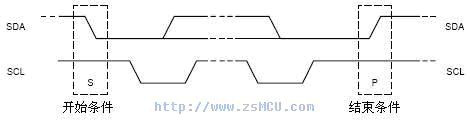
**I2C协议**

   2条双向串行线，一条数据线SDA，一条时钟线SCL。  
   SDA传输数据是大端传输，每次传输8bit，即一字节。  
   支持多主控(multimastering)，任何时间点只能有一个主控。  
   总线上每个设备都有自己的一个addr，共7个bit，广播地址全0.  
   系统中可能有多个同种芯片，为此addr分为固定部分和可编程部份，细节视芯片而定，看datasheet。  
  
**1.1 I2C位传输**  
   数据传输：SCL为高电平时，SDA线若保持稳定，那么SDA上是在传输数据bit；  
   若SDA发生跳变，则用来表示一个会话的开始或结束（后面讲）  
   数据改变：SCL为低电平时，SDA线才能改变传输的bit

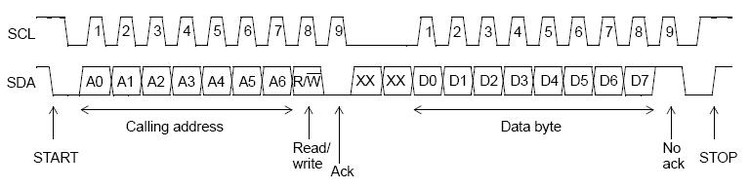


**1.2 I2C开始和结束信号**  
   开始信号：SCL为高电平时，SDA由高电平向低电平跳变，开始传送数据。  
   结束信号：SCL为高电平时，SDA由低电平向高电平跳变，结束传送数据。

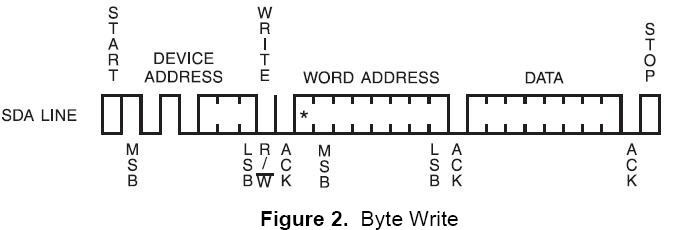


**1.3 I2C应答信号**

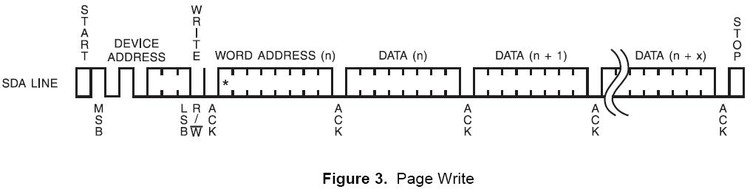
   Master每发送完8bit数据后等待Slave的ACK。  
   即在第9个clock，若从IC发ACK，SDA会被拉低。  
   若没有ACK，SDA会被置高，这会引起Master发生RESTART或STOP流程，如下所示：



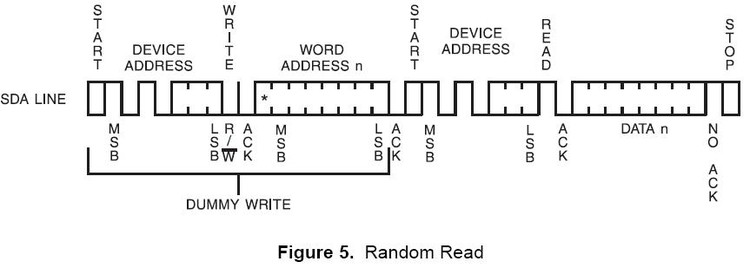
**1.4 I2C写流程**  
写寄存器的标准流程为：  
1.    Master发起START  
2.    Master发送I2C addr（7bit）和w操作0（1bit），等待ACK  
3.    Slave发送ACK  
4.    Master发送reg addr（8bit），等待ACK  
5.    Slave发送ACK  
6.    Master发送data（8bit），即要写入寄存器中的数据，等待ACK  
7.    Slave发送ACK  
8.    第6步和第7步可以重复多次，即顺序写多个寄存器  
9.    Master发起STOP  
  
***写一个寄存器***



***写多个寄存器***



**1.5 I2C读流程**  
  
读寄存器的标准流程为：  
1.    Master发送I2C addr（7bit）和w操作1（1bit），等待ACK  
2.    Slave发送ACK  
3.    Master发送reg addr（8bit），等待ACK  
4.    Slave发送ACK  
5.    Master发起START  
6.    Master发送I2C addr（7bit）和r操作1（1bit），等待ACK  
7.    Slave发送ACK  
8.    Slave发送data（8bit），即寄存器里的值  
9.    Master发送ACK  
10.    第8步和第9步可以重复多次，即顺序读多个寄存器  
  
***读一个寄存器***



***读多个寄存器***

