

华东交通大学

课程设计（论文）任务书

电气与自动化工程 学院 专业 电力牵引 班级 4 班

一、课程设计（论文）题目 双机双向通信设计

二、课程设计（论文）工作自 2020 年 12 月 20 日起

至 2020 年 12 月 24 日止。

三、课程设计（论文）的内容和要求：

双机双向通信设计

甲机接开关 K1,发光管 D1D2 及数码管 1, 乙机接开关 K2,发光管

D3D4 及数码管 2。按 K1 一次, D1D3 闪烁, 按 K1 第二次, D2D4 闪

烁, 按 K1 第三次, D1D2D3D4 均闪烁, 按 K1 第四次, D1D2D3D4

均熄灭; 每按下 K2 一次控制两个数码管显示加 1 (从 0 到 9)

2.课程设计说明书编写要求

(1) 设计说明书用专用稿纸书写或用 B5 纸打印, 要求论述清晰, 撰写整洁规范, 引用资料应说明出处。插图应该标注图题图序, 表格要有表题表序。

(2) 说明书内容应包括(装订次序): 题目、目录、正文、设计总结、参考文献、附录等。

(3) 报告内容应包括: 系统功能确定; 方案论证(方案对比, 技术经济分析); 整体设计论述; 硬件设计(电路接线, 元器件说明, 硬件资源分配); 软件设计(软件流程, 编程思想, 程序注释, 系统调试; 结论(对设计结果进行总结, 收获与体会); 附录(附录包括程序清单(必须加上合理的注释)和符合国标的所设计系统原理图的

图纸)。

学生签名：张瑜扬

2021 年 12 月 23 日

课程设计（论文）评审意见

平时表现 与态度（10%）		优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%
设计成果 （30%）	课题复杂度	优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%
	设计方案	优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%
	创新性和性能	优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%
设计说明 书质量 （30%）	内容完整度	优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%
	规范度	优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%
	图纸质量	优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%
答辩 （30%）	问题分析能力	优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%
	回答正确度	优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%
	使用工具能力	优（ ）、良（ ）、中（ ）、一般（ ）、差（ ）；	10%

总评成绩：优（ ）、良（ ）、中（ ）、及格（ ）、不及格（ ）

评阅人 _____ 职称 _____

2022 年 1 月 _____ 日

双机双向通信设计

摘要：这次课程设计的主要目的是自己设计一款双机双向通信设计，可以实现甲机接开关 K1,发光管 D1D2 及数码管 1 乙机接开关 K2,发光管 D3D4 及数码管 2 按 K1 一次,D1D3 闪烁, 按 K1 第二次, D2D4 闪烁, 按 K1 第三次, D1D2D3D4 均闪烁, 按 K1 第四次, D1D2D3D4 均熄灭。每按下 K2 一次控制两个数码管显示加 1 (从 0 到 9)。

数据在线路中常用的数据传输方法主要有并行通讯、串行通讯、异步传输和同步传输。

并行传输指的是数据以成组的方式，在多条并行信道上同时进行传输。常用的就是将一个字符代码的几位二进制码，分别在几个并行行道上进行传输。例如，采用 8 单位代码的字符，可以用 8 个信道并行传输，一次传送一个字符，因此收、发双方不存在字符的同步问题，不需要加“起”、“止”信号或者其他信号来实现收、发双方的字符同步，这是并行传输的一个主要优点。但是，并行传输必须有并行信道，这带来了设备上或实施条件的限制。串行传输是构成字符的二进制代码在一条信道上以位（码元）为单位，按时间顺序逐位传输的方式。按位发送，逐位接收，同时还要确认字符，所以要采取同步措施。速度虽慢，但只需一条传输信道，投资小，易于实现，是数据传输采用的主要传输方式。也是计算机通信采取的一种主要方式。

此次的双机双向通信设计，由 stm32f103c8t6 单片机最小控制系统、oled 显示电路、stc89c52rc 单片机最小控制系统、nrf2401l2.4g 无线传输电路、蓝牙串口 hc06 电路模块、按键控制系统电路和 RGB 彩色 led 灯电路组成。

关键词：串口通讯；无线传输；按键控制；单片机；

目录

一、 总体方案论述.....	5
1.1 任务分析.....	5
1.2 方案对比.....	5
1.3 系统总设计.....	8
二、 硬件设计.....	9
2.1 oled 显示屏电路.....	9
2.2 nrf2401l 串口电路.....	11
2.3 数码管显示电路.....	15
三、 软件设计.....	16
3.1 软件流程框图.....	16
四、 系统调试结果.....	20
4.1 Keil 软件调试.....	20
4.2 Proteus 软件仿真过程.....	20
4.3 仿真结果.....	20
五、 总结与体会.....	23
5.1 心得体会.....	23
参考文献.....	25
附录.....	26

一、 总体方案论述

1.1 任务分析

双机双向通信设计

- 甲机接开关 K1,发光管 D1D2 及数码管 1
- 乙机接开关 K2,发光管 D3D4 及数码管 2
- 按 K1 一次, D1D3 闪烁, 按 K1 第二次, D2D4 闪烁, 按 K1 第三次, D1D2D3D4 均闪烁, 按 K1 第四次, D1D2D3D4 均熄灭。
- 每按下 K2 一次控制两个数码管显示加 1 (从 0 到 9)

1.2 方案对比

在设计的过程中, 我思考出了如下方案:

显示模块的论证与选择:

方案一: 用数码管进行显示, 外部电路包括 2 个数码管, 4 个 led 发光二极管, 以及 2 个排阻。

方案二: 采用 0.96 寸 oled 进行显示, 需要一个 0.96oled 显示屏模块。

通过比较两个方案，方案一器件较多，连线比较繁琐。方案二中的oled屏幕由于采用了模块化设计，市面上的产品已经将必要的外围电路集成，较方案一硬件电路更加简单，但对程序的时序要求较为复杂。

综合考虑，在protues仿真里面采用方案一进行显示，在实物制作中选择方案二进行显示。

主控模块的论证与选择：

方案一：采用stc89c52rc单片机。

STC89C52 是一种低功耗、高性能 CMOS 8 位微控制器，具有 8K 在系统可编程 Flash 存储器。在单芯片上，拥有灵巧的 8 位 CPU 和在系统可编程 Flash，使得 STC89C52 为众多嵌入式控制应用系统提供高灵活、超有效的解决方案。具有以下标准功能：8k 字节 Flash，512 字节 RAM，32 位 I/O 口线，看门狗定时器，内置 4KBEEPROM，MAX810 复位电路，三个 16 位定时器/计数器，一个 6 向量 2 级中断结构，全双工串行口。

方案二：采用 stm32f103c8t6 单片机。

STM32F103C8T6 是一款基于 ARM Cortex-M 内核 STM32 系列的 32 位的微控制器，程序存储器容量是 64KB，需要电压 2V~3.6V，工作温度为 -40°C ~ 85°C。STM32F103C8T6 整个系统结构可以分为由 ARM 公司设

