

计算机原理与嵌入式系统设计实验报告（第四、五次）



目录

[(一) 实验目的 2](#_Toc27578732)

[(二) 实验要求 2](#_Toc27578733)

[(三) 程序分析 2](#_Toc27578734)

[1. 实现输入信息、循环播放 2](#_Toc27578735)

[2. 实现菜单选择 3](#_Toc27578736)

[(四) 程序代码 4](#_Toc27578737)

[1. 实现输入信息、循环播放 4](#_Toc27578738)

[1) 输入日期循环 4](#_Toc27578739)

[2) 循环显示（主循环） 5](#_Toc27578740)

[3) 半屏显示笑脸的子函数 6](#_Toc27578741)

[2. 实现菜单选择 7](#_Toc27578742)

[1) 主循环 7](#_Toc27578743)

[2) 输入日期的子函数 8](#_Toc27578744)

[3) 循环显示子函数 9](#_Toc27578745)

[(五) 实验效果（详见视频） 10](#_Toc27578746)

[(六) 实验总结 11](#_Toc27578747)

# 实验目的

较全面掌握RL78/G13系列嵌入式微控制器的程序设计技术。

利用基础模块设计实验的积累，完成较为复杂的综合实验任务。

# 实验要求

第四次：

在前三周工程的基础上，学习并调通例程文件夹里RL78\_FunctionDemoCodenew例程，在第3次工程基础上，添加完成在LCD循环显示不同的界面：1、个人信息（或自己设计的显示内容，键盘可以输入，修改后的内容参与新一轮的循环显示），2、代表交大的图片，3、谢谢+“笑脸”图片；实验后写好实验报告下次实验提交（电子版），实验报告包括实验步骤（必要时截图），附上所添程序段并标明位置，运行结果拍图或视频、工程文件。

第五次：

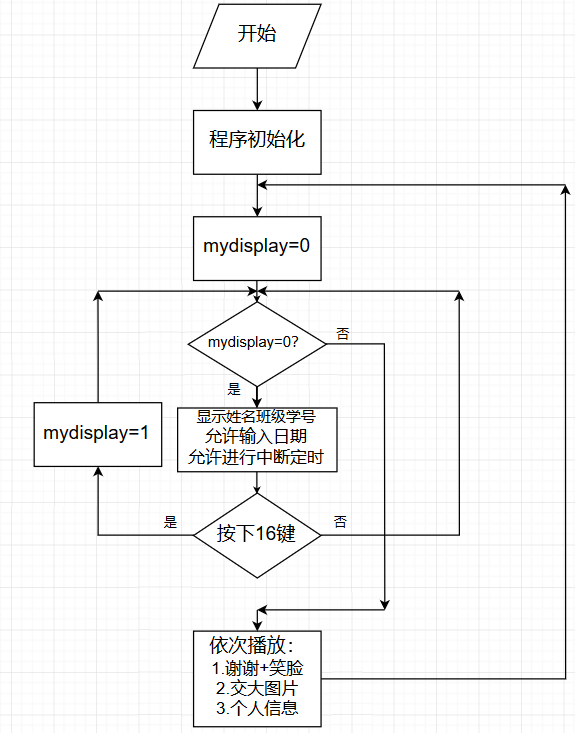
新建工程，要求设计一个菜单并在液晶上显示（例如：秒表、个人信息、图片循环显示），通过按键选择并启动菜单项，实现第2-4次实验所有功能。实验后写好实验报告下次实验提交（电子版），实验报告包括实验步骤（必要时截图），附上所添程序段并标明位置，运行结果拍图或视频、工程文件。

