

# 《微机系统与接口技术课程设计》

## 实验报告



实验名称 多接口系统设计

班 级 2015211307

学 号 2015211906/2015211338

姓 名 王睿嘉/赵紫君

指导教师 张 杰

# 北京邮电大学课程设计报告

课程设计名称	微机系统与接口技术课程设计		学院	计算机学院	指导教师	张杰
班级	班内序号	学号		学生姓名	成绩	
2015211307	31	2015211906		王睿嘉		
2015211307	28	2015211338		赵紫君		
课程设计内容	实验内容：利用两台 TPC-ZK 微机原理实验平台实现两机间通信 实验目的： <ul style="list-style-type: none"><li>通过本实验了解串行通信原理；</li><li>理解计算机软件、硬件如何配合工作，如何对计算机进行功能扩展；</li><li>学会一般接口电路设计过程和驱动代码编写方法；</li><li>学会如何进行软件、硬件调试。</li></ul> 实验方法：编程和上机实验结合，小组合作					
学生课程设计报告 (附页)	详见课程设计报告					
课程设计成绩评定	遵照实践教学大纲并根据以下四方面综合评定成绩： <ul style="list-style-type: none"><li>1、课程设计目的任务明确，选题符合教学要求，份量及难易程度；</li><li>2、团队分工是否恰当与合理；</li><li>3、综合运用所学知识，提高分析问题、解决问题及实践动手能力的效果；</li><li>4、是否认真、独立完成属于自己的课程设计内容，课程设计报告是否思路清晰、文字通顺、书写规范。</li></ul> 评语：  成绩：  指导教师签名：  年 月 日					

## 目录

<b>一、实验目的 .....</b>	<b>4</b>
<b>二、实验内容 .....</b>	<b>4</b>
2.1 硬件部分 .....	4
2.2 软件部分 .....	4
2.3 多接口系统框图 .....	5
<b>三、实验原理 .....</b>	<b>5</b>
3.1 模块设计 .....	5
3.2 实验原理图 .....	7
<b>四、实验程序 .....</b>	<b>12</b>
<b>五、实验结果 .....</b>	<b>20</b>
5.1 单机通信模式 .....	20
5.2 双机通信模式 .....	20
<b>六、实验心得 .....</b>	<b>21</b>
6.1 遇到的问题及解决方法 .....	21
6.2 收获体验 .....	21

## 一、实验目的

进一步加深和巩固对所学知识点的理解和掌握；  
综合运用所学知识、工程原理来分析和完成本课程设计；  
理解计算机软件、硬件如何配合工作，能对计算机硬件进行功能扩展；  
掌握接口电路设计过程和驱动程序的编写方法；  
学会如何区分系统中软件、硬件故障和调试手段；  
培养计算机软硬件的综合设计能力和调试能力。

## 二、实验内容

### 2.1 硬件部分

4x4 小键盘接口设计（利用 8255 C 口）；  
异步串行通信接口设计（8254 为波特率发生器）；  
4 位数码管显示器设计。

### 2.2 软件部分

键盘扫描程序，将按键 0~F 转换成 0~15 的二进制值；

串行接口通信程序，发送/接收均为查询方式；

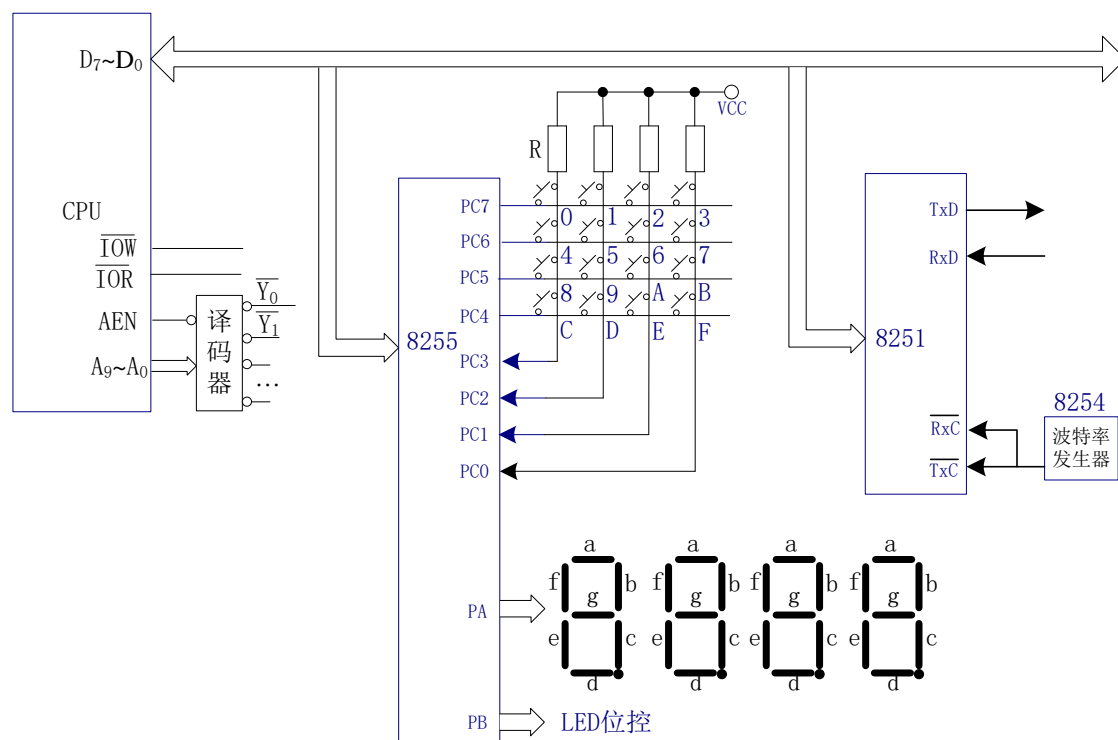
4 位七段数码管显示程序，左移位方式显示 0~9、A~F；

将串行接口发送端和接收端环回，当在 4x4 小键盘上敲击按键时，经串口发送、接收后将对应键值显示在 4 位数码管上。（要求：依次敲击 1、2、3、4、5，则依次显示 xxx1、xx12、x123、1234、2345）；

