

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Коротков Сергей Леонидович
Должность: Директор филиала СамГУПС в г.
Ижевске
Дата подписания: 28.04.2021 12:14:41
Уникальный программный ключ:
d3c1f7ec2252b3b19e5caaa8cefa396a11af1dc5

**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА
ФИЛИАЛ ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО
ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ
ВЫСШЕГО ОБРАЗОВАНИЯ
САМАРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ПУТЕЙ СООБЩЕНИЯ**

**в г. Ижевске
(филиал СамГУПС в г. Ижевске)**

**Комплект оценочных средств
по учебной дисциплине
«Микропроцессоры и микропроцессорные системы»
Общепрофессионального цикла
09.02.02 Компьютерные сети
(базовый уровень)**

2020

СОДЕРЖАНИЕ

1. <u>Паспорт комплекта контрольно-измерительных материалов</u>	4
2. <u>Результаты освоения учебной дисциплины, подлежащие проверке</u>	5
3. <u>Оценка освоения учебной дисциплины</u>	7
3.1. <u>Формы и методы оценивания</u>	7
3.2. <u>Типовые задания для оценки освоения учебной дисциплины</u>	13
4. <u>Контрольно-измерительные материалы для итоговой аттестации по учебной дисциплине</u>	64

1. Паспорт комплекта контрольно-измерительных материалов

В результате освоения учебной дисциплины «Микропроцессоры и микропроцессорные системы» обучающийся должен обладать, предусмотренными ФГОС по специальности 09.02.02 Компьютерные сети (*базовый уровень*) следующими умениями, знаниями, которые формируют профессиональную и общую компетенцию:

У1. Создавать и отлаживать программы реального времени.

З1. Особенности программирования микропроцессорных систем реального времени;

З2. Методы микропроцессорной реализации типовых функций управления;

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

Формой аттестации по учебной дисциплине является **дифференцированный зачет**.

2. Результаты освоения учебной дисциплины, подлежащие проверке

В результате аттестации по учебной дисциплине осуществляется комплексная проверка следующих умений и знаний, а также динамика формирования общих компетенций:

Таблица 1

Результаты обучения (освоенные умения, усвоенные знания, освоенные компетенции)	Основные показатели оценки результатов	Формы и методы контроля и оценки результатов обучения
<p>Умения:</p> <ul style="list-style-type: none"> - создавать и отлаживать программы реального времени <p>Знания:</p> <ul style="list-style-type: none"> - особенности программирования микропроцессорных систем реального времени; - методы микропроцессорной реализации типовых функций управления 	<p>Умения:</p> <ul style="list-style-type: none"> .уметь создавать и отлаживать программы <p>Знания:</p> <ul style="list-style-type: none"> -знать особенности программирования микропроцессорных систем - знать методы микропроцессорной реализации типовых функций управления 	<p><i>Входной контроль:</i></p> <ul style="list-style-type: none"> - устный опрос, тестирование, собеседование, <p><i>Текущий контроль:</i></p> <ul style="list-style-type: none"> -опрос, семинар, коллоквиум, -практические занятия; -самостоятельная проверочная работа, -выполнение индивидуальных заданий, рефератов; -самоконтроль, взаимопроверка; -рейтинговый метод оценки самостоятельной работы обучающихся; -тестирование (в том числе компьютерное); -нетрадиционные занятия, <p><i>Тематический (периодический) контроль:</i></p> <ul style="list-style-type: none"> -защита практических работ, -отчёт по практике, индивидуальным домашним заданиям, рефератам; - зачёт, <p><i>Рубежный контроль:</i></p>
ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес	<ul style="list-style-type: none"> – демонстрация интереса к будущей профессии, – увлеченность инновациями в данной профессиональной области, 	
ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество	<ul style="list-style-type: none"> – выбор и применение методов и способов решения профессиональных задач в области организации сетевого администрирования; – оценка эффективности и качества выполнения; 	
ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.	<ul style="list-style-type: none"> – организация эффективного поиска необходимой информации; – умение использовать различных виды источников, в том числе включая электронные; 	
ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.	<ul style="list-style-type: none"> – организация самостоятельных занятий при изучении профессионального модуля, – планирование повышения уровня профессиональной компетентности 	
ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности	<ul style="list-style-type: none"> – анализ инноваций в области организации сетевого администрирования; 	
ПК 1.5. Выполнять требования нормативно-технической документации, иметь опыт оформления проектной	<ul style="list-style-type: none"> – правильность, техническая и юридическая грамотность применения нормативно-технической документации в 	

документации.	<p>области информационных технологий;</p> <ul style="list-style-type: none"> – продуктивность участия в планировании развития программно-технической базы организации; – аргументированность обоснования предложений по реализации стратегии организации в области информационных технологий; – продуктивность участия в научных конференциях, семинарах; <p>точность и грамотность оформления технологической документации, её соответствие действующим правилам и руководствам.</p>	<p>-контрольная работа по разделу, <i>Итоговый контроль:</i> -экзамен.</p>
---------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------

3. Оценка освоения учебной дисциплины:

3.1. Формы и методы оценивания

Предметом оценки служат умения и знания, предусмотренные ФГОС по дисциплине «Микропроцессоры и микропроцессорные системы» направленные на формирование общих и профессиональных компетенций.

Контроль и оценка результатов освоения учебной дисциплины осуществляется преподавателем в процессе проведения практических занятий, а также выполнения обучающимися практических и контрольных работ. Итоговая аттестация в форме дифференцированного зачета: все практические, контрольные работы и тематические самостоятельные работы выполнены на положительные оценки

Элемент учебной дисциплины	Формы и методы контроля					
	Текущий контроль		Рубежный контроль		Аттестация	
	Форма контроля	Проверяемые ОК, У, З	Форма контроля	Проверяемые ОК, У, З	Форма контроля	Проверяемые ОК, У, З
Введение. История развития						
Раздел 1 Микропроцессоры						
Тема 1.1 Архитектура МП	Устный опрос Самостоятельная работа	У1,З1,ОК1				
Тема 1.2 Организация блоков памяти	Устный опрос Самостоятельная работа Практическая работа №1	У1,З1,ОК1,ОК2				
Тема 1.3 Внутренняя организация МП	Устный опрос Самостоятельная работа	У1,З1,ОК1,ОК2				
Тема 1.4 Классификация МП	Устный опрос Самостоятельная работа	У1,З1,ОК1,ОК2, ОК8				
Тема 1.5 Набор команд МП	Устный опрос Самостоятельная работа	У1,З1,ОК1,ОК2, ОК8				
Тема 1.6 Способы адресации операндов	Устный опрос Самостоятельная работа					
Тема 1.7 Производство процессоров. Основные типы МП, их	Устный опрос Самостоятельная работа	У1,З1,ОК1,ОК2, ОК8				

характеристики. Возможности и области применения						
Тема 1.8 Магистрالی, связывающие блоки МП	Устный опрос Самостоятельная работа	У1,31,OK1,OK2, OK8				
Тема 1.9 Принцип работы МП	Устный опрос Самостоятельная работа	У1,31,OK1,OK2, OK8				
Раздел 2 Микропроцессор ные системы						
Тема 2.1 Архитектура микропроцессорн ой системы	Устный опрос Самостоятельная работа	У1,31,32,OK1, OK2, OK4,OK8				
Тема 2.2 Базовая структура ЭВМ как микропроцессорн ой системы	Устный опрос Самостоятельная работа	У1,31,32,OK1, OK2, OK4,OK8				
Тема 2.3 Выбор и оценка качества микропроцессорн ого комплекта	Устный опрос Самостоятельная работа Лабораторная работа №1 Лабораторная работа №2 Лабораторная работа №3 Лабораторная работа №4	У1,31,32,OK1, OK2, OK4,OK8				
Раздел 3 Выстраиваемые микропроцессор ные системы						
Тема 3.1 Интерфейсы встраиваемых	Устный опрос Самостоятельная работа Практическая работа №2	У1,31,32,OK1,OK 2, OK4,OK8				

микропроцессорных систем (МПС)	Практическая работа №3					
Тема 3.2 Уровни представления МПС	Устный опрос Самостоятельная работа	У1,31,32,ОК1,ОК2, ОК4,ОК8				
Тема 3.3 Особенности проектирования МПС	Устный опрос Самостоятельная работа	У1,31,32,ОК1,ОК2, ОК4,ОК8				
Тема 3.4 Проверка правильности проектирования МПС	Устный опрос Самостоятельная работа	У1,31,32,ОК1,ОК2, ОК4,ОК8				
Тема 3.5 Средства разработки МПС	Устный опрос Самостоятельная работа	У1,31,32,ОК1,ОК2, ОК4,ОК8				
Раздел 4. Микроконтроллеры						
Тема 4.1 Семейство микроконтроллеров. Общая характеристика. Номенклатура семейства, состав. Направления развития элементной базы.	Устный опрос Самостоятельная работа Лабораторная работа №5 Лабораторная работа №6 Лабораторная работа №7 Лабораторная работа №8 Лабораторная работа №9 Лабораторная работа №10	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				
Тема 4.2 Модульный принцип построения микроконтроллер	Устный опрос Самостоятельная работа	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				

ов						
Тема 4.3 Программируемые контроллеры прерываний	Устный опрос Самостоятельная работа	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				
Тема 4.4. Контроллеры прямого доступа к памяти (КПДП)	Устный опрос Самостоятельная работа	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				
Раздел 5. Процессорное ядро микроконтроллера		У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				
Тема 5.1 Программно-логическая модель микроконтроллера	Устный опрос Самостоятельная работа	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				
Тема 5.2 Способы адресации. Система команд микроконтроллера	Устный опрос Самостоятельная работа Лабораторная работа №11 Лабораторная работа №12 Лабораторная работа №13 Лабораторная работа №14	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				
Тема 5.3 особенности организации системы прерываний	Устный опрос Самостоятельная работа	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				
Тема 5.4 Организация памяти и доступа к ней	Устный опрос Самостоятельная работа	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				

Тема 5.5 Программирование микроконтроллеров	Устный опрос Самостоятельная работа	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9				
Тема 5.6 Режимы работы микроконтроллера	Устный опрос Самостоятельная работа	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9			Дифф. зачет	У1,31,32,ОК1, ОК2,ОК4,ОК8, ОК9

3.2. Задания для оценки освоения учебной дисциплины

Лабораторная работа № 1

Команды загрузки регистров. Команды пересылки

Теоретические сведения

В микропроцессоре *Intel 8080* 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 показано в табл. 3.3.

Таблица 3.3

Распределение битов в байте состояния

7 6 5 4 3 2 1 0

S Z 0 AC 0 P 1 C

Регистры общего назначения могут использоваться для манипуляции 16-разрядными данными. Для этого регистры объединяются в пары следующим образом: ***BC, DE, HL***, где 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.

Программа работы

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*, последовательность команд, представленных в табл. 3.4.

