



Министерство образования Республики Беларусь

Учреждение образования  
«Гомельский государственный технический  
университет имени П. О. Сухого»

Кафедра «Автоматизированный электропривод»

**В. А. Савельев, В. И. Луковников**

**МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА  
В АВТОМАТИЗИРОВАННОМ  
ЭЛЕКТРОПРИВОДЕ**

**ЛАБОРАТОРНЫЙ ПРАКТИКУМ  
по одноименной дисциплине  
для студентов специальности 1-53 01 05  
«Автоматизированные электроприводы»  
дневной и заочной форм обучения**

Гомель 2010

УДК 62-83-52:004.315(075.8)  
ББК 31.291+32.965.7я73  
С12

*Рекомендовано научно-методическим советом  
факультета автоматизированных и информационных систем  
ГГТУ им. П. О. Сухого  
(протокол № 4 от 14.12.2009 г.)*

Рецензент: канд. техн. наук, доц. каф. «Электроснабжение»  
ГГТУ им. П. О. Сухого *Т. В. Алферова*

- Савельев, В. А.**  
С12 Микропроцессорные средства в автоматизированном электроприводе : лаборатор. практикум по одноим. дисциплине для студентов специальности 1-53 01 05 «Автоматизированные электроприводы» днев. и заоч. форм обучения / В. А. Савельев, В. И. Луковников. – Гомель : ГГТУ им. П. О. Сухого, 2010. – 26 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <http://lib.gstu.local>. – Загл. с титул. экрана.

Содержит краткие теоретические сведения, задания к лабораторным работам по разделу «Система команд микропроцессора КР580ВМ80А» дисциплины «Микропроцессорные средства в автоматизированном электроприводе».

Для студентов специальности 1-53 01 05 «Автоматизированные электроприводы» дневной и заочной форм обучения.

УДК 62-83-52:004.315(075.8)  
ББК 31.291+32.965.7я73

© Учреждение образования «Гомельский государственный технический университет имени П. О. Сухого», 2010

## **Лабораторная работа № 1**

### **Знакомство с работой на учебной микро-ЭВМ «Электроника 580»**

#### ***1.1. Цель работы***

1. Изучить порядок подготовки микро-ЭВМ «Электроника 580» к работе и правила техники безопасности при ее эксплуатации.
2. Ознакомиться с назначением, составом и техническими характеристиками учебной микро-ЭВМ «Электроника 580».
3. Изучить органы управления и режимы работы микро-ЭВМ «Электроника 580».
4. Изучить принцип адресного хранения информации и организацию памяти микро-ЭВМ «Электроника 580».
6. Получить начальные навыки работы с микро-ЭВМ «Электроника 580».

#### ***1.2. Указания мер безопасности***

##### **ЗАПРЕЩАЕТСЯ**

1. Включать устройство в сеть, не подключив заземление.
2. Работать с устройством при снятых кожухах.
3. Бросать шнур во избежание поломок вилки.
4. Производить смену предохранителей не отсоединив шнур питания от сетевой розетки.

#### ***1.3. Краткие сведения из теории***

##### ***1.3.1. Назначение и состав микро-ЭВМ «Электроника 580»***

Микро-ЭВМ «Электроника 580» предназначена для создания микропроцессорных систем различного назначения и для отладки их программного обеспечения. Кроме того, она может быть использована для обучения инженерного персонала работе с микропроцессорным комплектом КР580.

В устройстве применены восьмиразрядный микропроцессор (МП) типа КР580ВМ80А и оперативное запоминающее устройство (ОЗУ) на ИС типа КР565РУ2А общей емкостью 2 килобайта (Кбайта) с адресным пространством  $8000_{16} \dots 87FF_{16}$  (нижний индекс «16» означает, что число записано в шестнадцатеричном коде).

Для осуществления диалога с пользователем предусмотрены клавиатура и цифровой дисплей, действие которых обеспечивается программой-монитором (аналог базовой системы ввода-вывода

(BIOS)) объемом 1 Кбайт с адресацией  $0000_{16} \dots 03FF_{16}$ , записанной в постоянное запоминающее устройство (ПЗУ) типа К573РФ22 емкостью 1 Кбайта, либо К573РФ2 емкостью 2 Кбайта.

Монитор позволяет загрузить в ОЗУ программу пользователя, переписывать ее на бытовой магнитофон, считывать с магнитофона в ОЗУ, выполнить программу пользователя в режиме отладки (в пошаговом режиме или с остановкой по заданным условиям), осуществить прогон программы пользователя.

Клавиатура содержит 25 клавиш. Верхний и правый ряды клавиш содержат командные клавиши, функции которых описаны ниже. Оставшиеся 16 клавиш используются для ввода шестнадцатеричных цифр (от «0» до «F»). С помощью клавиши «RST» формируется сигнал сброса. Опрос остальных клавиш производится с помощью ИС интерфейса типа КР580ВВ55А, которая, кроме того, может использоваться для целей ввода-вывода.