两组间进行双机通信，一方用小键盘键入数字，另一方接收到后在液晶屏上显示；

在液晶显示器上实现显示功能。

## 2.3 多接口系统框图



## 三、实验原理

### 3.1 模块设计

#### 1. 扫描矩阵式键盘

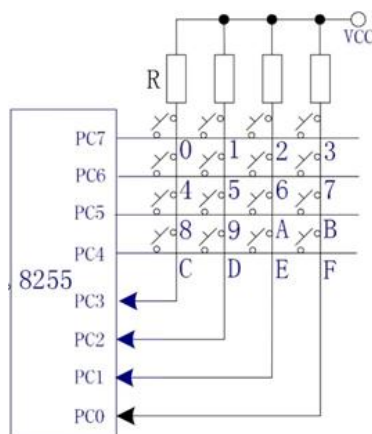
这是一个4行4列的矩阵式键盘，采用可编程并行接口芯片8255完成输入/输出。其中，8255的C口高4位用作输出，低4位用作输入，4条输出行线和4条输入列线构成16个交叉点，每个交叉点处有一个按键开关，从而构成一个具有16个按键的键盘。

采用行扫描法判断键盘按下键的位置。先将全部行线置为低电平，判断键盘中是否有键按下（检测列线是否有低电平），再逐行输出低电平，判断按下键的具体行（检测列线是否有低电平），最后循环左移自列线读入值，确定按下键的列（低电平对应列）。

## 2. 可编程并行接口 (8255)

8255A 芯片是由数据总线缓冲器、读/写控制逻辑、A、B、C 3 个双向 I/O 数据端口及相关控制电路组成。在实验中，使用 8255A 来完成矩阵式键盘的输入/输出及七段数码显像管/LCD 液晶屏显示的控制和数据输入。

8255 的 C 口高 4 位用作矩阵式键盘输出接口，低 4 位用作矩阵式键盘输入接口。在七段数码显像管中，A 口 8 位用作数据输入接口，B 口高 4 位用作位控输入接口。在 LCD 液晶屏中，A 口 8 位用作数据输入接口，B 口低 3 位用作控制接口。



## 3. 串行接口 (8251)

8251 是可编程串行通信接口芯片，可以实现串行同步通信，也可以实现串行异步通信。此次实验中，采用异步通信方式，因为考虑到双机通信时，两个实验台使用同一时钟源较为麻烦，且单机通信时，同步方式会出错，故采用异步方式更为灵活。

实验中的自发自收、双机通信皆使用 8251 芯片，采用异步通信，1 比特停止位，无校验，8 比特数据位，波特率因子为 64。

发送数据口 TxD，接收数据口 RxD。自发自收时，将自己实验台的 TxD 和 RxD 相连即可；双机通信时，两实验台的 TxD 和 RxD 交叉相连即可。

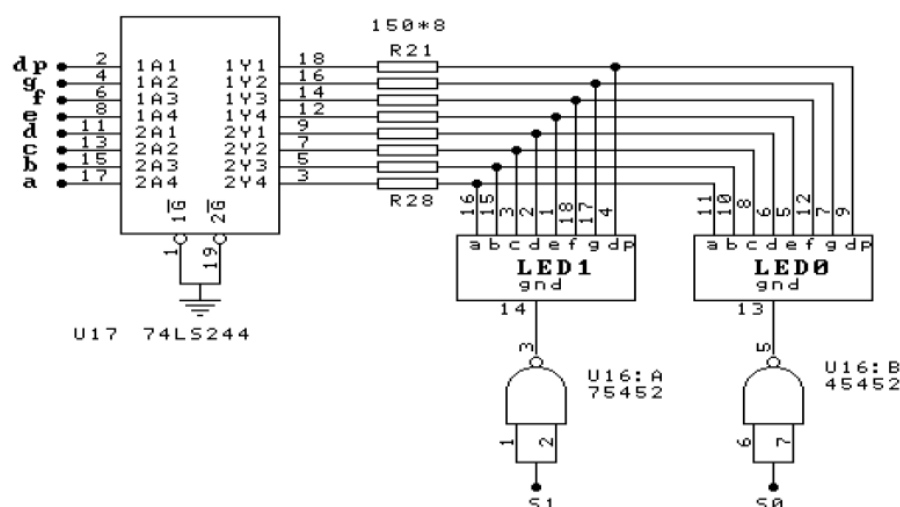
## 4. 定时与计数 (8254)

8254 芯片在该实验中用作频率发生器（方式 2），二进制计数，给串行接口芯片 8251 提供工作时钟。在本次实验中，使用 8254 的计数器 0。提供收发时钟是为了让 8251 正常进行收发工作。

## 5. 七段数码显像管

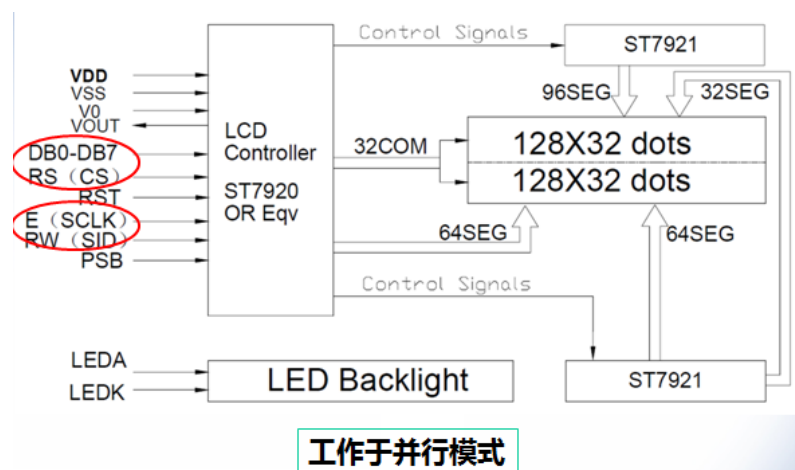
使用 8255 与七段数码显像管连接，移位显示发送的数据。在实验中，8255

通过 A 口传送数据到七段数码显像管, B 口传送位控信号。需要注意的是, 在送下一位数码显像管的数据和位控信号前需要先对之前的位控信号清零, 从而防止数据重叠显示。



