

实验5：七段数码管的使用

小组成员:吕建瑶1811400,郑佶1811464,吴京1811440

实验内容

1. 手动位选和段选：将手动数字量输入模块的引脚接七段数码管显示模块的相应引脚,拨动用于位选和段选的开关,观察数码管的显示内容.
2. 程序位选和段选：用 8255A 作接口（端口使用自行确定,最好和键盘矩阵所用端口不要重复）,通过程序在数码管上显示一个 4 位十进制数（数据自定）.
3. 和 4×4 键盘矩阵结合：在 4×4 键盘矩阵上按一个键（0 ~ 9）,将该键对应的编号（0 ~ 9）在最低位数码管上显示出来（原有的显示被去掉）.要求编成循环结构,上述操作可重复进行.按非数字键,程序退出.
4. (选做) 在实验③的基础上改进：在 4×4 键盘矩阵上按一个键（0 ~ 9）,将原有的显示向左移一位,新输入的按键编号（0 ~ 9）在最低位数码管上显示出来；按“B”键原有显示向右移一位,最高位补 0 或不显示；按其他键程序退出.

程序代码

```
//2.c
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <bios.h>
#include <ctype.h>
#include <process.h>

void key(void);

//*****根据查看配置信息修改下列符号值*****
#define IOY0          0x3000
//*****
#define MY8255_A      IOY0 + 0x00*2
#define MY8255_B      IOY0 + 0x01*2
#define MY8255_C      IOY0 + 0x02*2
#define MY8255_MODE   IOY0 + 0x03*2

void main()
{
    int a = 18, b = 52;

    outp(MY8255_MODE, 0x80);

    while(1)
    {

        outp(MY8255_A, a);
        outp(MY8255_B, b);
        key();
    }
}

void key(void)
```

```
{  
    if (bioskey(1) != 0)  
    {  
        exit(0);  
    }  
}
```

```
//3.c  
#include <stdio.h>  
#include <stdlib.h>  
#include <conio.h>  
#include <bios.h>  
#include <ctype.h>  
#include <process.h>  
  
void dis(void);  
void key(void);  
void clear(void);  
void ccsan(void);  
void putbuf(void);  
void getkey(void);  
void delay(int time);  
  
//*****根据查看配置信息修改下列符号值*****  
#define IOY0 0x3000  
//*****  
#define MY8255_A IOY0 + 0x00*2  
#define MY8255_B IOY0 + 0x01*2  
#define MY8255_C IOY0 + 0x02*2  
#define MY8255_MODE IOY0 + 0x03*2  
  
char a[] = {0x3f, 0x06, 0x5b, 0x4f,  
            0x66, 0x6d, 0x7d, 0x07,  
            0x7f, 0x6f, 0x77, 0x7c,  
            0x39, 0x5e, 0x79, 0x71};  
  
char m[] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};  
  
char b[] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x00};  
int cc;  
int b_n=0;  
int n;  
  
void main()  
{  
    outp(MY8255_MODE, 0x81);  
    b_n = 5;  
  
    while(1)  
    {  
        dis();  
        clear();  
        ccsan();  
        if (cc)  
        {  
            dis();  
            delay(0x100);  
        }  
    }  
}
```

```

        delay(0x100);
        clear();
        ccscan();
        if (cc)
        {
            getkey();
        }
        key();
    }

void getkey(void)
{
    int i;
    int j = 0xfe;
    char buff[12];
    for(i=0;i<=3;i++)
    {
        outp(MY8255_A, j);

        itoa(j, buff, 2);

        printf("j: %s\n", buff);
        if ( !( (inp(MY8255_C)) & 0x01) )
        {
            n = i + 0;
            putbuf();
            return;
        }

        if ( !( (inp(MY8255_C)) & 0x02) )
        {
            n = i + 4;
            putbuf();
            return;
        }

        if ( !( (inp(MY8255_C)) & 0x04) )
        {
            n = i + 8;
            putbuf();
            return;
        }

        if ( !( (inp(MY8255_C)) & 0x08) )
        {
            n = i + 12;
            putbuf();
            return;
        }

        j <= 1;
    }
}

void ccsan(void)