Цифровой дисплей выполнен на восьми светодиодных семисегментных индикаторах. Его действие основано на принципе прямого доступа к памяти.

### ***1.3.2. Технические характеристики микро-ЭВМ «Электроника 580»***

Тип центрального процессора – параллельный.

Разрядность параллельно обрабатываемой информации – 8 двоичных разрядов.

Максимальное число внешних устройств – 180 устройств ввода, 180 устройств вывода.

Тактовая частота – 2,0 МГц.

Количество команд – 78. Система команд соответствует системе команд МП КР580ВМ80А.

Виды адресации – непосредственная, прямая, регистровая и косвенно-регистровая.

Ввод и вывод информации осуществляется в гексадецимальном (hexadecimal) коде (шестнадцатеричная система счисления).

Режимы работы: пошаговый, прогон в автоматическом режиме, прогон с остановом по заданному адресу и числу проходов.

Возможность прерывания – до 8 векторов.

Возможность согласования по быстродействию с медленно работающими устройствами.

### **1.3.3. Внешний вид и назначение органов управления микро-ЭВМ «Электроника 580»**

На передней панели устройства расположены:

- 1) клавиатура, включающая 9 командных клавиш и 16 клавиш данных;
- 2) индикатор адреса («АДРЕС») и данных («РЕГИСТР» и «ДАННЫЕ»), содержащий 8 разрядов;
- 3) индикатор («ФЛАГ») состояния признаков переноса («С») и нуля («Z»);
- 4) тумблеры включения-выключения («СЕТЬ») и задания режима работы («ПРОГОН» и «ОТЛАДКА»);
- 5) индикаторы работы с магнитофоном («МФ. ВХОД» и «МФ. ВЫХОД»).

На задней стенке находится гнездо для подключения магнитофона, на передней стенке – разъемы для подключения других внешних устройств.



*Рис. 1. Внешний вид микро-ЭВМ «Электроника 580»*

#### ***1.3.4. Назначение командных клавиш.***

«**RST**» - служит для формирования сигнала сброса (ReStart) устройства. Для начала работы с микро-ЭВМ «Электроника 580» после включения тумблера «сеть» необходимо нажать клавишу «RST», при этом на индикаторе должно появиться сообщение «8200 --??». Здесь и в дальнейшем знаком «-» обозначен пробел, а знаком «?» какая-либо ранее записанная в ОЗУ или случайная информация.

«**ADDR**» - служит для перевода устройства в режим задания адреса (**ADDRess**) ячейки памяти. Для чтения содержимого ячейки памяти с адресом **XXXX** необходимо нажать клавиши:

**ADDR X X X X**

После этого в разрядах 1-4 индикатора отобразится заданный адрес ячейки памяти, а в разрядах 7 и 8 - ее содержимое.

«**NEXT**» - служит для увеличения на единицу адреса индицируемой ячейки памяти или регистра МП. Нажатие клавиши «NEXT» выведет на индикатор информацию об адресе и содержимом следующей ячейки памяти.

«**MEM**» - служит для перевода устройства в режим записи данных в ячейку памяти (**MEMory**). Для записи данных в ячейку памяти с адресом **XXXX** необходимо нажать последовательность клавиш:

**ADDR X X X X MEM**

После этого в четырех левых разрядах дисплея высветится адрес ячейки памяти, в двух правых разрядах - ее содержимое и появится десятичная точка. При отсутствии на дисплее десятичной точки данные в память записаны не будут. Дальнейшее нажатие одной или двух цифровых клавиш изменят содержимое данной ячейки. В двух правых разрядах появятся новые данные. Если в набранных данных имеется ошибка, то ее можно исправить нажатием требуемых цифровых клавиш. Сколько бы клавиш не было нажато, в любом случае, в двух правых разрядах высвечиваются значения двух последних нажатых клавиш.

Для перехода к адресу следующей ячейки необходимо нажать клавишу «NEXT». При этом нет необходимости повторного нажатия клавиши «MEM». Повторное нажатие клавиши «MEM» уменьшает на единицу адрес ячейки памяти.

При попытке ввести данные без предварительного нажатия клавиши «MEM», а так же, если на дисплее установлен адрес области

ПЗУ, либо адрес, физически отсутствующий в ОЗУ, на дисплее высветится надпись «Err». В этом случае для восстановления предыдущего адреса и разрешения ввода данных в память необходимо нажать клавишу «MEM».

«CLR» - (CLear) служит для восстановления начального значения адреса или данных, если после их ввода не нажималось других командных клавиш.

«REG» - служит для отображения содержимого восьмиразрядного регистра (REGister) МП. Для чтения содержимого одного из регистров необходимо нажать следующие клавиши:

### REG X

где X - клавиша с наименованием соответствующего регистра:

A - аккумулятор;

B, C, D, E, H, L - регистры общего назначения;

F - регистр признаков (флаговый);

SP – указатель стека.