# 程序分析

## 实现输入信息、循环播放

使用了一个标志变量mydisplay，初始化置其值为0。在主循环刚开始再加入一个循环，做输入循环，循环条件为mydisplay=0在这个循环内可以实现前面几次实验所提到的功能。包括显示姓名班级学号、定时计数、还有输入日期（包括日期的删除修改等等）。在确认输入完毕以后按下16键，置mydisplay为1，跳出输入循环。此时程序进入主循环，由于标志变量mydisplay=1，在每次执行到输入循环位置时，都不进如循环，实现笑脸、交大图片、个人信息的循环播放。其中，个人信息（包括输入的日期）用数组实现，对日期的输入即为对数组的修改，在循环播放时调用的是同一个数组，因此实现了输入信息加入循环的要求。

程序流程图如下：



## 实现菜单选择

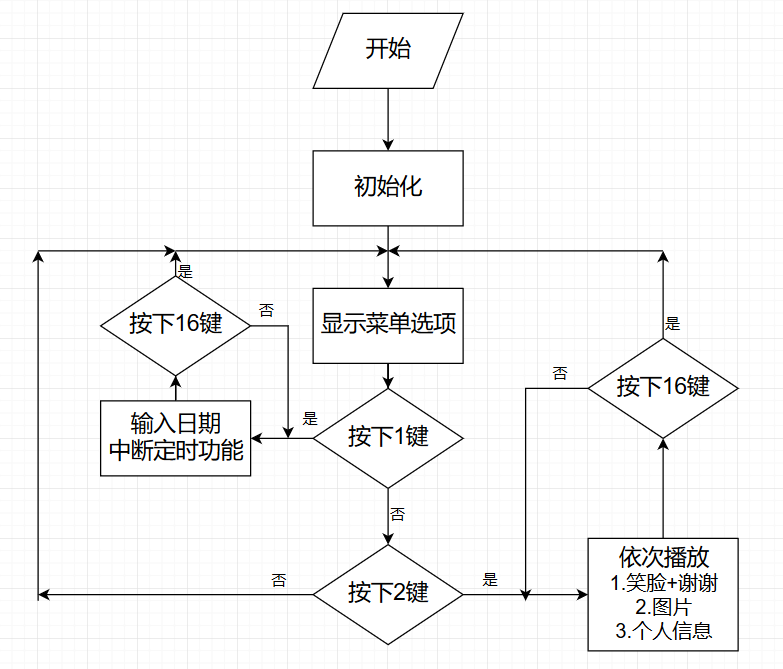
前面的实验虽然实现了功能，但是其本身还有一些小问题，比如说日期只能输入一次，加入循环以后就不能修改内容。因此我定义了一个菜单界面，在菜单界面可以选择修改信息，定时计数，循环播放。在修改、计数界面可以进行前面几次实验的定时、修改信息等等功能；在循环播放界面可以实现第四次实验的循环播放图片功能，而且个人信息保留为最后一次修改的内容。

在菜单界面按1进入修改、计数界面，按2进入循环播放界面。在另外两个子界面按16即可退出。

为了保证程序的可读性，我把两个子界面的功能封装为两个子函数void setdate()

和void Loop\_display()，在主函数中直接调用即可实现功能。

程序流程图如下：



# 程序代码

## 实现输入信息、循环播放

### 输入日期循环

while (mydisplay==0)

{

lcd\_display(0," 任泽华 "

" 自动化71班 "

" 2171411498 ");

lcd\_display(48,date);

Keyboard\_scan();

if(num\_keyboard==14){sflag=1;}

if(sflag==1)num\_10 = num/10;

else {num=0;num\_10=0;}

if(bflag==0 && num\_10==10){ R\_PCLBUZ0\_Start(); bflag=1; delaya();R\_PCLBUZ0\_Stop();}

if(num\_keyboard==13){bflag=0; num=0;num\_keyboard=0;sflag=0;}

if(num\_keyboard!=0)

display();

if(num\_keyboard==13){bflag=0; num=0;num\_keyboard=0;sflag=0;}

leddispaly();

//delayb(1000);

//delayb(1000);

NOP();