计的 Cortex-M3 内核和 ST 公司在此基础上优化的总线矩阵、DMA(Direct Memory Access, 直接内存读取)、AHB、APB1 以及 APB2 上挂载的外设等两个部分。Cortex-M3 摒弃了冯·诺依曼结构(普林斯顿结构), 采用了将指令存储和数据存储分开的哈佛结构(Harvard Architecture), 这样一来 Cortex-M3 同时拥有了独立的 32-bit 指令总线和 32-bit 数据总线, 数据访问将不再占用指令总线, 同时读取指令和数据后提升了 MCU 运行速度。

通过比较两个方案, 方案一的程序结构较为简单, 同时价格便宜, 但存在内存不足的问题。方案二的程序结构相对复杂, 但性能优越, 将 adc 转换等功能集成到主控芯片上, 减少了外围电路的连接, 同时可以采用 st (意法半导体) 公司自研的与之对应的 cubemx 编程结合, 可以一键化初始基本外设, 减少开发难度, 并且时序封装较为简便, 可以充分控制复杂的外设。综合考虑: 与 oled 相连的单片机采用 stm32f103 控制, 与 led 相连的单片机用 stc89c52rc 控制。

通讯方式的论证与选择:

方案一: 直接采用串口进行通讯

将 stm32c8t6 与 stc89c52rc 的串口通过杜邦线直接连接起来进行通讯。

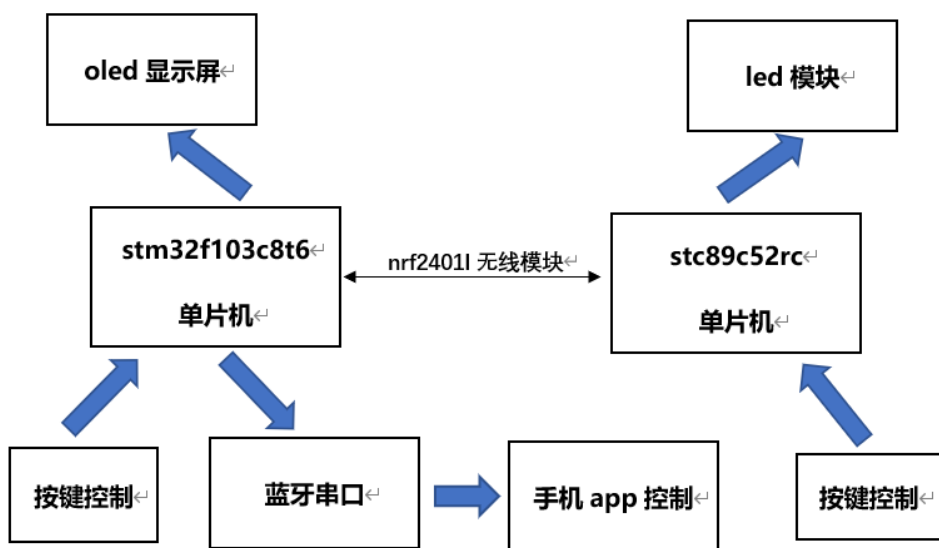
方案二: 采用无线通讯技术

利用基于 spi 通讯协议的 nrf2401l 无线 2.4g 通讯模块将两块开发板连接起来。同时采用蓝牙串口，在手机端显示 stm32f103c8t6 单片机的工作内容。通过比较两个方案，方案一的硬件设计简单，程序结构简单，但由于存在 stc89c52rc 内部参考电压为 5v，与 stm32f103c8t6c 芯片内部参考电压为 3.3v 不一致，直接连接可能存在烧坏芯片的问题。方案二中单片机与 2.4g 模块连接后可以通过电平转换的芯片并通过无线连接，不存在烧毁芯片的问题，但程序设计对时序的要求较高，同时蓝牙模块的使用需要调整正确的波特率，并与手机端的 app 相结合。

综合考虑：在 protues 仿真里面采用方案一进行两块 stc89c52rc 单片机的通讯，在实物制作时采用方案二进行 stc89c52rc 和 stm32f103c8t6 单片机间的无线通讯。

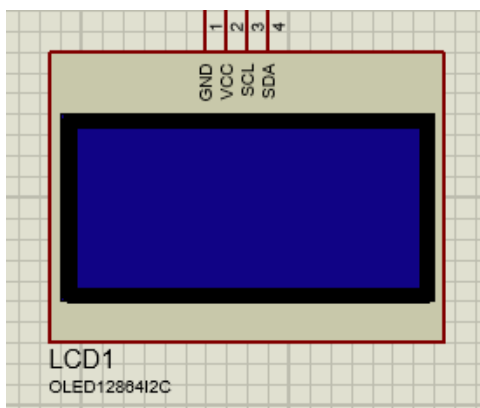
1.3 系统总设计

总体设想：利用按键输入设置不同的工作状态，stm32f103 单片机所控制的 oled 屏幕上显示对应的工作状态，并将这个信息通过 nrf2401l 无线模块发送给 stc89c52rc 单片机，同时控制不同的 led 灯的工作状态。反之也可以通过 stc89c52rc 所连的按键的触发发送对应的工作状态信息给 stm32f103c8t6 单片机，在 oled 屏幕上显示不同的内容。同时 stm32 单片机通过蓝牙与手机相连接，实时发送工作状态。



二、 硬件设计

2.1 oled 显示屏电路



接口定义：

GND：电源地

VCC：电源正（3.3 ~ 5V）

SCL：OLED 的 D0 脚，在 IIC 通信中为时钟管脚

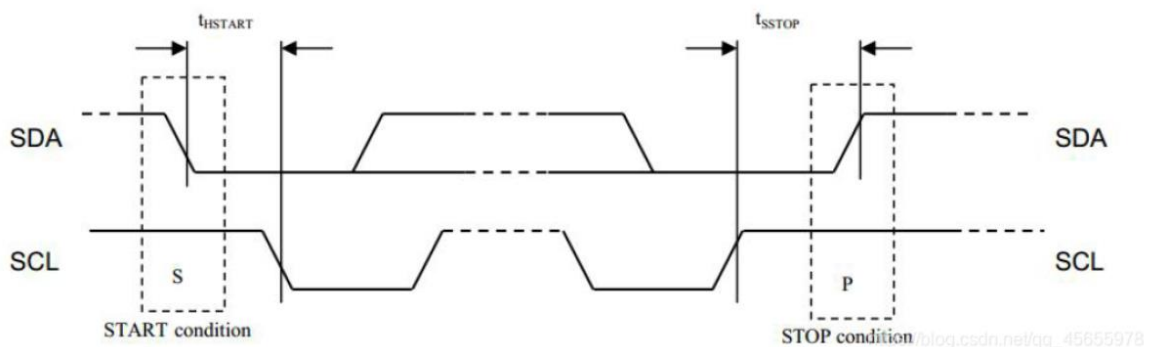
SDA：OLED 的 D1 脚，在 IIC 通信中为数据管脚

0.96OLED（4Pin）模块采用 SSD1306 为驱动芯片，模块带有稳压芯片，支持软件模拟 IIC 通讯与硬件 IIC 通讯，上电自动复位，功耗低，自发光自由视角。

