Национальный Технический Университет Украины

“Киевский Политехнический Институт”

Факультет Информатики и Вычислительной Техники

Кафедра вычислительной техники

Расчетно-графическая работа

по дисциплине “Архитектура компьютера”

Выполнил:

студент 3-го курса

группы ИВ-21

Кузьменко В.З.

№ зач. кн. 2114

1000010000102

Руководитель:

Ткаченко В.В.

Киев – 2014

Содержание

[Введение 3](#_Toc405142010)

[1. Обзор МПС 4](#_Toc405142011)

[2. Архитектура МПС 5](#_Toc405142012)

[2.1 Память программ 5](#_Toc405142013)

[2.2 Память данных 6](#_Toc405142014)

[2.3 Внешнее устройство 6](#_Toc405142015)

[2.4 Режим прерываний (КПП) 7](#_Toc405142016)

[2.5 Прямой доступ к памяти (КПДП) 8](#_Toc405142017)

[3. Система команд 9](#_Toc405142018)

[3.1 Основные команды 9](#_Toc405142019)

[3.1.2 Команды передачи управления. 11](#_Toc405142020)

[3.1.3 Команды управления режимом работы МК 13](#_Toc405142021)

[4. Выполнение индивидуального задания 14](#_Toc405142022)

[4.1 Алгоритм программы 14](#_Toc405142023)

[4.2. Карта распределения памяти внешних устройств. 14](#_Toc405142024)

[4.3. Блок-схема алгоритма вычисления выражения 15](#_Toc405142025)

[4.4 Код микропрограммы 16](#_Toc405142026)

[5. Структурная схема системы. 17](#_Toc405142027)

[6. Выводы 17](#_Toc405142028)

[7. Список литературы. 17](#_Toc405142029)

Введение

Микропроцессоры (МП) представляют собой автономные функционально законченные устройства, состоящие из одной или нескольких программно-управляемых интегральных микросхем высокой степени интеграции, включающие все средства, необходимые для обработки информации и управления данными, и рассчитанные на совместную работу с устройствами памяти и ввода-вывода информации.

Структура микропроцессора должна удовлетворять трем основным требованиям: быть функционально гибкой, обеспечить достаточно высокое быстродействие и допускать недорогую технологическую реализацию. Высокая функциональная гибкость микропроцессора, необходимая для создания эффективного программного обеспечения, достигается микропрограммным управлением, за счет побайтовой обработки и адресации данных, развитой системы прерываний и большего числа внутренних регистров.

1. Обзор МПС

Микропроцессорная система (МПС) представляет собой функционально законченное изделие, состоящее из одного или нескольких устройств, главным образом микропроцессорных: микропроцессора и/или микроконтроллера.

В микроЭВМ шины адреса и данных объединены в ШАД. Разрядность ШАД - 32 бита, что обеспечивает адресацию 8МБ оперативной памяти и работу с 32-битными данными. Для работы с ШАД имеются интерфейсные регистры РА и РД. Регистр РА может выдавать на ШАД свою младшую, либо старшую часть.

Обмен данных между внешними устройствами и оперативной памятью осуществляется через КПДП. Обработка векторных прерываний от ВУ выполняется КПП. КПДП и КПП совмещены и распределены по интерфейсам внешних устройств.

Микроконтроллер состоит из следующих узлов:

* однокристальной ЭВМ со схемой внешнего тактового генератора и схемой формирования сигнала «сброс»;
* регистра-защелки младшего байта адреса внешнего запоминающего устройства;
* памяти программ, объемом 4 Кбайта;
* памяти данных, объемом 1 Кбайт со страничной адресацией 256 байт на страницу и схемой выбора ОЗУ;
* схемы управления записью-чтением внешних устройств;
* адаптера параллельного интерфейса со схемами приемника и передатчика по стандарту ИРПС;
* трехканального таймера;
* контроллера клавиатуры и индикации;
* схемы прерываний.



Рисунок 1.1 - Структурная схема МК48

# 2. Архитектура МПС

Разрабатывается микропроцессорная система (МПС), ядром которой является процесор 1816ВЕ48.

