



МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
имени Н.Э. БАУМАНА

Учебное пособие

Методические указания
по выполнению Лабораторной работы №3
по единому комплексному заданию по блоку дисциплины

«Проектирование микропроцессорных систем (МП и МК)»

МГТУ имени Н.Э. Баумана

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
имени Н.Э. БАУМАНА

Методические указания
по выполнению Лабораторной работы №3
по единому комплексному заданию по блоку дисциплины

«Проектирование микропроцессорных систем (МП и МК)»

Москва
МГТУ имени Н.Э. Баумана

2012

УДК 681.3.06(075.8)
ББК 32.973-018
И201

Методические указания по выполнению лабораторной работы №3 по единому комплексному заданию по блоку дисциплины «Проектирование микропроцессорных систем (МП и МК)» / Коллектив авторов – М.: МГТУ им. Н.Э. Баумана, 2012. – 41 с.: ил.

В методических указаниях рассмотрены основные этапы, их последовательность и содержание по выполнению лабораторной работы №3 по единому комплексному заданию по блоку дисциплины «Проектирование микропроцессорных систем (МП и МК)».

Ил. 39. Табл. 5. Библиогр. 7 назв.

УДК 681.3.06(075.8)

© МГТУ им. Н.Э. Баумана, 2012

ВВЕДЕНИЕ

Ускорение научно-технического прогресса в народном хозяйстве требует практического освоения микропроцессорной техники. Наряду со сведениями о внутренней структуре микро-ЭВМ, которые используются для встраивания в оборудование или для построения ЭВМ различной мощности, необходимо осваивать методы программирования микро-ЭВМ и микропроцессоров (МП), так как эффективность их использования зависит главным образом от качества программного обеспечения.

Программирование на языках высокого уровня позволяет в ряде случаев упростить процесс создания программ, их отладку и проверку. Однако в случае основного применения МП и микро-ЭВМ на их основе в качестве автономных контроллеров в системах автоматического управления (САУ) характерно использование языков низкого уровня - ассемблера или даже машинных языков. Это ускоряет выполнение программы и экономит память машины. Кроме того, программирование в машинных кодах в какой-то мере раскрывает специфику функционирования каждого типа процессора, в то время как на языках высокого уровня можно программировать любую ЭВМ, не вдаваясь в подробности внутренней структуры. Поэтому для ознакомления с работой микропроцессорных систем (МПС) целесообразно начать обучение с программирования в машинных кодах.

В первой части практикума проводится описание стенда лабораторного СЛ-1 и учебной лабораторной микро-ЭВМ УМПК-80/ВМ, построенной на основе микропроцессорного комплекта (МПК) серии КР580. Рассматриваются режимы ее работы и особенности использования в практикуме.

Во второй части практикума приводятся сами лабораторные работы, которые позволяют ознакомиться с функционированием микро-ЭВМ, подключением к ней различных устройств, особенностями программирования различных задач.

Приведенные в практикуме описания отдельных блоков и программного обеспечения позволяют студенту самостоятельно проектировать микро-ЭВМ и контроллеры и использовать их в проектировании ЭС.

1. Описание стенда лабораторного СЛ-1

Стенд СЛ-1 предназначен для размещения на нем модулей учебных микропроцессорных комплексов лаборатории МПС “Пирамида”.

Стенд выполняет защитную, несущую и экранирующую функции, а также служит для электрического соединения модулей между собой. Связь модулей осуществляется с помощью разъемов.

Электрическое питание стенда осуществляется от блока вторичного электропитания с выходными напряжениями -5В , $+5\text{В}$, -12В , $+12\text{В}$ и $+15\text{В}$.

Стенд позволяет реализовать в целях обучения любую существующую или разрабатываемую структуру МПС за счет применения регулярной электрической разводки разъемов. Данная разводка позволяет подключать любые модули одного учебного микропроцессорного комплекса УМПК-80 на произвольное посадочное место, используя разъемы для подключения модулей.

Стенд - прибор настольного типа (рис.1) состоит из основания (1), кросс-платы (2), пяти разъемов (3) для подключения модулей, блока питания и бытового магнитофона, крышек: металлической и пластмассовой.

Стенд осуществляет электрическое соединение модулей между собой и подачу питающих напряжений на модули от блока питания. Основным питающим напряжением комплексов и определяющим работоспособность блока питания является напряжение 5В .

Его наличие, а значит и наличие остальных напряжений на стенде индицируется светодиодом, выведенным на крышку. Выполнение ряда программ в УМПК сопровождается звуковым сигналом, выдаваемым пьезокерамическим звонком (звуковым излучателем), смонтированным на основании.

Стенд соединяется с блоком питания ленточным кабелем. При включении блока питания в сеть и нажатии на кнопку “СЕТЬ” загорается светодиод “ПИТАНИЕ”. При этом стенд совместно с блоком питания и модулями готов к работе.

2. Описание микро-ЭВМ УМПК-80/ВМ

Микро-ЭВМ УМПК-80/ВМ (в дальнейшем - модуль) предназначена для работы в стенде лабораторном СЛ-1 УМПК-80 из набора лаборатории изучения микропроцессорных систем “Пирамида”.

Основным назначением модуля является изучение схмотехнического и программного обеспечения микропроцессорной быстродействующей интегральной схемы (МП БИС) КР580ВМ80А, изучение схмотехнического и программного обеспечения МПК КР580, знакомство с особенностями построения, режимами работы и методами сопряжения с внешними устройствами микро-ЭВМ, построенного на базе данного МПК, а также обучение методам разработки и программирования микровычислительных средств студентов при изучении курса “Микропроцессоры в САУ”.

2.1. Технические характеристики

Модуль УМПК-80/ВМ представляет собой печатный узел, имеющий два разъема для подключения к стенду СЛ-1 и один разъем для сопряжения с внешними устройствами.

Электрическое питание от блока питания с напряжениями $-5\text{В}\pm 5\%$; $+5\text{В}\pm 5\%$; $+12\text{В}\pm 5\%$.

Разрядность магистрали адреса	- 16 бит.
Разрядность магистрали данных	- 8 бит.
Частота тактового питания	- 2 МГц.
Опорная частота тактового генератора	- 18 МГц.
Объем ПЗУ	- 2 Кбайта.
Объем ОЗУ	- 2 Кбайта.

Модуль производит индикацию светодиодным знаковым дисплеем в шестнадцатиричном коде значения: адреса - 4 разряда; данных - 2 разряда.

Модуль производит индикацию светодиодами состояния (в двоичном коде): магистрали адреса; магистрали данных; магистрали управления; входного регистра; выходного регистра.

Модуль позволяет исследовать методы построения микровычислительных средств, а также исследовать характеристики как МП БИС, так и отдельных интерфейсных БИС, входящих в состав МПК серии КР580.

Модуль позволяет исследовать методы программирования микровычислительных средств, в процессе выполнения программы, с клавиатуры и бытового магнитофона и содержит имитатор входного устройства.

Операционная система модуля позволяет выводить на светодиодный знаковый дисплей содержимое ОЗУ, ПЗУ и регистров МП БИС, а также обеспечивает возможность записи информации на бытовой кассетный магнитофон.

2.2 Устройство и принцип работы.

Функциональная схема микро-ЭВМ УМПК-80/ВМ показана на рис.2. Ее структура обычна для микро-ЭВМ с фиксированной разрядностью. Магистралы микро-ЭВМ состоят из трех шин: - однонаправленной шины адреса (16 разрядов) – **ША**; двунаправленной шины данных (8 разрядов) – **ШД**; однонаправленной шины управления (5 разрядов) – **ШУ**. В отдельных прямоугольниках схемы указаны цифры, соответствующие блокам на рис.3.

Ниже приводится поблочное описание устройства и принципа работы модуля по функциональной схеме, а в скобках указаны названия и позиционные обозначения элементов согласно схеме электрической принципиальной

На плате модуля размещаются:

1. Блок процессора - **БП** (DD21-КР580ВМ80А-МП БИС; DD18-КР580ГФ24 - тактовый генератор; DD25, DD26-КР580ВА86 - буфер магистралы адреса; DD27-КР580ВК28 - системный контроллер и буфер магистралы данных; DD40 - КР155ЛП10 - буфер внешней магистралы модуля).

Для работы МП БИС модуля требуются три питающих напряжения: $-5В \pm 10\%$, $+5В \pm 5\%$, $+12В \pm 5\%$ и тактовое питание - две последовательности тактовых импульсов $\Phi 1$ и $\Phi 2$ частотой от 500 кГц до 2 МГц (тактовая частота 2 МГц) с амплитудой 12В.

