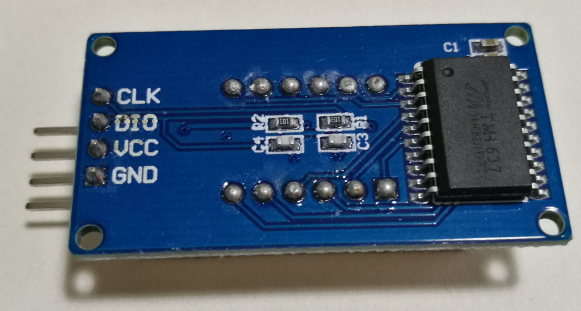
### TM1637驱动四位数码管

一、认识TM1637四位数码管



TM1637是专用电路，内置MCU。采用功率CMOS工艺，显示模式(8段×6位)，支持共阳数码管输出，键扫描(8×2bit)，增强型抗干扰按键识别电路，辉度调节电路(占空比8级可调)，两线串行接口(CLK,DIO)。

**电路连接**

开发板:PB6--TM1637:CLK

开发板:PB7--TM1637:DIO

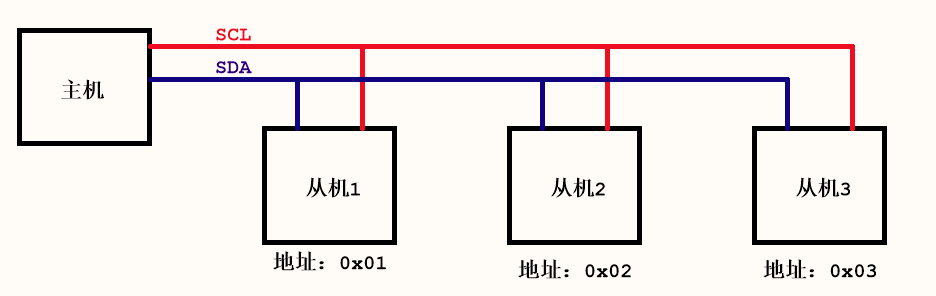
开发板: V3 --TM1637:VCC

开发板: G--TM1637:GND

1. I2C总线介绍

I2C总线是由Philips公司开发的一种简单、双向二线制同步串行总线。它只需要两根线即可在连接于总线上的器件之间传送信息。传输速率在标准模式下可达100Kbit/s，快速模式下可达400Kbit/s，高速模式下可达3．4Mbit/s。

SDA（串行数据线）和SCL（串行时钟线）都是双向I/O线，接口电路为开漏输出．需通过上拉电阻接电源VCC．当总线空闲时．两根线都是高电平。

I2C 一般应用于芯片之间的通信，它的传输距离短，但其好处是 IIC 支持一主多从的挂载方式，因此主机和多个从机之间的通信线只要两条就够了，示意图如下：

三、I2C总线时序图、电平信号

1.起始信号START

在时钟线 SCL 高电平期间数据线 SDA 发生下降沿跳变产生起始信号

void startIIC(void)

{

IIC\_SDA\_OUT(); //设置SDA引脚为开漏输出

IIC\_SCL\_H(); //拉高SCL

IIC\_SDA\_H(); //拉高SDA,目前是空闲状态

Delay\_us(IIC\_SPEED); //延时一段时间

IIC\_SDA\_L(); //拉低SDA，在SCL高电平器件产生下降沿

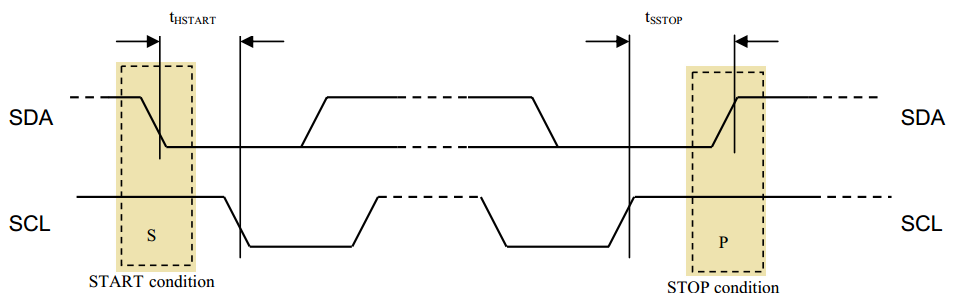
Delay\_us(IIC\_SPEED); //延时一段时间

IIC\_SCL\_L(); //拉低时钟，完成一个时钟周期,准备传输数据

}

2.结束信号STOP

在时钟线 SCL 高电平期间数据线 SDA 发生上升沿跳变产生停止信号



void stopIIC(void)

{

IIC\_SDA\_OUT();

IIC\_SCL\_L();

Delay\_us(IIC\_SPEED);

IIC\_SDA\_L(); //SDA维持低电平

Delay\_us(IIC\_SPEED);