В состав разрабатываемой МПС должны входить процессор (П), основная память (ОП), содержащая ОЗУ и ПЗУ, а также внешние устройства (ВУ), контроллеры прерываний и прямого доступа к памяти.

Микропроцессор имеет 16-разрядную операционную часть. Объем внешней памяти программ 4К, внешней памяти данных – 64К. Шина адреса и данных объединённая, КПП и КПДП - централизованные. Каждое ВУ имеет фиксированный адрес в адресном пространстве периферийных устройств.

2.1 Память программ

Память программ реализована в резидентном СППЗУ емкостью 1 Кбайт. Максимальное адресное пространство, отводимое для программ, составляет 4 Кбайт. Карта адресов памяти программ показана на рисунке (2.1.1).

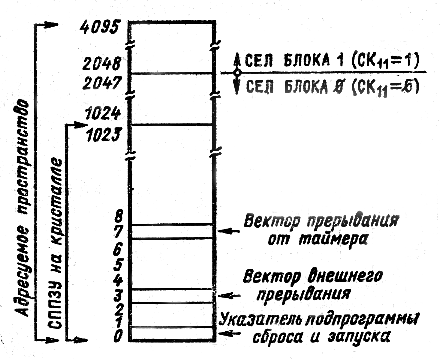


Рис 2.1.1 Карта адресов памяти программ

2.2 Память данных

Память данных- резидентная память данных емкостью 64 байта. Имеет в своем составе два банка рабочих регистров 0-7 и 24-31 по восемь регистров в каждом. Структура регистров (карта адресов) памяти данных МК представлена на следующем рисунке.

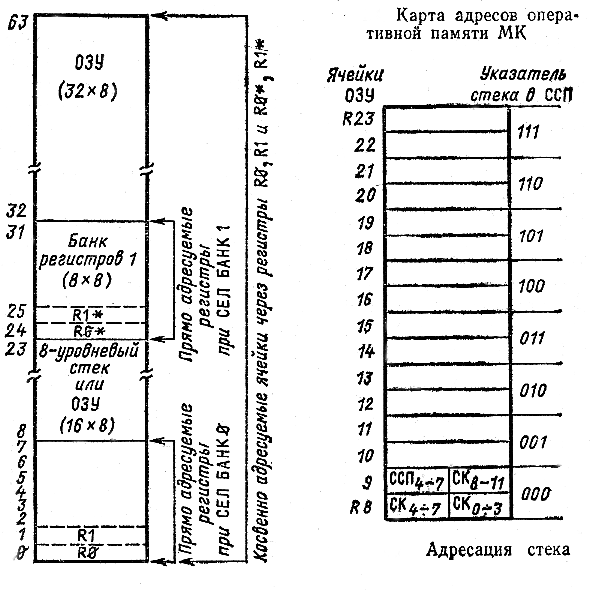


Рис. 2.2.1 Карта адресов памяти данных

Выбор одного из двух банков выполняется по команде "SEL RB". Рабочие регистры доступны командам с прямой адресацией, а все ячейки ОЗУ доступны по командам с косвенной адресацией. В качестве регистров косвенного адреса используются регистры R0, R1, R0\*, R1\*

Практически все команды с обращением к ОЗУ оперируют с одним байтом. Однако по командам вызова и возврата осуществляется доступ к двухбайтным словам. В памяти данных слова хранятся так, что старший байт слова располагается в ячейке с большим адресом. Необходимо отметить, что в памяти программ порядок расположения байт по старшинству при хранении двухбайтных слов обратный.

## 2.3 Внешнее устройство

ВУ имеет фиксированный адрес в адресном пространстве периферийных устройств. Помимо регистра данных и регистра состояния каждое ВУ содержит регистр режима, регистр начального адреса ОП, регистр-счетчик. Эти регистры обеспечивают прямой доступ ВУ к памяти. Доступ к этим регистрам пользователь может получить через команды IN и OUT микропроцессора в адресном пространстве ВУ. Организация памяти MFT (с фиксированными разделами) с защитой по маскам.

2.4 Режим прерываний (КПП)

Под прерыванием понимают временную приостановку выполнения программы и переход на другую подпрограмму с возможностью возврата на прерванную.

В процессе инициализации системы процессор записывает в регистр состояния ВУ единицу в бит разрешения прерывания, если это устройство будет работать в режиме прерывания.