Таблица 3.4

Программа загрузки внутренних регистров микропроцессора

Адрес Команда Машинный код Комментарий

0000 **MVI A,00** 3E 00 загрузка регистра $A \leftarrow 00h$
 0002 **MVI B,01** 06 01 загрузка регистра $B \leftarrow 01h$
 0004 **MVI C,02** 0E 02 загрузка регистра $C \leftarrow 02h$
 0006 **MVI D,03** 16 03 загрузка регистра $D \leftarrow 03h$
 0008 **MVI E,04** 1E 04 загрузка регистра $E \leftarrow 04h$
 000A **MVI H,05** 26 05 загрузка регистра $H \leftarrow 05h$
 000C **MVI L,06** 2E 06 загрузка регистра $L \leftarrow 06h$

Выполнить эту последовательность команд в пошаговом режиме и пронаблюдать, 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. 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: Записать в память, начиная с адреса **0000h**, последовательность команд, представленных в табл. 3.5.

Таблица 3.5

Программа загрузки регистровых пар

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.

Выполнить выше описанную последовательность команд в пошаговом режиме и пронаблюдать, как изменяется содержимое регистров **B, C, D, E, H, L**. Значения регистров должны быть следующими:

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

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: Записать в память, начиная с адреса **0000h**, последовательность команд, представленных в табл. 3.6.

Таблица 3.6

Программа загрузки указателя стека

Адрес Команда Машинный код Комментарий

0000 **LXI SP, 0B01h** 31 01 0B загрузка указателя стека: $SP \leftarrow 0B01h$

0003 **LXI SP, 0100h** 31 00 01 загрузка указателя стека: $SP \leftarrow 0100h$

Выполнить 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: Записать в память, начиная с адреса **0000h**, последовательность команд, представленных в табл. 3.7.

Таблица 3.7

Программа косвенной загрузки указателя стека

Адрес Команда Машинный код Комментарий

0000 **LXI H, 0210h** 21 10 02 загрузка $HL \leftarrow 0210h$

0003 **SPHL** F9 загрузка $SP \leftarrow HL$

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

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: Записать в память, начиная с адреса **0000h**, последовательность команд, представленных в табл. 3.8.

Таблица 3.8

Программа пересылки данных между регистрами

Адрес Команда Машинный код Комментарий

0000 **MVI A, FFh** 3E FF загрузка регистра $A \leftarrow FFh$

0002 **MOV B, A** 47 пересылка $B \leftarrow A$

0003 **MOV C, B** 48 пересылка $C \leftarrow B$

0004 **MOV D, C** 51 пересылка $D \leftarrow C$

0005 **MOV E, D** 5A пересылка $E \leftarrow D$

0006 **MOV H, E** 63 пересылка $H \leftarrow E$

0007 **MOV L, H** 6С пересылка $L \leftarrow H$

Выполнить эту последовательность команд в пошаговом режиме и пронаблюдать, 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.

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: Записать в память по адресу **0000h**, последовательность команд, представленных в табл. 3.9.

Таблица 3.9

Программа косвенной загрузки программного счетчика

Адрес Команда Машинный код Комментарий

0000 **LXI H, 0100h** 21 00 01 **HL ← 0100h**

0003 **PCHL** E9 **PC ← HL**

Выполнить эту последовательность команд в пошаговом режиме и пронаблюдайте, как изменяется содержимое регистров **H, L** и программного счетчика **PC**.

Контрольные задания

1. Написать и выполнить программу загрузки регистров в соответствии с табл. 3.10.

Таблица 3.10

Варианты заданий загрузки регистров

Вариант **1 2 3 4 5 6**

B: ← **F0h B:** ← **01h B:** ← **AEh B:** ← **21h B:** ← **23h B:** ← **BBh**
C: ← **33h C:** ← **35h C:** ← **FBh C:** ← **16h C:** ← **45h C:** ← **CCh**
D: ← **EEh D:** ← **EAh D:** ← **35h D:** ← **E1h D:** ← **10h D:** ← **D1h**
E: ← **AAh E:** ← **A1h E:** ← **26h E:** ← **D5h E:** ← **62h E:** ← **EEh**
H: ← **00h H:** ← **A1h H:** ← **16h H:** ← **01h H:** ← **A5h H:** ← **AAh**
L: ← **19h L:** ← **18h L:** ← **AAh L:** ← **20h L:** ← **97h L:** ← **FFh**
A: ← **FFh A:** ← **F5h A:** ← **FEh A:** ← **25h A:** ← **F1h A:** ← **43h**

Вариант **7 8 9 10 11 12**

B: ← **23h B:** ← **23h B:** ← **0Eh B:** ← **11h B:** ← **23h B:** ← **8Bh**
C: ← **45h C:** ← **45h C:** ← **0Bh C:** ← **66h C:** ← **33h C:** ← **1Ch**
D: ← **10h D:** ← **1Ah D:** ← **05h D:** ← **11h D:** ← **15h D:** ← **11h**
E: ← **62h E:** ← **A1h E:** ← **36h E:** ← **55h E:** ← **1Dh E:** ← **E3h**
H: ← **A5h H:** ← **1Bh H:** ← **06h H:** ← **00h H:** ← **D1h H:** ← **A5h**
L: ← **97h L:** ← **19h L:** ← **0Ah L:** ← **22h L:** ← **2Dh L:** ← **F5h**
A: ← **95h A:** ← **53h A:** ← **0Eh A:** ← **66h A:** ← **F3h A:** ← **44h**

Проверить правильность выполнения программы.

2. Написать и выполнить программу загрузки регистровых пар в соответствии с табл. 3.11.

Таблица 3.11

Варианты заданий загрузки регистровых пар

Вариант 1 2 3

$BC \leftarrow FFFFh$ $BC \leftarrow 00FFh$ $BC \leftarrow 0000h$

$DE \leftarrow 0123h$ $DE \leftarrow 0124h$ $DE \leftarrow 0F0Fh$

$HL \leftarrow 55AAh$ $HL \leftarrow 5555h$ $HL \leftarrow 1579h$

Вариант 4 5 6

$BC \leftarrow 0E0Eh$ $BC \leftarrow 1100h$ $BC \leftarrow DDDDh$

$DE \leftarrow E0E0h$ $DE \leftarrow 4545h$ $DE \leftarrow ABCDh$

$HL \leftarrow 000Eh$ $HL \leftarrow 536Ah$ $HL \leftarrow DCBAh$

Вариант 7 8 9

$BC \leftarrow 13EFh$ $BC \leftarrow 1234h$ $BC \leftarrow FEDCh$

$DE \leftarrow A734h$ $DE \leftarrow 5678h$ $DE \leftarrow BA98h$

$HL \leftarrow 1FA9h$ $HL \leftarrow 9ABCh$ $HL \leftarrow 7654h$

Вариант 10 11 12

$BC \leftarrow 3210h$ $BC \leftarrow 2468h$ $BC \leftarrow 0000h$

$DE \leftarrow 35DFh$ $DE \leftarrow ACE2h$ $DE \leftarrow 1111h$

$HL \leftarrow 5555h$ $HL \leftarrow 468Ah$ $HL \leftarrow 2222h$

Проверить правильность работы программы.

3. Написать и выполнить программы загрузки регистра указателя стека, в соответствии с табл. 3.12, с использованием команд непосредственной и косвенной загрузки.

Таблица 3.12

Варианты заданий загрузки регистра указателя стека

Вариант 1 2 3

$SP \leftarrow 0100h$ $SP \leftarrow 0101h$ $SP \leftarrow 0802h$

$SP \leftarrow 0200h$ $SP \leftarrow 0202h$ $SP \leftarrow 0812h$

$SP \leftarrow 0300h$ $SP \leftarrow 0203h$ $SP \leftarrow 0822h$

Вариант 4 5 6

$SP \leftarrow 0200h$ $SP \leftarrow 0800h$ $SP \leftarrow 0702h$

Окончание табл. 3.12

$SP \leftarrow 0201h$ $SP \leftarrow 0900h$ $SP \leftarrow 0802h$

$SP \leftarrow 0202h$ $SP \leftarrow 0000h$ $SP \leftarrow 0902h$

Вариант 7 8 9

$SP \leftarrow 0200h$ $SP \leftarrow 000Ah$ $SP \leftarrow 0110h$

$SP \leftarrow 0222h$ $SP \leftarrow 000Bh$ $SP \leftarrow 0120h$

$SP \leftarrow 0333h$ $SP \leftarrow 0A0Bh$ $SP \leftarrow 0130h$

Вариант 10 11 12

$SP \leftarrow 0800h$ $SP \leftarrow 0110h$ $SP \leftarrow 0115h$

$SP \leftarrow 0825h$ $SP \leftarrow 0111h$ $SP \leftarrow 0178h$

$SP \leftarrow 0850h$ $SP \leftarrow 0800h$ $SP \leftarrow 0564h$

Проверить правильность работы программы.

4. Написать и выполнить программу пересылки, в соответствии с табл. 3.13, предварительно загрузив регистры.

Таблица 3.13

Варианты заданий пересылки данных

Вариант 1 2 3

$B \leftarrow A$ (значением 00h) $D \leftarrow H$ (значением 01h) $B \leftarrow C$ (значением 10h)
 $C \leftarrow L$ (значением 0Eh) $E \leftarrow C$ (значением 02h) $C \leftarrow A$ (значением 20h)
 $H \leftarrow B$ (значением 0Fh) $B \leftarrow L$ (значением 03h) $A \leftarrow B$ (значением 30h)

Вариант 4 5 6

$L \leftarrow H$ (значением 0Fh) $H \leftarrow A$ (значением FFh) $C \leftarrow B$ (значением 15h)
 $C \leftarrow A$ (значением 0Eh) $E \leftarrow A$ (значением EEh) $C \leftarrow H$ (значением 25h)
 $D \leftarrow E$ (значением 0Bh) $B \leftarrow C$ (значением D1h) $A \leftarrow L$ (значением 35h)

Вариант 7 8 9

$A \leftarrow H$ (значением 22h) $B \leftarrow C$ (значением 00h) $B \leftarrow C$ (значением 15h)
 $C \leftarrow B$ (значением 11h) $D \leftarrow E$ (значением 0Eh) $C \leftarrow A$ (значением 20h)
 $D \leftarrow L$ (значением 10h) $H \leftarrow L$ (значением 0Fh) $A \leftarrow B$ (значением 2Ah)

Вариант 10 11 12

$H \leftarrow A$ (значением 00h) $B \leftarrow A$ (значением AAh) $A \leftarrow H$ (значением 21h)
 $E \leftarrow A$ (значением 01h) $C \leftarrow L$ (значением 0Ah) $C \leftarrow B$ (значением 22h)
 $B \leftarrow C$ (значением 05h) $H \leftarrow B$ (значением 00h) $D \leftarrow L$ (значением 23h)

Проверить правильность выполнения программы.

5. Написать и выполнить программу перехода с адреса **0000h** на адреса в соответствии с табл. 3.14.

Таблица 3.14

Варианты заданий перехода на адрес

Вариант 1 2 3

0100h 0050h 0080h
0200h 0060h 0090h
0300h 0070h 0100h

64

Окончание табл. 3.12

Вариант 4 5 6

0100h 0800h 0900h
0200h 0200h 0200h
0800h 0100h 0100h

Вариант 7 8 9

0100h 0200h 0100h
0200h 0250h 0200h
0800h 0700h 0150h

Вариант 10 11 12

0110h 0100h 0050h
0210h 0150h 0100h
0250h 0700h 0150h

Проверить правильность выполнения программы.

