

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ РФ

КЕМЕРОВСКИЙ ТЕХНОЛОГИЧЕСКИЙ ИНСТИТУТ
ПИЩЕВОЙ ПРОМЫШЛЕННОСТИ

Кафедра АПП и АСУ

ЛАБОРАТОРНЫЙ ПРАКТИКУМ

Методические указания
для студентов, обучающихся по специальности
220301 «Автоматизация технологических процессов и
производств»,
всех форм обучения

Кемерово 2008

Составители:

Р.В. Котляров, ассистент;
Д.Л. Поздняков, доцент, канд. техн. наук,

*Рассмотрено и утверждено на заседании кафедры АПП и АСУ
Протокол № 6 от 29.01.08*

*Рекомендовано методической комиссией
механического факультета
Протокол № 2 от 10.12.08*

Представлены методические указания для выполнения лабораторных работ по дисциплине «Микропроцессорные средства автоматизации», рекомендуемая литература.

ПРЕДИСЛОВИЕ

Микропроцессорный комплект БИС серии КР580 предназначен для создания широко класса средств вычислительной техники и обработки информации. На основе комплекта строятся микроЭВМ контрольно-измерительных систем, микроЭВМ для управления технологическими процессами, контроллеры периферийных устройств, бытовых приборов, игровых автоматов и т.д.

При работе с микропроцессором наиболее распространенным является программирование на языке ассемблера, в котором используется сокращенное написание английских слов (мнемоническое обозначение) для наименования каждой команды. Язык ассемблера относят к машинно-зависимым языкам программирования. Он позволяет получать высококачественные по быстродействию и используемой памяти программы, однако труден в освоении.

Каждый микропроцессор имеет свой набор команд. Поэтому, приступая к изучению какого-либо конкретного микропроцессора, необходимо познакомиться с его системой команд и принятыми в ней сокращениями. Система команд микропроцессора КР580ИК80А содержит 244 команды, 78 из которых являются основными. По функциональному признаку команды данного микропроцессора делятся на следующие группы: команды передачи данных (из регистра в регистр или память и из памяти в регистр); арифметические команды (сложение, вычитание, инкремент, декремент); логические команды (И, ИЛИ, исключающее ИЛИ, сравнение, сдвиг, инвертирование); команды передачи управления и обработки подпрограмм; команды ввода/вывода и управления состоянием процессора.

Лабораторный практикум направлен на освоение системы команд микропроцессорного комплекта БИС серии КР580, на развитие навыков программирования на языке ассемблера данного микропроцессорного комплекта.

Лабораторная работа №1

ИССЛЕДОВАНИЕ СИСТЕМНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ «МОНИТОР» УЧЕБНОГО МИКРОПРОЦЕССОРНОГО КОМПЛЕКТА (УМК)

Цель работы: приобретение практических навыков в работе с микро-ЭВМ, построенной на микропроцессорном комплекте КР580, с системным программным обеспечением «Монитор».

Теоретические сведения

Описание учебного микропроцессорного комплекта на базе набора БИС серии КР580

Учебный микропроцессорный комплект (УМК) представляет собой законченную микро-ЭВМ и предназначен для:

- изучения основ проектирования и обслуживания микро-ЭВМ на базе микропроцессора К580ИК80А;
- обучения основам программирования микропроцессора К580ИК80А;
- разработки и изготовления макетов блоков управления технологическим оборудованием.

Технические характеристики УМК:

- Тип микропроцессора – К580ИК80А.
- Объем ОЗУ – 1 кбайт.
- Объем ПЗУ – 2 кбайт.
- Возможность прерываний – 1 вектор.
- Программное обеспечение – системная программа «Монитор».

Память в УМК распределена следующим образом:

- по адресам $0000_{16} \div 07FF_{16}$ включительно расположены 2 кбайт памяти ПЗУ. Из них первый 1 кбайт памяти ПЗУ занимает программа «Монитор», а второй 1 кбайт памяти зарезервирован за пользователем;
- по адресам $0800_{16} \div 0BFF_{16}$ включительно расположен 1 кбайт памяти ОЗУ пользователя. Причем надо иметь в виду, что

при своей работе программа «Монитор» использует последние 54 ячейки ОЗУ для записи оперативной информации. Поэтому реально за пользователем для написания и отладки программ остается объем памяти с ячейки 0800 до ячейки 0BC9 включительно.

Органы управления и контроля

Органы управления и контроля УМК показаны на рис.1.

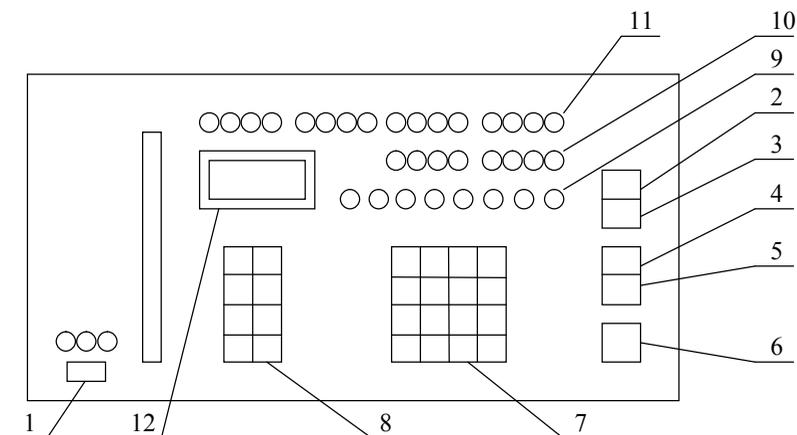


Рис.1. Органы управления и контроля УМК

1. Сеть (~) – включение питания УМК.
2. СБ – сброс системы и возврат к монитору в любой момент времени. При нажатии кнопки СБ в левой позиции дисплея должен появиться знак (-), показывающий готовность УМК к работе.
3. ПР – прерывание выполнения программы. При этом состояние всех регистров сохраняется в ОЗУ, что позволяет продолжить выполнение программы с точки останова.
4. РБ-ШГ – выбор режима работы (автоматический или шаговый). При нажатии кнопки УМК переходит в пошаговый режим работы.

5. КМ-ЦК – выбор величины шага. Если кнопка не нажата, то за шаг выполняется одна команда, если нажата – один цикл.

6. ШГ – выполнение очередного шага при работе в пошаговом режиме.

7. 0-9, A-F – кнопки ввода данных в шестнадцатеричном коде.

8. Директивные клавиши:

- П – чтение и изменение содержимого памяти;
- РГ – чтение и изменение регистров МП;
- СТ – передача управления программе пользователя;
- КС – определение контрольной суммы массива памяти;
- ЗК – заполнение массива памяти константой;
- ПМ – перемещение массива памяти в адресном пространстве;
- Пробел «_» – для разделения нескольких переменных при вводе;
- ВП – конец директивы.

Замечание: при неправильной работе с клавиатурой в крайней правой позиции дисплея индицируется знак «?».

9. Светодиоды – индикация регистра состояний.

10. Светодиоды – индикация шины данных.

11. Светодиоды – индикация шины адреса.

12. Дисплей.

Директивы программы «Монитор» УМК

Программа «Монитор» УМК представляет собой диалоговую систему, предназначенную для реализаций следующих функций:

- чтение и модификация содержимого памяти;
- чтение и модификация содержимого регистров МП;
- выполнение программ пользователя;
- копирование областей памяти;
- вычисление контрольной суммы;
- заполнение массива памяти константой.

Для выполнения той или иной функции «Монитора» оператор должен ввести с клавиатуры встроенного дисплея соответствующую директиву и необходимые параметры.

Результат выполнения команды отображается на дисплее. При этом в начале необходимо произвести запуск программы «Монитор», для этого необходимо включить питание УМК и нажать кнопку «Сброс».

После этого на дисплее будет выведен символ «---», и программа переходит в режим приема директив.

Чтение и модификация содержимого памяти

Для чтения содержимого памяти необходимо нажать последовательно клавиши:

«П»XXXX«ВП»,

где XXXX – шестнадцатеричный адрес ячейки памяти, например 0B00₁₆.

После подачи данной команды в информационной части дисплея высвечивается содержимое ячейки памяти.