После нажатия указанных клавиш в разряде 5 индикатора отобразится имя регистра, а в разрядах 7 и 8 его содержимое. Для записи данных в регистр после нажатия указанных клавиш необходимо нажать одну или две шестнадцатиричных клавиш.

«STEP» - служит для выполнения очередной команды МП.

«BRK» - (BReaK) служит для задания адреса контрольной точки в программе.

Можно выполнять программы пользователя с введением контрольных точек, т.е. адресов на которых необходимо прервать выполнение программы для проверки промежуточных результатов.

«RUN» - служит для запуска программы на выполнение с остановом на введенной контрольной точке или команде останова МП.

#### ***1.3.5. Организация памяти микро-ЭВМ «Электроника 580»***

Во время работы микропроцессора команды и данные необходимо хранить и выбирать по мере необходимости. Для этой цели служат запоминающие устройства (ЗУ). Различают ЗУ двух типов: постоянные запоминающие устройства и оперативные запоминающие устройства.

ПЗУ используются для хранения команд и констант, т.е. такой информации, которая остается неизменной при работе микропроцессорной системы. В англоязычной технической литературе аббревиа-

туре ПЗУ соответствует ROM (Read Only Memory, т.е. память только для чтения).

ПЗУ делятся на три класса:

- масочные ПЗУ, программируемые изготовителем (ROM);
- однократно программируемые (пользователем) ПЗУ (PROM);
- многократно программируемые ПЗУ (EPROM).

ПЗУ микро-ЭВМ «Электроника 580» имеет максимально возможную емкость 8 Кбайт (при наличии дополнительных ИС ПЗУ) и может содержать до восьми ИС типа К573РФ22 с организацией 1024x8 бит, либо до четырех ИС типа К573РФ2 с организацией 2048x8 бит. Указанный тип ИС представляет собой многократно программируемое ПЗУ со стиранием информации ультрафиолетовым облучением.

Как минимум одна ИС К573РФ22 всегда установлена и пространство с адресами с  $0000_{16}$  по  $03FF_{16}$  (1 Кбайт) занято под программу-монитор. Из них 768 ячеек с адресами с  $0000_{16}$  по  $02FF_{16}$  занимает программа-монитор. Следующие 256 ячеек с адресами от  $0300_{16}$  по  $03FF_{16}$  отводятся для дополнительной области монитора, которую пользователь может запрограммировать в случае необходимости расширить функциональные возможности программы-монитора.

Для использования адресного пространства с  $0400_{16}$  по  $0FFF_{16}$  (дополнительные 3 Кбайта) необходимо установить дополнительные ИС. Для использования адресного пространства с  $1000_{16}$  по  $1FFF_{16}$  необходимо установить дополнительную плату ЗУ.

**ОЗУ** используются для хранения данных, изменяющихся в процессе работы системы, например, исходных данных, промежуточных результатов расчетов. Аббревиатуре ОЗУ соответствует англоязычная RAM (Random Access Memory, т.е. память произвольного доступа). Информация может быть записана в ОЗУ и считана из него в любом порядке адресов.

Различают два типа ОЗУ:

- статические (SRAM);
- динамические (DRAM).

Динамические ОЗУ эффективны при построении памяти относительно большого объема. Память же малого объема обычно реализуют на более быстродействующих статических элементах.

ОЗУ микро-ЭВМ «Электроника 580» построено на базе ИС КР565РУ2А. Эта ИС представляет собой статическое ОЗУ и имеет

объем 1 Кбит с организацией 1024x1 бит. Для получения восьмиразрядного слова используют 8 таких ИС.

Общая емкость ОЗУ составляет 2 Кбайта, а при подключении дополнительной платы емкость ОЗУ возрастает до 4 Кбайт. В адресном пространстве микро-ЭВМ «Электроника 580» под встроенное ОЗУ отведены 2048 адресов начиная с  $8000_{16}$  и заканчивая  $87FF_{16}$ , а также 2048 адресов с  $8800_{16}$  по  $8FFF_{16}$  зарезервированы под дополнительную плату ЗУ.

Шина адреса микропроцессора КР580ВМ80А, который применен в микро-ЭВМ «Электроника 580», имеет 16 разрядов. Т.е. при ее помощи можно адресовать до  $2^{16} = 65536$  ячеек памяти (64 Кбайта). Как было отмечено выше, память микро-ЭВМ включает значительно меньшее количество ячеек. Оставшееся адресное пространство находится в распоряжении пользователя и может стать доступными при установке дополнительных ИС памяти и разработке дешифратора вновь занимаемых адресов.

#### **1.4. Задания к лабораторной работе**

##### **Задание № 1.**

В ячейку памяти  $8210_{16}$  записать число  $1A_{16}$ . Просмотреть содержимое двух последующих ячеек и вернуться к исходной ячейке. Записать в таблицу 1.1 порядок нажатия клавиш, информацию на дисплее при нажатии каждой клавиши и комментарий выполняемых действий.

