**南京信息工程大学 实验（实习）报告**

实验名称 8253和8259设计电子表 日期 2024.12.20 指导教师 岳键

系 计算机学院、网络空间安全学院专业 计算机科学与技术年级 2022级班次 6姓名张瑞晨学号 202283290159

## 实验目的

1.掌握中断控制器8259A与微处理机接口的原理和方法。

2.学习8253编程定时/计数器的工作方式。

3.掌握中断控制器8259A的应用编程。

## 二、实验内容

1.实验原理

本实验基于8253的定时器功能和8259中断控制器的中断管理能力，通过编程实现电子钟显示。

定时器部分：

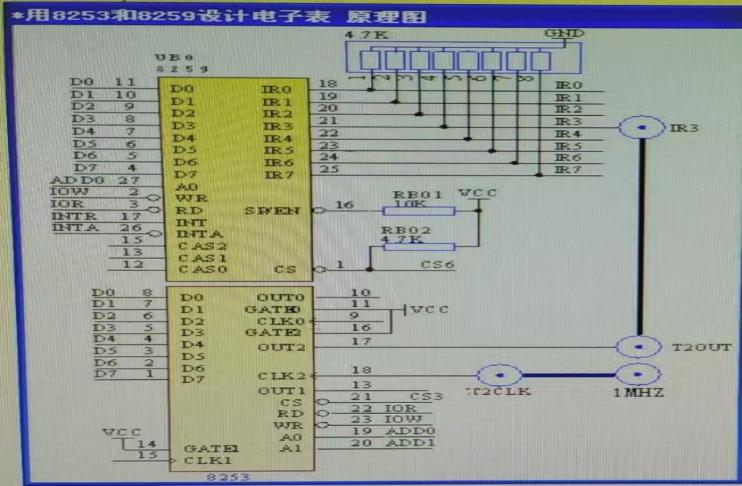
8253的通道2工作在方式3（方波发生器方式），输入时钟信号为1MHz。实验设定初值为0C350H，使得每隔1/20秒完成一次计数并生成方波。此方波输出至OUT2，引入8259作为中断信号。

中断控制器部分：

8259配置为单片模式，使用边沿触发，3号中断输入（IR3）与8253的OUT2连接。8259在接收到中断请求信号时，通知CPU进行中断服务。在中断服务程序中，记录中断次数，每累计20次（即1秒）更新电子钟的显示时间。如果信号不符要求，则转至7号中断（IR7），显示“ERR”提示。

计时计算：

输入1MHz的时钟信号，每次计数周期为1/1,000,000秒，设定计数初值为50000（即0C350H），可生成1/20秒的周期信号。



显示部分：

电子钟的时间缓冲区更新后，调用显示函数以更新实验仪显示器上的时间信息。

2.实验线路连接

1.8253A芯片的T2CLK引出插孔连分频输出插孔1MHZ。

2.8259A的IR3插孔和8253A的T20UT插孔相连。

1. 实验程序

CODE SEGMENT

ASSUME CS:CODE

INTPORT1 EQU 0020H

INTPORT2 EQU 0021H

INTQ3 EQU INTERUPT3

INTQ EQU INTERUPT7

CONTPORT EQU OODFH

DATAPORT EQU OODEH

DATA EQU 0500H

DATA EQU 0508H

DATA3 EQU 0518H

DATA4 EQU 0520H

DATA5 EQU 0528H

DATA6 EQU 0529H

DATAT EQU 052AH

DATAS EQU 052BH

DATA9 EQU 052CH

DATA10 EQU 052EH

ORG 1800H

START: JMP TINT1

TINT1: CALL FORMAT

CLD

MOV DX,004BH

MOV AL,OB6H

OUT DX,AL

MOV DX,004AH

MOV AL,5OH

OUT DX,AL

MOV AL,OC3H

OUT DX,AL

MOV SI,DATA3

CALL LEDDISP

MOV AX,OH

MOV DS,AX

CALL URINTVER

MOV AL,13H

MOV DX,INTPORT1

OUT DX, AL

MOV AL,O8H

MOV DX,INTPORT2

OUT DXAL

MOV AL,OBH

OUT DX,AL

MOV AL,OF7H

OUT DX,AL

WAITING: STI

THP WAITING

WRINTVER: MOV AX,OH

MOV ES,AX

MOV DI,DO2CH

LEA AX,INTQ3

STOSW

MOV AX,CS

STOSW

MOV DI,003CH

LEA AX,INTQ7

STOSW

MOV AX,CS

STOSW RET

INTERUPT3 : CLI

MOV AL,DS:[DATA6]