При последующем нажатии клавиши «ВП» на дисплей будет выводиться адрес и содержимое следующих по порядку ячеек памяти, например, 0B01₁₆, 0B02₁₆ и т.д.

Для изменения (модификации) содержимого ячеек памяти необходимо последовательно нажать следующие клавиши:

«П»XXXX«_»Д1«_»Д2...«_»ДN«ВП»,

где Д1, Д2, ..., ДN – данные (шестнадцатеричное двухбайтовое число), подлежащее записи в память, например 01.

Т.е. в этом случае данные Д1 записываются по адресу XXXX, Д2 по адресу на единицу больше и т.д. вплоть до подачи команды «ВП», которая прекращает запись данных в память.

Чтение и модификация содержимого регистров МП

Для чтения содержимого регистров необходимо нажать клавишу «РГ», а затем идентификатор регистра, которыми являются символы, определяющие регистры МП.

А – регистр А (8 бит).

В – регистр В (8 бит).

С – регистр С (8 бит).

Д – регистр D (8 бит).

Е – регистр Е (8 бит).

Н – регистр Н (8 бит).

L – регистр L (8 бит).

F – регистр признаков F (8 бит).

SL – младший байт указателя стека (8 бит).

SH – старший байт указателя стека (8 бит).

PL – младший байт регистра команд (8 бит).

PH – старший байт регистра команд (8 бит).

При этом на дисплее высвечивается содержимое указанного регистра в виде шестнадцатеричных цифр.

Например, последовательно нажав клавиши «РГ» А, имеем на дисплее содержимое аккумулятора процессора А.

Для изменения содержимого регистра необходимо после вывода содержимого регистра на дисплей с помощью клавиш набрать новое значение регистра (шестнадцатеричное однобайтовое число) и нажать клавишу «_». В качестве нового значения содержимого регистра при этом фиксируются две последние введенные цифры.

После нажатия клавиши «_» можно вводить идентификатор следующего регистра.

При необходимости перехода к следующему регистру без изменения содержимого первого необходимо, не набирая новых данных, нажать клавишу «_».

Для завершения директивы нажимается клавиша «ВП».

Выполнение программы пользователя

Передача управления программе пользователя осуществляется при нажатии следующих кнопок:

«СТ»АДРЕС1«_»АДРЕС2«ВП»»,

где АДРЕС1, АДРЕС2 – начальный и конечный адрес программы.

АДРЕС1 и АДРЕС2 должны указывать первый байт команды. Если задан только АДРЕС1, то программа выполняется с указанного адреса без возможности остановки в какой-либо точке. Если АДРЕС1 опущен (вместо него вводится «_»), то выполнение программы начинается с текущего значения счетчика команд и будет прервано в точке останова (АДРЕС 2). В момент останова на дисплей выводится адрес точки останова и запоминается значение всех регистров, которые можно прочесть, используя кнопку «РГ».

Определение контрольной суммы массива памяти

Для определения контрольной суммы массива памяти последовательно нажимают следующие клавиши:

«КС»АДРЕС1«_»АДРЕС2«ВП»»,

где АДРЕС1 и АДРЕС2 соответственно начальный и конечный адрес массива памяти.

Контрольная сумма массива представляет собой сумму содержимого всех ячеек массива по модулю 256 без учета переполнения.

После выполнения директивы на экране дисплея индицируется контрольная сумма массива.

Заполнение массива памяти константой

Для заполнения массива памяти константой необходимо последовательно нажать следующие клавиши:

«ЗК»АДРЕС1«_»АДРЕС2«_»«Д»«ВП»»,

где АДРЕС1 и АДРЕС2 соответственно начальный и конечный адреса массива памяти.

Д – байт данных, подлежащий занесению в память.

Подпрограмма директивы заполняет массив памяти данными Д начиная с адреса АДРЕС1 по АДРЕС2 включительно.

Попытка заполнить информацией последние 54 ячейки ОЗУ приводит к разрушению стека монитора.

Перемещение массива памяти в адресном пространстве

Для перемещения массива памяти в адресном пространстве необходимо последовательно нажать следующие клавиши:

«ПМ»АДРЕС1«_»АДРЕС2«_»АДРЕС3«ВП»,

где АДРЕС1 и АДРЕС2 соответственно начальный и конечный адрес перемещения массива. АДРЕС3 – начальный адрес размещения.

По этой директиве массив памяти, ограниченный адресами АДРЕС1 и АДРЕС2 включительно, переписывается в область памяти, начиная с адреса АДРЕС3.

Массивы перемещения и назначения не должны перекрываться. В противном случае происходит потеря информации.

Прерывание выполнения программы пользователя

Для прерывания выполнения программы пользователя необходимо нажать клавишу «ПР». При этом управление передается подпрограмме обработки прерывания. Подпрограмма сохраняет состояние всех регистров процессора и производит передачу управления монитору.

На дисплее индицируется содержимое счетчика команд, которое на единицу больше адреса последнего байта последней выполненной команды.

После этого пользователь может вызвать выполнение любой из существующих директив. Выполнение прерванной программы возможно, начиная с адреса останова или другого любого адреса.

Пошаговое выполнение программы

Имеются две разновидности пошагового выполнения программы: поцикловый режим и покомандный режим. В поцикловом режиме процессор переводится в состояние «ожидание» при выполнении каждого рабочего цикла, а в покомандном режиме лишь при чтении первого байта команды.

Для вызова пошагового режима: установить переключатель «РБ/ШГ» в состояние «ШГ». При этом происходит подключение световой индикации. Далее переключателем «КМ/ЦК» выбрать один из режимов работы и передать управление выполняемой программе.

После этого на табло индикации отобразиться начальный адрес программы, данные по этому адресу и содержимое регистра состояния. Для выхода из этого режима выполните одно из следующих действий: нажмите «СБР» или установите переключатель «РБ/ШГ» в состояние «РБ» и нажмите кнопку «ШГ».

Оборудование и приборы

Исследование системного программного обеспечения «Монитор» осуществляется на учебной микро-ЭВМ УМК.

Порядок выполнения работы

1. Ознакомиться с директивами программы «Монитор» УМК согласно разделу 2 данного методического пособия.
2. Осуществить запись числа $1F_{16}$ в ячейку памяти 0800_{16} ОЗУ УМК.
3. Проверить запись числа $1F_{16}$ в ячейке памяти 0800_{16} (осуществить индикацию этого числа из памяти).
4. Записать число $1F_{16}$ во все регистры процессора А, В, С, D, E, H, L и проверить запись данного числа в регистры.
5. Записать число $1F_{16}$ в массив памяти, начиная с адреса 0800_{16} по адрес $08FF_{16}$ и проверить запись данного числа по всем адресам.

6. Переместить массив памяти по адресам $0800_{16} \div 08FF_{16}$ в область памяти, начиная с адреса $0A00_{16}$. Проверить выполнение данной директивы УМК.

7. Записать в память, отладить и выполнить программу сложения двух двоично-десятичных целых положительных чисел, для чего:

- занести, начиная с ячейки памяти 0850_{16} записанную в виде машинных кодов программу сложения;

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0850	21	00	0A	7E	23	86	00	32	00	09	23	7E	23	8E	00	32
0860	01	09	76													

- занести в ячейку памяти $0A00$ младший байт первого слагаемого;
- занести в ячейку памяти $0A01$ младший байт второго слагаемого;
- занести в ячейку памяти $0A02$ старший байт первого слагаемого;
- занести в ячейку памяти $0A03$ старший байт второго слагаемого;
- выполнить соответствующей директивой монитора записанную в память программу;
- индцировать из памяти по адресам 0900 – младший байт суммы, 0901 – старший байт суммы;
- проверить результат сложения и в случае неправильности результата проверить правильность записи программы, исправить ошибки и повторить сложение.

Меры безопасности

При выполнении лабораторной работы запрещается:

- эксплуатировать УМК при незакрепленной лицевой панели;

- соединять и разъединять разъемы УМК при включенном питании;
- оставлять УМК во включенном состоянии без наблюдения;
- закрывать вентиляционные щели;
- вскрывать лицевую панель УМК.

