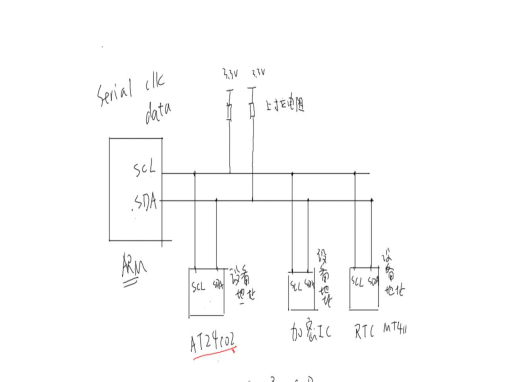
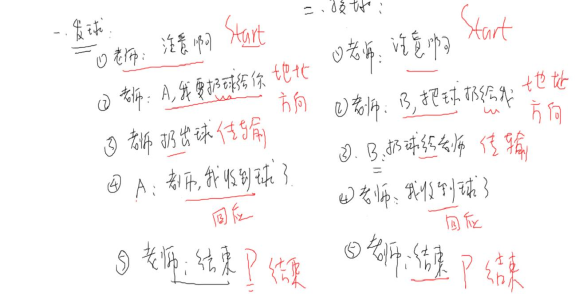
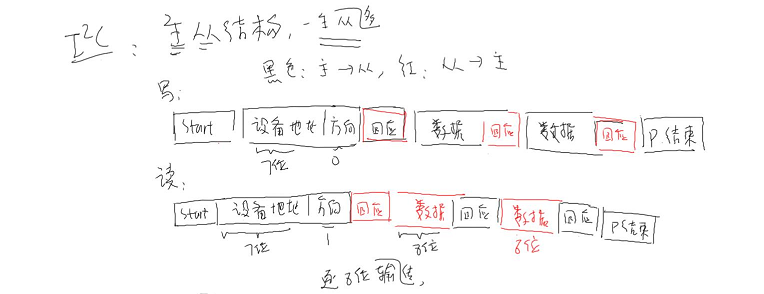
第019课 I2C