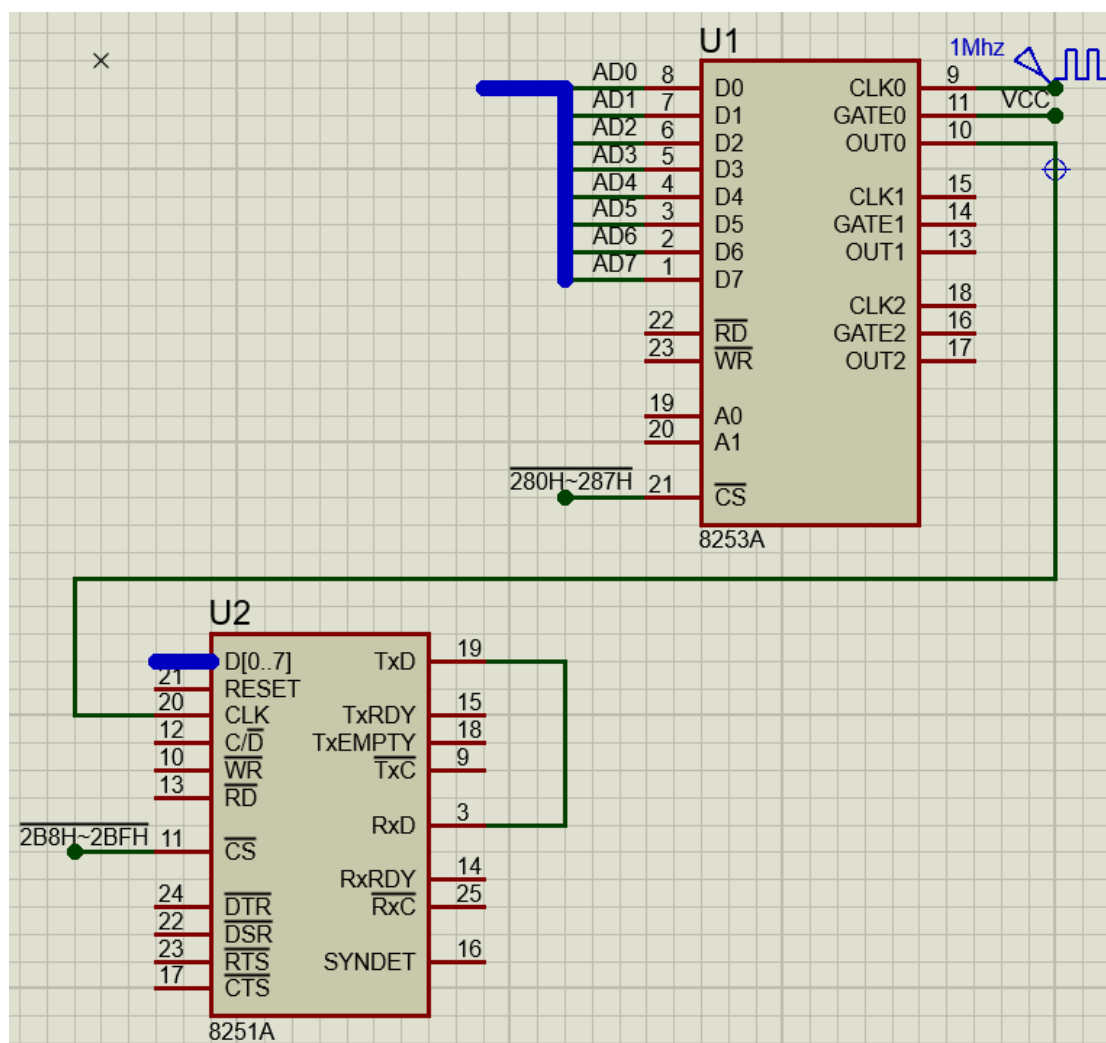
## 6. 液晶屏显示

使用 8255 与 128X64 LCD 模块连接, 显示接收的数据。在实验中, 采用半宽字型进行显示, 将 8 位元写入 DDRAM 中, 范围为 02H—7FH。其中数字 0—9 的编码为 30H—39H, 字母 A—F 的编码为 41H—46H, 即 ASCII 码。