Когда ВУ готово к обмену, устанавливается бит готовности в регистре состояния своим контроллером. По совпадении сигналов готовности и разрешения прерывания формируется низким уровнем сигнал требование прерывания на общей одеопроводной шине.

Мы используем централизованный контролер ПП.

Дешифратор МК преобразует МК прерывания в необходимые управляющие сигналы для блоков микросхемы. Рг Пр служит для записи входных сигналов прерывания, то есть IRQi (ВУ) и представляет собой 8-разрядный регистр.

## 2.5 Прямой доступ к памяти (КПДП)

Режим ПДП является самым скоро­стным способом обмена, который реализуется с помощью специальных аппа­ратных средств — контроллеров ПДП без использования программного обеспе­чения. Для осуществления режима ПДП контроллер должен выполнить ряд последовательных операций (рис. 2.5.1):

1) принять запрос DREQ на ПДП от ВУ;

2) сформировать запрос HRQ на захват шин для ЦП;

3) принять сигнал HLDA, подтверждающий этот факт после того, как ЦП войдет в состояние захвата (ШД, ША, ШУ в z-состоянии);

4) сформировать сигнал DACK, сообщающий ВУ о начале выполнения циклов ПДП;

5) сформировать на ША адрес ячейки памяти, предназначенный для обмена;

6) выработать сигналы MR, IOW и MW, IOR, обеспечивающие управление обменом;

7) по окончании ПДП либо повторить цикл ПДП, изменив адрес, либо прекратить ПДП, сняв запросы на ПДП.

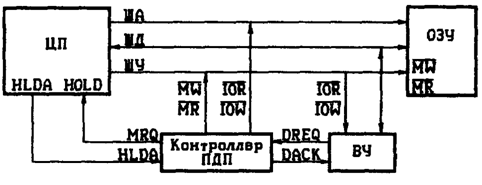


Рис 2.5.1 Структурная схема централизованного КПДП.

# 3. Система команд

## 3.1 Основные команды

Основные команды МК48 включают в себя 96 основных команд и ориентированы на реализацию процедур управления. Все команды имеют формат один или два байта (70% команд однобайтные). Время выполнения команд составляет 2.5 или 5.0 мкс (один или дна машинных цикла соответственно) при тактовой частоте 6.0 МГц. Большинство команд выполняется за один машинный цикл. За два машинных цикла выполняются команды с непосредственным операндом, ввода/вывода и передачи управления.

Пример некоторых операций показан в таблице 3.1:

Таблица 3.1

|  |  |
| --- | --- |
| Название команды | Мнемокод |
| Пересылка регистра в аккумулятор | MOV А,Rn |
| Пересылка байта из РПД в аккумулятор | MOV A,@Ri |
| Пересылка непосредственного операнда в аккумулятор | MOV A,#d |
| Пересылка аккумулятора в регистр | MOV Rn,A |
| Пересылка непосредственного операнда в регистр | MOV Rn,#d |
| Пересылка аккумулятора в РПД | MOV @Ri,A |
| Пересылка непосредственного операнда в РПД | MOV @Ri,#d |

Команды арифметические и логические

Группа команд арифметических операции. Данная группа состоит из 12 команд и позволяет выполнять следующие операции над 8-битными целыми двоичными числами без знака(Таблица 3.1.1.1.)

Таблица 3.1.1.1

|  |  |
| --- | --- |
| Название команды | Мнемокод |
| Сложение регистра с аккумулятором | ADD А,Rn |
| Сложение константы с аккумулятором | ADD A,#d |
| Сложение регистра с аккумулятором и переносом | ADDC A,Rn |
| Инкремент аккумулятора | INC А |
| Инкремент регистра | INC Rn |
| Декремент аккумулятора | DEC A |
| Декремент регистра | DEC Rn |

1. Группа команд логических операций. Данная группа состоит из 28 команд и позволяет выполнять следующие операции над байтами. Пример некоторых операций показан в таблице 3.1.1.2:

Таблица 3.1.1.2

|  |  |
| --- | --- |
| Название команды | Мнемокод |
| Логическое И регистра и аккумулятора | ANL A,Rn |
| Логическое ИЛИ константы и  аккумулятора | ORL A,#d |
| Исключающее ИЛИ регистра и аккумулятора | XRL A,Rn |
| Сброс аккумулятора | CLR A |
| Инверсия аккумулятора | CPL A |
| Циклический сдвиг влево аккумулятора | RL А |
| Циклический сдвиг вправо аккумулятора | RR А |
| Сброс переноса | CLR С |
| Сброс флага F1 | CLR F1 |
| Инверсия переноса | CPL С |
| Инверсия флага F1 | CPL F1 |

Пример программы с приведённым алгоритмом:



Рис. 3.1.1 Флгоритм вычисления функции

Код програми

;F = (X1-X2)/4 + (X5 + X6) \*5

In A, P1

Mov R1, A

In A, P1

Mov R2, A

In A, P1

Mov R3, A

In A, P1

Mov R4, A

In A, P1

Mov A, R2

Cpl A

Inc A

Add A, R1

Clr C

Rrc A; (X1-X2)/2

Clr C

Rrc A; (X1-X2)/4

Clr C

Mov R0, A

;X5+X6

Mov A, R3

Add A,R4

;(X5+X6)\*5

Clr C

Mov R5 , A

Rlc A ;(X5 + X6)\*2

Clr C

Rlc A ;(X5 + X6)\*4

Clr C

Add A,R5;(X5 + X6)\*5

Mov R3,A

Clr C

Mov A, R0

Add A,R3

Clr C

Out P1,A

## 3.1.2 Команды передачи управления.

Таблица 3.1.2.1

|  |  |
| --- | --- |
| Название команды | Мнемокод |
| Безусловный переход | JMP ad11 |
| Декремент регистра и переход, если не нуль | DJNZ Rn,ad |
| Переход, если перенос | JC ad |
| Переход ,если нет переноса | JNC ad |
| Переход, если аккумулятор содержит нуль | JZ ad |
| Переход, если аккумулятор содержит не нуль | JNZ ad |
| Переход, если флаг F1 установлен | JF1 ad |
| Возврат из подпрограммы | RET |

Пример программы с приведённым алгоритмом:



Рис. 3.1.2.1 Алгоритм вычисления функции

Листинг программы:

; LOAD OPERANDS

SEL RB1

MOV R0, #00H

MOV R7, #07H

LOAD\_LOOP:

IN A, P2

MOV @R0, A

INC R0

DJNZ R7, LOAD\_LOOP

CLR A

XCH A, R0

NOP

; R3 = R2-R1-1

SEL RB0

MOV A ,R1

CPL A

ADDC A, #1

MOV R1, A

SEL RB1

MOV A ,R1

CPL A

ADDC A, #0

MOV R1, A

CLR C

SEL RB0

MOV A ,R2

ADDC A, R1

MOV R3, A

SEL RB1

MOV A ,R2

ADDC A, R1

MOV R3, A

CLR C

SEL RB0

MOV A, R3

ADDC A, #FFH

MOV R3, A

SEL RB1

MOV A, R3

ADDC A, #FFH

MOV R3, A

NOP

;CYCLE:

MOV R7, #3H

CYCLE:

; C=1?

JC COUNT

CALL MPP

;R4=(R3+R1+1)/2

CLR C

SEL RB0

MOV A,R1

INC A

MOV R4, A

SEL RB1

MOV A, R1

ADDC A , #0H

MOV R4, A

CLR C

SEL RB0

MOV A,R4

ADDC A , R3

MOV R4 , A

SEL RB1

MOV A , R4

ADDC A , R3

MOV R4 , A

CLR C

SEL RB1

MOV A, R4

RRC A

MOV R4, A

SEL RB0

MOV A, R4

RRC A

MOV R4, A

;R6 = R4VR3

CLR C

MOV A , R4

ORL A ,R3

MOV R6 , A

SEL RB1

MOV A, R4

ORL A , R3

MOV R6 , A

SEL RB1

DJNZ R7 , CYCLE

;R3=R6+25H

CLR C

SEL RB0

MOV A, R6

ADDC A , #25H

MOV R3 , A

SEL RB1

MOV A, R6

ADDC A, #0H

MOV R3, A

NOP

JMP ENDED

;R6-1

CLR C

SEL RB0

MOV A, R0

ADDC A, #FFH

MOV R1, A

SEL RB1

MOV A, R6

ADDC A, #FFH

MOV R1, A

CLR C

SEL RB0

MOV A , R1

RLC A

MOV R1 , A

SEL RB1

MOV A , R1

RLC A

MOV R1 , A

;F0 =1 ?

