ЛАБОРАТОРНАЯ РАБОТА №1

**Тема: *НАЗНАЧЕНИЕ МП. ОСНОВНОЙ ЦИКЛ РАБОТЫ МП ФОН-НЕЙМАНОВСКОЙ АРХИТЕКТУРЫ.***

**Цель работы: *Получить представление о процессах происходящих в ЭВМ с фон-Неймановской архитектурой во время выполнения программы. Познакомиться с понятием «машинный цикл».***

**Теоретические сведения и пример, иллюстрирующий работу микроЭВМ:**

Архитектура типичной небольшой вычислительной системы на основе микроЭВМ содержит все **5 основных блоков цифровой машины**: устройство ввода информации, управляющее устройство (УУ), арифметико-логическое устройство (АЛУ) (входящие в состав микропроцессора), запоминающие устройства (ЗУ) и устройство вывода информации.

**Микропроцессор** координирует работу всех устройств цифровой системы с помощью **шины управления** (ШУ). Помимо ШУ имеется **адресная шина** (ША), которая служит для выбора определенной ячейки памяти, порта ввода или порта вывода. По информационной шине или **шине данных** (ШД) осуществляется двунаправленная пересылка данных к микропроцессору и от микропроцессора. Важно отметить, что МП может посылать информацию в память микроЭВМ или к одному из портов вывода, а также получать информацию из памяти или от одного из портов ввода.

Постоянное запоминающее устройство (ПЗУ) в микроЭВМ содержит программу инициализации ЭВМ. Программы пользователя могут быть загружены в запоминающее устройство с произвольной выборкой (ОЗУ) из внешнего запоминающего устройства (ВЗУ).

**Основные характеристики микропроцессора**

Микропроцессор характеризуется:

1) ***тактовой частотой***, определяющей максимальное время выполнения переключения элементов в ЭВМ;

2) ***разрядностью***, т.е. максимальным числом одновременно обрабатываемых двоичных разрядов.

**Разрядность МП** обозначается m/n/k/ и включает:

m - разрядность внутренних регистров, определяет принадлежность к тому или иному классу процессоров;

n - разрядность шины данных, определяет скорость передачи информации;

k - разрядность шины адреса, определяет размер адресного пространства. Например, МП i8088 характеризуется значениями m/n/k=16/8/20;

3) ***архитектурой***. Понятие архитектуры микропроцессора включает в себя систему команд и способы адресации, возможность совмещения выполнения команд во времени, наличие дополнительных устройств в составе микропроцессора, принципы и режимы его работы. Выделяют понятия микроархитектуры и макроархитектуры.

**Микроархитектура микропроцессора** - это аппаратная организация и логическая структура микропроцессора, регистры, управляющие схемы, арифметико-логические устройства, запоминающие устройства и связывающие их информационные магистрали.

**Макроархитектура** - это система команд, типы обрабатываемых данных, режимы адресации и принципы работы микропроцессора.

В общем случае под **архитектурой ЭВМ** понимается абстрактное представление машины в терминах основных функциональных модулей, языка ЭВМ, структуры данных.

**В качестве примера, иллюстрирующего работу микроЭВМ,** рассмотрим процедуру, для реализации которой нужно выполнить следующую последовательность элементарных операций:

1. Нажать клавишу с буквой "А" на клавиатуре.

2. Поместить букву "А" в память микроЭВМ.

3. Вывести букву "А" на экран дисплея.

Это типичная процедура ввода-запоминания-вывода, рассмотрение которой дает возможность пояснить принципы использования некоторых устройств, входящих в микроЭВМ.

На рис. 2 приведена подробная диаграмма выполнения процедуры ввода-запоминания-вывода. Обратите внимание, что команды уже загружены в первые шесть ячеек памяти. Хранимая программа содержит следующую цепочку команд:

1. Ввести данные из порта ввода 1.
2. Запомнить данные в ячейке памяти 200.
3. Переслать данные в порт вывода 10.

В МП на рис. 1 выделены еще два новых блока - регистры: **аккумулятор** и **регистр команд**.