```

```

{
    outp(MY8255_A, 0x00);
    cc = inp(MY8255_C);
    cc = (~cc) & 0x0F;
    //printf("cc: %d\n", cc);
}

void dis(void)
{
    int i;
    int j = 0xdf;
    char buff[12];
    outp(MY8255_A, j);
    if(b[0] > 9)
        exit(0);
    outp(MY8255_B, m[b[0]]);
    itoa(a[b[i]], buff, 2);
    printf("\t%s\t\n", buff);

    delay(0x100);
    j >>= 1;
    j |= 0x80;
}

void clear(void)
{
    outp(MY8255_B, 0x00);
}

void putbuf(void)
{
    b[0] = n;
    b_n--;
    if (b_n == -1)
    {
        b_n = 5;
    }

    dis();
    clear();
    ccsan();
    while (cc)
    {
        dis();
        clear();
        ccsan();
    }
}

void key(void)
{
    if (bioskey(1) != 0)
    {
        exit(0);
    }
}

void delay(int time)

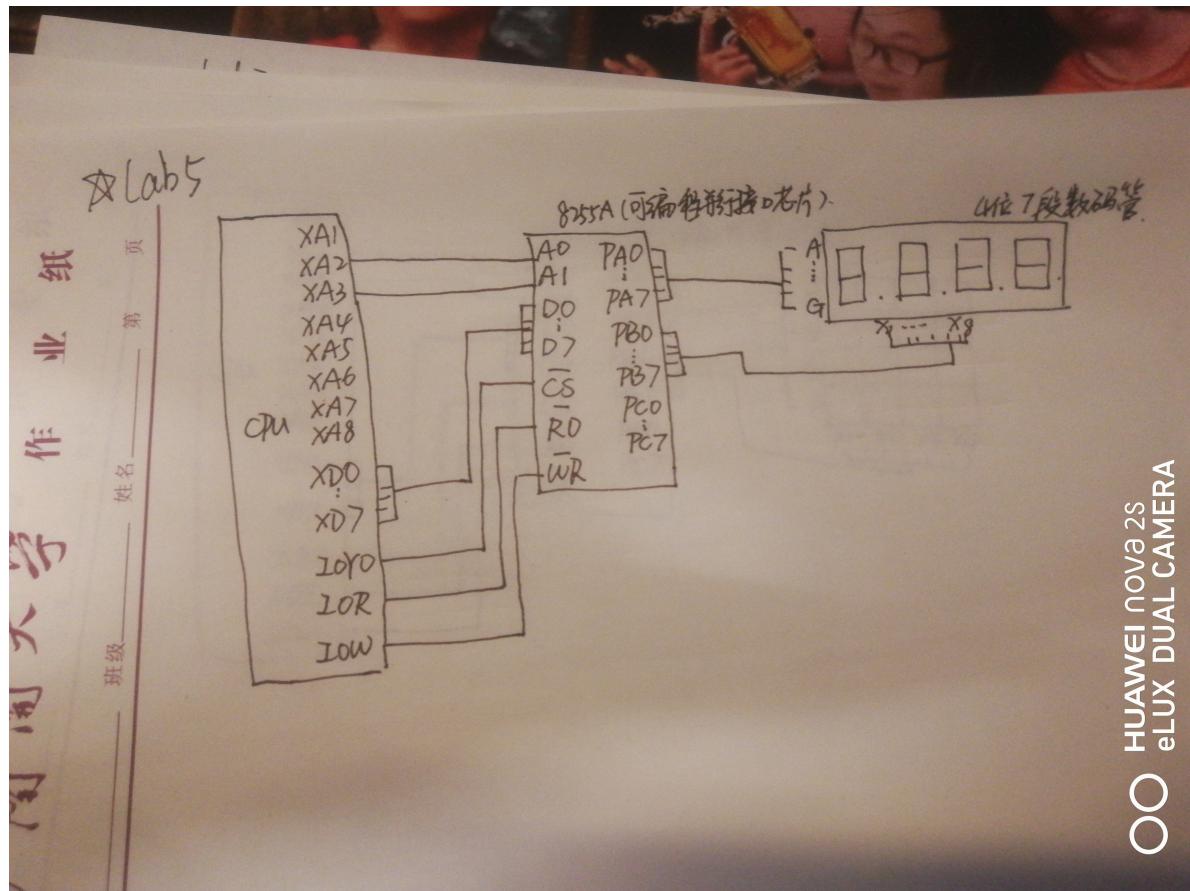
```