Тактовые последовательности $\Phi 1$ и $\Phi 2$ вырабатываются тактовым генератором D18, который также стробирует внешние сигналы RESIN (“Начальная установка”) и ADYIN (“Готовность”) для МП БИС. Частота генератора определяется кварцевым резонатором BQ1 и выбрана равной 18000 кГц. Сигналы $\Phi 1$ и $\Phi 2$ формируются делением опорной частоты 18000 кГц на 9.

Буфер магистрали адреса (DD25, DD26) предназначен для усиления сигналов магистрали адреса МП БИС A0 - A15. Во время доступа к магистрали адреса других ведущих устройств (например, контроллера прямого доступа к памяти (ПДП)) буфер может быть отключен нулевым сигналом AEN, подаваемым через инвертор DD4.2 на выводы DD25 и DD26.

Для формирования сигналов магистрали управления и усиления магистрали данных используется системный контроллер DD27. Он формирует первичные сигналы управления $\overline{I/OR}$, $\overline{I/OW}$, \overline{MEMR} , \overline{MEMW} , \overline{INTA} и усиливает сигналы, поступающие из МП БИС на двунаправленную магистраль данных модуля.

2. Блок управления картой памяти - **БУКП** (DD31 - K531КП11П - мультиплексор сигналов магистрали управления; и DD30 - K555ЛИ1, DD8 - K555ЛЛ1, DD34 - KR555ЛА2).

При функционировании - БУКП использует первичные сигналы управления $\overline{I/OR}$, $\overline{I/OW}$, \overline{MEMR} , \overline{MEMW} , \overline{INTA} поступающие с DD27. Выбор карты памяти осуществляется переключателем SA7.1.

Магистраль данных и управления по нулевому сигналу AEN также переводится в высокоимпедансное состояние.

3. Блок запоминающих устройств - **БЗУ** (DD29 - K537PY8Б - оперативное запоминающее устройство – **ОЗУ**; DD28 - KP556PT7 - постоянное запоминающее устройство – **ПЗУ**; DD33 - KP580BA86 – буфер - **Бф**; DD32 - K555ИД7 - схема дешифрации адреса - **СхДа**; DD2.6 - K555ЛН1; DD5.4, D30.3 - K555ЛИ1, SA6 – ВДМ1-2 – «Блокировка ЗУ», SA2.4 – ВДМ1-4 – «Защита ОЗУ»).

Обе микросхемы имеют третье состояние по выходам данных, что позволяет соединить их параллельно и подключить к магистрали данных через буфер DD33 - KP580BA86.

Сигналы выбора микросхем формирует схема дешифрации адреса DD32, DD2 - K555ЛН1, DD5, DD30 - K555ЛИ1. Дешифратор определяет адреса ПЗУ от 0000H до 07FFH и ОЗУ от 0800H до 0FFFH.

Переключателем SA6 - БЗУ может быть отключен.

4. Блок дешифрации адресов устройства ввода-вывода - **БДА** (DD35 - K155PE3 - дешифратор адресов, DD36 - K555ЛА1, DD17 - K155ЛЕ3, SA7.2 – ВДМ1-2 – «Блокировка УВВ»).

Устройствами ввода-вывода информации модуля УМПК-80/ВМ являются:

- 1) клавиатура - **К** с регистром чтения клавиатуры - **РгЧК**;
- 2) светодиодный знаковый дисплей - **Д** с регистрами сканирования - **РгСк** и сегментов - **РгСр**;
- 3) входной регистр - **РгВв** с имитатором - **И**;
- 4) выходной регистр в - **РгВыв**;
- 5) схема преобразования сигнала (схема звуковой сигнализации) - **СхПрС** и регистр вывода информации на магнитофон - **РгВывМг**;
- 6) регистр ввода информации с магнитофона - **РгВв**.

Для формирования сигналов выбора устройств, ввода-вывода, в модуле в качестве дешифратора используются ИМС DD35 - K155PE3 (предварительная дешифрация осуществляется схемой на элементах DD36 - K555ЛА1 и DD17 - K155ЛЕ3).

Применение в качестве дешифратора устройств микросхемы ПЗУ позволяет сократить число используемых элементов в схеме, а также, при необходимости, произвольно менять адреса устройств ввода-вывода (микросхема ПЗУ установлена в контактирующем устройстве).

В модуле использованы следующие адреса УВВ:

- 1) 04H (вывод) - СхПрС и РгВыв Мг;
- 2) 04H (ввод) - РгВвМг;

- 3) 05Н (вывод) - PГВыв;
- 4) 05Н (ввод) - PГВв;
- 5) 06Н (вывод) - PГСг;
- 6) 06Н (ввод) - PГЧК;
- 7) 07Н (вывод) - PГСк;
- 8) 08Н - OFH (вывод) - СхУпр.

Переключателем SA7.2 дешифратор УВВ может быть отключен.

5. Блок клавиатуры и дисплея - **БКД** (DD22 - KP580ИР82 - регистр сканирования – PГСк; DD23 - KP580ИР83 - регистр сегментов – PГСг; DD41 - K155ЛП11 - регистр чтения клавиатуры – PГЧК; DD24 - K561ЛН2, DD9 - K561ПУ4, DD1 - K555ЛН1, VT2 - VT7 - КТ350А, НГ1 - НГ6 - АЛС333Б - светодиодные знаковые индикаторы и кнопки клавиатуры).

Клавиатура модуля УМПК-80/ВМ содержит 26 дискретных клавиш .

На плате модуля размещены 16 цифровых шестнадцатиричных клавиш: “0”, “1”, “2”, “3”, “4”, “5”, “6”, “7”, “8”, “9”, “А”, “В”, “С”, “D”, “Е”, “F” и 10 клавиш управления: “R” - начальная установка; “Ст” – стоп; “П” – пуск; “ОтА” - отыскание адреса; “ОтPг” - отыскание регистра; “ШК” - шаг команды; “ШЦ” - шаг цикла; “ЗвУв” - запись/увеличить; “Ум” – уменьшить; и “ПрСч” - программный счетчик.

Дисплей предназначен для индикации в шестнадцатиричном коде содержимого ячеек текущего адреса, данных и внутренних, программно-доступных регистров МП БИС. Он включает в себя знаковые индикаторы АЛС333Б и управляется программно в динамическом режиме регистрами сегментов DD23 - KP580ИР83 и сканирования DD22 - KP580ИР82 через инверторы DD24 - K561ЛН2 с выходными транзисторными ключами VT2 - VT7 типа КТ350А.

6. Блок интерфейса магнитофона - **БИМ** (DD12.2 - K555ТМ2, DD41 - K155ЛП11, DD42 - K554СА3).

Блок интерфейса магнитофона содержит схему звуковой сигнализации (**СхПрС**) и схему ввода-вывода информации на магнитофон **PГВвМг** и **PГВывМг**.

СхПрС и **РгВывМг** содержат триггер (DD12 К555ТМ2), устанавливаемый и сбрасываемый программно по младшему биту магистрали данных. Тон звуковой частоты, как для формирования звука, так и для кодирования выводимой на магнитофон информации определяется последовательностью записываемых в триггер единиц и нулей, а также промежутками времени между ними.

РгВвМг содержит компаратор DD42 К554СА3А с положительной обратной связью и источник опорного напряжения на светодиоде HL30. Сигнал с выхода компаратора считывается МП БИС через буфер DD41 и обрабатывается программно.

7. Блок имитации периферийных устройств - **БИПУ** (DD16 - КР580ВА86-входной регистр – **РгВв**; DD6 - КР580ИР82 - выходной регистр – **РгВыв**; DD9, DD10 - К561ПУ4, DD1 - К555ЛН1, HL1 - HL16 – светодиоды; SA1 - переключатель с периферийными устройствами).

БИПУ содержит входной и выходной регистры, а также имитатор внешнего устройства, представляющий собой 8 переключателей, информация с которых отображается светодиодами HL1 - HL8 . Модуль можно сопрягать с периферийными устройствами через разъем X3 (8 линий на ввод информации, подключенных к выводам выходного регистра DD6).

8. Блок управления режимами работы МП БИС: - **БУ** (DD15 - КР580ВА86, - регистр кода прерывания – **РгКПр**; DD3 - К555ТМ8, DD12.1, DD11 - К555ЛЛ1, DD5 - К555ЛИ1, DD13, DD14 - К555ТМ2, DD7 - К555ЛА3, DD1, DD2, DD4 - К555ЛН1, SA2 - переключатели выбора режима).

БУ содержит схему управления - **СхУпр**, схему обработки сигналов - **СхОС** и регистр кода прерывания - **РгКПр**.

Схема управления - **СхУпр** предназначена для отладки программного обеспечения в пошаговых режимах. В модуле предусмотрено два таких режима:

- 1) режим выполнения программы по шагам команд;
- 2) режим выполнения программы по шагам машинных циклов.

