	<i>E←C (значением 02h)</i>	<i>C←A (значением 20h)</i>
	<i>H←B (значением 0Fh)</i>	<i>B←L (значением 03h)</i>	<i>A←B (значением 30h)</i>
Вариант	4	5	6
	<i>L←H (значением 0Fh)</i>	<i>H←A (значением FFh)</i>	<i>C←B (значением 15h)</i>
	<i>C←A (значением 0Eh)</i>	<i>E←A (значением EEh)</i>	<i>C←H (значением 25h)</i>
	<i>D←E (значением 0Bh)</i>	<i>B←C (значением D1h)</i>	<i>A←L (значением 35h)</i>
Вариант	7	8	9
	<i>A←H (значением 22h)</i>	<i>B←C (значением 00h)</i>	<i>B←C (значением 15h)</i>
	<i>C←B (значением 11h)</i>	<i>D←E (значением 0Eh)</i>	<i>C←A (значением 20h)</i>
	<i>D←L (значением 10h)</i>	<i>H←L (значением 0Fh)</i>	<i>A←B (значением 2Ah)</i>
Вариант	10	11	12
	<i>H←A (значением 00h)</i>	<i>B←A (значением AAh)</i>	<i>A←H (значением 21h)</i>
	<i>E←A (значением 01h)</i>	<i>C←L (значением 0Ah)</i>	<i>C←B (значением 22h)</i>
	<i>B←C (значением 05h)</i>	<i>H←B (значением 00h)</i>	<i>D←L (значением 23h)</i>

Проверьте правильность выполнения программы.

2.3.5. Напишите и выполните программу перехода с адреса *0000h* на адреса

Вариант	1	2	3
	<i>0100h</i>	<i>0050h</i>	<i>0080h</i>
	<i>0200h</i>	<i>0060h</i>	<i>0090h</i>
	<i>0300h</i>	<i>0070h</i>	<i>0100h</i>
Вариант	4	5	6
	<i>0100h</i>	<i>0800h</i>	<i>0900h</i>
	<i>0200h</i>	<i>0200h</i>	<i>0200h</i>
	<i>0800h</i>	<i>0100h</i>	<i>0100h</i>
Вариант	7	8	9
	<i>0100h</i>	<i>0200h</i>	<i>0100h</i>
	<i>0200h</i>	<i>0250h</i>	<i>0200h</i>
	<i>0800h</i>	<i>0700h</i>	<i>0150h</i>
Вариант	10	11	12
	<i>0110h</i>	<i>0100h</i>	<i>0050h</i>
	<i>0210h</i>	<i>0150h</i>	<i>0100h</i>
	<i>0250h</i>	<i>0700h</i>	<i>0150h</i>

Проверьте правильность выполнения программы.

Лабораторная работа №3

МЕТОДЫ АДРЕСАЦИИ ПАМЯТИ. КОМАНДЫ РАБОТЫ С ПАМЯТЬЮ

3.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Память представляется как последовательность ячеек размером в один байт. Каждая ячейка имеет свой адрес в диапазоне от 0 до 65535. Для удобства обычно используется шестнадцатеричное значение адреса, тогда диапазон адресации составляет **0000h – FFFFh**.

В микропроцессорной системе адресации адрес ячейки памяти указывается в самой команде во втором и третьем байтах команды (прямая адресация). В общем виде это выглядит следующим образом

КОП *ad16*

где КОП – код операции (чтение или запись); ***ad16*** – адрес ячейки памяти.

В памяти такая команда будет размещена следующим образом

КОП *ad16* (младший байт) *ad16* (старший байт)

т.е. после байта кода операции располагается сначала младший байт адреса, а затем – старший.

Косвенная адресация предполагает, что адрес ячейки памяти будет располагаться в регистровых парах ***HL, DE, BC***. Для каждой конкретной команды работы с памятью закреплена своя регистровая пара. Таким образом, прежде чем выполнить такую команду необходимо сначала задать адрес в соответствующей регистровой паре.

Например,

LXI H, 0800h

MOV M, A

;запись в память содержимое регистра *A* по адресу,
;находящемуся в регистровой паре ***HL***

или

LXI D, 0900H

STAX D

;запись в память содержимое регистра *A* по адресу,
;находящемуся в регистровой паре ***DE***

3.2. ПРОГРАММА РАБОТЫ

3.2.1. Команды записи в память с прямой адресацией.

Существуют две команды прямой адресации записи в память:

STA ad16 запись в память по прямому адресу ***ad16*** содержимого регистра *A*;

SHLD ad16 запись в память содержимого регистровой пары ***HL***. Причем по адресу ***ad16*** будет записано содержимое регистра *L*, а по адресу ***ad16+1*** будет записано содержимое регистра *H*.

Пример: Запишите в память, начиная с адреса **0000h**, коды следующих команд, используя прямую адресацию

Адрес	Команда	Машинный код	Комментарий
0000	<i>MVI A, FFh</i>	3E FF	Запись в аккумулятор значения <i>FFh</i>
0002	<i>STA 0110h</i>	32 10 01	Запись в память содержимого регистра <i>A</i> по адресу <i>0110h</i>
0005	<i>LXI H, 3536h</i>	21 36 35	Загрузка регистровой пары <i>HL</i> числом <i>3536h</i> . Младший байт данных загружается в регистр <i>L</i> , а старший байт – в регистр <i>H</i> .
0008	<i>SHLD 0150h</i>	22 50 01	Запись в память содержимого регистра <i>L</i> по адресу <i>0150h</i> , содержимого регистра <i>H</i> по адресу <i>0151h</i>

Выполните эту последовательность команд в пошаговом режиме и наблюдайте, как изменяется содержимое регистров *A*, *H*, *L* и содержимое ячеек памяти *0110h*, *0150h*, *0151h*. Значения регистров и ячеек памяти должны быть следующими

$$A = FFh; H = 35h; L = 36h; (0110h) = FFh; (0150h) = 36h; (0151h) = 35h$$

3.2.2. Команды чтения памяти с прямой адресацией.

Аналогично командам записи с прямой адресацией существуют две команды чтения памяти с конкретным адресом

LDA *ad16* загрузка регистра *A* из ячейки памяти с адресом *ad16*;

LHLD *ad16* чтение памяти по прямому адресу *ad16* в регистровую пару **HL**. При этом в регистр *H* будет записано содержимое ячейки с адресом *ad16+1*, а в регистр *L* содержимое ячейки памяти с адресом *ad16*.

Пример: Запишите в память по адресу *0000h* коды следующих команд

Адрес	Команда	Машинный код	Комментарий
0000	LDA 0190h	3A 90 01	Чтение в регистр <i>A</i> содержимого ячейки с адресом <i>0190h</i>
0003	LHLD 0190h	2A 90 01	Чтение в регистр <i>L</i> содержимого ячейки с адресом <i>0190h</i> , а в регистр <i>H</i> содержимого ячейки с адресом <i>0191h</i>

Вручную внесите в ячейки памяти следующие значения

$$(0190h) = ABh; (0191h) = CDh.$$

Выполните эту последовательность команд в пошаговом режиме и наблюдайте, как изменяется содержимое регистров *A*, *H*, *L*. Значения регистров должны быть следующими:

$$A = ABh; H = CDh; L = ABh.$$

3.2.3. Команды чтения/записи в память с косвенной адресацией.

Общий вид команды

MOV *M*, *R* запись в память содержимого регистра;

MOV *R*, *M* загрузка регистра из ячейки памяти, адрес которой находится в регистровой паре **HL**. *R* – регистр общего назначения *A*, *B*, *C*, *D*, *E*, *H*, *L*.

Пример: Запишите в память, начиная с адреса *0000h*, коды следующей программы

Адрес	Команда	Машинный код	Комментарий
0000	MVI A, 0AAh	3E AA	загрузка регистров
0002	MVI B, 0BBh	06 BB	
0004	MVI C, 0CCh	0E CC	
0006	MVI D, 0DDh	16 DD	
0008	MVI E, 0EEh	1E EE	
000A	LXI H, 0100h	21 00 01	; загрузка HL=0100h , адрес <i>M</i>
000D	MOV M, A	77	; запись в <i>M=A</i> , по адресу HL
000E	LXI H, 0101h	21 01 01	
0011	MOV M, C	71	
0012	LXI H, 0102h	21 02 01	
0015	MOV M, B	70	

0016	<i>LXI H, 0103h</i>	21 03 01	
0019	<i>MOV M, E</i>	73	
001A	<i>LXI H, 0104h</i>	21 04 01	
001D	<i>MOV M, D</i>	72	
001E	<i>LXI H, 0105h</i>	21 05 01	
0021	<i>MOV M, H</i>	74	
0022	<i>LXI H, 0106h</i>	21 06 01	
0025	<i>MOV M, L</i>	75	

Выполните эту последовательность команд. Значения ячеек памяти должны быть следующими

*0100h = AAh; 0101h = CCh; 0102h = BBh; 0103h = EEh; 0104h = DDh; 0105h = 01h;
0106h = 06h*

Пример: Запишите в память, начиная с адреса *0000h*, коды следующей программы

Адрес	Команда	Машинный код	Комментарий
0000	<i>LXI H, 0100h</i>	21 00 01	загрузка <i>HL=0100h</i> , адрес <i>M</i>
0003	<i>MOV E, M</i>	5E	чтение <i>E=M</i> , по адресу <i>HL</i>
0004	<i>LXI H, 0101h</i>	21 01 01	и т.д.
0007	<i>MOV D, M</i>	56	
0008	<i>LXI H, 0102h</i>	21 02 01	
000B	<i>MOV C, M</i>	4E	
000C	<i>LXI H, 0103h</i>	21 03 01	
000F	<i>MOV B, M</i>	46	
0010	<i>LXI H, 0104h</i>	21 04 01	
0013	<i>MOV A, M</i>	7E	
0014	<i>LXI H, 0105h</i>	21 05 01	
0017	<i>MOV H, M</i>	66	
0018	<i>LXI H, 0106h</i>	21 06 01	
001B	<i>MOV L, M</i>	6E	

Заполните вручную соответствующие ячейки памяти (*0100h = AAh, 0101h = CCh, 0102h = BBh, 0103h = EEh, 0104h = DDh, 0105h = 01h, 0106h = 06h*). Выполните эту последовательность команд. Значения регистров должны быть следующими

A = DDh; B = EEh; C = BBh; D = CCh; E = AAh; H = 01h; L = 06h

3.2.4. Команды чтения/записи при адресации через регистровые пары *BC, DE*.

STAX B запись содержимого регистра *A* в память, адрес в регистровой паре *BC*;
STAX D запись содержимого регистра *A* в память, адрес в регистровой паре *DE*;
LDAX B чтение содержимого памяти в регистр *A*, адрес в регистровой паре *BC*;
LDAX D чтение содержимого памяти в регистр *A*, адрес в регистровой паре *DE*.

Пример: Запишите в память, начиная с адреса 0000h, коды программы

Адрес	Команда	Машинный код	Комментарий
0000	LXI B, 0100h	01 00 01	Загрузка BC←0100h
0003	MVI A, 0Fh	3E 0F	Загрузка A←0Fh
0005	STAX B	02	Запись в M←A по адресу BC
0006	LXI D, 0110h	11 10 01	Загрузка в DE←0110h
0009	MVI A, 0F0h	3E F0	Загрузка в A←F0h
000B	STAX D	12	Запись в M←A по адресу DE

Выполните эту последовательность команд. Значения ячеек памяти должны быть следующими

$$0100h = 0Fh, 0110h = F0h.$$

Пример: Запишите в память, начиная с адреса 0000h, коды программы

Адрес	Команда	Машинный код	Комментарий
0000	LXI D, 0100h	11 00 09	загрузка в DE←0100h
0003	LDAX D	1A	чтение в A←M по адресу DE
0004	MOVL, A	6F	пересылка L←A
0005	LXI B, 0110h	01 10 09	загрузка в BC←0110h
0008	LDAX B	0A	чтение в A←M по адресу BC
0009	MOVH, A	67	;пересылка H←A

Заполните вручную соответствующие ячейки памяти (**0100h←0Fh, 0110h←F0h**).
Выполните эту последовательность команд. Значения регистров должны быть следующими
H←F0h, L←0Fh.

3.3. КОНТРОЛЬНЫЕ ЗАДАНИЯ

3.3.1. Напишите и выполните программу записи данных в память из регистра **A**, в соответствии с таблицей. Для этого используйте команду загрузки регистра **A** и команду записи в память регистра **A** по прямому адресу.

Вариант 1	Адрес	0100	0105	0107	010C	0120	0126
	Данные	00h	01h	05h	0Ah	BBh	12h
Вариант 2	Адрес	0200	0202	0205	020C	0215	0220
	Данные	25h	12h	50h	A0h	BCh	1Dh
Вариант 3	Адрес	0101	0110	0120	0130	0135	0140
	Данные	15h	A1h	5Ah	A6h	00h	21h
Вариант 4	Адрес	0100	0105	0107	010B	0123	0126
	Данные	22h	33h	44h	55h	AAh	CCh
Вариант 5	Адрес	0200	0210	0220	0226	0228	0240
	Данные	FFh	10h	01h	03h	0Bh	12h
Вариант 6	Адрес	0100	0105	010C	0115	0120	0130
	Данные	66h	12h	05h	0Ah	D0h	D3h

Проверьте правильность работы программы.

3.3.2. Напишите и выполните программу записи данных из регистровой пары **HL** в соответствии с таблицей. Для этого используйте команду загрузки регистровой пары **HL** и команду записи в память регистровой пары **HL** по прямому адресу.

Вариант 1	Адрес	<i>0100</i>	<i>0101</i>	<i>0111</i>	<i>0112</i>	<i>0120</i>	<i>0121</i>
	Данные	00h	07h	09h	0Bh	B0h	12h
Вариант 2	Адрес	<i>0200</i>	<i>0201</i>	<i>0205</i>	<i>0206</i>	<i>0210</i>	<i>0211</i>
	Данные	25h	12h	50h	0Ah	0Ch	D1h
Вариант 3	Адрес	<i>0101</i>	<i>0102</i>	<i>0120</i>	<i>0121</i>	<i>0135</i>	<i>0136</i>
	Данные	15h	1Ah	A5h	6Ah	00h	21h
Вариант 4	Адрес	<i>0100</i>	<i>0101</i>	<i>0107</i>	<i>0108</i>	<i>0123</i>	<i>0124</i>
	Данные	22h	33h	44h	55h	AAh	CCh
Вариант 5	Адрес	<i>0210</i>	<i>0211</i>	<i>0220</i>	<i>0221</i>	<i>0228</i>	<i>0229</i>
	Данные	FFh	1Ah	A1h	03h	B0h	12h
Вариант 6	Адрес	<i>0150</i>	<i>0151</i>	<i>0161</i>	<i>0162</i>	<i>0200</i>	<i>0201</i>
	Данные	66h	B2h	05h	A0h	0Dh	3Dh

Проверьте правильность работы программы.