Лабораторная работа № 2

Методы адресации памяти. Команды работы с памятью

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**

Программа работы

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 (табл. 3.15)

66

Таблица 3.15

Пример кода программы

Адрес	Команда	Машинный код	Комментарий
-------	---------	--------------	-------------

0000	MVI A, FFh	3E FF	Запись в аккумулятор значения FFh
------	-------------------	-------	------------------------------------------

0002	STA 0110h	32 10 01	Запись в память содержимого регистра A по адресу 0110h
------	------------------	----------	----------------------------------------------------------------------

0005	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 .
------	---------------------	----------	------------------------------------------------------------------------------------------------------------------------------------------------------

0008 **SHLD 0150h** 22 50 01 Запись в память содержимого регистра **L** по адресу **0150h**, содержимого регистра **H** по адресу **0151h**

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

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 (табл. 3.16).

Таблица 3.16

Пример кода программы

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

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. 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 (табл. 3.17).

Таблица 3.17

Пример кода 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 **M**
 000D **MOV M, A** 77 запись в **M=A**, 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 **LXI H, 0103h** 21 03 01
 0019 **MOV M, E** 73
 001A **LXI H, 0104h** 21 04 01
 001D **MOV M, D** 72
 001E **LXI H, 0105h** 21 05 01
 0021 **MOV M, H** 74
 0022 **LXI H, 0106h** 21 06 01
 0025 **MOV M, L** 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 (табл. 3.18).

Таблица 3.18

Пример кода программы

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

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 (табл. 3.19).

Таблица 3.19

Пример кода 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 (табл. 3.20).

Таблица 3.20

Пример кода 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 **MOV L, 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 **MOV H, 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.

Контрольные задания

1. Напишите и выполните программу записи данных в память из регистра *A*, в соответствии с табл. 3.21. Для этого используйте команду загрузки регистра *A* и команду записи в память регистра *A* по прямому адресу.

Таблица 3.21

Варианты заданий записи данных в память

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.

2. Напишите и выполните программу записи данных из регистровой пары **HL** в соответствии с табл. 3.22. Для этого используйте команду загрузки регистровой пары **HL** и команду записи в память регистровой пары **HL** по прямому адресу.

Таблица 3.22

Варианты заданий записи данных из регистровой пары

Вариант 1

Адрес 0100 0101 0111 0112 0120 0121

Данные 00h 07h 09h 0Bh B0h 12h

Вариант 2

Адрес 0200 0201 0205 0206 0210 0211

Данные 25h 12h 50h 0Ah 0Ch D1h

Вариант 3

Адрес 0101 0102 0120 0121 0135 0136

Данные 15h 1Ah A5h 6Ah 00h 21h

Вариант 4

Адрес 0100 0101 0107 0108 0123 0124

Данные 22h 33h 44h 55h AAh CCh

Вариант 5

Адрес 0210 0211 0220 0221 0228 0229

Данные FFh 1Ah A1h 03h B0h 12h

Вариант 6

Адрес 0150 0151 0161 0162 0200 0201

Данные 66h B2h 05h A0h 0Dh 3Dh

Проверьте правильность работы программы.

3. Напишите и выполните программу записи данных в память в соответствии с табл. 3.23. Для этого используйте команды загрузки регистровой пары **HL** и регистра **A** и команды записи в память регистровой пары **HL** и регистра **A** по прямому адресу.

Таблица 3.23

Варианты заданий записи данных в память

Вариант 1

Адрес 0100 0101 0111 0115 0120 0121

Данные 22h 17h 09h 0Ah 80h 90h

Вариант 2

Адрес 0150 0201 0202 0210 0212 0213

Данные 35h 42h 1Ah 0Ah C0h D3h

Вариант 3

Адрес 0101 0102 0120 0125 0135 0136

Данные 16h 1Ah 50h 60h 0Ah 2Dh

Вариант 4

Адрес 0100 0105 0107 0108 0110 0124

Данные 25h 13h 43h 56h 0Ah C0h

Вариант 5

Адрес 0210 0211 0220 0221 0228 0240

Данные 0Fh 10h 12h D3h B1h 1Ch

Вариант 6

Адрес 0150 0151 0160 0162 0200 0201

Данные 65h B4h 05h A1h 0Ch 35h

Проверьте правильность работы программы.

4. Напишите и выполните программу загрузки регистров **B, C, D, E, H, L** из памяти в соответствии с табл. 3.24. Используйте команды чтения памяти в регистр **A** по непосредственному адресу и команды пересылки. Значение соответствующих ячеек памяти необходимо заполнить вручную.

Таблица 3.24

Варианты заданий загрузки регистров

Вариант 1

Адрес 0100h 0105h 0107h 010Ch 0120h 0126h

Данные 00h 01h 15h 1Ah 1Bh 22h

Регистр **B C D E H L**

Вариант 2

Адрес 0200h 0205h 0215h 020Ch 0216h 0220h

Данные 35h 26h 51h A1h B0h 11h

Регистр **C B D H E L**

Вариант 3

Адрес 0101h 0110h 0125h 0133h 0135h 0145h

Данные 16h 19h 50h A6h 0Bh 21h

Регистр **B D C E H L**

Окончание табл. 3. 24

Вариант 4

Адрес 0100h 0105h 0110h 0111h 0112h 0126h

Данные 21h 35h 44h 56h 0Ah C0h

Регистр **B C D E H L**

Вариант 5

Адрес 0200h 0211h 0221h 0226h 0230h 0240h

Данные F0h 10h 19h 13h B0h 23h

Регистр **C B D H E L**

Вариант 6

Адрес 0100h 0105h 010Ch 0115h 0120h 0130h

Данные 69h 37h 56h 0Ah D6h 03h

Регистр **L D C E H B**

Проверьте правильность работы программы.

5. Напишите и выполните программу загрузки регистров **B, C, D, E, H, L** из памяти в соответствии с табл. 3.25. Используйте команды чтения памяти в регистровую пару **HL** и команды пересылки. Значение соответствующих ячеек памяти необходимо заполнить вручную.

Таблица 3.25

Варианты заданий загрузки регистров

Вариант 1

Адрес 0100h 0101h 0108h 0109h 0121h 0122h

Данные 01h 05h 25h 10h 1Bh 2Ah

Регистр **B C D E L H**

Вариант 2

Адрес 0200h 0201h 0215h 0216h 0220h 0221h

Данные 35h 26h 51h 0A1h 0B0h 11h

Регистр **B C D E L H**

Вариант 3

Адрес 0101h 0102h 0125h 0126h 0135h 0136h

Данные 16h 29h 60h 0A7h 3Bh A1h

Регистр **B C D E L H**

Вариант 4

Адрес 0110h 0111h 0150h 0151h 0212h 0213h

Данные 99h 35h 88h 56h 5Ah 0Ch

Регистр **B C D E L H**

Вариант 5

Адрес 0200h 0201 0221 0222h 0230h 0231

Данные 0F5h 0EEh 1Eh 1Ah 0BBh 33h

Регистр **B C D E L H**

Вариант 6

Адрес 0105h 0106h 010Ch 010Dh 0183h 0184h

Данные 19h 38h 96h 0A1h 16h 13h

Регистр **B C D E L H**

Проверьте правильность работы программы.

6. Напишите и выполните программу перезаписи данных из одних ячеек памяти (адрес 1) в другие (адрес 2) в соответствии с табл. 3.26. Значение соответствующих ячеек памяти необходимо заполнить вручную.

Таблица 3.26

Варианты заданий перезаписи данных

Вариант 1

Адрес 1 0100h 0101h 0108h 0109h 0121h 0125h

Адрес 2 0200h 0201h 0208h 0209h 0235h 0240h

Данные 11h 51h 25h 1Ah 10h 2Bh

Вариант 2

Адрес 1 0200h 0201h 0215h 0216h 0220h 0229h

Адрес 2 0250h 0251h 0208h 0209h 0135h 0140h

Данные 35h 26h 51h A1h B0h 11h

Вариант 3

Адрес 1 0101h 0102h 0120h 0126h 0135h 0136h

Адрес 2 0300h 0301h 0308h 0310h 0335h 0336h

Данные 26h 29h 61h A8h 4Bh 11h

Вариант 4

Адрес 1 0110h 0111h 0150h 0180h 0212h 0213h

Адрес 2 0200h 0201h 0205h 0207h 0250h 0251h

Данные 9Ah 35h 90h 58h 6Ah 1Ch

Вариант 5

Адрес 1 0200 0201 0221 0222h 0230h 0240h

Адрес 2 0100 0101 0121 0122h 0140h 0150h

Данные F7h E0h 10h A6h 0Bh 33h

Вариант 6

Адрес 1 0105h 0106h 010Ch 010Dh 0180h 0190h

Адрес 2 0210h 0211h 0220h 0221h 0290h 0300h

Данные 1Ah 3Dh 96h A1h 1Eh 1Dh

Проверьте правильность работы программы.

7. Напишите и выполните программу записи в память содержимого регистров в соответствии с табл. 3.27, используя команды записи в память с косвенной адресацией (через регистровую пару **HL**).

Таблица 3.27

Варианты заданий записи в память содержимого регистров

Регистры **A B C D E H L**

Вариант 1

Адрес 0100h 0101h 0102h 0103h 0104h 0105h 0106h

Данные FFh EEh BBh 00h AAh 01h 06h

Вариант 2

Адрес 0200h 0201h 0202h 0203h 0204h 0205h 0206h

Данные 00h E1h B1h 01h A1h 02h 06h

Окончание табл. 3. 27

Вариант 3

Адрес 0110h 0111h 0112h 0113h 0114h 0115h 0116h

Данные 34h 12h B0h 50h A5h 01h 16h

Вариант 4

Адрес 0125h 0126h 0127h 0128h 0129h 0130h 0131h

Данные 99h E1h 0Bh 05h 00h 01h 31h

Вариант 5

Адрес 0130h 0131h 0132h 0133h 0134h 0135h 0136h

Данные 22h E7h 05h 06h 05h 01h 36h

Вариант 6

Адрес 0100h 0101h 0102h 0103h 0104h 0105h 0106h

Данные 98h 10h 20h 30h 22h 01h 06h

Проверьте правильность работы программы.

8. Напишите и выполните программу чтения содержимого памяти в соответствующие регистры (табл. 3.28), используя команды чтения из памяти с косвенной адресацией (через регистровую пару **HL**).