Пошаговые режимы реализованы программно-аппаратно, причем первый с использованием прерывания программы после выполнения каждой команды, а второй путем установки в "0" сигнала READY МП БИС на каждом машинном цикле. Выполнение программы по командам производится с использованием клавиши "ШК", а машинным циклам - клавишей "ШЦ". Нажатие клавиши "Ст" выводит модуль из режима выполнения программы по машинным циклам, а также (в других случаях) производит останов программы, выполняемой в реальном масштабе времени при условии, если разрешены прерывания МП БИС. При нажатии на клавишу "Ст" появляется высокий потенциал на выходе INT МП БИС, что вызывает прерывание выполняемой программы, при этом по сигналу INTA, в цикле обработки прерывания, на магистраль данных регистром кода прерывания DD15 выдается вектор прерывания CFH. Выбор режима работы схемы управления производится записью соответствующего двоичного числа в регистр выбора режима DD3, что производится программно.

Для расширения функциональных возможностей модуля используется схема обработки сигналов - СхОС, позволяющая с помощью переключателей SA2 подавать на модуль внешние управляющие сигналы, коммутировать выходные сигналы МП БИС и защищать ОЗУ от случайного стирания рабочей программы.

В верхнем положении переключателей SA2 сигналы коммутируются следующим образом:

- 1) A2.1 - выключен внешний сигнал READY;
- 2) A2.2 - выключен внешний сигнал HOLD;
- 3) A2.3 - выключен внешний сигнал INT;
- 4) A2.4 - включена защита первой половины ОЗУ (адреса 0800H - 0BFFH) от записи в него информации.

9. Блок индикации состояния магистралей - **БИСМ** (DD10, DD20, DD37, DD39 - K561ПУ4, D19 - K561ЛН2 - буфера светодиодов, HL22-HL29 КИПМ01-1К - светодиоды магистрали данных, HL31-HL21 - КИПМ02 - светодиоды магистрали адреса, HL17-HL21 - КИПМ02 - светодиоды магистрали управления).

На плате микро-ЭВМ расположены следующие составные части (рис.3).

1. Блок процессора – **БП**.
2. Блок управления картой памяти – **БУКП**.
3. Блок запоминающих устройств – **БЗУ**.
4. Блок дешифрации адресов устройства ввода-вывода – **БДА**.
5. Блок клавиатуры и дисплея – **БКД**.
6. Блок интерфейса магнитофона – **БИМ**.
7. Блок имитации периферийных устройств – **БИПУ**.
8. Блок управления режимами работы МП БИС – **БУ**.
9. Блок индикации состояния магистралей - **БИСМ**.

2.3. Порядок работы микро-ЭВМ

2.3.1. Подготовка к работе. Включить микро-ЭВМ в сеть. Перевести тумблер “Сеть” на правой боковине корпуса в положение “вкл”. Вверху на лицевой панели должен загореться светодиод “Сеть”. В случае неисправностей цепи питания микро-ЭВМ будет гореть красный светодиод “Перегрузка”. Вывод микро-ЭВМ из этого режима возможен выключением и повторным включением тумблера “Сеть”. Если таким способом устранить перегрузку не удастся, то необходимо выключить питание и обратиться к преподавателю.

2.3.2. Проверка функционирования. После нормального включения микро-ЭВМ осуществляет самопроверку работоспособности своих блоков:

- проводится проверка установки регистров МП. При неправильной установке на дисплее высвечивается сообщение: **SPEr**;

- проверяется правильность работы ПЗУ путем суммирования кодов, записанных в ячейках. При неправильной работе ПЗУ на дисплее высвечивается сообщение: **П З У**, сопровождаемое звуковым сигналом;

- проводится проверка записи и воспроизведения информации из ОЗУ. При сбоях на дисплее высвечивается сообщение: **О З У**, сопровождаемое звуковым сигналом;

- проверяется работа дисплея. При этом на короткое время все сегменты индикаторов должны зажечься;

- осуществляется проверка функционирования выходного регистра путем записи в него единиц. При этом должны загореться все желтые светодиоды выходного порта.

Если все эти проверки прошли нормально, то окончание их сопровождается звуковым сигналом и на дисплее высвечивается надпись: **НАЧАЛО**. После этого микро-ЭВМ готова к работе.

2.3.3. Работа с памятью и регистрами. Для работы с ячейками памяти и регистрами МП используются клавиши **ОтА**, **ОтРг**, **ПрСЧ**, **ЗпУв**, **Ум**. При начальной установке микро-ЭВМ счетчик команд устанавливается в положение, соответствующее началу области ОЗУ –0800Н. Все ячейки ОЗУ и регистры МП обнуляются.

Содержимое счетчика команд можно просмотреть на дисплее, нажав клавишу **ПрСч**. При необходимости просмотреть ячейки, начиная с адреса, отличного 0800Н, необходимо нажать клавишу **ОтА**. При этом на дисплее загорается нижние горизонтальные сегменты, сигнализирующие о готовности принять адрес. Набор адреса начинают с набора на цифровой клавиатуре старшей 16-ричной цифры, а затем после набора последующих цифр она сдвигается влево на одно знакоместо. После набора всех четырех цифр адреса на двух крайних левых знакоместах появится число, записанное в данной ячейке памяти.

Содержимое этой ячейки можно изменить путем нажатия на цифровые клавиши. После первого нажатия на крайне правом индикаторе загорается сегмент запятой, что означает, что микро-ЭВМ находится в режиме ввода в память. После набора информации (две 16-ричные цифры - сначала старшая, затем младшая) для ее записи в ячейку необходимо нажать клавишу **ЗпУв**. При этом на дисплее автоматически индицируется адрес и содержимое следующей ячейки памяти.

Примечание: Если по ошибке произойдет попытка записи в область ПЗУ или область, не соответствующую номерам ячеек ОЗУ, то микро-ЭВМ не выполняет команду и выдает звуковой сигнал. Номер ячейки на дисплее не изменяется.

Если необходимо просто просмотреть содержимое ячеек памяти, то после набора начального адреса нужно последовательно нажимать клавишу ЗпУв. При необходимости просмотреть содержимое ячеек в обратном порядке, нужно нажимать клавишу Ум.

Иногда возникает необходимость в просмотре и модификации содержимого внутренних регистров МП. Для этой цели служит клавиша **ОтРг**. При ее нажатии на дисплее возникает надпись А□□, где - □□ содержимое регистра А микропроцессора. Это содержимое можно изменять набором на цифровой клавиатуре на дисплее новых данных и нажатием клавиши **ЗпУв**. После ее нажатия на дисплее будет выведено содержимое следующего регистра FL □□ - содержимое регистра признаков МП (смысл каждого ряда указан на передней панели микро-ЭВМ). Далее выводится:

- В□□ - содержимое регистра В МП,
- С□□ - содержимое регистра С МП,
- Д□□ - содержимое регистра D МП,
- Е□□ - содержимое регистра Е МП,
- Н□□ - содержимое регистра Н МП,
- L□□ - содержимое регистра L МП,
- SPH□□- старший байт указателя стека,
- SPL□□- младший байт указателя стека,
- PLH□□- старший байт счетчика команд МП,
- PCL□□- младший байт счетчика команд МП.

Выход из режима индикации содержимого регистров осуществляется нажатием клавиши **ПрСч**. При этом на дисплей выводится адрес и содержимое той ячейки, на которой остановилась программа.

2.4. Запуск программы

2.4.1. Режим запуска. Запуск программ возможен в трех режимах: автоматическом, когда программа выполняется с адреса, находящегося на дисплее и до конеч-

ного оператора, в пошаговом режиме, когда программа останавливается каждый раз при выполнении одного оператора, либо каждый раз по выполнении машинного цикла.

2.4.2. Автоматическое исполнение команд. Для запуска программы в этом режиме необходимо с помощью клавиши **ОтА** и цифровой клавиатуры набрать адрес начала работы программы. Затем нажать клавишу **П**, при этом дисплей погаснет, что свидетельствует о выполнении программы. В этом режиме можно прервать исполнение программы, нажав клавишу **Ст**, при этом на дисплее будет индицироваться адрес останова и содержание соответствующей ячейки. Просмотрев при необходимости содержимое регистров, можно продолжить выполнение программы, нажав клавишу **П**. По окончании выполнения программы на дисплее высвечивается адрес нормального останова

2.4.3. Пошаговое выполнение программы по операторам. В этом режиме вместо клавиши **П** нажимают клавишу **ШК**. При этом на дисплее будет индицироваться адрес следующего по программе оператора и само его значение. Последовательно нажимая клавишу **ШК** можно просмотреть выполнение программы до конца.