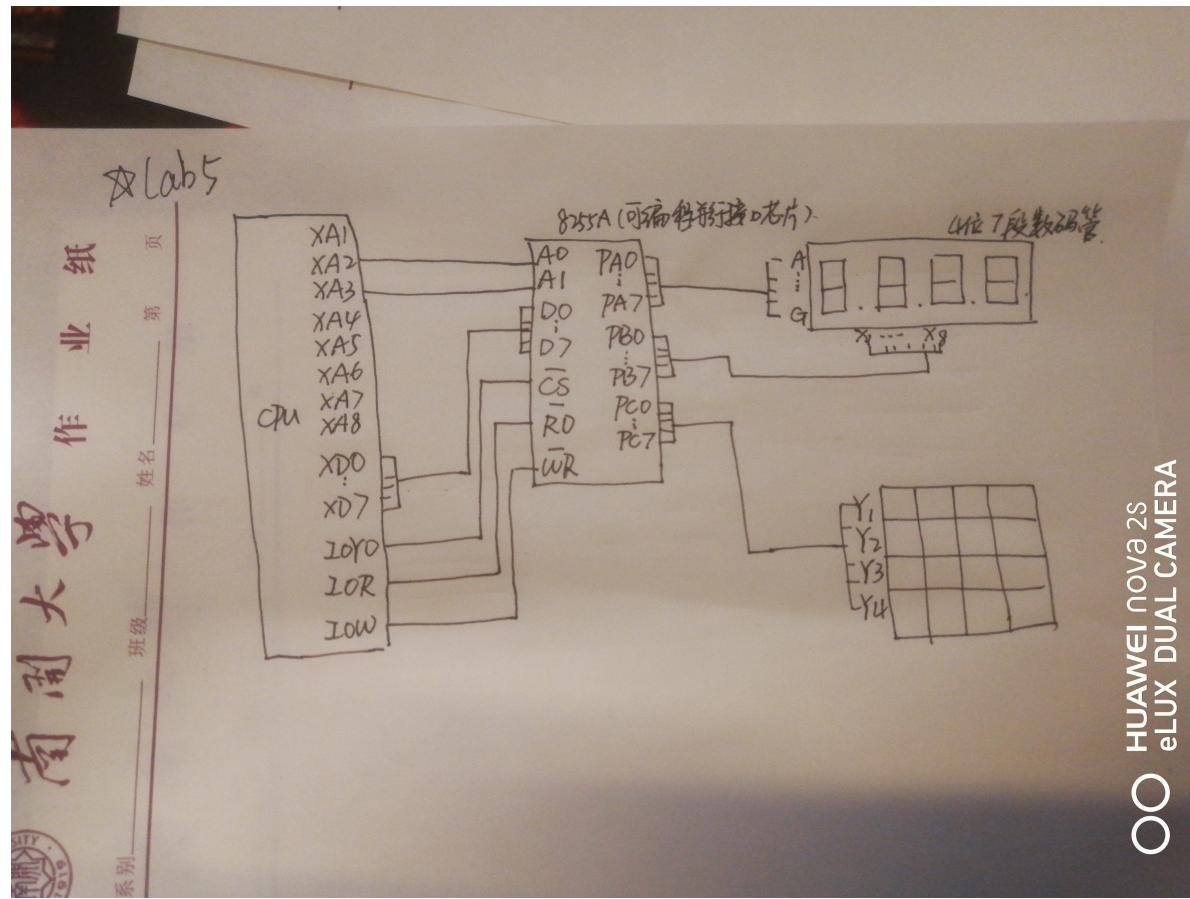
```

{
    int i;
    int j;
    for(i=0;i<=time;i++)
    {
        for(j=0;j<=0x100;j++)
        {
        }
    }
    return;
}

```

抽象接线图





实验连线图