Таблица 1.1

Клавиши	Информация на индикаторе		Комментарий
	1 2 3 4	5 6 7 8	

##### **Задание № 2.**

В один из регистров общего назначения записать число  $A1_{16}$ . Записать в таблицу 1.1 порядок нажатия клавиш, информацию на дисплее при нажатии каждой клавиши и комментарий выполняемых действий.

##### **Задание № 3.**

Ввести в устройство программу сложения двух чисел  $57_{16}$  и  $B5_{16}$ .

Таблица 1.2

Адрес	Код	Мнемокод	Примечание
8200	3E	MVI A	Запись в аккумулятор числа $57_{16}$
8201	57	57	
8202	06	MVI B	Запись в регистр В числа $B5_{16}$
8203	B5	B5	
8204	80	ADDB	Сложение чисел $57_{16}$ и $B5_{16}$
8205	76	HLT	Останов

Выполняя программу в пошаговом режиме, после выполнения каждого шага, записывать в таблицу 1.3 содержимое регистров А и В.

Таблица 1.3

№ шага	Рег. А	Рег. В	Комментарий
0	XX	XX	Случайные данные
1			

### **1.5. Содержание отчета**

В отчете привести наименование и цель работы, а также заполненные в соответствии с заданиями 1-3 таблицы 1.1 и 1.3 с подробными комментариями.

### **1.6. Контрольные вопросы**

1. Объяснить отличие терминов «микро-ЭВМ», «микропроцессор», «центральный процессор».
2. Что такое оперативная память микро-ЭВМ?
3. Чем отличается постоянное запоминающее устройство от оперативного?
4. Назначение программы-монитора?
5. Описать назначение функциональных клавиш микро-ЭВМ «Электроника 580».
6. Каким образом просмотреть содержимое ячейки памяти, регистра, регистровой пары, указателя стека?
7. Каким образом можно осуществлять переход от одной ячейки памяти к другой?
8. Как запустить программу на выполнение в пошаговом режиме?
9. Что представляет собой запись программы в мнемосодах и в машинных кодах?

## Лабораторная работа № 2

### Изучение команд пересылки данных

#### 2.1. Цель работы

1. Ознакомиться со способами адресации микропроцессора.
2. Изучить действие команд пересылки данных.
3. Получить практические навыки составления простейших программ.

#### 2.2. Команды пересылки данных

Непосредственная загрузка регистра	MVI r, данные
Непосредственная загрузка регистровой пары	LXI rp, данные
Непосредственная загрузка ячейки памяти	MVI M, данные
Прямая загрузка аккумулятора	LDA, адрес
Прямая загрузка аккумулятора из порта ввода	IN, адрес
Прямая запись аккумулятора в память	STA, адрес
Прямая запись аккумулятора в порт вывода	OUT, адрес
Прямая загрузка регистровой пары HL	LHLD, адрес
Прямая запись регистровой пары HL в память	SHLD, адрес
Пересылка из регистра в регистр	MOV r1, r2
Обмен данными между регистровыми парами HL и DE	XCHG
Обмен данными между двумя верхними ячейками стека и регистровой парой HL	XTHL
Передача в указатель стека содержимого регистровой пары HL	SPHL
Косвенная загрузка аккумулятора	LDAX rp
Косвенная запись аккумулятора в память	STAX rp
Пересылка из памяти в регистр	MOV r, M
Пересылка из регистра в память	MOV M, r

#### 2.3. Задания к лабораторной работе

##### Задание № 1.

1. Ввести в память микро-ЭВМ следующую программу:

Мнемокод	Операнд
LDA	813F
ADD A <sup>1</sup>	
STA	8140
HLT <sup>2</sup>	

2. Записать в ячейку памяти по адресу 813F<sub>16</sub> любое число, например 12<sub>10</sub>, а по адресу 8140<sub>16</sub> и в аккумулятор - число 00.

3. Выполняя программу в пошаговом режиме, после выполнения каждого шага, записывать в таблицу 2.1 содержимое аккумулятора и ячеек памяти с адресами 813F<sub>16</sub> и 8140<sub>16</sub>.

Таблица 2.1

№ ша-га	Рег. А	813F	8140	Комментарий
0	00	12	00	Исходные данные
1				
2				

4. Прокомментировать действия микро-ЭВМ на каждом шаге.

## Задание № 2.

1. Ввести в память микро-ЭВМ следующую программу:

Мнемокод	Операнд
LXI H	813F
MOV A, M	
ADD A	
INX H <sup>3</sup>	
MOV M, A	
HLT	

2. Записать в ячейку памяти по адресу 813F<sub>16</sub> любое число, например 12<sub>10</sub>, а по адресу 8140<sub>16</sub> и в регистры А, H, L - число 00.

<sup>1</sup> Команда **ADD A** производит сложение содержимого аккумулятора с самим собой.

<sup>2</sup> Команда **HLT** осуществляет останов программы.

<sup>3</sup> Команда **INX H** производит увеличение на 1 содержимого регистровой пары HL.