3.3.3. Напишите и выполните программу записи данных в память в соответствии с таблицей. Для этого используйте команду загрузки регистровой пары **HL** и регистра **A** и команды записи в память регистровой пары **HL** и регистра **A** по прямому адресу.

Вариант 1	Адрес	<i>0100</i>	<i>0101</i>	<i>0111</i>	<i>0115</i>	<i>0120</i>	<i>0121</i>
	Данные	22h	17h	09h	0Ah	80h	90h
Вариант 2	Адрес	<i>0150</i>	<i>0201</i>	<i>0202</i>	<i>0210</i>	<i>0212</i>	<i>0213</i>
	Данные	35h	42h	1Ah	0Ah	C0h	D3h
Вариант 3	Адрес	<i>0101</i>	<i>0102</i>	<i>0120</i>	<i>0125</i>	<i>0135</i>	<i>0136</i>
	Данные	16h	1Ah	50h	60h	0Ah	2Dh
Вариант 4	Адрес	<i>0100</i>	<i>0105</i>	<i>0107</i>	<i>0108</i>	<i>0110</i>	<i>0124</i>
	Данные	25h	13h	43h	56h	0Ah	C0h
Вариант 5	Адрес	<i>0210</i>	<i>0211</i>	<i>0220</i>	<i>0221</i>	<i>0228</i>	<i>0240</i>
	Данные	0Fh	10h	12h	D3h	B1h	1Ch
Вариант 6	Адрес	<i>0150</i>	<i>0151</i>	<i>0160</i>	<i>0162</i>	<i>0200</i>	<i>0201</i>
	Данные	65h	B4h	05h	A1h	0Ch	35h

Проверьте правильность работы программы.

3.3.4. Напишите и выполните программу загрузки регистров **B, C, D, E, H, L** из памяти в соответствии с таблицей. Используйте команды чтения памяти в регистр **A** по непосредственному адресу и команды пересылки. Значение соответствующих ячеек памяти необходимо заполнить вручную.

Вариант 1	Адрес	<i>0100h</i>	<i>0105h</i>	<i>0107h</i>	<i>010Ch</i>	<i>0120h</i>	<i>0126h</i>
	Данные	00h	01h	15h	1Ah	1Bh	22h
	Регистр	B	C	D	E	H	L

Вариант 2	Адрес	<i>0200h</i>	<i>0205h</i>	<i>0215h</i>	<i>020Ch</i>	<i>0215h</i>	<i>0220h</i>
	Данные	35h	26h	51h	A1h	B0h	11h
	Регистр	C	B	D	H	E	L
Вариант 3	Адрес	<i>0101h</i>	<i>0110h</i>	<i>0125h</i>	<i>0133h</i>	<i>0135h</i>	<i>0145h</i>
	Данные	16h	19h	50h	A6h	0Bh	21h
	Регистр	B	D	C	E	H	L
Вариант 4	Адрес	<i>0100h</i>	<i>0105h</i>	<i>0110h</i>	<i>0111h</i>	<i>0112h</i>	<i>0126h</i>
	Данные	21h	35h	44h	56h	0Ah	C0h
	Регистр	B	C	D	E	H	L
Вариант 5	Адрес	<i>0200h</i>	<i>0211h</i>	<i>0221h</i>	<i>0226h</i>	<i>0230h</i>	<i>0240h</i>
	Данные	F0h	10h	19h	13h	B0h	23h
	Регистр	C	B	D	H	E	L
Вариант 6	Адрес	<i>0100h</i>	<i>0105h</i>	<i>010Ch</i>	<i>0115h</i>	<i>0120h</i>	<i>0130h</i>
	Данные	69h	37h	56h	0Ah	D6h	03h
	Регистр	L	D	C	E	H	B

Проверьте правильность работы программы.

3.3.5. Напишите и выполните программу загрузки регистров **B, C, D, E, H, L** из памяти в соответствии с таблицей. Используйте команды чтения памяти в регистровую пару **HL** и команды пересылки. Значение соответствующих ячеек памяти необходимо заполнить вручную.

Вариант 1	Адрес	<i>0100h</i>	<i>0101h</i>	<i>0108h</i>	<i>0109h</i>	<i>0121h</i>	<i>0122h</i>
	Данные	01h	05h	25h	10h	1Bh	2Ah
	Регистр	B	C	D	E	L	H
Вариант 2	Адрес	<i>0200h</i>	<i>0201h</i>	<i>0215h</i>	<i>0216h</i>	<i>0220h</i>	<i>0221h</i>
	Данные	35h	26h	51h	A1h	B0h	11h
	Регистр	B	C	D	E	L	H
Вариант 3	Адрес	<i>0101h</i>	<i>0102h</i>	<i>0125h</i>	<i>0126h</i>	<i>0135h</i>	<i>0136h</i>
	Данные	16h	29h	60h	A7h	3Bh	A1h
	Регистр	B	C	D	E	L	H
Вариант 4	Адрес	<i>0110h</i>	<i>0111h</i>	<i>0150h</i>	<i>0151h</i>	<i>0212h</i>	<i>0213h</i>
	Данные	99h	35h	88h	56h	5Ah	0Ch
	Регистр	B	C	D	E	L	H
Вариант 5	Адрес	<i>0200h</i>	<i>0201</i>	<i>0221</i>	<i>0222h</i>	<i>0230h</i>	<i>0231</i>
	Данные	F5h	EEh	1Eh	1Ah	BBh	33h
	Регистр	B	C	D	E	L	H
Вариант 6	Адрес	<i>0105h</i>	<i>0106h</i>	<i>010Ch</i>	<i>010Dh</i>	<i>0183h</i>	<i>0184h</i>
	Данные	19h	38h	96h	A1h	16h	13h
	Регистр	B	C	D	E	L	H

Проверьте правильность работы программы.

3.3.6. Напишите и выполните программу перезаписи данных из одних ячеек памяти (адрес 1) в другие (адрес 2) в соответствии с таблицей. Значение соответствующих ячеек памяти необходимо заполнить вручную.

Вариант 1	Адрес 1	<i>0100h</i>	<i>0101h</i>	<i>0108h</i>	<i>0109h</i>	<i>0121h</i>	<i>0125h</i>
	Адрес 2	<i>0200h</i>	<i>0201h</i>	<i>0208h</i>	<i>0209h</i>	<i>0235h</i>	<i>0240h</i>
	Данные	<i>11h</i>	<i>51h</i>	<i>25h</i>	<i>1Ah</i>	<i>10h</i>	<i>2Bh</i>
Вариант 2	Адрес 1	<i>0200h</i>	<i>0201h</i>	<i>0215h</i>	<i>0216h</i>	<i>0220h</i>	<i>0229h</i>
	Адрес 2	<i>0250h</i>	<i>0251h</i>	<i>0208h</i>	<i>0209h</i>	<i>0135h</i>	<i>0140h</i>
	Данные	<i>35h</i>	<i>26h</i>	<i>51h</i>	<i>A1h</i>	<i>B0h</i>	<i>11h</i>
Вариант 3	Адрес 1	<i>0101h</i>	<i>0102h</i>	<i>0120h</i>	<i>0126h</i>	<i>0135h</i>	<i>0136h</i>
	Адрес 2	<i>0300h</i>	<i>0301h</i>	<i>0308h</i>	<i>0310h</i>	<i>0335h</i>	<i>0336h</i>
	Данные	<i>26h</i>	<i>29h</i>	<i>61h</i>	<i>A8h</i>	<i>4Bh</i>	<i>11h</i>
Вариант 4	Адрес 1	<i>0110h</i>	<i>0111h</i>	<i>0150h</i>	<i>0180h</i>	<i>0212h</i>	<i>0213h</i>
	Адрес 2	<i>0200h</i>	<i>0201h</i>	<i>0205h</i>	<i>0207h</i>	<i>0250h</i>	<i>0251h</i>
	Данные	<i>9Ah</i>	<i>35h</i>	<i>90h</i>	<i>58h</i>	<i>6Ah</i>	<i>1Ch</i>
Вариант 5	Адрес 1	<i>0200</i>	<i>0201</i>	<i>0221</i>	<i>0222h</i>	<i>0230h</i>	<i>0240h</i>
	Адрес 2	<i>0100</i>	<i>0101</i>	<i>0121</i>	<i>0122h</i>	<i>0140h</i>	<i>0150h</i>
	Данные	<i>F7h</i>	<i>E0h</i>	<i>10h</i>	<i>A6h</i>	<i>0Bh</i>	<i>33h</i>
Вариант 6	Адрес 1	<i>0105h</i>	<i>0106h</i>	<i>010Ch</i>	<i>010Dh</i>	<i>0180h</i>	<i>0190h</i>
	Адрес 2	<i>0210h</i>	<i>0211h</i>	<i>0220h</i>	<i>0221h</i>	<i>0290h</i>	<i>0300h</i>
	Данные	<i>1Ah</i>	<i>3Dh</i>	<i>96h</i>	<i>A1h</i>	<i>1Eh</i>	<i>1Dh</i>

Проверьте правильность работы программы.

3.3.7. Напишите и выполните программу записи в память содержимого регистров в соответствии с таблицей, используя команды записи в память с косвенной адресацией (через регистровую пару **HL**).

	Регистры	A	B	C	D	E	H	L
Вариант 1	Адрес	<i>0100h</i>	<i>0101h</i>	<i>0102h</i>	<i>0103h</i>	<i>0104h</i>	<i>0105h</i>	<i>0106h</i>
	Данные	<i>FFh</i>	<i>EEh</i>	<i>BBh</i>	<i>00h</i>	<i>AAh</i>	<i>01h</i>	<i>06h</i>
Вариант 2	Адрес	<i>0200h</i>	<i>0201h</i>	<i>0202h</i>	<i>0203h</i>	<i>0204h</i>	<i>0205h</i>	<i>0206h</i>
	Данные	<i>00h</i>	<i>E1h</i>	<i>B1h</i>	<i>01h</i>	<i>A1h</i>	<i>02h</i>	<i>06h</i>
Вариант 3	Адрес	<i>0110h</i>	<i>0111h</i>	<i>0112h</i>	<i>0113h</i>	<i>0114h</i>	<i>0115h</i>	<i>0116h</i>
	Данные	<i>34h</i>	<i>12h</i>	<i>B0h</i>	<i>50h</i>	<i>A5h</i>	<i>01h</i>	<i>16h</i>
Вариант 4	Адрес	<i>0125h</i>	<i>0126h</i>	<i>0127h</i>	<i>0128h</i>	<i>0129h</i>	<i>0130h</i>	<i>0131h</i>
	Данные	<i>99h</i>	<i>E1h</i>	<i>0Bh</i>	<i>05h</i>	<i>00h</i>	<i>01h</i>	<i>31h</i>
Вариант 5	Адрес	<i>0130h</i>	<i>0131h</i>	<i>0132h</i>	<i>0133h</i>	<i>0134h</i>	<i>0135h</i>	<i>0136h</i>
	Данные	<i>22h</i>	<i>E7h</i>	<i>05h</i>	<i>06h</i>	<i>05h</i>	<i>01h</i>	<i>36h</i>
Вариант 6	Адрес	<i>0100h</i>	<i>0101h</i>	<i>0102h</i>	<i>0103h</i>	<i>0104h</i>	<i>0105h</i>	<i>0106h</i>
	Данные	<i>98h</i>	<i>10h</i>	<i>20h</i>	<i>30h</i>	<i>22h</i>	<i>01h</i>	<i>06h</i>

Проверьте правильность работы программы.

3.3.8. Напишите и выполните программу чтения содержимого памяти в соответствующие регистры, используя команды чтения из памяти с косвенной адресацией (через регистровую пару **HL**).

	Регистры	D	B	C	A	E	H	L
Вариант 1	Адрес	<i>0100h</i>	<i>0101h</i>	<i>0102h</i>	<i>0103h</i>	<i>0104h</i>	<i>0105h</i>	<i>0106h</i>
	Данные	<i>FFh</i>	<i>EEh</i>	<i>BBh</i>	<i>00h</i>	<i>AAh</i>	<i>01h</i>	<i>06h</i>
Вариант 2	Адрес	<i>0200h</i>	<i>0201h</i>	<i>0202h</i>	<i>0203h</i>	<i>0204h</i>	<i>0205h</i>	<i>0206h</i>
	Данные	<i>00h</i>	<i>E1h</i>	<i>B1h</i>	<i>01h</i>	<i>A1h</i>	<i>02h</i>	<i>06h</i>
Вариант 3	Адрес	<i>0110h</i>	<i>0111h</i>	<i>0112h</i>	<i>0113h</i>	<i>0114h</i>	<i>0115h</i>	<i>0116h</i>
	Данные	<i>34h</i>	<i>12h</i>	<i>B0h</i>	<i>50h</i>	<i>A5h</i>	<i>01h</i>	<i>16h</i>
Вариант 4	Адрес	<i>0125h</i>	<i>0126h</i>	<i>0127h</i>	<i>0128h</i>	<i>0129h</i>	<i>0130h</i>	<i>0131h</i>
	Данные	<i>99h</i>	<i>E1h</i>	<i>0Bh</i>	<i>05h</i>	<i>00h</i>	<i>01h</i>	<i>31h</i>
Вариант 5	Адрес	<i>0130h</i>	<i>0131h</i>	<i>0132h</i>	<i>0133h</i>	<i>0134h</i>	<i>0135h</i>	<i>0136h</i>
	Данные	<i>22h</i>	<i>E7h</i>	<i>05h</i>	<i>06h</i>	<i>05h</i>	<i>01h</i>	<i>36h</i>
Вариант 6	Адрес	<i>0100h</i>	<i>0101h</i>	<i>0102h</i>	<i>0103h</i>	<i>0104h</i>	<i>0105h</i>	<i>0106h</i>
	Данные	<i>98h</i>	<i>10h</i>	<i>20h</i>	<i>30h</i>	<i>22h</i>	<i>01h</i>	<i>06h</i>

Проверьте правильность работы программы.

3.3.9. Напишите и выполните программу записи данных в две области памяти, используя для адресации регистровую пару **BC** и регистровую пару **DE** в соответствии с таблицей.