2.4.4. Пошаговое выполнение программы по циклам. Для запуска программы в этом режиме нажимается клавиша **ШЦ**. При этом становится возможным на каждом шаге оценить состояние каждого разряда шин микро-ЭВМ. **ША** по светодиодам А0 ... А15, **ШД** по светодиодам Д0 ... Д7, **ШУ** по светодиодам Чт, Зп, ОЗУ, ПЗУ, Вх, Вых УУ. Дисплей в этом режиме гаснет, нажимая последовательно клавишу **ШЦ** можно просмотреть выполнение программы по машинным циклам. Вывод микро-ЭВМ из этого режима осуществляется нажатием клавиши **Ст**. При этом на дисплее высвечивается последний адрес команды, которая выполнялась в пошаговом режиме.

Примечание: 1) Повторное нажатие на клавишу **Ст** вызывает вызов управляющей программы и перезапуск микро-ЭВМ. На дисплее высвечивается надпись **Н А Ч А л о**, но содержимое регистров МП и ячеек ОЗУ сохраняется.

2) Клавиша **Р** позволяет перезапустить микро-ЭВМ, причем осуществляется тестирование, обнуление всех ячеек ОЗУ и вывод надписи **Н А Ч А л о**.

В табл.2.1 приведены для справки функции всех функциональных клавиш микро-ЭВМ.

Таблица 2.1

Клавиша	Наименование клавиши	Выполняемая операция	Примечание
1	2	3	4
Р	Сброс	Сброс регистров МП в 0 и перезапуск микро-ЭВМ	Регистры МП и ячейки ОЗУ обнуляются, проводится тестирование и зажигается надпись НАЧАЛО
Ст	Стоп	Останов запущенной на счет программы	Повторное нажатие клавиши приводит к перезапуску, но без потери информации в регистрах и ячейках ОЗУ
П	Пуск	Пуск программы на счет в автоматическом режиме	
ОтА	Отыскание адреса	Запуск режима ввода адреса ячейки	На дисплее загорается нижние горизонтальные сегменты
ОтРг	Отыскание регистра	Запуск режима просмотра и модификации содержимого регистров МП	Порядок просмотра: А,В,С,Д,Е,Н,Л,С,Р,РС,А и т.д. Просмотр - нажатие кл. ЗпУв , выход из режима - нажатие клавиши ПрСч .
ЗпУв	Запись/увеличить	Запись набранной информации в ячейку и увеличение СК на 1	Если информация не набрана на цифровой клавиатуре, то переход к просмотру следующей ячейки.
Ум	Уменьшить	Уменьшение содержимого СК на 1	
ШК	Шаг команды	Запуск в режиме пошагового выполнения программы по операторам	Последовательное выполнение программы по командам при нажатиях на клавишу ЗпУв .
ШЦ	Шаг цикла	Запуск в режиме пошагового выполнения программы по циклам	Последовательное выполнение программы по командам при нажатиях на клавишу Зп Ув . Выход из режима Ст .
ПрСч	Программный счетчик	Вывод на дисплей текущего значения СК	

3. Лабораторная работа № 1.

Ознакомление с работой учебной

микро-ЭВМ. Выполнение простейших программ.

3.1. Цель работы. Ознакомление со структурой микро-ЭВМ, распределением памяти, режимами работы. Получение навыков отладки простейших программ.

3.2. Задание для домашней подготовки.

1. Ознакомиться с описанием учебной микро-ЭВМ.

2. По рекомендованной литературе изучить минимальную типовую структуру микро-ЭВМ, методы организации магистралей, подключение блоков к магистралям.

3. По литературе ознакомиться с системой команд МП, способами адресации, временными диаграммами выполнения каждой команды. Изучить методы программирования на языке Ассемблера.

4. Рассмотреть правила выполнения команд INR A, DCR A, ADD A, ANA A, ORA A, CMP A.

5. Рассмотреть результат выполнения программы 3.1 (см. ниже) при записи по адресу 0803 команд, приведенных в п.4 задания. Результаты выполнения программы 3.1 при различных командах занести в табл.3.1.

Таблица 3.1.

Команда, записанная по адресу 0B00	Число, записанное по адресу 0803	Число, записанное по адресу 0B01

6. Разработать программы: а) увеличения на 5 числа, записанного по адресу 0B00 и записи результата по адресу 0B01 (программа 3.3); б) сложения чисел, записанных по адресам 0B00 и 0B01 и записи результата по адресу 0B01 (программа 3.4); в) сравнения чисел в адресах 0B00 и 0B01 и записи большего из них в регистр В (программа 1.5)

3.3. Краткие сведения из теории.

Программа записывается в память микро-ЭВМ в последовательные ячейки в виде двоичных цифр. Удобно представлять их в шестнадцатичном виде. Кроме того, каждая команда имеет свое сокращенное наименование или мнемонику. Эта мнемоника вместе с операндами (т.е. данными, с которыми оперирует команда) и представляют собой запись программы на языке Ассемблера. Одно-, двух- или трех-байтовые команды занимают в памяти соответственно 1, 2 или 3 ячейки.

Рассмотрим простейшую программу (программа 3.1), извлекающую число из ячейки с адресом 0B00, инвертирующую это число и записывающую результат в ячейку с адресом 0B01. Программа записывается в таблицу: первым столбцом которой является адрес соответствующего оператора программы; вторым - машинный код оператора; третьим - метка, т.е. символическое имя оператора, на который есть ссылка в программе; четвертым - мнемоника, т.е. обозначение команды вместе с операндами (если они есть) и, наконец, пятым - комментарии, в которые записывается в кратком виде сущность команды или группы команд. Таким образом, исходная программа выглядит следующим образом:

Программа 3.1

Адрес	Код	Метка	Мнемоника	Комментарии
0800	3A000B		LDA, 0B00	; Поместить в аккумулятор число из ячейки.
0803	2F		CMA	; Инвертировать содержимое аккумулятора.
0804	32010B		STA, 0B01	; Возвратить в ячейку содержимое аккумулятора.
0807	CF		RST 1	; Прервать выполнение программы.

Как видно из записи, каждая строчка соответствует одной команде, независимо от ее длины. Поэтому, например, первая команда, которая является трехбайтной, содержит в столбце “Код” сразу три байта, а следующая команда начинается с адреса 0803. Такой способ записи более компактен и нагляден. Последний оператор представляет собой команду программного прерывания, после которого МП пере-

ходит к адресу 0008. В этом проявляется особенность управляющей программы микро-ЭВМ: начиная с адреса 0008, записан ряд операторов, позволяющих возвратиться из программы пользователя к управляющей программе с сохранением содержимого всех регистров.

Следует обратить внимание на первый и третий операторы. Они являются примером прямого способа адресации, когда адрес расположения информации помещен непосредственно в теле оператора. Эту же программу можно переписать, используя способ косвенной адресации (программа 3.2).

Программа 3.2.

Адрес	Код	Метка	Мнемоника	Комментарии
0800	21000B		LXI H, 0B00	; Записать в пару регистров HL адрес ячейки памяти.
0803	7E		MOV A, M	; Поместить в аккумулятор содержимое этой ячейки.
0804	2F		CMA	; Инвертировать число в регистре A.
0805	23		INX H	; Увеличить содержимое пары HL на 1.
0806	77		MOV M, A	; Записать число из A в ячейку памяти, адрес которой помещен в HL.
0807	CF		RST 1	; Возврат в основную программу.

В данном конкретном случае программа занимает столько же ячеек, что и программа 3.1. однако иногда, таким образом, удастся сэкономить память.

3.4. Задания к лабораторной работе

3.4.1. Исследовать порядок включения микро-ЭВМ.

1. Подключить шнур питания к сети.
2. Включить тумблер - Сеть.

В результате выполнения тестовых программ, светодиоды выходного регистра и сегменты дисплея будут включены примерно на 2с. После чего микро-ЭВМ подает

звуковой сигнал и выводит на дисплей надпись - **Н А Ч А л о**. Если микро-ЭВМ не выполняет этих действий, необходимо попытаться перезапустить ее, нажав клавишу - **Р**, или выключением и повторным включением напряжения сети.

3.4.2. Исследовать содержимое памяти

1. Нажать на клавишу - **ОтА**. При этом на дисплее должны гореть нижние горизонтальные сегменты.

2. Последовательно набрать на клавиатуре цифры 0,8,0,0. Убедиться, что при этом каждая цифра записывается в младший разряд адресного дисплея (на четвертое справа знакоместо), а предыдущие цифры сдвигаются влево на один разряд. При введении четвертой цифры на двух крайних правых знакоместах появится содержимое этой ячейки (в данном случае это будет 00).