Контрольные вопросы

1. Функциональное назначение программы «Монитор» в микро-ЭВМ.
2. Функции программы «Монитор» в УМК.
3. Каким образом выполняются директивы «Монитора»:
 - при записи данных в память и их индикации;
 - при проверке и модификации регистров процессора;
 - при выполнении программы пользователя;
 - при копировании области памяти;
 - при заполнении массива памяти константой;
 - при прерывании программы пользователя.
4. БИС МПК серии K580ИК80А.
5. Выполнение команд в МПК серии K580ИК80А.
6. Архитектура МПК серии K580ИК80А.
7. Форматы представления данных в МП системах.

Оформление отчета

Отчет по лабораторной работе должен содержать:

1. краткое описание директив «Монитора» и сведения по их выполнению УМК (выполняется или не выполняется данная директива).
2. запись программы пользователя и сведения по выполнению программы (какие числа заносились в ячейки памяти для сложения и какой результат получен).
3. выводы по работоспособности УМК в указанных режимах. При этом указывается заводской номер УМК, на котором проводилась лабораторная работа.

Лабораторная работа №2

ИССЛЕДОВАНИЕ ВЫПОЛНЕНИЯ КОМАНД АРИФМЕТИЧЕСКИХ И ЛОГИЧЕСКИХ ОПЕРАЦИЙ МИКРОПРОЦЕССОРА КР580ИК80А

Цель работы: закрепление теоретических знаний по изучению команд арифметических и логических операций языка программирования ассемблера микропроцессора КР580ИК80А.

Теоретические сведения

Под управлением команд арифметических и логических операций микропроцессор может выполнять в АЛУ различные арифметические (табл. 1) и логические (табл. 2) операции с числами.

Таблица 1

Список арифметических команд

Мнемоника	Описание	Формат, байт	Число тактов	Флаги S Z AC PC
ADD r ADD M ADI D8	<u>Сложение</u> $A \leftarrow A+r$ $A \leftarrow A+M$ $A \leftarrow A+D8$	1 1 2	4 7 7	++++ ++++ ++++
ADC r ADC M ACI D8	<u>Сложение с переносом</u> $A \leftarrow A+r+C$ $A \leftarrow A+M+C$ $A \leftarrow A+D8+C$	1 1 2	4 7 7	++++ ++++ ++++
SUB r SUB M SUI D8	<u>Вычитание</u> $A \leftarrow A-r$ $A \leftarrow A-M$ $A \leftarrow A-D8$	1 1 2	4 7 7	++++ ++++ ++++

Продолжение табл. 1

Мнемоника	Описание	Формат, байт	Число тактов	Флаги S Z AC PC
	<u>Вычитание с заемом</u>			
SBB r	A←A-r-C	1	4	+++++
SBB M	A←A-M-C	1	7	+++++
SBI D8	A←A-D8-C	2	7	+++++

В списке арифметических команд приняты следующие обозначения: r – содержимое регистров микропроцессора A, B, C, D, E, H, L; M – содержимое ячейки памяти (ОЗУ или ПЗУ), адрес которой указан (предварительно записан) в регистровой паре HL; D8 – байт данных, непосредственно содержащийся в команде; A – содержимое регистра A (аккумулятора микропроцессора); C – бит переноса, образующийся в триггере переноса процессора при сложении предыдущих чисел (младших байтов чисел), либо бит заема при вычитании предыдущих чисел (младших байтов чисел).

В результате выполнения микропроцессором арифметических операций, кроме результата этих операций, которые записываются в аккумулятор A, в регистре признаков F образуется байт признаков PSW, содержащий следующие признаки результатов арифметических операций (рис. 1).

S	Z	0	AC	0	P	1	C
Отрицательный результат (знак)	Нулевой результат		Перенос из 3-го разряда (дополнительный перенос)		Результат содержит четное число «1»		Перенос из 7-го разряда

Рис. 1. Структура регистра признаков

Данные признаки используются в командах условной передачи управления для реализации переходов по условиям при написании и отработке программы.

Таблица 2

Список логических команд

Мнемоника	Описание	Формат, байт	Число такты	Флаги S Z AC PC
	<u>Логическое умножение</u> (операция И)			
ANA r	$A \& r \rightarrow A$	1	4	++0+0
ANA M	$A \& M \rightarrow A$	1	7	++0+0
ANI D8	$A \& D8 \rightarrow A$	2	7	++0+0
	<u>Логическое сложение</u> (операция ИЛИ)			
ORA r	$A+r \rightarrow A$	1	4	++0+0
ORA M	$A+M \rightarrow A$	1	7	++0+0
ORI D8	$A+D8 \rightarrow A$	2	7	++0+0
	<u>Инvertирование</u> <u>аккумулятора</u> (операция НЕ)			
CMA	$A \leftarrow \bar{A}$	1	4	-----
	<u>Инvertирование</u> <u>флага переноса</u>			
CMC	$C \leftarrow \bar{C}$	1	4	---- \bar{C}
	<u>Исключающие ИЛИ</u> (операция ИЛИ-НЕ)			
XRA r	$A \oplus r \rightarrow A$	1	4	++0+0
XRA M	$A \oplus M \rightarrow A$	1	7	++0+0
XRI D8	$A \oplus D8 \rightarrow A$	2	7	++0+0

В списке логических команд приняты аналогичные обозначения. В списке вспомогательных арифметических и логических команд (табл. 3) под гр понимается содержимое регистровых пар BC, DE, HL, SP.

Таблица 3

Список вспомогательных арифметических и логических команд

Мнемоника	Описание	Формат, байт	Число так- тов	Флаги S Z AC PC
INR r INR M INX rp	<u>Инкремент (увеличение на «1»)</u> $r \leftarrow r+1$ $M \leftarrow M+1$ $rp \leftarrow rp+1$	1 1 1	5 10 5	+++++ +++++ -----
DCR r DCR M DCX rp	<u>Декремент (уменьшение на «1»)</u> $r \leftarrow r-1$ $M \leftarrow M-1$ $rp \leftarrow rp-1$	1 1 1	5 10 5	+++++ +++++ -----
DAD rp	<u>Сложение содержимого регистровых пар</u> $HL \leftarrow HL+rp$	1	10	-----
CMP r CMP M CPI D8	<u>Сравнение</u> $A - r \Rightarrow F$ $A - M \Rightarrow F$ $A - D8 \Rightarrow F$	1 1 2	4 7 7	+++++ +++++ +++++
DAA	<u>Десятичная коррекция</u> Преобразование содержимого аккумулятора в двоично-десятичный код.	1	4	+++++
STC	<u>Установка C</u> $1 \rightarrow C$	1	4	---- 1
RLC	<u>Циклический сдвиг содержимого аккумулятора влево</u> Все биты A смещаются на 1 разряд влево. Старший разряд A переходит в его нулевой разряд и регистр признака C.	1	4	---- +

Продолжение табл. 3

RRC	<u>Циклический сдвиг содержимого аккумулятора вправо</u> Все биты А смещаются на 1 разряд вправо. Старший разряд А и С приобретают значение младшего разряда А.	1	4	-----+
RAL	<u>Арифметический сдвиг содержимого аккумулятора влево</u> Все биты А смещаются на 1 разряд влево. Старший разряд А переходит в С, а С в младший разряд А.	1	4	-----+
RAR	<u>Арифметический сдвиг содержимого аккумулятора вправо</u> Выполняется аналогично RAL, но вправо.	1	4	-----+

Оборудование и приборы

Лабораторная работа выполняется на эмуляторе учебного микропроцессорного комплекта (процессор КР580ИК80А). Описание работы в эмуляторе приведено в приложении.

Порядок выполнения работы

1 Исследование команд арифметических операций

Задание 1: выполнить операцию сложения двух однобайтных чисел с применением команд: ADD r, ADD M, ADI D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 04	MVI	A, 04	04→A
0802	06 08	MVI	B, 08	08→B
0804	80	ADD	B	A+B→A
0805	76	HLT		Конец

Просмотреть и зафиксировать в отчете содержимое регистров А (аккумулятора), В, F (регистра состояний).