CLR F0

CPL F0

JC CONT\_MPP

;R2 = R4&R1

SEL RB0

MOV A , R4

ANL A, R1

MOV R2,A

SEL RB1

MOV A , R4

ANL A, R1

MOV R2,A

;R5 = R2-R0-1

CLR C

SEL RB0

MOV A, R0

CPL A

ADDC A, #1

MOV R0, A

SEL RB1

MOV A, R0

CPL A

ADDC A, #0

MOV R0, A

CLR C

SEL RB0

MOV A, R2

ADDC A, R0

MOV R5, A

SEL RB1

MOV A, R2

ADDC A, R0

MOV R5, A

CLR C

SEL RB0

MOV A, R0

ADDC A, #FFH

MOV R0, A

SEL RB1

MOV A, R0

ADDC A, #FFH

MOV R0, A

CONT\_MPP:

;Blok 11

;R3=R3+R5

CLR C

SEL RB0

MOV A , R3

ADDC A , R5

MOV R3, A

SEL RB1

MOV A , R3

ADDC A , R5

MOV R3, A

NOP

RET

ENDED:

NOP

END

## 3.1.3 Команды управления режимом работы МК

Группа команд управления режимом работы МК. В эту группу входят команды управления таймером/счетчиком, прерываниями и флагами переключения банков регистров и банков ПП. Пример некоторых операций показан в таблице 3.5:

Таблица 3.5

|  |  |
| --- | --- |
| Название команды | Мнемокод |
| Запуск таймера | STRT T |
| Запуск счетчика | STRT CNT |
| Останов таймера/счетчика | STOP TCNT |
| Выбор нулевого банка регистров | SEL RB0 |
| Выбор первого банка регистров | SEL RB1 |
| Выбор нулевого банка ПП | SEL МВ0 |
| Выбор первого банка ПП | SEL МВ1 |
| Холостая команда | NOP |

Пример программы:

STARS: DIS TCNT1 ; Заперт прерываний от таймера

CLR A ; Сброс аккумулятора

MOV T,A ; Сброс таймера

MOV R5,A ; Сброс регистра 5

STRT T ; Запуск таймера

M: JTF COUNT ; Если TF=1, то переход к COUNT и сброс TF

JMP M1 ; Цикл

COUNT: INC R5 ; Инкремент регистра 5

MOV A,R5 ; Пересылка содержимого R5 ваккумулятор

JB3 INT ; Переход к подпрограмме обслуживания прерывания IN,

; если бит А.3 равен 1

JMP M1 ;Переход, если бит А.3 не равен 1

. . .

. . .

INT: STOP TCNT ; Останов таймера

JMP 07H ; Переход к ячейке 7 (вектор прерывания от счетчика событий)

# 4. Выполнение индивидуального задания

## 4.1 Алгоритм программы первый способ умножения 24 разрядные операнды



Рис. 4.2.1. Операционная схема умножения

## 4.2. Карта распределения памяти внешних устройств.

Таблица 4.2.1 Карта распределения памяти внешних устройств

|  |  |  |  |
| --- | --- | --- | --- |
| F1h | 1111 0001 | PУC | ВВ51 |
| F0h | 1111 0000 | РД |
| … | | | |
| 37h | 0011 0111 | РД | ВУ45 |
| 36h | 0011 0110 | РС |
| 35h | 0011 0101 | РД | ВУ24 |
| 34h | 0011 0100 | РС |
| … | | | |
| 25h | 0010 0101 | РД | ВУ17 |
| 24h | 0010 0100 | РС |
| 23h | 0010 0011 | PYC | ВВ55 |
| 22h | 0010 0010 | PC |
| 21h | 0010 0001 | PB |
| 20h | 0010 0000 | PA |
| 1Fh | 0001 1111 | РД | ВУ16 |
| 1Eh | 0001 1110 | РС |
| … | | | |
| 1h | 0000 0001 | РД | ВУ1 |
| 0h | 0000 0000 | РС |