Таблица 3.28

Варианты заданий чтения содержимого памяти

Регистры

D B C A E H L

Вариант 1

Адрес 0100h 0101h 0102h 0103h 0104h 0105h 0106h

Данные 0FFh 0EEh 0BBh 00h 0AAh 01h 06h

Вариант 2

Адрес 0200h 0201h 0202h 0203h 0204h 0205h 0206h

Данные 00h 0E1h 0B1h 01h 0A1h 02h 06h

Вариант 3

Адрес 0110h 0111h 0112h 0113h 0114h 0115h 0116h

Данные 34h 12h 0B0h 50h 0A5h 01h 16h

Вариант 4

Адрес 0125h 0126h 0127h 0128h 0129h 0130h 0131h

Данные 99h 0E1h 0Bh 05h 00h 01h 31h

Вариант 5

Адрес 0130h 0131h 0132h 0133h 0134h 0135h 0136h

Данные 22h 0E7h 05h 06h 05h 01h 36h

Вариант 6

Адрес 0100h 0101h 0102h 0103h 0104h 0105h 0106h

Данные 98h 10h 20h 30h 22h 01h 06h

Проверьте правильность работы программы.

9. Напишите и выполните программу записи данных в две области памяти, используя для адресации регистровую пару **BC** и регистровую пару **DE** в соответствии с табл. 3.29.

Таблица 3.29

Варианты заданий записи данных в две области памяти

Вариант 1

Адрес 1 0100h 0101h 0108h 0109h 0121h 0125h

Адрес 2 0200h 0201h 0208h 0209h 0235h 0240h

Данные 11h 51h 25h 1Ah 10h 2Bh

Вариант 2

Адрес 1 0200h 0201h 0215h 0216h 0220h 0229h

Адрес 2 0250h 0251h 0208h 0209h 0135h 0140h

Данные 35h 26h 51h A1h B0h 11h

Вариант 3

Адрес 1 0101h 0102h 0120h 0126h 0135h 0136h

Адрес 2 0300h 0301h 0308h 0310h 0335h 0336h

Данные 26h 29h 61h A8h 4Bh 11h

Вариант 4

Адрес 1 0110h 0111h 0150h 0180h 0212h 0213h

Адрес 2 0200h 0201h 0205h 0207h 0250h 0251h

Данные 9Ah 35h 90h 58h 6Ah 1Ch

Вариант 5

Адрес 1 0200h 0201h 0221h 0222h 0230h 0240h

Адрес 2 0100h 0101h 0121h 0122h 0140h 0150h

Данные F7h E0h 10h A6h 0Bh 33h

Вариант 6

Адрес 1 0105h 0106h 010Ch 010Dh 0180h 0190h

Адрес 2 0210h 0211h 0220h 0221h 0290h 0300h

Данные 1Ah 3Dh 96h A1h 1Eh 1Dh

Проверьте правильность работы программы.

10. Напишите и выполните программу перезаписи данных из одной области памяти (адресуйте через **BC**) в другую область памяти (адресуйте через **DE**) в соответствии с табл. 3.30. Значение соответствующих ячеек памяти необходимо заполнить вручную.

Таблица 3.30

Варианты заданий загрузки регистров

Вариант 1 Адрес 1 0100h 0105h 0108h 0109h 0121h 0125h

Адрес 2 0200h 0205h 0208h 0209h 0221h 0225h

Данные **11h 51h 25h 1Ah 10h 2Bh**

Вариант 2 Адрес 1 0200h 0201h 0215h 0216h 0220h 0229h

Адрес 2 0100h 0101h 0115h 0116h 0120h 0129h

Данные **35h 26h 51h A1h B0h 11h**

Вариант 3 Адрес 1 0101h 0102h 0120h 0126h 0135h 0136h

Адрес 2 0301h 0302h 0320h 0326h 0335h 0336h

Данные **26h 29h 61h A8h 4Bh 11h**

Вариант 4 Адрес 1 0110h 0120h 0150h 0180h 0212h 0213h

Адрес 2 0210h 0220h 0050h 0080h 0112h 0113h

Данные **9Ah 35h 90h 58h 6Ah 1Ch**

Вариант 5 Адрес 1 0200h 0201h 0221h 0222h 0230h 0240h

Адрес 2 0100h 0101h 0121h 0122h 0130h 0140h

Данные **F7h E0h 10h A6h 0Bh 33h**

Вариант 6 Адрес 1 0105h 0106h 010Ch 010Dh 0180h 0190h

Адрес 2 0205h 0206h 020Ch 020Dh 0280h 0290h

Данные 1Ah 3Dh 96h A1h 1Eh 1Dh

Проверьте правильность работы программы.

77

Лабораторная работа № 3

Арифметические команды

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 предполагают, 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 аккумулятор. 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 адресу в **HL** и регистровых пар на 1. Затрагиваемые биты признаков аналогичны команде инкремент.

Программа работы

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 (табл. 3.31), реализующей

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

Таблица 3.31

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **ADD B** 80 $A \leftarrow A+B$

0001 **LXI H**, 0100h 21 00 01 Загрузка $HL \leftarrow 0100h$

0004 **ADD M** 86 $A \leftarrow A+(M)$

0005 **ADI 1** C6 01 $A \leftarrow A+1$

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.32 и проверьте полученные результаты.

Таблица 3.32

Варианты заданий

Вариант 1 2 3 4 5 6

A 00h 00h 00h F0h FFh 55h

B 00h 02h 10h 0Eh 00h AAh

M 00h 03h 45h 00h 00h FFh

Вариант 7 8 9 10 11 12

A 00h 0Ah AAh 0Fh F0h 66h

B 05h 02h 11h 0Eh 01h 0Ah

M 10h 13h 45h 15h 09h 11h

Пример: Запишите в памяти, начиная с адреса 0000h, коды программы сложения 16-разрядных чисел, используя команды 8-разрядного сложения (табл. 3.33)

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

Таблица 3.33

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **MOV A**, C 79

0001 **ADD E** 83 Сложение младших байтов, установка бита переноса, если переполнение

0002 **MOV L**, A 6F Младший байт результата в регистр **L**

0003 **MOV A**, B 78

0004 **ADC D** 8A Сложение старших байтов с учетом переноса

0005 **MOV H**, A 67 Старший байт результата в регистр **H**

Выполните программ, предварительно задавая исходные значения в соответствии с табл. 3.34. Проверьте результат.

Таблица 3.34

Варианты заданий

Вариант 1 2 3 4 5 6

BC 0001h 02C5h F000h 8137h 809Fh FFFFh

DE 00FEh 03F1h 0FFFh 72D9h 8121h 0000h

Вариант 7 8 9 10 11 12

BC 0005h 01C0h F100h 1234h 819Fh 00FFh

DE 00F0h 02C1h 0FF0h 70D0h 8122h 0001h

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 (табл. 3.35), реализующей функцию

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

Таблица 3.35

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **SUB B** 90 $(A)=(A)-(B)$

0001 **LXI H**, 0100h 21 00 01 Загрузка **HL=0100h** адрес **M**

0004 **SUB M** 96 $(A)=(A)-(M)$

0005 **SBI 1** DE 01 $(A)=(A)-1$

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.36. Проверьте полученные результаты.

80

Таблица 3.36

Варианты заданий

Вариант 1 2 3 4 5 6

A FFh 00h 01h 25h 00h 05h

B 01h FFh 01h 20h 00h 06h

M 01h 00h 00h 04h 10h FFh

Вариант 7 8 9 10 11 12

A F0h 03h 11h 15h 03h 15h

B 01h F0h 13h 10h 01h 06h

M 02h 11h 02h 04h 02h F0h

Пример: Запишите в памяти, начиная с адреса **0000h**, коды программы вычитания 16-разрядных чисел (табл. 3.37)

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

Таблица 3.37

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **MOV A, E** 7B

0001 **SUB C** 91 Вычитание младший байтов $(A)=(E)-(C)$

0002 **MOV L, A** 6F Если **E<C**, то перенос=1

0003 **MOV A, D** 7A

0004 **SBB B** 98 Вычитание старшего байта с учетом переноса $(D)-(B)-C$

0005 **MOV H, A** 67

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.38. Проверьте полученный результат.

Таблица 3.38

Варианты заданий

Вариант 1 2 3 4 5 6

BC 0001h 02C5h F000h 8137h 809Fh FFFFh

DE 00FEh 03F1h 0FFFh 72D9h 8121h 0000h

Вариант 7 8 9 10 11 12

BC 0005h 01C0h F100h 1234h 819Fh 00FFh

DE 00F0h 02C1h 0FF0h 70D0h 8122h 0001h

81

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 (табл. 3.39), реализующей $(HL)=(BC)+(DE)$

Таблица 3.39

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **MOV H, B** 60 Пересылка $(H) \leftarrow (B)$

0001 **MOV L, C** 69 Пересылка $(H) \leftarrow (B)$

0002 **DAD D** 19 $(HL)=(HL)+(DE)$

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.40. Проверьте результаты.

Таблица 3.40

Варианты заданий

Вариант 1 2 3 4 5 6

BC 0000h 7FFFh 8000h 55AAh ECB9h FFFFh

DE 7FFFh 8000h 8000h AA55h 1347h 8000h

Вариант 7 8 9 10 11 12

BC 0100h 7000h 1000h 11AAh ECB Dh F00Fh

DE 7FF0h 8010h AA00h 0055h 0001h 8011h

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

Таблица 3.41

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **INR R** -- $(R)=(R)+1$

Выполните данную команду для следующих регистров и исходных значений и проверьте полученные результаты (табл. 3.42).

Таблица 3.42

Варианты заданий

Вариант 1 2 3 4 5 6

R A B C D E H

Исх.зн. 01h 05h 07h 10h 12h 13h

Вариант 7 8 9 10 11 12

R L A B D E H

Исх.зн. 1Ah 1Dh 50h CAh 0Bh ABh

Пример: Запишите в памяти, начиная с адреса **0000h**, коды команд (табл. 3.43).

Таблица 3.43

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **LXI H, 0100h** 21 00 01 Загрузить **HL=0100h**, адрес **M**

0003 **INR M** 34 (M)=(M)+1

Выполните данную последовательность команд, для следующих исходных значений содержимого ячейки памяти (табл. 3.44) и проверьте полученные результаты.

Таблица 3.44

Варианты заданий

Вариант 1 2 3 4 5 6

Исх.зн. 11h 05h 07h 1Ch 12h 19h

Вариант 7 8 9 10 11 12

Исх.зн. 1Ah 10h 51h C0h 0Bh A0h

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

Таблица 3.45

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **INX R -- (R)=(R)+1**

Выполните данную команду, для следующих пар регистров и исходных значений и проверьте полученные результаты (табл. 3.46).

Таблица 3.46

Варианты заданий

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

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 Таблица 3.47

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **DCR R -- (R)=(R)-1**

Выполните данную команду, для следующих регистров и исходных значений и проверьте полученные результаты (табл. 3.48).

Таблица 3.48

Варианты заданий

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

Таблица 3.49

Пример кода программы

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 (табл. 3.50) и проверьте полученные результаты.

Таблица 3.50

Варианты заданий

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

Таблица 3.51.

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **DCX R 2B (R)=(R)-1**

Выполните данную команду, для следующих пар регистров и исходных значений и проверьте полученные результаты (табл. 3.52).

Таблица 3.52

Варианты заданий

Вариант 1 2 3 4 5 6

R BC DE HL BC DE HL

Исх.зн. 010Fh 05AAh 0700h 0F00h FA12h AF1Fh

Вариант 7 8 9 10 11 12

R BC DE HL BC DE HL

Исх.зн. 1AF0h 1D35h 5100h CA01h 0B0Ah AB0Ah

Контрольные задания

1. Напишите и выполните команду реализующую $(C)=(D)+(E)$, в соответствии с табл. 3.53.