3. Выполняя программу в пошаговом режиме, после выполнения каждого шага, записывать в таблицу 2.2 содержимое регистров А, Н, L и ячеек памяти с адресами  $813F_{16}$  и  $8140_{16}$ .

4. Прокомментировать действия микро-ЭВМ на каждом шаге.

Таблица 2.2

№ шага	Рег. А	Рег. Н	Рег. L	813F	8140	Комментарий
0	00	00	00	12	00	Исходные данные
1						
2						

#### **2.4. Содержание отчета**

В отчете привести наименование и цель работы, тексты программ задач №1 и №2, заполненные таблицы 2.1 и 2.2 с подробными комментариями.

#### **2.5. Контрольные вопросы**

1. Прокомментируйте выполнение следующих команд пересылки данных

- а) с непосредственной адресацией;
- б) с прямой адресацией;
- в) с регистровой адресацией;
- г) с косвенно-регистровой адресацией.

2. Какие способы адресации использованы в программе задания №2.

3. Придумайте другие примеры использования команд пересылки данных с различными способами адресации.

### **Лабораторная работа № 3**

#### **Изучение арифметических команд (часть 1)**

##### **3.1. Цель работы**

1. Изучить действие арифметических команд сложения и десятичной коррекции.

2. Получить практические навыки составления программ.

### **3.2. Краткие сведения из теории (арифметические команды)**

Сложение с регистром	ADD r
Косвенное сложение с памятью	ADD M
Сложение с непосредственными данными	ADI, данные
Сложение содержимого регистровых пар	DAD rp
Сложение с учетом переноса с регистром	ADC r
Сложение с учетом переноса с памятью косвенное	ADC M
Сложение с учетом переноса с непосредственными данными	ACI, данные
Десятичная коррекция аккумулятора	DAA
Вычитание с регистром	SUB r
Косвенное вычитание с памятью	SUB M
Вычитание с непосредственными данными	SUI, данные
Вычитание с регистром и заемом	SBB r
Косвенное вычитание с памятью и заемом	SBB M
Вычитание с непосредственными данными и заемом	SBI, данные
Положительное приращение регистра	INR r
Положительное приращение памяти	INR M
Отрицательное приращение регистра	DCR r
Отрицательное приращение памяти	DCR M
Положительное приращение регистровой пары	INX rp
Отрицательное приращение регистровой пары	DCX rp

### **3.3. Задания к лабораторной работе**

#### **Задание № 1.**

Произвести сложение двух чисел  $51120_{10}$  и  $12460_{10}$ , используя косвенно-регистровую адресацию.

Для этого необходимо:

а) указанные числа перевести из десятичного формата в шестнадцатеричный;

б) составить программу вычисления в соответствии со следующим алгоритмом:

1. Загрузить младший байт (МБ) первого слагаемого в аккумулятор.

2. Сложить младший байт второго слагаемого с аккумулятором.

3. Результат (младший байт суммы) записать в память.

4. Загрузить старший байт (СБ) первого слагаемого в аккумулятор.

5. Сложить старший байт второго слагаемого с аккумулятором с учетом бита переноса.
6. Результат (старший байт суммы) записать в память.
7. Останов.

Текст программы и исходные данные желательно расположить в адресном пространстве ОЗУ в соответствии с таблицей 3.1.

Таблица 3.1

Адрес	Содержимое
8000	Текст программы
...	
81FF	
8200	МБ 1-го слагаемого
8201	СБ 1-го слагаемого
8202	МБ 2-го слагаемого
8203	СБ 2-го слагаемого
8204	МБ суммы
8205	СБ суммы

в) ввести исходные данные и выполнить программу в пошаговом режиме;

г) после выполнения каждого шага программы заносить в таблицу 3.2 содержимое регистров А, Н, L и ячеек памяти с адресами  $8204_{16}$  и  $8205_{16}$ ;

Таблица 3.2

№ шага	Рег. А	Рег. Н	Рег. L	8204	8205	Комментарий
0	00	00	00	00	00	Исходные данные
1						
2						

д) перевести результат из шестнадцатеричного формата в десятичный.

### Задание № 2.

Произвести сложение двух чисел  $89_{10}$  и  $76_{10}$ , в двоично-десятичном формате, используя косвенно-регистровую адресацию.

Для этого необходимо:

а) указанные числа перевести из десятичного формата в двоично-десятичный, а затем в шестнадцатеричный;

б) составить программу вычисления в соответствии со следующим алгоритмом:

1. Загрузить младший байт первого слагаемого в аккумулятор.
2. Сложить младший байт второго слагаемого с аккумулятором.
3. Произвести десятичную коррекцию результата.
4. Результат (младший байт суммы) записать в память.
5. Загрузить старший байт первого слагаемого в аккумулятор.
6. Сложить старший байт второго слагаемого с аккумулятором с учетом бита переноса.
7. Произвести десятичную коррекцию результата.
8. Результат (старший байт суммы) записать в память.
9. Останов.

Текст программы и исходные данные желательно расположить в адресном пространстве ОЗУ в соответствии с таблицей 3.3.