WDTE = 0xAC;

if(num\_keyboard==16) {num\_keyboard=0; mydisplay=1;break;}

}

### 循环显示（主循环）

while (1U)

{

while (mydisplay==0)

{

//此处为（1）中的输入信息部分

}

lcd\_write(0x01,0);

LcdFill\_REC();

delayb(1000);

delayb(1000);

WDTE = 0xAC;

lcd\_displayth(0,string);

delayb(1000);

delayb(1000);

WDTE = 0xAC;

LcdClear(); //Clear Graph;

lcd\_write(0x01,0);

delayb(1000);

delayb(1000);

WDTE = 0xAC;

LcdFill\_REC2();

delayb(1000);

delayb(1000);

WDTE = 0xAC;

LcdClear(); //Clear Graph;

lcd\_write(0x01,0);

delayb(1000);

delayb(1000);

LcdFill\_REC3();

delayb(1000);

delayb(1000);

WDTE = 0xAC;

LcdClear(); //Clear Graph;

lcd\_write(0x01,0);

delayb(1000);

delayb(1000);

lcd\_display(0," 任泽华 "

" 自动化71班 "

" 2171411498 ");

lcd\_display(48,date);

delayb(1000);

delayb(1000);

lcd\_write(0x01,0);

delayb(1000);

delayb(1000);

}

### 半屏显示笑脸的子函数

void lcd\_displayth(unsigned char pos, void \*str)

{

unsigned char \*p, \*p1;

unsigned char i,lenth;

if(pos > 63)

return;

p = p1 = (unsigned char \*)str;

lenth = 0;

while(\*p1)

{

lenth++;

p1++;

if(lenth >= 64 - pos)

break;

}

if((pos%16)!=0)

CursorSet(pos);

for(i = pos; i - pos < lenth; i++)

{

if(i == 0)

lcd\_write(0x80+4, 0);

else if(i == 8)

lcd\_write(0x90+4, 0);

else if(i == 16)

lcd\_write(0x88+4, 0);

else if(i == 24)

lcd\_write(0x98+4, 0);

lcd\_write(\*p++, 1);

delay(200);

}

}

## 实现菜单选择

### 主循环

while (1U)

{

lcd\_display(0," 菜 单 "

"1.键盘输入 定时"

"2.循环显示图片 "

"3.按16退出 ");

//"");

Keyboard\_scan();

switch(num\_keyboard)

{

case(1):mydisplay=0;setdate();break;

case(2):Loop\_display();break;

}

}

### 输入日期的子函数

void setdate()

{

while (mydisplay==0)

{

num\_keyboard=0;

lcd\_display(0," 任泽华 "

" 自动化71班 "

" 2171411498 ");

lcd\_display(48,date);

Keyboard\_scan();

if(num\_keyboard==14){sflag=1;}

if(sflag==1)num\_10 = num/10;

else {num=0;num\_10=0;}

if(bflag==0 && num\_10==10){ R\_PCLBUZ0\_Start(); bflag=1; delaya();R\_PCLBUZ0\_Stop();}

if(num\_keyboard==13){bflag=0; num=0;num\_keyboard=0;sflag=0;}

if(num\_keyboard!=0)

display();

if(num\_keyboard==13){bflag=0; num=0;num\_keyboard=0;sflag=0;}

leddispaly();

//delayb(1000);

//delayb(1000);

NOP();

WDTE = 0xAC;

if(num\_keyboard==16) {num\_keyboard=0; mydisplay=1;lcd\_write(0x01,0);break;}

}

}

### 循环显示子函数

void Loop\_display()

{

while(1)

{

Keyboard\_scan();if(num\_keyboard==16){num\_keyboard=0; LcdClear();break;}

lcd\_write(0x01,0);

LcdFill\_REC();

delayb(1000);

delayb(1000);

WDTE = 0xAC;

Keyboard\_scan();if(num\_keyboard==16){num\_keyboard=0;LcdClear(); break;}

lcd\_displayth(0,string);

delayb(1000);

delayb(1000);