Вариант 1	Адрес 1	<i>0100h</i>	<i>0101h</i>	<i>0108h</i>	<i>0109h</i>	<i>0121h</i>	<i>0125h</i>
	Адрес 2	<i>0200h</i>	<i>0201h</i>	<i>0208h</i>	<i>0209h</i>	<i>0235h</i>	<i>0240h</i>
	Данные	<i>11h</i>	<i>51h</i>	<i>25h</i>	<i>1Ah</i>	<i>10h</i>	<i>2Bh</i>
Вариант 2	Адрес 1	<i>0200h</i>	<i>0201h</i>	<i>0215h</i>	<i>0216h</i>	<i>0220h</i>	<i>0229h</i>
	Адрес 2	<i>0250h</i>	<i>0251h</i>	<i>0208h</i>	<i>0209h</i>	<i>0135h</i>	<i>0140h</i>
	Данные	<i>35h</i>	<i>26h</i>	<i>51h</i>	<i>A1h</i>	<i>B0h</i>	<i>11h</i>
Вариант 3	Адрес 1	<i>0101h</i>	<i>0102h</i>	<i>0120h</i>	<i>0126h</i>	<i>0135h</i>	<i>0136h</i>
	Адрес 2	<i>0300h</i>	<i>0301h</i>	<i>0308h</i>	<i>0310h</i>	<i>0335h</i>	<i>0336h</i>
	Данные	<i>26h</i>	<i>29h</i>	<i>61h</i>	<i>A8h</i>	<i>4Bh</i>	<i>11h</i>
Вариант 4	Адрес 1	<i>0110h</i>	<i>0111h</i>	<i>0150h</i>	<i>0180h</i>	<i>0212h</i>	<i>0213h</i>
	Адрес 2	<i>0200h</i>	<i>0201h</i>	<i>0205h</i>	<i>0207h</i>	<i>0250h</i>	<i>0251h</i>
	Данные	<i>9Ah</i>	<i>35h</i>	<i>90h</i>	<i>58h</i>	<i>6Ah</i>	<i>1Ch</i>
Вариант 5	Адрес 1	<i>0200h</i>	<i>0201h</i>	<i>0221h</i>	<i>0222h</i>	<i>0230h</i>	<i>0240h</i>
	Адрес 2	<i>0100h</i>	<i>0101h</i>	<i>0121h</i>	<i>0122h</i>	<i>0140h</i>	<i>0150h</i>
	Данные	<i>F7h</i>	<i>E0h</i>	<i>10h</i>	<i>A6h</i>	<i>0Bh</i>	<i>33h</i>
Вариант 6	Адрес 1	<i>0105h</i>	<i>0106h</i>	<i>010Ch</i>	<i>010Dh</i>	<i>0180h</i>	<i>0190h</i>
	Адрес 2	<i>0210h</i>	<i>0211h</i>	<i>0220h</i>	<i>0221h</i>	<i>0290h</i>	<i>0300h</i>
	Данные	<i>1Ah</i>	<i>3Dh</i>	<i>96h</i>	<i>A1h</i>	<i>1Eh</i>	<i>1Dh</i>

Проверьте правильность работы программы.

3.3.10. Напишите и выполните программу перезаписи данных из одной области памяти (адресуйте через **BC**) в другую область памяти (адресуйте через **DE**) в соответствии с таблицей. Значение соответствующих ячеек памяти необходимо заполнить вручную.

Вариант 1	Адрес 1	<i>0100h</i>	<i>0105h</i>	<i>0108h</i>	<i>0109h</i>	<i>0121h</i>	<i>0125h</i>
	Адрес 2	<i>0200h</i>	<i>0205h</i>	<i>0208h</i>	<i>0209h</i>	<i>0221h</i>	<i>0225h</i>
	Данные	<i>11h</i>	<i>51h</i>	<i>25h</i>	<i>1Ah</i>	<i>10h</i>	<i>2Bh</i>
Вариант 2	Адрес 1	<i>0200h</i>	<i>0201h</i>	<i>0215h</i>	<i>0216h</i>	<i>0220h</i>	<i>0229h</i>
	Адрес 2	<i>0100h</i>	<i>0101h</i>	<i>0115h</i>	<i>0116h</i>	<i>0120h</i>	<i>0129h</i>
	Данные	<i>35h</i>	<i>26h</i>	<i>51h</i>	<i>A1h</i>	<i>B0h</i>	<i>11h</i>
Вариант 3	Адрес 1	<i>0101h</i>	<i>0102h</i>	<i>0120h</i>	<i>0126h</i>	<i>0135h</i>	<i>0136h</i>
	Адрес 2	<i>0301h</i>	<i>0302h</i>	<i>0320h</i>	<i>0326h</i>	<i>0335h</i>	<i>0336h</i>
	Данные	<i>26h</i>	<i>29h</i>	<i>61h</i>	<i>A8h</i>	<i>4Bh</i>	<i>11h</i>
Вариант 4	Адрес 1	<i>0110h</i>	<i>0120h</i>	<i>0150h</i>	<i>0180h</i>	<i>0212h</i>	<i>0213h</i>
	Адрес 2	<i>0210h</i>	<i>0220h</i>	<i>0050h</i>	<i>0080h</i>	<i>0112h</i>	<i>0113h</i>
	Данные	<i>9Ah</i>	<i>35h</i>	<i>90h</i>	<i>58h</i>	<i>6Ah</i>	<i>1Ch</i>
Вариант 5	Адрес 1	<i>0200h</i>	<i>0201h</i>	<i>0221h</i>	<i>0222h</i>	<i>0230h</i>	<i>0240h</i>
	Адрес 2	<i>0100h</i>	<i>0101h</i>	<i>0121h</i>	<i>0122h</i>	<i>0130h</i>	<i>0140h</i>
	Данные	<i>F7h</i>	<i>E0h</i>	<i>10h</i>	<i>A6h</i>	<i>0Bh</i>	<i>33h</i>
Вариант 6	Адрес 1	<i>0105h</i>	<i>0106h</i>	<i>010Ch</i>	<i>010Dh</i>	<i>0180h</i>	<i>0190h</i>
	Адрес 2	<i>0205h</i>	<i>0206h</i>	<i>020Ch</i>	<i>020Dh</i>	<i>0280h</i>	<i>0290h</i>
	Данные	<i>1Ah</i>	<i>3Dh</i>	<i>96h</i>	<i>A1h</i>	<i>1Eh</i>	<i>1Dh</i>

Проверьте правильность работы программы.

Лабораторная работа №4

АРИФМЕТИЧЕСКИЕ КОМАНДЫ

4.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

В микропроцессоре *Intel 8080* предусмотрены следующие команды двоичной арифметики: сложение 8-разрядных чисел; сложение 16-разрядных чисел; вычитание 4-разрядных чисел; инкремент; декремент.

Все арифметические операции с 8-разрядными операндами предполагает, что один из операторов размещается в регистре аккумулятора, а другой либо в регистре, либо в памяти (при этом адрес ячейки задается в регистровой паре *HL*), либо является непосредственным числом, заданным в самой команде. Вычитания производятся всегда из регистра аккумулятора. Результат арифметической операции записывается в аккумуляторе. Кроме того, по результату арифметических операций сложения и вычитания устанавливаются биты признаков *C* – переноса, *Z* – нуля, *S* – знака, *P* – четности, *AC* – вспомогательного переноса.

Команды сложения 16-разрядных чисел, так называемой удвоенной точности, предусматривают, что один из операндов находится в регистровой паре *HL*, а второй – либо, в *DE*, либо в *BC*. Результат записывается в *HL*. Кроме того, по результату операции устанавливается либо сбрасывается бит переноса — *C*.

Команды инкремента увеличивают содержимое регистров, ячейки памяти по адресу в *HL* и регистровых пар на 1. Команда инкремент регистра и памяти изменяет биты признаков *Z, S, P, AC*. Инкремент регистровых пар не затрагивает биты признаков.

Команды декремента уменьшают содержимое регистров, ячейки памяти по адреса в *HL* и регистровых пар на 1. Затрагиваемые биты признаков аналогичны команде инкремент.

4.2. ПРОГРАММА РАБОТЫ

4.2.1. Команды сложения 8-разрядных чисел.

ADD R сложение аккумулятора с содержимым одного из регистров *B, C, D, E, H, L*.

ADD M сложение аккумулятора с содержимым ячейки памяти (адрес в *HL*).

ADI d8 сложение аккумулятора с непосредственным числом *d8*.

ADC R сложение аккумулятора с содержимым одного из регистров и бита переноса *C*.

ADC M сложение аккумулятора с содержимым ячейки памяти (адрес *HL*) и бита переноса *C*.

ACI d8 сложение аккумулятора с непосредственным числом *d8* и бита переноса *C*.

Пример: Запишите в памяти, начиная с адреса *0000h*, коды программы, реализующей

$$A=A+B+(M)+I$$

Адрес	Команда	Машинный код	Комментарий
0000	<i>ADD B</i>	80	$A \leftarrow A+B$
0001	<i>LXI H, 0100h</i>	21 00 01	Загрузка $HL \leftarrow 0100h$
0004	<i>ADD M</i>	86	$A \leftarrow A+(M)$
0005	<i>ADI 1</i>	C6 01	$A \leftarrow A+1$

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей и проверьте полученные результаты.

Вариант	1	2	3	4	5	6
<i>A</i>	<i>00h</i>	<i>00h</i>	<i>00h</i>	<i>F0h</i>	<i>FFh</i>	<i>55h</i>
<i>B</i>	<i>00h</i>	<i>02h</i>	<i>10h</i>	<i>0Eh</i>	<i>00h</i>	<i>AAh</i>
<i>M</i>	<i>00h</i>	<i>03h</i>	<i>45h</i>	<i>00h</i>	<i>00h</i>	<i>FFh</i>

Вариант	7	8	9	10	11	12
<i>A</i>	<i>00h</i>	<i>0Ah</i>	<i>AAh</i>	<i>0Fh</i>	<i>F0h</i>	<i>66h</i>
<i>B</i>	<i>05h</i>	<i>02h</i>	<i>11h</i>	<i>0Eh</i>	<i>01h</i>	<i>0Ah</i>
<i>M</i>	<i>10h</i>	<i>13h</i>	<i>45h</i>	<i>15h</i>	<i>09h</i>	<i>11h</i>

Пример: Запишите в памяти, начиная с адреса 0000h, коды программы сложения 16-разрядных чисел, используя команды 8-разрядного сложения

$$(HL)=(DE)+(BC)$$

Адрес	Команда	Машинный код	Комментарий
0000	<i>MOV A, C</i>	79	
0001	<i>ADD E</i>	83	Сложение младших байтов, установка бита переноса, если переполнение
0002	<i>MOVL, A</i>	6F	Младший байт результата в регистр <i>L</i>
0003	<i>MOV A, B</i>	78	
0004	<i>ADC D</i>	8A	Сложение старших байтов с учетом переноса
0005	<i>MOVH, A</i>	67	Старший байт результата в регистр <i>H</i>

Выполните программ, предварительно задавая исходные значения в соответствии с таблицей. Проверьте результат.

Вариант	1	2	3	4	5	6
<i>BC</i>	<i>0001h</i>	<i>02C5h</i>	<i>F000h</i>	<i>8137h</i>	<i>809Fh</i>	<i>FFFFh</i>
<i>DE</i>	<i>00FEh</i>	<i>03F1h</i>	<i>0FFFh</i>	<i>72D9h</i>	<i>8121h</i>	<i>0000h</i>
Вариант	7	8	9	10	11	12
<i>BC</i>	<i>0005h</i>	<i>01C0h</i>	<i>F100h</i>	<i>1234h</i>	<i>819Fh</i>	<i>00FFh</i>
<i>DE</i>	<i>00F0h</i>	<i>02C1h</i>	<i>0FF0h</i>	<i>70D0h</i>	<i>8122h</i>	<i>0001h</i>

4.2.2. Команды вычитания 8-разрядных чисел.

SUB R вычитание из аккумулятора содержимого одного из регистров *B, C, D, E, H, L*;

SUB M вычитание из аккумулятора содержимого ячейки памяти (адрес в *HL*);

SUI d8 вычитание из аккумулятора непосредственного числа *d8*;

SBB R вычитание из аккумулятора содержимого одного из регистров *B, C, D, E, H, L* минус бит переноса *C*;

SBB M вычитание из аккумулятора содержимого ячейки памяти (адрес в *HL*) минус бит переноса *C*;

SBI d8 вычитание из аккумулятора непосредственного числа *d8* минус бит переноса *C*.

Пример: Запишите в памяти, начиная с адреса 0000h, коды программы, реализующей

$$(A)=(A)-(B)-(M)-1$$

Адрес	Команда	Машинный код	Комментарий
0000	<i>SUB B</i>	90	$(A)=(A)-(B)$
0001	<i>LXI H, 0100h</i>	21 00 01	Загрузка <i>HL=0100h</i> адрес <i>M</i>
0004	<i>SUB M</i>	96	$(A)=(A)-(M)$
0005	<i>SBI 1</i>	DE 01	$(A)=(A)-1$

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей. Проверьте полученные результаты.

Вариант	1	2	3	4	5	6
<i>A</i>	<i>FFh</i>	<i>00h</i>	<i>01h</i>	<i>25h</i>	<i>00h</i>	<i>05h</i>
<i>B</i>	<i>01h</i>	<i>FFh</i>	<i>01h</i>	<i>20h</i>	<i>00h</i>	<i>06h</i>
<i>M</i>	<i>01h</i>	<i>00h</i>	<i>00h</i>	<i>04h</i>	<i>10h</i>	<i>FFh</i>
Вариант	7	8	9	10	11	12
<i>A</i>	<i>F0h</i>	<i>03h</i>	<i>11h</i>	<i>15h</i>	<i>03h</i>	<i>15h</i>
<i>B</i>	<i>01h</i>	<i>F0h</i>	<i>13h</i>	<i>10h</i>	<i>01h</i>	<i>06h</i>
<i>M</i>	<i>02h</i>	<i>11h</i>	<i>02h</i>	<i>04h</i>	<i>02h</i>	<i>F0h</i>

Пример: Запишите в памяти, начиная с адреса *0000h*, коды программы вычитания 16-разрядных чисел

$$(HL)=(DE)-(BC)$$

Адрес	Команда	Машинный код	Комментарий
0000	<i>MOV A, E</i>	7B	
0001	<i>SUB C</i>	91	Вычитание младший байтов $(A)=(E)-(C)$
0002	<i>MOVL, A</i>	6F	Если $E < C$, то перенос=1
0003	<i>MOV A, D</i>	7A	
0004	<i>SBB B</i>	98	Вычитание старшего байта с учетом переноса $(D)-(B)-C$
0005	<i>MOV H, A</i>	67	

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей. Проверьте полученный результат.

Вариант	1	2	3	4	5	6
<i>BC</i>	<i>0001h</i>	<i>02C5h</i>	<i>F000h</i>	<i>8137h</i>	<i>809Fh</i>	<i>FFFFh</i>
<i>DE</i>	<i>00FEh</i>	<i>03F1h</i>	<i>0FFFh</i>	<i>72D9h</i>	<i>8121h</i>	<i>0000h</i>
Вариант	7	8	9	10	11	12
<i>BC</i>	<i>0005h</i>	<i>01C0h</i>	<i>F100h</i>	<i>1234h</i>	<i>819Fh</i>	<i>00FFh</i>
<i>DE</i>	<i>00F0h</i>	<i>02C1h</i>	<i>0FF0h</i>	<i>70D0h</i>	<i>8122h</i>	<i>0001h</i>

4.2.3. Команды сложения с удвоенной точностью.

