目录

[实验1 存储器读写实验 1](#_Toc28121445)

[1 实验目的 1](#_Toc28121446)

[2 实验设备 1](#_Toc28121447)

[3 实验内容 1](#_Toc28121448)

[4 实验原理 1](#_Toc28121449)

[5 实验步骤 3](#_Toc28121450)

[6 总结 3](#_Toc28121451)

[实验2 中断实验 4](#_Toc28121452)

[1 实验目的 4](#_Toc28121453)

[2 实验设备 4](#_Toc28121454)

[3 实验内容及步骤 4](#_Toc28121455)

[4 总结 9](#_Toc28121456)

[实验3 可编程并行接口实验 10](#_Toc28121457)

[1 实验目的 10](#_Toc28121458)

[2 实验设备 10](#_Toc28121459)

[3 实验内容 10](#_Toc28121460)

[4 实验步骤 10](#_Toc28121461)

[5 总结 13](#_Toc28121462)

[实验4 串行通信实验 14](#_Toc28121463)

[1 实验目的 14](#_Toc28121464)

[2 实验设备 14](#_Toc28121465)

[3 实验内容 14](#_Toc28121466)

[4 实验步骤 14](#_Toc28121467)

[5 总结 20](#_Toc28121468)

[实验5 可编程定时器/计数器实验 21](#_Toc28121469)

[1 实验目的 21](#_Toc28121470)

[2 实验设备 21](#_Toc28121471)

[3 实验内容 21](#_Toc28121472)

[4 实验步骤 21](#_Toc28121473)

[5 总结 24](#_Toc28121474)

[实验6 数据采集实验 25](#_Toc28121475)

[1 实验目的 25](#_Toc28121476)

[2 实验设备 25](#_Toc28121477)

[3 设计内容 25](#_Toc28121478)

[4 实验原理 25](#_Toc28121479)

[5 实验步骤 28](#_Toc28121480)

[6 总结 32](#_Toc28121481)

## 实验1 存储器读写实验

### 1 实验目的

1. 了解存储器扩展的方法和存储器的读/写。

2. 掌握CPU对16位存储器的访问方法。

### 2 实验设备

PC机一台，TD-PITE实验装置或TD-PITC实验装置一套，示波器一台。

### 3 实验内容

编写实验程序，将0000H～000FH共16个数写入SRAM的从0000H起始的一段空间中，然后通过系统命令查看该存储空间，检测写入数据是否正确。

### 4 实验原理

存储器是用来存储信息的部件，是计算机的重要组成部分，静态RAM是由MOS管组成的触发器电路，每个触发器可以存放1位信息。只要不掉电，所储存的信息就不会丢失。因此，静态RAM工作稳定，不要外加刷新电路，使用方便。但一般 **图1.1 62256引脚图**

SRAM 的每一个触发器是由6个晶体管组成，SRAM芯片的集成度不会太高，目前较常用的有6116（2K×8位），6264（8K×8位）和62256（32K×8位）。本实验平台上选用的是62256，两片组成32K×16位的形式，共64K字节。62256的外部引脚图如图1.1所示。

本系统采用准32位CPU，具有16位外部数据总线，即D0、D1、…、D15，地址总线为BHE＃（＃表示该信号低电平有效）、BLE＃、A1、A2、…、A20。存储器分为奇体和偶体，分别由字节允许线BHE＃和BLE＃选通。

存储器中，从偶地址开始存放的字称为规则字，从奇地址开始存放的字称为非规则字。处理器访问规则字只需要一个时钟周期，BHE＃和BLE＃同时有效，从而同时选通存储器奇体和偶体。处理器访问非规则字却需要两个时钟周期，第一个时钟周期BHE＃有效，访问奇字节；第二个时钟周期BLE＃有效，访问偶字节。处理器访问字节只需要一个时钟周期，视其存放单元为奇或偶，而BHE＃或BLE＃有效，从而选通奇体或偶体。写规则字和非规则字的简单时序图如图4.2所示。

**图1.2 写规则字（左）和非规则字（右）简单时序图**

实验单元电路图



**图1.3 SRAM单元电路图**

实验程序清单（MEM1.ASM）

SSTACK SEGMENT STACK

DW 32 DUP(?)