Рассмотрим прохождение команд и данных внутри микроЭВМ с помощью номеров на диаграмме. Напомним, что **микропроцессор - это центральный узел, управляющий перемещением всех данных и выполнением операций**.

Итак, при выполнении типичной процедуры ввода-запоминания-вывода в микроЭВМ происходит следующая последовательность действий:

1. МП выдает адрес **100** на шину адреса. По шине управления поступает сигнал, устанавливающий память программ (конкретную микросхему) в режим считывания.

2. ЗУ программ пересылает первую команду ("Ввести данные") по шине данных, и МП получает это закодированное сообщение. Команда помещается в регистр команд. МП декодирует (интерпретирует) полученную команду и определяет, что команда имеет продолжение – в следующем байте памяти хранится указание на то, откуда осуществляется ввод данных.

3. МП выдает адрес **101** на ША; ШУ используется для перевода памяти программ в режим считывания.

4. Из памяти программ на ШД пересылается операнд "Из порта 1". Этот операнд находится в программной памяти в ячейке **101**. Код операнда (содержащий адрес порта 1) передается по ШД к МП и направляется в регистр команд. МП теперь декодирует полную команду ("Ввести данные из порта 1").

5. МП, используя ША и ШУ, связывающие его с устройством ввода, открывает **порт 1**. Цифровой код буквы "А" передается в аккумулятор внутри МП и запоминается. Важно отметить, что при обработке каждой программной команды МП действует согласно микропроцедуре **выборки-декодирования- исполнения.**

6. МП обращается к ячейке **102** по ША. ШУ используется для перевода памяти программ в режим считывания.

7. Код команды "Запомнить данные" подается на ШД и пересылается в МП, где помещается в регистр команд.

8. МП дешифрирует эту команду и определяет, что команда имеет продолжение – в следующем байте памяти хранится указание на то, по какому адресу следует записать данные. МП обращается к ячейке памяти **103** и приводит в активное состояние вход считывания микросхем памяти программ.

9. Из памяти программ на ШД пересылается код сообщения "В ячейке памяти 200". МП воспринимает этот операнд и помещает его в регистр команд. Полная команда "Запомнить данные в ячейке памяти 200" выбрана из памяти программ и декодирована.

10. Теперь начинается процесс выполнения команды. МП пересылает адрес 200 на ША и активизирует вход записи, относящийся к памяти данных.

11. МП направляет хранящуюся в аккумуляторе информацию в память данных. Код буквы "А" передается по ШД и записывается в ячейку 200 этой памяти. Выполнена вторая команда. Процесс запоминания не разрушает содержимого аккумулятора. В нем по-прежнему находится код буквы "А".

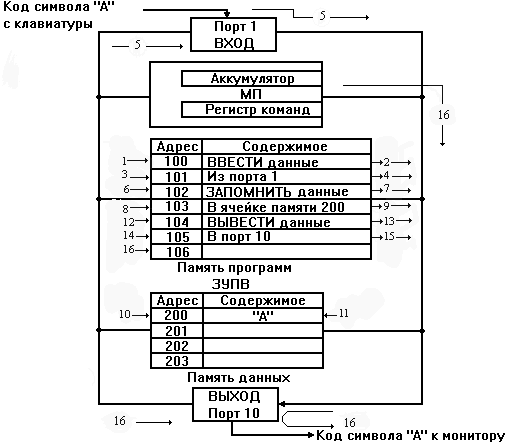
12. МП обращается к ячейке памяти 104 для выбора очередной команды и переводит память программ в режим считывания.

13. Код команды вывода данных пересылается по ШД к МП, который помещает ее в регистр команд, дешифрирует и определяет, что команда имеет продолжение.

14. МП выдает адрес 105 на ША и устанавливает память программ в режим считывания.

15. Из памяти программ по ШД к МП поступает код команды "В порт 10", который далее помещается в регистр команд.

16. МП дешифрирует полную команду "Вывести данные в порт 10". С помощью ША и ШУ, связывающих его с устройством вывода, МП открывает порт 10, пересылает код буквы "А" (все еще находящийся в аккумуляторе) по ШД. Буква "А" выводится через порт 10 на экран дисплея.



