1、AT24C02简介

  AT24C01/02/04/08/16是一个1K/2K/4K/8K/16K位(AT24C02大小为256字节)串行CMOS E2PROM内部含有128/256/512/1024/2048个8位字节，CATALYST公司的先进CMOS技术实质上减少了器件的功耗AT24C01有一个8字节页写缓冲器 AT24C02/04/08/16有一个16字节页写缓冲器该器件通过I2C总线接口进行操作有一个专门的写保护功能。

2、AT24C02特性

  ①、与 400KHz I2C 总线兼容

  ②、1.8 到 6.0 伏工作电压范围

  ③、低功耗 CMOS 技术

  ④、写保护功能 当 WP 为高电平时进入写保护状态

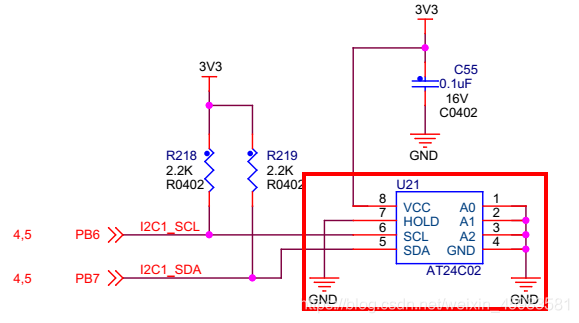
  ⑥、页写缓冲器

  ⑦、自定时擦写周期

  ⑧、1,000,000编程/擦除周期

  ⑨、可保存数据100年

3、硬件原理图

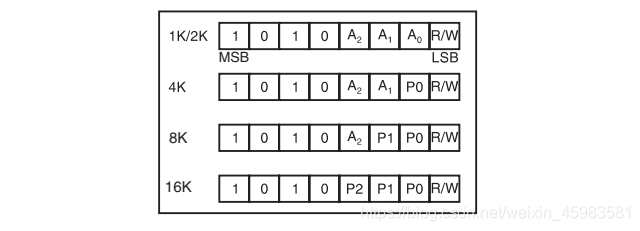


**4、管脚描述**



A0、A1、A2、三个脚不同的电平可以形成不同的地址，最多8种。原理图中A0、A1、A2、三个脚全部接地，也就是硬件决定不可更改，默认为A0=0、A1=0、A2=0。同时写保护脚WP(原理图中为HOLD)，也是接地。因此默认也是关闭写保护的。

5、从器件地址

1. A0、A1和A2对应器件的管脚1、2和3

2. P0、P1、P2对应存储阵列地址字地址。

3. R/W为读写方向位，1为读，0为写。

  对应AT24C02硬件原理图而言，A0、A1、A2、三个脚被硬件接地，即A0=0、A1=0、A2=0。所以AT24C02读地址为1010 0001(0xA1),AT24C02写地址为1010 0000(0xA0)。

  这里再简单的介绍一下P0、P1、P2的含义，对于AT24C02而言大小为256字节，而对于AT24C08而言大小为1024(4\*256)字节。假如P0=0、P1=0,即代表选中第一个256字节的首地址，P0=0、P1=0,即代表选中第一个256的首地址，P0=1、P1=0,即代表选中第二个256的首地址，以此类推。

1. 写周期限制

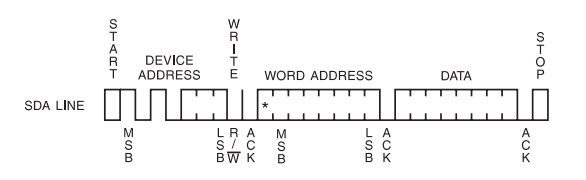
IMG_256

  写周期时间是指从一个写时序的有效停止信号到内部编程/擦除周期结束的这一段时间。在写周期期间，总线接口电路禁能，SDA保持为高电平，器件不响应外部操作。

  简单的说就是，当向AT24C02执行写操作时，从发出写命令到写完成最多10ms。因此在写程序时，每次写操作后都延迟10ms以保证稳定。

7、读写操作时序

 7.1、向指定地址写一个字节



/\*

函数功能：向AT24c02指定地址写一字节数据

\*/

void At24c02\_Write\_OneByte(u8 addr,u8 data)

{

IIC\_Start();

IIC\_Write\_Byte(At24c02\_Write\_Addr);//发送写地址0XA0

if(IIC\_Check\_Ack())return ;

IIC\_Write\_Byte(addr);

if(IIC\_Check\_Ack())return ;

IIC\_Write\_Byte(data);

if(IIC\_Check\_Ack())return ;