SSTACK ENDS

CODE SEGMENT

START PROC FAR

ASSUME CS:CODE

MOV AX, 8000H ; 存储器扩展空间段地址

MOV DS, AX

AA0: MOV SI, 0000H ; 数据首地址

 MOV CX, 0010H

MOV AX, 0000H

AA1: MOV [SI], AX

INC AX

INC SI

INC SI

LOOP AA1

MOV AX,4C00H

INT 21H ;程序终止

START ENDP

CODE ENDS

END START

**图1.4 SRAM实验接线图**

### 5 实验步骤

（注：本章实验选择16位寄存器）

1. 实验接线图如图1.4所示，按图接线。

2. 编写实验程序，经编译、链接无误后装入系统。

3. 先运行程序，待程序运行停止。

4. 通过D命令查看写入存储器中的数据：

D8000：0000 回车，即可看到存储器中的数据，应为0001、0002、…、000F共16个字。

5. 改变实验程序，按非规则字写存储器，观察实验结果。

6. 改变实验程序，按字节方式写存储器，观察实验现象。

7. 将接线图中CS连MY1，实验程序如何改变？观察实验结果。

注：MY0：8000H或6000H均可，MY1：A000H或7000H均可。

### 6 总结

## 实验2 中断实验

### 1 实验目的

1. 掌握8259中断控制器的工作原理。

2. 学习8259的应用编程方法。

3. 掌握8259级联方式的使用方法。

### 2 实验设备

PC机一台，TD-PITE实验装置或TD-PITC实验装置一套。

### 3 实验内容及步骤

1. 中断控制器8259简介

在Intel 386EX芯片中集成有中断控制单元（ICU），该单元包含有两个级联中断控制器，一个为主控制器，一个为从控制器。该中断控制单元就功能而言与工业上标准的82C59A是一致的，操作方法也相同。从片的INT连接到主片的IR2信号上构成两片8259的级联。

在TD-PITE实验系统中，将主控制器的IR6、IR7以及从控制器的IR1开放出来供实验使用，主片8259的IR4供系统串口使用。8259的内部连接及外部管脚引出如图3.1：



**图3.1 8259内部连续及外部管脚引出图**

表3.1列出了中断控制单元的寄存器相关信息。

表3.1 ICU寄存器列表

|  |  |  |
| --- | --- | --- |
| 寄存器 | 口地址 | 功能描述 |
| ICW1（主）  ICW1（从）  （只写） | 0020H  00A0H | 初始化命令字1：  决定中断请求信号为电平触发还是边沿触发。 |
| ICW2（主）  ICW2（从）  （只写） | 0021H  00A1H | 初始化命令字2：  包含了8259的基址中断向量号，基址中断向量是IR0的向量号，基址加1就是IR1的向量号，依此类推。 |
| ICW3（主）  （只写） | 0021H | 初始化命令字3：  用于识别从8259设备连接到主控制器的IR信号，内部的从8259连接到主8259的IR2信号上。 |
| ICW3（从）  （只写） | 00A1H | 初始化命令字3：  表明内部从控制器级联到主片的IR2信号上。 |
| ICW4（主）  ICW4（从）  （只写） | 0021H  00A1H | 初始化命令字4：  选择特殊全嵌套或全嵌套模式，使能中断自动结束方式。 |
| OCW1（主）  OCW1（从）  （读/写） | 0021H  00A1H | 操作命令字1：  中断屏蔽操作寄存器，可屏蔽相应的中断信号。 |
| OCW2（主）  OCW2（从）  （只写） | 0020H  00A0H | 操作命令字2：  改变中断优先级和发送中断结束命令。 |
| OCW3（主）  OCW3（从）  （只写） | 0020H  00A0H | 操作命令字3：  使能特殊屏蔽方式，设置中断查询方式，允许读出中断请求寄存器和当前中断服务寄存器。 |
| IRR（主）  IRR（从）  （只读） | 0020H  00A0H | 中断请求：  指出挂起的中断请求。 |
| ISR（主）  ISR（从）  （只读） | 0020H  00A0H | 当前中断服务：  指出当前正在被服务的中断请求。 |
| POLL（主）  POLL（从）  （只读） | 0020H  0021H  00A0H  00A1H | 查询状态字：  表明连接到8259上的设备是否需要服务，如果有中断请求，该字表明当前优先级最高的中断请求。 |

初始化命令字1寄存器（ICW1）说明见图3.2所示。



**图3.2 初始化命令字1寄存器**

初始化命令字2寄存器（ICW2）说明见图3.3所示。



**图3.3 初始化命令字2寄存器**

初始化命令字3寄存器（ICW3）说明，主片见图3.4，从片见图3.5。



**图3.4 主片初始化命令字3寄存器**



**图3.5 从片初始化命令字3寄存器**

初始化命令字4寄存器（ICW4）说明见图3.6。



**图3.6 初始化命令字4寄存器**

操作命令字1寄存器（OCW1）说明见图3.7。



**图3.7 操作命令字1寄存器**

操作命令字2寄存器（OCW2）说明如图3.8所示。



**图3.8 操作命令字2寄存器**

操作命令字3寄存器（OCW3）说明如图3.9所示。



**图3.9 操作命令字3寄存器**

查询状态字（POLL）说明如图3.10所示。



**图3.10 程序状态字寄存器**

在对8259进行编程时，首先必须进行初始化。一般先使用CLI指令将所有的可屏蔽中断禁止，然后写入初始化命令字。8259有一个状态机控制对寄存器的访问，不正确的初始化顺序会造成异常初始化。在初始化主片8259时，写入初始化命令字的顺序是：ICW1、ICW2、ICW3、然后是ICW4，初始化从片8259的顺序与初始化主片8259的顺序是相同的。