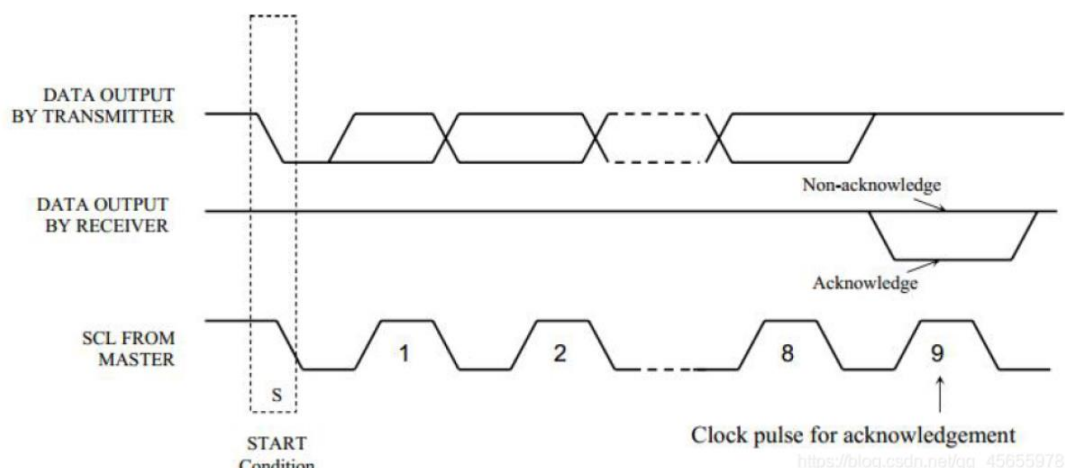
SSD1306 有 3 中寻址模式：页寻址模式、水平寻址模式、垂直寻址模式。

寻址方式决定了写入数据的方式。

起始条件与停止条件时序图



应答时序图



控制原理：

1. 主设备开始数据通讯通过开始条件，开始条件的定义如下图。开始条件发生在 SCL 高电平期间，SDA 产生一个下降沿的跳变。
2. 从地址紧跟着开始信号，用于设备从设备。对于 SSD1306 来说，从地址可能是 “b01111100” 或者 “b01111101”。通过改变 “SA0 位” 为 0 或 1，(D/C 引脚作为 SA0)。
3. 写模式发生在 R/W#位是低电平。
4. 应答信号在接收到一个字节的的数据后有从设备产生，包括从地址和 R/W#位，参考下面关于应答信号的图示。应答信号定义为在 SCL 时钟周期的高电平期间 SDA 拉低。
5. 在从地址发送结束后，不管是控制字节还是数据姐姐都可以发送通过 SDA 线。一个控制字节主要包括 Co 位和 D/C#位在六个 0 后。
 - A. 如果 Co 位为低电平，发送的后续信息只能是数据字节。
 - B. D/C#位决定下一个数据字节是数据还是命令字节，如果 D/C#字节设置为低电平，这意味着下一个字节是命令字节，如果 D/C#为是高电平，下一个数据字节是数据字节，将被存储在 GDDRAM 中，GDDRAM 的列地址指针在每一个数据写之后自加一。
6. 应答为在接收到一个数据字节或只命令字节后产生。
7. 写模式在接受到停止信号之后结束。停止条件定义如下图所示，停止信号发生在 SCL 高点平期间 SDA 的上升沿，作为停止信号。

2.2 nrf2401l 串口电路

结构方框图:

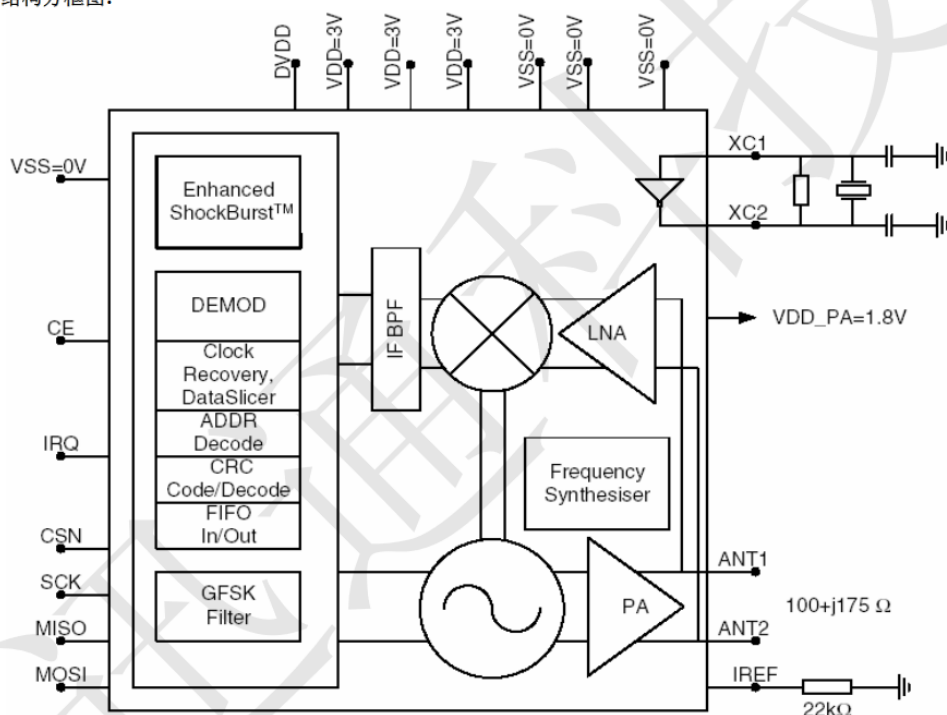


图 1 nRF24L01 及外部接口

引脚及其功能:

引脚	名称	引脚功能	描述
1	CE	数字输入	RX 或 TX 模式选择
2	CSN	数字输入	SPI 片选信号
3	SCK	数字输入	SPI 时钟
4	MOSI	数字输入	从 SPI 数据输入脚
5	MISO	数字输出	从 SPI 数据输出脚
6	IRQ	数字输出	可屏蔽中断脚
7	VDD	电源	电源 (+3V)
8	VSS	电源	接地 (0V)
9	XC2	模拟输出	晶体振荡器 2 脚
10	XC1	模拟输入	晶体振荡器 1 脚/外部时钟输入脚
11	VDD_PA	电源输出	给 RF 的功率放大器提供的+1.8V 电源
12	ANT1	天线	天线接口 1
13	ANT2	天线	天线接口 2
14	VSS	电源	接地 (0V)
15	VDD	电源	电源 (+3V)
16	IREF	模拟输入	参考电流
17	VSS	电源	接地 (0V)
18	VDD	电源	电源 (+3V)
19	DVDD	电源输出	去耦电路电源正极端
20	VSS	电源	接地 (0V)