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 40 08	LXI	H, 0840	08→H, 40→L
0803	3E 04	MVI	A, 04	04→A
0805	36 08	MVI	M, 08	08→M
0807	86	ADD	M	A+M→A
0808	76	HLT		Конец

Просмотреть и зафиксировать в отчете содержимое регистров А, М (виртуальный регистр), F, регистровой пары HL.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 04	MVI	A, 04	04→A
0802	C6 1F	ADI	1F	A+1F→A
0804	76	HLT		Конец

Просмотреть и зафиксировать в отчете содержимое регистров А (аккумулятора), F (регистра признаков).

Задание 2: выполнить операцию вычитания двух однобайтных чисел с применением команд: SUB r, SUB M, SUI D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 08	MVI	A, 08	08→A
0802	0E 06	MVI	C, 06	04→C
0804	91	SUB	C	A-C→A
0805	76	HLT		Конец

Просмотреть и зафиксировать в отчете содержимое регистров А, С, F.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 40 08	LXI	H, 0840	08→H, 40→L
0803	3E 06	MVI	A, 06	06→A
0805	36 08	MVI	M, 08	08→M
0807	96	SUB	M	A-M→A
0808	76	HLT		Конец

Просмотреть и зафиксировать в отчете содержимое регистров А, М, F, регистровой пары HL.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 1F	MVI	A, 1F	1F→A
0802	D6 1F	SUI	1F	A-1F→A
0804	76	HLT		Конец

Просмотреть и зафиксировать в отчете содержимое регистров A, F.

Задание 3: выполнить операцию сложения двух двухбайтных чисел с применением одной из команд: ADC r, ADC M, ACI D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 8A	MVI	A, 8A	8A→A
0802	06 B4	MVI	B, B4	B4→B
0804	80	ADD	B	A+B→A
0805	32 50 08	STA	0850	A→0850
0808	3E 08	MVI	A, 08	08→A
080A	06 10	MVI	B, 10	10→B
080C	88	ADC	B	A+B+C→A
080D	32 51 08	STA	0851	A→0851
0810	76	HLT		Конец

Данная программа выполняет операцию $88A_{16}+10B4_{16}$. Сначала складываются младшие байты чисел, затем старшие с учетом переноса в старший байт. Младший байт суммы содержится в ячейке памяти 0850, старший байт суммы в ячейке памяти 0851. Проверить результат сложения. Проверить и зафиксировать в отчете содержимое регистра F после сложения младших и старших байтов, а также результат сложения. Программу выполнять в режиме трассировки.

Задание 4: выполнить операцию вычитания двух двухбайтных чисел с применением одной из команд: SBB r, SBB M, SBI D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 27	MVI	A, 27	27→A
0802	D6 83	SUI	83	A-83→A
0804	32 50 08	STA	0850	A→0850
0807	3E 21	MVI	A, 21	21→A
0809	DE 10	SBI	10	A-10-C→A
080B	32 51 08	STA	0851	A→0851
080E	76	HLT		Конец

Данная программа выполняет операцию $2127_{16}-1083_{16}$. Сначала из младшего байта уменьшаемого вычитается младший байт вычитаемого, результат помещается в ячейку памяти 0850. Далее с учетом заема в младший байт из старшего байта уменьшаемого вычитается старший байт вычитаемого, результат помещается в ячейку памяти 0851. Проверить результат вычитания. Проверить и зафиксировать в отчете содержимое регистра F после вычитания младших и старших байтов, а также результат вычитания чисел – ячейки памяти 0850 (младший байт разности) и 0851 (старший байт разности). Программу выполнять в режиме трассировки.

2 Исследование команд логических операций

Для выполнения заданий данного пункта необходимо перейти к двоичной форме записи содержания регистров.

Задание 5: выполнить операцию логического умножения чисел 10100011_2 ($A3_{16}$) и 10001011_2 ($8B_{16}$) с помощью любой команды: ANA r, ANA M, ANA D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E A3	MVI	A, A3	$10100011 \rightarrow A$
0802	06 8B	MVI	B, 8B	$10001011 \rightarrow B$
0804	A0	ANA	B	$A \& B \rightarrow A$
0805	76	HLT		Конец

Проверить правильность результата и зафиксировать в отчете содержимое регистров A, F.

Задание 6: с помощью байта-маски 00100001_2 (21_{16}) сбросить 1 и 4 биты байта 00110011_2 (33_{16}) с применением одной из команд: ANA r, ANA M, ANA D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 40 08	LXI	H, 0840	$08 \rightarrow H, 40 \rightarrow L$
0803	3E 33	MVI	A, 33	$00110011 \rightarrow A$
0805	36 21	MVI	M, 21	$00100001 \rightarrow M$
0807	A6	ANA	M	$A \& M \rightarrow A$
0808	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистров А, М, F.

Задание 7: выполнить операцию логического сложения чисел 10100011_2 ($A3_{16}$) и 10001011_2 ($8B_{16}$) с применением одной из команд: ORA r, ORA M, ORI D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 40 08	LXI	H, 0840	08→H, 40→L
0803	3E A3	MVI	A, A3	10100011→A
0805	36 8B	MVI	M, 8B	10001011→M
0807	B6	ORA	M	A+M→A
0808	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистров А, М, F.

Задание 8: с помощью байта-маски 10001000_2 (88_{16}) установить 3 и 7 биты байта и 00110011_2 (33_{16}) с применением одной из команд: ORA r, ORA M, ORI D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 33	MVI	A, 33	00110011→A
0802	0E 88	MVI	C, 88	10001000→C
0804	B1	ORA	C	A&C→A
0805	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистров А, С, F.

Задание 9: с помощью команды CMA инвертировать число 10001010_2 ($8A_{16}$).

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 8A	MVI	A, 8A	10001010→A
0802	2F	CMA		\bar{A} →A
0803	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистров А, F.

Задание 10: инвертировать флаг переноса регистра F.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3F	CMC		$C \leftarrow \bar{C}$
0802	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистра F до и после выполнения программы.

Задание 11: инвертировать 1, 3 и 7 биты байта 10110001_2 (B_{16}) с помощью байта-маски 10001010_2 ($8A_{16}$), используя одну из команд: XRA r, XRA M, XRI D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E B1	MVI	A, B1	$10110001 \rightarrow A$
0802	0E 8A	MVI	C, 8A	$10001010 \rightarrow C$
0804	A9	XRA	C	$A \oplus C \rightarrow A$
0805	76	HLT		Конец

Проверить и зафиксировать содержимое регистров A, C, F.

Задание 12: поразрядно сравнить два числа 10110001_2 и B_{16} , используя одну из команд: XRA r, XRA M, XRI D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E B1	MVI	A, B1	$10110001 \rightarrow A$
0802	EE B1	XRI	B1	$A \oplus C \rightarrow A$
0804	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистров A, F.

3 Исследование команд вспомогательных логических и арифметических операций

Задание 13: сравнить числа $3D_{16}$ и 22_{16} , $3D_{16}$ и $8F_{16}$, $3D_{16}$ и $3D_{16}$, используя одну из команды: CMP r, CMP M, CPI D8.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 3D	MVI	A, 3D	$3D \rightarrow A$
0802	06 22	MVI	B, 22	$22 \rightarrow B$

0804	B8	СМР	В	А-В→F
0805	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистров А, В, F.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 40 08	LXI	Н, 0840	08→Н, 40→L
0803	3E 3D	MVI	А, 3D	3D→В
0805	36 8F	MVI	М, 8F	8F→М
0807	BE	СМР	М	А-М→F
0808	76	HLT		Конец

Проверить и зафиксировать содержимое регистров А, М, F.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 3D	MVI	А, 3D	3D→А
0802	FE 3D	СРІ	3D	А-3D→F
0804	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистров А, F.

Задание 14: задан массив чисел: $0A00 \leftarrow 01_{16}$; $0A01 \leftarrow 05_{16}$; $0A02 \leftarrow 02_{16}$; $0A03 \leftarrow 01_{16}$. Пользуясь командами INX r и DCX r, определить сумму элементов массива. Перед выполнением программы необходимо ввести массив в память.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 01 0A	LXI	Н, 0A01	0A→Н, 01→L
0803	3A 00 0A	LDA	А, 0A00	(0A00)→А
0806	86	ADD	М	А+М→А
0807	23	INX	Н	0A→Н, 02→L
0808	86	ADD	М	А+М→А
0809	23	INX	Н	0A→Н, 03→L
080A	86	ADD	М	А+М→А
080B	76	HLT		Конец

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 02 0A	LXI	Н, 0A02	0A→Н, 02→L
0803	3A 03 0A	LDA	А, 0A03	(0A03)→А
0806	86	ADD	М	А+М→А
0807	2B	DCX	Н	0A→Н, 01→L
0808	86	ADD	М	А+М→А

0809	2В	DCX	Н	0A→H, 00→L
080A	86	ADD	М	A+M→A
080B	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистров А, F.