DAD H сложение $(HL)=(HL)+(HL)$

DAD B сложение $(HL)=(HL)+(BC)$

DAD D сложение $(HL)=(HL)+(DE)$

Пример: Запишите в памяти, начиная с адреса *0000h*, коды программы, реализующей $(HL)=(BC)+(DE)$

Адрес	Команда	Машинный код	Комментарий
0000	<i>MOV H, B</i>	60	Пересылка $(H) \leftarrow (B)$
0001	<i>MOVL, C</i>	69	Пересылка $(H) \leftarrow (B)$
0002	<i>DAD D</i>	19	$(HL)=(HL)+(DE)$

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей. Проверьте результаты.

Вариант	1	2	3	4	5	6
<i>BC</i>	<i>0000h</i>	<i>7FFFh</i>	<i>8000h</i>	<i>55AAh</i>	<i>ECB9h</i>	<i>FFFFh</i>
<i>DE</i>	<i>7FFFh</i>	<i>8000h</i>	<i>8000h</i>	<i>AA55h</i>	<i>1347h</i>	<i>8000h</i>
Вариант	7	8	9	10	11	12
<i>BC</i>	<i>0100h</i>	<i>7000h</i>	<i>1000h</i>	<i>11AAh</i>	<i>ECBDh</i>	<i>F00Fh</i>
<i>DE</i>	<i>7FF0h</i>	<i>8010h</i>	<i>AA00h</i>	<i>0055h</i>	<i>0001h</i>	<i>8011h</i>

4.2.4. Команды инкремента.

INR R увеличение на 1 содержимого регистра *A, B, C, D, E, H, L*;

INR M увеличение на 1 содержимого ячейки памяти, адрес в *HL*;

INX R увеличение на 1 содержимого регистровой пары *BC, DE, HL, SP* (указателя стека). В команде указывается идентификатор старшего регистра, например, *INX B*.

Пример: Запишите в памяти, начиная с адреса *0000h*, код команды

Адрес	Команда	Машинный код	Комментарий
0000	<i>INR R</i>	--	$(R)=(R)+1$

Выполните данную команду, для следующих регистров и исходных значений и проверьте полученные результаты.

Вариант	1	2	3	4	5	6
<i>R</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>H</i>
<i>Исх.зн.</i>	<i>01h</i>	<i>05h</i>	<i>07h</i>	<i>10h</i>	<i>12h</i>	<i>13h</i>
Вариант	7	8	9	10	11	12
<i>R</i>	<i>L</i>	<i>A</i>	<i>B</i>	<i>D</i>	<i>E</i>	<i>H</i>
<i>Исх.зн.</i>	<i>1Ah</i>	<i>1Dh</i>	<i>50h</i>	<i>CAh</i>	<i>0Bh</i>	<i>ABh</i>

Пример: Запишите в памяти, начиная с адреса *0000h*, кода команд

Адрес	Команда	Машинный код	Комментарий
0000	<i>LXI H, 0100h</i>	21 00 01	Загрузить <i>HL=0100h</i> , адрес <i>M</i>
0003	<i>INR M</i>	34	$(M)=(M)+1$

Выполните данную последовательность команд, для следующих исходных значений содержимого ячейки памяти и проверьте полученные результаты.

Вариант	1	2	3	4	5	6
<i>Исх.зн.</i>	<i>11h</i>	<i>05h</i>	<i>07h</i>	<i>1Ch</i>	<i>12h</i>	<i>19h</i>
Вариант	7	8	9	10	11	12
<i>Исх.зн.</i>	<i>1Ah</i>	<i>10h</i>	<i>51h</i>	<i>C0h</i>	<i>0Bh</i>	<i>A0h</i>

Пример: Запишите в памяти, начиная с адреса *0000h*, код команды

Адрес	Команда	Машинный код	Комментарий
0000	<i>INX R</i>	--	$(R)=(R)+1$

Выполните данную команду, для следующих пар регистров и исходных значений и проверьте полученные результаты.

Вариант	1	2	3	4	5	6
R	BC	DE	HL	BC	DE	HL
Исх.зн.	010F	05AA	070B	0FFF	FA12	AF13
Вариант	7	8	9	10	11	12
R	BC	DE	HL	BC	DE	HL
Исх.зн.	1AFD	1D35	5123	CA00	0B0F	AB01

4.2.5. Команда декремента.

DCR R уменьшение на 1 содержимого регистра **A, B, C, D, E, H, L**;

DCR M уменьшение на 1 содержимого ячейки памяти, адрес в **HL**;

DCX R уменьшение на 1 содержимого регистровой пары **BC, DE, HL, SP** (указателя стека). В команде указывается идентификатор старшего регистра, например, **DCX B**.

Пример: Запишите в память начиная с адреса **0000h**, код команды

Адрес	Команда	Машинный код	Комментарий
0000	DCR R	--	(R)=(R)-1

Выполните данную команду, для следующих регистров и исходных значение и проверьте полученные результаты.

Вариант	1	2	3	4	5	6
R	A	B	C	D	E	H
Исх.зн.	11h	05h	0Ah	10h	12h	10h
Вариант	7	8	9	10	11	12
R	L	A	B	D	E	H
Исх.зн.	01h	10h	5Fh	0Ah	1Bh	00h

Пример: Запишите в памяти, начиная с адреса **0000h**, код команды

Адрес	Команда	Машинный код	Комментарий
0000	LXI H, 0100h	21 00 01	Загрузить HL=0100h , адрес M
0003	DCR M	35	(M)=(M)-1

Выполните данную последовательность команд, для следующих исходных значений содержимого ячейки памяти и проверьте полученные результаты.

Вариант	1	2	3	4	5	6
Исх.зн.	F1h	05h	01h	21h	11h	19h
Вариант	7	8	9	10	11	12
Исх.зн.	F0h	FFh	25h	C0h	00h	0Fh

Пример: Запишите в памяти, начиная с адреса *0000h*, код команды

Адрес	Команда	Машинный код	Комментарий
0000	<i>DCX R</i>	2B	<i>(R)=(R)-1</i>

Выполните данную команду, для следующих пар регистров и исходных значений и проверьте полученные результаты.

Вариант	1	2	3	4	5	6
<i>R</i>	<i>BC</i>	<i>DE</i>	<i>HL</i>	<i>BC</i>	<i>DE</i>	<i>HL</i>
<i>Исх.зн.</i>	<i>010Fh</i>	<i>05AAh</i>	<i>0700h</i>	<i>0F00h</i>	<i>FA12h</i>	<i>AF1Fh</i>
Вариант	7	8	9	10	11	12
<i>R</i>	<i>BC</i>	<i>DE</i>	<i>HL</i>	<i>BC</i>	<i>DE</i>	<i>HL</i>
<i>Исх.зн.</i>	<i>1AF0h</i>	<i>1D35h</i>	<i>5100h</i>	<i>CA01h</i>	<i>0B0Ah</i>	<i>AB0Ah</i>

4.3. КОНТРОЛЬНЫЕ ЗАДАНИЯ

4.3.1. Напишите и выполните команду реализующую $(C)=(D)+(E)$

Вариант	1	2	3	4	5	6
<i>D</i>	<i>10h</i>	<i>FFh</i>	<i>C7h</i>	<i>19h</i>	<i>AAh</i>	<i>E5h</i>
<i>E</i>	<i>80h</i>	<i>01h</i>	<i>08h</i>	<i>49h</i>	<i>55h</i>	<i>F0h</i>
Вариант	7	8	9	10	11	12
<i>D</i>	<i>70h</i>	<i>0Fh</i>	<i>C5h</i>	<i>12h</i>	<i>0Ah</i>	<i>EEh</i>
<i>E</i>	<i>81h</i>	<i>11h</i>	<i>88h</i>	<i>46h</i>	<i>50h</i>	<i>F0h</i>

4.3.2. Напишите и выполните программу сложения содержимого двух ячеек памяти (содержимое ячеек необходимо заполнить вручную)

$$(M1)=(M2)+(M3)$$

Адрес $M1=0100h$

Адрес $M2=0101h$

Адрес $M3=0102h$

Вариант	1	2	3	4	5	6
<i>M2</i>	<i>00h</i>	<i>FEh</i>	<i>D5h</i>	<i>22h</i>	<i>61h</i>	<i>19h</i>
<i>M3</i>	<i>F0h</i>	<i>02h</i>	<i>C2h</i>	<i>BBh</i>	<i>95h</i>	<i>33h</i>
Вариант	7	8	9	10	11	12
<i>M2</i>	<i>11h</i>	<i>F0h</i>	<i>DDh</i>	<i>23h</i>	<i>66h</i>	<i>18h</i>
<i>M3</i>	<i>F1h</i>	<i>12h</i>	<i>02h</i>	<i>B1h</i>	<i>09h</i>	<i>3Ah</i>

4.3.3. Напишите и выполните программу сложения $(HL)=(BC)+(E)+4000h$ и заполните таблицу

Вариант	1	2	3	4	5	6
<i>E</i>	<i>00h</i>	<i>FEh</i>	<i>D5h</i>	<i>22h</i>	<i>61h</i>	<i>19h</i>
<i>BC</i>	<i>0AFFh</i>	<i>00FEh</i>	<i>0123h</i>	<i>0A15h</i>	<i>2122h</i>	<i>2223h</i>
Вариант	7	8	9	10	11	12
<i>E</i>	<i>11h</i>	<i>F0h</i>	<i>DDh</i>	<i>23h</i>	<i>66h</i>	<i>18h</i>
<i>BC</i>	<i>0AF0h</i>	<i>A100h</i>	<i>9123h</i>	<i>0615h</i>	<i>2100h</i>	<i>0E23h</i>

4.3.4. Напишите и выполните программу реализующую
 $(C)=(D)-(E)-10h$

Вариант	1	2	3	4	5	6
<i>D</i>	<i>01h</i>	<i>DAh</i>	<i>F0h</i>	<i>85h</i>	<i>9Fh</i>	<i>FFh</i>
<i>E</i>	<i>FEh</i>	<i>10h</i>	<i>0Fh</i>	<i>72h</i>	<i>81h</i>	<i>0Ah</i>
Вариант	7	8	9	10	11	12
<i>D</i>	<i>16h</i>	<i>0Ch</i>	<i>39h</i>	<i>75h</i>	<i>F0h</i>	<i>EEh</i>
<i>E</i>	<i>98h</i>	<i>10h</i>	<i>01h</i>	<i>02h</i>	<i>16h</i>	<i>DDh</i>

4.3.5. Напишите и выполните программу вычитания содержимого двух ячеек памяти (содержимое ячеек необходимо заполнить вручную)
 $(M1)=(M2)-(M3)$; Адрес $M1=0100h$; Адрес $M2=0101h$; Адрес $M3=0102h$

Вариант	1	2	3	4	5	6
<i>M2</i>	<i>AAh</i>	<i>0Eh</i>	<i>D5h</i>	<i>22h</i>	<i>61h</i>	<i>19h</i>
<i>M3</i>	<i>F0h</i>	<i>02h</i>	<i>C2h</i>	<i>B0h</i>	<i>91h</i>	<i>3Ah</i>
Вариант	7	8	9	10	11	12
<i>M2</i>	<i>11h</i>	<i>F0h</i>	<i>D0h</i>	<i>21h</i>	<i>66h</i>	<i>18h</i>
<i>M3</i>	<i>FFh</i>	<i>12h</i>	<i>02h</i>	<i>B1h</i>	<i>19h</i>	<i>36h</i>

4.3.6. Напишите и выполните программу вычитания $(HL)=(BC)-(E)-0FFFh$.

Вариант	1	2	3	4	5	6
<i>E</i>	<i>00h</i>	<i>F0h</i>	<i>15h</i>	<i>23h</i>	<i>62h</i>	<i>18h</i>
<i>BC</i>	<i>0AF0h</i>	<i>01F1h</i>	<i>5123h</i>	<i>1A14h</i>	<i>2112h</i>	<i>2222h</i>
Вариант	7	8	9	10	11	12
<i>E</i>	<i>11h</i>	<i>F1h</i>	<i>15h</i>	<i>26h</i>	<i>67h</i>	<i>00h</i>
<i>BC</i>	<i>0A12h</i>	<i>0101h</i>	<i>5123h</i>	<i>1A15h</i>	<i>2113h</i>	<i>2332h</i>

4.3.7. Напишите и выполните программу заполнения массива по заданному индексу элемента массива в соответствии с таблицей.

Вариант	1					
Базовый элемент массива	0100h					
Номер элемента	<i>00h</i>	<i>03h</i>	<i>06h</i>	<i>11h</i>	<i>18h</i>	<i>1Fh</i>
Содержимое элемента массива	<i>00h</i>	<i>01h</i>	<i>02h</i>	<i>03h</i>	<i>04h</i>	<i>05h</i>
Вариант	2					
Базовый элемент массива	0200h					
Номер элемента	<i>00h</i>	<i>03h</i>	<i>07h</i>	<i>10h</i>	<i>12h</i>	<i>15h</i>
Содержимое элемента массива	<i>01h</i>	<i>11h</i>	<i>02h</i>	<i>AAh</i>	<i>B4h</i>	<i>15h</i>
Вариант	3					
Базовый элемент массива	0300h					
Номер элемента	<i>01h</i>	<i>05h</i>	<i>06h</i>	<i>12h</i>	<i>1Ah</i>	<i>1Bh</i>
Содержимое элемента массива	<i>00h</i>	<i>22h</i>	<i>33h</i>	<i>44h</i>	<i>04h</i>	<i>55h</i>

Вариант	4					
Базовый элемент массива	0150h					
Номер элемента	<i>02h</i>	<i>07h</i>	<i>0Ah</i>	<i>0Bh</i>	<i>12h</i>	<i>13h</i>
Содержимое элемента массива	<i>05h</i>	<i>07h</i>	<i>12h</i>	<i>13h</i>	<i>42h</i>	<i>55h</i>
Вариант	5					
Базовый элемент массива	0050h					
Номер элемента	<i>00h</i>	<i>01h</i>	<i>03h</i>	<i>05h</i>	<i>07h</i>	<i>09h</i>
Содержимое элемента массива	<i>05h</i>	<i>17h</i>	<i>F1h</i>	<i>BAh</i>	<i>ABh</i>	<i>33h</i>
Вариант	6					
Базовый элемент массива	0100h					
Номер элемента	<i>00h</i>	<i>02h</i>	<i>04h</i>	<i>06h</i>	<i>08h</i>	<i>0Ah</i>
Содержимое элемента массива	<i>A0h</i>	<i>A1h</i>	<i>A2h</i>	<i>A2h</i>	<i>A3h</i>	<i>AFh</i>

4.3.8. Напишите и выполните программу перезаписи содержимого массива 1, заданного в задании **4.3.7**, в массив 2 в соответствии таблицей.