Таблица 3.53

Варианты заданий

Вариант **1 2 3 4 5 6**

D 10h FFh C7h 19h AAh E5h

E 80h 01h 08h 49h 55h F0h

Вариант **7 8 9 10 11 12**

D 70h 0Fh C5h 12h 0Ah EEh

E 81h 11h 88h 46h 50h F0h

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**

Таблица 3.54

Варианты заданий

Вариант **1 2 3 4 5 6**

M2 00h FEh D5h 22h 61h 19h

M3 F0h 02h C2h BBh 95h 33h

Вариант **7 8 9 10 11 12**

M2 11h F0h DDh 23h 66h 18h

M3 F1h 12h 02h B1h 09h 3Ah

3. Напишите и выполните программу сложения

$(HL)=(BC)+(E)+4E5Fh$ и заполните таблицу

Таблица 3.55

Варианты заданий

Вариант **1 2 3 4 5 6**

E 00h FEh D5h 22h 61h 19h

BC 0AFFh 00FEh 0123h 0A15h 2122h 2223h

Вариант **7 8 9 10 11 12**

E 11h F0h DDh 23h 66h 18h

BC 0AF0h A100h 9123h 0615h 2100h 0E23h

86

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

Таблица 3.56

Варианты заданий

Вариант **1 2 3 4 5 6**

D 01h DAh F0h 85h 9Fh FFh

E FEh 10h 0Fh 72h 81h 0Ah

Вариант 7 8 9 10 11 12

D 16h 0Ch 39h 75h F0h EEh

E 98h 10h 01h 02h 16h DDh

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

Таблица 3.57

Варианты заданий

Вариант 1 2 3 4 5 6

M2 AAh 0Eh D5h 22h 61h 19h

M3 F0h 02h C2h B0h 91h 3Ah

Вариант 7 8 9 10 11 12

M2 11h F0h D0h 21h 66h 18h

M3 FFh 12h 02h B1h 19h 36h

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

Таблица 3.58

Варианты заданий

Вариант 1 2 3 4 5 6

E 00h F0h 15h 23h 62h 18h

BC 0AF0h 01F1h 5123h 1A14h 2112h 2222h

Вариант 7 8 9 10 11 12

E 11h F1h 15h 26h 67h 00h

BC 0A12h 0101h 5123h 1A15h 2113h 2332h

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

Таблица 3.59

Варианты заданий

Вариант 1

Базовый элемент массива 0100h

Номер элемента 00h 03h 06h 11h 18h 1Fh

Содержимое элемента массива 00h 01h 02h 03h 04h 05h

Вариант 2

Базовый элемент массива 0200h

Номер элемента 00h 03h 07h 10h 12h 15h

Содержимое элемента массива 01h 11h 02h AAh B4h 15h

Вариант 3

Базовый элемент массива 0300h

Номер элемента 01h 05h 06h 12h 1Ah 1Bh

Содержимое элемента массива 00h 22h 33h 44h 04h 55h

Вариант 4

Базовый элемент массива 0150h

Номер элемента 02h 07h 0Ah 0Bh 12h 13h

Содержимое элемента массива 05h 07h 12h 13h 42h 55h

Вариант 5

Базовый элемент массива 0050h

Номер элемента 00h 01h 03h 05h 07h 09h

Содержимое элемента массива 05h 17h F1h BAh ABh 33h

Вариант 6

Базовый элемент массива 0100h

Номер элемента 00h 02h 04h 06h 08h 0Ah

Содержимое элемента массива A0h A1h A2h A2h A3h AFh

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

Таблица 3.60

Варианты заданий

Вариант 1

Базовый элемент массива 2 0200h

Номер элемента 2 02h 04h 06h 08h 0Ah 0Ch

Вариант 2

Базовый элемент массива 2 0100h

Номер элемента 2 02h 04h 07h 08h 0Ah 0Bh

Окончание табл. 3.60

Вариант 3

Базовый элемент массива 2 0200h

Номер элемента 2 01h 02h 03h 04h 05h 06h

Вариант 4

Базовый элемент массива 2 0300h

Номер элемента 2 00h 01h 03h 07h 09h 0Bh

Вариант 5

Базовый элемент массива 2 0150h

Номер элемента 2 01h 02h 06h 07h 09h 0Ah

Вариант 6

Базовый элемент массива 2 0200h

Номер элемента 2 01h 02h 03h 04h 05h 06h

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

Таблица 3.61

Варианты заданий

Вариант 1 2 3

Массив 0050h – 0054h 0060h – 0064h 0070h – 0074h

Данные 00h – 04h 10h – 14h 11h – 15h

Вариант 4 5 6

Массив 0080h – 0084h 0090h – 0094h 0100h – 0104h

Данные 00h – 04h 0Ah – 0Eh 20h – 24h

Вариант 7 8 9

Массив 0200h – 0204h 0210h – 0214h 00220h – 00224h

Данные 50h – 54h 1Ah – 1Eh 00h – 04h

Вариант 10 11 12

Массив 0230h – 0234h 0240h – 0244h 0250h – 0254h

Данные 0Bh – 0Fh 1Bh – 1Fh 05h – 09h

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

Таблица 3.62

Варианты заданий

Вариант 1 2 3

Массив 0054h – 0050h 0064h – 0060h 0074h – 0070h

Данные 05h – 01h 16h – 12h 19h – 15h

Окончание табл. 3.60

Вариант 4 5 6

Массив 0084h – 0080h 0094h – 0090h 0104h – 0100h

Данные 03h – FFh 0Eh – 0Ah 25h – 21h

Вариант 7 8 9

Массив 0204h – 0200h 0214h – 0210h 00224h – 0220h

Данные 54h – 50h 1Eh – 1Ah 09h – 05h

Вариант 10 11 12

Массив 0234h – 0230h 0244h – 0240h 0254h – 0250h

Данные 0Fh – 0Bh 1Fh – 1Bh 0Ah – 06h

Лабораторная работа №4

Логические команды

Теоретические сведения

Для реализации логических операций в системе 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 или 00110010 = 10111011,

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 и 00110010 = 00100000

Команды «ИСКЛЮЧАЮЩЕГО ИЛИ» реализуют логическую операцию сложения по модулю два. Результат равен 1, если соответ-

ствующие биты противоположны $\langle 1 \text{ и } 0 \rangle$, и равен 0, если они одинаковы. Например

10101001 ИСКЛЮЧАЮЩЕЕ ИЛИ 00110010 = 10011011

Команда инверсии реализует операцию «**ОТРИЦАНИЕ**» содержимого аккумулятора. Например

10101001 ОТРИЦАНИЕ 01010110.

Программа работы

1. Команды логического сложения

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

91

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

ORI d8 с непосредственным операндом.

Пример: Запишите в память, начиная с адреса **0000h** коды программы (табл. 3.63), реализующей выражение

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

Таблица 3.63

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 ORA C B1 $A \leftarrow A \square C$

0001 LXI H, 0100h 21 00 01 $HL \leftarrow 0100h$

0004 ORA M B6 $A \leftarrow A \square M$

0005 ORI 80h F9 80 $A \leftarrow A \square 80h$

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.65

Таблица 3.64

Варианты заданий

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

Запишите в память, начиная с адреса **0000h**, коды программы, реализующей выражения (табл. 3.65)

$HL \leftarrow BC \square DE$

Таблица 3.65

Пример кода программы

Адрес Команда Машинный код

Комментарий

0000 MOV A, C 79 Пересылка $A \leftarrow C$

0001 ORA E B3 $A \leftarrow A \square E$

0002 MOV L, A 6F Пересылка $L \leftarrow A$, младшего байта ре-

зультата.

0003 **MOV A, B** 78 Пересылка $A \leftarrow B$

92

Окончание табл. 3.65

0004 **ORA D** B2 $A \leftarrow A \square D$

0005 **MOV H, A** 67 Пересылка $H \leftarrow A$, старшего байта ре-

зультата.

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.66

Таблица 3.66

Варианты заданий

Вариант 1 2 3 4 5 6

BC 0111h 0B22h 0C00h 0F0Dh 0E00h 0F10h

DE 1200h 0100h 0A00h 1011h 1001h 1100h

Вариант 7 8 9 10 11 12

BC 02AAh 1001h 1011h 2200h AA01h 9925h

DE 1100h 05EAh 0122h 1033h 0010h 1011h

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 (табл. 3.67)

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

Таблица 3.67

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **ANA C** A2 $A \leftarrow A \square D$

0001 **LXI H, 0100h** 21 00 01 $HL \leftarrow 0100h$

0004 **ANA M** A6 $A \leftarrow A \square M$

0005 **ANI 7Fh** E6 7F $A \leftarrow A \square 7Fh$

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.68

Таблица 3.68

Варианты заданий

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

93

Окончание табл. 3.68

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 (табл. 3.69)

HL←BC□DE

Таблица 3.69

Пример кода программы

Адрес Команда Машинный код

Комментарий

0000 **MOV A, C** 79 Пересылка **A←C**

0001 **ANA E** A3 **A←A□E**

0002 **MOV L, A** 6F Пересылка **L←A** младшего байта результата

0003 **MOV A, D** 7A Пересылка **A←D**

0004 **ANA B** A0 **A←A□B**

0005 **MOV H, A** 67 Пересылка **H←A**, старшего байта результата.

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.70

Таблица 3.70

Варианты заданий

Вариант 1 2 3 4 5 6

BC 0111h 0B22h 0C00h 0F0Dh 0E00h 0F10h

DE 1200h 0100h 0A00h 1011h 1001h 1100h

Вариант 7 8 9 10 11 12

BC 02AAh 1001h 1011h 2200h AA01h 9925h

DE 1100h 05EAh 0122h 1033h 0010h 1011h

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.

94

Пример: Запишите в память, начиная с адреса **0000h** коды программы, реализующей выражение (табл. 3.71)

A←A□C□M□AAh

Таблица 3.71

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **XRA A** AF **A←A□A, A=0**

0001 **XRA E** AB **A←A□E**

0002 **LXI H, 900h** 21 00 09 **HL←0100h**

0005 **XRA M** AE **A←A□M**

0006 **XRI 7Fh** EE AA **A←A□AAh**

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.72

Таблица 3.72

Варианты заданий

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 (табл. 3.73)

HL ← (A □ B) □ DE □ (HL □ C)

Таблица 3.73

Пример кода программы

Адрес Команда Машин-
ный код

Комментарий

0000 **ORA B** B0 A ← A □ B

0001 **ANA E** A3 A ← A □ E

0002 **MOV A, C** 79 Пересылка A ← C

0003 **XRA L** A A ← A □ L

0004 **XRA E** AB A ← A □ E

95

Окончание табл. 3.73

0005 **MOV L, A** 6F Пересылка L ← A, младшего байта результата

0006 **MOV A, 7C** Пересылка A ← H, старшего байта результата

0007 **XRA D** AA A ← A □ D

0008 **MOV H, 67** Пересылка H ← A, старшего байта результата

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.74

Таблица 3.74

Варианты заданий

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 **исх** 3355h 7921h 68ACh DBF1h DB1Eh D2FEh

C A2h 36h 78h C5h 1Dh 0Dh

4. Команда ИНВЕРСИЯ.