*Рис. 1. Диаграмма выполнения процедуры ввода-запоминания-вывода.*

В большинстве микропроцессорных систем (МПС) передача информации осуществляется способом, аналогичным рассмотренному выше. Наиболее существенные различия возможны в блоках ввода и вывода информации.

Итак, **центральным блоком ЭВМ является микропроцессор** – устройство, непосредственно осуществляющее преобразование информации. Все блоки соединены между собой с помощью шины. На микропроцессор возложены две основные функции: обработка данных и управление машиной. Для выполнения этих важных функций в структурной схеме микропроцессора три основных устройства: **АЛУ**, **УУ** и набор **регистров**.

**Работа микропроцессора представляет последовательную реализацию микропроцедур выборки-дешифрации-исполнения.** Фактическая последовательность операций в МПС определяется командами, записанными в памяти программ.

Таким образом, в МПС **микропроцессор выполняет следующие функции**:

***- выборку команд программы из основной памяти;***

***- дешифрацию команд;***

***- выполнение арифметических, логических и других операций, закодированных в командах;***

***- управление пересылкой информации между регистрами и основной памятью, между устройствами ввода/вывода;***

***- отработку сигналов от устройств ввода/вывода, в том числе реализацию прерываний с этих устройств;***

***- управление и координацию работы основных узлов МП.***

**Логическая структура микропроцессора**, т. е. конфигурация составляющих микропроцессор логических схем и связей между ними, определяется функциональным назначением и фактически определяет его архитектуру. Срабатывание электронных блоков микропроцессора в определенной последовательности приводит к выполнению заданных архитектурой микропроцессора функций, т. е. к реализации вычислительных алгоритмов.



*Рис 2. Схема центрального микропроцессора.*

Изображенная на рисунке структура работает следующим образом:

1. Адрес, содержащийся в счетчике команд, через регистр адреса передается в память и оттуда считывается команда, помещаемая в регистр команд.

2.Счетчик команд наращивается на длину выбранной команды, чтобы указывать на команду, расположенную в памяти вслед за выбранной.

3.Устройство управления дешифрует код команды, содержащейся в регистре команд. Если выбранная команда является безадресной, то проверяется, не является ли она командой останова. Если она является таковой, то на этом цикл заканчивается.

Если команда не является командой останова, товыполняются действия, предписываемые кодом операции команды, после чего следует возврат к п.1.

Если выбранная команда является командой перехода, то проверяется, выполняются ли условия перехода. Если не выполняются, то следует возврат к п.1. Если выполняются, то адрес перехода из адресной части команды перехода, находящейся в регистре команд, модифицирует содержимое счетчика команд так, чтобы он указывал адрес перехода, после чего осуществляется возврат к п. 1.

Если выбранная команда не является командой перехода, то проверяется, не является ли она командой пересылки. Если она является таковой, то содержимое аккумулятора записывается по адресу, извлекаемому из адресной части команды, находящейся в регистре команд, после чего следует возврат к п.1.

Если выбранная команда не является командой пересылки, то из ячейки памяти, адрес которой берется из адресной части команды, находящейся в регистре команд, считывается операнд. Он вместе с содержимым аккумулятора передается в АЛУ, где выполняются действия, предписываемые кодом операции команды, после чего следует возврат к п.1.

Из описания работы операционного устройства можно заметить, что процесс выполнения команды сводится к определенной последовательности открывания и закрывания **вентильных схем** Описание того, какую вентильную схему и когда открывать, представляет собой **программу выполнения команды**. Эти программы принято называть **микропрограммами** команд, а команды, образующие микропрограммы - **микрокомандами**. Простейшая микрокоманда состоит из набора битов, каждый из которых управляет одной из вентильных схем операционного устройства. Действие, выполняемое под управлением каждого из этих разрядов микрокоманды, называется **микрооперацией**.