系统启动时，主片8259已被初始化，且4号中断源（IR4）提供给与PC联机的串口通信使用，其它中断源被屏蔽。中断矢量地址与中断号之间的关系如下表所示：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 主片中断序号 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 功能调用 | 08H | 09H | 0AH | 0BH | 0CH | 0DH | 0EH | 0FH |
| 矢量地址 | 20H～23H | 24H～27H | 28H～2BH | 2CH～2FH | 30H～33H | 34H～37H | 38H～3BH | 3CH～3FH |
| 说明 | 未开放 | 未开放 | 未开放 | 未开放 | 串口 | 未开放 | 可用 | 可用 |
| 从片中断序号 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 功能调用 | 30H | 31H | 32H | 33H | 34H | 35H | 36H | 37H |
| 矢量地址 | C0H～C3H | C4H～C7H | C8H～CBH | CCH～CFH | D0H～D3H | D4H～D7H | D8H～DBH | DCH～DFH |
| 说明 | 未开放 | 可用 | 未开放 | 未开放 | 未开放 | 未开放 | 未开放 | 未开放 |

2. 8259单中断实验

实验接线图如图3.11所示，单次脉冲输出与主片8259的IR7相连，每按动一次单次脉冲，产生一次外部中断，在显示屏上输出一个字符“7”。

**图3.11 8259单中断实验接线图**

实验程序清单

SSTACK SEGMENT STACK

DW 32 DUP(?)

SSTACK ENDS

CODE SEGMENT

ASSUME CS:CODE

START: PUSH DS

MOV AX, 0000H

MOV DS, AX

MOV AX, OFFSET MIR7 ;取中断入口地址

MOV SI, 003CH ;中断矢量地址

MOV [SI], AX ;填IRQ7的偏移矢量

MOV AX, CS ;段地址

MOV SI, 003EH

MOV [SI], AX ;填IRQ7的段地址矢量

CLI

POP DS

;初始化主片8259

MOV AL, 11H

OUT 20H, AL ;ICW1

MOV AL, 08H

OUT 21H, AL ;ICW2

MOV AL, 04H

OUT 21H, AL ;ICW3

MOV AL, 01H

OUT 21H, AL ;ICW4

MOV AL, 6FH ;OCW1

OUT 21H, AL

STI

AA1: NOP

JMP AA1

MIR7: STI

MOV DL, 37H

MOV AH, 02H

INT 21H

IRET

CODE ENDS

END START

实验步骤

（1）按图3.11连接实验线路。

（2）编写实验程序，经编译、链接无误后装入系统。

（3）运行程序，重复按单次脉冲开关KK1＋，显示屏会显示字符“7”，说明响应了中断。

3. 8259级联实验

实验接线图如图3.12所示，KK1＋连接到主片8259的IR7上，KK2＋连接到从片8259的IR1上，当按一次KK1＋时，显示屏上显示字符“M7”，按一次KK2＋时，显示字符“S1”。编写程序。

**图3.12 8259级联实验**

实验程序清单（INTCAS1.ASM）（中断服务程序略）

SSTACK SEGMENT STACK

DW 32 DUP(?)

SSTACK ENDS

CODE SEGMENT

ASSUME CS:CODE

START: PUSH DS

MOV AX, 0000H

MOV DS, AX

MOV AX, OFFSET MIR7 ;取中断入口地址

MOV SI, 003CH ;中断矢量地址

MOV [SI], AX ;填IRQ7的偏移矢量

MOV AX, CS ;段地址

MOV SI, 003EH

MOV [SI], AX ;填IRQ7的段地址矢量

CLI

POP DS

;初始化主片8259

MOV AL, 11H

OUT 20H, AL ;ICW1

MOV AL, 08H

OUT 21H, AL ;ICW2

MOV AL, 04H

OUT 21H, AL ;ICW3

MOV AL, 01H

OUT 21H, AL ;ICW4

MOV AL, 6FH ;OCW1

OUT 21H, AL

STI

AA1: NOP

JMP AA1

MIR7: STI

CALL DELAY

MOV AX,0137H

INT 10H

MOV AX,0120H

INT 10H

MOV AL,20H

OUT 20H,AL

IRET

DELAY: PUSH CX

MOV CX,0F00H

AA0： PUSH AX

POP AX

LOOP AA0

POP CX

RET

CODE ENDS

END START

实验步骤

（1）按图3.12连接实验线路。

（2）输入程序，编译、链接无误后装入系统。

（3）运行程序，按动KK1＋或KK2＋，观察实验结果，验证实验程序的正确性。

（4）若同时按下KK1＋和KK2＋，观察实验结果

### 4 总结

## 实验3 可编程并行接口实验

### 1 实验目的

1. 学习并掌握8255的工作方式及其应用。

2. 掌握8255典型应用电路的接法。

3. 掌握程序固化及脱机运行程序的方法。

### 2 实验设备

PC机一台，TD-PITE实验装置或TD-PITC实验装置一套。

### 3 实验内容

1. 基本输入输出实验。编写程序，使8255的A口为输入，B口为输出，完成拨动开关到数据灯的数据传输。要求只要开关拨动，数据灯的显示就发生相应改变。

2. 流水灯显示实验。编写程序，使8255的A口和B口均为输出，数据灯D7～D0由左向右，每次仅亮一个灯，循环显示，D15～D8与D7～D0正相反，由右向左，每次仅点亮一个灯，循环显示。