表 3、nRF24L01 引脚功能

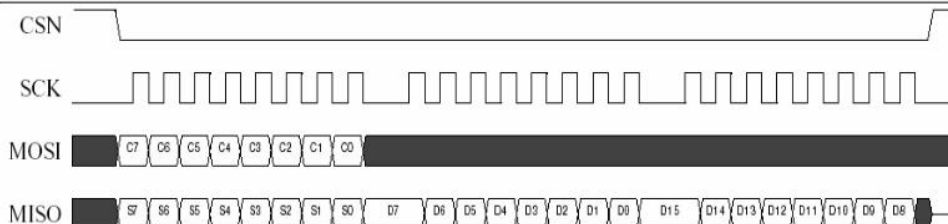


图 8、SPI 读操作

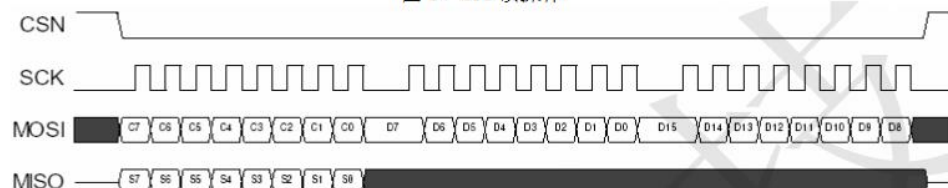


图 9、SPI 写操作

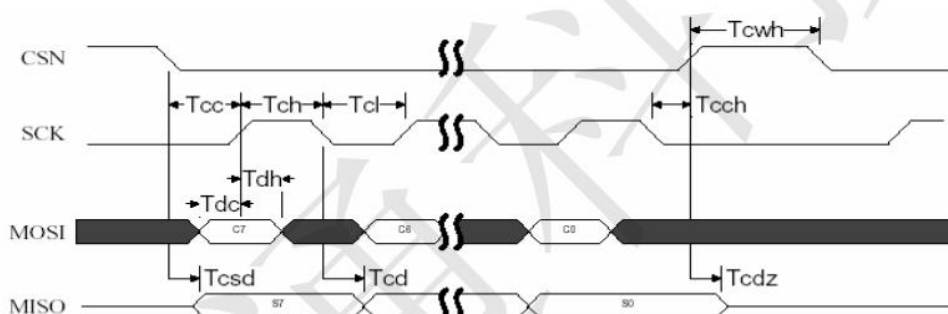


图 10、SPI NOP 操作时序图

nRF24L01 单端匹配网络：晶振、偏置电阻、去藕电容。

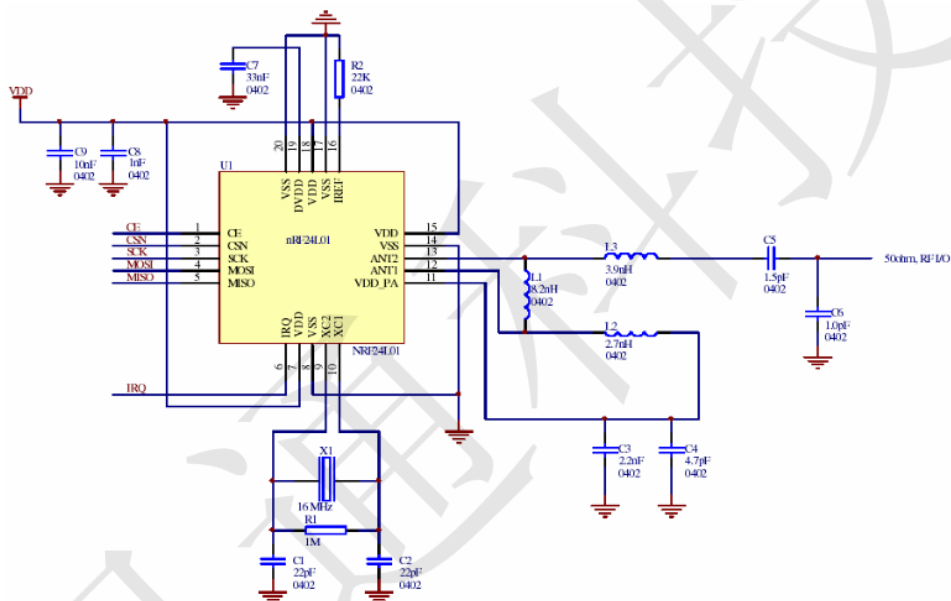


图 13、nRF24L01 单端 50 Ω 射频输出原理图

概述:

nRF24L01 是一款新型单片射频收发器件, 工作于 2.4 GHz ~ 2.5 GHz ISM 频段。内置频率合成器、功率放大器、晶体振荡器、调制器等功能模块, 并融合了增强型 ShockBurst 技术, 其中输出功率和通信频道可通过程序进行配置。nRF24L01 功耗低, 在以 -6 dBm 的功率发射时, 工作电流也只有 9 mA; 接收时, 工作电流只有 12.3 mA, 多种低功率工作模式, 工作在 100mw 时电流为 160mA, 在数据传输方面实现相对 WiFi 距离更远, 但传输数据量不如 WiFi (掉电模式和空闲模式) 使节能设计更方便。

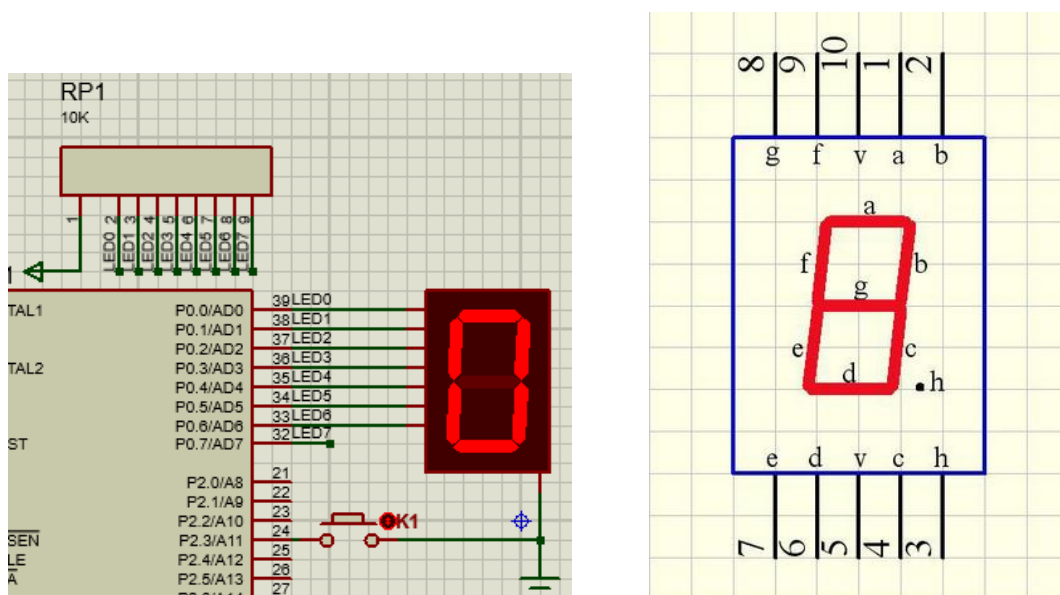
控制原理:

发射数据时, 首先将 nRF24L01 配置为发射模式: 接着把接收节点地址 TX_ADDR 和有效数据 TX_PLD 按照时序由 SPI 口写入 nRF24L01 缓存区, TX_PLD 必须在 CSN 为低时连续写入, 而 TX_ADDR 在发射时写入一次即可, 然后 CE 置为高电平并保持至少 10 μ s, 延迟 130 μ s 后发射数据; 若自动应答开启, 那么 nRF24L01 在发射数据后立即进入接收模式, 接收应答信号 (自动应答接收地址应该与接收节点地址 TX_ADDR 一致)。如果收到应答, 则认为此次通信成功, TX_DS 置高, 同时 TX_PLD 从 TX FIFO 中清除; 若未收到应答, 则自动重新发射该数据 (自动重发已开启), 若重发次数 (ARC) 达到上限, MAX_RT 置高, TX FIFO 中数据保留以便再次重发; MAX_RT 或 TX_DS 置高时, 使 IRQ 变低, 产生中断, 通知 MCU。最后发射成功时, 若 CE 为低则 nRF24L01 进入空闲模式 1; 若发送堆栈中有数据且 CE 为高, 则进入下一次发射; 若发送堆栈中无数据且 CE 为高, 则进入空闲模式 2。