Таблица 3.3

Адрес	Содержимое
8000	Текст программы
...	
81FF	
8200	МБ 1-го слагаемого
8201	СБ 1-го слагаемого
8202	МБ 2-го слагаемого
8203	СБ 2-го слагаемого
8204	МБ суммы
8205	СБ суммы

в) ввести исходные данные и выполнить программу в пошаговом режиме;

г) после выполнения каждого шага программы заносить в таблицу 3.4 содержимое регистров А, Н, L и ячеек памяти с адресами 8204 и 8205;

Таблица 3.4

№ ша- га	Рег. А	Рег. Н	Рег. L	8204	8205	Комментарий
0	00	00	00	00	00	Исходные данные
1						
2						

д) перевести результат из шестнадцатеричного формата в десятичный.

### **3.4. Содержание отчета**

В отчете привести наименование и цель работы, тексты программ задач №1 и №2, записанные в мнемосодах и машинных кодах, заполненные таблицы 3.2 и 3.4 с подробными комментариями.

### **3.5. Контрольные вопросы**

1. Прокомментируйте выполнение следующих команд сложения:

- а) с непосредственной адресацией;
- б) с регистровой адресацией;
- в) с косвенно-регистровой адресацией.

2. Прокомментируйте выполнение следующих команд сложения с переносом:

- а) с непосредственной адресацией;
- б) с регистровой адресацией;
- в) с косвенно-регистровой адресацией.

3. Прокомментируйте выполнение команды десятичной коррекции аккумулятора.

4. Запишите тексты программ заданий №1 и №2 с использованием различных способов адресации.

5. Как изменятся значения флагов регистра состояния и содержимое аккумулятора в результате выполнения операций:

- а) сложения;
- б) сложения с переносом;
- в) десятичной коррекции.

6. Придумайте другие примеры использования команд сложения, сложения с переносом, десятичной коррекции.

## Лабораторная работа № 4 Изучение арифметических команд (часть 2)

### 4.1. Цель работы

1. Изучить действие арифметических команд вычитания иращения.
2. Получить практические навыки составления программ.

### 4.2. Задания к лабораторной работе

#### Задание № 1.

Произвести вычитание двух чисел  $30\ 812_{10}$  и  $12\ 460_{10}$ , используя косвенно-регистровую адресацию.

Для этого необходимо:

- а) указанные числа перевести из десятичного формата в шестнадцатеричный;
- б) составить программу<sup>4</sup> вычисления в соответствии со следующим алгоритмом:
  1. Загрузить младший байт (МБ) уменьшаемого в аккумулятор.
  2. Вычесть младший байт вычитаемого из аккумулятора.
  3. Результат (младший байт разности) записать в память.

Таблица 4.1

Адрес	Содержимое
8000	Текст программы
...	
81FF	
8200	МБ уменьшаемого
8201	СБ уменьшаемого
8202	МБ вычитаемого
8203	СБ вычитаемого
8204	МБ разности
8205	СБ разности

<sup>4</sup> Текст программы и исходные данные желательно расположить в адресном пространстве ОЗУ в соответствии с таблицей 4.1.

4. Загрузить старший байт (СБ) уменьшаемого в аккумулятор.

5. Вычесть старший байт вычитаемого из аккумулятора с учетом бита переноса.

6. Результат (старший байт разности) записать в память.

7. Останов.

в) ввести исходные данные и выполнить программу в пошаговом режиме;

г) после выполнения каждого шага программы заносить в таблицу 4.2 содержимое регистров А, Н, L и ячеек памяти с адресами 8204 и 8205;

д) перевести результат из шестнадцатеричного формата в десятичный.

Таблица 4.2

№ шага	Рег. А	Рег. Н	Рег. L	8204	8205	Комментарий
0	00	00	00	00	00	Исходные данные
1						
2						

## Задание № 2.

Произвести умножение двух чисел  $12_{10}$  и  $3_{10}$ , используя косвенно-регистровую адресацию.

Для этого необходимо:

а) указанные числа перевести из десятичного формата в шестнадцатеричный;

б) составить программу<sup>5</sup> вычисления в соответствии со следующим алгоритмом:

1. Загрузить первый сомножитель в регистр В.

2. Загрузить второй сомножитель в регистр С.

3. Загрузить содержимое регистра В в аккумулятор.

4. Уменьшить на 1 содержимое регистра С.

<sup>5</sup> Текст программы и исходные данные желательно расположить в адресном пространстве ОЗУ в соответствии с таблицей 4.3.

Таблица 4.3

Адрес	Содержимое
8000	Текст программы
...	
81FF	
8200	1-й сомножитель
8201	2-й сомножитель
8202	Произведение

5. Сложить содержимое аккумулятора с содержимым регистра В;

6. Уменьшить на 1 содержимое регистра С.

7. Если содержимое регистра С не равно 0, перейти к п.5<sup>6</sup>;

8. Результат (произведение) записать в память.

9. Останов.