### 4 实验步骤

1. 基本输入输出实验

本实验使8255端口A工作在方式0并作为输入口，端口B工作在方式0并作为输出口。用一组开关信号接入端口A，端口B输出线接至一组数据灯上，然后通过对8255芯片编程来实现输入输出功能。具体实验步骤如下述：

（1）实验接线图如图2.1所示，按图连接实验线路图。

（2）编写实验程序，经编译、连接无误后装入系统。

（3）运行程序，改变拨动开关，同时观察LED显示，验证程序功能。

（4）点击“调试”下拉菜单中的“固化程序”项，将程序固化到系统存储器中。

（5）将短路跳线JDBG的短路块短接到RUN端，然后按复位按键，观察程序是否正常运行；关闭实验箱电源，稍等后再次打开电源，看固化的程序是否运行，验证程序功能。

（6）实验完毕后，请将短路跳线JDBG的短路块短接到DBG端。



**图2.1 8255基本输入输出实验接线图**

实验程序清单（A82551.ASM）略

DW 32 DUP(?)

SSTACK ENDS

CODE SEGMENT

ASSUME CS:CODE

START: MOV DX,0646H

MOV AL,90H

OUT DX,AL

LOP: MOV DX,0640H

IN AL,DX

CALL DELAY

MOV DX,0642H

OUT DX,AL

JMP LOP

DELAY: PUSH CX

MOV CX,0F00H

L: PUSH AX

POP AX

LOOP L

POP CX

RET

CODE ENDS

END START

2. 流水灯显示实验

使8255的A口和B口均为输出，数据灯D7～D0由左向右，每次仅亮一个灯，循环显示，D15～D8与D7～D0正相反，由右向左，每次仅点亮一个灯，循环显示。实验接线图如图2.2所示。实验步骤如下所述：

（1）按图2.2连接实验线路图。

（2）编写实验程序，经编译、链接无误后装入系统。

（3）运行程序，观察LED灯的显示，验证程序功能。

（4）自己改变流水灯的方式，编写程序。

（5）固化程序并脱机运行。



**图2.2 8255流水灯实验接线图**

实验程序清单（A82552.ASM）

SSTACK SEGMENT STACK

DW 32 DUP(?)

SSTACK ENDS

CODE SEGMENT

ASSUME CS:CODE

START: MOV DX,0646H

MOV AL,80H

OUT DX,AL

MOV AL,80H

MOV BL,01H

LOP: MOV DX,0642H

OUT DX,AL

ROL AL,1

MOV DX,0640H

PUSH AX

MOV AL,BL

OUT DX,AL

ROR AL,1

MOV BL,AL

POP AX

CALL DELAY

JMP LOP

DELAY: PUSH CX

MOV CX,0FF00H

L: PUSH AX

POP AX

LOOP L

POP CX

RET

CODE ENDS

END START

### 5 总结

实验过程中，获得了很多收获，获得了很多感悟，当然也遇到了很多困难。但我们都一一克服了他们，成功的完成了实验。并在解决问题，克服困难的过程中，发现了自己平时忽略的，隐藏的问题，以及一些不该出现的粗心大意的小毛病。通过这些，我们认识的更加深刻，了解的更加深入。做到了学以致用，对知识掌握得更加牢固。通过了这一学期对微机原理的学习，真的对它有了一个全新的认识，我会坚持对它的学习，使自己在汇编语言上有一个长足的提高。

## 实验4 串行通信实验

### 1 实验目的

1. 掌握8251的工作方式及应用。

2. 了解有关串口通讯的知识。

### 2 实验设备

PC机两台，TD-PITE实验装置或TD-PITC实验装置两套。

### 3 实验内容

1.自收自发实验，将3000H起始的10个单元中的初始数据发送到串口，然后自接收并保存到4000H起始的内存单元中。

2. 双机通讯实验，本实验需要两台实验装置，其中一台作为接收机，一台作为发送机，发送机将3000H～3009H内存单元中共10个数发送到接收机，接收机将接收到的数据直接在屏幕上输出显示。

### 4 实验步骤

1. 自收自发实验

通过自收自发实验，可以验证硬件及软件设计，常用于自测试。具体实验步骤如下：

（1）参考实验接线图如图4.1所示，按图连接实验线路。

（2）编写实验程序，编译、链接无误后装入系统。

（3）使用E命令更改4000H起始的10个单元中的数据。

（4）运行实验程序，待程序运行停止。

（5）查看3000H起始的10个单元中的数据，与初始化的数据进行比较，验证程序功能。

实验参考例程（A82512.ASM）

M8251\_DATA EQU 0600H ;端口定义

M8251\_CON EQU 0602H

M8254\_2 EQU 06C4H

M8254\_CON EQU 06C6H

SSTACK SEGMENT STACK

DW 64 DUP(?)