Итак***, процесс выполнения программы в ЭВМ представляет собой последовательность командных циклов.***

Каждая команда в процессоре выполняется на протяжении командного цикла. **Командный цикл** состоит из:

- цикла выборки команды;

- цикла дешифрации команды;

- цикла выполнения команды.

Продолжительность цикла выборки команды зависит от формата команды (количества байтов в машинном коде команды). Продолжительность цикла выполнения команды зависит от способа адресации операндов. Например, при выполнении команд с регистровой адресацией не нужно дополнительное обращение к памяти для чтения операнда, а в командах с косвенной адресацией такое обращение необходимо. Поэтому ***продолжительность командного цикла в процессоре различна для различных команд и определяется количеством обращений к памяти или внешнему устройству***.

Интервал, на протяжении которого осуществляется ***одно обращение процессора к памяти или к внешнему устройству, называется* машинным циклом**.

**Командный цикл** микропроцессора состоит из некоторого количества машинных циклов (в зависимости от типа команды). В команде может быть от ***одного до пяти***машинных циклов.

Машинный цикл, в свою очередь, разбивается на некоторое количество машинных тактов. В течение каждого **машинного такта** выполняется элементарное действие (микрооперация) в микропроцессоре. ***Количество тактов*** в цикле определяется кодом команды и равно от *3* до *5*. Продолжительность такта задается периодом импульсов синхронизации, подаваемых от тактового генератора. ***Все внутренние операции микропроцессора и формирование внешних сигналов происходят в моменты времени, определяемые тактовыми импульсами.***

Таким образом, ***командный цикл процессора состоит из некоторого количества машинных циклов, а каждый машинный цикл – из определенного количества тактов.***

В зависимости от действий, выполняемых процессором, различают следующие типы машинных циклов:

1. выборка команды;
2. чтение памяти;
3. запись в память;
4. чтение стека;
5. запись в стек;
6. ввод данных из внешнего устройства;
7. вывод данных на внешнее устройство;
8. прерывание.

Первым машинным циклом команды всегда является цикл ***выборка команды***, на протяжении которого осуществляется выборка из памяти байтов кода команды по адресу, определяемому содержимым счетчика команд. Далее для выборки операндов при выполнении команд с прямой и косвенной адресацией выполняется цикл ***чтение памяти***. Для записи операндов или сохранения адресов при выполнении команд пересылок нужны машинные циклы ***запись в память***.

В командах с обращением к стеку выполняются циклы ***чтение стека*** и ***запись в стек***. Адрес памяти определяется регистром указателем стека. Для реализации команд ввода/вывода выполняются машинные циклы ***ввод*** и ***вывод*,** для организации прерываний в программе – циклы ***прерывание***.

**Пример.** Этапы командного цикла при выполнении арифметической команды:

1. выборка команды из памяти;
2. дешифрация кода операции (КОП) команды;
3. формирование адреса следующей команды;
4. формирование адресов операндов;
5. выборка операндов;
6. выполнение операции, заданной командой.

***Рассмотрим пример выполнения во времени команды непосредственного сложения (ADD) содержимого ЯП и регистра-аккумулятора:***

*1-ый машинный цикл М1*: **выборка** команды из памяти. Занимает 4 машинных такта Т1, Т2, Т3, Т4. В течение периода Т4 машинного цикла МП **декодирует** команду сложения и решает, что для ее завершения ему надо два дополнительных цикла;

*2-ой машинный цикл М2*: **считывается** следующий байт из памяти, которым является операнд (занимает 3 машинных такта Т1, Т2, Т3);

*3-ий машинный цикл М3*: **выполнение** операции ADD в АЛУ (занимает 2 машинных такта Т1, Т2).

|  |  |  |
| --- | --- | --- |
| ***Цикл команды*** | | |
| **М1** | **М2** | **М3** |
| **Т1, Т2, Т3, Т4** | **Т1, Т2, Т3** | **Т1, Т2** |
| *Извлечение КОП* | *Считывание из памяти* | *Сложение* |

**М – машинные циклы**

**Т – машинные такты**

Следует заметить, что машинные циклы занимают не одинаковое время, но не больше, чем цикл команды.

