

学生实验实习报告册

学年学期： 202 -202 学年 春秋学期

课程名称： 微控制器原理与应用

学生学院： 先进制造工程学院

专业班级： 14042101

学生学号： 2021214698

学生姓名： 邬东升

联系电话： 18883663519

重庆邮电大学教务处制

实验一 LED 显示实验

班 级 14042101 学 号 2021214698 姓 名 邬东升

第 17 周 星 期 一 第 3-4 节 成 绩

一、实验目的；

1.LED 显示的基本原理和功能。

2.LED 和单片机的硬件接口和软件设计方法。

二、实验内容

本实验系统使用 STM32 芯片，控制 LED。

1. 实验步骤 （流水灯）

四、实验结果 (代码)

Main.c文件：

#include "stm32f10x.h"

#include "Delay.h"

#include "led.h"

#define speed 100

int main(void)

{

LED\_Init();

while (1)

{

GPIO\_Write(GPIOA, ~0x0001); //GPIOA的端口状态，下同：0000 0000 0000 0001

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0002); //0000 0000 0000 0010

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0004); //0000 0000 0000 0100

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0008); //0000 0000 0000 1000

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0010); //0000 0000 0001 0000

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0020); //0000 0000 0010 0000

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0040); //0000 0000 0100 0000

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0080); //0000 0000 1000 0000

Delay\_ms(speed);

}

}

Delay.c文件 //延时函数

#include "stm32f10x.h"

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

}

实验二 外部中断实验

班 级 学 号 姓 名

第 15 周 星 期 一 第 3-4 节 成 绩

一、实验目的

1 ．掌握外部中断的初始化设置。

2 ．掌握中断服务的原理。

3 ．掌握中断的编程方法和调试方法。

二、实验内容

本实验系统使用 STM32 芯片，D1 指示灯不断闪烁，表示程序正常运行，当通过不同

按键控制 D2 、D3 灯的状态。

1. 实验步骤

四、实验结果 (代码)

头文件

#include "stm32f10x.h"

#include "led.h"

#include "exti\_key.h"

#include "Delay.h"

#define speed 100

void sb\_1()

{

GPIO\_Write(GPIOA, ~0x0000);

GPIO\_Write(GPIOA, ~0x0001); //GPIOA的端口状态，下同：0000 0000 0000 0001

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0002); //0000 0000 0000 0010

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0004); //0000 0000 0000 0100

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0008); //0000 0000 0000 1000

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0010); //0000 0000 0001 0000

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0020); //0000 0000 0010 0000

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0040); //0000 0000 0100 0000

Delay\_ms(speed);

GPIO\_Write(GPIOA, ~0x0080); //0000 0000 1000 0000

Delay\_ms(speed);

}

int main(void)

{

LED\_Init();

EXTI\_Key\_Init();

GPIO\_Write(GPIOA, ~0x0000);

while(1)

{

/\*

if(a==1)

{

sb\_1();

}

if(a==0)

{

sb\_2();

}

\*/

}

}

中断配置：

//中断函数配置

void EXTI1\_IRQHandler(void)

{

if(EXTI\_GetITStatus(EXTI\_Line1) != RESET)

{

// GPIO\_WriteBit(GPIOB,GPIO\_Pin\_5,(BitAction)((1-GPIO\_ReadOutputDataBit(GPIOB,GPIO\_Pin\_5))));

Delay\_ms(300);

sb\_1();

GPIO\_Write(GPIOA, ~0x0000);

EXTI\_ClearITPendingBit(EXTI\_Line1);

}

}

void EXTI4\_IRQHandler(void)

{

if(EXTI\_GetITStatus(EXTI\_Line4) != RESET)

{

// GPIO\_WriteBit(GPIOB,GPIO\_Pin\_6,(BitAction)((1-GPIO\_ReadOutputDataBit(GPIOB,GPIO\_Pin\_6))));

Delay\_ms(300);

sb\_2();

GPIO\_Write(GPIOA, ~0x0000);

EXTI\_ClearITPendingBit(EXTI\_Line4);