SSTACK ENDS

CODE SEGMENT

ASSUME CS:CODE

START: MOV AX, 0000H

MOV DS, AX

;初始化8254，得到收发时钟

MOV AL, 0B6H

MOV DX, M8254\_CON

OUT DX, AL

MOV AL, 0CH

MOV DX, M8254\_2

OUT DX, AL

MOV AL, 00H

OUT DX, AL

;复位8251

CALL INIT

CALL DALLY

;8251方式字

MOV AL,7EH

MOV DX, M8251\_CON  **图4.1 自收自发实验接线图**

OUT DX, AL

CALL DALLY

;8251控制字

MOV AL, 34H

OUT DX, AL

CALL DALLY

MOV DI, 3000H

MOV SI, 4000H

MOV CX, 000AH

A1: MOV AL, [SI]

PUSH AX

MOV AL, 37H

MOV DX, M8251\_CON

OUT DX, AL

POP AX

MOV DX, M8251\_DATA

OUT DX, AL ;发送数据

MOV DX, M8251\_CON

A2: IN AL, DX ;判断发送缓冲是否为空

AND AL, 01H

JZ A2

CALL DALLY

A3: IN AL, DX ;判断是否接收到数据

AND AL, 02H

JZ A3

MOV DX, M8251\_DATA

IN AL, DX ;读取接收到的数据

MOV [DI], AL

INC DI

INC SI

LOOP A1

MOV AX,4C00H

INT 21H ;程序终止

INIT: MOV AL, 00H ;复位8251子程序

MOV DX, M8251\_CON

OUT DX, AL

CALL DALLY

OUT DX, AL

CALL DALLY

OUT DX, AL

CALL DALLY

MOV AL, 40H

OUT DX, AL

RET

DALLY: PUSH CX

MOV CX,3000H

A5: PUSH AX

POP AX

LOOP A5

POP CX

RET

CODE ENDS

END START

2. 双机通讯实验

使用两台实验装置，一台为发送机，一台为接收机，进行两机间的串行通讯。实验步骤如下：

（1）按图4.2连接实验线路。

（2）为两台机器分别编写实验程序，编译、链接后装入系统。

（3）为发送机初始化发送数据。在发送机3000H～3009H内存单元写入ASCII值：30，31，32，33，34，35，36，37，38，39共10个数。

（4）首先运行接收机上的程序，等待接收数据，然后运行发送机上的程序，将数据发送到串口。

（5）观察接收机端屏幕上的显示是否与发送机端初始的数据相同，验证程序功能。

屏幕将会显示字符：0123456789

实验参考例程（接收机）

M8251\_DATA EQU 0600H

M8251\_CON EQU 0602H

M8254\_2 EQU 06C4H

M8254\_CON EQU 06C6H

SSTACK SEGMENT STACK

DW 64 DUP(?)

SSTACK ENDS

CODE SEGMENT

ASSUME CS:CODE START: MOV AL, 0B6H ; 初始化 8254

MOV DX, M8254\_CON

OUT DX, AL

MOV AL, 0CH

MOV DX, M8254\_2

OUT DX, AL

MOV AL, 00H

OUT DX, AL

CLI CALL INIT ; 复位 8251

CALL DALLY

MOV AL, 7EH

MOV DX, M8251\_CON

OUT DX, AL

CALL DALLY

MOV AL, 34H

OUT DX, AL

CALL DALLY MOV AX, 0152H ; 输出显示字符 'R'

INT 10H

MOV DI, 3000H

MOV CX, 000AH

A1: IN AL, DX

AND AL, 02H

JZ A1

MOV DX, M8251\_DATA

IN AL, DX

AND AL, 7FH

MOV [DI],AL

INC DI

LOOP A1

MOV AL, 00H

9

MOV SI, 300AH

MOV [SI], AL

MOV AH, 06H

MOV BX, 3000H INT 10H ; 输出显示接收到的数据

STI

A2: JMP A2 INIT: MOV AL, 00H ; 复位 8251子程序

MOV DX, M8251\_CON

OUT DX, AL

CALL DALLY

OUT DX, AL

CALL DALLY

OUT DX, AL

CALL DALLY

MOV AL, 40H

OUT DX, AL

RET

DALLY: PUSH CX

MOV CX, 3000H

A3: PUSH AX

POP AX

LOOP A3

POP CX

RET

CODE ENDS

END START

实验参考程序（发送机）

M8251\_DATA EQU 0600H

M8251\_CON EQU 0602H

M8254\_2 EQU 06C4H

M8254\_CON EQU 06C6H

SSTACK SEGMENT STACK

DW 64 DUP(?)

SSTACK ENDS

CODE SEGMENT

ASSUME CS:CODE START: MOV AL, 0B6H ; 初始化 8254, 得到收发时钟

MOV DX, M8254\_CON

OUT DX, AL

MOV AL, 0CH

MOV DX, M8254\_2

OUT DX, AL

10

MOV AL, 00H

OUT DX, AL CALL INIT ; 复位 8251

CALL DALLY

MOV AL, 7EH

MOV DX, M8251\_CON OUT DX, AL ;8251 方式字

CALL DALLY

MOV AL, 34H OUT DX, AL ;8251 控制字

CALL DALLY

MOV DI, 3000H

MOV CX, 000AH

A1: MOV AL, [DI]