接收数据时, 首先将 nRF24L01 配置为接收模式, 接着延迟 130 μ s 进入接收状态等待数

据的到来。当接收方检测到有效的地址和 CRC 时，就将数据包存储在 RX FIFO 中，同时中断标志位 RX_DR 置高，IRQ 变低，产生中断，通知 MCU 去取数据。若此时自动应答开启，接收方则同时进入发射状态回传应答信号。最后接收成功时，若 CE 变低，则 nRF24L01 进入空闲模式 1。

2.3 数码管显示电路

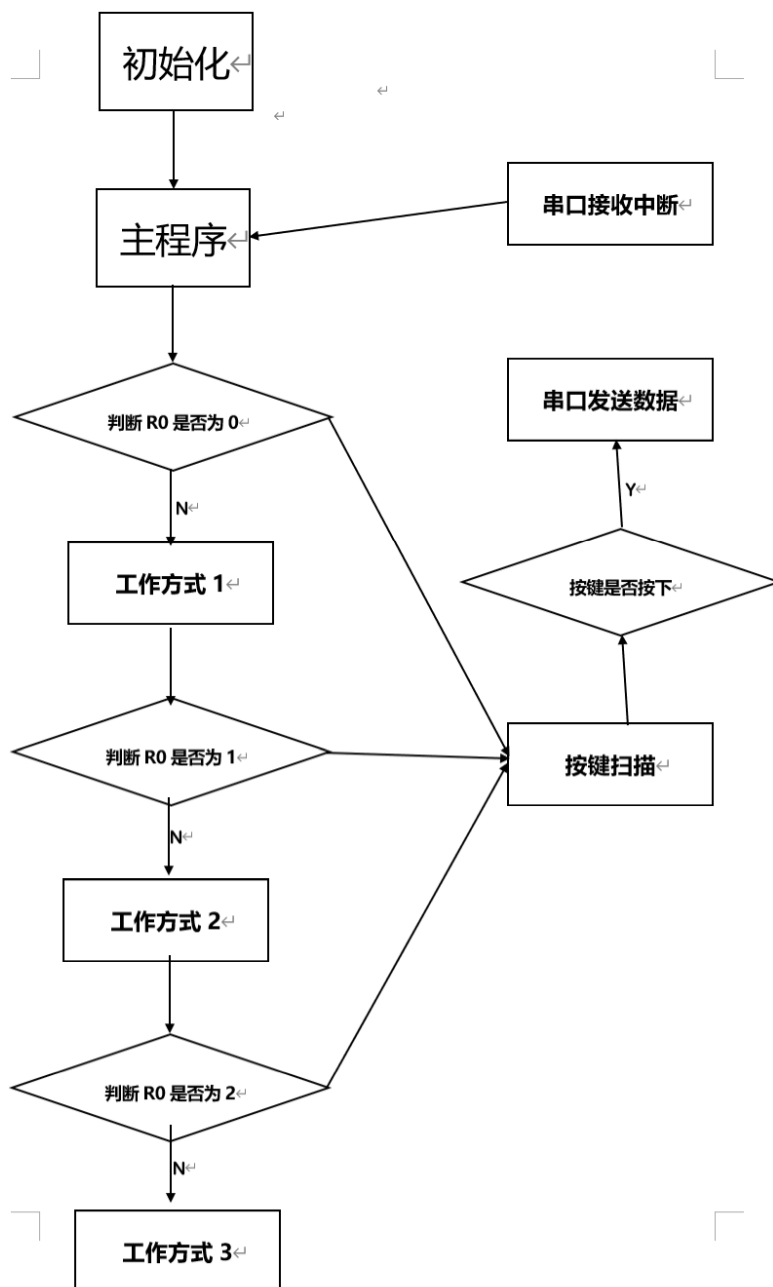


数码管的一种是半导体发光器件，数码管可分为七段数码管和八段数码管，区别在于八段数码管比七段数码管多一个用于显示小数点的发光二极管单元 DP (decimal point)，其基本单元是发光二极管。七段数码管是一类价格便宜 使用简单，通过对其不同的管脚输入相对的电流，使其发亮，从而显示出数字能够显示 时间、日期、温度等所有可用数字表示的参数的器件。

七段数码管分为共阳极及共阴极，共阳极的七段数码管的正极（或阳极）为八个发光二极管的共有正极，其他接点为独立发光二极管的负极（或阴极），使用者只需把正极接电，不同的负极接地就能控制七段数码管显示不同的数字。共阴极的七段数码管与共阳极的只是接驳方法相反而已。

三、 软件设计

3.1 软件流程框图




```

SMG EQU P0                ;IO 定义, EQU 相当于 define
LED1 EQU P2^7             ;bit 位地址符号伪指令, 将位地址赋给字符名称
LED2 EQU P2^6
BUTTON BIT P2.3

        ORG 0000H
        LJMP START

        ORG 0023H
        LJMP UART_ISR

        ORG 0030H
START:
        MOV SP,#70H
        MOV TMOD,#20H      ;定时器 1 设为方式 2
        MOV TL1,#0FDH      ;波特率为 9600,教材 210 页表
        MOV TH1,#0FDH
        MOV PCON,#00H      ;波特率不加速,pcon 为电源控制寄存器, 令
smod=0, 系统波特率不加倍
        MOV SCON,#50H      ;串口设为方式 1 双机通讯接收控制允许, 0101
0000 参考教材 203 页
        SETB ES             ;开串口中断,置 1 操作
        SETB TR1            ;启动定时器 1
        SETB EA             ;开总中断
        SETB BUTTON         ;读取前置高, 定义按键初值为 1
        MOV R0,#0           ;接收数据
        MOV R1,#0           ;发送数据
        MOV R2,#0
        MOV DPTR,#BCDto7_TAB

MAIN:
        MOV A,R1
        MOVC A,@A+DPTR      ;变址寻址, 查表指令, 取数据
        CPL A               ;直接寻址位取反 (使用共阴数码管按位取反)
        MOV SMG,A           ;将 A 值赋值给 P0 端口

        LCALL DELY10ms      ;调用 delay 延时函数

