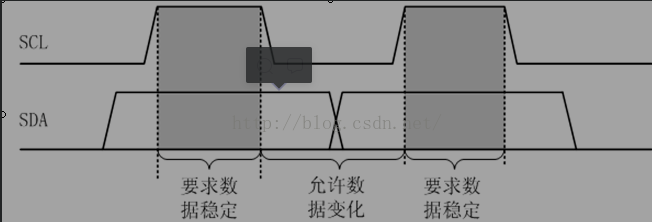
IIC时序

1.数据的有效性

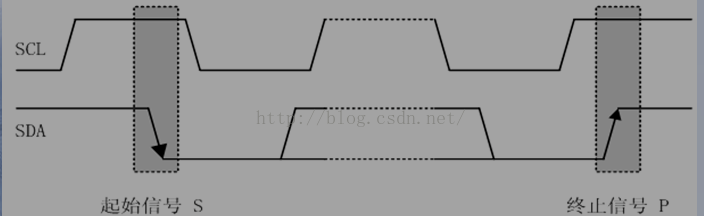
在时钟的高电平周期内，SDA线上的数据必须保持稳定，数据线仅可以在时钟SCL为低电平时改变。如图所示：



2.起始和结束条件

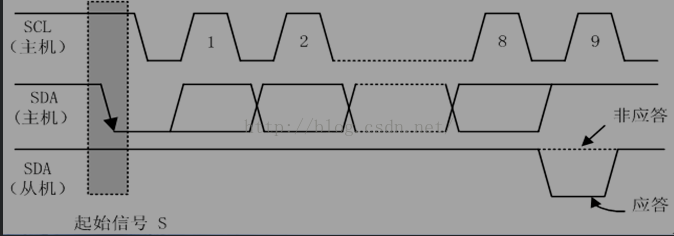
起始条件：当SCL为高电平的时候，SDA线上由高到低的跳变被定义为起始条件。

结束条件：当SCL为高电平的时候，SDA线上由低到高的跳变被定义为停止条件，要注意起始和终止信号都是由主机发出的，连接到I2C总线上的器件，若具有I2C总线的硬件接口，则很容易检测到起始和终止信号。总线在起始条件之后，视为忙状态，在停止条件之后被视为空闲状态，对起始条件和结束条件的描述如下图所示。

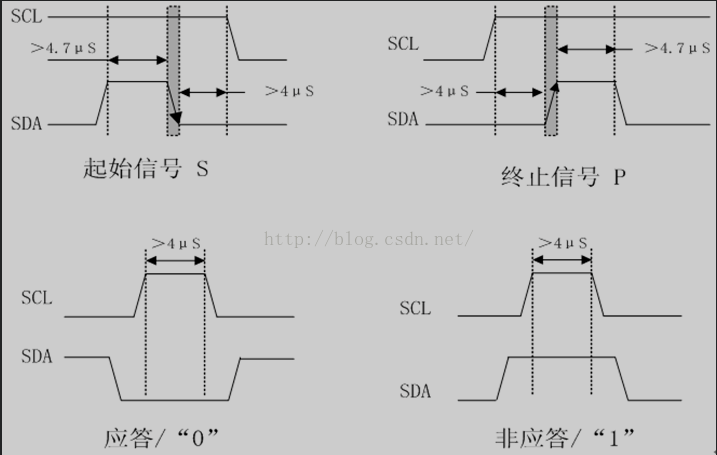


3. 应答

每当主机向从机发送完一个字节的数据，主机总是需要等待从机给出一个应答信号，以确认从机是否成功接收到了数据，从机应答主机所需要的时钟仍是主机提供的，应答出现在每一次主机完成8个数据位传输后紧跟着的时钟周期，低电平0表示应答，1表示非应答，如图所示。



4.时序

5.