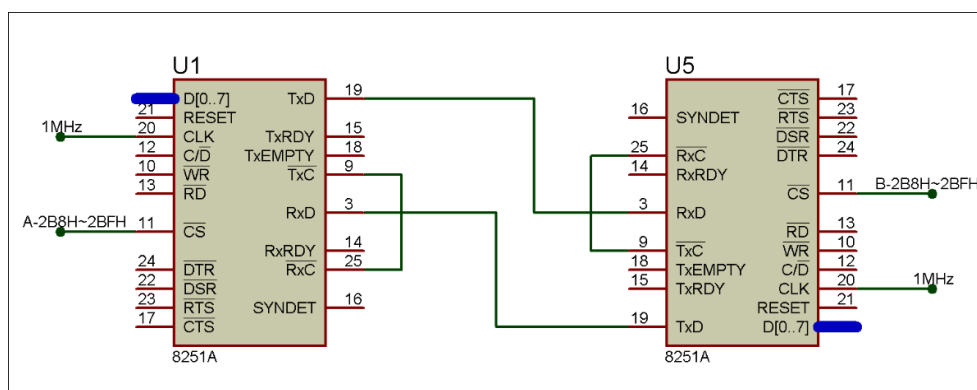


## 3.2 实验原理图

### A. 8251、8254 单机通信

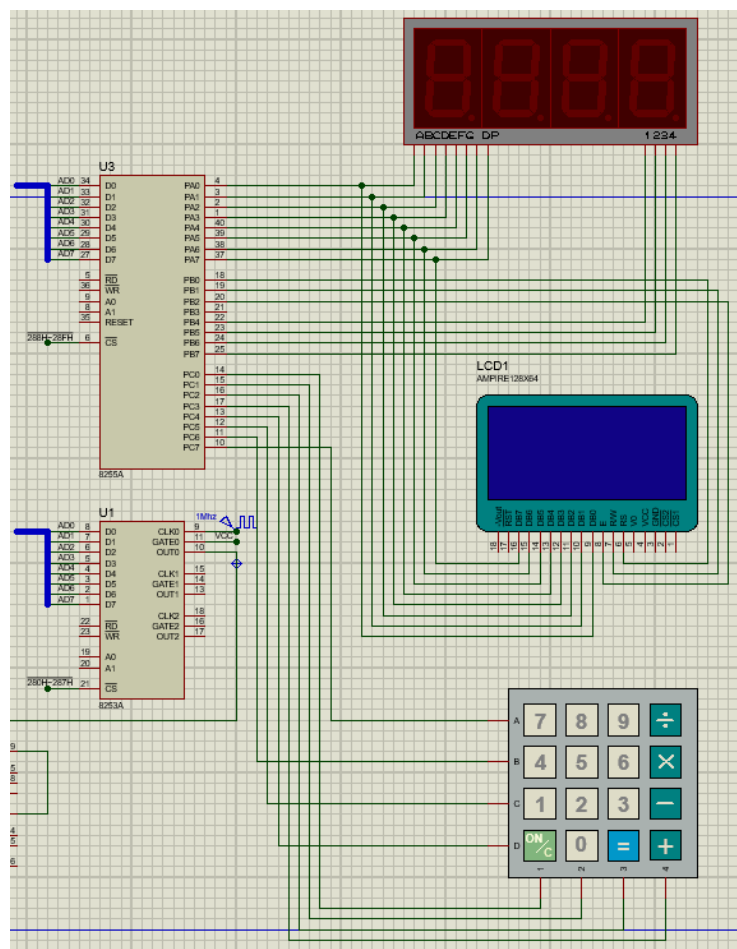


## B. 8251 双机通信

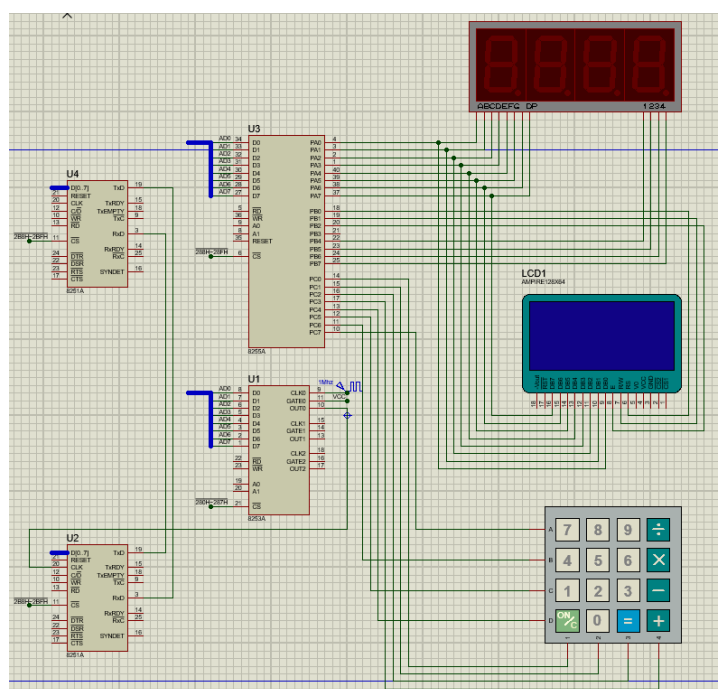


## C. 8255、矩阵式键盘、LCD 液晶显示屏、七段数码显像管



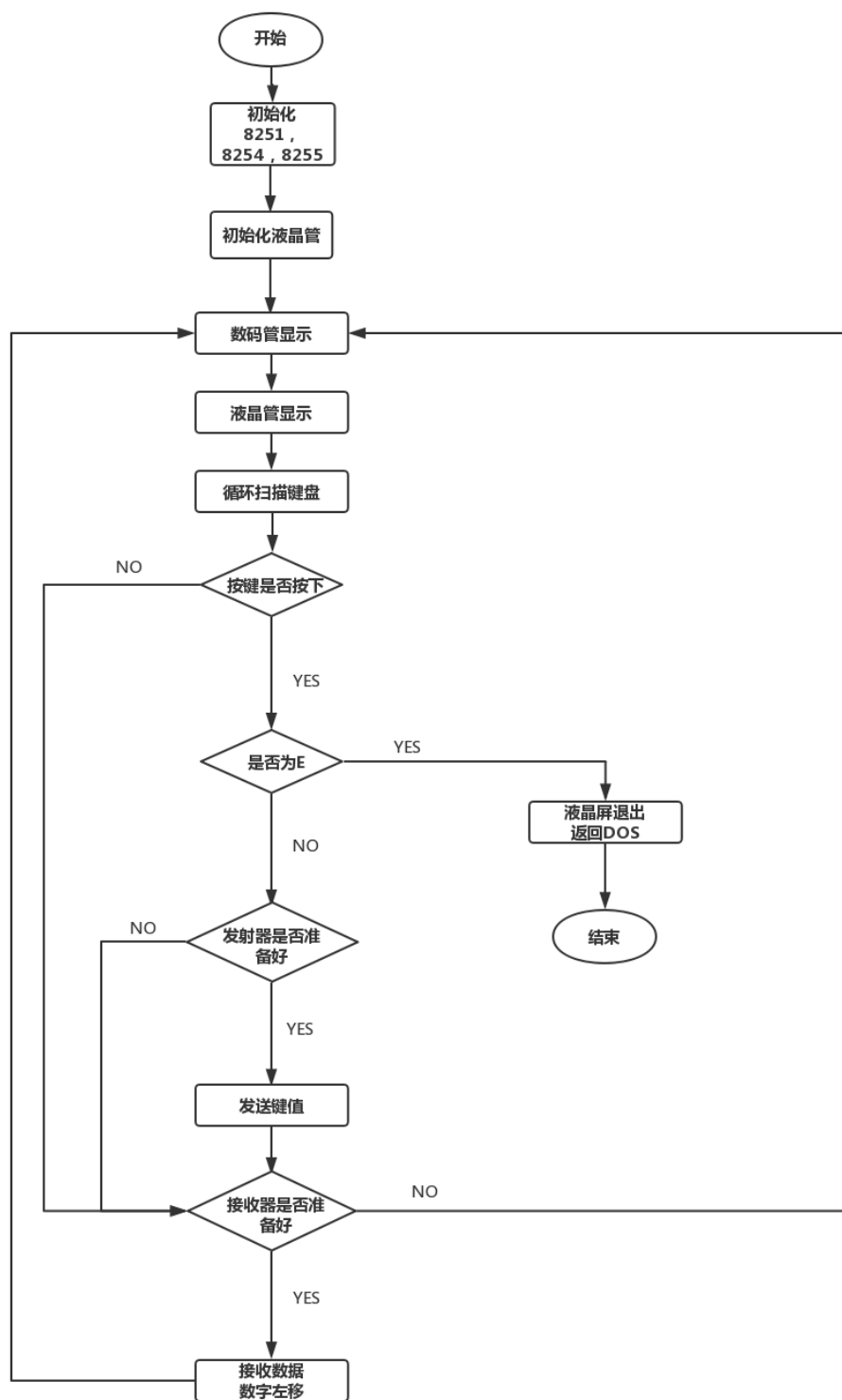


D.总接线图

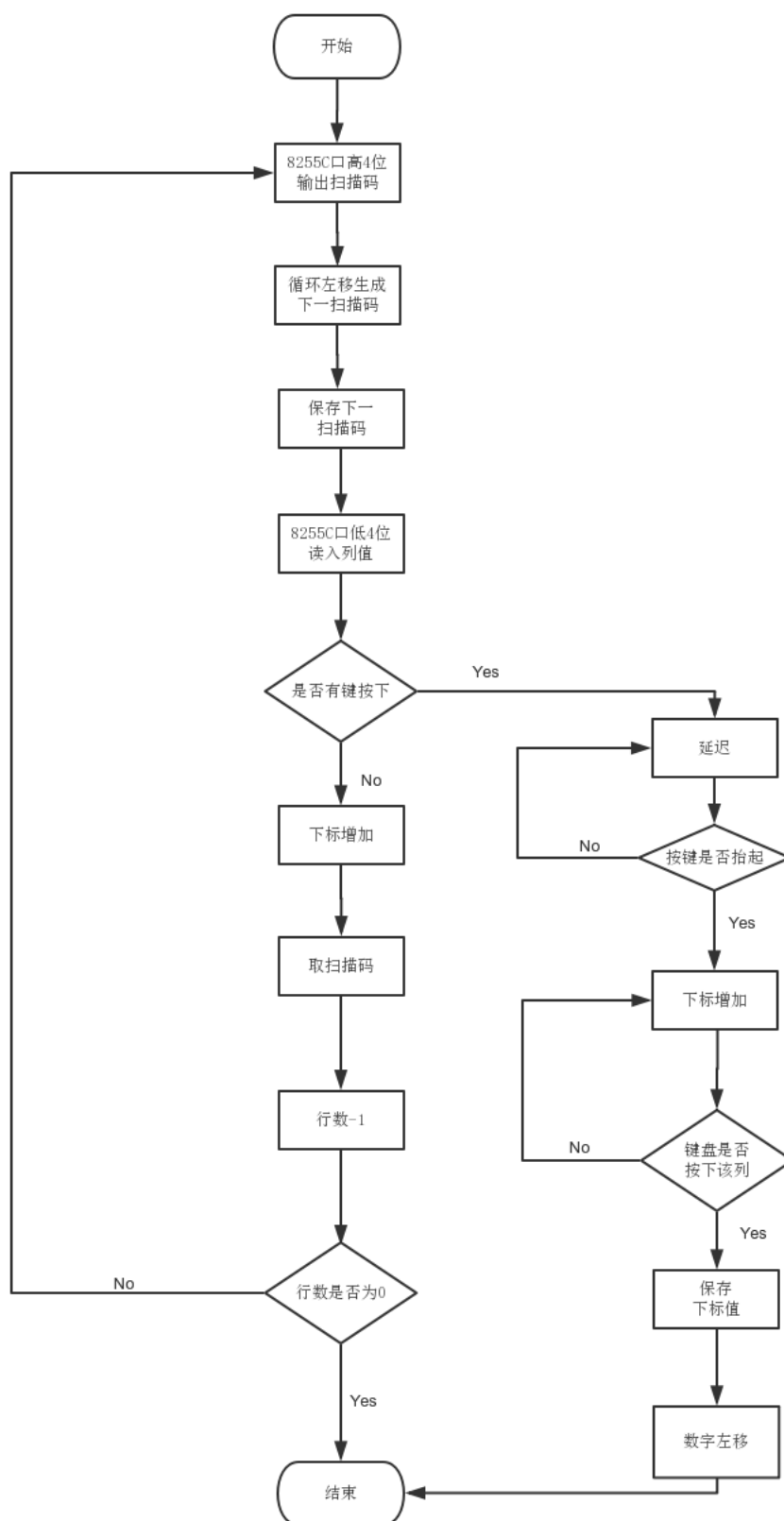


### 3.3 实验流程图

#### 1. 主程序



## 2. 键盘扫描程序



## 四、实验程序

```

001 ; 常量
002 C8251PORTIO EQU 2B8H ;8251 IO端口
003 C8251PORTCTRL EQU 2B9H ;8251 控制端口
004 C8254PORT0 EQU 280H ;8254 端口0
005 C8254PORTCTRL EQU 283H ;8254 控制端口
006 C8255PORTA EQU 288H ;8255 A口
007 C8255PORTB EQU 289H ;8255 B口
008 C8255PORTC EQU 28AH ;8255 C口
009 C8255PORTCTRL EQU 28BH ;8255 控制端口
010
011 ; 数据段
012 DATA SEGMENT
013 KEY DB 71H,79H,5EH,39H,7CH,77H,6FH,7FH,07H,7DH,6DH,66H,4FH,5BH,06H,3FH ;数码管
014 ASCII DB 46H,45H,44H,43H,42H,41H,39H,38H,37H,36H,35H,34H,33H,32H,31H,30H ;ASCII码
015 BUFFER1 DB 00H,00H,00H,00H
016 BUFFER2 DB 00H,00H,00H,00H