Задание 15: задан массив чисел: $0A00 \leftarrow 01_{16}$; $0A01 \leftarrow 05_{16}$; $0A02 \leftarrow 02_{16}$; $0A03 \leftarrow 01_{16}$. Пользуясь командами INR М и DCR М уменьшить 1 и 2 элементы массива на единицу, 3 и 4 элементы массива увеличить на единицу. Перед выполнением программы необходимо ввести массив в память МП.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 00 0A	LXI	Н, 0A00	0A→H, 00→L
0803	35	DCR	М	М-1→М
0804	23	INX	Н	0A→H, 01→L
0805	35	DCR	М	М-1→М
0806	23	INX	Н	0A→H, 02→L
0807	34	INR	М	М+1→М
0808	23	INX	Н	0A→H, 03→L
0809	34	INR	М	М+1→М
080A	76	HLT		Конец

Проверить правильность выполнения программы. Записать исходный и модифицированный массивы данных.

Задание 16: определите сумму трех однобайтных чисел, находящихся по адресам 0921 (03_{16}), 0B44 (05_{16}), 0D88 (01_{16}), пользуясь командой сложения содержимого регистровых пар DAD г. Предварительно необходимо записать числа в память МП.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 21 09	LXI	Н, 0921	09→H, 21→L
0803	86	ADD	М	A+M→A
0804	01 23 02	LXI	В, 0223	02→В, 23→С
0807	09	DAD	В	0B→H, 44→L
0808	86	ADD	М	A+M→A
0809	11 44 02	LXI	D, 0244	02→D, 44→E
080C	19	DAD	D	0D→H, 88→L
080D	86	ADD	М	A+M→A
080E	76	HLT		Конец

Проверить правильность выполнения программы. Записать содержимое регистров А, F, M, регистровых пар HL, BC, DE.

Задание 17: найти сумму двух десятичных однобайтных чисел 77 и 18.

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	3E 77	MVI	A, 77	77→A
0802	C6 18	ADI	18	18+A→A
0804	27	DAA		A ₁₆ →A ₁₀
0805	76	HLT		Конец

Проверить и зафиксировать в отчете содержимое регистров А, F.

Контрольные вопросы

1. Перечень, назначение и процесс выполнения команд арифметических операций.
2. Перечень, назначение и процесс выполнения команд логических операций.
3. Перечень, назначение и процесс выполнения вспомогательных арифметических и логических команд.

Оформление отчета

1. Задание.
2. Программы исследования каждой команды.
3. Содержимое регистров и ячеек памяти.
4. Содержимое регистра с признаками результата.
5. Выводы по корректности работы программ.

Лабораторная работа №3

ИССЛЕДОВАНИЕ КОМАНД ПЕРЕСЫЛКИ ДАННЫХ МИКРОПРОЦЕССОРА КР580ИК80А

Цель работы: закрепление теоретических знаний по изучению команд операций пересылки данных языка программирования ассемблера микропроцессора КР580ИК80А.

Теоретические сведения

Под управлением команд пересылки микропроцессор может выполнять обмен между внутренними регистрами процессора, между внутренними регистрами и памятью микро-ЭВМ.

Список команд пересылки микропроцессора КР580ИК80А включает следующие команды (табл. 1).

Таблица 1

Команды пересылки данных

Мнемоника	Описание	Формат, байт	Число тактов	Флаги S Z AC PC
MOV r_1, r_2	Пересылка данных из регистра r_2 в регистр r_1 : $r_1 \leftarrow r_2$	1	5	----
MOV M, r	Пересылка данных из регистра r в регистр M: $M \leftarrow r$	1	7	----
MOV r, M	Пересылка данных из регистра M в регистр r: $r \leftarrow M$	1	7	----
MVI r, Д8	Запись в регистр r одного байта данных: $r \leftarrow Д8$	2	7	----
MVI M, Д8	Запись в регистр M одного байта данных: $M \leftarrow Д8$	2	10	----
LDA A16	В аккумулятор загружается число из адреса A16: $A \leftarrow (A16)$	3	13	----

Продолжение табл. 1

STA A16	Выгрузка содержимого аккумулятора по указанному адресу: (A16) \leftarrow A	3	13	-----
LDAX rp	Загрузка в аккумулятор числа из ячейки памяти с адресом, указанным в регистровой паре (B-C или D-E): A \leftarrow (rp)	1	7	-----
STAX rp	Выгрузка из аккумулятора по адресу в регистровой паре (B-C или D-E): (rp) \leftarrow A	1	7	-----
LXI rp, D16	Непосредственная запись в регистровую пару (B, D, H, SP) двухбайтного числа: (rp) \leftarrow D16	3	10	-----
LHLD A16	Загрузка пары регистров H-L из двух соседних ячеек памяти, начиная с адреса указанного в команде: L \leftarrow (A16); H \leftarrow (A16+1)	3	10	-----
SHLD A16	Обратно предыдущей команде: (A16) \leftarrow L; (A16+1) \leftarrow H	3	16	-----
XCHG	Обмен данными между регистровыми парами H-L и D-E: H \leftrightarrow D, L \leftrightarrow E	1	4	-----
XTHL	Обмен между регистровой парой H-L и двумя верхними ячейками стека, при этом содержимое указателя стека не меняется: L \leftrightarrow (SP); H \leftrightarrow (SP+1)	1	18	-----
SPHL	Занесение содержимого пары регистров H-L в указатель стека: SP \leftarrow HL	1	5	-----
PUSH rp	Занесение в стек содержимого пары регистров (B, D, H). Содержимое указателя стека уменьшается на 2. (SP) \leftarrow мл. г; (SP+1) \leftarrow ст. г.	1	11	-----

Окончание табл. 1

POP rp	Извлечение пары чисел в регистровую пару из стека (обратно предыдущей команде): мл.г←(SP); ст.г←(SP+1); SP=SP+2	1	10	-----
PUSH PSW	Копирование в стек аккумулятора и регистра признаков: (SP) ←F; (SP+1) ←A	1	11	-----
POP PSW	Выдача из стека данных в аккумулятор и регистр признаков F←(SP); A←(SP+1)	1	10	+++++
IN port	Ввод в аккумулятор данных из внешнего устройства, подключенного к указанному в команде порту: A←(port)	2	10	-----
OUT port	Выгрузка 1-го байта из аккумулятора в соответствующий порт: (port)←A	2	10	-----

В списке команд пересылки под r, r₁ и r₂ понимается содержимое регистров процессора A, B, C, D, E, H, L. Под M – содержимое ячейки памяти (ОЗУ или ПЗУ), адрес которой предварительно записан в регистровой паре HL. Под D8 – байт данных, непосредственно содержащийся в команде. Под D16 – двухбайтовое число, непосредственно содержащееся в команде. Под (rp) – содержимое регистровых пар. Под A16 – двухбайтовый адрес, непосредственно записанный в команде. Под (A16) – содержимое ячейки памяти (ОЗУ или ПЗУ), адрес которой указан в команде.

Оборудование и приборы

Лабораторная работа выполняется на эмуляторе учебного микропроцессорного комплекта (процессор КР580ИК80А).

Порядок выполнения работы

1 Исследование команд обмена данными регистров и ячеек памяти

Задание 1: составить программу последовательной записи числа $1F_{16}$ в регистры A, B, C, D, E, H, L, пользуясь командами MOV r_1, r_2 и MVI $r, D8$. Программу выполнить в режиме трассировки, фиксируя в отчете содержимое регистров до и после выполнения каждой команды.