CALL SEND

CALL DALLY

INC DI

LOOP A1

A2: JMP A2 INIT: MOV AL, 00H ; 复位 8251子程序

MOV DX, M8251\_CON

OUT DX, AL

CALL DALLY

OUT DX, AL

CALL DALLY

OUT DX, AL

CALL DALLY

MOV AL, 40H

OUT DX, AL

RET

DALLY: PUSH CX

MOV CX, 3000H

A4: PUSH AX

POP AX

LOOP A4

POP CX

RET SEND: PUSH AX ;数据发送子程序

PUSH DX

MOV AL, 31H

MOV DX, M8251\_CON

OUT DX, AL

MOV DX, M8251\_DATA

OUT DX, AL

11

MOV DX, M8251\_CON

A3: IN AL, DX

AND AL, 01H

JZ A3

POP DX

POP AX

RET

CODE ENDS

END START



**图4.2 双机通讯实验接线图**

### 5 总结

不可否认，本程序存在不少缺点和不足，但通过这次课程设计的却找出了自己在学习上的不足，对以后的工作也有指导作用。我相信在以后的学习中会克服这些不足，达到熟练掌握汇编语言的目的！

## 实验5 可编程定时器/计数器实验

### 1 实验目的

1. 掌握8254的工作方式及应用编程。

2. 掌握8254典型应用电路的接法。

### 2 实验设备

PC机一台，TD-PITE实验装置或TD-PITC实验装置一套，示波器一台。

### 3 实验内容

1. 计数应用实验。编写程序，应用8254的计数功能，使用单次脉冲模拟计数，使每当按动‘KK1＋’5次后，产生一次计数中断，并在屏幕上显示一个字符‘M’。

2. 定时应用实验。编写程序，应用8254的定时功能，产生一个1s的方波。

### 4 实验步骤

1. 计数应用实验

编写程序，将8254的计数器0设置为方式0，计数值为十进制数4，用单次脉冲KK1＋作为CLK0时钟，OUT0连接MIR7，每当KK1＋按动5次后产生中断请求，在屏幕上显示字符“M”。

实验步骤：

（1）实验接线如图5.1所示。

（2）编写实验程序，经编译、链接无误后装入系统。

（3）运行程序，按动KK1＋产生单次脉冲，观察实验现象。

（4）改变计数值，验证8254的计数功能。



**图5.1 8254计数应用实验接线图**

实验程序清单（A82541.ASM）部分略

A8254 EQU 06C0H

B8254 EQU 06C2H

C8254 EQU 06C4H

CON8254 EQU 06C6H

SSTACK SEGMENT STACK

DW 32 DUP(?)

SSTACK ENDS

CODE SEGMENT

ASSUME CS:CODE, SS:SSTACK

START: PUSH DS

MOV AX, 0000H

MOV DS, AX

MOV AX, OFFSET IRQ7 ;取中断入口地址

MOV SI, 003CH ;中断矢量地址

MOV [SI], AX ;填IRQ7的偏移矢量

MOV AX, CS ;段地址

MOV SI, 003EH

MOV [SI], AX ;填IRQ7的段地址矢量

CLI

POP DS

;初始化主片8259

MOV AL, 11H

OUT 20H, AL ;ICW1

MOV AL, 08H

OUT 21H, AL ;ICW2

MOV AL, 04H

OUT 21H, AL ;ICW3

MOV AL, 01H

OUT 21H, AL ;ICW4

MOV AL, 6FH ;OCW1

OUT 21H, AL

;8254

MOV DX, CON8254

MOV AL, 10H ;计数器0，方式0

OUT DX, AL

MOV DX, A8254

MOV AL, 04H

OUT DX, AL

STI

AA1: JMP AA1

IRQ7: MOV DX, A8254

MOV AL, 04H

OUT DX, AL

MOV AX, 014DH

INT 10H ;显示字符M

MOV AX, 0120H

INT 10H

MOV AL, 20H

OUT 20H, AL ;中断结束命令

IRET

CODE ENDS

END START

END START

2. 定时应用实验

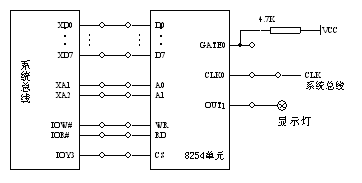
编写程序，将8254的计数器0设置为方式3，用信号源1MHz作为CLK0时钟，OUT1为波形输出1s方波。

实验步骤：

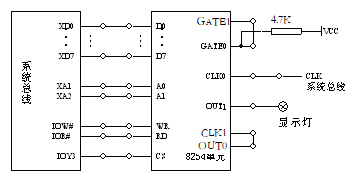
（1）部分接线图如图5.2所示，其余请补充。

（2）根据实验内容，编写实验程序，经编译、链接无误后装入系统。

（3）运行实验程序，用OUT1连显示灯，观察灯的输出状态，验证程序功能。



**图5.2 8254定时器应用实验接线图**



**图5.3 8254定时器应用实验完整接线图**

实验程序清单（A82542.ASM）

A8254 EQU 06C0H

B8254 EQU 06C2H

C8254 EQU 06C4H

CON8254 EQU 06C6H

SSTACK SEGMENT STACK

DW 32 DUP(?)