```

	CJNE R0,#0,NE1	;条件转移指令, 比较 R0 与 0, 若不相等则转移
	CLR LED1	
	CLR LED2	;LED 全灭
	JMP KEYF	
NE1:		
	CJNE R0,#1,NE2	;LED 13 闪烁
	CLR LED2	
	INC R2	
	CJNE R2,#20,KEYF	;闪烁时间间隔约 20ms
	MOV R2,#0	
	CPL LED1	
	JMP KEYF	
NE2:		
	CJNE R0,#2,NE3	;LED 24 闪烁
	CLR LED1	
	INC R2	
	CJNE R2,#20,KEYF	
	MOV R2,#0	
	CPL LED2	
	JMP KEYF	
NE3:		
	CJNE R0,#3,KEYF	; LED 同时闪烁
	CLR LED1	
	CLR LED2	
	INC R2	
	CJNE R2,#20,KEYF	
	MOV R2,#0	
	CPL LED1	
	CPL LED2	
KEYF:		;keyf 判断按键按下的函数
	JB BUTTON,NOPRESS	;位条件转移指令, 若 button 为 1, 则
转移		
	INC R1	

```

CJNE R1,#10,NEX
MOV R1,#0

```

NEX:

```

MOV A,R1
;ADD A,#'0'
MOV SBUF,A           ;发送数据
JNB TI,$             ;等待发送完成
CLR TI
JNB BUTTON,$

```

NOPRESS:

```

AJMP MAIN

```

UART_ISR:

```

PUSH ACC             ;压栈指令
JNB RI,OUT           ;若 RI 为 0 则转移收
MOV A,SBUF           ;收数据
;SUBB A,#'0'         ;转 ASCII,为了在 pc 上显示, 但实际调用时有一点问题
MOV R0,A
CLR RI               ;清收中断标志

```

OUT:

```

POP ACC
RETI

```

BCDto7_TAB:

DB 0xC0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x90,0xFF;0~9,不显示 (共阳数码管段码)