}

}

实验三 定时器实验

班 级 14042101 学 号 2021214698 姓 名 邬东升

第 16 周 星 期一 第 3-4 节 成 绩

二、实验目的

1 ．掌握定时器中断的初始化设置。

2 ．掌握定时器的原理。

二、实验内容

本实验系统使用 STM32 芯片，通过 TIM4 的更新中断控制 D2 指示灯间隔 500ms 秒

状态取反，主函数控制 D1 指示灯不断闪烁。

三、实验步骤

四、实验结果 (代码)

主函数

Main.c文件

#include "TIM3\_PWM.h"

#include "system.h"

#include "SysTick.h"

#include "led.h"

#include "usart.h"

#include "hwjs.h"

u8 smgduan[16]={0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07,

0x7F, 0x6F, 0x77, 0x7C, 0x39, 0x5E, 0x79, 0x71};//0~F 数码管段选数据

int main()

{

int sum = 0;

u16 ccr=100;

// u8 Date1=0;

// int d =100; //单次调节转速

TIM3\_PWM\_Init(899,0);

SysTick\_Init(72);

NVIC\_PriorityGroupConfig(NVIC\_PriorityGroup\_2); //中断优先级分组 分2组

SMG\_Init();

Hwjs\_Init();LED\_Init();USART1\_Init(115200);

while(1)

{

//DigDisplay(0,0,0,0,0, ccr/100,0,0);

// GPIO\_WriteBit(GPIOA,GPIO\_Pin\_0,(BitAction)(0));GPIO\_WriteBit(GPIOA,GPIO\_Pin\_1,(BitAction)(0));

GPIO\_WriteBit(GPIOA,GPIO\_Pin\_2,(BitAction)(0));

sum = ccr/100;

GPIO\_Write(SMG\_PORT,(u16)(~smgduan[sum])<<8);

//delay\_ms(1000);

if(hw\_jsbz==1) //如果红外接收到

{hw\_jsbz=0; //接受标志清零

switch(hw\_jsm)

{

case 0x00FF6897:printf("0");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=0; break;

case 0x00FF30CF:printf("1");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=100;break;

case 0x00FF18E7:printf("2");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=200;break; case 0x00FF7A85:printf("3");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=300;break; case 0x00FF10EF:printf("4");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=400;break; case 0x00FF38C7:printf("5");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=500;break; case 0x00FF5AA5:printf("6");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz); ccr=600;break; case 0x00FF42BD:printf("7");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=700;break;

case 0x00FF4AB5:printf("8");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=800;break;

case 0x00FF52AD:printf("9");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=900;break; default: break;

}

// Date1=hw\_jsm;

hw\_jsm=0; //接收码清零

}

TIM\_SetCompare2(TIM3,ccr);//设置ccr寄存器的值

}

}

对遥控器发送过来的信号解码：

u8 HW\_jssj()

{

u8 t=0;

while(GPIO\_ReadInputDataBit(GPIOA,GPIO\_Pin\_8)==1)//高电平

{

t++;

delay\_us(20);

if(t>=250) return t;//超时溢出

}

return t;

}

void EXTI9\_5\_IRQHandler(void) //红外遥控外部中断

{

u8 Tim=0,Ok=0,Data,Num=0;

while(1)

{

if(GPIO\_ReadInputDataBit(GPIOA,GPIO\_Pin\_8)==1)

{

Tim=HW\_jssj();//获得此次高电平时间

if(Tim>=250) break;//不是有用的信号

if(Tim>=200 && Tim<250)

{

Ok=1;//收到起始信号

}

else if(Tim>=60 && Tim<90)

{

Data=1;//收到数据 1

}

else if(Tim>=10 && Tim<50)

{

Data=0;//收到数据 0

}

if(Ok==1)

{

hw\_jsm<<=1;

hw\_jsm+=Data;

if(Num>=32)

{

hw\_jsbz=1;

break;

}

}

Num++;

}

}

EXTI\_ClearITPendingBit(EXTI\_Line8);

}

实验四 串口通信实验

班 级 14042101 学 号 2021214698 姓 名 邬东升

第 16 周 星 期 三 第 5-6 节 成 绩

三、实验目的

。

1 ．了解串行口的工作原理以及发送的方式。

2 ．了解 PC 机通讯的基本要求。

3. 了解在一个 RS-232 通信实验的基础上。利用单片机的 TXD ，RXD 口，学习 RS-485 差 分串行接口的使用。

二、实验内容

本实验系统使用 STM32 芯片，实现单片机与 PC 机进行通信。

三、实验步骤

1. 实验结果 (代码)