INC AL

MOV DS:[DATA6],AL

CMP AL,14H

JC PLAY

MOV BYTE PTR DS:[DATA6],00H

MOV SI,DATA4

CALL LEDDISP

CALL NEUTIME

CALL CONVERSALL

PLAY: MOV AL,20H

MOV DX,INTPORT1

OUT DX,AL I

STI IRET

INTERUPT7: CLI

MOV SI,DATA1

CALL LEDDISP

MV AL, 20H

OV DX,INTPORT1

OUT DX,AL

IRET

LEDDISP: AL,90H

OV DX,CONTPORT

OUT DX,AL

MOV BYTE PTR DS:[DATA5],00H

LED1: CMP BYTE PTR DS:[DATA5],07H

JA LED2

MOV BL,DS:[DATAS]

MOV BHH

MOV AL,CS:[BX+SI]

MOV DX,DATAPORT

OUT DX,AL

ADD BYTE PTR DS:[DATA5],01H

JNZ LED1

LED2: RET

FORMAT: MOV BX,0

MOV UORD PTR DS:[BX+DATA1],5050H

ADD BX,2

MOV VORD PTR DS:[BX+DATA1].0079H

ADD BX,2

MOV WORD PTR DS:[BX+DATA1],0000M

ADD BX,2

MOV WORD PTR DS:[BX+DATA1],0000H

ADD BX,2

MOV WORD PTR DS:[BX+DATA1],063FH

ADD BX,2

MOV WORD PTR DS:[BX+DATA1],4F5BH

ADD BX,2

MOV HORD PTR DS:[BX+DATA1],6D66H

ADD BX,2

MOV UORD PTR DS:[BX+DATA1],077DH

ADD BX,2

OV WORD PTR DS:[BX+DATA1],6F7FH

ADD BX,2

MOV UORD PTR DS:[BX+DATA1],7C77H

ADD BX,2

MOV WORD PTR DS:[BX+DATA1],5E39H

ADD BX,2

MOV WORD PTR DS:[BX+DATA1],7179H

ADD BX,2

MOV WORD PTR DS:[BX+DATA1],3F3FH

ADD Bx,2

MOV VORD PTR DS:[BX+DATA1],3F40H

ADD BX,2

MOV VORD PTR DS:[BX+DATA1],403FH

ADD BX,2

MOV HORD PTR DS:[BX+DATA1J,3F3FH

ADD BX,2

MOV ORD PTR DS:[BX+DATA1],3F3FH

ADD BX,2

MOV WORD PTR DS:[BX+DATA1],3F40H

ADD B,2

MOV WORD PTR DS:[BX+DATA1],403FH

ADD BX,2

MOV WORD PTR DS:[BX+DATA1],3F3FH

MOV BYTE PTR DS:[DATA6],D0D

MOV BYTE PTR DS:[DATA7],0D

MOV BYTE PTR DS:[DATAS]DOD

MOV BYTE PTR DS:[DATA9],00D

MOV WORD PTR DS:[DATA10],DATA4

RET

CONVERS1: MOV BH,OH

AND AL,OFH

MOV BL,AL

MOV AL,CS:[BX+DATA2]

RET

ONVERS2: PUSH AX

CALL CONVERS1

MOV BX,WORD PTR DS:[DATA10)

MOV DS:[BX],AL

ADD WORD PTR DS:[DATA10],01H

POP AX

PUSH CX

HOV CL4H

SHR AL,CL

POP CX

CALL CONVERS1

MOV BX,WORD PTR DS:[DATA10]

HOV DS:[BX],AL

RET

CONVERSALL: HOV AL,BYTE PTR DS:[DATA9]

CALL CONVERS2

ADD VORD PTR DS:[DATA101.02

MOV AL,BYTE PTR DS:[DATAB]

CALL CONVERS2

ADD WRD PTR DS:[DATA10].02H

MOV AL,BYTE PTR DS:[DATA7]