3. Нажать на клавишу - **Зп Ув**. При этом адрес увеличится на 1, а содержимое не изменится. Последовательно нажимая клавишу - **ЗпУв**, проверить содержимое нескольких первых ячеек ОЗУ.

4. Нажать клавишу - **ОтА** и ввести адрес 0000 (первый адрес ПЗУ). На дисплее должно появиться содержимое первой ячейки - 26. Последовательно нажимая на клавишу - **ЗпУв** просмотреть содержимое нескольких ячеек ПЗУ.

5. Убедиться, что содержимое ячеек можно посмотреть в обратном порядке, нажимая последовательно клавишу - **Ум**.

3.4.3. Запись числа в ячейки памяти

1. Нажать на клавишу - **Р**.

2. Вывести на дисплей содержимое ячейки с адресом 0800 (см. п.3.4.2).

3. Набрать на цифровой клавиатуре 1. Убедиться, что это число вводится в крайне правое знакоместо дисплея. При этом в разряде дисплея появляется десятичная точка, что сигнализирует о режиме ввода в ячейку (т.е. на дисплее находится не реальное содержимое, а число, которое мы хотим ввести).

4. Нажать на клавишу 2. Убедиться, что цифра 1 передвинулась влево на один разряд, а двойка заняла место единицы. Проверить, что последующий ввод других цифр приводит к такому же эффекту.

5. Записать набранную информацию в ячейку, нажав на клавишу - **ЗпУв**. Убедиться, что адрес ячейки увеличился на 1, а десятичная точка погасла.

6. Записать любое число в ячейку 0801.

7. Нажать на клавишу - **Ум**. При этом на дисплее появится адрес и содержимое ячейки 0800. Убедиться, что записанная на предыдущем шаге информация сохранилась.

8. Осуществить вывод на дисплей содержимое первой ячейки ПЗУ (адрес 0000).

9. Записать по этому адресу какое-либо число. Убедиться, что после нажатия клавиши - **ЗпУв** микро-ЭВМ выдает звуковой сигнал, квалифицируя это действие, как ошибку. При этом увеличение адреса не происходит.

3.4.4. Записать числа в регистры МП

1. Нажать на клавишу - **ОтРг**. На дисплей будет выведено содержимое аккумулятора (регистра А). Многократно нажимая на клавишу - **ЗпУв**, проверить последовательность вывода на дисплей содержимое других регистров МП.

2. С помощью цифровых клавиш изменить содержимое какого-либо регистра МП. Нажать на клавишу - **ЗпУв**. Клавишей - **Ум** возвратиться к отображению содержимого предыдущего регистра и убедиться, что оно поменялось на число, набранное на предыдущем шаге.

3. Самостоятельно вывести микро-ЭВМ из этого режима.

3.4.5. Осуществление пуска программы

1. Вывести на дисплей содержимое ячейки ПЗУ 05В0. По этому адресу в ПЗУ записана программа исполнения известной мелодии. Нажать на клавишу - **П**, при этом микро-ЭВМ исполнит ее.

2. Убедиться, что выполнение программы может быть прервано нажатием клавиши - **Ст**. При останове программы на дисплей будет выводиться адрес, записанный в программном счетчике, и соответствующий код этого оператора. Проверить возможность вывода на дисплей содержимого регистров МП на момент останова.

3. Нажатием на клавишу - **ПрСч** вывести на дисплей адрес, на котором было прервано исполнение программы. Нажать на клавишу - **П** продолжить выполнение программы.

3.4.6. Исследование программы 3.1.

1. Ввести в микро-ЭВМ программу 3.1.
2. Записать по адресу 0В00 исследуемое число.
3. Осуществить пуск программы с адресом 0800. Проверить результат выполнения программы - содержимое ячейки 0В01.

4. Исследовать процесс выполнения программы по шагам, для чего нажать клавишу ШК. После выполнения каждой команды анализировать содержимое регистров МП.

5. Исследовать выполнение программы 3.1 по машинным циклам. Обратить внимание на последовательность передачи и преобразования информации в микро-ЭВМ при выполнении каждой команды (по светодиодам состояния шин микро-ЭВМ).

6. Заменяя, в программе 3.1 оператор СМА на указанные в задании операторы, исследовать выполнение программы 3.1. Проверить таблицу 3.1, заполненную при выполнении домашнего задания.

3.4.7. Исследование программы 3.2.

1. Ввести в микро-ЭВМ программу 3.2.
2. Записать по адресу 0В00 исследуемое число.
3. Осуществить пуск программы с адреса 0800. Проверить результат выполнения программы по числу, находящемуся по адресу 0В01.

4. Исследовать процесс выполнения команды MOV F, M по машинным циклам.

5. Ввести и исследовать выполнение программы, в которой предусматривается первоначальная загрузка числа в ячейку 0В00.

3.4.8. Исследование программы 3.3.

1. Ввести в микро-ЭВМ программу, разработанную при домашней подготовке.

2. Осуществить пуск программы с начального адреса и проверить результат ее выполнения по числу, записанному по адресу 0B01 при числах 05 и FE, записанных по адресу 0B00.

3.4.9. Исследование программы 3.4.

1. Ввести в микро-ЭВМ программу 3.4. разработанную при домашней подготовке.

2. Проверить результат ее выполнения по числу в ячейке 0B01, последовательно записывая по адресам пуск 0B00 и 0BA0 соответственно числа 0B и B0, FE и B5 и осуществляя пуск программы.

3. Видоизменить и исследовать программу 3.4 для случая, когда сумма двух чисел будет превышать 25610.

3.4.10. Исследование программы 3.5.

1. Ввести в микро-ЭВМ программу 3.5, разработанную при домашней подготовке.

2. Записать по адресам 0B00 и 0B01 исследуемые числа.

3. Осуществить пуск программы 3.5. Проверить результат ее выполнения по числу, записанному в регистре В.

3.5. Содержание отчета

Отчет должен содержать:

1. Структурную схему учебной микро-ЭВМ.

2. Карту распределения памяти между ОЗУ и ПЗУ.

3. Информацию о содержании внутренних регистров МП после начальной установки микро-ЭВМ.

4. Заполненную табл. 3.1 для случаев выполнения программы 3.1 при использовании команд, представленных в п.4 домашнего задания.

5. Видоизмененную программу 3.2, записанную в машинных кодах, позволяющую первоначально записывать в ячейку 0B00 исследуемое число.

6. Разработанные при домашней подготовке программы 3.3, 3.4, 3.5.

7. Результаты исследований работы программ по п. 3.4.8 - 3.4.10.

3.6. Задания для самопроверки

1. Показать структурную схему микро-ЭВМ
2. Что такое карта памяти микро-ЭВМ? Какие адреса отведены под ПЗУ и ОЗУ?
3. Рассказать о возможных режимах работы микро-ЭВМ.
4. Как записать числа в регистры МП или память микро-ЭВМ?
5. Какие узлы проверяются в микро-ЭВМ в процессе выполнения пусков, при начальной установке? Какие сообщения выводятся на дисплей при сбоях в узлах микро-ЭВМ и при нормальном запуске?
6. Как выйти из режима выполнения программы по командам?
7. Описать функции каждой клавиши микро-ЭВМ.
8. Какие значения записываются в ячейке ОЗУ в регистры МП при начальной установке микро-ЭВМ?
9. За сколько машинных тактов выполняются каждая команда в программах 3.1 и 3.2?
10. Какие способы адресации используются в микро-ЭВМ, построенной на МП серии K580?
11. При выполнении, каких команд, приведенных в п.5, задействуются разряды регистра флажков МП?
12. Что происходит, если при суммировании двух чисел их сумма в аккумуляторе МП превышает 256_{10} ?
13. Составить программу записи в аккумулятор содержимого ячейки памяти, суммирования этого содержимого с числом во второй ячейке, инвертирования окончательного результата и засылки его в первую ячейку памяти. Оценить длину программы в байтах и ее быстродействие.
14. Составить программу, аналогичную п.13, с применением косвенной адресации. Оценить длину и быстродействие программы.

15. Написать программу сравнения содержимого четырех соседних ячеек памяти и записи большего из них в регистр А, а адреса этой ячейки - в пару регистров HL.

16. Составить программу, аналогичную п.15, только необходимо записать в регистр А меньшее из этих четырех чисел.

17. Какие разряды регистра флажков F задействуются при нахождении максимального числа (п.15) и минимального (п.16) числа из некоторого массива данных?

18. Какие операции в системе команд МП КР580ВМ80А управляют непосредственным изменением разрядов регистра флажков - F? Какие разряды можно изменить? Как “сбросить” бит переноса?

19. Перечислить команды шестнадцатиричной арифметики и логики, которые входят в состав системы команд МП КР580ВМ80А. Объяснить их действие.