Вариант	1					
Базовый элемент массива 2	0200h					
Номер элемента 2	<i>02h</i>	<i>04h</i>	<i>06h</i>	<i>08h</i>	<i>0Ah</i>	<i>0Ch</i>
Вариант	2					
Базовый элемент массива 2	0100h					
Номер элемента 2	<i>02h</i>	<i>04h</i>	<i>07h</i>	<i>08h</i>	<i>0Ah</i>	<i>0Bh</i>
Вариант	3					
Базовый элемент массива 2	0200h					
Номер элемента 2	<i>01h</i>	<i>02h</i>	<i>03h</i>	<i>04h</i>	<i>05h</i>	<i>06h</i>
Вариант	4					
Базовый элемент массива 2	0300h					
Номер элемента 2	<i>00h</i>	<i>01h</i>	<i>03h</i>	<i>07h</i>	<i>09h</i>	<i>0Bh</i>
Вариант	5					
Базовый элемент массива 2	0150h					
Номер элемента 2	<i>01h</i>	<i>02h</i>	<i>06h</i>	<i>07h</i>	<i>09h</i>	<i>0Ah</i>
Вариант	6					
Базовый элемент массива 2	0200h					
Номер элемента 2	<i>01h</i>	<i>02h</i>	<i>03h</i>	<i>04h</i>	<i>05h</i>	<i>06h</i>

4.3.9. Напишите и выполните программу заполнения массива памяти данными, используя команды инкремент пары регистров и регистра.

Вариант	1	2	3
<i>Массив</i>	<i>0050h – 0054h</i>	<i>0060h – 0064h</i>	<i>0070h – 0074h</i>
<i>Данные</i>	<i>00h – 04h</i>	<i>10h – 14h</i>	<i>11h – 15h</i>
Вариант	4	5	6
<i>Массив</i>	<i>0080h – 0084h</i>	<i>0090h – 0094h</i>	<i>0100h – 0104h</i>
<i>Данные</i>	<i>00h – 04h</i>	<i>0Ah – 0Eh</i>	<i>20h – 24h</i>
Вариант	7	8	9
<i>Массив</i>	<i>0200h – 0204h</i>	<i>0210h – 0214h</i>	<i>00220h – 0224h</i>
<i>Данные</i>	<i>50h – 54h</i>	<i>1Ah – 1Eh</i>	<i>00h – 04h</i>
Вариант	10	11	12
<i>Массив</i>	<i>0230h – 0234h</i>	<i>0240h – 0244h</i>	<i>0250h – 0254h</i>
<i>Данные</i>	<i>0Bh – 0Fh</i>	<i>1Bh – 1Fh</i>	<i>05h – 09h</i>

4.3.10. Напишите и выполните программу заполнения массива памяти данными, используя команды декремент пары регистров и регистра.

Вариант	1	2	3
<i>Массив</i>	<i>0054h – 0050h</i>	<i>0064h – 0060h</i>	<i>0074h – 0070h</i>
<i>Данные</i>	<i>05h – 01h</i>	<i>16h – 12h</i>	<i>19h – 15h</i>
Вариант	4	5	6
<i>Массив</i>	<i>0084h – 0080h</i>	<i>0094h – 0090h</i>	<i>0104h – 0100h</i>
<i>Данные</i>	<i>03h – FFh</i>	<i>0Eh – 0Ah</i>	<i>25h – 21h</i>
Вариант	7	8	9
<i>Массив</i>	<i>0204h – 0200h</i>	<i>0214h – 0210h</i>	<i>00224h – 0220h</i>
<i>Данные</i>	<i>54h – 50h</i>	<i>1Eh – 1Ah</i>	<i>09h – 05h</i>
Вариант	10	11	12
<i>Массив</i>	<i>0234h – 0230h</i>	<i>0244h – 0240h</i>	<i>0254h – 0250h</i>
<i>Данные</i>	<i>0Fh – 0Bh</i>	<i>1Fh – 1Bh</i>	<i>0Ah – 06h</i>

Лабораторная работа №5

ЛОГИЧЕСКИЕ КОМАНДЫ

5.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Для реализации логических операций в систем команд микропроцессора K580BM80A предусмотрены следующие логические команды: логическое сложение; логическое умножение; исключающее ИЛИ; инверсия.

Все логические команды выполняются побитно с 8-разрядными операндами. При этом один из операндов размещается в регистре – накопителе, аккумуляторе, а второй – либо в одном из регистров общего назначения, либо в ячейке памяти или задается во втором байте команды. Результат выполнения команды записывается в аккумулятор. При этом бит переноса - сбрасывается в 0, а остальные биты устанавливаются в соответствии с результатом выполнения команды.

Команды логического сложения реализуют логическую операцию «ИЛИ». Результат равен 1, если хотя бы один из соответствующих битов равен 1, и равен 0, если оба равны 0. Например

$$10101001 \vee 00110010 = 10111011,$$

где \vee - обозначение операции «ИЛИ».

Команды логического умножения реализуют логическую операцию «И». Результат равен 1, если оба бита равны 1, и равен 0, если хотя бы один из них равен 0. Например

$$10101001 \wedge 00110010 = 00100000$$

где \wedge - обозначение операции «И».

Команды «ИСКЛЮЧАЮЩЕГО ИЛИ» реализуют логическую операцию сложения по модулю два. Результат равен 1, если соответствующие биты противоположны <1 и 0>, и равен 0, если они одинаковы. Например

$$10101001 \oplus 00110010 = 10011011$$

где \oplus – обозначение операции «ИСКЛЮЧАЮЩЕЕ ИЛИ».

Команда инверсии реализует операцию «ОТРИЦАНИЕ» содержимого аккумулятора. Например

$$\overline{10101001} = 01010110.$$

5.2. ПРОГРАММА РАБОТЫ

5.2.1. Команды логического сложения

ORA R с регистром *A, B, C, D, E, H, L*;

ORA M с ячейкой памяти, адрес ячейки памяти *HL*;

ORI d8 с непосредственным операндом.

Пример: Запишите в память, начиная с адреса *0000h* коды программы, реализующей выражение

$$A \leftarrow A \vee C \vee M \vee 80h$$

Адрес	Команда	Машинный код	Комментарий
0000	<i>ORA C</i>	B1	$A \leftarrow A \vee C$
0001	<i>LXI H, 0100h</i>	21 00 01	$HL \leftarrow 0100h$
0004	<i>ORA M</i>	B6	$A \leftarrow A \vee M$
0005	<i>ORI 80h</i>	F9 80	$A \leftarrow A \vee 80h$

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей

Вариант	1	2	3	4	5	6
<i>A</i>	<i>01h</i>	<i>0Bh</i>	<i>0Ch</i>	<i>0Dh</i>	<i>0Eh</i>	<i>0Fh</i>
<i>C</i>	<i>12h</i>	<i>01h</i>	<i>0Ah</i>	<i>10h</i>	<i>10h</i>	<i>11h</i>
<i>M = 0100h</i>	<i>25h</i>	<i>F0h</i>	<i>10h</i>	<i>00h</i>	<i>02h</i>	<i>00h</i>
Вариант	7	8	9	10	11	12
<i>A</i>	<i>02h</i>	<i>10h</i>	<i>10h</i>	<i>22h</i>	<i>AAh</i>	<i>99h</i>
<i>C</i>	<i>11h</i>	<i>05h</i>	<i>01h</i>	<i>10h</i>	<i>00h</i>	<i>10h</i>
<i>M = 0100h</i>	<i>22h</i>	<i>10h</i>	<i>10h</i>	<i>55h</i>	<i>02h</i>	<i>01h</i>

5.2.2. Запишите в память, начиная с адреса *0000h*, коды программы, реализующей выражения

$$HL \leftarrow BC \vee DE$$

Адрес	Команда	Машинный код	Комментарий
0000	<i>MOVA, C</i>	79	Пересылка $A \leftarrow C$
0001	<i>ORA E</i>	B3	$A \leftarrow A \vee E$
0002	<i>MOVL, A</i>	6F	Пересылка $L \leftarrow A$, младшего байта результата.
0003	<i>MOVA, B</i>	78	Пересылка $A \leftarrow B$
0004	<i>ORA D</i>	B2	$A \leftarrow A \vee D$
0005	<i>MOVH, A</i>	67	Пересылка $H \leftarrow A$, старшего байта результата.

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей

Вариант	1	2	3	4	5	6
<i>BC</i>	<i>0111h</i>	<i>0B22h</i>	<i>0C00h</i>	<i>0F0Dh</i>	<i>0E00h</i>	<i>0F10h</i>
<i>DE</i>	<i>1200h</i>	<i>0100h</i>	<i>0A00h</i>	<i>1011h</i>	<i>1001h</i>	<i>1100h</i>
Вариант	7	8	9	10	11	12
<i>BC</i>	<i>02AAh</i>	<i>1001h</i>	<i>1011h</i>	<i>2200h</i>	<i>AA01h</i>	<i>9925h</i>
<i>DE</i>	<i>1100h</i>	<i>05EAh</i>	<i>0122h</i>	<i>1033h</i>	<i>0010h</i>	<i>1011h</i>

5.2.2. Команды логического умножения.

ANA R с регистром *A, B, C, D, E, H, L*;

ANA M с ячейкой памяти, адрес ячейки памяти *HL*;

ANI d8 с непосредственным операндом.

Пример: Запишите в память, начиная с адреса *0000h* коды программы, реализующей выражение

$$A \leftarrow A \wedge C \wedge M \wedge 7Fh$$

Адрес	Команда	Машинный код	Комментарий
0000	<i>ANA C</i>	A2	$A \leftarrow A \wedge D$
0001	<i>LXI H, 0100h</i>	21 00 01	$HL \leftarrow 0100h$
0004	<i>ANA M</i>	A6	$A \leftarrow A \wedge M$
0005	<i>ANI 7Fh</i>	E6 7F	$A \leftarrow A \wedge 7Fh$

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей

Вариант	1	2	3	4	5	6
<i>A</i>	<i>01h</i>	<i>0Bh</i>	<i>0Ch</i>	<i>0Dh</i>	<i>0Eh</i>	<i>0Fh</i>
<i>C</i>	<i>12h</i>	<i>01h</i>	<i>0Ah</i>	<i>10h</i>	<i>10h</i>	<i>11h</i>
<i>M = 0100h</i>	<i>25h</i>	<i>F0h</i>	<i>10h</i>	<i>00h</i>	<i>02h</i>	<i>00h</i>
Вариант	7	8	9	10	11	12
<i>A</i>	<i>02h</i>	<i>10h</i>	<i>10h</i>	<i>22h</i>	<i>AAh</i>	<i>99h</i>
<i>C</i>	<i>11h</i>	<i>05h</i>	<i>01h</i>	<i>10h</i>	<i>00h</i>	<i>10h</i>
<i>M = 0100h</i>	<i>22h</i>	<i>10h</i>	<i>10h</i>	<i>55h</i>	<i>02h</i>	<i>01h</i>

Пример: Запишите в память, начиная с адреса *0000h*, коды программы, реализующей выражения

$$HL \leftarrow BC \wedge DE$$

Адрес	Команда	Машинный код	Комментарий
0000	<i>MOVA, C</i>	79	Пересылка $A \leftarrow C$
0001	<i>ANA E</i>	A3	$A \leftarrow A \wedge E$
0002	<i>MOVL, A</i>	6F	Пересылка $L \leftarrow A$ младшего байта результата
0003	<i>MOVA, D</i>	7A	Пересылка $A \leftarrow D$
0004	<i>ANA B</i>	A0	$A \leftarrow A \wedge B$
0005	<i>MOVH, A</i>	67	Пересылка $H \leftarrow A$, старшего байта результата.

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей

Вариант	1	2	3	4	5	6
<i>BC</i>	<i>0111h</i>	<i>0B22h</i>	<i>0C00h</i>	<i>0F0Dh</i>	<i>0E00h</i>	<i>0F10h</i>
<i>DE</i>	<i>1200h</i>	<i>0100h</i>	<i>0A00h</i>	<i>1011h</i>	<i>1001h</i>	<i>1100h</i>
Вариант	7	8	9	10	11	12
<i>BC</i>	<i>02AAh</i>	<i>1001h</i>	<i>1011h</i>	<i>2200h</i>	<i>AA01h</i>	<i>9925h</i>
<i>DE</i>	<i>1100h</i>	<i>05EAh</i>	<i>0122h</i>	<i>1033h</i>	<i>0010h</i>	<i>1011h</i>

5.2.3. Команды «ИСКЛЮЧАЮЩЕЕ ИЛИ».

XRA R с регистром *A, B, C, D, E, H, L*;

XRA M с ячейкой памяти, адрес ячейки памяти *HL*;

XRI d8 с непосредственным операндом.

Пример: Запишите в память, начиная с адреса *0000h* коды программы, реализующей выражение

$$A \leftarrow A \oplus C \oplus M \oplus AAh$$

Адрес	Команда	Машинный код	Комментарий
0000	XRA A	AF	$A \leftarrow A \oplus A, A=0$
0001	XRA E	AB	$A \leftarrow A \oplus E$
0002	LXI H, 900h	21 00 09	HL ← 0100h
0005	XRA M	AE	$A \leftarrow A \oplus M$
0006	XRI 7Fh	EE AA	$A \leftarrow A \oplus AAh$

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей

Вариант	1	2	3	4	5	6
A	01h	0Bh	0Ch	0Dh	0Eh	0Fh
C	12h	01h	0Ah	10h	10h	11h
M = 0100h	25h	F0h	10h	00h	02h	00h
Вариант	7	8	9	10	11	12
A	02h	10h	10h	22h	AAh	99h
C	11h	05h	01h	10h	00h	10h
M = 0100h	22h	10h	10h	55h	02h	01h

Пример: Запишите в память, начиная с адреса **0000h**, коды программы, реализующей выражения

$$HL \leftarrow (A \vee B) \wedge DE \oplus (HL \oplus C)$$

Адрес	Команда	Машинный код	Комментарий
0000	ORA B	B0	$A \leftarrow A \vee B$
0001	ANA E	A3	$A \leftarrow A \wedge E$
0003	MOV A, C	79	Пересылка $A \leftarrow C$
0004	XRA L	A	$A \leftarrow A \oplus L$
0005	XRA E	AB	$A \leftarrow A \oplus E$
0006	MOVL, A	6F	Пересылка $L \leftarrow A$, младшего байта результата
0007	MOV A, H	7C	Пересылка $A \leftarrow H$, старшего байта результата
0008	XRA D	AA	$A \leftarrow A \oplus D$
0009	MOVH, A	67	Пересылка $H \leftarrow A$, старшего байта результата.

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей

Вариант	1	2	3	4	5	6
A	05h	73h	21h	DEh	15h	D5h
B	66h	2Fh	EBh	32h	31h	05h
DE	6712h	36CDh	BAFFh	F235h	F235h	F235h
HL ucx	3355h	7921h	68ACh	DBF1h	DB1Eh	D2FEh
C	A2h	36h	78h	C5h	1Dh	0Dh

5.2.4. Команда ИНВЕРСИЯ.

СМА инверсия аккумулятора.