Адрес	Код	Мнемоника	Операнд
0800	3E 1F	MVI	A, 1F
0802	47	MOV	B, A
0803	48	MOV	C, B
0804	51	MOV	D, C
0805	5A	MOV	E, D
0806	63	MOV	H, E
0807	6C	MOV	L, H
0808	76	HLT	

Задание 2: составить программу обмена данными между регистрами A и C с применением регистра M. Программу выполнить в режиме трассировки, фиксируя в отчете содержимое регистров до и после выполнения каждой команды.

Адрес	Код	Мнемоника	Операнд
0800	21 20 08	LXI	H, 0820
0803	3E 05	MVI	A, 05
0805	0E 21	MVI	C, 21
0807	77	MOV	M, A
0808	79	MOV	A, C
0809	4E	MOV	C, M
080A	76	HLT	

Задание 3: записать последовательно элементы массива в регистры E, D, L, H. Исходный массив: 0A00 (05_{16}); 0A01 (02_{16}); 0A02 (07_{16}); 0A03 (01_{16}). Предварительно записать исходный массив в память.

Адрес	Код	Мнемоника	Операнд
0800	2A 00 0A	LHLD	0A00
0803	EB	XCHG	
0804	2A 02 0A	LHLD	0A02
0807	76	HLT	

Задание 4: в регистрах записаны числа: D – 01₁₆, E – 07₁₆, H – 02₁₆, L – 05₁₆. Образовать массив {07, 01, 05, 02} в памяти, начиная с ячейки 0835. Предварительно записать числа в регистры.

Адрес	Код	Мнемоника	Операнд
0800	22 37 08	SHLD	0837
0803	EB	XCHG	
0804	22 35 08	SHLD	0835
0807	76	HLT	

Задание 5: составить программу обмена данными между двумя ячейками памяти 0830 (78₁₆) и 0831 (E3₁₆). Предварительно записать числа в память.

Адрес	Код	Мнемоника	Операнд
0800	3A 30 08	LDA	0830
0803	47	MOV	B, A
0804	3A 31 08	LDA	0831
0807	32 30 08	STA	0830
080A	78	MOV	A, B
080B	32 31 08	STA	0831
080E	76	HLT	

Задание 6: составить программу обмена данными между ячейками памяти 0820 (78₁₆) и регистром D (0A₁₆). Предварительно записать данные в ячейку памяти и регистр. Для промежуточного хранения данных использовать регистр M.

Адрес	Код	Мнемоника	Операнд
0800	21 40 08	LXI	H, 0840
0803	01 20 08	LXI	B, 0820
0806	0A	LDAX	B
0807	77	MOV	M, A
0808	7A	MOV	A, D
0809	02	STAX	B
080A	56	MOV	D, M
080B	76	HLT	

2 Исследование команд обмена данными со стеком

Задание 7: организовать стековую область памяти, начиная с ячейки 0820. Поменять местами содержимое регистровых пар В-С и D-E, используя команды обмена данными со стеком. Предварительно заполнить регистры данными: В – 12_{16} , С – 34_{16} , D – AB_{16} , E – CD_{16} .

Адрес	Код	Мнемоника	Операнд	Комментарий
0800	21 20 08	LXI	H, 0820	08→H, 20→L
0803	F9	SPHL		HL→SP
0804	C5	PUSH	B	B→(SP+1), C→(SP), SP+2→SP
0805	D5	PUSH	D	D→(SP+1), E→(SP), SP+2→SP
0806	C1	POP	B	(SP+1)→B, (SP)→C, SP-2→SP
0807	D1	POP	D	(SP+1)→D, (SP)→E, SP-2→SP
0808	76	HLT		Конец

3 Составление программ управления

Задание 8: составить программу в соответствии с предложенной блок-схемой (рис.1), проверить корректность ее работы.

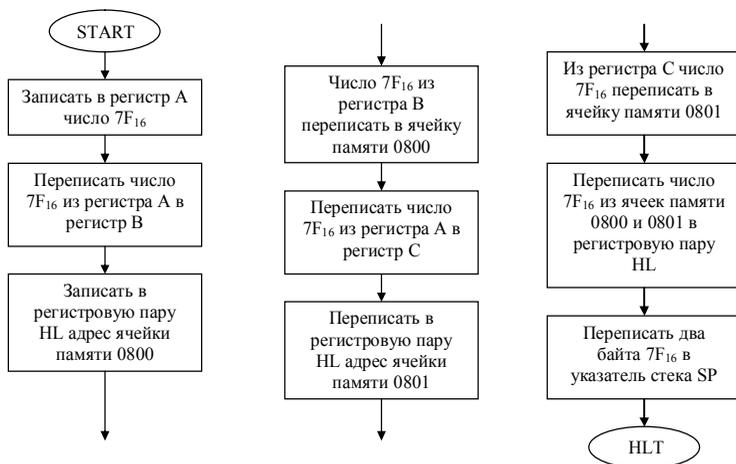


Рис. 1. Блок-схема программы управления

Контрольные вопросы

1. Перечень, назначение и процесс выполнения команд пересылки данных.
2. Понятие стековой области. Перечень, назначение и процесс выполнения команд обмена данными со стеком.
3. Перечень, назначение и процесс выполнения команд обмена данными с внешними устройствами.

Оформление отчета

1. Задание.
2. Программы исследования команд пересылки данных.
3. Содержимое регистров и ячеек памяти.
4. Выводы по корректности работы программ.

Лабораторная работа №4

ИССЛЕДОВАНИЕ КОМАНД УПРАВЛЕНИЯ, ВВОДА-ВЫВОДА И РАБОТЫ СО СТЕКОМ МИКРОПРОЦЕССОРА КР580ИК80А

Цель работы: закрепление теоретических знаний по изучению команд операций управления, ввода-вывода и работы со стеком языка программирования ассемблера микропроцессора КР580ИК80А.

Теоретические сведения

С помощью команд управления, ввода-вывода и работы со стеком микропроцессор может выполнять операции обращения к подпрограммам в программе по различным условиям, безусловные переходы в программе, осуществлять обмен данными с устройствами ввода-вывода через интерфейсные схемы и выполнять операции с использованием стековой памяти.

Список команд управления микропроцессора КР580ИК80А включает следующие команды (табл. 1).

Таблица 1

Команды управления

Мнемоника	Описание	Число тактов
JMP A16	<u>Безусловный переход</u> к команде, адрес которой определяется двумя байтами данной команды: PC←A16.	10
EI	<u>Разрешение прерывания.</u>	4
DI	<u>Запрещение прерывания.</u>	4
HLT	<u>Останов</u> работы программы.	7
NOP	<u>Нет операции</u> (пустая операция).	4

Продолжение табл. 1

CALL A16	<u>Обращение к подпрограмме:</u> перед выполнением этой команды в стековый регистр SP должен быть записан начальный адрес памяти, отведенной под стековую память. В процессе выполнения команды в программе осуществляется переход к команде, адрес которой указан A16. При этом текущее содержимое счетчика команд заносится в стековую память, адрес которой указан в регистре SP: $(SP) \leftarrow \text{мл. PC}$; $(SP+1) \leftarrow \text{ст. PC}$; $SP = SP - 2$; $PC \leftarrow A16$	17
RET	<u>Возврат из подпрограммы:</u> по этой команде, которая записывается в конце каждой подпрограммы, осуществляется переход из подпрограммы к команде, адрес которой записан в верхней паре ячеек стековой области, и происходит увеличение содержимого регистра SP на 2: $(SP) \rightarrow \text{мл. PC}$; $(SP+1) \rightarrow \text{ст. PC}$; $SP = SP + 2$.	10
Jcc ADR	<u>Условные переходы:</u> если условие (cc из табл. 2) истинно, то переход к команде, адрес которой ADR указан в команде, иначе выполняется команда, расположенная вслед за данной.	10
Ccc ADR	<u>Условное обращение к подпрограмме:</u> если условие (cc из табл. 2) истинно, то выполняется команда CALL; иначе выполняется команда, расположенная вслед за данной.	11/17
Rcc	<u>Условный возврат из подпрограммы:</u> если условие (cc из табл. 2) истинно, то выполняется команда RET; иначе выполняется команда, расположенная вслед за данной.	5/11

Условия, по которым выполняются команды МП КР580ИК80А Jcc ADR, Ccc ADR и Rcc, определяются байтом признаков PSW в регистре признаков F, содержащем признаки результатов. Коды условий команд условных переходов приведены в табл. 2.