20. Для каких целей используется бит переноса, который формируется при логических и арифметических операциях микропроцессора?

4.Лабораторная работа №2.

Программирование микропроцессорных систем

4.1 Цель работы. Исследование методов организации обмена с устройствами ввода-вывода, программирование сложных задач обработки информации.

4.2. Краткие сведения из теории.

4.2.1. Обмен с внешними устройствами в МП системах осуществляется по командам ввода в аккумулятор IN A1, где A1- адрес устройства ввода (максимально-возможное число устройств ввода - равно 256_{10}), и вывода байта из аккумулятора OUT A2 (число устройств вывода тоже равно 256_{10}).

В качестве устройств ввода-вывода, используются различные интегральные схемы. В простейшем случае - это просто управляемые регистры. В учебной микро-ЭВМ есть одно устройство ввода, которое позволяет вводить информацию, набранную на переключателях, а также одно устройство вывода, позволяющее выводить из микро-ЭВМ информацию на светодиодный индикатор. Основу этих устройств со-

ставляет ИС многорежимного буферного регистра (МБР) типа К589ИР12. Схема подключения входного порта (так еще называют устройство сопряжения микро-ЭВМ с внешними устройствами) показана на рис.4. ШД микро-ЭВМ подключается к выводам D_0, \dots, D_7 МБР. К выходам регистра Q_0, \dots, Q_7 подключены кнопки S_1, \dots, S_7 , с помощью которых моделируется информация, которую нужно ввести в микро-ЭВМ, а также буферные схемы и светодиоды для отображения вводимой информации. В данном режиме (когда на входах $STB = 1$, а $MD = 0$) информация передается от выходов Q_0, \dots, Q_7 к входам D_0, \dots, D_7 , когда на входах $CS1$ и $CS2$ присутствует уровень лог.0. Перезапись информации с переключателем на ШД происходит, когда на ША выставлен адрес данного порта (тогда на выходе схемы дешифрации адреса будет уровень лог.0), в момент подачи сигнала чтения из устройства ввода-вывода (**ЧТВВ** = 0).

Порт вывода микро-ЭВМ также содержит МБР, который сигналами $STB = MD = 1$ установлен в режим, когда информация с ШД передается в МБР и запоминается там до прихода следующего сигнала записи в устройство ввода-вывода (**ЗПВВ**). Выход регистра нагружен на светодиодные индикаторы, показывающие значения каждого разряда выводимых данных. Схема дешифрации адреса (**СхДША**) имеет ту же функцию, что и в предыдущей схеме.

Простейшая программа (Программа 4.6) перезаписи числа из входного порта (его адрес в микро-ЭВМ равен 20) в выходное устройство (с адресом 30) выглядит следующим образом:

Программа 4.6

Адрес	Код	Метка	Мнемоника	Комментарии
0800	DB20	CNT:	IN 20	; Записать число из порта с адресом 20 в регистр А.
0802	D330		OUT30	; Записать число из регистра А в порт 30.
0804	C30008		JMP CNT	; Переход на метку CNT.

Во многих случаях при выполнении программы ее прямолинейный ход разветвляется в зависимости от выполнения определенных условий. Разветвление осуществляется с помощью операторов условного перехода. Эти операторы (см. систему команд МП), передают управление на оператор, адрес которого указан во втором и третьем байтах команды, если выполняется определенное условие. Выполнение условия проверяется по состоянию регистра флажков, в котором фиксируется результат предыдущей операции. Как известно, пять разрядов этого регистра несут информацию о наличии нуля в аккумуляторе, возникновении переноса из старшего разряда аккумулятора и из третьего разряда в четвертый, четности и знаке результата. Однако следует помнить, что не все команды МП могут воздействовать на признаки.

Во многих случаях при выполнении программ необходимо проверять или видоизменять (маскировать) состояние одного или нескольких разрядов числа. Это можно осуществить с помощью следующих логических операций с содержимым аккумулятора:

- логического умножения числа в регистре А на маску, которое обнуляет разряд числа, если разряд маски равен 0, и не изменяет разряд числа, если разряд маски равен 1 - операция ANI B₂ или операция ANA R, где B₂ - второй байт двухбайтной команды (маска), а R - регистр МП в котором помещена маска. Само число помещается в аккумулятор;

- логического сложения числа в аккумуляторе и маски, которое устанавливает соответствующий разряд числа в 1, если разряд маски равен 1, и не изменяет разряд числа, если разряд маски равен 0- операторы ORI B₂ и ORA R;

- логического - “исключающего ИЛИ”, которое инвертирует соответствующий разряд числа, если разряд маски равен 1 и не изменяет его, если разряд маски равен 0 (операторы XRI B₂ и XRA R).

При выполнении всех логических команд задействует разряды регистра признаков: Z - нуля, S - знака, P- четности, AC - дополнительного переполнения. Это позволяет проверить состояние любого разряда числа и организовать условные пе-

реходы. Программа маскирования отдельных разрядов числа (программа 4.7) помещена ниже:

Программа 4.7

Адрес	Код	Метка	Мнемоника	Комментарии
0800	DB20	CNT:	IN 20	; Получение числа из входного устройства.
0802	E620		ANI 20	; Маскирование пятого разряда.
0804	D330		OUT 30	; Запись результата в выходной порт 30.
0806	C30008		JMP CNT	; Переход на начало.

В программе определения появления 1 в пятом разряде числа, записанного со входного порта (программа 4.8) используется оператор условного перехода в случае нулевого результата при выполнении логической операции:

Программа 4.8

Адрес	Код	Метка	Мнемоника	Комментарии
0800	DB20	WAIT:	IN 20	; Получение числа из входного устройства.
0802	E620		ANI 20	; Маскирование пятого разряда.
0804	CA0008		JZ WAIT	; Идти на WAIT, если в пятом разряде был 0 (Z=0).
0807	CF		RST1	; Возврат основную программу.

Если необходимо проверять сразу несколько разрядов числа, то в программе предусматривают несколько циклов проверки. Например, в программе 4.9 предусматривается проверка появления 1 во втором и в пятом разрядах. Эта программа содержит два цикла.

Программа 4.9

Адрес	Код	Метка	Мнемоника	Комментарии
0800	DB20	WAIT1:	IN 20	; Получение числа из входного регистра.

0802	E604		ANI 00000100	; Маскирование второго разряда.
0804	CA0008		JZ WAIT1	; Если нет, продолжить WAIT1.
0807	3EFF		MVI A, FF	; Если да, зажечь светодиоды. Вывод в выходной регистр всех единиц.
0809	D330		OUT 30	; Вывод числа в порт 30.
080B	DB20	WAIT2:	IN 20	; Ввод числа из входного порта 20.
080D	E620		ANI 00100000	; Маскирование пятого разряда.
080F	CA0B08		JZ WAIT2	; Включен ли пятый тумблер? Если нет, продолжить WAIT2.
0812	3E00		MVI A, 00	; Если да, погасить светодиоды выходного регистра.
0814	D330		OUT 30	; Вывод числа в порт 30.
0816	C30008		JMP WAIT I	; Переход на начало. Повторить программу.

В этой программе в зависимости от номера включенного тумблера гаснут или зажигаются все светодиоды выходного порта микро-ЭВМ.

4.2.2. Подпрограмма и стек

Так как объем памяти микро-ЭВМ ограничен, то целесообразно, как и в языках высокого уровня, однотипные операции объединять в подпрограммы. В системе команд МП есть ряд операторов, относящихся к области организации подпрограмм. Прежде всего это оператор CALL B2B3 - команда вызова подпрограммы, расположенной по адресу, помещенному во втором и третьем байтах B2 и B3, и оператор RET - возврат из подпрограммы. Однако в отличие от языков высокого уровня в ассемблере подпрограмма не имеет явно выраженных входных и выходных параметров. Эти параметры задаются в неявном виде, в основном через стек. Кроме того, при вызове подпрограммы в стек автоматически записывается адрес возврата, который затем при использовании оператора RET загружается из стека в счетчик ко-

манд. В данном разделе работа со стеком будет рассмотрена на конкретных примерах.

Первая подпрограмма, которая будет рассмотрена, является программой организаций временной задержки. Известно, что многие внешние устройства работают медленнее микропроцессора, поэтому он должен ждать когда они будут готовы. Среди способов организации такого ожидания существует процесс организации временной программной задержки. Известно, что каждая операция в МП занимает определенное число тактов синхронизирующего сигнала. Каждый такт длится примерно 0,5 мкс. Поэтому организовав цикл из нескольких операторов, можно по выходе из цикла получить требуемое время задержки. На рис.5 показана структурная схема подпрограммы временной задержки, в которой число проходов по кольцу до условного перехода определяется содержанием регистра В.