WDTE = 0xAC;

Keyboard\_scan();if(num\_keyboard==16){num\_keyboard=0;LcdClear(); break;}

LcdClear(); //Clear Graph;

lcd\_write(0x01,0);

delayb(1000);

delayb(1000);

WDTE = 0xAC;

Keyboard\_scan();if(num\_keyboard==16){num\_keyboard=0;LcdClear(); break;}

LcdFill\_REC2();

delayb(1000);

delayb(1000);

WDTE = 0xAC;

Keyboard\_scan();if(num\_keyboard==16){num\_keyboard=0; LcdClear();break;}

LcdClear(); //Clear Graph;

lcd\_write(0x01,0);

delayb(1000);

delayb(1000);

LcdFill\_REC3();

delayb(1000);

delayb(1000);

WDTE = 0xAC;

Keyboard\_scan();if(num\_keyboard==16){num\_keyboard=0;LcdClear(); break;}

LcdClear(); //Clear Graph;

lcd\_write(0x01,0);

delayb(1000);

delayb(1000);

lcd\_display(0," 任泽华 "

" 自动化71班 "

" 2171411498 ");

lcd\_display(48,date);

delayb(1000);

delayb(1000);

lcd\_write(0x01,0);

delayb(1000);

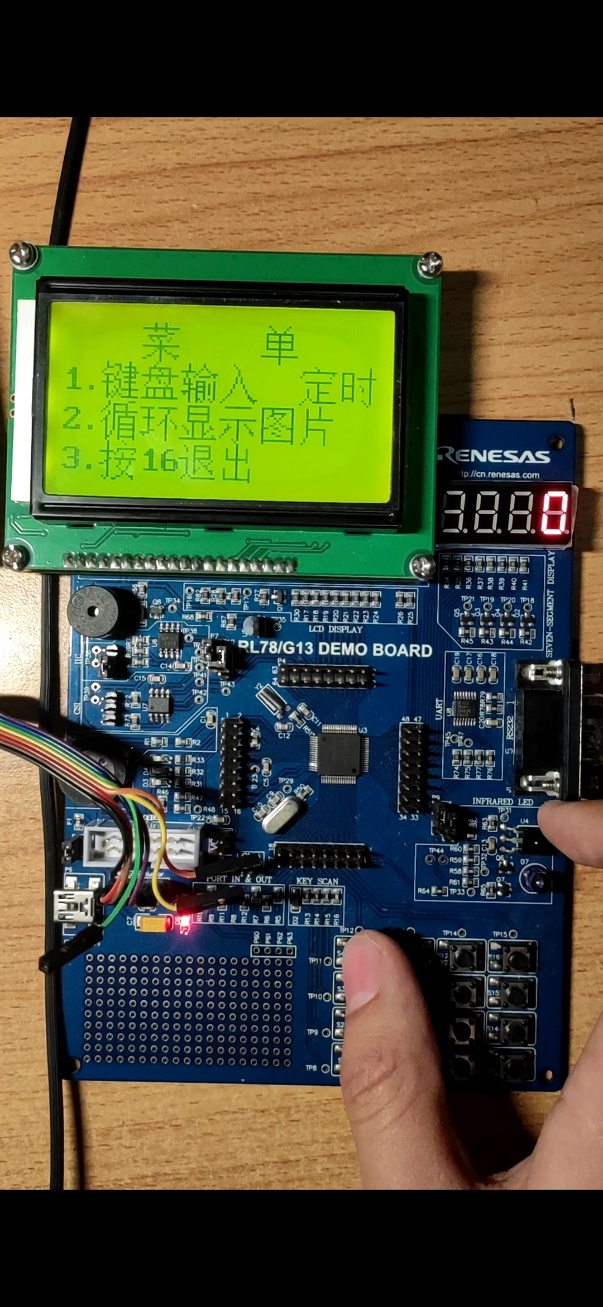
delayb(1000);

