

I2C总线

I2C总线(Inter-IC Bus)是一种通用的串行总线，是用于IC器件之间连接的二线制总线。他通过串行数据线(Serial Data Lines, SDL)及串行时钟线(Serial ClockLine, SCL)两线在连接到总线上的器件之间传送信息，并根据地址识别每个器件。

术语/terminology

master/slave 又叫做 主机/从机 主控制器/被控制器 master的作用：

- 获取I2C总线，其实就是发送一个启动信号；发送操作命令，其实就是发送被操作的从机的地址以及操作方式，读或者写；释放总线，其实就是发送一个停止信号。
- 发送时钟，一直控制着clk线。

slave的作用：

- 响应master的相关操作。例如，在master发送操作命令之后，slave给出一个应答信号（ack）。

transmit/receive 发送端/接收端 transmit的作用：

- 控制dat线上的电平，将数据传送到dat线上。

receiver的作用：

- 获取dat线上的电平，得到数据。

master/slave既能够当做发送端，又可以当做接收端。在一次操作的过程中，指的是从启动信号开始一直到停止信号结束，master和slave双方是不能够改变的，但是transmit和receive的角色是可能反复交换的。详情请参考I2C总线工作时序。

I2C总线工作时序

在I2C总线通信的过程中，参与通信的双方互相之间所传输的信息种类归纳如下。

主控器向被控器发送的信息种类有：启动信号、停止信号、7位地址码、读 / 写控制位、10位地址码、数据字节、重新启动信号、应答信号、时钟脉冲。

被控器向主控器发送的信息种类有：应答信号、数据字节、时钟低电平。

I2C实例

MCU通过I2C总线控制一个外围设备。MCU上集成了I2C模块，说明无需使用GPIO来模拟I2C总线。MCU来控制外围设备，说明MCU总是作为master，以后简称master，外围设备也有用I2C接口，总是作为slave，以后简称slave端。

注意：MCU上的I2C模块一般能够通过寄存器配置，作为master端或者slave端，在阅读该模块说明时，要注意区分理解。

实例一：探测slave端的地址

- 1.初始化I2C模块，重点是中断位和CLK线的时钟。
- 2.发送启动信号，查询启动信号成功。
- 3.发送从0~127（7位地址）的地址+操作位
- 4.检测应答信号是否成功，如果成功，则说明在I2C总线上有该地址的slave。
- 5.重复步骤3，可以检测I2C总线上所有的slave的地址。
- 6.发送停止信号，结束MCU对I2C总线的控制。

说明：以上只是一个大概的步骤，如果I2C模块已经初始化了，则无需第一步了。第二步中的查询启动信号成功和第四步中的检测应答信号是否收到，一般都是通过中断来判断是否有状态更新，然后查看状态信息对应的寄存器。

实例二：读/写slave端

- 1.初始化I2C模块，重点是中断位和CLK线的时钟。
- 2.发送启动信号，查询启动信号是否成功。
- 3.发送slave的地址+操作位（0表示写，1表示读）。
- 4.检测应答信号是否成功，如果成功，则说明在I2C总线上有该地址的slave，并且做出了应答。
- 5.如果是读操作，那么就等待中断的到来，判断状态位是否标示读操作完成。如果是，那么就可以从I2C模块的数据寄存器（肯定都有这个寄存器）中获取这个数据。如果是写操作，则将数据写入I2C模块的数据寄存器，然后检测应答信号是否成功。
- 6.可以重复第五步进行重复的读或者写。
- 7.发送停止信号，结束MCU对I2C总线的控制。

说明：以上需要说明的是，一般情况，中断标志位在每一次操作后都需要手动清零。

内容
I2C总线术语/terminology I2C总线工作时序 I2C实例