CALL CONVERS2

MOV VORD PTR DS:[DATA10],DATA4

RET

NEWTIME: CHP BYTE PTR DS:[DATA9],59H

JC ADD1

OV BYTE PTR DS:[DATA9],OD

CMP BYTE PTR DS:[DATAB),59H

TC ADD2

MOV BYTE PTR D3:[DATAB],OD

CMP BYTE PTR DS:[DATA7],23H

JC ADD3

MOV BYTE PTR DSI[DATA7],OD

JMP OK

ADD1: MOV AL,BYTE PTR DS:[DATA9]

ADD AL,1D

DAA

MOV BYTE PTR DS:[DATA9],AL

OK

ADD2: MOV AL,BYTE PTR DS:[DATAS]

ADD AL,1D

DAA

MOV BYTE PTR DS:[DATA8],AL

JMP OK

ADD3: MOV AL,BYTE PTR DS:[DATA7]

ADD AL,1D

DAA

MOV BYTE PTR DS:[DATA7],AL

OK: RET

CODE ENDS

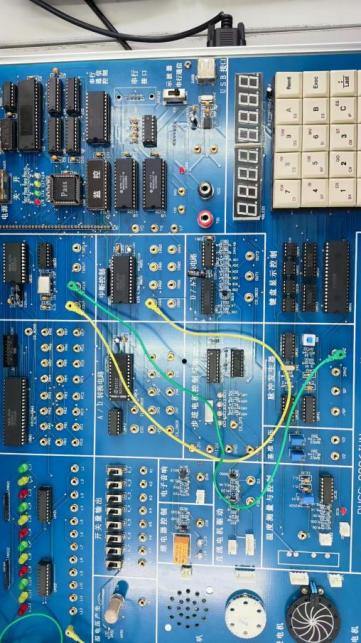
END START

三、实验步骤

**1.实验线路连接**：

①8253芯片的T2CLK端口连接到分频输出（1MHz）插孔。

②8259芯片的IR3端口连接到8253芯片的T2OUT端口。

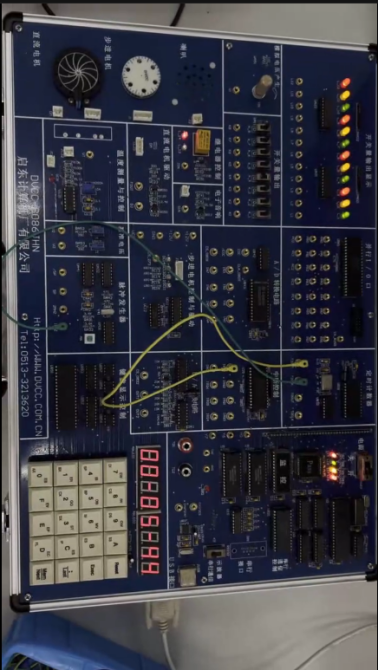


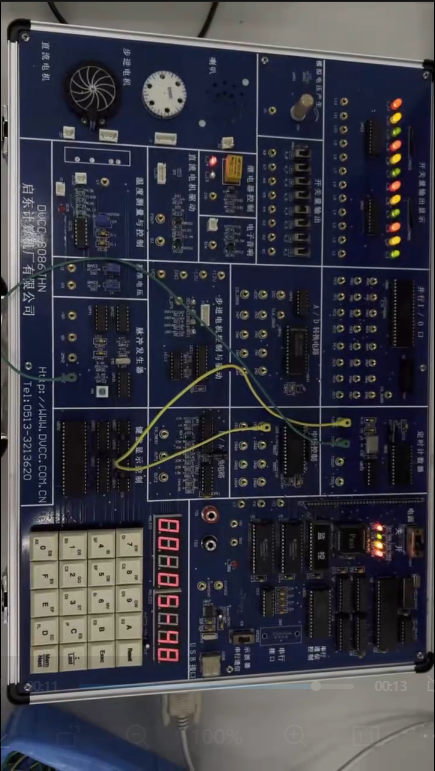
**2.程序运行**：

①加载实验程序，运行后，实验仪显示器上显示电子钟。

②通过观察秒数变化和ERR提示信号，验证实验效果。

结果如下图。





1. **实验总结**

通过本实验，进一步掌握了8253定时器的工作原理及其在生成定时信号中的作用。同时，学会了通过8259A中断控制器对中断请求的处理方法，并理解了中断处理程序的结构和设计技巧。实验中成功实现了电子钟的功能，较好地整合了硬件编程和中断服务的理论与实践。