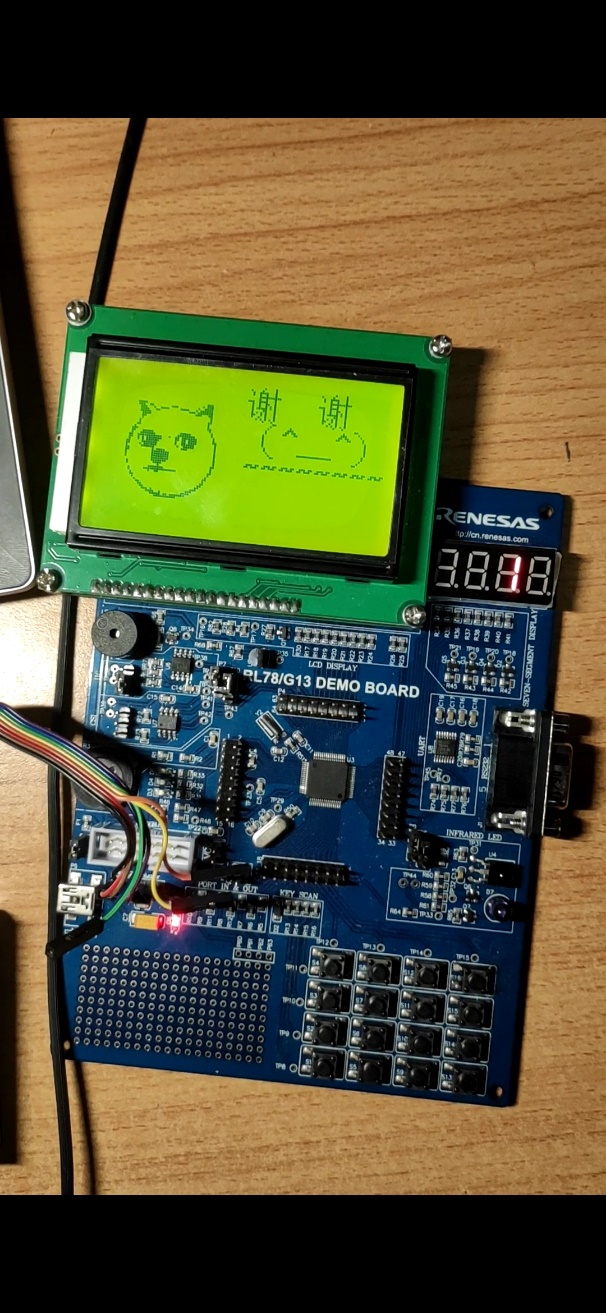
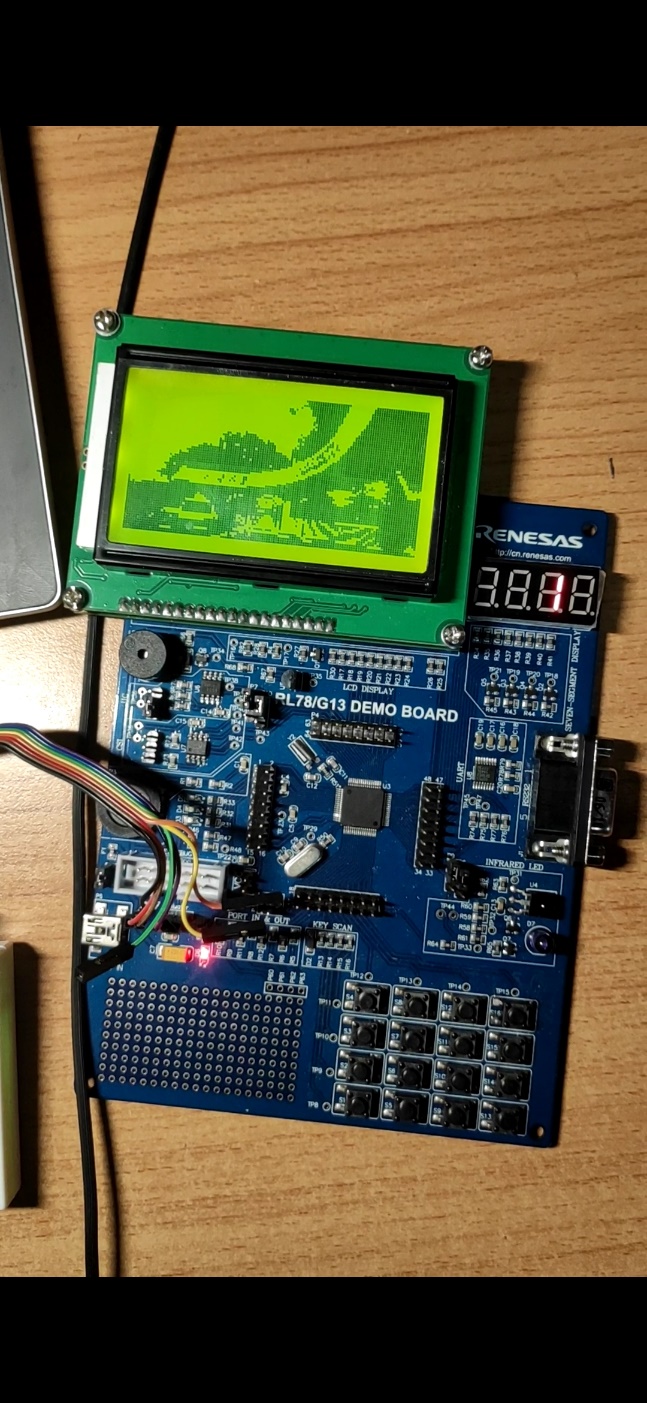
Keyboard\_scan();if(num\_keyboard==16){num\_keyboard=0;LcdClear(); break;}