Общее время задержки определяется по формуле

$$T_D = t_1 + (t_2 + t_3 + t_4) N_1 + t_5,$$

где $t_{1,...,5}$ - времена выполнения соответствующих операций, указанных на рис.5, N_1 - число, первоначально записанное в регистре В. Подпрограмма DLY (программа 4.10) представляет собой программу временной задержки, записанную в соответствии с алгоритмом рис.5.

Программа 4.10

<u>Адрес</u>	<u>Код</u>	<u>Метка</u>	<u>Мнемоника</u>	<u>Комментарий</u>
0900	41		MOV B,C	; Запись числа из регистра С в регистр В
0901	00	DLY1:	NOP	; Пустая операция.
0902	05		DCR B	; Уменьшение содержимого В на 1.
0903	C20109		JNZ DLY1	; При не нулевом содержимом В идти на DLY1.
0906	C9		RET	; Возврат из подпрограммы.

Рассмотрим пример использования этой подпрограммы при организации звуковых сигналов в микро-ЭВМ. В ней звуковые сигналы формируются простейшей це-

пью в виде триггера, на вход поступают поочередно сигналы высокого и низкого уровня. Синхронизация триггера (вход С) осуществляется импульсами ЗПВВ, связанными по И с выходом схемы дешифрации адреса. Таким образом, это устройство работает как некоторый внешний порт с адресом 80. Алгоритм работы его чрезвычайно прост. Сначала в этот порт записывается 0, затем включается подпрограмма временной задержки, и затем в порт записывается 1. Фактически время задержки между записью 0 и 1 и определяет частоту звуковых колебаний. Изменять частоту можно изменением числа N1 (см.рис.5). Подпрограмма 3.1 реализует генерацию звуковых колебаний с частотой, задаваемой из основной программы MAIN (программа 4.12).

Программа 4.11

Адрес	Код	Метка	Мнемоника	Комментарии
0A00	AF	BPP:	XRA A	; Очистка аккумулятора.
0A01	D380		OUT BP	; Запись 00 в порт с адресом 80.
0A03	CD0009		CALL DLY	; Вызов подпрограммы временной задержки.
0A06	2F		CMA	; Инвертировать содержимое аккумулятора.
0A07	D380		OUT BP	; Записать FF в порт с адресом 80.
0A09	CD0009		CALL DLY	; Вызов подпрограммы временной задержки.
0A0C	C9		PET	; Возврат из подпрограммы.

Программа 4.12

Адрес	Код	Метка	Мнемоника	Комментарии
0800	DB20	MAIN:	IN 20	; Записать число N1 из входного регистра в аккумулятор.
0802	4F		MOV C,A	; Передать это число в регистр С.
0803	CD000A		CALL BPP	; Вызов подпрограммы звуковых колебаний.
0806	C3 0008		JMP MAIN	; Идти на начало.

В составе управляющей программы микро-ЭВМ имеется подпрограмма временной задержки, параметры которой задаются содержимым пары регистров BC. По адресу 0429 записана подпрограмма DEL A, осуществляющая фиксированную задержку на 1 мс и не имеющая входных параметров, а по адресу 0430 записана подпрограмма DEL B, которая осуществляет задержку на время, определяемое по содержимому пары регистров BC. Таким образом, если необходимо получить задержку в 1 мс, то нужно просто обратиться к подпрограмме DEL A, а если этой задержки не хватает, то необходимо вызвать подпрограмму DEL B, предварительно загрузив в пару регистров BC определенное, заранее рассчитанное число (программа 4.13)

Программа 4.13

Адрес	Код	Метка	Мнемоника	Комментарии
0429	C5	DELA:	PUSH B	; Сохранить содержимое BC в стеке.
042A	010100		LX1 B, 0001	; Установить длительность в 1 мс.
042D	C33104		JMP DEL1	; Переход в подпрограмму DEL B.
0430	C5	DELB:	PUSH B	; Сохранить содержимое B и C в стеке.
0431	F5	DEL1:	PUSH PSW	; Сохранить содержимое PSW в стеке.
0432	AF		XRA A	; Очистка аккумулятора.
0433	D5		PUSH D	; Сохранить содержимое D и E в стеке.
0434	1667	DEL2:	MVI D, TIME	; Загрузить счетчик 1 мс задержки.
0436	15	DEL3:	DCR D	; Уменьшить содержимое счетчика D на 1.
0437	C23604		JNZ DEL3	; Если не 0, то на DEL 3.
043A	0B		DCX B	; Уменьшить содержимое счетчика длительности B на 1.
043B	B8		CMP B	; Сравнение B с 0.
043C	C23404		JNZ DEL2	; Если не 0, то на DEL2.
043F	B9		CMP C	; Сравнение C с 0.
0440	C23404		JNZ DEL2	; Если не 0, то на DEL2.
0443	D1		POP D	; Восстановить содержимое регистров D и

				Е.
0444	F1		POP PSW	; То же, PSW.
0445	C1		POP B	; То же B и C.
0446	C9		RET	; Возврат из подпрограммы.

4.3. Задания для домашней подготовки

1. Ознакомиться с работой МБР К598ИР12 и БИС параллельного интерфейса К580ИК55.

2. Изучить способы обмена информацией между микро-ЭВМ и внешними устройствами. Ознакомиться с командами ввода-вывода МП КР580ВМ80А.

3. Изобразить схемы подключения устройств ввода-вывода к микро-ЭВМ, используемых в лабораторной работе.

4. Ознакомиться с группой логических и арифметических команд и команд условной передачи управления. Изучить принципы формирования отдельных разрядов регистра флажков.

5. Самостоятельно разработать программы: а) включения светодиодов выходного устройства, если число, записанное во входном устройстве, больше 3; б) включения светодиодов выходного устройства, если число, записанное во входном устройстве больше 3, но меньше 8.

6. Видоизменить программу 4.8 так, чтобы микро-ЭВМ реагировала на 0 в пятом разряде при записанных 1 во всех остальных разрядах.

7. Ознакомиться с группой команд МП вызова и возврата из подпрограммы.

8. Определить, при каких числах, записанных в регистре В, подпрограмма 4.10 будет осуществлять минимальное и максимальное время задержки. Найти эти времена, если машинный такт $T = 0.5$ мкс.

9. Разработать программу 5- и 10-секундной временной задержки. Составить программу, последовательно включающую светодиоды выходного устройства на время соответственно 10 и 5 с. Использовать разработанные ранее подпрограммы в п.9 задания.

10. Определить, для подпрограммы 4.13: при каких числах в паре регистров BC будет максимальное и минимальное время задержки, и каковы эти времена при длительности машинного такта в 0.5 мкс.

11. Разобрать программу 4.14, определяющую, какой из восьми переключателей входного устройства установлен в положение 0: а) составить алгоритм работы программы 4.14; б) установить, в каком регистре МП находится информация о номере переключателя входного устройства, который установлен в 0; в) определить, как будет работать программа 4.14, если на входном устройстве будут установлены в положение 0 несколько переключателей.

Программа 4.14

Адрес	Код	Метка	Мнемоника	Комментарии
0800	310009		LXI SP,0900	; Записать в указатель стека адрес 0900.
0803	DB20	WAITC:	IN 20	; Получить число из входного порта 20.
0805	FEFF		CPI FF	; Содержит ли какой-либо разряд число 0
0807	CA0308		JZ WAITC	; Если нет, то ждать на WAITC.
080A	CD1008		CALL INSW	; Если да, то вызов п/п определения разряда, в котором записан 0.
080D	CF		RST 1	; Окончить выполнение программы.
0810	06FF	INSW:	MVI B, FF	; Записать в B число FF.
0812	04	SRCH:	INR B	; Увеличить содержимое B на 1.
0813	0F		RRC	; Сдвиг содержимого A вправо.
0814	DA1206		JC SRCH	; Если C =1, то на SRCH.
0817	C9		RET	; Возврат из подпрограммы.

4.4. Задания к лабораторной работе

4.4.1. Исследовать программу 4.6

1. Ввести в микро-ЭВМ программу 4.6. Осуществить пуск программы.

2. Убедиться, что микро-ЭВМ постоянно переписывает данные с входного устройства в выходное. Для этого с помощью переключателей входного порта во время работы программы изменять числа, записанные в нем.

4.4.2. Исследовать программу 4.7.

1. Ввести в микро-ЭВМ программу 4.7. Осуществить пуск и исследовать результат по числу, записанному в выходное устройство.

2. Заменяя в программе двухбайтную команду ANI B2 на однобайтные ANA A, XRA A, ORA A, исследовать результат выполнения программы.

4.4.3. Исследовать программу 4.8.