017 LCD_END DB 45H,4EH,44H ;END
018 LINEKEY DB ?
019 DATA ENDS
020
021 ; 代码段
022 CODE SEGMENT
023 ASSUME CS:CODE,DS:DATA
024
025 START:
026 MOV AX,DATA
027 MOV DS,AX ;DS指向数据段
028
029 ; 初始化8251
030 MOV DX,C8251PORTCTRL
031 XOR AL,AL
032 OUT DX,AL ;预初始化
033 MOV AL,40H
034 OUT DX,AL ;软复位
035
036 NOP ;延时
037
038 MOV AL,01001111B ;方式命令字: 异步方式, 1bit停止位, 无校验, 8bit数据位, 波特率因子64
039 MOV DX,C8251PORTCTRL
040 OUT DX,AL
041
042 MOV AL,00100111B ;工作命令字: 正常操作, 请求发送, 接受发送允许
043 OUT DX,AL
044
045 ; 初始化8254
046 MOV AL,00010100B ;控制字: 计数器0, 低字节, 方式2, 二进制计数
047 MOV DX,C8254PORTCTRL
048 OUT DX,AL
049
050 MOV AX,24H ;计数初值
051 MOV DX,C8254PORT0
052 OUT DX,AL
053
054 ; 初始化8255
055 MOV AL,10000001B ;方式选择控制字: A组方式0, A口输出, C口高四位输出, B组方式0, B口输出, C组低四位输入
056 MOV DX,C8255PORTCTRL
057 OUT DX,AL
058
059 ; 初始化液晶屏
060 CALL CLEAR ;清屏
061
062 CALL DISPLAYONLCD ;显示
063
064 ; 循环扫描键盘及8251状态口
065 INPUT:
066 MOV AL,00H
067 MOV DX,C8255PORTC
068 OUT DX,AL