Таблица 2

Коды условий команд условных переходов

Мнемоника (сс)	Условие	Мнемоника (сс)	Условие
NZ	Не нуль (Z=0)	Z	Нуль (Z=1)
NC	Нет переноса (C=0)	C	Перенос (C=1)
PO	Нечетность (P=0)	PE	Четность (P=1)
P	Плюс (S=0)	M	Минус (S=1)

В качестве примера рассмотрим команду JP 0810. По этой команде осуществляется условный переход в программе к команде, записанной по адресу 0810, в случае, если в результате выполнения предыдущей команды в регистре F S = 0, т.е. получено положительное число, в противном случае за командой JP 0810 выполняется следующая по очереди в программе команда.

Другой пример – JM 0810. По этой команде осуществляется условный переход к команде, записанной по адресу 0810, в случае, если в результате предыдущей команды получено отрицательное число (S = 1), в противном случае выполняется команда, расположенная вслед за JM 0810.

Список команд ввода-вывода содержит всего две команды. Адрес порта (port) указывается однобайтным числом D8, например, IN F3.

Оборудование и приборы

Лабораторная работа выполняется на эмуляторе учебного микропроцессорного комплекта (процессор KP580ИК80А).

Порядок выполнения работы

Задание 1: для исследования команд JMP A16, CALL A16, RZ, NOP и HLT записать и пошагово выполнить подпрограмму, ко-

торая представляет собой подпрограмму умножения двух 8-разрядных двоичных чисел без знака.

Адрес	Код	Метка	Мнемоника	Операнд	Комментарий
0850	31 C0 0B	START	LXI	SP, 0BC0	Начало стековой области определить адресом 0BC0
0853	00		NOP		
0859	CD 60 08		CALL	0860	Перейти к команде по адресу 0860
085C	00		NOP		
085F	76		HLT		
0860	21 00 00	MULT	LXI	H, 0000	Обнулить регистровую пару HL
0863	4A		MOV	C, D	Множимое загрузить в младший байт регистровой пары BC
0864	06 00		MVI	B, 00	Пара BC содержит множимое
0866	7B	CYCLE	MOV	A, E	Множитель загрузить в аккумулятор
0867	A7		ANA	A	Множитель равен нулю?
0868	C8		RZ		Если да, то возврат по адресу 085C
0869	09		DAD	B	Если нет, то прибавить множимое
086A	1D		DCR	E	Вычесть 1 из множителя
086B	C3 66 08		JMP	0866	Перейти по адресу 0866

В данной программе подпрограмма умножения MULT представляется в виде подпрограммы, вызываемой командой CALL A16, при обращении к которой множимое должно находиться в регистре D, а множитель – в регистре E микропроцессора KP580ИК 80А. Результат умножения, который может быть уже 16-разрядным числом, получается в регистровой паре HL. Поэтому для выполнения данной программы предварительно в регистры D и E нужно занести множимое и множитель в виде двоичных (шестнадцатеричных) чисел.

Задание 2: исследовать команды управления PCHL и JZ. Для этого предыдущую программу умножения изменить с использованием данных команд и выполнить ее в режиме трассировки.

Адрес	Код	Метка	Мнемоника	Операнд	Комментарий
0850	21 5A 08	START	LXI	H, 085A	
0853	00		NOP		
0854	00		NOP		
0855	E9		PCHL		
0856	00		NOP		
0857	00		NOP		
0858	00		NOP		
0859	76		HLT		
085A	21 00 00	MULT	LXI	H, 0000	
085D	4A		MOV	C, D	
085E	06 00		MVI	B, 00	
0860	7B	CYCLE	MOV	A, E	
0861	A7		ANA	A	
0862	CA 58 08		JZ	0858	
0865	09		DAD	B	
0866	1D		DCR	E	
0867	C3 60 08		JMP	0860	

Задание 3: исследовать команды работы со стеком PUSH H, POP H, XTHL, SPHL. Для этого предыдущую программу умножения изменить с использованием данных команд и выполнить ее в режиме трассировки.

Адрес	Код	Метка	Мнемоника	Операнд	Комментарий
0850	21 57 08	START	LXI	H, 0857	
0853	E9		PCHL		
0854	00		NOP		
0856	76		HLT		
0857	21 90 08	MULT	LXI	H, 0890	
085A	F9		SPHL		
085B	00		NOP		
085C	4A		MOV	C, D	
085D	06 00		MVI	B, 00	
085F	21 00 00		LXI	H, 0000	
0862	E3		XTHL		
0863	21 00 00		LXI	H, 0000	
0866	E1	CYCLE	POP	H	
0867	7B		MOV	A, E	
0868	A7		ANA	A	
0869	CA 54 08		JZ	0854	
086C	09		DAD	B	
086D	1D		DCR	E	
086E	E5		PUSH	H	
086F	C3 66 08		JMP	0866	

Задание 4: разработать программу деления двух однобайтных чисел с раздельной записью целого и остатка деления.

Контрольные вопросы

1. Назначение команд управления JMP, CALL, PCHL, RET, EI, DI, HLT, NOP, Jcc, Ccc, Rcc.
2. Назначение команд ввода-вывода IN port, OUT port.
3. Назначение команд работы со стеком PUSH гр, POP гр, XTHL, SPHL.

Оформление отчета

1. Задание.
2. Программы исследования команд управления, ввода-вывода и работы со стеком.
3. Результаты выполнения программ.
4. Выводы по корректности работы программ.

Список рекомендуемой литературы

1. *Вершинин О.Е.* Применение микропроцессоров для автоматизации технологических процессов / О.Е. Вершинин. – Ленинград: Энергоатомиздат, 1986. – 208 с., ил.
2. *Гилмор Ч.* Введение в микропроцессорную технику: Пер. с англ. – М.: Мир, 1984. – 334 с., ил.
3. Микропроцессоры и микроЭВМ в системах автоматического управления: Справочник / С.Т. Хвощ, Н.Н. Варлинский, Е.А. Попов; Под ред. С.Т. Хвоща. – Ленинград: Машиностроение, 1987. – 640 с., ил.
4. *Токхейм Р.* Основы цифровой электроники: Пер. с англ. – М.: Мир, 1988. – 392 с., ил.

Основы работы с эмулятором микропроцессорного комплекта БИС серии КР580

Окно эмулятора учебного микропроцессорного комплекта (УМК) БИС серии КР580 содержит следующие рабочие поля (рис.П1):

1. Код программы (основное рабочее поле).
2. Память (Alt+M).
3. Флаги и порты (Alt+F).
4. Регистры (Alt+R).

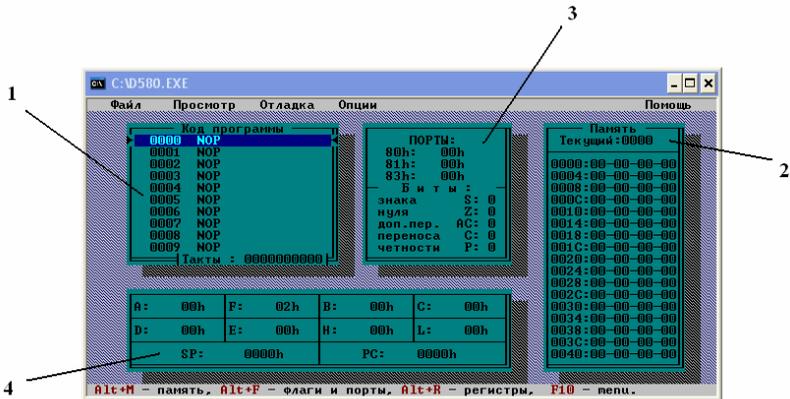


Рис.П1. Интерфейс эмулятора учебного микропроцессорного комплекта БИС серии КР580

Активация любого рабочего поля осуществляется только из основного рабочего поля (**Код программы**). При этом внизу рабочего окна появляются подсказки для работы в каждом рабочем поле эмулятора. Возврат в основное поле – Esc.

Поле **Код программы** содержит программу пользователя, записанную на языке ассемблера. В поле **Память** программу записывают в машинном коде. Переход к нужной ячейке памяти – Tab.