DELY10ms:

```

MOV R6,#10          ;延时
D2:
MOV R7,#248
DJNZ R7,$
DJNZ R6,D2
RET

```

END

四、 系统调试结果

4.1 Keil 软件调试

程序按照模块进行调试，先将每个模块的程序编辑出来，然后再进行调试，如果各个模块都没有错误时，再进行整合，最后再总体进行调试。

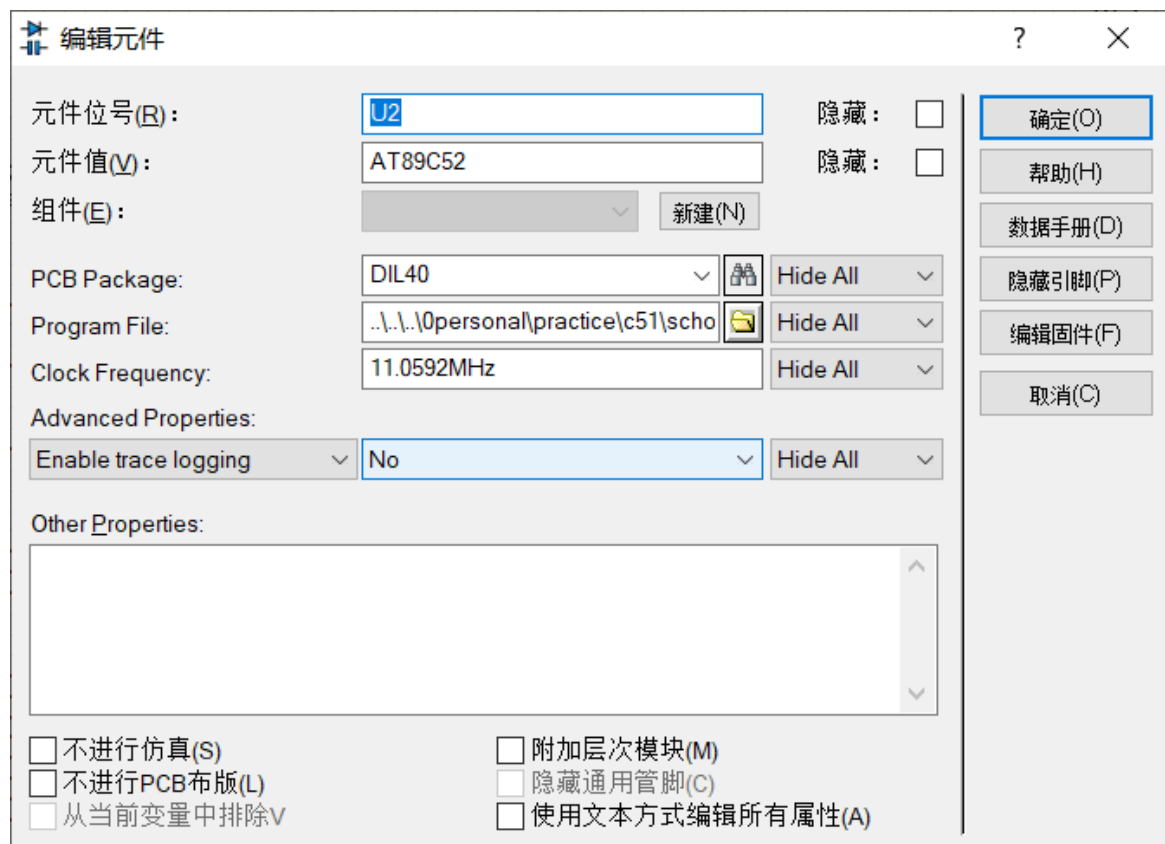
4.2 Proteus 软件仿真过程

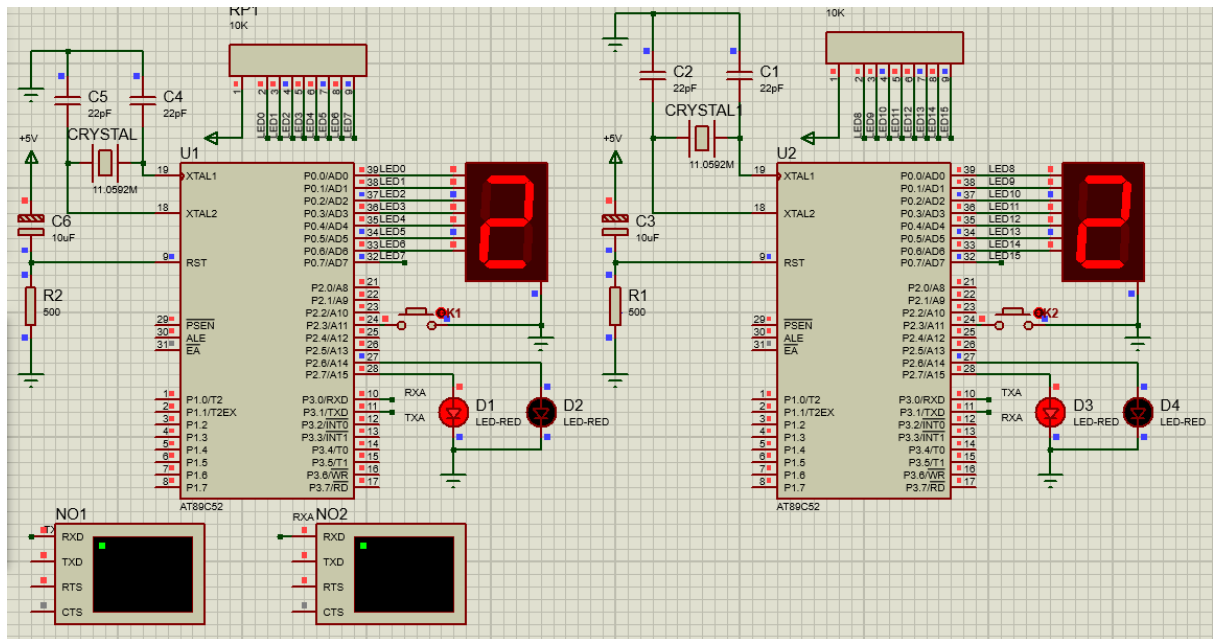
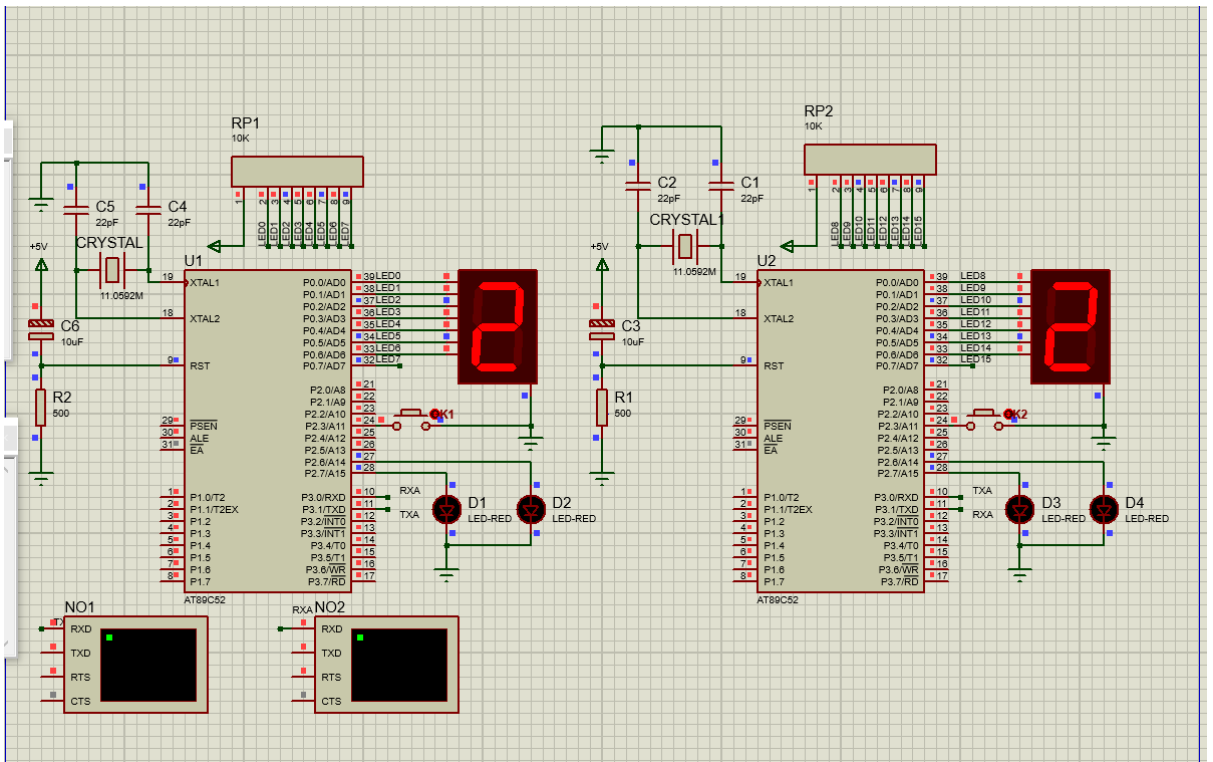
序号	名称	型号	数目/个	用途
1	单片机	AT89C51	2	控制核心
2	晶振	11.0592MHZ	2	晶振电路
3	电容	22pF	4	晶振电路
4	电解电容	10 μ F	2	复位电路
5	排阻	10K	2	上拉电阻
6	电阻	500 欧	2	分压
7	数码管	七段共阴	2	显示
8	Led 灯	Led	4	显示
9	按键	Button	2	控制
11	底座	单片机底座	1	安装芯片

4.3 仿真结果

1.根据原理图安置芯片及连线。原理图见附录 B。

2.接入代码：在 Keil 运行代码，代码运行无误会生成 HEX 文件，然后再 proteus 上双基单片机模型，会弹出一个如下对话框。点击文件按钮，选择所需要的程序文件（.HEX）。选择合适的工作频率即可确认,然后运行调试。





五、 总结与体会

5.1 心得体会

单片机 (Single-Chip Microcomputer) 是一种集成电路芯片, 是采用超大规模集成电路技术把具有数据处理能力的中央处理器 CPU、随机存储器 RAM、只读存储器 ROM、多种 I/O 口和中断系统、定时器/计数器等功能, 在工业控制领域广泛应用。单片机出现在我们身边的任何角落, 大到登陆火星“毅力号”火星车, 小到随处可见的共享单车, 只要需要控制的地方, 就离不开单片机, 更不用说汽车了。现在一辆汽车上的单片机数目超过了上百个 (行车电脑), 对于从事相关电子行业的人来说, 可以说是必须要点亮的知识树!

作为一名电气专业的大三学生, 我觉得做单片机课程设计是十分有意义的, 而且是十分必要的。单片机课程设计可以把我们所学的专业基础理论知识运用到实践中。在电路中用 Proteus 软件实现了仿真, 在这过程中, 使我对电路设计和单片机的使用等都有了新的认识。同时通过这次设计我学会了 Proteus 和 Keil 软件的使用方法, 掌握了从系统的需要、方案的设计、功能模块的划分、原理图的设计和电路图的仿真的设计流程, 积

累了不少经验。

在这次课程设计中，我的选题是双机双向通讯。刚开始拿到这个题目的时候，我想的做法是通过电平检测来制作，但是仔细研究了题目之后发现，这次的题目要求是需要我们利用单片机的串口通讯来进行制作。

在做本次课程设计的过程中，我感触最深的当属查阅大量的设计资料了。为了让自己的设计更加完善，查阅这方面的设计资料是十分必要的，同时也是必不可少的。我们是在做单片机课程设计，一切都要有据可依，有理可寻，不切实际的构想永远只能是构想，永远无法升级为设计。在寻找相关的外设如何使用时，我查阅了大部分资料，并且看了许多例程，来为我的设计添加功能。

通过本次设计，我对单片机这门课有了进一步的了解。无论是在硬件连接方面还是在软件编程方面。本次设计采用了STC89C52RC单片机芯片，与以往的单片机课程实验相比增加了许多新的功能，使其功能更为完善，应用领域也更为广泛。

通过这次设计，我懂得了学习的重要性，了解到理论知识与实践相结合的重要意义，学会了坚持、耐心和努力，这将为自己今后的学习和工作做出了最好的榜样。我觉得作为一名电牵专业的学生，单片机的课程设计是很有意义的。更重要的是如何把自己平时所学的东西应用到实际中。