Пример: Запишите в память, начиная с адреса $0000h$ коды программы, реализующей выражение

$$A \leftarrow NOT B AND NOT C$$

Адрес	Команда	Машинный код	Комментарий
000	<i>MOV A, B</i>	78	Пересылка $A \leftarrow B$
001	<i>CMA</i>	2F	$A \leftarrow NOT A$
002	<i>MOV B, A</i>	47	Пересылка $B \leftarrow A$
003	<i>MOV A, C</i>	79	Пересылка $A \leftarrow C$
004	<i>CMA</i>	2F	$A \leftarrow NOT A$
005	<i>ANA B</i>	A0	$A \leftarrow A \wedge B$
006	<i>CMA</i>	2F	$A \leftarrow NOT A$

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей

Вариант	1	2	3	4	5	6
<i>B</i>	<i>00h</i>	<i>FFh</i>	<i>25h</i>	<i>39h</i>	<i>ABh</i>	<i>EDh</i>
<i>C</i>	<i>FFh</i>	<i>02h</i>	<i>56h</i>	<i>21h</i>	<i>13h</i>	<i>C3h</i>
<i>A</i>	<i>FFh</i>	<i>FFh</i>	<i>77h</i>	<i>39h</i>	<i>BBh</i>	<i>EFh</i>

Пример: Запишите в память, начиная с адреса $0000h$, коды программы, реализующей выражения

$$M3 \leftarrow NOT (NOT M1 \vee NOT M2)$$

Адреса ячеек памяти $M1-900h$, $M2-901h$, $M3-900h$.

Адрес	Команда	Машинный код	Комментарий
000	<i>LXI H, 900h</i>	21 00 09	$HL \leftarrow 900h$ адрес M
003	<i>MOV A, M</i>	7E	$A \leftarrow M1$
004	<i>CMA</i>	2F	$A \leftarrow NOT M1$
005	<i>MOV C, A</i>	4F	Пересылка $C \leftarrow A$, промежуточного результата
006	<i>INX H</i>	23	$HL \leftarrow HL+1$, адрес $M2$
007	<i>MOV A, M</i>	7E	$A \leftarrow M2$
008	<i>CMA</i>	2F	$A \leftarrow NOT M2$
009	<i>ORA C</i>	B1	$A \leftarrow A \vee C$
00A	<i>CMA</i>	2F	$A \leftarrow NOT A$
00B	<i>INX H</i>	23	$HL \leftarrow HL+1$, адрес $M3$
00C	<i>MOV M, A</i>	7E	$M \leftarrow A$

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей.

Вариант	1	2	3	4	5	6
<i>M1</i>	<i>37h</i>	<i>43h</i>	<i>09h</i>	<i>78h</i>	<i>78h</i>	<i>78h</i>
<i>M2</i>	<i>29h</i>	<i>5Eh</i>	<i>F6h</i>	<i>95h</i>	<i>95h</i>	<i>95h</i>
<i>M3</i>	<i>21h</i>	<i>42h</i>	<i>94h</i>	<i>10h</i>	<i>10h</i>	<i>10h</i>

5.3. КОНТРОЛЬНЫЕ ЗАДАНИЯ

5.3.1. Напишите и выполните программы реализации выражения.

$$HL \leftarrow B \vee C \vee DE \vee 8800h$$

Вариант 1	Регистры	<i>C</i>	<i>B</i>	<i>D</i>	<i>E</i>
	Данные	<i>00h</i>	<i>10h</i>	<i>11h</i>	<i>F0h</i>
Вариант 2	Регистры	<i>C</i>	<i>B</i>	<i>D</i>	<i>E</i>
	Данные	<i>01h</i>	<i>11h</i>	<i>A2h</i>	<i>00h</i>
Вариант 3	Регистры	<i>C</i>	<i>B</i>	<i>D</i>	<i>E</i>
	Данные	<i>00h</i>	<i>11h</i>	<i>0Bh</i>	<i>0Ah</i>
Вариант 4	Регистры	<i>C</i>	<i>B</i>	<i>D</i>	<i>E</i>
	Данные	<i>05h</i>	<i>06h</i>	<i>11h</i>	<i>A0h</i>
Вариант 5	Регистры	<i>C</i>	<i>B</i>	<i>D</i>	<i>E</i>
	Данные	<i>A0h</i>	<i>0Ah</i>	<i>00h</i>	<i>11h</i>
Вариант 6	Регистры	<i>C</i>	<i>B</i>	<i>D</i>	<i>E</i>
	Данные	<i>05h</i>	<i>19h</i>	<i>00h</i>	<i>12h</i>

5.3.2. Напишите и выполните программу реализации выражения

$$(M3) = (M1) \vee (M2)$$

Адреса ячеек памяти соответственно равны

$$M1 = 0100h$$

$$M2 = 0101h$$

$$M3 = 0102h$$

Вариант 1	Ячейки	<i>M1</i>	<i>M2</i>
	Данные	<i>0Fh</i>	<i>E0h</i>
Вариант 2	Ячейки	<i>M1</i>	<i>M2</i>
	Данные	<i>11h</i>	<i>02h</i>
Вариант 3	Ячейки	<i>M1</i>	<i>M2</i>
	Данные	<i>AAh</i>	<i>55h</i>
Вариант 4	Ячейки	<i>M1</i>	<i>M2</i>
	Данные	<i>01h</i>	<i>10h</i>
Вариант 5	Ячейки	<i>M1</i>	<i>M2</i>
	Данные	<i>02h</i>	<i>F0h</i>
Вариант 6	Ячейки	<i>M1</i>	<i>M2</i>
	Данные	<i>0Bh</i>	<i>B1h</i>

5.3.3. Напишите и выполните программы реализации выражения

$$HL \leftarrow (HL \wedge BC \vee DE) \wedge 03FFh$$

Вариант 1	Регистры	<i>BC</i>	<i>HL</i>	<i>DE</i>
	Данные	<i>00FFh</i>	<i>0010h</i>	<i>1000h</i>
Вариант 2	Регистры	<i>BC</i>	<i>HL</i>	<i>DE</i>
	Данные	<i>0101h</i>	<i>2125h</i>	<i>2211h</i>
Вариант 3	Регистры	<i>BC</i>	<i>HL</i>	<i>DE</i>
	Данные	<i>2233h</i>	<i>4400h</i>	<i>1000h</i>
Вариант 4	Регистры	<i>BC</i>	<i>HL</i>	<i>DE</i>
	Данные	<i>0010h</i>	<i>5000h</i>	<i>300Fh</i>
Вариант 5	Регистры	<i>BC</i>	<i>HL</i>	<i>DE</i>
	Данные	<i>0F0Fh</i>	<i>0FA0h</i>	<i>0001h</i>
Вариант 6	Регистры	<i>BC</i>	<i>HL</i>	<i>DE</i>
	Данные	<i>0F0Bh</i>	<i>1000h</i>	<i>22FFh</i>

5.3.4. Напишите и выполните программу реализации выражения

$$(M2) \leftarrow A \wedge (M1) \vee C \wedge D$$

Адреса ячеек памяти $M1=0100h$, $M2=0101h$.

Вариант 1	Регистры	<i>A</i>	<i>M1</i>	<i>C</i>	<i>D</i>
	Данные	<i>00h</i>	<i>10h</i>	<i>FFh</i>	<i>0Ah</i>
Вариант 2	Регистры	<i>A</i>	<i>M1</i>	<i>C</i>	<i>D</i>
	Данные	<i>FFh</i>	<i>05h</i>	<i>15h</i>	<i>20h</i>
Вариант 3	Регистры	<i>A</i>	<i>M1</i>	<i>C</i>	<i>D</i>
	Данные	<i>0Eh</i>	<i>12h</i>	<i>22h</i>	<i>55h</i>
Вариант 4	Регистры	<i>A</i>	<i>M1</i>	<i>C</i>	<i>D</i>
	Данные	<i>A1h</i>	<i>1Ah</i>	<i>11h</i>	<i>BBh</i>
Вариант 5	Регистры	<i>A</i>	<i>M1</i>	<i>C</i>	<i>D</i>
	Данные	<i>EEh</i>	<i>11h</i>	<i>BBh</i>	<i>22h</i>
Вариант 6	Регистры	<i>A</i>	<i>M1</i>	<i>C</i>	<i>D</i>
	Данные	<i>0Fh</i>	<i>77h</i>	<i>EEh</i>	<i>B1h</i>

5.3.5. Напишите и выполните программу реализации выражения

$$HL \leftarrow L \oplus D \vee E \wedge H \oplus B \vee C \oplus 0Fh$$

Вариант 1	Регистры	<i>HL</i>	<i>DE</i>	<i>BC</i>
	Данные	<i>0000h</i>	<i>5610h</i>	<i>FF48h</i>
Вариант 2	Регистры	<i>HL</i>	<i>DE</i>	<i>BC</i>
	Данные	<i>FF00h</i>	<i>0235h</i>	<i>1B35h</i>
Вариант 3	Регистры	<i>HL</i>	<i>DE</i>	<i>BC</i>
	Данные	<i>1001h</i>	<i>F00Fh</i>	<i>0CBAh</i>
Вариант 4	Регистры	<i>HL</i>	<i>DE</i>	<i>BC</i>
	Данные	<i>E3E0h</i>	<i>15B1h</i>	<i>B32Bh</i>

Вариант 5	Регистры	<i>HL</i>	<i>DE</i>	<i>BC</i>
	Данные	<i>2377h</i>	<i>EDAEh</i>	<i>B341h</i>
Вариант 6	Регистры	<i>HL</i>	<i>DE</i>	<i>BC</i>
	Данные	<i>0BC5h</i>	<i>1125h</i>	<i>2020h</i>

5.3.6. Напишите и выполните программу реализации выражения.

$$M \leftarrow M \oplus AAh \wedge H \oplus B \vee E$$

Вариант 1	Регистры	<i>M</i>	<i>H</i>	<i>B</i>	<i>E</i>
	Данные	<i>EEh</i>	<i>B1h</i>	<i>EEh</i>	<i>B1h</i>
Вариант 2	Регистры	<i>M</i>	<i>H</i>	<i>B</i>	<i>E</i>
	Данные	<i>FFh</i>	<i>05h</i>	<i>FFh</i>	<i>0Ah</i>
Вариант 3	Регистры	<i>M</i>	<i>H</i>	<i>B</i>	<i>E</i>
	Данные	<i>10h</i>	<i>FFh</i>	<i>77h</i>	<i>EEh</i>
Вариант 4	Регистры	<i>M</i>	<i>H</i>	<i>B</i>	<i>E</i>
	Данные	<i>EEh</i>	<i>EEh</i>	<i>11h</i>	<i>B1h</i>
Вариант 5	Регистры	<i>M</i>	<i>H</i>	<i>B</i>	<i>E</i>
	Данные	<i>05h</i>	<i>B1h</i>	<i>FFh</i>	<i>0Ah</i>
Вариант 6	Регистры	<i>M</i>	<i>H</i>	<i>B</i>	<i>E</i>
	Данные	<i>FFh</i>	<i>05h</i>	<i>15h</i>	<i>20h</i>

5.3.7. Напишите и выполните реализации выражения

$$HL \leftarrow NOT DE \oplus NOT BC \wedge NOT C \vee NOT B$$

Вариант 1	Регистры	<i>D</i>	<i>B</i>	<i>C</i>	<i>E</i>
	Данные	<i>FFh</i>	<i>0Ch</i>	<i>33h</i>	<i>10h</i>
Вариант 2	Регистры	<i>D</i>	<i>B</i>	<i>C</i>	<i>E</i>
	Данные	<i>EEh</i>	<i>23h</i>	<i>32h</i>	<i>F3h</i>
Вариант 3	Регистры	<i>D</i>	<i>B</i>	<i>C</i>	<i>E</i>
	Данные	<i>ACh</i>	<i>26h</i>	<i>B2h</i>	<i>41h</i>
Вариант 4	Регистры	<i>D</i>	<i>B</i>	<i>C</i>	<i>E</i>
	Данные	<i>12h</i>	<i>73h</i>	<i>00h</i>	<i>16h</i>
Вариант 5	Регистры	<i>D</i>	<i>B</i>	<i>C</i>	<i>E</i>
	Данные	<i>1Ah</i>	<i>A4h</i>	<i>53h</i>	<i>Aeh</i>
Вариант 6	Регистры	<i>D</i>	<i>B</i>	<i>C</i>	<i>E</i>
	Данные	<i>F4h</i>	<i>C1h</i>	<i>22h</i>	<i>EEh</i>

5.3.8. Напишите и выполните реализации выражения

$$M6 \leftarrow NOT M1 \oplus M2 \vee NOT M3 \wedge NOT M4 \vee M5$$

Адреса ячеек памяти

$$\begin{array}{ll}
 M1=900h & M4=903h \\
 M2=901h & M5=904h \\
 M3=902h & M6=905h
 \end{array}$$

Вариант 1	Регистр	<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>
	Данные	<i>FFh</i>	<i>AAh</i>	<i>A4h</i>	<i>00h</i>	<i>74h</i>
Вариант 2	Регистр	<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>
	Данные	<i>00h</i>	<i>BEh</i>	<i>4Bh</i>	<i>56h</i>	<i>00h</i>
Вариант 3	Регистр	<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>
	Данные	<i>05h</i>	<i>26h</i>	<i>33h</i>	<i>ACh</i>	<i>FFh</i>
Вариант 4	Регистр	<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>
	Данные	<i>34h</i>	<i>BBh</i>	<i>51h</i>	<i>C4h</i>	<i>A7h</i>
Вариант 5	Регистр	<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>
	Данные	<i>56h</i>	<i>FAh</i>	<i>43h</i>	<i>DCh</i>	<i>00h</i>
Вариант 6	Регистр	<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>
	Данные	<i>C3h</i>	<i>FFh</i>	<i>15h</i>	<i>ADh</i>	<i>CCh</i>

Лабораторная работа №6

КОМАНДЫ СРАВНЕНИЯ

6.1 ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ.

Система команд микропроцессора *Intel 8080* содержит три типа команд сравнения:

1. Сравнение содержимого аккумулятора с содержимым регистра;
2. Сравнение содержимого аккумулятора с содержимым ячейки памяти;
3. Сравнение содержимого аккумулятора с непосредственным операндом.

Команды сравнения выполняются посредством внутреннего вычитания из содержимого аккумулятора, соответственно, содержимого регистра, ячейки памяти и непосредственного операнда. Содержимое аккумулятора при этом не изменяется. В результате сравнения устанавливаются биты признаков следующим образом

Результат сравнения	Признак	
	(Ноль)	(Перенос C)
Равно	1	0
Больше	0	0
Меньше	0	1

6.2. ПРОГРАММА РАБОТЫ

6.2.1. Команды сравнения содержащего регистра с содержимым аккумулятора.

СМР А содержимым регистра *A*

СМР В содержимым регистра *B*

СМР С содержимым регистра *C*

СМР D содержимым регистра *D*

СМР E содержимым регистра *E*

СМР H содержимым регистра *H*

СМР L содержимым регистра *L*

Пример: Запишите в память, начиная с адреса *0000h*, коды программ сравнения содержимого регистров *C* и *B*.