```

```

069 |
070 SCAN1:
071     MOV AL,00H
072     MOV DX,C8255PORTB
073     OUT DX,AL
074
075     CALL DISPLAY           ;数码管显示
076     CALL DISPLAYONLCD    ;LCD显示
077
078     MOV DX,C8255PORTC
079     IN AL,DX              ;读入列值
080     AND AL,0FH            ;取低4位
081     CMP AL,0FH
082     JZ SCAN2              ;无键按下
083
084     CALL DELAY2
085
086     MOV BL,4              ;行数
087     MOV BH,4              ;列数
088     MOV AL,0EFH           ;扫描码
089     MOV CH,0FFH           ;小键盘下标值
090
091 ROW:
092     MOV DX,C8255PORTC
093     OUT DX,AL              ;C口高四位输出扫描码
094     ROL AL,1              ;循环左移生成下一扫描码
095     MOV AH,AL              ;保存下一扫描码
096
097     MOV DX,C8255PORTC
098     IN AL,DX              ;C口低四位读入列值
099 |     MOV LINEKEY,AL
100     AND AL,0FH            ;取低四位
101     CMP AL,0FH            ;判断是否有列线为0
102     JNZ HANDLE            ;有键按下
103
104     ADD CH,BH              ;下标增加
105     MOV AL,AH              ;取扫描码
106     DEC BL                 ;行计数-1
107     JNZ ROW                ;行数不为0, 继续扫描
108
109     JMP SCAN2
110
111 HANDLE:
112     MOV DX,C8255PORTC
113     CALL DELAY
114     IN AL,DX
115     CMP AL,LINEKEY
116     JZ HANDLE              ;按键未抬起
117
118     MOV AL,LINEKEY
119
120 COL:
121     INC CH                  ;下标值增加
122     RCR AL,1
123     JC COL                  ;未按下
124
125     MOV AL,CH              ;保存下标值
126
127     CALL DISPLAYONSCREEN   ;屏幕显示
128
129 |     CALL MOVE1            ;数字左移
130
131 ;8251发送
132     CMP AL,01H
133     JZ EXIT                ;按下E, 退出
134
135     PUSH AX
136     MOV DX,C8251PORTCTRL
137     IN AL,DX
138     AND AL,01H

```

```

139 | POP AX
140 JZ SCAN2 ;发送器未准备好
141
142 MOV DX,C8251PORTIO
143 OUT DX,AL ;发送键值
144
145 SCAN2:
146 MOV DX,C8251PORTCTRL
147 IN AL,DX ;读状态口
148 AND AL,02H
149 JZ INPUT ;接收器未准备好
150
151 MOV DX,C8251PORTIO
152 IN AL,DX ;接收数据
153
154 CALL MOVE2 ;数字左移
155
156 JMP INPUT
157
158 EXIT:
159 CALL LCD_EXIT ;LCD退出
160 MOV AX,4C00H
161 INT 21H ;返回DOS
162 ;-----数码管显示-----
163 DISPLAY PROC
164 PUSH CX
165 PUSH AX
166 PUSH DX
167 PUSH SI
168
169 MOV AL,00H
170 MOV DX,C8255PORTB
171 | OUT DX,AL
172 MOV AL,BUFFER1[0]
173 AND AX,00FFH
174 MOV SI,AX
175 MOV AL,KEY[SI]
176 MOV DX,C8255PORTA
177 OUT DX,AL ;送入第一个数码管数据
178 MOV AL,80H
179 MOV DX,C8255PORTB
180 OUT DX,AL ;送入第一个数码管位控信号
181
182 CALL DELAY1
183
184 MOV AL,00H
185 MOV DX,C8255PORTB
186 OUT DX,AL
187 MOV AL,BUFFER1[1]
188 AND AX,00FFH
189 MOV SI,AX
190 MOV AL,KEY[SI]
191 MOV DX,C8255PORTA
192 OUT DX,AL ;送入第二个数码管数据
193 MOV AL,40H
194 MOV DX,C8255PORTB
195 OUT DX,AL ;送入第二个数码管位控信号
196
197 CALL DELAY1
198
199 MOV AL,00H
200 MOV DX,C8255PORTB

```

```

201 | OUT DX,AL
202 MOV AL,BUFFER1[2]
203 AND AX,00FFH
204 MOV SI,AX
205 MOV AL,KEY[SI]
206 MOV DX,C8255PORTA
207 OUT DX,AL ;送入第三个数码管数据
208 MOV AL,20H
209 MOV DX,C8255PORTB
210 OUT DX,AL ;送入第三个数码管位控信号
211
212 CALL DELAY1
213
214 MOV AL,00H
215 MOV DX,C8255PORTB
216 OUT DX,AL
217 MOV AL,BUFFER1[3]
218 AND AX,00FFH
219 MOV SI,AX
220 MOV AL,KEY[SI]
221 MOV DX,C8255PORTA
222 OUT DX,AL ;送入第四个数码管数据
223 MOV AL,10H
224
225 MOV DX,C8255PORTB
226 OUT DX,AL ;送入第四个数码管位控信号
227
228 CALL DELAY1
229
230 MOV AL,00H
231 | MOV DX,C8255PORTB
232 OUT DX,AL
233
234 POP SI
235 POP DX
236 POP AX
237 POP CX
238 RET
239 DISPLAY ENDP
240 ;-----屏幕显示-----
241 DISPLAYONSCREEN PROC
242 PUSH AX
243
244 AND AX,0FH
245 MOV SI,AX
246 MOV DL,ASCII[SI] ;按键对应ASCII码
247 MOV AH,02H
248 INT 21H
249
250 CALL DISPLAY ;数码管显示
251
252 POP AX
253 RET
254 DISPLAYONSCREEN ENDP
255 ;-----液晶屏显示-----
256 DISPLAYONLCD PROC
257 PUSH DX
258 PUSH AX
259 PUSH SI
260

