

《 关于STM32 I2C不合理的看法 》
// netjob 2008-08-22

1. 关于 SCL时钟设置的处理。

在软件模拟年代，SCL是50%的PWM，就是说 SCL低周期:SCL 高周期=1:1，这是我们一贯的认识与做法。

而实在ATMEL与NXP的 ARM 中的I2C也是如此对待。只不过ATMEL叫TWI.

由于ATMEL公司本身也生产I2C的FLASH和SPI FLASH；我们有理由相信这种1:1是可行可靠的。

我们知道I2C是NXP（菲利普半导体）的专利，它应该是权威了，但我们仔细看NXP的ARM7，它的I2C 时钟高、低周期 是可变的。我们可以设1:1，也可以设1:2 等等。虽然NXP 手册上是使用1:1的设计。

简单的说：SCL时钟高低周期比例定了下来（1:1），那么周期越长I2C的速度就越慢。

例如：

I2C的时钟是48MHZ, 要I2C工作200KHZ, 我们只要 把计算出的周期写入I2C寄存器就可以了。

方法：48,000,000/(200,000*2)=120;

也就是说把时钟分频【120】，I2C 就可以工作200KHZ了。

而ST公司的STM32名堂就来了，它来了什么 保准模式，快速模式，周期比例有什么 1:2、16:9。

还搞了个SCL时钟最大上升时间，它也分模式，STM32的I2C设置中就要输入这项内容。

而ST公司确实是把简单的问题复杂化了。

2. 关于状态位的查询与清除

这个事实上是很简单，但说起来比较麻烦。

在I2C协议中，我们都要根据标志位来判断分析I2C总线上发生的事情。例如START发送完毕了，数据发送完毕了等等。

ATMEL公司

ATMEL的TWI可以说 非常精简。

程序只要关心的标志：

TXRDY：发送保持寄存器就绪

RXRDY：接收保持寄存器就绪

TXCOMP：传输完成

NXP公司

NXP的I2C可以说 非常高效。它的每个状态标志位都有相应的清标志位，这样当I2C总线上发生的事情触发对应的标志置位后，我们可以人为的清除它。

而且它的标志位很统一，用一个【中断事件标志位】来表示，详细内容可以访问内容寄存器。

ATMEL公司和NXP公司的I2C都只有【一个】 状态寄存器。

而ST公司呢，STM32要操作【两个】状态寄存器，判断这两个寄存器的标志位。标志名目繁多，不同事件要查询不同标志位；也没有相应的清除标志位。这也是我们最担心的，我们不知到底这个标志是新的事件产生的还是老的标志没被清除。

虽然STM32告诉了我们每个标志的清除方法。

我的STM32初学者笔记有完整的初始化和读写函数代码，大家可以参考参考。