в) ввести исходные данные и выполнить программу в пошаговом режиме;

г) после выполнения каждого шага программы заносить в таблицу 4.4 содержимое регистров А, В, С, Н, L, ячейки памяти с адресом 8202 и состояние флага нулевого результата;

Таблица 4.4

№ шага	Регистры					8202	Флаг нуля	Комментарий
	А	В	С	Н	L			
0	00	00	00	00	00	00	-	Исходные данные
1								
2								

д) перевести результат из шестнадцатеричного формата в десятичный.

### 4.3. Содержание отчета

В отчете привести наименование и цель работы, тексты программ задач №1 и №2, записанные в мнемосодах и машинных кодах, заполненные таблицы 4.2 и 4.4 с подробными комментариями.

### 4.4. Контрольные вопросы

<sup>6</sup> Команда **JNZ**, адрес осуществляет переход к выполнению команды, записанной по указанному адресу, при отсутствии нулевого результата (флаг нуля не установлен)

1. Прокомментируйте выполнение следующих команд вычитания:
  - а) с непосредственной адресацией;
  - б) с регистровой адресацией;
  - в) с косвенно-регистровой адресацией.
2. Прокомментируйте выполнение следующих команд вычитания с заемом:
  - а) с непосредственной адресацией;
  - б) с регистровой адресацией;
  - в) с косвенно-регистровой адресацией.
3. Прокомментируйте выполнение команд положительного и отрицательного приращения
  - а) регистра;
  - б) ячейки памяти;
  - в) регистровой пары.
4. Запишите тексты программ заданий №1 и №2 с использованием различных способов адресации.
5. Как изменятся значения флагов регистра состояния и содержимое аккумулятора в результате выполнения операций:
  - а) вычитания;
  - б) вычитания с заемом;
  - в) положительного и отрицательного приращения.
6. Придумайте другие примеры использования команд сложения, сложения с переносом, десятичной коррекции.

## **Лабораторная работа № 5**

### **Изучение логических команд**

#### **5.1. Цель работы**

1. Изучить действие логических команд.
2. Получить практические навыки составления программ.

#### **5.2. Краткие сведения из теории (логические команды)**

И над регистром и аккумулятором	ANA r
И над косвенно адресуемой памятью и аккумулятором	ANA M
И над непосредственными данными и аккумулятором	ANI, данные
ИЛИ над регистром и аккумулятором	ORA r
ИЛИ над косвенно адресуемой памятью и аккумулятором	ORA M
ИЛИ над непосредственными данными и аккумулятором	ORI, данные

Исключающее ИЛИ над регистром и аккумулятором	XRA r
Исключающее ИЛИ над косвенно адресуемой памятью и аккумулятором	XRA M
Исключающее ИЛИ над непосредственными данными и аккумулятором	XRI, данные
Инверсия аккумулятора	CMA
Сравнение регистра с аккумулятором	CMP r
Сравнение косвенно адресуемой памяти с аккумулятором	CMP M
Сравнение непосредственных данных с аккумулятором	CPI, данные
Циклический сдвиг аккумулятора влево	RLC
Циклический сдвиг аккумулятора вправо	RRC
Циклический сдвиг аккумулятора влево через бит переноса	RAL
Циклический сдвиг аккумулятора вправо через бит переноса	RAR

### **5.3. Задания к лабораторной работе**

#### **Задание № 1.**

В заданном массиве (10x1) среди чисел, содержащих во втором и пятом разрядах двоичного кода единицу, найти наибольшее.

Для этого необходимо:

а) составить программу<sup>7</sup> обработки массива в соответствии со следующим алгоритмом:

1. Загрузить в один из регистров общего назначения (РОН) количество элементов массива.
2. Загрузить в регистровую пару HL начальный адрес массива.
3. Обнулить регистр результата.
4. Сравнить текущий элемент массива с «маской».
5. Если второй и пятый разряды двоичного кода текущего элемента не содержат единицы, перейти к п.9.
6. Сравнить текущий элемент с числом, находящемся в регистре результата.
7. Если текущий элемент меньше содержимого регистра результата, перейти к п.9.

<sup>7</sup> Текст программы и исходные данные желательно расположить в адресном пространстве ОЗУ в соответствии с таблицей 5.1.

8. Записать текущий элемент в регистр результата.
9. Увеличить на 1 текущий адрес массива.
10. Уменьшить на 1 число элементов массива.
11. Если число элементов массива не равно 0, то перейти к п.4.
12. Останов.

Таблица 5.1

Адрес	Содержимое
8000... 81FF	Текст программы
8200	Массив
...	
820A	

б) ввести исходный массив (при вводе массива предусмотреть не менее трех различных элементов, содержащих единицу во втором и пятом разрядах двоичного кода) и выполнить программу в пошаговом режиме;

в) после выполнения каждого шага программы заносить в таблицу 5.2 содержимое регистров А, В, С, Н, L, а также состояние флагов нулевого результата и переноса;

Таблица 5.2

№ шага	Регистры					Флаг нуля	Флаг переноса	Комментарий
	А	В	С	Н	L			
0	00	00	00	00	00	X	X	Исходные данные
1								
2								

### Задание № 2.

Произвести вычитание двух чисел методом сложения с дополнительным кодом (см. лабораторную работу №4).