СМА инверсия аккумулятора.

Пример: Запишите в память, начиная с адреса 0000h коды программы, реализующей выражение (табл. 3.75)

A ← NOT B AND NOT C

Таблица 3.75

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **MOV A, B** 78 Пересылка $A \leftarrow B$

0001 **CMA 2F** $A \leftarrow \text{NOT } A$

0002 **MOV B, A** 47 Пересылка $B \leftarrow A$

0003 **MOV A, C** 79 Пересылка $A \leftarrow C$

0004 **CMA 2F** $A \leftarrow \text{NOT } A$

0005 **ANA B A0** $A \leftarrow A \square B$

0006 **CMA 2F** $A \leftarrow \text{NOT } A$

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.76

96

Таблица 3.76

Варианты заданий

Вариант 1 2 3 4 5 6

B 00h FFh 25h 39h ABh EDh

C FFh 02h 56h 21h 13h C3h

A FFh FFh 77h 39h BBh EFh

Пример: Запишите в память, начиная с адреса **0000h**, коды программы, реализующей выражения (табл. 3.77)

$M3 \leftarrow \text{NOT} (\text{NOT } M1 \square \text{NOT } M2)$

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

Таблица 3.77

Пример кода программы

Адрес Команда Машинный код

Комментарий

000 **LXI H,**

900h

21 00 09 **HL** \leftarrow **900h** адрес **M**

003 **MOV A, M** 7E $A \leftarrow M1$

004 **CMA 2F** $A \leftarrow \text{NOT } M1$

005 **MOV C, A** 4F Пересылка $C \leftarrow A$, промежуточного результата

006 **INX H** 23 **HL** \leftarrow **HL**+1, адрес **M2**

007 **MOV A, M** 7E $A \leftarrow M2$

008 **CMA 2F** $A \leftarrow \text{NOT } M2$

009 **ORA C B1** $A \leftarrow A \square C$

00A **CMA 2F** $A \leftarrow \text{NOT } A$

00B **INX H** 23 **HL** \leftarrow **HL**+1, адрес **M3**

00C **MOV M, A** 77 $M \leftarrow A$

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.78.

Таблица 3.78

Варианты заданий

Вариант 1 2 3 4 5 6

M1 37h 43h 09h 78h 78h 78h

M2 29h 5Eh F6h 95h 95h 95h

M3 21h 42h 94h 10h 10h 10h

97

Контрольные задания

1. Напишите и выполните программы реализации выражения (табл. 3.79).

HL←B□C□DE□ 8800h

Таблица 3.79

Варианты заданий

Вариант 1

Регистры **C B D E**

Данные 00h 10h 11h F0h

Вариант 2

Регистры **C B D E**

Данные 01h 11h A2h 00h

Вариант 3

Регистры **C B D E**

Данные 00h 11h 0Bh 0Ah

Вариант 4

Регистры **C B D E**

Данные 05h 06h 11h A0h

Вариант 5

Регистры **C B D E**

Данные A0h 0Ah 00h 11h

Вариант 6

Регистры **C B D E**

Данные 05h 19h 00h 12h

2. Напишите и выполните программу реализации выражения (табл. 3.80)

(M3)=(M1)□ (M2)

Адреса ячеек памяти соответственно равны

M1=0100h

M2=0101h

M3=0102h

Таблица 3.80

Варианты заданий

Вариант 1

Ячейки **M1 M2**

Данные 0Fh E0h

Вариант 2

Ячейки **M1 M2**

Данные 11h 02h

Вариант 3

Ячейки **M1 M2**

Данные AAh 55h

Вариант 4

Ячейки M1 M2

Данные 01h 10h

Вариант 5

Ячейки M1 M2

Данные 02h F0h

98

Окончание табл. 3.80

Вариант 6

Ячейки M1 M2

Данные 0Bh B1h

3. Напишите и выполните программы реализации выражения
(табл. 3.81)

HL ← (HL □ BC □ DE) □ 03FFh

Таблица 3.81

Варианты заданий

Вариант 1 Регистры **BC HL DE**

Данные 00FFh 0010h 1000h

Вариант 2 Регистры **BC HL DE**

Данные 0101h 2125h 2211h

Вариант 3 Регистры **BC HL DE**

Данные 2233h 4400h 1000h

Вариант 4 Регистры **BC HL DE**

Данные 0010h 5000h 300Fh

Вариант 5 Регистры **BC HL DE**

Данные 0F0Fh 0FA0h 0001h

Вариант 6 Регистры **BC HL DE**

Данные 0F0Bh 1000h 22FFh

4. Напишите и выполните программу реализации выражения
(табл. 3.82)

(M2) ← A □ (M1) □ C □ D

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

Таблица 3.82

Варианты заданий

Вариант 1 Регистры **A M1 C D**

Данные 00h 10h FFh 0Ah

Вариант 2 Регистры **A M1 C D**

Данные FFh 05h 15h 20h

Вариант 3 Регистры **A M1 C D**

Данные 0Eh 12h 22h 55h

Вариант 4 Регистры **A M1 C D**

Данные A1h 1Ah 11h BBh

Вариант 5 Регистры **A M1 C D**

Данные EEh 11h BBh 22h

Вариант 6

Регистры **A M1 C D**

Данные 0Fh 77h EEh B1h

99

5. Напишите и выполните программу реализации выражения
(табл. 3.83)

H ← L □ D □ E □ H □ B □ C □ 0Fh

Таблица 3.83

Варианты заданий

Вариант 1

Регистры **HL DE BC**

Данные 0000h 5610h FF48h

Вариант 2

Регистры **HL DE BC**

Данные FF00h 0235h 1B35h

Вариант 3

Регистры **HL DE BC**

Данные 1001h F00Fh 0CBAh

Вариант 4

Регистры **HL DE BC**

Данные E3E0h 15B1h B32Bh

Вариант 5

Регистры **HL DE BC**

Данные 2377h EDAEh B341h

Вариант 6

Регистры **HL DE BC**

Данные 0BC5h 1125h 2020h

6. Напишите и выполните программу реализации выражения
(табл. 3.84).

M ← M □ AAh □ H □ B □ E

Таблица 3.84

Варианты заданий

Вариант 1

Регистры **M H B E**

Данные EEh B1h EEh B1h

Вариант 2

Регистры **M H B E**

Данные FFh 05h FFh 0Ah

Вариант 3

Регистры **M H B E**

Данные 10h FFh 77h EEh

Вариант 4

Регистры **M H B E**

Данные EEh EEh 11h B1h

Вариант 5

Регистры **M H B E**

Данные 05h B1h FFh 0Ah

Вариант 6

Регистры **M H B E**

Данные FFh 05h 15h 20h

7. Напишите и выполните реализации выражения (таблица 3.85).

HL←NOT DE □ NOT BC □ NOT C □ NOT B

100

Таблица 3.85

Варианты заданий

Вариант 1

Регистры **D B C E**

Данные FFh 0Ch 33h 10h

Вариант 2

Регистры **D B C E**

Данные EEh 23h 32h F3h

Вариант 3

Регистры **D B C E**

Данные ACh 26h B2h 41h

Вариант 4

Регистры **D B C E**

Данные 12h 73h 00h 16h

Вариант 5

Регистры **D B C E**

Данные 1Ah A4h 53h AEh

Вариант 6

Регистры **D B C E**

Данные F4h C1h 22h EEh

8. Напишите и выполните реализации выражения (табл. 3.86).

M6← NOT M1 □ M2 □ NOT M3 □ NOT M4 □ M5

Адреса ячеек памяти

M1=900h M4=903h

M2=901h M5=904h

M3=902h M6=905h

Таблица 3.86

Варианты заданий

Вариант 1

Регистр **M1 M2 M3 M4 M5**

Данные FFh AAh A4h 00h 74h

Вариант 2

Регистр **M1 M2 M3 M4 M5**

Данные 00h BEh 4Bh 56h 00h

Вариант 3

Регистр **M1 M2 M3 M4 M5**

Данные 05h 26h 33h ACh FFh

Вариант 4

Регистр **M1 M2 M3 M4 M5**

Данные 34h BBh 51h C4h A7h

Вариант 5

Регистр **M1 M2 M3 M4 M5**

Данные 56h FAh 43h DCh 00h

Вариант 6

Регистр **M1 M2 M3 M4 M5**

Данные C3h FFh 15h ADh CCh

Лабораторная работа № 5

Команды сравнения

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 (табл. 3.87).

Таблица 3.87

Значения флагов при выполнении команды сравнения

Результат сравнения Признак

(Ноль, бит Z) (Перенос C)

Равно 1 0

Больше 0 0

Меньше 0 1

Главным образом эти команды используются перед командами условных переходов (переход по признаку), которые будут рассматриваться в последующей лабораторной работе.

Программа работы

1. Команды сравнения содержащего регистра с содержимым аккумулятора.

СМР А команда сравнения регистра **A** с содержимым регистра **A**

СМР В команда сравнения регистра **A** с содержимым регистра **B**

СМР С команда сравнения регистра **A** с содержимым регистра **C**

СМР D команда сравнения регистра **A** с содержимым регистра **D**

СМР E команда сравнения регистра **A** с содержимым регистра **E**

СМР H команда сравнения регистра **A** с содержимым регистра **H**

СМР L команда сравнения регистра **A** с содержимым регистра **L**

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

Таблица 3.88

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **MOV A, C** 79 Пересылка (A)←(C)

0001 **CMP B** B8 Сравнение с (B), (A)-(B)

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.89. Проверьте полученные результаты.

Таблица 3.89

Варианты заданий

C 01h 01h FFh 43h 55h 03h 20h

B 01h 09h 00h FFh 55h 55h 15h

Ноль **Z** 1 0 0 0 1 0 0

Перенос **C** 0 1 0 1 0 0 0

Знак **S** 0 1 1 1 0 0 0

Четность **P** 1 0 1 1 1 1 0

2. Команды сравнения с памятью

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

Пример: Запишите в память, начиная с адреса **0000h**, коды программы сравнения с содержимым ячейки памяти (табл. 3.90).

Таблица 3.90

Пример кода программы

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 табл. 3.91. Проверьте полученные результаты.

103

Таблица 3.91

Пример программы

A 21h **BAh** **E9h**

M=0100h 00h 21h 39h 19h FFh **BAh** **E9h** 10h **F5h**

Ноль **Z** 0 1 0 0 0 1 1 0 0

Перенос **C** 0 0 1 0 1 0 0 0 1

Знак **S** 0 0 1 0 1 0 0 0 1

Четность **P** 1 1 1 0 1 1 1 0 0

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.

Таблица 3.92

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **CPI 7FH** FE 7F Сравнение A и 7Fh

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.93.