IIC协议规定，当接受到一个字节（8bit）后，数据接收方必须向数据发送方返回一个低电平信号，此信号称作应答信号（表示上一个数据成功接受可以继续接受）。若未返回应答信号，则认为数据接收方出现故障。由于单片的这端是IIC程序，而外设那端是IIC电路，所以当单片机发送数据时，外设的IIC电路会自动返回应答信号（前提外设没故障）。当单片机接收数据的时候，应答信号就得我们自己写了。

先把时钟线拉低，再把数据线拉低，最后把时钟线先拉高，这样就告诉外设赶紧把数据线上的低电平才进去，应答信号就这样返回了。

6.等待应答信号

单片机发送完一个字节后面必须跟一个等外应答函数，万一外设挂了呢，单片机还在傻傻的发送。

7.接收函数

跟发送一样，只是把数据一位一位接受进来，记得要返回应答信号哟。

首先我们要确定这个字节接收完毕后还需不需要继续接受字节，继续ACK=1,不继续ACK=0。循环中，时钟线拉低，先允许外设把数据线0、1变换，在时钟线拉高，禁止数据线变化（把外设送到数据线上的电平固定住）。 当i=0时，receive<<=1;不起任何作用，但是以后就有用了，有大用处。再判断下数据线上电平是高还是低，假设IIC\_SDA=1，则receive++就是把外设输出的1方到receive的最低位上去，这样一位数据就接受进来了。循环第二次，此时i=1，仍旧数据线拉低，再拉高，先允许变化再固定，\*\*receive<<=1起作用了，把刚才接受到的1移到次低位上去，给即将要接收的电平腾个地，\*\*之后的在判断什么什么的就都一样了哈，读者自己分析。八次循环以后，一个字节就接受到了。别忘了应答信号哟。最后把接受到了的数据返回，则一个字节就真正接收到了。

8.释放总线

IIC是开漏结构，释放总线必须把SDA线拉高。

9. 主机发送数据流程

（1）主机在检测到总线为“空闲状态”（即 SDA、SCL 线均为高电平）时，发送一个启动信号“S”，开始一次通信的开始

（2）主机接着发送一个命令字节。该字节由 7 位的外围器件地址和 1 位读写控制位 R/W组成（此时为写 R/W=0）

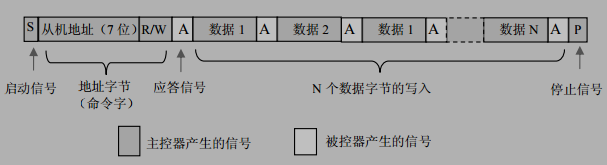
（3）相对应的从机收到命令字节后向主机回馈应答信号 ACK（ACK=0）

（4）主机收到从机的应答信号后开始发送第一个字节的数据

（5）从机收到数据后返回一个应答信号 ACK

（6）主机收到应答信号后再发送下一个数据字节

（7）当主机发送最后一个数据字节并收到从机的 ACK 后，通过向从机发送一个停止信号P结束本次通信并释放总线。从机收到P信号后也退出与主机之间的通信



注意：①主机通过发送地址码与对应的从机建立了通信关系，而挂接在总线上的其它从机虽然同时也收到了地址码，但因为与其自身的地址不相符合，因此提前退出与主机的通信；②主机的一次发送通信，其发送的数据数量不受限制。主机是通过 P 信号通知发送的结束，从机收到 P 信号后退出本次通信；③主机的每一次发送后都是通过从机的 ACK 信号了解从机的接收状况，如果应答错误则重发。

10. 主机接收数据流程

（1）主机发送启动信号后，接着发送命令字节（此时为读 R/W=1）

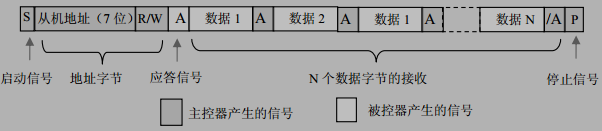
（2）对应的从机收到地址字节后，返回一个应答信号并向主机发送数据

（3）主机收到数据后向从机反馈一个应答信号

（4）从机收到应答信号后再向主机发送下一个数据

（5）当主机完成接收数据后，向从机发送一个“非应答信号（ACK=1）”，从机收到ASK=1 的非应答信号后便停止发送

（6）主机发送非应答信号后，再发送一个停止信号，释放总线结束通信



注意：主机所接收数据的数量是由主机自身决定，当发送“非应答信号/A”时从机便结束传送并释放总线（非应答信号的两个作用：前一个数据接收成功，停止从机的再次发送。

**IIC\_OLED**

1.OLED的地址和寄存器地址

既然是使用IIC进行通讯，就必须在写数据的时候依次发送OLED地址和寄存器地址）。

2. 起来就是，每次写命令或数据都要有如下步骤

1. IIC起始信号
2. IIC发送OLED地址：0x78
3. 应答信号
4. IIC发送寄存器地址（命令或数据）
5. 应答信号
6. IIC发送命令或数据
7. 应答信号
8. IIC结束信号