Адрес	Команда	Машинный код	Комментарий
0000	<i>MOV A, C</i>	79	Пересылка $(A) \leftarrow (C)$
0001	<i>СМР В</i>	B8	Сравнение с (B) , $(A)-(B)$

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей. Проверьте полученные результаты.

<i>C</i>	<i>01h</i>	<i>01h</i>	<i>FFh</i>	<i>43h</i>	<i>55h</i>	<i>03h</i>	<i>20h</i>
<i>B</i>	<i>01h</i>	<i>09h</i>	<i>00h</i>	<i>FFh</i>	<i>55h</i>	<i>55h</i>	<i>15h</i>
Ноль <i>Z</i>	<i>1</i>	<i>0</i>	<i>0</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>0</i>
Перенос <i>C</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>0</i>	<i>0</i>
Знак <i>S</i>	<i>0</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>0</i>	<i>0</i>	<i>0</i>
Четность <i>P</i>	<i>1</i>	<i>0</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>0</i>

6.2.2. Команды сравнения с памятью

CMP M сравнение содержимого регистра *A* с содержимым ячейки памяти, адрес которой задан в паре регистров **HL**.

Пример: Запишите в память, начиная с адреса **0000h**, коды программы сравнения с содержимым ячейки памяти.

Адрес	Команда	Машинный код	Комментарий
0000	LXI H, 0100h	21 00 01	Загрузка HL=0100h
0003	CMP M	BE	Сравнение содержимого A и M

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей. Проверьте полученные результаты.

<i>A</i>	<i>21h</i>			<i>BAh</i>			<i>E9h</i>		
<i>M=0100h</i>	<i>00h</i>	<i>21h</i>	<i>39h</i>	<i>19h</i>	<i>FFh</i>	<i>BAh</i>	<i>E9h</i>	<i>10h</i>	<i>F5h</i>
Ноль <i>Z</i>	0	1	0	0	0	1	1	0	0
Перенос <i>C</i>	0	0	1	0	1	0	0	0	1
Знак <i>S</i>	0	0	1	0	1	0	0	0	1
Четность <i>P</i>	1	1	1	0	1	1	1	0	0

6.2.3. Команды сравнения с непосредственным операндом.

CPI d8 сравнение содержимого регистра *A* с числом, заданным во втором байте команды, **d8** – байт.

Пример: Запишите в память, начиная с адреса **0000h**, коды команды сравнения с непосредственным операндом.

Адрес	Команда	Машинный код	Комментарий
0000	CPI 7Fh	FE 7F	Сравнение <i>A</i> и 7Fh

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей.

Вариант	1	2	3	4	5	6
<i>A</i>	<i>00h</i>	<i>80h</i>	<i>7Fh</i>	<i>B3h</i>	<i>25h</i>	<i>F7h</i>
Ноль <i>Z</i>	0	0	1	0	0	0
Перенос <i>C</i>	1	0	0	0	1	0
Знак <i>S</i>	1	0	0	0	1	0
Четность <i>P</i>	1	0	1	0	1	1

6.3. КОНТРОЛЬНЫЕ ЗАДАНИЕ

6.3.1. Напишите и выполните программу сравнения содержимого регистров **H** и **L**. Заполните таблицу.

Вариант 1	H	L	Ноль Z	Перенос C	Знак S	Четность P
	FFh	0Fh				
Вариант 2	H	L	Ноль Z	Перенос C	Знак S	Четность P

	<i>00h</i>	<i>0Ah</i>				
Вариант 3	<i>H</i>	<i>L</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>01h</i>	<i>00h</i>				
Вариант 4	<i>H</i>	<i>L</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>ABh</i>	<i>BAh</i>				
Вариант 5	<i>H</i>	<i>L</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>10h</i>	<i>10h</i>				
Вариант 6	<i>H</i>	<i>L</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>12h</i>	<i>11h</i>				

6.3.2. Напишите и выполните программу сравнения содержимого регистра *B* с содержимым ячейки памяти по адресу *0110h*. Заполните таблицу.

Вариант 1	<i>A</i>	<i>M</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>00h</i>	<i>FFh</i>				
Вариант 2	<i>A</i>	<i>M</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>05h</i>	<i>05h</i>				
Вариант 3	<i>A</i>	<i>M</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>FFh</i>	<i>01h</i>				
Вариант 4	<i>A</i>	<i>M</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>12h</i>	<i>15h</i>				
Вариант 5	<i>A</i>	<i>M</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>0Ah</i>	<i>0Bh</i>				
Вариант 6	<i>A</i>	<i>M</i>	Ноль <i>Z</i>	Перенос <i>C</i>	Знак <i>S</i>	Четность <i>P</i>
	<i>22h</i>	<i>20h</i>				

6.3.3. Напишите и выполните программу сравнения содержимого регистра *H* с непосредственным операндом. Заполните таблицу.

Вариант 1	Операнд	<i>5Ah</i>				
	<i>A</i>	<i>01h</i>	<i>10h</i>	<i>5Ah</i>	<i>FFh</i>	<i>5Bh</i>
	Ноль <i>Z</i>					
	Перенос <i>C</i>					
	Знак <i>S</i>					
	Четность <i>P</i>					
Вариант 2	Операнд	<i>05h</i>				
	<i>A</i>	<i>10h</i>	<i>1Ah</i>	<i>05h</i>	<i>01h</i>	<i>51h</i>
	Ноль <i>Z</i>					
	Перенос <i>C</i>					
	Знак <i>S</i>					
	Четность <i>P</i>					
Вариант 3	Операнд	<i>10h</i>				
	<i>A</i>	<i>11h</i>	<i>10h</i>	<i>00h</i>	<i>01h</i>	<i>50h</i>
	Ноль <i>Z</i>					
	Перенос <i>C</i>					
	Знак <i>S</i>					

	Четность <i>P</i>					
Вариант 4	Операнд	<i>FEh</i>				
	<i>A</i>	<i>01h</i>	<i>FEh</i>	<i>FFh</i>	<i>F0h</i>	<i>5Ah</i>
	Ноль <i>Z</i>					
	Перенос <i>C</i>					
	Знак <i>S</i>					
	Четность <i>P</i>					
Вариант 5	Операнд	<i>0Ah</i>				
	<i>A</i>	<i>00h</i>	<i>11h</i>	<i>0Ah</i>	<i>1Fh</i>	<i>0Bh</i>
	Ноль <i>Z</i>					
	Перенос <i>C</i>					
	Знак <i>S</i>					
	Четность <i>P</i>					
Вариант 6	Операнд	<i>22h</i>				
	<i>A</i>	<i>22h</i>	<i>11h</i>	<i>33h</i>	<i>FFh</i>	<i>00h</i>
	Ноль <i>Z</i>					
	Перенос <i>C</i>					
	Знак <i>S</i>					
	Четность <i>P</i>					

Лабораторная работа №7

КОМАНДЫ СДВИГА

7.1. ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ

В системе команд микропроцессора *Intel 8080* предусмотрены следующие команды сдвига: циклический сдвиг влево *RLC*; циклический сдвиг вправо *RRC*; сдвиг влево через перенос *RAL*; сдвиг вправо через перенос *RAR*.

Команды сдвига выполняются в регистре – накопителе, аккумуляторе над 8 – разрядными операндами. Результат заносится в аккумулятор.

Команда циклического сдвига влево *RLC* перемещает каждый бит байта на один разряд влево. При этом содержимое старшего разряда записывается в младший разряд и в бит переноса.

Исходное содержимое флага переноса аккумулятора

Флаг переноса	Данные							
<i>C</i>	<i>b7</i>	<i>b6</i>	<i>b5</i>	<i>b4</i>	<i>b3</i>	<i>b2</i>	<i>b1</i>	<i>b0</i>

После *RLC*

Флаг переноса	Данные							
<i>C</i>	<i>b6</i>	<i>b5</i>	<i>b4</i>	<i>b3</i>	<i>b2</i>	<i>b1</i>	<i>b0</i>	<i>b7</i>

Команда циклического сдвига вправо *RRC* перемещает каждый бит байта на 1 разряд вправо. При этом содержимое младшего разряда записывается в старший разряд и в бит переноса.

Исходное содержимое флага переноса и аккумулятора

Флаг переноса	Данные							
<i>C</i>	<i>b7</i>	<i>b6</i>	<i>b5</i>	<i>b4</i>	<i>b3</i>	<i>b2</i>	<i>b1</i>	<i>b0</i>

После *RRC*

Флаг переноса	Данные							
<i>b0</i>	<i>b0</i>	<i>b7</i>	<i>b6</i>	<i>b5</i>	<i>b4</i>	<i>b3</i>	<i>b2</i>	<i>b1</i>

Команда сдвига влево через перенос *RAL* перемещает содержимое каждого бита байта влево на 1 разряд. При этом содержимое бита переноса записывается в младший разряд, а содержимое старшего разряда заносится в бит переноса. Используя эту команду, можно реализовать операцию умножения на число кратное 2.

Исходное содержимое флага переноса и аккумулятора

Флаг переноса	Данные							
<i>C</i>	<i>b7</i>	<i>b6</i>	<i>b5</i>	<i>b4</i>	<i>b3</i>	<i>b2</i>	<i>b1</i>	<i>b0</i>

После *RAL*

Флаг переноса	Данные							
<i>C</i>	<i>b6</i>	<i>b5</i>	<i>b4</i>	<i>b3</i>	<i>b2</i>	<i>b1</i>	<i>b0</i>	<i>C</i>

Команда сдвига вправо через перенос перемещает содержимое каждого разряда байта на один разряд. При этом в старший разряд байта записывается значение бита переноса, а в него заносится содержимое младшего разряда байта. Используя эту команду, можно реализовать операцию деления на число кратное 2.

Исходное содержимое флага переноса и аккумулятора

Флаг переноса	Данные							
<i>C</i>	<i>b7</i>	<i>b6</i>	<i>b5</i>	<i>b4</i>	<i>b3</i>	<i>b2</i>	<i>b1</i>	<i>b0</i>

После *RAR*

Флаг переноса	Данные							
<i>b0</i>	<i>C</i>	<i>b7</i>	<i>b6</i>	<i>b5</i>	<i>b4</i>	<i>b3</i>	<i>b2</i>	<i>b1</i>

7.2. ПРОГРАММА РАБОТЫ

7.2.1. Команды циклического сдвига.

RLC циклический сдвиг вправо;

RRC циклический сдвиг влево.

Пример: Запишите в память, начиная с адреса 0000h, коды программы реализующей операцию циклического сдвига байта на 4 разряда, используя команды *RLC*.

Адрес	Команда	Машинный код	Комментарий
0000	<i>RLC</i>	07	;циклический сдвиг влево на 1 разряд
0001	<i>RLC</i>	07	;циклический сдвиг влево на 1 разряд
0002	<i>RLC</i>	07	;циклический сдвиг влево на 1 разряд
0003	<i>RLC</i>	07	;циклический сдвиг влево на 1 разряд

Выполните эту программу, предварительно задавая исходные значения в соответствии с таблицей.

<i>A</i> исх.	<i>00</i>	<i>0F</i>	<i>F0</i>	<i>81</i>	<i>A5</i>	<i>67</i>
<i>A</i> рез.	<i>00</i>	<i>F0</i>	<i>0F</i>	<i>18</i>	<i>5A</i>	<i>76</i>

Пример: Запишите в память, начиная с адреса 0000h, коды программы, реализующей операцию объединения старших тетрад байтов, в регистрах *B* и *C*, в один

Адрес	Команда	Машинный код	Комментарий
0000	<i>MOVA, C</i>	79	
0001	<i>RRC</i>	0F	перемещение старшей тетрады 1-го байта
0002	<i>RRC</i>	0F	
0003	<i>RRC</i>	0F	на место младшей тетрады.
0004	<i>RRC</i>	0F	
0005	<i>ANI 0Fh</i>	E6 0F	выделение старшей тетрады 1-го байта.
0006	<i>MOV C, A</i>	4F	
0007	<i>MOVA, B</i>	78	
0008	<i>ANI F0h</i>	E6 F0	выделение старшей тетрады 2-го байта.
000B	<i>ORA C</i>	B1	объединение двух байт в один.

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей. Сравните результаты.

<i>C</i>	<i>72h</i>	<i>F0h</i>	<i>51h</i>	<i>19h</i>
<i>B</i>	<i>9Fh</i>	<i>0Fh</i>	<i>A3h</i>	<i>86h</i>
<i>A рез.</i>	<i>97h</i>	<i>0Fh</i>	<i>A5h</i>	<i>81h</i>

7.2.2. Команды сдвига через перенос.

RAL сдвиг влево через перенос;

RAR сдвиг вправо через перенос.

Пример: Запишите в память, начиная с адреса *0000h*, коды программы, реализующей циклический сдвиг влево на 1 разряд содержимого пары регистров **HL**.

Адрес	Команда	Машинный код	Комментарий
0000	ORA A	B7	сброс переноса в 0
0001	MOV A, L	7D	сдвиг влево L на 1 разряд через перенос
0002	RAL	17	0 – в младший разряд L
0003	MOV L, A	6F	
0004	MOV A, H	7C	сдвиг влево H на 1 разряд через перенос
0005	RAL	17	с учетом переноса из L
0006	MOV H, A	67	старший разряд P в перенос
0007	MOV A, L	1D	перенос – в младший разряд L
0008	ACI 0	CE 00	
000A	MOV L, A	6F	

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей.

HL исх.	<i>FFFC h</i>	<i>8002 h</i>	<i>3578 h</i>	<i>FFFF h</i>	<i>0000 h</i>	<i>1111 h</i>
HL рез.	<i>FFF9 h</i>	<i>0005 h</i>	<i>6AF0 h</i>	<i>FFFF h</i>	<i>0000 h</i>	<i>2222 h</i>

Пример: Запишите в память, начиная с адреса *0000h*, коды программы, реализующей операцию умножения на 4, содержимого регистра **C**.

Адрес	Команда	Машинный код	Комментарий
0000	MOV A, C	79	
0001	ORA A	B7	сброс бита переноса
0002	RAL	17	умножение на 2
0003	RAL	17	умножение на 2
000 4	MOV B, A	47	результат в B

ПРИМЕЧАНИЕ исходное значение не должно превышать 63.

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей.

<i>C</i>	<i>00h</i>	<i>02h</i>	<i>10h</i>	<i>2Fh</i>	<i>33h</i>	<i>3Ah</i>
<i>B</i>	<i>00h</i>	<i>08h</i>	<i>40h</i>	<i>BC h</i>	<i>CC h</i>	<i>E8 h</i>

Пример: Запишите в память, начиная с адреса *0000h*, коды программы, деления на 8 содержимого регистра *H*. Целая часть результата помещена в регистр *D*, остаток в *E*.