Таблица 3.93

Вариант задания

Вариант **1 2 3 4 5 6**

A **00h 80h 7Fh B3h 25h F7h**

Ноль **Z 0 0 1 0 0 0**

Перенос **C 1 0 0 0 1 0**

Знак **S 1 0 0 0 1 0**

Четность **P 1 0 1 0 1 1**

Контрольные задание

1. Напишите и выполните программу сравнения содержимого регистров **H** и **L**. Заполните табл. 3.94.

Таблица 3.94

Вариант задания

Вариант 1

H L Ноль **Z** Перенос **C** Знак **S** Четность **P**
FFh 0Fh

104

Окончание табл. 3.94

Вариант 2

H L Ноль **Z** Перенос **C** Знак **S** Четность **P**
00h 0Ah

Вариант 3

H L Ноль **Z** Перенос **C** Знак **S** Четность **P**
01h 00h

Вариант 4

H L Ноль **Z** Перенос **C** Знак **S** Четность **P**
ABh BAh

Вариант 5

H L Ноль **Z** Перенос **C** Знак **S** Четность **P**
10h 10h

Вариант 6

H L Ноль **Z** Перенос **C** Знак **S** Четность **P**
12h 11h

2. Напишите и выполните программу сравнения содержимого регистра **A** с содержимым ячейки памяти по адресу **0110h**. Заполните табл. 3.95.

Таблица 3.95

Вариант задания

Вариант 1

A M Ноль Z Перенос C Знак S Четность P
00h FFh

Вариант 2

A M Ноль Z Перенос C Знак S Четность P
05h 05h

Вариант 3

A M Ноль Z Перенос C Знак S Четность P
FFh 01h

Вариант 4

A M Ноль Z Перенос C Знак S Четность P
12h 15h

Вариант 5

A M Ноль Z Перенос C Знак S Четность P
0Ah 0Bh

Вариант 6

A M Ноль Z Перенос C Знак S Четность P
22h 20h

3. Напишите и выполните программу сравнения содержимого регистра **H** с непосредственным операндом. Заполните табл. 3.96.

Таблица 3.96

Вариант задания

Вариант 1

Операнд 5Ah

A 01h 10h 5Ah FFh 5Bh

Ноль Z

Перенос C

Знак S

Четность P

Вариант 2

Операнд 05h

A 10h 1Ah 05h 01h 51h

Ноль Z

Перенос C

Знак S

Четность P

Вариант 3

Операнд 10h

A 11h 10h 00h 01h 50h

Ноль Z

Перенос C

Знак S

Четность P

Вариант 4

Операнд FEh

A 01h FEh FFh F0h 5Ah

Ноль **Z**
 Перенос **C**
 Знак **S**
 Четность **P**
 Вариант 5
 Операнд 0Ah
 A 00h 11h 0Ah 1Fh 0Bh
 Ноль **Z**
 Перенос **C**
 Знак **S**
 Четность **P**
 Вариант 6
 Операнд 22h
 A 22h 11h 33h FFh 00h
 Ноль **Z**
 Перенос **C**
 Знак **S**
 Четность **P**

Лабораторная работа № 6

Команды сдвига

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

Таблица 3.97

Содержимой байта до выполнения команды циклического сдвига влево

Флаг переноса Данные

C b7 b6 b5 b4 b3 b2 b1 b0

После **RLC**

Таблица 3.98

Содержимой байта после выполнения команды циклического сдвига влево

Флаг переноса Данные

C b6 b5 b4 b3 b2 b1 b0 b7

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

Таблица 3.99

Содержимой байта до выполнения команды циклического сдвига вправо

Флаг переноса Данные

C b7 b6 b5 b4 b3 b2 b1 b0

После **RRC**

107

Таблица 3.100

Содержимой байта после выполнения команды циклического сдвига вправо

Флаг переноса Данные

b0 b0 b7 b6 b5 b4 b3 b2 b1

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

Таблица 3.101

Содержимой байта до выполнения команды сдвига влево

Флаг переноса Данные

C b7 b6 b5 b4 b3 b2 b1 b0

После **RAL**

Таблица 3.102

Содержимой байта после выполнения команды сдвига влево

Флаг переноса Данные

C b6 b5 b4 b3 b2 b1 b0 C

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

Таблица 3.103

Содержимой байта до выполнения команды сдвига вправо

Флаг переноса Данные

C b7 b6 b5 b4 b3 b2 b1 b0

После **RAR**

Таблица 3.104

Содержимой байта после выполнения команды сдвига вправо

Флаг переноса Данные

b0 C b7 b6 b5 b4 b3 b2 b1

108

Программа работы

1. Команды циклического сдвига.

RRC циклический сдвиг вправо;

RRC циклический сдвиг вправо.

Пример: Запишите в память, начиная с адреса 0000h, коды программы реализующей операцию циклического сдвига байта на 4 разряда, используя команды **RLC** (табл. 3.105).

Таблица 3.105

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **RLC** 07 ;циклический сдвиг влево на 1 разряд

0001 **RLC** 07 ;циклический сдвиг влево на 1 разряд

0002 **RLC** 07 ;циклический сдвиг влево на 1 разряд

0003 **RLC** 07 ;циклический сдвиг влево на 1 разряд

Выполните эту программу, предварительно задавая исходные значения в соответствии с табл. 3.106.

Таблица 3.106

Вариант задания

А исх. 00 0F F0 81 A5 67

А рез. 00 F0 0F 18 5A 76

Пример: Запишите в память, начиная с адреса 0000h, коды программы, реализующей операцию объединения старших тетрад байтов, в регистрах **B** и **C**, в один (табл. 3.107).

Таблица 3.107

Пример кода программы

Адрес Команда Машинный код

Комментарий

0000 **MOV A, C** 79

0001 **RRC** 0F перемещение старшей тетрады 1-го байта

0002 **RRC** 0F

0003 **RRC** 0F на место младшей тетрады.

0004 **RRC** 0F

0005 **ANI 0Fh** E6 0F выделение старшей тетрады 1-го байта.

0006 **MOV C, A** 4F

109

Окончание табл. 3.107

0007 **MOV A, B** 78

0008 **ANI F0h** E6 F0 выделение старшей тетрады 2-го байта.

000B **ORA C** B1 объединение двух байт в один.

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.108. Сравните результаты.

Таблица 3.108

Вариант задания

C 72h **F0h** **51h** **19h**

B 9Fh 0Fh A3h 86h

A рез. 97h 0Fh A5h 81h

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** (табл. 3.109).

Таблица 3.109

Пример кода программы

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

110

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.110.

Таблица 3.110

Вариант задания

HL исх. FFFCh 8002h 3578h FFFFh 0000h 1111h

HL рез. FFF9h 0005h 6AF0h FFFFh 0000h 2222h

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

Таблица 3.111

Пример кода программы

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

0004 **MOV B, A** 47 результат в B

ПРИМЕЧАНИЕ исходное значение не должно превышать 63.

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.112.

Таблица 3.112

Вариант задания

C 00h 02h 10h 2Fh 33h 3Ah

B 00h 08h 40h BCh CCh E8h

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**.

Таблица 3.113

Пример кода программы

Адрес Команда Машинный код

Комментарий

0000 **MOV A, H** 7C

0001 **ORA A** B7 сброс переноса в 0

0002 **RAR IF** A=A/2

111

Окончание табл. 3.113

0003 **ORA A** B7 сброс переноса в 0

0004 **RAR IF** A=A/2

0005 **ORA A** B7 сброс переноса в 0

0006 **RAR IF** A=A/2

0007 **MOV D, A** 57 **D=H/8** – целая часть

0008 **MOV A, H** 7C

0009 **ANI 07h** E6 07 выделение остатка результата

000B **MOV E, A** 5F остаток в **E**

Выполните программу, предварительно задавая исходные значения в соответствии с табл. 3.113. Проверьте результаты.

Таблица 3.113

Вариант задания

H 05h 08h 10h 35h 79h FFh

D 00h 01h 02h 06h 0Fh 1Fh

E 05h 00h 00h 05h 01h 07h

Контрольные задания

1. Напишите и выполните программу (табл. 3.114) объединения младших тетрад двух байтов в один. Младшую тетраду 2-го байта поместить в старшую тетраду результирующего байта.

Таблица 3.114

Вариант задания

Вариант 1

E (1 байт) **D** (2 байт)

F0h 11h

Вариант 2

E (1 байт) D (2 байт)

0Fh AAh

Вариант 3

E (1 байт) D (2 байт)

22h 11h

Вариант 4

E (1 байт) D (2 байт)

55h ABh

Вариант 5

E (1 байт) D (2 байт)

35h 53h

Вариант 6

E (1 байт) D (2 байт)

77h 01h

112

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 (табл. 3.115).

Таблица 3.115

Вариант задания

Вариант 1

L (1 байт) H (2 байт)

ABh 17h

Вариант 2

L (1 байт) H (2 байт)

F0h 1Ah

Вариант 3

L (1 байт) H (2 байт)

22h 11h

Вариант 4

L (1 байт) H (2 байт)

AAh BBh

Вариант 5

L (1 байт) H (2 байт)

77h 01h

Вариант 6

L (1 байт) H (2 байт)

35h 53h

3. Напишите и выполните программу циклического сдвига на 3 разряда содержимого пары регистров (табл. 3.116).

Таблица 3.116

Вариант задания

Вариант 1

DE

FF00h

Вариант 2

HL

22F0h

Вариант 3

BC

0220h

Вариант 4

DE

3563h

Вариант 5

HL

ABCDh

Вариант 6

BC

D32Ah

113

4. Напишите и выполните программу деления содержимого пары регистров *BC* на 8. $BC=BC/8$ (табл. 3.117).

Таблица 3.117

Вариант задания

Вариант 1

DE

1F40h

Вариант 2

HL

1DC8h

Вариант 3

BC

42B0h

Вариант 4

DE

5988h

Вариант 5

HL

A7C8h

Вариант 6

BC

4570h

Лабораторная работа № 7

Команды безусловного и условных переходов. ввод-вывод данных

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 битовпризнаков:

- ☐ 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 табл. 3.118.

Таблица 3.118

Значения флагов при выполнении команд переходов

Признак Ноль **Z** Перенос **C** Четность **P** Знак **S**

Команда **1 0 1 0 1 0 1 0**

JZ Да --- --- --- --- --- ---

JNZ --- Да --- --- --- --- ---

JS --- --- Да --- --- --- ---

JNC --- --- --- Да --- --- ---

JPE --- --- --- --- Да --- ---

JPO --- --- --- --- --- Да ---

JM --- --- --- --- --- --- Да ---

JP --- --- --- --- --- --- --- Да

115

Программа работы

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 (табл. 3.119).

Таблица 3.119

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **LXI H, 0100h** 21 00 01 Загрузка в **HL** адреса перехода

0003 **MVI A, 00h** 3E 00 Загрузка в **A=0**

0005 **INR A** Инкремент **A**

0006 **PCHL** E9 Переход **(PC)←(HL)**

0100 **JMP 0000h** Переход по адресу **0000h**

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 (табл. 3.120).