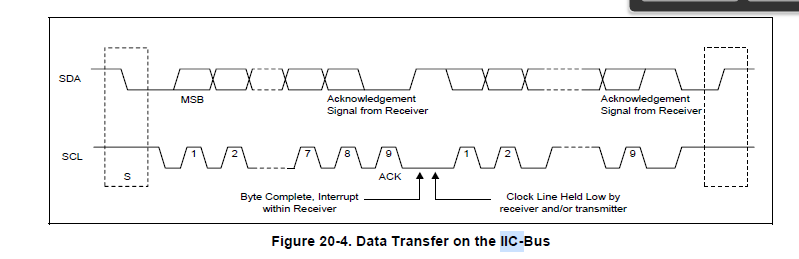
第001节\_I2C协议与EEPROM





IIC主从结构。每次传输由主节点发起





Start 信号：在SCL高电平期间， SDA从高到低

P停止信号: 在SCL高电平期间, SDA从低到高

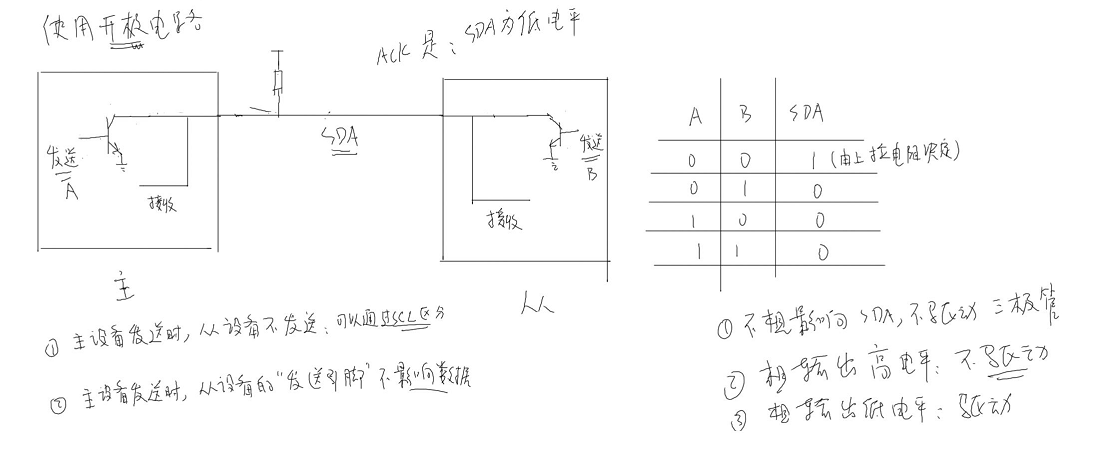
地址为7bit，数据为8bit.用9个时钟传输数据。

先传输高位MSB,第九个始终是应答（7位的地址，第8位表示方向，第9位表示应答.如果应答表示存在此地址）。然后继续发8bit数据,1bit应答以此循环

SCL低电平是，SDA可以变换高电平不能变化

但是主设备发送与从设备发送一条数据线，如何传输？

IIC 规定如何传输数据但是数据的含义由设备决定



1．主设备发送时，从设备不发送，可以通过SCL时钟来区分。（因为应答在第九个时钟发生）

2. 主设备发送时，从设备发送引脚不影响数据

采用开极电路如上图所示，要加上上拉电阻否则位悬空态状态无法确认。

当发送为A端接受为B端。当A发送端为1则A导通所以SDA为电平，为0时截至不导通。B端同理。

当一方不想影响一方的时候，则不用驱动开极电路即可。

当想要SDA为高电平则需要，不需要驱动开极电路则 给0

当主设备发给从设备时：

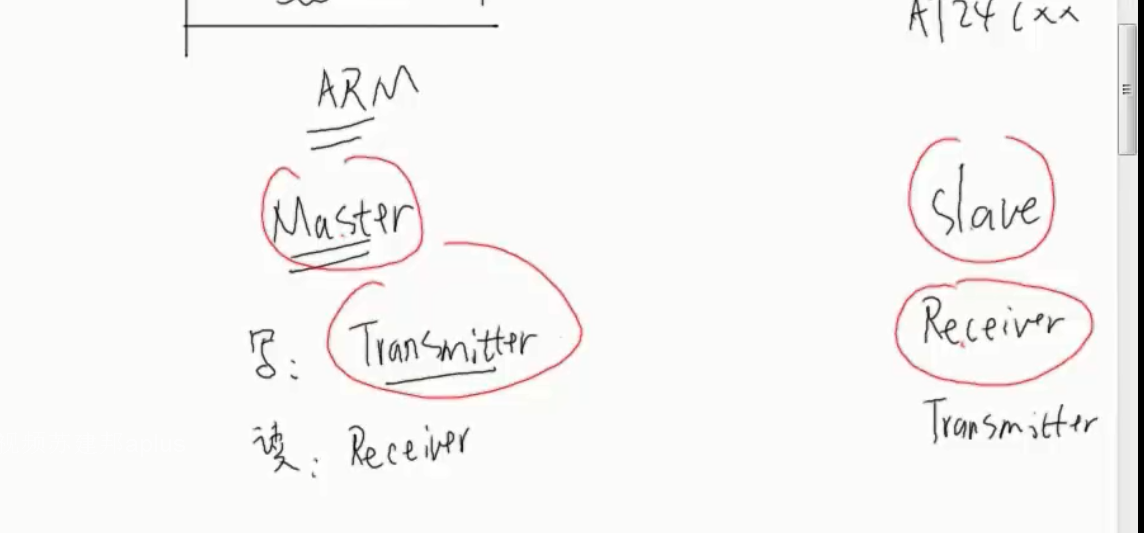
前8clock.从设备不影响，从设备不驱动三极管。主设备决定数据

第9 clock。从设备不驱动三极管，从设备决定数据。

第9个时钟后，假如SCL拉低，则大家都忙不工作。此时SDA为高，当从设备的驱动三极管则SDA为低。从而设备的应答（Ack）为低电平。

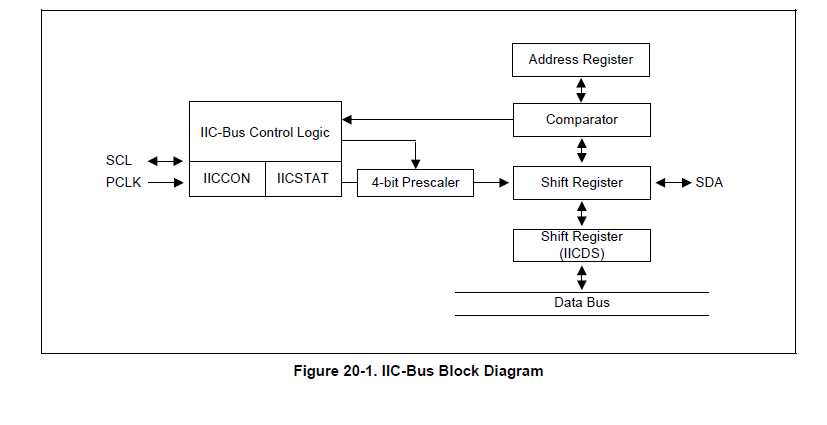
课后作业：阅读AT24CXX芯片手册了解其他读写方法

第002节\_S3C2440的I2C控制器



IIC控制器简化了我们的工作。

主从结构master读写的时候，分别时receive 和transmiter

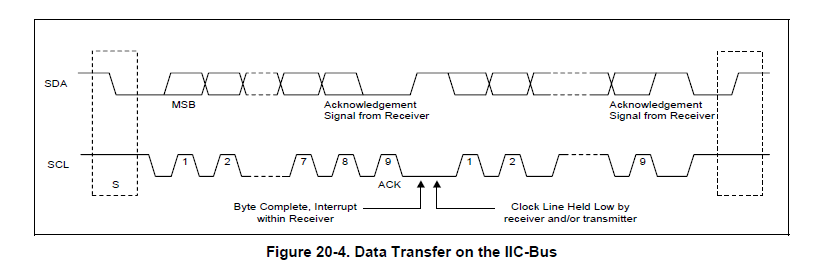


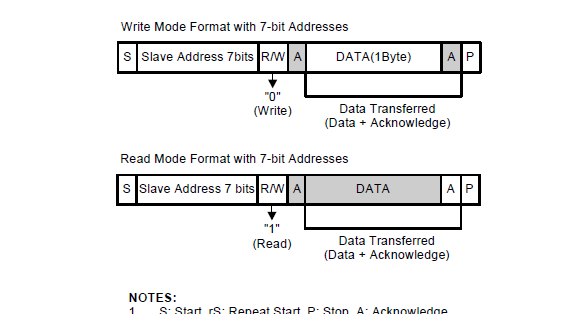
PLCK是50M可以通过分频去调整为合适的SCL。开始信号可以通过设置IICSTAT寄存器控制

要发送书数据的时候将数据填充到IICDS他会自动发出时钟和数据。