Подведем итог сказанному:

**АЛУ** выполняет первую функцию микропроцессора – ***обработку данных***, их вычисление и перемещение. Через два входа в АЛУ поступают данные, результат передается дальше через единственный выход. Практически все операции над двоичными числами сводятся к двум: сложение и сдвиг.

Вторая функция микропроцессора – ***управление машиной***, или выполнение программы, обеспечивается **УУ** с помощью **дешифратора команд**, преобразующего коды операторов программы в сигналы, поступающие на все узлы микропроцессора. Коды команд программы дают информацию о том, какие операции надо выполнить, где в памяти расположены данные, куда надо направить результат и где расположена следующая за выполняемой команда.

Кроме команд программы на УУ поступают от генератора тактовые импульсы, синхронизирующие работу микропроцессора. Эта синхронизация заключается в выполнении микропроцессором двух непрерывно повторяющихся шагов: считывание очередной команды (***цикл выборки***) и выполнения операции, предписываемой этой командой (***цикл выполнения***). Совокупность этих двух шагов образует **командный цикл**. УУ выполняет и некоторые другие функции, например, управляет внутренней шиной, т.е. определяет, какое устройство ЭВМ в данный момент времени может ею пользоваться, так как к ней подключено множество устройств и необходим определенный порядок приема и передачи данных.

**Устройство управления** в строгой последовательности в рамках тактовых и цикловых временных интервалов работы микропроцессора (**такт** - минимальный рабочий интервал, в течение которого совершается одно элементарное действие; **цикл** - интервал времени, в течение которого выполняется одна машинная операция) осуществляет:

* выборку команды;
* интерпретацию ее с целью анализа формата, служебных признаков и вычисления адреса операнда (операндов);
* установление номенклатуры и временной последовательности всех функциональных управляющих сигналов;
* генерацию управляющих импульсов и передачу их на управляющие шины функциональных частей микроЭВМ;
* анализ результата операции и изменение своего состояния так, чтобы определить месторасположение (адрес) следующей команды.

Итак**, командный цикл** в общем случае включает:

1. Извлечение команды из памяти (IF - Instruction Fetch).
2. Декодирование команды (Instruction Decoding – ID).
3. Извлечение операндов из памяти или из регистров (MEM).
4. Выполнение (Execute - EX).
5. Запись результатов в память или регистр (Write Back – WB).

**ПРАКТИЧЕСКОЕ ЗАДАНИЕ № 1.**

1. Написать программу.

2. Разместить ее в ячейках памяти.

3. Определить количество командных циклов, которые должен выполнить процессор для решения задачи. Дать их устное описание.

***Необходимые условия:***

* последней командой каждой программы должна быть команда ОСТАНОВ; обрабатываемые значения необходимо ввести с клавиатуры (порт 1);
* результаты работы вывести на экран (порт 10);
* информация об операции занимает 1 байт памяти;
* информация о местоположении операндов и результата также занимает один байт;
* данные с клавиатуры заносить в регистры общего назначения РОНы) Р1 – Р8, результат получать в регистре аккумуляторе – РА.

Блок управления работает в режиме постоянного повторения алгоритма, называемого **машинным циклом: *выборка, декодирование, выполнение***.

1. На этапе **выборки** УУ извлекает из основной памяти ту команду, которая должна выполняться следующей. Ее адрес находится в регистре ***счетчике команд*** (СК). Выбранная команда помещается в ***регистр команд*** УУ, а значение в регистре СК автоматически увеличивается на длину выбранной команды и содержит адрес следующей команды.
2. На этапе **декодирования** происходит определение выполняемой операции и в зависимости от нее выделяются поля операндов.
3. На этапе **выполнения** УУ активизирует схему, предназначенную для выполнения поставленной задачи.