#include "TIM3\_PWM.h"

#include "system.h"

#include "SysTick.h"

#include "led.h"

#include "usart.h"

#include "hwjs.h"

#include <ctype.h>

u8 smgduan[16]={0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07,

0x7F, 0x6F, 0x77, 0x7C, 0x39, 0x5E, 0x79, 0x71};//0~F 数码管段选数据

u16 ccr=100;

int main()

{

int sum = 0;

// u8 Date1=0;

// int d =100; //单次调节转速

TIM3\_PWM\_Init(899,0);

SysTick\_Init(72);

NVIC\_PriorityGroupConfig(NVIC\_PriorityGroup\_2); //中断优先级分组 分2组

SMG\_Init();

Hwjs\_Init();

LED\_Init();

USART1\_Init(9600);

while(1)

{sum = ccr/100;

GPIO\_Write(SMG\_PORT,(u16)(~smgduan[sum])<<8);

//delay\_ms(1000);

if(hw\_jsbz==1) //如果红外接收到

{

hw\_jsbz=0; //接受标志清零

switch(hw\_jsm)

{

case 0x00FF6897:printf("0");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=0; break;

case 0x70:printf("-");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);break;

case 0x00FF30CF:printf("1");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=100; break;

case 0x00FF18E7:printf("2");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=200; break;

case 0x00FF7A85:printf("3");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=300; break;

case 0x00FF10EF:printf("4");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=400; break;

case 0x00FF38C7:printf("5");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=500; break;

case 0x00FF5AA5:printf("6");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz); ccr=600; break;

case 0x00FF42BD:printf("7");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=700; break;

case 0x00FF4AB5:printf("8");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=800; break;

case 0x00FF52AD:printf("9");printf("Key:%x, cnt:%d\n", hw\_jsm, hw\_jsbz);ccr=900; break;

default: break;

}

hw\_jsm=0; //接收码清零

}

TIM\_SetCompare2(TIM3,ccr);//设置ccr寄存器的值

}

}

void TIM3\_IRQHandler(void)

{

if (TIM\_GetITStatus(TIM3, TIM\_IT\_Update) != RESET)

{

GPIO\_WriteBit(GPIOB,GPIO\_Pin\_5,(BitAction)((1-GPIO\_ReadInputDataBit(GPIOB,GPIO\_Pin\_5))));

TIM\_ClearITPendingBit(TIM3, TIM\_IT\_Update );

}

}

void USART1\_IRQHandler(void) //串口1中断服务程序

{

if(USART\_GetITStatus(USART1, USART\_IT\_RXNE) != RESET) //接收中断

{

hw\_jsm =USART\_ReceiveData(USART1);//(USART1->DR); //读取接收到的数据

hw\_jsbz=1;

if(islower(hw\_jsm))

{

GPIO\_SetBits(GPIOA,GPIO\_Pin\_5);

printf("熄灭LED灯\n");

}

if(isupper(hw\_jsm))

{

GPIO\_ResetBits(GPIOA,GPIO\_Pin\_5);

printf("点亮LED灯 \n");

}

if(hw\_jsm == '0') ccr=0;

if(hw\_jsm == '1')ccr=100;

if(hw\_jsm == '2')ccr=200;

if(hw\_jsm == '3')ccr=300;

if(hw\_jsm == '4')ccr=400;

if(hw\_jsm == '6')ccr=600;

if(hw\_jsm == '7')ccr=700;

if(hw\_jsm == '8')ccr=800;

if(hw\_jsm == '9')ccr=900;

USART\_SendData(USART1, islower(hw\_jsm) ? toupper(hw\_jsm) : tolower(hw\_jsm));

USART\_SendData(USART1,hw\_jsm);

while(USART\_GetFlagStatus(USART1,USART\_FLAG\_TC) != SET);

}

USART\_ClearFlag(USART1,USART\_FLAG\_TC);

}

串口重定向到printf函数：

int fputc(int ch,FILE \*p) //函数默认的，在使用printf函数时自动调用

{

USART\_SendData(USART1,(u8)ch);

while(USART\_GetFlagStatus(USART1,USART\_FLAG\_TXE)==RESET);

return ch;

}