Для этого необходимо:

а) составить программу<sup>8</sup> в соответствии со следующим алгоритмом:

<sup>8</sup> Текст программы и исходные данные желательно расположить в адресном пространстве ОЗУ в соответствии с таблицей 5.3.

1. Загрузить вычитаемое в аккумулятор.
2. Инвертировать содержимое аккумулятора.
3. Увеличить полученное число на единицу.
4. К полученному дополнительному коду вычитаемого прибавить уменьшаемое.
5. Сохранить полученный результат как разность.
6. Сдвинуть полученный результат влево через флаг переноса.
7. Если содержимое флага переноса равно нулю, перейти к п.10.
8. Проинвертировать результат полученный в п.4 с последующим прибавлением к нему единицы младшего разряда.
9. Сохранить полученный результат как разность.
10. Останов.

в) ввести исходные данные и выполнить программу в пошаговом режиме;

г) после выполнения каждого шага программы заносить в таблицу 5.4 содержимое регистров А и Е и состояние флага переноса;

Таблица 5.3

Адрес	Содержимое
8000	Текст программы
...	
81FF	
Рег. В	Уменьшаемое
Рег. С	Вычитаемое
Рег. Е	Разность

Таблица 5.4

№ шага	Регистры		Флаг переноса	Комментарий
	А	Е		
0	00	00	-	<i>Исходные данные</i>
1				
2				

#### **5.4. Содержание отчета**

В отчете привести наименование и цель работы, текст программы задач №1 и №2, заполненные таблицы 5.2 и 5.4 с подробными комментариями.

#### **5.5. Контрольные вопросы**

1. Прокомментируйте выполнение команд логического сложения с различными способами адресации.
2. Прокомментируйте выполнение команд логического умножения с различными способами адресации.
3. Прокомментируйте выполнение команд «исключающее ИЛИ» с различными способами адресации.
4. Что представляет собой маскирование данных? Как выполнить маскирование данных с использованием команд:
  - а) логического сложения;
  - б) логического умножения;
  - в) исключяющее ИЛИ.
5. Как сравнить два числа, используя следующие команды:
  - а) исключяющее ИЛИ;
  - б) вычитание;
  - в) сравнение.
6. Прокомментируйте выполнение команд циклического сдвига влево и вправо:
  - а) без учета флага переноса;
  - б) с учетом флага переноса.
7. Приведите примеры использования логических команд.

### **ЛИТЕРАТУРА**

1. Гилмор Ч. Введение в микропроцессорную технику: Пер. с англ. – М.: Мир, 1984. – 334с., ил.
2. Микропроцессоры: В 3 кн. Кн.3: Средства отладки: Лаб. практикум и задачник: Учеб. для техн. вузов / Н.В. Воробьев, В.Л. Горбунов, А.В. Горячев и др.; Под ред. Л.Н. Преснухина. – Мн.: Выш. шк., 1987. – 287с.: ил.
3. МикроЭВМ: В 8 кн.: Практ. пособие / Под ред. Л.Н. Преснухина. Кн. 3. Семейство ЭВМ «Электроника К1» / А.В. Кобылинский, А.В. Горячев, Н.Г. Сабадаш, В.В. Проценко. – М.: Высш. шк., 1988. – 191с.: ил.

## СОДЕРЖАНИЕ

Лабораторная работа № 1. Знакомство с работой на учебной микро-ЭВМ «Электроника 580».....	3
Лабораторная работа № 2. Изучение команд пересылки данных.....	11
Лабораторная работа № 3. Изучение арифметических команд (часть 1).....	13
Лабораторная работа № 4. Изучение арифметических команд (часть 2).....	18
Лабораторная работа № 5. Изучение логических команд.....	21
Литература.....	25

**Савельев Вадим Алексеевич  
Луковников Вадим Иванович**

**МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА  
В АВТОМАТИЗИРОВАННОМ  
ЭЛЕКТРОПРИВОДЕ**

**Лабораторный практикум  
по одноименной дисциплине  
для студентов специальности 1-53 01 05  
«Автоматизированные электроприводы»  
дневной и заочной форм обучения**

Подписано в печать 24.06.20109.

Формат 60x84/16. Бумага офсетная. Гарнитура «Таймс».

Цифровая печать. Усл. печ. л. 1,63. Уч.-изд. л. 1,4.

Изд. № 1.

E-mail: [ic@gstu.by](mailto:ic@gstu.by)

<http://www.gstu.by>

Отпечатано на цифровом дуплекаторе  
с макета оригинала авторского для внутреннего использования.

Учреждения образования «Гомельский государственный  
технический университет имени П. О. Сухого».

246746, г. Гомель, пр. Октября, 48.