Таблица 3.120

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **MVI C, 10h** 0E 10 Загрузка **C←10h**, длина массива

0002 **LXI H, 100h** 21 00 01 **HL←0100h**, начальный адрес массива

0005 **MVI M, 0** 36 Загрузка **(HL) = 0**

0007 **INX H** 23 **HL←HL+1**, следующий адрес

0008 **DCR C** 0D **C = C-1**, длина массива

0009 **JNZ 005h** C2 05 00 Продолжать, если длина массива не равна 0

Проверьте результаты выполнения программы.

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

Таблица 3.121

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **MVI C, 10h** 0E 10 Загрузка в **C←10h**, длина массива

0002 **LXI H, 100h** 21 00 09 Загрузка **HL←100h**, начальный адрес массива

0005 **MOV A,M** 7E Чтение **A←(HL)**

0006 **CPI 0** FE 00 **A←0**

0008 **JZ 00Dh** CA 0D 00 Переход, если да

000B **MVI M,0** 36 00 Нет, загрузка **(HL)←0**

000D **INX H** 23 **HL←HL+1**, следующий адрес

000E **DCR C** 0D **C = C-1**, длина массива

000F **JNZ 005h** C2 05 00 Продолжать, если длина массива не равна 0

Проверьте результаты выполнения программы.

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 **B**. Если присут-

стует перенос, в аккумулятор записывается **FFh**, в противном случае, в аккумулятор кладется **00h** (табл. 3.122).

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

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

Таблица 3.122

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **MVI A,FFh** 3E FF Загрузка $A \leftarrow FFh$

0001 **MVI B,10h** 06 10 Загрузка $B \leftarrow 10h$

0003 **ORA B** B0 $A \leftarrow A \vee B$

0005 **JC 100h** DA 00 01 Если перенос = 1, перейти в ячейку с адресом 0100h

0006 **JNC 200h** D2 00 02 Если перенос = 0, перейти в ячейку с адресом 0200h

1000 **MVI A,FFh** 3E FF Загрузка $A \leftarrow FFh$

2000 **MVI A,00h** 3E 00 Загрузка $A \leftarrow 00h$

4. Команды перехода по принципу **P** – 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 **C** (табл. 3.123).

Таблица 3.123

Пример кода программы

Адрес Команда Машинный код Комментарий

0000 **MOV A,C** 79 Пересылка $A \leftarrow C$ исходного байта

0001 **ANI 7Fh** E6 7F Обнуление старшего разряда

0003 **ORA A** B7 $A \leftarrow A \vee A$ – установка бита четности

0004 **JPO** E2 09 00 Переход, если исходный байт четный

0007 **ORI 80h** F6 80 Дополнить до четности

0009 **MOV C,A** 4F Результат

Выполните программу, задавая исходные значения в соответствии с табл. 3.124. Проверьте результат.

Таблица 3.124

Варианты заданий

C исходное **00 FF B6 80 CD 75**

C результат **00 FF 36 00 4D F5**

5. Команды перехода по признаку – знак.

JM ADR переход, если $S = 1$.

JP ADR переход, если $S = 0$.

Команды ввод-вывод данных

Теоретическое обоснование

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** (табл. 3.125).

Таблица 3.125

Пример кода программы

Адрес Команда Машинный код

Комментарий

0000 **MVI A, 00H** 3E 00 Загрузка **A←00h**

0002 **MVI C, FFH** 0E FF Загрузка **C←FFh**

0004 **MVI D, FFH** 16 FF Загрузка **D←FFh**

0006 **MOV B,A** 47 Загрузка **B←A**

0007 **IN 12H** DB 12 Считывание порта **12H**

0008 **DCR A** 3D A←A-1

0009 **JC 07H** DA 07 00 Переход, если **C=1**

000C **MOV A,B** 78 Загрузка **A←B**

000D **INR A** 3C A←A+1

000E **MOV B,A** 47 Загрузка **B←A**

000F **OUT 10H** D3 10 Вывод в порт **10h**

0010 **CALL 200H** CD 00 02 Вызов подпрограммы по адресу **0200h**

0013 **JMP 07H** C3 07 00 Переход на адрес **07h**

2000 **DCR C** 0D C←C-1

2001 **JNC 200H** D2 00 02 Переход, если **C=0**

2004 **DCR D** 15 D←D-1

2005 **JNC 204H** D2 04 02 Переход, если **C=0**

2008 **RET** C9 Возврат из подпрограммы

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.

119

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.

4. Контрольно-измерительные материалы для итоговой аттестации по учебной дисциплине

Предметом оценки являются умения и знания. Контроль и оценка осуществляются с использованием следующих форм и методов: практические занятия, устный ответ у доски, тестирование, самостоятельная работа, лабораторные работы.

Оценка освоения дисциплины предусматривает использование накопительной / рейтинговой системы оценивания и проведение дифференцированного зачета; в зависимости от рейтингового балла студент может быть освобожден от написания зачетной работы.

I. ПАСПОРТ

Назначение:

Предназначен для контроля и оценки результатов освоения учебной дисциплины «Микропроцессоры и микропроцессорные системы» по специальности 09.02.02 Компьютерные сети (*базовый уровень*) следующими умениями, знаниями, которые формируют профессиональную компетенцию, и общими компетенциями:

Умения

У1. Создавать и отлаживать программы реального времени.

Знания

31. Особенности программирования микропроцессорных систем реального времени;

32. Методы микропроцессорной реализации типовых функций управления;

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

II. ЗАДАНИЯ ДЛЯ ДИФФЕРЕНЦИАЛЬНОГО ЗАЧЕТА

1. Микропроцессор и микропроцессорный комплект. Основные понятия и классификация
2. Основные характеристики и структуры МП K580BM80
3. Способы адресации через регистр R7 для Э-60
4. Структура типовой микропроцессорной системы
5. Арифметические команды МП K580BM80
6. Схема и работа МПС средней сложности на основе МП K1810BM86
7. Шины микропроцессорной системы
8. Команда DAA МП K580BM80
9. Режим ПДП на магистрали Q-BUS
10. Функционирование МПС
11. Флаговый регистр МП K580BM80
12. Прямая и косвенная индексная адресация
13. Команды управления МП K580BM80
14. Работа по прерыванию на магистрали Q-BUS
15. Цикл чтения из памяти
16. Состав магистрали Q-BUS
17. Режимы работы МПС
18. Назначение системного контроллера K1810BF88
19. Цикл записи в память
20. Регистры общего назначения МП K580BM80
21. Прямая и косвенная автодекрементная адресации
22. Структура МП K1810BM86
23. Понятие интерфейса
24. Команды работы со стеком МП K580BM80
25. Работа МПС в режиме ПДП (общий подход)
26. Система команд МП K580BM80
27. Распределение адресного пространства учебной микроЭВМ
28. Классификация прерываний
29. Команда расширенной арифметики Э-60
30. Координация взаимодействия устройств на магистрали
31. Работа МПС в режиме прерываний (общий подход)
32. Однонаправленная и двунаправленная шины
33. Команды работы с подпрограммами МП K580BM80
34. Цикл ВВОД (ЧТЕНИЕ) магистрали Q-BUS
35. Команда RST МП K580BM80
36. Децентрализованная селекция магистрали
37. Цикл ВЫВОД (ЗАПИСЬ) на магистрали Q-BUS
38. Методы адресации МП K580BM80
39. Назначение сигналов на выделенных контактах МП K1810BM86 в максимальном режиме
40. Цикл ВВОД-ПАУЗА-ВЫВОД на магистрали Q-BUS
41. Двухшинная и трехшинная МПС
42. Устройство ввода и индикации МП K580BM80
43. Задачи интерфейса
44. Назначение и структура ГТИ K1810GF84
45. Команды переходов Э-60
46. Централизованная селекция магистрали
47. Арифметические команды Э-60
48. Функциональная клавиатура МП K580BM80
49. Структура МП K1801BM1
50. Структура и работа минимально укомплектованной МПС на основе МП K1810BM86

51. Синхронные и асинхронные процессы и магистрали
52. Способы и адресации МП K1801BM1
53. Регистры МП K580BM80
54. Назначение выводов МП K1810BM86
55. Адресная, стековая и ассоциативная память
56. Команды пересылки Э-60
57. Флаговый регистр и сумматор адреса МП K1810BM86
58. Сегментация памяти МП K1810BM86
59. Логические команды Э-60
60. Работа МПС в режиме прерываний (общий подход)
61. Прямая и косвенная регистровая адресация
62. Структура и работа минимально укомплектованной МПС на основе МП K1810BM86
63. Адресные пространства памяти и устройств ввода-вывода
64. Прямая и косвенная автоинкрементные адресации
65. Формирование физического адреса МП K1810BM86
66. Режимы работы МПС
67. Команды пересылки МП K580BM80
68. Циклы ЧТЕНИЕ и ЗАПИСЬ МП K1810BM86
69. Система команд МП K1801BM1
70. Логические команды МП K580BM80
71. Назначение регистров МП K1810BM86
72. Централизованная селекция магистрали
73. Команды переходов МП K580BM80
74. Схема и работа МПС средней сложности на основе МП K1810BM86
75. Работа по прерыванию на магистрали Q-BUS
76. Функционирование МП K580BM80 по тактам
77. Назначение отдельных узлов МП K1801BM1
78. Классификация интерфейсов по функциональному назначению
79. Синхронизация передачи битов, байтов и массивов слов
80. Способы и адресация через регистр R7 для Э-60
81. Микропроцессор Intel 80186
82. Координация взаимодействия.
83. Организация памяти МП I80286.
84. Регистры МП I80286.
85. Классификация ИС ПЛ.
86. Типы памяти конфигурации ИС ПЛ.
87. Типовые схемотехнические решения ИС ПЛ.
88. Приемы дополнительной обработки сигналов в ИС ПЛ.
89. Базовые свойства ИС ПЛ.
90. ИС FPGA.
91. ИС CPLD.
92. ИС ПЛ комбинированной архитектуры.
93. Структура ИС ПЛ FLEX 10K.
94. Структура логического элемента FLEX 10K.
95. Задачи, решаемые с помощью сигнальных процессоров.
96. Типовые функции, реализуемые DSP.
97. Обобщенная архитектура DSP.
98. Основные операции DSP.
99. Структура DSP 56002.
100. Структура и основные характеристики МК C167.
101. Особенности МК TC1796.
102. Особенности нечеткой логики

III КРИТЕРИИ ОЦЕНКИ

Оценка **«отлично»** выставляется студенту, если он глубоко и прочно усвоил программный материал, правильно отвечает на тестовые задания, безошибочно выполняет практическую работу, тесно увязывает теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний.

Оценка **«хорошо»** выставляется студенту, если он твердо знает материал, грамотно и верно выполняет практическую работу, не допускает существенных неточностей в ответе на тестовые вопросы, правильно применяет теоретические положения при решении практических задач.

Оценка **«удовлетворительно»** выставляется студенту, если он имеет знания основного материала, но не усвоил его деталей, допускает неточности, нарушения при выполнении практической работы, испытывает затруднения при выполнении практических заданий, имеет ошибки в решении тестов

Оценка **«неудовлетворительно»** выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями и ошибками выполняет практические задания.