```

```
261 ;显示第一个数字
262     MOV DX,C8255PORTA
263     MOV AL,96H
264     OUT DX,AL           ;设定显示位置
265
266     CALL LCD_CTRL       ;送入指令
267
268     MOV DX,C8255PORTA
269     MOV AL,BUFFER2[0]
270     AND AX,0FFH
271     MOV SI,AX
272     MOV AL,ASCII[SI]
273     OUT DX,AL
274
275     CALL LCD_DATA       ;送入数据
276
277     CALL DELAY
278
279 ;显示第二个数字
280     MOV DX,C8255PORTA
281     MOV AL,94H
282     OUT DX,AL           ;设定显示位置
283
284     CALL LCD_CTRL       ;送入指令
285
286     MOV DX,C8255PORTA
287     MOV AL,BUFFER2[1]
288     AND AX,0FFH
289     MOV SI,AX
290     MOV AL,ASCII[SI]
291     OUT DX,AL
292
293     CALL LCD_DATA       ;送入数据
294
295     CALL DELAY
296
297 ;显示第三个数字
298     MOV DX,C8255PORTA
299     MOV AL,92H
300     OUT DX,AL           ;设定显示位置
301
302     CALL LCD_CTRL       ;送入指令
303
304     MOV DX,C8255PORTA
305     MOV AL,BUFFER2[2]
306     AND AX,0FFH
307     MOV SI,AX
308     MOV AL,ASCII[SI]
309     OUT DX,AL
310
311     CALL LCD_DATA       ;送入数据
312
313     CALL DELAY
314
315 ;显示第四个数字
316     MOV DX,C8255PORTA
317     MOV AL,90H
318     OUT DX,AL           ;设定显示位置
319
320     CALL LCD_CTRL       ;送入指令
```



```
321 |
322     MOV DX,C8255PORTA
323     MOV AL,BUFFER2[3]
324     AND AX,0FFH
325     MOV SI,AX
326     MOV AL,ASCII[SI]
327     OUT DX,AL
328
329     CALL LCD_DATA           ;送入数据
330
331     CALL DELAY
332
333     POP SI
334     POP AX
335     POP DX
336     RET
337 DISPLAYONLCD     ENDP
338 ;-----液晶屏送入指令-----
339 LCD_CTRL     PROC
340     MOV DX,C8255PORTB
341     MOV AL,00000000B           ;写指令
342     OUT DX,AL
343
344     CALL DELAY
345
346 ;锁存
347     MOV AL,00000100B
348     OUT DX,AL
349
350     CALL DELAY
351 |
352     MOV AL,00000000B
353     OUT DX,AL
354
355     CALL DELAY
356
357     CALL DISPLAY           ;数码管显示
358
359     RET
360 LCD_CTRL     ENDP
361 ;-----液晶屏送入数据-----
362 LCD_DATA     PROC
363     MOV DX,C8255PORTB
364
365     MOV AL,00000001B           ;写数据
366     OUT DX,AL
367
368     CALL DELAY
369
370 ;锁存
371     MOV AL,00000101B
372     OUT DX,AL
373
374     CALL DELAY
375
376     MOV AL,00000001B
377     OUT DX,AL
378
379     CALL DELAY
380
```

```

381 | RET
382 LCD_DATA ENDP
383 ;-----液晶屏清屏-----
384 CLEAR PROC
385     MOV DX,C8255PORTA
386     MOV AL,00001100B
387     OUT DX,AL ;启动显示
388
389     CALL LCD_CTRL ;送入指令
390
391     MOV DX,C8255PORTA
392     MOV AL,00000001B
393     OUT DX,AL ;清屏
394
395     CALL LCD_CTRL ;送入指令
396
397     RET
398 CLEAR ENDP
399 ;-----液晶屏退出-----
400 LCD_EXIT PROC
401     CALL CLEAR
402
403     MOV SI,0
404     MOV BL,92H ;设定显示位置
405     MOV CX,3
406
407 LP:
408     MOV DX,C8255PORTA
409     MOV AL,BL
410     OUT DX,AL ;设定显示位置
411 |
412     CALL LCD_CTRL ;送入指令
413
414     MOV DX,C8255PORTA
415     MOV AL,LCD_END[SI]
416     OUT DX,AL
417
418     CALL LCD_DATA ;送入数据
419
420     INC SI
421     INC BL
422     LOOP LP
423
424     RET
425 LCD_EXIT ENDP
426 ;-----移位-----
427 MOVE1 PROC
428     PUSH AX
429
430     MOV AL,BUFFER1[2]
431     MOV BUFFER1[3],AL
432     MOV AL,BUFFER1[1]
433     MOV BUFFER1[2],AL
434     MOV AL,BUFFER1[0]
435     MOV BUFFER1[1],AL
436
437     POP AX
438
439     MOV BUFFER1[0],AL ;新数据放入BUFFER1[0]中
440

```

```
441 | RET
442 MOVE1 ENDP
443
444 MOVE2 PROC
445     PUSH AX
446
447     MOV AL,BUFFER2[2]
448     MOV BUFFER2[3],AL
449     MOV AL,BUFFER2[1]
450     MOV BUFFER2[2],AL
451     MOV AL,BUFFER2[0]
452     MOV BUFFER2[1],AL
453
454     POP AX
455
456     MOV BUFFER2[0],AL      ;新数据放入BUFFER2[0]中
457
458     RET
459 MOVE2 ENDP
460 ;-----延迟-----
461 DELAY PROC
462     PUSH CX
463
464     MOV CX,0FFFH
465 LP1:
466     LOOP LP1
467
468     POP CX
469     RET
470 DELAY ENDP
471 |
472 DELAY1 PROC
473     PUSH CX
474
475     MOV CX,24H
476 LP2:
477     CALL DELAY
478     LOOP LP2
479
480     POP CX
481     RET
482 DELAY1 ENDP
483
484 DELAY2 PROC
485     PUSH CX
486
487     MOV CX,60H
488 LP3:
489     CALL DELAY
490     LOOP LP3
491
492     POP CX
493     RET
494 DELAY2 ENDP
495
496 CODE ENDS
497     END START
```

## 五、实验结果

### 5.1 单机通信模式

矩阵式键盘扫描的结果通过 DOS 调用在电脑屏幕上显示。

数码管方面，启动程序时显示“FFFF”，以后每次按键后都左移位显示数据，当按键为“E”时，程序退出，数码管清空、全灭。如，当输入数据依次为“12345E”时，数码管上相应显示的结果为“FFFF”“FFF1”“FF12”“F123”“1234”“2345”，全灭。

液晶屏方面，启动程序时显示“FFFF”，以后每次按键后都左移位显示数据，当按键为“E”时，程序退出，液晶显示屏上显示“END”。如，当输入数据依次为“12345E”时，液晶显示屏上相应显示的结果为“FFFF”“FFF1”“FF12”“F123”“1234”“2345”“END”。