SSTACK ENDS

CODE SEGMENT

ASSUME CS:CODE

START: MOV DX, CON8254 ;8254

MOV AL, 36H ;计数器0，方式3

OUT DX, AL

MOV DX, A8254

MOV AL, 0E8H

OUT DX, AL

MOV AL, 03H

OUT DX, AL

AA1: JMP AA1

CODE ENDS

END START

注：系统时钟CLK为1MHZ，上面一个时钟为12M，左边时钟有1.83MHZ

### 5 总结

通过学习和使用汇编语言，向上为理解各种软件系统的原理，打下技术理论基础；向下为掌握硬件系统的原理，打下实践应用基础。不仅巩固了书本所学的知识，还具有一定的灵活性，发挥了我们的创造才能。

# 实验6 数据采集实验

### 1 实验目的

1．掌握8255A的工作原理及使用方法

2．进一步了解ADC0809的性能及编程方法。

3．进一步掌握七段数码管显示数字的原理及编程方法。

### 2 实验设备

PC机一台，TD-PITE实验装置或TD-PITC实验装置一套。

### 3 设计内容

1. 基础实验：编写实验程序，将ADJ单元中提供的0V～5V信号源作为ADC0809的模拟输入量，进行A/D转换，转换结果通过变量进行显示。
2. 提高实验：将ADJ单元中提供的0V～5V信号源送入ADC0809的通道0(IN0)。编程采集IN0输入的电压，并把转换后的数据以十六进制的形式在七段数码管上显示，范围00～FFH。当采集到的电压值超过4.5V时，喇叭发声报警。

### 4 实验原理

1. ADC0809

ADC0809包括一个8位的逐次逼近型的ADC部分，并提供一个8通道的模拟多路开关和联合寻址逻辑。用它可直接输入8个单端的模拟信号，分时进行A/D转换，在多点巡回检测、过程控制等应用领域中使用非常广泛。ADC0809的主要技术指标为：

· 分辨率：8位

· 单电源：＋5V

· 总的不可调误差：±1LSB

· 转换时间：取决于时钟频率

· 模拟输入范围：单极性 0～5V

· 时钟频率范围：10KHz～1280KHz

ADC0809的外部管脚如图6.1所示，地址信号与选中通道的关系如表6.1所示。



**图6.1 ADC0809外部引脚图**

**表6.1 地址信号与选中通道的关系**

|  |  |  |  |
| --- | --- | --- | --- |
| 地 址 | | | 选中通道 |
| A | B | C |
| 0 | 0 | 0 | IN0 |
| 0 | 0 | 1 | IN1 |
| 0 | 1 | 0 | IN2 |
| 0 | 1 | 1 | IN3 |
| 1 | 0 | 0 | IN4 |
| 1 | 0 | 1 | IN5 |
| 1 | 1 | 0 | IN6 |
| 1 | 1 | 1 | IN7 |

模/数转换单元电路图如图6.2所示：



**图6.2 模/数转换电路图**

2．如图6.3、图6.4所示，8255A的PA0～PA6分别与七段数码管的段码驱动输入端a～g相连，8255A的PB0、PB1、PB2与位码驱动输入端X1、X2、X3相连，控制数码管的选通。

ADC0809的转换结束信号EOC与8255A的PC7相连，通过查询方式判断ADC0809的通道0（IN0）是否转换结束。



**图6.3 ADC0809连线图**

D0-D7

600H~606H

CS

a

PA0

b

PA1

c

PA2

d

PA3

e

PA4

f

PA5

g

PA6

X1

PB0

X2

PB1

EOC

PC7

8255A

X3

PB2

GATE0

PC3

**图6.4 8255A连线图**



**图6.5 8254连线图**

### 5 实验步骤

**基础实验1：**

1. 按图6.6连接实验线路。

2. 编写实验程序，经编译、链接无误后装入系统。

3. 将变量VALUE添加到变量监视窗口中。

4. 在JMP START语句行设置断点，使用万用表测量ADJ端的电压值，计算对应的采样值，然后运行程序。

5. 程序运行到断点处停止运行，查看变量窗口中VALUE的值，与计算的理论值进行比较，看是否一致（可能稍有误差，相差不大）。

6. 调节电位器，改变输入电压，比较VALUE与计算值，反复验证程序功能。



**图6.6 AD转换实验接线图**

实验程序清单（AD1.ASM）略

SSTACK SEGMENT STACK

DW 64 DUP(?)