IIC\_SCL\_H(); //当SCL由低电平高电平时,SDA由低变高则代表停止信号

Delay\_us(IIC\_SPEED);

IIC\_SDA\_H();

//Delay\_us(IIC\_SPEED);

}

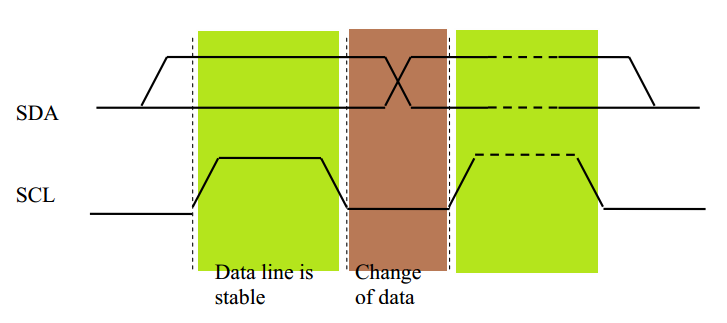
3.数据改变信号

数据线 SDA 上的二进制数据要在时钟线 SCL 为低电平期间发生改变 。

4.数据传输信号

在数据传输期间，时钟线 SCL 为高电平期间，如果数据线 SDA 为高电平则代表二进制

1. 同理，时钟线 SCL 为高电平期间，如果数据线 SDA 为低电平则代表二进制 0。



/\*\*

\* 功能：发起一个字节数据

\* 参数： onebyte：待发送数据

\*/

void sendIICByte(u8 onebyte)

{

u8 i;

IIC\_SDA\_OUT();

for(i=0;i<8;++i)

{

IIC\_SCL\_L(); //时钟信号为低电平时,准备改变数据位IIC\_SDA

Delay\_us(IIC\_SPEED);

if(onebyte & 0x01) //判断第0位数据,传输第0位数据

{

IIC\_SDA\_H(); //如果该位为1,输出高电平

}else

{

IIC\_SDA\_L(); //如果该位为0,输出低电平

}

Delay\_us(IIC\_SPEED); //保持数据稳定

IIC\_SCL\_H(); //时钟信号保持高电平,传输该位数据

Delay\_us(IIC\_SPEED);

IIC\_SCL\_L();

Delay\_us(IIC\_SPEED);

onebyte >>= 1; //数据右移1位,准备下一个1bit数据

}

}

4.应答信号ACK

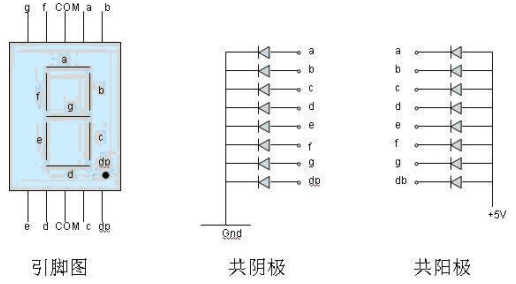
在时钟线 SCL 为高电平期间数据线 SDA 保持低电平为应答信号

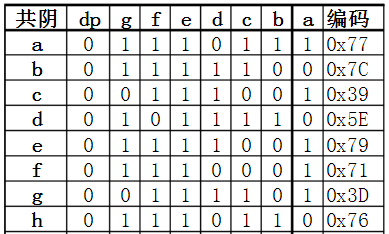
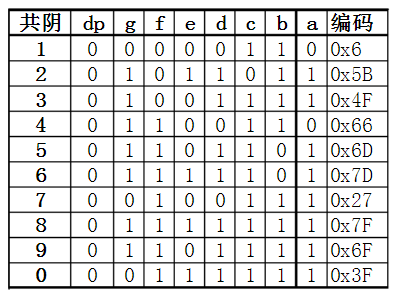
5.非应答信号NACK

在时钟线 SCL 为高电平期间数据线 SDA 保持高电平为非应答信号

四、TM1637驱动程序TM1637.c部分代码解读

1. TM1637\_Init() 实现TM1637引脚初始化
2. TM1637\_WriteCommand(0x40) 给TM1637模块发出命令指令，指示按固定地址进行控制显示位。
3. TM1637\_WriteCommand(0x80) 关闭TM1637模块显示
4. TM1637\_WriteCommand(0x8A) 控制TM1637模块显示亮度（0x8A - 0x8F）
5. TM1637\_WriteData( n , index) 在第n位数码管显示序号为index的字符
6. Set\_Point(Flag) 控制冒号是否显示，Flag为1显示，为0不显示冒号
7. DisplayChar(unsigned char address\_index, unsigned char data\_index)
8. 七段数码管结构与编码





unsigned char NumDis[18]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0X6F,\

0x77,0x7C,0x39,0x5E,0x79,0x71,0x76,0x0

};//0123456789ABCDEFH空格(不显示)