**电创第二次作业：**

**第四周要求：**

1. 第二次作业需要提交一个点击按键切换led开关的文件
2. LED延时交替开关

**Task1**

在老师给的template中，我添加了Delay.c用来延时一定时间，添加Key.c来定义按键，添加led.c来控制led。

Delay.c

void Delay\_us(uint32\_t xus)

{

SysTick->LOAD = 72 \* xus; //设置定时器重装值

SysTick->VAL = 0x00; //清空当前计数值

SysTick->CTRL = 0x00000005; //设置时钟源为HCLK，启动定时器

while(!(SysTick->CTRL & 0x00010000)); //等待计数到0

SysTick->CTRL = 0x00000004; //关闭定时器

}

主要是上述的计时方法，使用定时器来计时us。

Led.c

void LED\_GPIO\_Config(){

// 打开 GPIOB 端口的时钟

RCC\_APB2PeriphClockCmd(LED\_GPIO\_CLK, ENABLE);

// 配置IO口为输出

GPIO\_InitTypeDef GPIO\_InitStruct;

GPIO\_InitStruct.GPIO\_Mode=GPIO\_Mode\_Out\_PP;

GPIO\_InitStruct.GPIO\_Pin=LED\_GPIO\_Pin;

GPIO\_InitStruct.GPIO\_Speed=GPIO\_Speed\_10MHz;

GPIO\_Init(LED\_GPIO\_PORT, &GPIO\_InitStruct);

}

void LED\_ON(void)

{

// 输出电平

GPIO\_ResetBits(LED\_GPIO\_PORT, LED\_GPIO\_Pin); //LED on

}

void LED\_OFF(void)

{

// 输出电平

GPIO\_SetBits(LED\_GPIO\_PORT, LED\_GPIO\_Pin); //LED on

}

包含了初始化led灯的方法，以及开灯关灯的方法。

Key.c

#include "stm32f10x.h" // Device header

#include "Delay.h"

#include "Key.h"

void Key\_Init(void)

{

RCC\_APB2PeriphClockCmd(RCC\_APB2Periph\_GPIOA, ENABLE);

GPIO\_InitTypeDef GPIO\_InitStructure;

GPIO\_InitStructure.GPIO\_Mode = GPIO\_Mode\_IPU;

GPIO\_InitStructure.GPIO\_Pin = GPIO\_Pin\_0;

GPIO\_InitStructure.GPIO\_Speed = GPIO\_Speed\_50MHz;

GPIO\_Init(GPIOA, &GPIO\_InitStructure);

}

uint8\_t Key\_Scan(void)

{

if( GPIO\_ReadInputDataBit(GPIOA, GPIO\_Pin\_0) == KEY\_ON )

{

Delay\_ms(5);

if(GPIO\_ReadInputDataBit(GPIOA, GPIO\_Pin\_0) == KEY\_ON)

{

// 松手检测

while( GPIO\_ReadInputDataBit(GPIOA, GPIO\_Pin\_0) == KEY\_ON );

return KEY\_ON;

}

else

{

return KEY\_OFF;

}

}

else

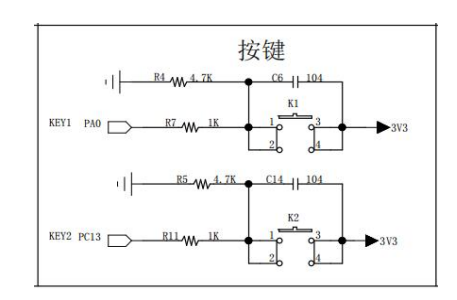
{

return KEY\_OFF;

}

}

根据ppt中图可知，Key1对应PA0口。

Key.c文件定义了按键初始化、按键按下后识别的方法。

Main.c

#include "stm32f10x.h"

#include "led.h"

#include "Delay.h"

#include "Key.h"

int main(void)

{

LED\_GPIO\_Config();

Key\_Init();

int state = 0;

while(1)

{

if(Key\_Scan() == 0) state = 1 - state;

if(state == 1) LED\_ON();

else LED\_OFF();

}

}

首先初始化按键和LED灯，然后在按下按键后状态改变一次，将对应状态赋值给LED输出端口PB5，即可实现按下按键后切换LED灯状态。

**Task2**

int main(void)

{

LED\_GPIO\_Config();

while(1){

LED\_ON();

Delay\_ms(1000);

LED\_OFF();

Delay\_ms(1000);

}

}

改变主函数如上所示，即实现LED延时开关。

**第五周作业要求：**

1. **自己写一遍LED\_BasicTImer。（已完成）**
2. **将定时器中断（需更改）修正为LED的延时开关。**

在直接使用project代码后，发现蜂鸣器延时开关。于是猜想是输出信号的端口设置不对，果然在led.h文件下发现LED初始化方法的宏定义中初始化的不是PB5。将Port改为PB5即可。

#ifndef \_\_LED\_H\_\_

#define \_\_LED\_H\_\_

#include "sys.h"

#define LED\_GPIO\_PIN GPIO\_Pin\_5

#define LED\_GPIO\_PORT GPIOB

#define LED\_GPIO\_CLK RCC\_APB2Periph\_GPIOB

void LED\_GPIO\_Config(void);

void LED\_Change(void);

#endif