Для координации действий, выполняемых на протяжении машинного цикла, необходимо обеспечить синхронизацию работы различных схем машины. С этой целью на соответствующие электронные схемы подается импульсный сигнал, называемый **сигналом синхронизации**. Амплитуда этого сигнала изменяется между уровнями 0 и 1, а различные схемы машины разрабатываются таким образом, чтобы они приводились в действие тем или иным фронтом импульса синхронизирующего сигнала. В результате тактовая частота этого сигнала фактически определяет ту скорость, с которой центральный процессор выполняет свой машинный цикл.

**\***- умножение (считаем простой командой)

**/** - деление (считаем простой командой)

|  |  |
| --- | --- |
| ***№ вар.*** | ***Задание*** |
| 1 | Y=a/b – c3 |
| 2 | Y=c\*d + 5\*a |
| 3 | Y=a\*b2\*c |
| 4 | Y=(a2 + b2)/2 |
| 5 | Y=(a-b)\*c-7 |
| 6 | Y=a/(c-b)2 |
| 7 | Y=d-b3/(a-1) |
| 8 | Y=(a+b+c)/3\*a |
| 9 | Y=4\*a + 2\*c/b2 |
| 10 | Y=(a-b)/(a+b) |
| ***№ вар.*** | ***Задание*** |
| 11 | Y= a2/c-23 |
| 12 | Y=a\*b+b\*c |
| 13 | Y=(a+b+c+d)\*5\*a |
| 14 | Y=(a-b)2-2\*d |
| 15 | Y=a/b/c-5 |
| 16 | Y=2\*c-d+23 |
| 17 | Y=2\*b-a+b/c |
| 18 | Y=4\*b/a+1 |
| 19 | Y=(4\*a-1)/b\*c |
| 20 | Y=25/c\*(b+a\*a) |
| ***№ вар.*** | ***Задание*** |
| 21 | Y=(c/4+28\*d)/3\*d |
| 22 | Y=(b/2-53/c)\*b2 |
| 23 | Y=(4\*a/b+1)/c-b |
| 24 | Y=a4-(b+c) |
| 25 | Y=a\*c+b\*c+d\*c |
| 26 | Y=(7\*a+3/b)2 |
| 27 | Y=(a-b\*4-1)/c/31 |
| 28 | Y=2\*c/a-d\*d |
| 29 | Y=(c-d\*23)/2\*a\*a |
| 30 | Y=(4/c+3\*a)/a\*b |

**Пример:** Описать командные циклы, которые должен выполнить процессор для решения следующей задачи: Y=A+B+C.

Пишем программу:

1. Ввод в регистр Р1 числа А из порта 1.
2. Ввод в регистр Р2 числа В из порта 1.
3. Ввод в регистр Р3 числа C из порта 1.
4. Сложение содержимого Р1, Р2, Р3 с записью результата в регистр P4.
5. Вывод результата в порт 10.
6. ОСТАНОВ

Размещаем ее в ОЗУ:

|  |  |
| --- | --- |
| ***адрес*** | ***содержимое*** |
| 100 | Ввести данные |
| 101 | Из порта1 в Р1 |
| 102 | Ввести данные |
| 103 | Из порта1 в Р2 |
| 104 | Ввести данные |
| 105 | Из порта1 в Р3 |
| 106 | Сложить данные |
| 107 | Из Р1, Р2 в PА |
| 108 | Сложить данные |
| 109 | Из Р3, РА в PА |
| 10А | Вывести данные |
| 10В | Из PА в порт10 |
| 10С | ОСТАНОВ |

Количество выполняемых циклов: 7

Прокомментируйте каждый цикл, выполняемый процессором.

**КОНТРОЛЬНЫЕ ВОПРОСЫ.**

1. Объяснить назначение процессора.
2. Объяснить назначение АЛУ.
3. Объяснить назначение УУ.
4. Раскрыть понятие «Машинный цикл».
5. Раскрыть понятие «Машинный такт».
6. Описать логический состав МП.
7. Назвать основные блоки ЭВМ фон-Неймановской архитектуры.
8. Объяснить суть принципа программного управления.
9. Раскрыть принцип хранимой в памяти программы.
10. Объяснить суть адресного принципа.
11. Объяснить назначение регистров.
12. Назвать основные функциональные регистры процессора.