



重慶理工大學

实验报告

实验课堂表现			实验报告成绩	实验总成绩
A ()	B ()	C ()		

实验名称: 新发继电器接口实验

专业班级: 121070204

学 号: 12107980106

姓 名: 刘心鑫

联系电话: _____

指导老师: 黄刚

实验时间: 17周周五 8:30-11:30 (2023.6.9)

电气与工程学院 电工电子技术实验中心

【成绩】

【教师签名】

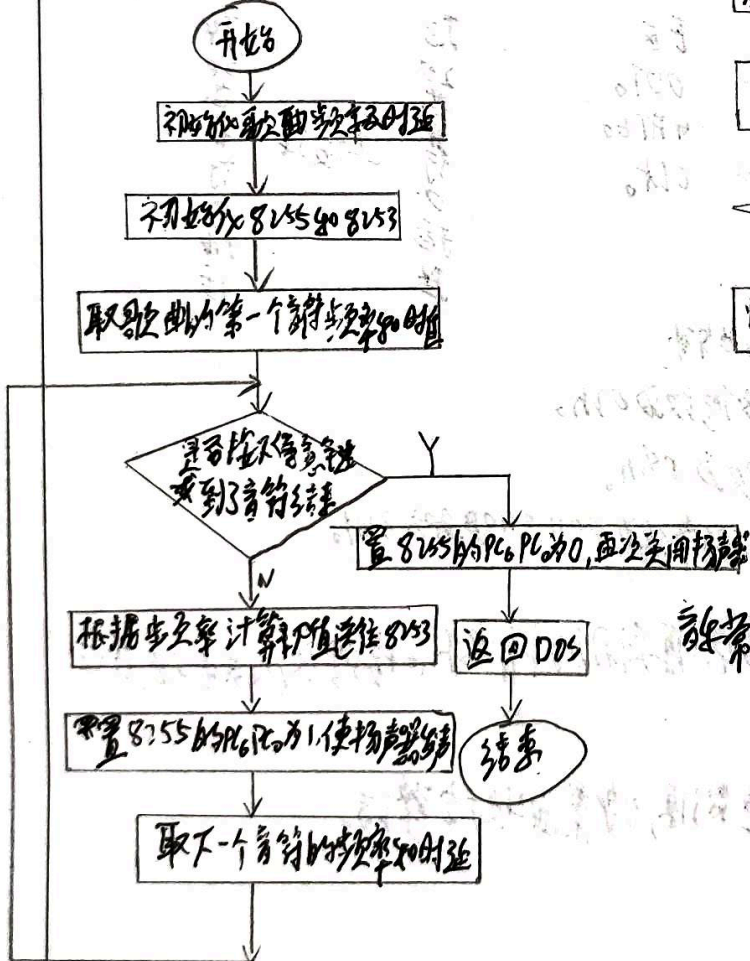
【实验目的】

通过音乐发生器实验,学习如何利用8253定时/计数器进行声音控制电路的设计原理与方法。

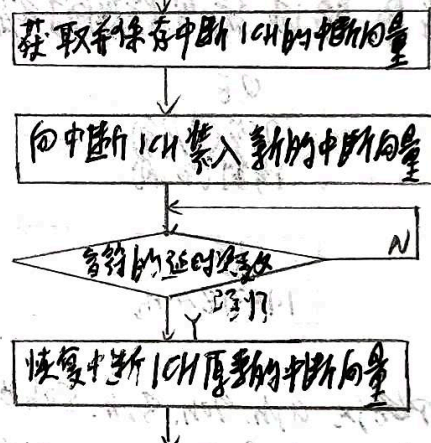
【实验原理及内容】

音乐发生器驱动模块电路原理如图 2.1.12 所示。模块包含喇叭 SPK, LM386A, 74LS08 以及 74LS245 和 LED 等。

主程序流程图:



延时子程序



常识: (1) 音调, 每个音符发声的高低即音符的频率

(2) 节拍, 每个音符发声的长短, 音符的持续时间, 即时延。

【实验设备】

将装有实验平台的计算机 1台
实验设备箱 1台
导线 若干

【实验方案及步骤】

1. 硬件连接：跳线设置：将电源模块的JP7跳接。

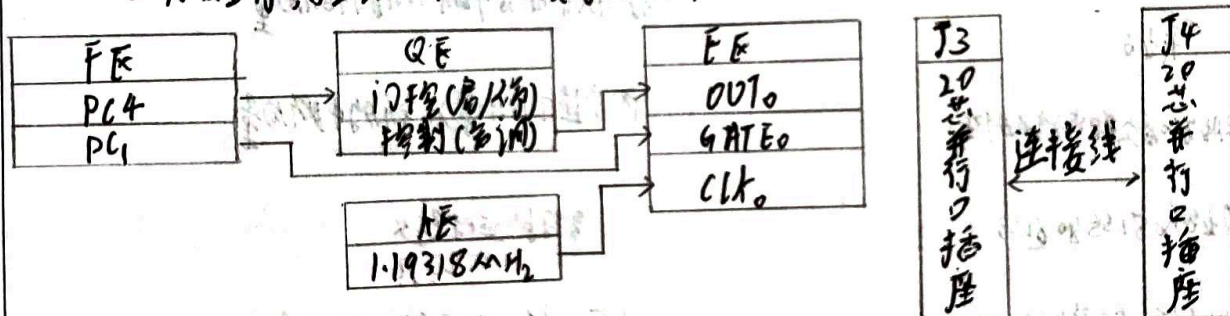
2. 硬件检测及程序设计

(1) 在软件集成环境下进行硬件检测，达到初始化芯片的目的。

(2) 根据参考电路，设计电路连线，并检查其正确性。

(3) 对8253A芯片初始化，并编写完整程序，实现实验要求的所有功能。

(4) 运行程序，验证编写程序的正确性，观察实验现象得出结论。



3 (1) 8253的初始化，90h, 90h, 82h, 80h均可。

(2) 关闭8253的T0计数器：即PC1=0，故修改为02h。

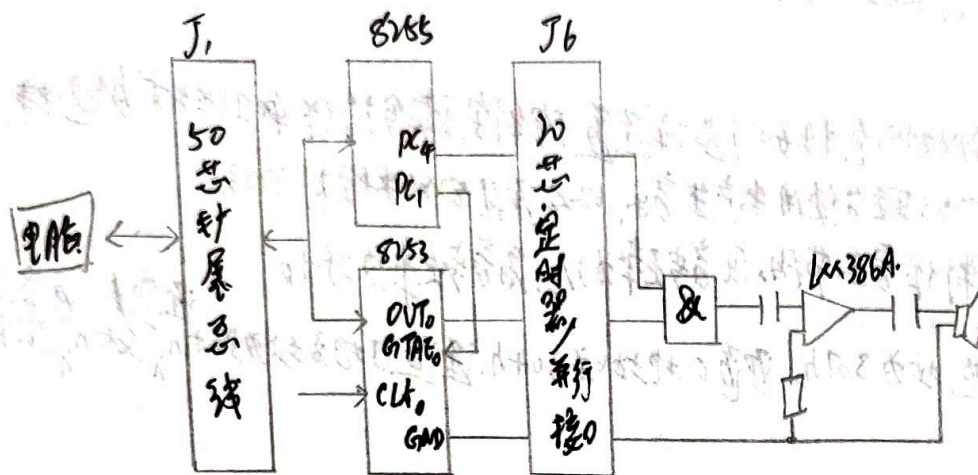
(3) 关闭喇叭，即置PC4=0，故修改为08h。

(4) 8253的初始化，由于使用了通道0，故修改为00110110B，即为36h。

4 将修改好的程序并检查无误后，在实验箱电源关闭的情况下，连接修改好的电路，并运行程序，验证蜂鸣器是否发声。

5 实验完成后，关闭电脑与实验箱电源，将桌面收拾整洁。

【实验电路图】



【实验数据处理及分析】

(1) 将“卡袋”简谱中的一段输入程序库，什码后附。

(2) 8255的初始值:

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
1	0	0	X	0	0	X	0

故 92h, 90h, 82h, 80h 均可。

(3) 关闭与打开 8255 的计数器。(PC₁=0/1)

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
0	0	0	0	0	0	1	0	→ 02H
0	0	0	0	0	0	1	1	→ 03H

(4) 关闭与打开喇叭 (PC₄=0/1)

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
0	0	0	0	1	0	0	0	→ 08H
0	0	0	0	1	0	0	1	→ 09H

(5) 8255的初始值。

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
0	0	1	1	0	1	1	0	→ 36H

【实验结论】

- (1) 通过音乐发生器实验, 学习了如何利用 8253 定时计数器进行音频控制电路的设计原理方法。
- (2) 学会了 8253 的初始化, 包括如何选择通道控制字, 读写操作和工作方式的选择。
- (3) 学会了 8255, 8253 联合使用来产生音乐, 以及通过 8255 来控制 8253。
- (4) 掌握了如何制作音乐节拍, 低音频率为 f_1 , 高音频率为 $2f_1$ 。
- (5) 8253 控制口地址为 $307h$, 通道 0 地址为 $304h$, 通道 1 地址为 $305h$, 通道 2 地址为 $306h$ 。