可以查询IICSTAT寄存器 可以确定是否收到ACK信号。收到的话就可以继续发送下一个数据，想结束的话IISTAT发送结束信号。

以下是发送的大致流程：





在第9个clock产生中断，SCL为低电平，谁都不能使用IIC，等处理完中断之后。

对于写：若无ACK出错，发出P信号。

若有ACk :1 仍有数据写入IICdata寄存器。清除中断。若无 结束

对于读：回应Ack 1.还有未读完的数据。清除中断启动传输 读IICdata寄存器 2.没有要读的发出p.

重点：发生IIC中断时SCL被拉低，防止继续使IIC.清除中断后才能继续使用

**READ-WRITE OPERATION：**

In Transmitter mode, when the data is transferred, the IIC-bus interface will wait until IIC-bus Data Shift (IICDS)

register receives a new data. Before the new data is written into the register, the SCL line will be held low, and

then released after it is written. The S3C2440A should hold the interrupt to identify the completion of current data

transfer. After the CPU receives the interrupt request, it should write a new data into the IICDS register, again.

In Receive mode, when data is received, the IIC-bus interface will wait until IICDS register is read. Before the new

data is read out, the SCL line will be held low and then released after it is read. The S3C2440A should hold the

interrupt to identify the completion of the new data reception.