SSTACK ENDS

PUBLIC VALUE ;设置全局变量，以便监视

DATA SEGMENT

VALUE DB ? ;AD转换结果

DATA ENDS

CODE SEGMENT

ASSUME CS:CODE,DS:DATA

START: MOV AX, DATA

MOV DS, AX

MOV DX, 640H ;启动AD采样

OUT DX, AL

CALL DELAY

IN AL, DX ;读AD采样结果

MOV VALUE,AL ;将结果送变量

JMP START ;在此处设置断点，观察窗口中的value值

DELAY: PUSH CX

PUSH AX

MOV CX, 100H

A5: MOV AX, 0800H

A6: DEC AX

JNZ A6

LOOP A5

POP AX

POP CX

RET

CODE ENDS

END START

**提高实验2：**

1. 根据实验原理自己设计并连接实验线路。

2. 编写实验程序，经编译、链接无误后装入系统。

3. 运行程序，调节电位器，观察LED结果。

4．编程提示

（1）ADC0809的IN0端口地址为：640H，8255A的端口地址为：A口：600H、 B口：602H、C口： 604H、控制口：606H。

（2）启动一次A/D转换，转换结束后将采集数据保存到存储器的显示缓冲区中，再调用LED显示功能显示数据。

（3）8254初始化，8254的端口地址设为680H～686H，通道0的工作方式设定为方式3，初值为1000，8255A的PC3初始电平设为低电平。

（4）数据显示完毕之后，判断缓冲区中采集数据是否大于229 (用数字0~255对应模拟电压0~5V的范围，4.5V相当于十进制数的229，(4.5/5)\*255=229)，如果是则将PC3置高电平，由8254输出方波驱动扬声器发声报警。

实验程序清单（略）

SSTACK SEGMENT STACK

DW 64 DUP(?)

SSTACK ENDS

PUBLIC BCD

DATA SEGMENT

TABLE1 DB 3FH,06H,5BH,4FH,66H,6DH,7DH,07H,7FH,6FH ;数码管表

;对应 0 1 2 3 4 5 数码管共阴极

BCD DB 0,0,0

AVERAGE DB 0

DATA ENDS

CODE SEGMENT

ASSUME CS:CODE, DS:DATA

START: MOV AX, DATA

MOV DS, AX

CALL GET

CALL JUG ;判断是否大于4.5

CALL CHBCD

CALL PRT ;调用显示子程序

JMP START

;BCD转换子程序

CHBCD: LEA BX, TABLE1

LEA SI, BCD

MOV CX,3 ;实现三位数码管的数值显示，依次送个位和小数点后的两位

MOV AL, AVERAGE

XOR AH, AH

LOOP2:MOV DL, 51 ;(X/5)256=得到的二进制电压值，变成十进制数的话是逆过程，256/5=51,所以除以51得到十进制电压

DIV DL ;平均值除以51

XLAT

MOV [SI], AL ;求bcd 码送入对应BCD 单元

INC SI

MOV AL, AH

XOR AH, AH

MOV DL, 10

MUL DL

LOOP LOOP2 ;精确到小数点后两位

;A/D转换子程序

GET: MOV BX, 0

MOV CX, 5

LOOP1: MOV DX, 0640H ;AD转换器的端口地址

OUT DX, AL ;启动AD转换器

CALL DALLY ;等待延迟

IN AL, DX ;从AD转换器的端口地址读出数据

XOR AH, AH ;AX高八位AH清0

ADD BX, AX ;将读出的数据进行累加

LOOP LOOP1 ;读5次数据 进行累加

MOV AX, BX

MOV DL, 5 ;为了数据精确求取5次的平均值

DIV DL

MOV AVERAGE, AL ;送入变量单元内

LEA SI, AVERAGE

MOV BL, [SI]

;源程序，显示程序段

PRT: MOV DX, 06C6H ;初始化8255

MOV AL, 80H ;10000000B ABC输出方式

OUT DX, AL

MOV CX, 100H

LOOP3:

LEA SI, BCD

MOV AH, 0FEH ;置位码

LOOP4: MOV AL, AH

MOV DX, 06C2H ;初始化B口

OUT DX, AL

MOV AL, [SI]

CMP AH, 0FEH

JNZ NEXT1

OR AL, 80H ;最高位置1 小数点点亮

NEXT1:

MOV DX, 06C0H ;初始化A口

OUT DX, AL

CALL DALLY

ROL AH, 1

INC SI

CMP AH, 0F7H

JNZ LOOP4

LOOP LOOP3

RET

DALLY: PUSH CX ;延时程序

PUSH AX

MOV CX, 10H

A77:MOV AX, 10H ;连续减10次，进行延时

A88:DEC AX

JNZ A88

LOOP A77

POP AX

POP CX

RET

;报警程序段

JUG: PUSH CX

PUSH AX

MOV AL, AVERAGE ;取出之前存入的平均值，放进AL

CMP AL, 0E2H ;4.5V占比重0.9，256\*0.9=226 用16进制表示为0E2H

JNB NEXT2 ;超过4.5V的警戒值，启动报警

MOV DX, 0686H

MOV AL, 36H

OUT DX, AL

MOV AL,0

OUT DX,AL

JMP NEXT3

NEXT2:MOV DX, 0686H ;8254的端口初始化，报警

MOV AL, 36H ;00110110B 计数器0，读写高低位，采用方式3 二进制计数

OUT DX, AL

MOV AX,2000

MOV DX, 0680H ;计数器0 送初值

OUT DX, AL

MOV AL,AH

OUT DX,AL

CALL DALLY

NEXT3: ;正常

POP AX

POP CX

RET

CODE ENDS

END START

### 6 总结

在对实验原理和思想有了深刻的了解之后，才开始写程序。写的过程中当然也发现很多问题，根据查到的类似的程序进行修改和琢磨，也会向学习好的同学请教，慢慢的明白重点在哪里。可能程序本身并不完美，会有瑕疵，但是这次的实验让我更好的理解了微机原理这门课，我想这才是最终目的。