进行防抖动设计，在按键抬起的瞬间发送扫描到的读数，然后更新两个显示屏。

### 5.2 双机通信模式

矩阵式键盘扫描的结果通过 DOS 调用在电脑屏幕上显示。

数码管方面，A 机器按键的结果在 A 机器的数码显像管上左移位显示，B 机器按键的结果在 B 机器的数码显像管上左移位显示，显示原理同单机通信。当 A 机器按键为“E”，A 机器退出程序同时数码管全灭，此时 B 机器可以通过按键“E”退出程序。B 机器按键为“E”时，B 机器退出程序同时数码管全灭，此时 A 机器可以通过按键“E”退出程序。

液晶屏方面，A 机器按键的结果在 B 机器的液晶显示屏上左移位显示，B 机器按键的结果在 A 机器的液晶显示屏上左移位显示，显示原理同单机通信。当 A 机器按键为“E”，A 机器退出程序并显示“END”，此后 B 机器的按键结果不再显示在 A 机器上，此时 B 机器可以通过按键“E”退出程序。B 机器按键为“E”时，B 机器退出程序并显示“END”，此后 A 机器的按键结果不再显示在 B 机器上，此时 A 机器可以通过按键“E”退出程序。

防抖动方式及原理与单机通信模式相同。

## 六、实验心得

### 6.1 问题及解决方法

在本次实验中，遇到的主要问题有以下九点：

- ① 七段数码管及矩阵式键盘的接线。对于数码管而言，排线接反会导致显示错误，非预期结果。对于矩阵式键盘而言，行列线连接顺序有误，会导致按键下标值计算有误。经多次尝试，更改接线方式，该问题得到解决；
- ② ASCII 码。A 所对应的 ASCII 码为 41H，0 所对应的为 30H。查阅 ASCII 码表并记录在代码中时出错。该问题的出现完全是由于粗心大意，但却造成了一些意想不到的 BUG，反复检查代码才得以发现；
- ③ 行列线扫描方法。起初，不太了解矩阵式键盘的数据读取方法，借鉴网上代码，最终理解行线输出扫描码，列线读入电平值，并通过移位确定具体按键的过程；
- ④ 数码管的重叠显示。左移位显示数字，但在不使用位控信号的情况下，会出现重叠现象。最终，8255 B 口高 4 位用作数码管位控信号，通过利用熄灭，送入数据，点亮的方法解决了该问题；
- ⑤ GATE0 悬空。起初，8254 GATE0 悬空，未接 VCC，8251 非正常运行。与邻组同学讨论，并比对接线。不同实验台悬空情况对应电平可能不同，因而为了保险，将 GATE0 接高，解决了 8251 错误运行问题；
- ⑥ LCD 使用。在上学期的微机原理与接口技术实验中并未使用过该器件，不了解其使用方法。同时，存在疑惑：LCD 与七段数码管共用 8255 A 口，如何以相同数据完成不同编码方式的显示。查阅 LCD 使用手册，发现它具有锁存功能，从而能够实现与数码管的分时复用。
- ⑦ LCD 的接线。利用 8255 B 口低三位作为 LCD 的控制接口。起初，接线颠倒，导致 LCD 出现雪花屏，无法正常显示。更改接线后，该问题得到解决；
- ⑧ 数码管闪烁。在未增加 LCD 显示这一附加功能时，数码管正常运行。但加入后，由于延时的存在，导致数码管频繁闪烁，异常丑陋。通过在大循环增加数码管显示函数调用次数，该问题得到解决；
- ⑨ LCD 与数码管分别显示收发数据。根据老师建议做出如下更改：使用两个 BUFFER 分别存储收发数据，以供 LCD 与数码管分别使用。同时，将发送数据显示在主机屏幕上，进一步方便纠错，明晰收发过程。

### 6.2 收获体验

王睿嘉：

微机原理与接口技术课程已结束一学期，开始渐渐淡忘各种关键芯片的使用方式及汇编语言的编写方法。实践出真知，本次实验是对已学知识的复习和巩固，

同时,也是对课堂和书本的补充。实际情形与已了解的原理大体一致,但又复杂许多。

对于软件方面,我们不断完善代码逻辑,熟悉芯片工作流程,如初始化等,考虑更为精细的实现问题,如按键抬起检测、防抖动等。硬件方面,接线正确与否起着至关重要的作用。同时,由于硬件的不稳定性,浮现许多知识盲区中的BUG,经单步调试,进一步加深了对8086的理解。

我认为实验的难点之一在于理解各芯片的交互过程。万事开头难,起初梳理各芯片如何交接花费了较长时间。在理解后,连点成线,确定总体框架,代码的编写、硬件的连接也就水到渠成。

难点之二在于调试。本次实验整合了所学的重要芯片,且加入了未曾了解的新器件LCD。个人对于硬件的接触不是很多,已形成了一出BUG就检查代码的思维定式。然而此次,很多BUG都出在硬件上,这开拓了我的调试思路。

本次小学期的任务乍一看并不多,也不难实现。但在实际操作过程中,会存在眼高手低的情况,也遇到了不少问题。所幸,经过仔细检查、反复尝试,最终都得到了较好的解决,收获颇丰。

#### 赵紫君:

通过这次微机接口课程设计,我重新复习了微机接口的课程,对汇编语言的编写加深了理解。

由于距离微机接口的理论课教学已经过去了半年多,我对8086的各种芯片的使用方式已经汇编语言的语法等等都忘得差不多了,因此在开始编写代码之前,我和队友都先认真进行了复习。本次课程设计主要使用的器件包括数码管,8251,8254,8255,LCD以及4x4小键盘等等,之前都是对它们单独进行操作,这次的实验需要将它们组合在一起进行使用。经过了几天之后,完成了基本的功能,非常幸运地,我们完成了加分项LCD。这次的LED和LCD显示需要参考课本以外的资料,尤其是LCD,但是网上感觉有些指令比较难找,但是我们还是很幸运找到需要的资料了。

这次实验过程中,遇到了一些问题,有自身的代码问题,比如之前提到的,也有硬件问题,比如有一台实验机,只有A键和B键按下后没有显示,换了另外一台实验机,就完美运行了,说明是实验台的问题。所以底层语言比高级程序语言更难,也体现在硬件是否给力。

我借这次小学期的课程设计,重新回顾了硬件方面的知识,为日后的学习打下了良好的基础。同时也非常感谢队友,因为我自己的汇编学的并不熟练,机智可爱的队友教会了我许多东西,而且我自己有时候比较粗心,细心的队友也发现我连线的出错。因此,这次小学期收获颇丰!