1. Ввести в микро-ЭВМ программу 4.8 и осуществить ее пуск. Убедиться, что при ее выполнении микро-ЭВМ реагирует лишь на те числа во входном устройстве, которые содержат 1 в пятом разряде, а также, что после окончания выполнения программы в разряде Z регистре флажков содержится 1.

2. Исследовать видоизмененную программу 4.8, которая позволяет реагировать микро-ЭВМ только в случае присутствия 0 в пятом и всех единиц во всех остальных разрядах.

4.4.4. Исследовать программу 4.9.

1. Ввести в микро-ЭВМ программу 4.9. Осуществить пуск программы и убедиться, что при наличии 1 лишь во втором разряде числа входного регистра светодиоды выходного порта включены и микро-ЭВМ работает в цикле WAIT2 ожидания появления единицы в пятом разряде числа.

2. Записать 0 во второй разряд входного устройства, и 1 в пятый разряд. Убедиться, что светодиоды выходного порта гаснут и микро-ЭВМ находится в цикле WAIT1 программы.

3. Установить единицы во втором и пятом разрядах входного устройства и убедиться, что микро-ЭВМ последовательно выполняет оба цикла программы.

4.4.5. Исследовать программы, разработанные в п.5 задания для домашней подготовки, самостоятельно.

4.4.6. Исследовать процесс выполнения команд вызова и возврата из подпрограмм, а также команд работы со стеком.

1. Ввести в микро-ЭВМ подпрограмму 4.15.

2. Выполнить программу 4.15 по командам в режиме пошагового исполнения программ. После каждой команды проверять содержимое регистров МП.

3. Выполнить команды CALL STDY, PLLS H PSW, POP, RET по машинным циклам и построить временные диаграммы их выполнения.

4. Заменить в программе 4.15 команду POP PSW на команду NOP (00) и проследить, как будет выполняться программа 4.15. Объяснить происшедшие изменения.

Программа 4.15.

Адрес	Код	Метка	Мнемоника	Комментарии
0800	31C00B		LXI SP,0BC0	; Настройка указателя стека.
0803	CD0708		CALL STDY	; Вызов п/п STDY.
0806	CF		RST I	; Возврат к управляющей программе.
0807	F5	STDY:	PUSH PSW	; Запись в стек содержимого всех регистров МП.
0808	C5		PUSH B	
0809	D5		PUSH D	
080A	E5		PUSH H	
080B	3E05		MVI A, 05	; Запись в рег. А числа 05.
080D	47		MOV B, A	; Записать в рег. В содержимое рег. А.
080E	87		ADD A	; Удвоить содержимое рег. А.
080F	5F		MOV E, A	; Записать в рег. Е содержимое рег. А.
0810	67		MOV H,A	; Записать в рег. H содержимое рег. А.
0811	E1		POP H	; Возврат из стека содержимого всех регистров МП.
0812	D1		POP D	
0813	C1		POP B	
0814	F1		POP PSW	

0815	C9		RET	; Возврат из подпрограммы.
------	----	--	-----	----------------------------

4.4.7. Исследование программы временной задержки (программа 4.12).

1. Записать полный текст программы, 4.12 для генерации звуковых сигналов программ 4.10 и 4.11. Ввести в микро-ЭВМ программу.
2. Установить на входном устройстве число 00.
3. Осуществить пуск программы с адреса 0800.
4. Проследить за изменением тона звука, формируемого микро-ЭВМ в процессе выполнения программы, увеличивая число, записанное во входном устройстве.

4.4.8. Исследовать программу регулируемой временной задержки на примере программы, последовательно включающей и выключающей светодиода входного устройства на 10 и 5с соответственно.

1. Ввести в микро-ЭВМ программу, разработанную в п.9 домашнего задания.
2. Осуществить пуск программы и проверить ее функционирование.
3. Изменить числа, записанные в регистрах В и С в подпрограмме регулируемой временной задержки (программа 4.13). Проверить возможность изменения длительности задержки.

4.4.9. Исследование программы 4.14.

1. Ввести программу 4.14 в микро-ЭВМ.
2. Установить во входном устройстве переключателя соответственно числу FF. Осуществить пуск программы. Убедиться, что микро-ЭВМ будет находиться в режиме ожидания появления 0 в любом разряде.
3. Установить в каком-либо разряде входного устройства 0 и проверить содержимое регистров МП после окончания выполнения программы.
4. Осуществить повторный запуск программы при наличии нулей в двух разрядах входного устройства. Какое число будет записано в соответствующем регистре МП после окончания выполнения программы ?

4.5. Содержание отчета

1. Схемы подключения внешних устройств к микро-ЭВМ.

2. Временные диаграммы процесса выполнения команд ввода-вывода данных.

3. Самостоятельно разработанные в процессе выполнения лабораторной работы, указанные в п.5 задания для домашней подготовки.

4. Полный перечень команд условных переходов, логических операций, вызова и возврата из подпрограмм МП КР580ВМ80А.

5. Ответы на вопросы, содержащиеся в п. 9, 10 задания для домашней подготовки.

6. Разработанные в процессе подготовки программы 5- и 10-секундной задержки.

7. Перечень команд работы со стеком.

8. Ответы на вопросы, поставленные в п. 11 задания для домашней подготовки.

4.6. Задания для самопроверки

1. С помощью, каких команд микро-ЭВМ осуществляет ввод-вывод данных?

2. Привести схемы дешифраторов адреса и внешних устройств, при организации обмена информацией с внешними устройствами с помощью различных команд.

3. При выполнении каких команд, приведенных в программе 4.7, задействуются разряды регистра флажков? По каким условиям записывается 1 в каждый разряд этого регистра?

4. Перечислите виды логических операций, выполняемых в МП.

5. Перечислить режимы работы программируемого параллельного интерфейса КР580ИК55.

6. Рассмотреть возможные способы организации обмена информацией между двумя микро-ЭВМ в параллельном коде с помощью схемы КР580ИК55.

7. В какой последовательности записывается и считывается из стека содержимое аккумулятора и регистра флажков при выполнении команд PUSH и PSW и POP PSW?

8. С помощью каких команд можно задать или переобозначить область памяти, отведенную под стек?

9. Указать порядок выполнения команд CALL, RET, RST.

10. Как нужно изменить программу 4.13, чтобы временная задержка определялась числами, записанными по адресам 0B00 и 0B01?

11. Какой минимальный адрес будет записан в указатель стека в процессе выполнения программы 4.15?

12. Как будет выполняться программа 4.15, если вместо команды POP B в ней будет записана команда NOP?

13. Какое максимальное время задержки может обеспечить программа 4.10, если длительность машинного цикла $T=1$ мкс?

14. Какое максимальное и минимальное время задержки обеспечивает программа 4.13, если длительность машинного цикла $T=0,5$ мкс?

15. Как увеличить время задержки при использовании программы 4.18?

16. Какие команды в системе команд МП КР580ВМ80А обеспечивают формирование признаков операций? В чем состоит особенность использования каждой такой команды для организации условных переходов.

17. Какими способами можно передать некоторые параметры в подпрограмму при использовании операторов CALL? В чем достоинства и недостатки этих способов?

18. Каким образом можно сохранить содержимое всех внутренних регистров МП при вызове подпрограммы, в которой эти регистры используются для других целей? Какова при этом будет структура подпрограммы?

19. Какие внешние устройства используются в микро-ЭВМ? Какие способы обмена с ними вы знаете? Сравните их, укажите достоинства и недостатки.

20. Какие способы передачи информации используются для обмена между МПС и внешними устройствами, если: а) внешнее устройство имеет скорость передачи информации значительно ниже, чем быстродействие МП; б) внешнее устройство имеет скорость передачи информации, выше, чем скорость приема данных командами IN и OUT в МПС?

21. Составить структурную схему программы, которая осуществляет ввод с клавиатуры кодов символов и переход на выполнение определенных частей про-

граммы в соответствии с кодом принимаемого символа. Какими двумя способами можно осуществить такой переход? Какой из двух способов вы выберете, если принимаемых символов достаточно много?

Список используемой литературы

1. Горбунов В.Л. и др. Микропроцессоры. Основы построения микро-ЭВМ. М.: Высшая школа, 1984
2. Горбунов В.Л., Панфилов Д.И. Микропроцессоры. Лабораторный практикум. - М.: Высшая школа, 1984
3. Балашов Е.П., Пузанков Д.В. Микропроцессоры и микропроцессорные системы. - М.: Радио и связь, 1981
4. Алексеенко А.Г. и др. Проектирование радиоэлектронной аппаратуры на микропроцессорах. - М.: Радио и связь, 1984
5. Коффон Дж. Технические средства микропроцессорных систем. - М.: Мир, 1983
6. Соучек Б. Микропроцессоры и микро-ЭВМ. М.: Советское радио, 1978