Адрес	Команда	Машинный код	Комментарий
0000	<i>MOV A, H</i>	7C	
0001	<i>ORA A</i>	B7	сброс переноса в 0
0002	<i>RAR</i>	IF	$A=A/2$
0003	<i>ORA A</i>	B7	сброс переноса в 0
0004	<i>RAR</i>	IF	$A=A/2$
0005	<i>ORA A</i>	B7	сброс переноса в 0
0006	<i>RAR</i>	IF	$A=A/2$
0007	<i>MOVD, A</i>	57	$D=H/8$ – целая часть
0008	<i>MOV A, H</i>	7C	
0009	<i>ANI 07h</i>	E6 07	выделение остатка результата
000B	<i>MOVE, A</i>	5F	остаток в <i>E</i>

Выполните программу, предварительно задавая исходные значения в соответствии с таблицей. Проверьте результаты.

<i>H</i>	<i>05h</i>	<i>08h</i>	<i>10h</i>	<i>35h</i>	<i>79h</i>	<i>FFh</i>
<i>D</i>	<i>00h</i>	<i>01h</i>	<i>02h</i>	<i>06h</i>	<i>0Fh</i>	<i>1Fh</i>
<i>E</i>	<i>05h</i>	<i>00h</i>	<i>00h</i>	<i>05h</i>	<i>01h</i>	<i>07h</i>

7.3. КОНТРОЛЬНЫЕ ЗАДАНИЯ

7.3.1. Напишите и выполните программу объединения младших тетрад двух байтов в один. Младшую тетраду 2-го байта поместить в старшую тетраду результирующего байта.

Вариант 1	<i>E (1 байт)</i>	<i>D (2 байт)</i>
	<i>F0h</i>	<i>11h</i>
Вариант 2	<i>E (1 байт)</i>	<i>D (2 байт)</i>
	<i>0Fh</i>	<i>AAh</i>
Вариант 3	<i>E (1 байт)</i>	<i>D (2 байт)</i>
	<i>22h</i>	<i>11h</i>
Вариант 4	<i>E (1 байт)</i>	<i>D (2 байт)</i>
	<i>55h</i>	<i>ABh</i>
Вариант 5	<i>E (1 байт)</i>	<i>D (2 байт)</i>
	<i>35h</i>	<i>53h</i>
Вариант 6	<i>E (1 байт)</i>	<i>D (2 байт)</i>
	<i>77h</i>	<i>01h</i>

7.3.2. Напишите и выполните программу, реализующую операцию логического умножения 3-х битов 5, 6, 7 одного байта на 3 бита 2, 3,4 второго байта, предварительно переместив умножаемые биты этих байт на младшие позиции, и обнулив остальные разряды.

Вариант 1	<i>L (1 байт)</i>	<i>H (2 байт)</i>
	<i>ABh</i>	<i>17h</i>

Вариант 2	<i>L (1 байт)</i>	<i>H (2 байт)</i>
	<i>F0h</i>	<i>1Ah</i>
Вариант 3	<i>L (1 байт)</i>	<i>H (2 байт)</i>
	<i>22h</i>	<i>11h</i>
Вариант 4	<i>L (1 байт)</i>	<i>H (2 байт)</i>
	<i>AAh</i>	<i>BBh</i>
Вариант 5	<i>L (1 байт)</i>	<i>H (2 байт)</i>
	<i>77h</i>	<i>01h</i>
Вариант 6	<i>L (1 байт)</i>	<i>H (2 байт)</i>
	<i>35h</i>	<i>53h</i>

7.3.3. Напишите и выполните программу циклического сдвига на 3 разряда содержимого пары регистров.

Вариант 1	<i>DE</i>
	<i>FF00h</i>
Вариант 2	<i>HL</i>
	<i>22F0h</i>
Вариант 3	<i>BC</i>
	<i>0220h</i>
Вариант 4	<i>DE</i>
	<i>3563h</i>
Вариант 5	<i>HL</i>
	<i>ABCDh</i>
Вариант 6	<i>BC</i>
	<i>D32Ah</i>

7.3.4. Напишите и выполните программу деления содержимого пары регистров *BC* на 8. $BC=BC/8$.

Вариант 1	<i>DE</i>
	<i>1F40h</i>
Вариант 2	<i>HL</i>
	<i>1DC8h</i>
Вариант 3	<i>BC</i>
	<i>42B0h</i>
Вариант 4	<i>DE</i>
	<i>5988h</i>
Вариант 5	<i>HL</i>
	<i>A7C8h</i>
Вариант 6	<i>BC</i>
	<i>4570h</i>

Лабораторная работа №8

КОМАНДЫ БЕЗУСЛОВНОГО И УСЛОВНЫХ ПЕРЕХОДОВ. ВВОД-ВЫВОД ДАННЫХ

8.1. ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ

В системе команд микропроцессора КР580ВМ80А предусмотрены команды изменения последовательности выполнения команд для организации циклов, обработки условий передачи управления и т.д. Существуют два типа команд перехода безусловный и условный.

При выполнении команд безусловного перехода осуществляется передача управления по адресу, заданному во втором и третьем байтах команды, либо по адресу, заданному в регистровой паре.

Команды условного перехода выполняются в том случае, если установлен или сброшен соответствующий бит признака, в противном случае команда игнорируется и выполняется следующая за ней команда.

Существуют команды условного перехода для следующих битов признаков

- бита нуля *Z*
- бита переноса *C*
- бита знака *S*
- бита четности *P*

Для каждого бита признака предусмотрены две команды перехода переход по установленному признаку (=1) и переход по сброшенному биту признака (=0).

Соответствие выполнения команды и признаков приведены в таблице.

Признак	Ноль <i>Z</i>		Перенос <i>C</i>		Четность <i>P</i>		Знак <i>S</i>	
	<i>1</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>1</i>	<i>0</i>
<i>JZ</i>	Да	---	---	---	---	---	---	---
<i>JNZ</i>	---	Да	---	---	---	---	---	---
<i>JC</i>	---	---	Да	---	---	---	---	---
<i>JNC</i>	---	---	---	Да	---	---	---	---
<i>JPE</i>	---	---	---	---	Да	---	---	---
<i>JPO</i>	---	---	---	---	---	Да	---	---
<i>JM</i>	---	---	---	---	---	---	Да	---
<i>JP</i>	---	---	---	---	---	---	---	Да

8.2. ПРОГРАММА РАБОТЫ

8.2.1. Команды безусловного перехода

JMP ADR16 безусловный переход по адресу указанному по 2 и 3 байтах команды.

PCHL безусловный переход по адресу, заданному в *HL*.

Пример: Запишите в память, начиная с адреса *0000h*, коды программы, осуществляющей инкремент аккумулятора, переход по адресу *0100h*, записанному в регистре *HL* и возврат в начало программы.

Адрес	Команда	Машинный код	Комментарий
0000	<i>LXI H, 0100h</i>	21 00 01	Загрузка в <i>HL</i> адреса перехода
0003	<i>MVI A, 00h</i>	3E 00	Загрузка в <i>A=0</i>

0005	<i>INR A</i>		Инкремент <i>A</i>
0006	<i>PCHL</i>	E9	Переход (<i>PC</i>)←(<i>HL</i>)
0100	<i>JMP 0000h</i>		Переход по адресу <i>0000h</i>

8.2.2. Команды перехода по признаку – ноль.

JZ ADR переход, если $Z = 1$;

JNZ ADR переход, если $Z = 0$.

Пример: Запишите в память, начиная с адреса *0000h*, программу заполнения *10h* ячеек памяти нулями.

Адрес	Команда	Машинный код	Комментарий
000	<i>MVI C, 10h</i>	0E 10	Загрузка $C \leftarrow 10h$, длина массива
002	<i>LXI H, 100h</i>	21 00 01	$HL \leftarrow 0100h$, начальный адрес массива
005	<i>MVI M, 0</i>	36	Загрузка (HL) = 0
007	<i>INX H</i>	23	$HL \leftarrow HL + 1$, следующий адрес
008	<i>DCR C</i>	0D	$C = C - 1$, длина массива
009	<i>JNZ 005h</i>	C2 05 00	Продолжать, если длина массива не равна 0

Проверьте результаты выполнения программы в соответствии с таблицей

<i>C</i>	<i>H</i>	$M(0100h) - M(010Fh)$
<i>00</i>	<i>110</i>	<i>00</i>

Пример: Запишите в память, программу заполнения *10h* ячеек памяти нулями.

Адрес	Команда	Машинный код	Комментарий
000	<i>MVI C, 10h</i>	0E 10	Загрузка в $C \leftarrow 10h$, длина массива
002	<i>LXI H, 100h</i>	21 00 09	Загрузка $HL \leftarrow 100h$, начальный адрес массива
005	<i>MOV A, M</i>	7E	Чтение $A \leftarrow (HL)$
006	<i>CPI 0</i>	FE 00	$A \leftarrow 0$
008	<i>JZ 00Dh</i>	CA 0D 00	Переход, если да
00B	<i>MVI M, 0</i>	36 00	Нет, загрузка (HL)←0
00D	<i>INX H</i>	23	$HL \leftarrow HL + 1$, следующий адрес
00E	<i>DCR C</i>	0D	$C = C - 1$, длина массива
00F	<i>JNZ 005h</i>	C2 05 00	Продолжать, если длина массива не равна 0

Проверьте результаты выполнения программ в соответствии с таблицей

<i>C</i>	<i>P</i>	$M(0100h) - M(010Fh)$
<i>00</i>	<i>110</i>	<i>00</i>

8.2.3. Команды перехода по признаку *C* – перенос.

JC ADR переход, если $C=1$;

JNC ADR переход, если $C=0$.

Пример: Запишите в память, начиная с адреса *0000h*, программу, выполняющую сложение аккумулятора с регистром В. Если присутствует перенос, в аккумулятор записывается *FFh*, в противном случае, в аккумулятор кладется *00h*.

1) *A=FFh* *B=10h*;

2) *A=00h* *B=10h*.

Адрес	Команда	Машинный код	Комментарий
000	<i>MVI A,FFh</i>	3E FF	Загрузка A ← FFh
001	<i>MVI B,10h</i>	06 10	Загрузка B ← 10h
003	<i>ORA B</i>	B0	$A \leftarrow A \vee B$
005	<i>JC 100h</i>	DA 00 01	Если перенос = 1, перейти в ячейку с адресом 0100h
006	<i>JNC 200h</i>	D2 00 02	Если перенос = 0, перейти в ячейку с адресом 0200h
100	<i>MVI A,FFh</i>	3E FF	Загрузка A ← FFh
200	<i>MVI A,00h</i>	3E 00	Загрузка A ← 00h

8.2.4. Команды перехода по принципу Р – четность.

JPE ADR переход если P = 1;

JPO ADR переход если P = 0.

Пример: Запишите в память, начиная с адреса *0000h*, программу дополнения байта по четности в старшем разряде. Исходное число в регистре С.

Адрес	Команда	Машинный код	Комментарий
000	<i>MOV A,C</i>	79	Пересылка $A \leftarrow C$ исходного байта
001	<i>ANI 7Fh</i>	E6 7F	Обнуление старшего разряда
003	<i>ORA A</i>	B7	$A \vee A$ – установка бита четности
004	<i>JPO</i>	E2 09 00	Переход, если исходный байт четный
007	<i>ORI 80h</i>	F6 80	Дополнить до четности
009	<i>MOV C,A</i>	4F	Результат

Выполните программу, задавая исходные значения в соответствии с таблицей. Проверьте результат.

<i>C</i> исходное	<i>00</i>	<i>FF</i>	<i>B6</i>	<i>80</i>	<i>CD</i>	<i>75</i>
<i>C</i> результат	<i>00</i>	<i>FF</i>	<i>36</i>	<i>00</i>	<i>4D</i>	<i>F5</i>

8.2.5. Команды перехода по признаку – знак.

JM ADR переход, если *S=1*.

JP ADR переход, если *S=0*.

8.2.6. Команды ввода/вывода.

IN adr8 считывание данных с порта;

OUT adr8 вывод данных в порт.

Пример: Запиши в память, начиная с адреса **0000h**, код программы, которая, при нажатой кнопке, инкрементирует переменную и выводит ее значение в порт со светодиодами. Адрес порта с переключателями **12h**, адрес порта со светодиодами **10h**.

Адрес	Команда	Машинный код	Комментарий
000	<i>MVI A, 00H</i>	3E 00	Загрузка $A \leftarrow 00h$
002	<i>MVI C, FFH</i>	0E FF	Загрузка $C \leftarrow FFh$
004	<i>MVI D, FFH</i>	16 FF	Загрузка $D \leftarrow FFh$
006	<i>MOV B,A</i>	47	Загрузка $B \leftarrow A$
007	<i>IN 12H</i>	DB 12	Считывание порта 12H
008	<i>DCR A</i>	3D	$A \leftarrow A-1$
009	<i>JC 07H</i>	DA 07 00	Переход, если $C=1$
00C	<i>MOV A,B</i>	78	Загрузка $A \leftarrow B$
00D	<i>INR A</i>	3C	$A \leftarrow A+1$
00E	<i>MOV B,A</i>	47	Загрузка $B \leftarrow A$
00F	<i>OUT 10H</i>	D3 10	Вывод в порт 10h
010	<i>CALL 200H</i>	CD 00 02	Вызов подпрограммы по адресу 0200h
013	<i>JMP 07H</i>	C3 07 00	Переход на адрес 07h
200	<i>DCR C</i>	0D	$C \leftarrow C-1$
201	<i>JNC 200H</i>	D2 00 02	Переход, если $C=0$
204	<i>DCR D</i>	15	$D \leftarrow D-1$
205	<i>JNC 204H</i>	D2 04 02	Переход, если $C=0$
208	<i>RET</i>	C9	Возврат из подпрограммы

8.3. КОНТРОЛЬНЫЕ ЗАДАНИЯ

1. Организовать на выводах порта эффект бегущей «1».
2. Организовать на выводах порта эффект бегущего «0».
3. Организовать мерцание светодиодов.
4. Организовать смену частоты мерцания светодиодов нажатием кнопки. При отпущенной кнопке одна частота мерцания, при нажатой кнопке другая частота.
5. Организовать смену направления бегущей «1» нажатием кнопки. При отпущенной кнопке «1» бежит в одну сторону, а при нажатой кнопке, в другую.
6. Организовать смену направления бегущего «0» нажатием кнопки. При отпущенной кнопке «0» бежит в одну сторону, а при нажатой кнопке, в другую.
7. Организовать эффект светофора. Старшие 3 бита порта – красный цвет, средние 2 бита – желтый и младшие 3 бита – зеленый.
8. Организовать эффект эквалайзера. Бегущая «1» туда и обратно.
9. Организовать эффект эквалайзера. Бегущий «0» туда и обратно.
10. Организовать инкремент и декремент по нажатию кнопки. По нажатию одной кнопки производится инкремент переменной и вывод ее в порт, по нажатию другой декремент и вывод в порт.
11. Организовать на выводах порта эффект бегущей «1». Длина пути бегущей «1» определяется нажатием кнопки. Какая кнопка нажата, до туда и бежит «1».
12. Организовать на выводах порта эффект бегущего «0». Длина пути бегущего «0» определяется нажатием кнопки. Какая кнопка нажата, до туда и бежит «0».

Все временные задержки организовать с помощью подпрограммы.