参考文献

- [1] 徐春辉, 陈忠斌, 章海亮. 单片微型计算机原理及应用. 北京: 电子工业出版社, 2017.8
- [2] 蔡明文, 冯先成. 单片机课程设计. 武汉: 华中科技大学出版社, 2007.3
- [3] 杨居义. 单片机课程设计指导. 北京: 清华大学出版社, 2009.9
- [4] 林立, 张俊亮. 单片机原理及应用: 基于 Proteus 和 Keil C. 北京: 电子工业出版社, 2013
- [5] 楼然苗, 李光飞. 单片机课程设计指导. 北京: 北京航空航天大学出版社, 2012.1
- [6] 周润景, 张丽娜, 丁莉. 基于 PROTEUS 的电路及单片机设计与仿真. 北京: 北京航空航天大学出版社, 2009.12

附录

程序一：

SMG EQU P0 ;IO 定义

LED1 BIT P2.7 ;

LED2 BIT P2.6

BUTTON BIT P2.3 ;

ORG 0000H

LJMP START

ORG 0023H

LJMP UART_ISR

ORG 0030H

START:

MOV SP,#60H

MOV TMOD,#20H ;定时器 1 设为方式 2

MOV TL1,#0FDH ;波特率为 9600, 教材 210 页

表

MOV TH1,#0FDH ;

MOV PCON,#00H ;波特率不加速

MOV SCON,#50H ;串口设为方式 1

SETB ES ;开串口中断

SETB TR1 ;启动定时器 1

```

        SETB  EA                                ;开总中断

        SETB  BUTTON                            ;读取前置高

        MOV   R0,#0                             ;接收数据

        MOV   R1,#0                             ;发送数据

        MOV   R2,#0                             ;闪烁时间变量

        MOV   DPTR,#BCDto7_TAB

MAIN:
        MOV   A,R0                               ;
        MOVC  A,@A+DPTR                         ;取数据
        CPL   A
        MOV   SMG,A

        ACALL DELY10ms                          ;调用 delay 延时函数

        CJNE  R1,#0,NE1                         ;LED 全灭

        CLR   LED1
        CLR   LED2
        JMP   KEYF

NE1:
        CJNE  R1,#1,NE2                         ;LED 13 闪烁

        CLR   LED2
        INC   R2

        CJNE  R2,#20,KEYF                       ;闪烁时间间隔约 20ms

        MOV   R2,#0
        CPL   LED1
        JMP   KEYF

NE2:

```

```

        CJNE R1,#2,NE3                ;LED 24 闪烁

        CLR  LED1
        INC  R2
        CJNE R2,#20,KEYF
        MOV  R2,#0
        CPL  LED2
        JMP  KEYF

NE3:

        CJNE R1,#3,KEYF                ; LED 同时闪烁

        INC  R2
        CJNE R2,#20,KEYF
        MOV  R2,#0
        CPL  LED1
        CPL  LED2
;        MOV  C,LED1
;        MOV  LED2,C

KEYF:

        JB   BUTTON,NOPRESS            ;按键按下

        INC  R1
        CJNE R1,#4,NEX
        MOV  R1,#0

NEX:

        MOV  A,R1
;        ADD  A,#'0'

        MOV  SBUF,A                    ;发送数据

        JNB  TI,$                      ;等待发送完成

        CLR  TI
        JNB  BUTTON,$

NOPRESS:

```

AJMP MAIN

UART_ISR:

PUSH ACC

JNB RI,OUT ;不是收中断

MOV A,SBUF ;收数据

; SUBB A,#'0' ;转 ASCII

MOV R0,A

CLR RI ;清收中断标志

OUT:

POP ACC

RETI

BCDto7_TAB:

DB 0xC0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x90,0xFF;0~9,不显示

DELY10ms:

MOV R6,#10 ;延时

D2:

MOV R7,#248

DJNZ R7,\$

DJNZ R6,D2

RET

END

程序二：

SMG EQU P0 ;IO 定义， EQU 相当于 define

LED1 EQU P2^7 ;bit 位地址符号伪指令， 将位地址赋给字符

名称

LED2 EQU P2^6

BUTTON BIT P2.3

ORG 0000H

LJMP START

ORG 0023H

LJMP UART_ISR

ORG 0030H

START:

MOV SP,#70H

MOV TMOD,#20H ;定时器 1 设为方式 2

MOV TL1,#0FDH ;波特率为 9600,教材 210 页

表

MOV TH1,#0FDH

MOV PCON,#00H ;波特率不加速,pcon 为电源

控制寄存器， 令 smod=0， 系统波特率不加倍

MOV SCON,#50H ;串口设为方式 1 双机通讯接

收控制允许， 0101 0000 参考教材 203 页

SETB ES	;开串口中断,置 1 操作
SETB TR1	;启动定时器 1
SETB EA	;开总中断
SETB BUTTON	;读取前置高, 定义按键初值为

1

MOV R0,#0	;接收数据
MOV R1,#0	;发送数据
MOV R2,#0	
MOV DPTR,#BCDto7_TAB	

MAIN:

MOV A,R1	
MOVC A,@A+DPTR	;变址寻址, 查表指令, 取数据
CPL A	;直接寻址位取反 (使用共阴数码管

按位取反)

MOV SMG,A	;将 A 值赋值给 P0 端口
LCALL DELY10ms	;调用 delay 延时函数
CJNE R0,#0,NE1	;条件转移指令, 比较 R0 与 0, 若不

相等则转移

CLR LED1	
CLR LED2	;LED 全灭

```

        JMP    KEYF
NE1:

        CJNE R0,#1,NE2                ;LED 13 闪烁

        CLR   LED2
        INC   R2

        CJNE R2,#20,KEYF              ;闪烁时间间隔约 20ms

        MOV   R2,#0
        CPL   LED1
        JMP   KEYF

NE2:

        CJNE R0,#2,NE3                ;LED 24 闪烁

        CLR   LED1
        INC   R2
        CJNE R2,#20,KEYF

        MOV   R2,#0
        CPL   LED2
        JMP   KEYF

NE3:

        CJNE R0,#3,KEYF                ; LED 同时闪烁

;        CLR   LED1
;        CLR   LED2
        INC   R2
        CJNE R2,#20,KEYF

        MOV   R2,#0
        CPL   LED1
        CPL   LED2
;        MOV   C,LED1
;        MOV   LED2,C

KEYF:                                     ;keyf 判断按键按下的函数

```


JB BUTTON,NOPRESS ;位条件转移指令, 若

button 为 1, 则转移

```
INC R1
CJNE R1,#10,NEX
MOV R1,#0
```

NEX:

```
MOV A,R1
; ADD A,#'0'

MOV SBUF,A ;发送数据

JNB TI,$ ;等待发送完成

CLR TI
JNB BUTTON,$
```

NOPRESS:

```
AJMP MAIN
```

UART_ISR:

```
PUSH ACC ;压栈指令

JNB RI,OUT ;若 RI 为 0 则转移收

MOV A,SBUF ;收数据

; SUBB A,#'0' ;转 ASCII, 为了在 pc 上显示, 但实际调用时
```

有一点问题

```
MOV R0,A

CLR RI ;清收中断标志
```

OUT:

```
POP    ACC
RETI
```

BCDto7_TAB:

```
DB 0xC0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x90,0xFF;0~9,不显示
```

(共阳数码管段码)

DELY10ms:

```
MOV    R6,#10        ;延时
```

D2:

```
MOV    R7,#248
```

```
DJNZ R7,$
```

```
DJNZ R6,D2
```

```
RET
```

END