## 4.3. Блок-схема алгоритма вычисления выражения



Рис. 4.3.1 Блок схема умножения 1 способом

4.4 Код микропрограммы

;RGX – R1.R2.R3 банк регістрів 0

;RGY – R4.R5.R6 банк регістрів 0

;RGZ – R1.R2.R3 банк регістрів 1

;Ініціалізація

Sel Rb0

In A, P1

Mov R1, A

In A, P1

Mov R2, A

In A, P1

Mov R3, A

In A, P1

Mov R4, A

In A, P1

Mov R5, A

In A, P1

Mov R6, A

Mov R7, #24

;Виділення кінцевого знаку

Mov A, #0

Clr F0

Add A, R1

Xrl A, R4

Cpl A

Jb7 Label

Cpl F0

;Знак RGX

Label:

Mov A, #0

Add A, R1

Jb7 Label1

Jmp Label2

Label1:

Mov A, R1

Anl A, 7fh

Mov R1, A

Label2:

;Перетворення в ДК RGY, якщо число від'ємне-----

Mov A, #0

Add A, R4

Jb7 Label3

Jmp Next

Label3:

Clr C

Mov A, R6

Cpl A

Inc A

Mov R6, A

Mov A, R5

Cpl A

Addc A, #0

Mov R5, A

Mov A, R4

Anl A, 7fh

Cpl A

Addc A, #0

Mov R4, A

;Множення

Next:

;Перевірка останнього біту RGX

Mov A, #1

Anl A, R3

Jz Mark1

;RGZ=RGZ+RGY

Sel Rb0

Mov A, #0

Orl A, R6

Sel Rb1

Clr C

Add A, R3

Mov R3, A

Sel Rb0

Mov A, #0

Orl A, R5

Sel Rb1

Addc A, R2

Mov R2, A

Sel Rb0

Mov A, #0

Orl A, R4

Sel Rb1

Addc A, R1

Mov R1, A

Mark1:

;0.RGZ->

Sel Rb1

Clr C

Mov A, R1

Rrc A

Mov R1, A

Mov A, R2

Rrc A

Mov R2, A

Mov A, R3

Rrc A

Mov R3, A

;RGZ(1).RGX->

Sel Rb0

Mov A, R1

Rrc A

Mov R1, A

Mov A, R2

Rrc A

Mov R2, A

Mov A, R3

Rrc A

Mov R3, A

Djnz R7, Next

;Встановлення знаку результату

Sel Rb1

Cpl F0

Jf0 Theend

Mov A, #80H

Orl A, R1

Mov R1, A

Theend:

End

# 5. Структурная схема системы.

Структурная схема МПС приведена на чертеже ИАЛЦ 460104 004. Е1. Ее ядром является микроконтроллер КР1816ВЕ48.

В состав МПС входят такие основные функциональные части:

1. Микроконтроллер МК-48.
2. Внешняя память данных 8K, внешняя память программ 4К.
3. Централизованный контроллер прямого доступа к памяти – КПДП.
4. Внешние устройства – 25 единиц.
5. Централизованный контроллер приоритетного прерывания – КПП.
6. Дополнительные порты P4,P5,P6.
7. Периферийный адаптер ВВ55 для подключения ВУ
8. Периферийный адаптер ВВ51

6. Выводы

Разработана микропроцессорная система на основе МК-48 с подключением внешней памяти данных, внешней памяти программ, периферийного адаптера, внешних устройств.

# 7. Список литературы.

1. Бояринов А.Е., Дьяков И.А. *–* Архитектура микроконтроллеров MCS-51 – Тамбов: “Издательство ТГТУ”, 2005.
2. Конспект лекций по курсу “Архитектура ЭВМ”.
3. Жабин В.И., Ткаченко В.В. – Однокристальные и микропрограммируемые ЭВМ. – Киев, “Диалектика”, 1995.
4. Жабин В.И., Ткаченко В.В., Макаров В.В., Зайцев А.А. – Архитектура однокристальных ЭВМ. – Киев, “Век”, 1997.