Режимы выполнения команд

Существуют три режима выполнения команд:

- Трассировка (F7).
- Трассировка и выполнение (F9).
- Быстрое выполнение (Ctrl+F9).

Работа с Меню

Выход в меню – F10. Каждый пункт основного меню содержит подпункты, необходимые для работы с эмулятором (табл. П1).

Таблица П1

Содержание основного меню эмулятора

Пункты меню	Подпункты
Файл	- загрузить программу (Alt+L) - выход в OS (Alt+X)
Просмотр	- изменение памяти (Alt+M) - изменение значений портов (Alt+F) - изменение регистров (Alt+R) - выбор формата значений регистров и портов (BIN или NEX)
Отладка	- выполнить один шаг (команду) (F7) - трассировать (выполнить) (F9) - быстрое выполнение программы (Ctrl+F9) - поставить (удалить) точку прерывания (Enter) - опции трассировки (Alt+O) - обнуление счетчика тактов (Alt+C)
Опции	- исходная точка трансляции программы - отключение трансляции выше исходной точки - включение трансляции выше исходной точки - сохранение состояния ресурсов и установок - восстановление состояния ресурсов и установок
Помощь	- как пользоваться отладчиком (F1) - как пользоваться компилятором - о программе

Порядок работы с эмулятором

Перед началом работы с эмулятором необходимо сохранить нулевые значения регистров и счетчика тактов (Опции/Сохранение состояния ресурсов и установок), чтобы в дальнейшем после выполнения каждой программы сбрасывать их значения (Опции/Восстановление состояния ресурсов и установок).

Команда пользователя записывается в машинном коде (табл. П2) в поле **Память** (Alt+M). Для заполнения данного поля нужно перейти к ячейке памяти, в которой содержится первая команда программы (Tab/Адрес ячейки). Далее ячейки заполняются последовательно.

Для выполнения программы вернуться в поле **Код программы** (Esc) и проверить правильность записи программы в мнемокоде. Быстрое выполнение программы осуществляется нажатием клавиш Ctrl+F9. Выполнение команды в режиме трассировки (т.е. по одной команде) требует вначале установки адреса ячейки с первой командой программы в счетчике команд PC (Alt+R).

Таблица П2

Кодирование команд микропроцессорного комплекта БИС серии КР580

		Вторая тетрада (HEX)															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Первая тетрада (HEX)	0	NOP	LXI B <B2> <B3>	STAX B	INX B	INR B	DCR B	MVI B <B2>	RLC	-	DAD B	LDAX B	DCX B	INR C	DCR C	MVI C <B2>	RRC

Продолжение табл. П2

Первая тетрада (HEX)					Вторая тетрада (HEX)															
5	4	3	2	1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
MOV D, B	MOV B, B	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV D, C	MOV B, C	LXI SP <B2> <B3>	LXI H <B2> <B3>	LXI D <B2> <B3>	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV D, D	MOV B, D	STA <L,ADR> <H,ADR>	SHLD <L,ADR> <H,ADR>	STAX D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV D, E	MOV B, E	INX SP	INX H	INX D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV D, H	MOV B, H	INR M	INR H	INR D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV D, L	MOV B, L	DCR M	DCR H	DCR D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV D, M	MOV B, M	MVI M B2>	MVI H B2>	MVI D <B2>	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV D, A	MOV B, A	STC	DAA	RAL	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV E, B	MOV C, B	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV E, C	MOV C, C	DAD SP	DAD H	DAD D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV E, D	MOV C, D	LDA <L,ADR> <H,ADR>	LHLD <L,ADR> <H,ADR>	LDAX D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV E, E	MOV C, E	DCX SP	DCX H	DCX D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV E, H	MOV C, H	INR A	INR L	INR E	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV E, L	MOV C, L	DCR A	DCR L	DCR E	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV E, M	MOV C, M	MVI A <B2>	MVI L <B2>	MVI E <B2>	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MOV E, A	MOV C, A	CMC	CMA	RAR	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Первая тетрада (HEX)						Вторая тетрада (HEX)																
C	B	A	9	8	7	6	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
RNZ	ORA B	ANA B	SUB B	ADD B	MOV M, B	MOV H, B																
POP B	ORA C	ANA C	SUB C	ADD C	MOV M, C	MOV H, C																
JNZ <LADR> <HADR>	ORA D	ANA D	SUB D	ADD D	MOV M, D	MOV H, D																
JMP <LADR> <HADR>	ORA E	ANA E	SUB E	ADD E	MOV M, E	MOV H, E																
CNZ <LADR> <HADR>	ORA H	ANA H	SUB H	ADD H	MOV M, H	MOV H, H																
PUSH B	ORA L	ANA L	SUB L	ADD L	MOV M, L	MOV H, L																
ADI <B2>	ORA M	ANA M	SUB M	ADD M	HLT	MOV H, M																
RST 0	ORA A	ANA A	SUB A	ADD A	MOV M, A	MOV H, A																
RZ	CMP B	XRA B	SBB B	ADC B	MOV A, B	MOV L, B																
RET	CMP C	XRA C	SBB C	ADC C	MOV A, C	MOV L, C																
JZ <LADR> <HADR>	CMP D	XRA D	SBB D	ADC D	MOV A, D	MOV L, D																
-	CMPE	XRA E	SBB E	ADC E	MOV A, E	MOV L, E																
CZ <LADR> <HADR>	CMP H	XRA H	SBB H	ADC H	MOV A, H	MOV L, H																
CALL <LADR> <HADR>	CMP L	XRA L	SBB L	ADC L	MOV A, L	MOV L, L																
ACI <B2>	CMP M	XRA M	SBB M	ADC M	MOV A, M	MOV L, M																
RST 1	CMP A	XRA A	SBB A	ADC A	MOV A, A	MOV L, A																

Окончание табл. П2

Первая тетрада (HEX)		Вторая тетрада (HEX)															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
⌘	⌘																
		⌘															
RP		RPO															
POP PSW		POP H															
JP<L ADR> <H ADR>		JPO <L ADR> <H ADR>															
DI		XTHL															
CP <L ADR> <H ADR>		CPO <L ADR> <H ADR>															
PUSH PSW		PUSH H															
ORI <B2>		ANI <B2>															
RST 6		RST 4															
RM		RPE															
SPLH		PCHL															
JM <L ADR> <H ADR>		JPE <L ADR> <H ADR>															
EI		XCHG															
CM <L ADR> <H ADR>		CPE <L ADR> <H ADR>															
-		-															
CPI <B2>		XRI <B2>															
RST 7		RST 5															

СОДЕРЖАНИЕ

Предисловие	3
Лабораторная работа №1. Исследование системного программного обеспечения "Монитор" учебного микропроцессорного комплекта (УМК)	4
Лабораторная работа №2. Исследование выполнения команд арифметических и логических операций микропроцессора КР580ИК80А.....	14
Лабораторная работа №3. Исследование команд пересылки данных микропроцессора КР580ИК80А.....	27
Лабораторная работа №4. Исследование команд управления, ввода-вывода и работы со стекком микропроцессора КР580ИК80А.....	34
Список рекомендуемой литературы	40
Приложение. Основы работы с эмулятором микропроцессорного комплекта БИС серии КР580	41

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ЛАБОРАТОРНЫЙ ПРАКТИКУМ

Методические указания
для студентов, обучающихся по специальности
220301 «Автоматизация технологических процессов и
производств»,
всех форм обучения

Составители:

Котляров Роман Витальевич
Поздняков Дмитрий Леонидович

Зав. редакцией *И.Н. Журина*
Редактор *Н.В. Шишкина*
Технический редактор *Т.В. Васильева*
Художественный редактор *Л.П. Токарева*

ЛР № 020524 от 02.06.97
Подписано в печать _____. Формат 60x84^{1/16}
Бумага типографская. Гарнитура Times
Уч.-изд.л. 5. Тираж ____ экз.
Заказ № ____

Оригинал-макет изготовлен в редакционно-издательском отделе
Кемеровского технологического института пищевой промышленности
650056, г. Кемерово, б-р Строителей, 47

ПЛД № 44-09 от 10.10.99
Отпечатано в лаборатории множительной техники
Кемеровского технологического института пищевой промышленности
650010, г. Кемерово, ул. Красноармейская, 52