【思考题】

卡农

dylanf 歌手
Pachelbel 作曲

1 = C $\frac{4}{4}$

canon

$\dot{3} - \dot{2} - | \dot{1} - 7 - | 6 - 5 - | 6 - 7 - | \dot{1} - 7 - | 6 - 5 - | 4 - 3 - |$
 $4 - 2 - | 1 \ 3 \ 5 \ 4 | 3 \ 1 \ 3 \ 2 | 1 \ 6 \ 1 \ 5 | 4 \ 6 \ 5 \ 4 | 3 \ 1 \ 2 \ 7 | \dot{1} \ \dot{3} \ \dot{5} \ 5 |$
 $6 \ 4 \ 5 \ 3 | 1 \ \dot{1} \ \dot{1} \ \underline{7} | \underline{\dot{1} \ 7} \ \underline{\dot{1} \ 1} \ \underline{7 \ 5} \ \underline{2 \ 3} | \underline{1 \ \dot{1}} \ \underline{7 \ 6} \ \underline{7 \ 3} \ \underline{5 \ 6} | \underline{4 \ 3} \ \underline{2 \ 4} \ \underline{3 \ 2} \ \underline{\dot{1} \ 7} |$
 $\underline{6 \ 5} \ \underline{4 \ 3} \ \underline{2 \ 4} \ \underline{3 \ 2} | \underline{1 \ 2} \ \underline{3 \ 4} \ \underline{5 \ 2} \ \underline{5 \ 4} | \underline{3 \ 6} \ \underline{5 \ 4} \ \underline{5 \ 4} \ \underline{3 \ 2} | \underline{1 \ 6} \ \underline{6 \ 7} \ \underline{\dot{1} \ 7} \ \underline{6 \ 5} |$
 $\underline{4 \ 3} \ \underline{2 \ 6} \ \underline{5 \ 6} \ \underline{5 \ 4} | 3 \ \dot{3} \ \dot{2} - | \dot{1} \ \dot{3} - \overset{\frown}{6} | \overset{\frown}{6} \ 5 - \overset{\frown}{6} | 6 - 7 - |$


```

out dx,al
mov al,02h
mov dx,303h
out dx,al
mov al,08h
mov dx,303h
out dx,al
mov al,36h
mov dx,307h
out dx,al

```

;关闭 8253 的 T2 计数器

;关闭喇叭 (8255 的 PC4=0)

;8253 初始化

```

mov dx,offset bg
mov ah,9
int 21h

```

;显示提示信息

```

mov si,offset freq
mov bp,offset time

```

;设置频率指针→si

;设置延时指针→bp

```

pp: mov ah,0bh
int 21h

```

;检测是否有任意键按下?

```

cmp al,00
jne end_sing

```

;al=0ffh/al=0

;若有, 则结束演奏

```

mov di,[si]
cmp di,0
je end_sing

```

;取 1 个音符的频率→di

;检测是否是乐曲结尾

;若是, 则结束演奏

```

mov dx,12h
mov ax,34dch
div di
mov dx,304h

```

;计算音符的频率所对应的 8253 计数初值

;1.19318MHz=1234dch

T1 305h T2 306h

;向 8253 装入计数初值 T0 304h

```

out dx,al
mov al,ah
out dx,al

```

;先装低字节

;再装高字节

```

mov dx,303h
mov al,09h
out dx,al
mov al,03h
out dx,al

```

;打开喇叭 (8255 的 PC4=1)

;打开 8253 的 T2 (PC1=1), 开始发声

mov bx,ds:[bp] ;bp/sp 做指针会在堆栈段 SS 寻址, 故段超越 ds, 取 1 个音符的延时次数→bx

call time_delay	;调用延时子程
mov dx,303h	;关闭 8253 的 T2 (PC1=0)
mov al,02h	
out dx,al	
mov al,08h	;关闭喇叭 (PC4=0)
out dx,al	
add si,2	;取下 1 个音符的频率
add bp,2	;取下 1 个音符的延时次数
jmp pp	;继续
end_sing:	
mov dx, 303h	;关闭 8253 的 T2
mov al, 0ch	
out dx, al	
mov al, 00h	;关闭喇叭
out dx, al	
pop bx	;寄存器出栈
pop bp	
pop si	
pop di	
mov ax,4c00h	;返回 DOS
int 21h	
star endp	
new_intr proc	
dec bx	;新中断服务程序
iret	;该服务程序只将延时次数-1
new_intr endp	
time_delay proc near	
pushf	;延时子程序
push ds	;压栈
push es	
push bx	
mov ah,35h	;取原中断 (INT1CH) 的向量, 并保存
mov al,1ch	
int 21h	
mov old_off, bx	
mov bx, es	


```
mov old_seg, bx
pop bx
pop es
push ds
```

;设置新中断服务程序的向量

```
mov ah, 25h
mov al, 1ch
mov dx, seg new_intr
mov ds, dx
lea dx, new_intr
int 21h
pop ds
TD:  cmp bx, 0h
     jnz TD
```

;延时次数已到?
;未到, 则继续

;已到, 则恢复原中断向量

```
mov ah, 25h
mov al, 1ch
mov dx, old_seg
push dx
mov dx, old_off
pop ds
int 21h
```

```
pop ds
popf
;time delay end
```

```
ret
time_delay endp
```

```
code ends
end star
```

;程序结束