}

}

# 实验效果（详见视频）

# 实验总结

在刚开始我第四次实验的功能是轮流播放笑脸、交大图片以后停在输入日期界面不动，等输入完成以后继续轮流播放，到此处以后仍然不动。老师说这并不是实验要实现的功能。经过思考与讨论，我决定采用双重循环的方式，通过标志变量的修改实现输入与循环的切换。第五次实验当中有一个小问题，就是在循环显示图片的时候不能同步实现对键盘的扫描，无法按键退出。我想到的办法是每幅图片播放完成以后扫一次键盘，这样就能实现按键的响应。

本次实验是瑞萨开发板的最后一次实验，这几周对于瑞萨开发板的学习让我收获良多。瑞萨也是我接触到的第一个嵌入式系统开发板，刚刚接触一个全新的软件，确实有些困难，按着老师的教程和视频一步步建立自己的第一个工程。其实这都是涉足一个新领域所必然经历的，经过几次实验，我对于这个软件的一些功能已经有了一些了解，也可以在读懂例程的基础上进行一些修改，加入自己的功能。我明白了在实际的工程当中，有时在一开始就完完全全搞懂一个东西是不现实的，应该先做起来，动手去运行一些例程代码，修改一些参数观察输出情况。然后在他人工作的基础上增加一些自己的东西。这对于新手来说是一个很快掌握一项技能的好方法。在老师的耐心指导和与同学们的探讨中，我逐渐学会了如何写代码，如何打断点调试，出了问题要怎么排查。这个过程虽然痛苦，但是真正把问题想明白了以后的心情是非常开心的。看着自己一行行敲出来的代码可以实现一个小小的功能，心里也会有小小的成就感。

就像老师说的，我们这些实验虽然是验证型的，也就是重复做别人做过的东西，但是我们正是在不断地实验中发现自己的短板，磨练自己的代码能力与思维能力。从这个方面来说，我认为这次的实验对我来说是很有意义的。