发送处理流程：

将要发送的的数据填入IICDT

发完产生中断.拉低SCl

中断处理程序判断状态。

有新的数据IICda等于新值。

接受流程：

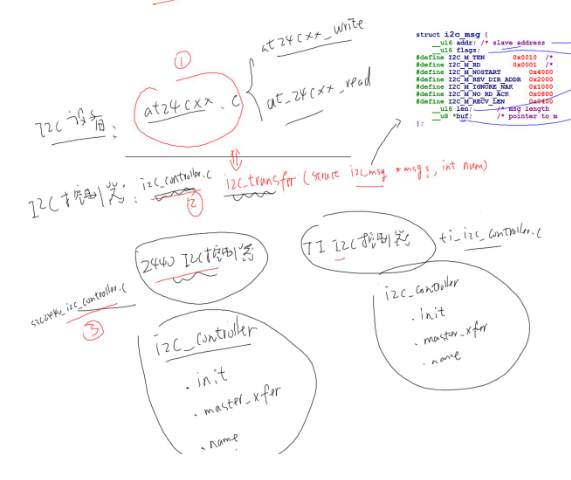
发起传输，接受data

产生中断，SCL拉低

中断处理程序，判断状态

读寄存器的值 value = iicdata ,读出后iic继续工作，继续读取数据

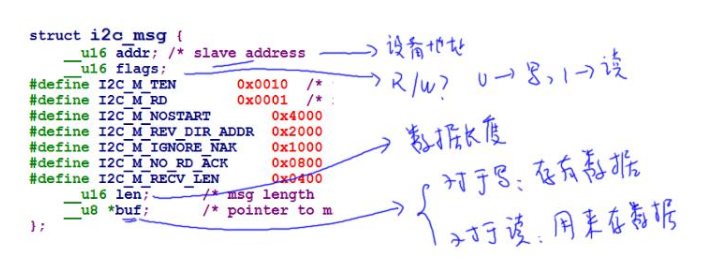
第003节\_程序框架：



最上层的我们供我们测试用的IIC\_test.c 调用at24cxx.c(erprom的型号)再去调用iic\_controler.c 去调用下层的具体的 2440\_controller.c或者别的iic控制器

上层的IIC 设备 控制下层的IIC控制器：

上层的IIIC设备参考内核的IIC\_transfer(struct i2c\_msg\*msg， int num);



Addr 要写的设备地址

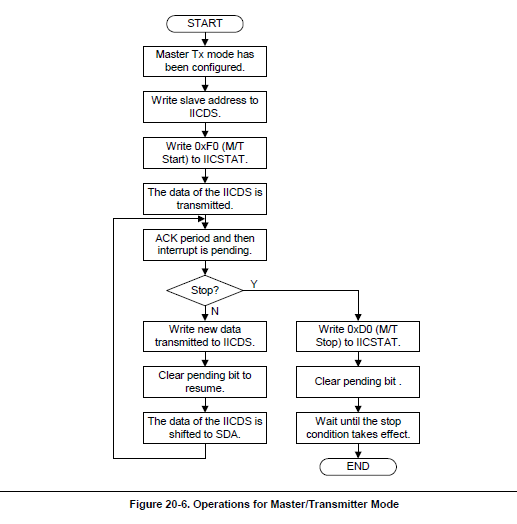
Flags 读/写

Buf 读写的数据

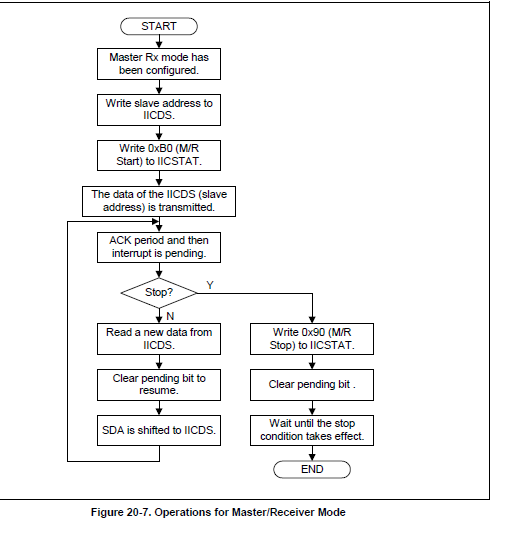
Led 数据的长度

第004节\_I2C控制器编程\_框架

Master 发送



Master 接受：



第005节\_I2C控制器编程\_中断

第006节\_EEPROM编程和测试代码

第007节\_测试

a. 中断没产生 : 未配置GPIO用于IIC功能

b. 只产生了一次中断, 并且出错 : tx err, no ack

解决方法: 启动传输之前 IICSTAT=(1<<4)

c. 第1次读OK，再次写卡死，复位再写仍卡死，重新上电再写OK:

读最后一个数据时，不要回应ACK给AT24CXX

课后作业:

1. 不使用s3c2440的I2C控制器，

把SCL、SDA对应的引脚设置为普通的GPIO功能，

使用GPIO实现I2C的操作