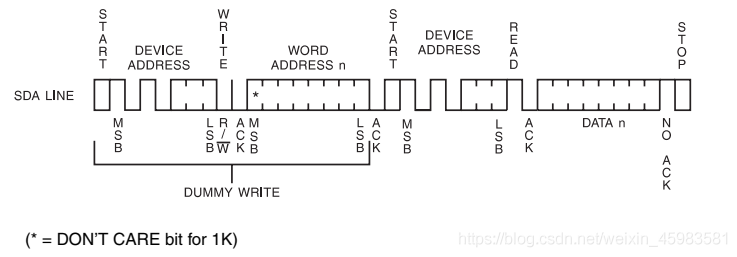
IIC\_Stop();

delay\_ms(10);//确保写完成

}

1.3.2 LPC2368中I2C总线操作  
    在1.1.4中我们已经讲过了对I2C总线的操作，但那只是从协议和时序上的描述，那我们如何从软件上去体现出来呢？接下来我们就讨论这个问题。  
    对I2C总线上主从设备的读写可使用两种方法，一是使用轮询的方式，二是使用中断的方式。轮询方式即是在一个循环中判断I2C状态寄存器当前的状态值来确定总线当前所处的状态，然后根据这个状态来进行下一步的操作。中断方式即是使能I2C中断，注册I2C中断服务程序，在服务程序中读取I2C状态寄存器的当前状态值，再根据状态值来确定下一步的操作。  
    不管使用哪种方法，看来I2C状态寄存器的值是至关重要的。这些状态值代表什么意思呢？下面我们描述一些常用的状态值(详细的状态值含义请参考数据手册)。

0x08: 表明主设备向总线已发出了一个起始条件；  
0x10: 表明主设备向总线已发出了一个重复的起始条件；  
0x18: 表明主设备向总线已发送了一个从设备地址(写方向)并且接收到从设备的应答；  
0x20: 表明主设备向总线已发送了一个从设备地址(写方向)并且接收到从设备的非应答；  
0x28: 表明主设备向总线已发送了一个数据字节并且接收到从设备的应答；  
0x30: 表明主设备向总线已发送了一个数据字节并且接收到从设备的非应答；  
0x40: 表明主设备向总线已发送了一个从设备地址(读方向)并且接收到从设备的应答；  
0x48: 表明主设备向总线已发送了一个从设备地址(读方向)并且接收到从设备的非应答；  
0x50: 表明主设备从总线上已接收一个数据字节并且返回了应答；  
0x58: 表明主设备从总线上已接收一个数据字节并且返回了非应答；

**7.2、向指定地址开始读数据**  


/\*

函数功能：向AT24c02指定地址开始读数据

\*/

void At24c02\_Read\_Data(u8 addr,u8 \*data,u8 len)

{

u8 i=0;

IIC\_Start();

IIC\_Write\_Byte(At24c02\_Write\_Addr);//发送写地址0XA0

if(IIC\_Check\_Ack())return ;

IIC\_Write\_Byte(addr);//写数据的地址

if(IIC\_Check\_Ack())return ;

IIC\_Start();

IIC\_Write\_Byte(At24c02\_Read\_Addr);//发送读地址0XA1

if(IIC\_Check\_Ack())return ;

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

{

data[i]=IIC\_Read\_Byte();

if(i==len-1)IIC\_Send\_Ack(1);

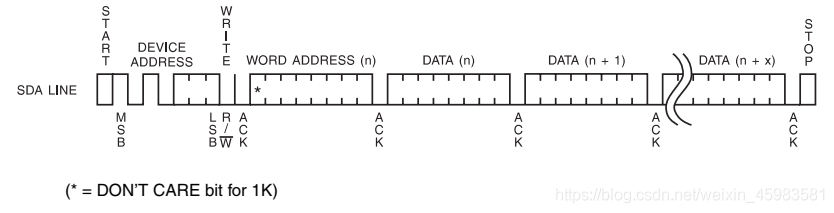
else IIC\_Send\_Ack(0);

}

IIC\_Stop();

}

7.3、页写



 用页写 AT24C01可一次写入8个字节数据 AT24C02/04/08/16可以一次写入16个字节的数据，页写操作的启动和字节写一样，不同在于传送了一字节数据后并不产生停止信号，主器件被允许发送停止信号P,AT24C01 P=7,AT24C02/04/08/16 P=15个额外的字节,每发送一个字节数据后AT24C01/02/04/08/16 产生一个应答位并将字节地址低位加1,高位保持不变。如果在发送停止信号之前主器件发送超过P+1个字节地址计数器将自动翻转，先前写入的数据被覆盖。

  简单的说，AT24C01一页为8字节，AT24C01/02/04/08/16为16字节。对于AT24C02而言，在一页内，每写一个字节写指针+1，当大于16字节时，又回到第一个字节的位置，先前写入的数据将会被覆盖。

/\*

函数功能：对AT24C02指定地址页内写数据

\*/

void At24c02\_Write\_Page(u8 addr,u8 \*data,u8 len)

{

u8 i=0;

IIC\_Start();

IIC\_Write\_Byte(At24c02\_Write\_Addr);//发送写地址0XA0

if(IIC\_Check\_Ack())return ;

IIC\_Write\_Byte(addr);//写数据的地址

if(IIC\_Check\_Ack())return ;

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

{

IIC\_Write\_Byte(\*data++);/

if(IIC\_Check\_Ack())return ;

}

IIC\_Stop();

delay\_ms(10);

}

 7.4、向指定地址跨页写数据

/\*

函数功能：对AT24C02指定地址跨页写数据

\*/

void At24c02\_Write\_Data(u8 addr,u8 \*data,u8 len)

{

u8 write\_len=16-addr%16;//起始页剩下的空间

if(write\_len>len)write\_len=len;

while(1)

{

At24c02\_Write\_Page(addr,data,write\_len);

if(write\_len==len)break;

addr+=write\_len;

data+=write\_len;

len-=write\_len;

if(len>16)write\_len=16;

else write\_len=len;

}

}