



串行通信接口SCI



实际开发时，经常会遇到这样的情况，例如做电机控制时需要显示ADC采样之后得到的电机电压、电流、转速等数据。当然首先想到的是可以为系统设计液晶屏来显示，不过还有一种不错的方法，就是将这些数据通过协议将其上传给计算机，然后通过计算机上的软件进行显示和监测。又例如在某些项目中需要计算机发送预先设定的指令来控DSP程序的运行方式。那F28335怎样才能和计算机之间实现数据的传输呢？最简单最常用的方法就是使用其内部的串行通信接口SCI。本章将详细介绍SCI的结构、特点及其工作原理。



SCI模块的概述

SCI是Serial Communication Interface的简称，即串行通信接口。SCI是一个双线的异步串口，换句话说，是具有接收和发送两根信号线的异步串口，一般可以看作是UART(通用异步接收/发送装置)。F28335的SCI支持与采用NRZ(non-return-to-zero)标准格式的异步外围设备之间进行数据通信。例如，设计时使用MAX3232芯片，将SCI设计成串口RS232，那么F28335就能够和其他使用RS232接口的设备进行通信。比如F28335内部的两个SCI之间，或者F28335的SCI同其他DSP的SCI之间均能实现通信。当然，F28335的SCI还可以设计成其他电平形式的串口，比如RS422、RS485等。



SCI模块的概述

F28335的内部具有三个相同的SCI模块，SCIA、SCIB和SCIC。每一个SCI模块都各有一个接收器和发送器，接收器用于实现数据的接收功能，发送器用于实现数据的发送功能。SCI的接收器和发送器各自都具有一个16级深度的FIFO队列，它们还都有自己独立的使能位和中断位，可以在半双工通信中进行独立的操作，或者在全双工通信中同时进行操作。

根据数据的传送方向，串行通信可以分为单工、半双工和全双工三种，如图15-1所示。单工是指设备A只能发送，而设备B只能接收。半双工是指设备A和B都能接收和发送，但是同一时间只能接收或者发送。全双工是指，在任意时刻，设备A和设备B都能同时接收或者发送。因为F28335的SCI具有能够独立使能和工作的接收器和发送器，所以其既可以工作于半双工方式，也可以工作于全双工方式。

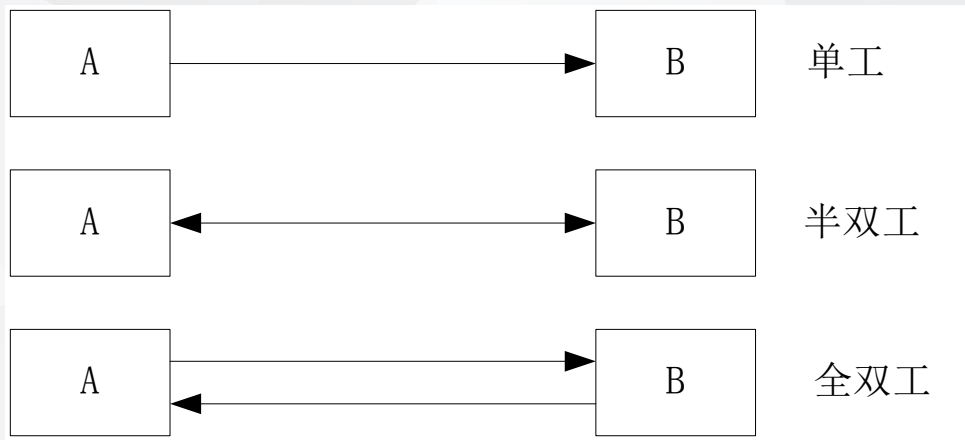


图15-1 串行通信的三种方式



SCI模块的概述·SCI模块的特点

由于三个SCI模块的功能是相同的，只是寄存器的命名有所不同，所以如果不做特殊说明，下面均以SCIA为例来进行讲解。SCIA与CPU的接口如图15-2所示。SCIA模块的特点如下：

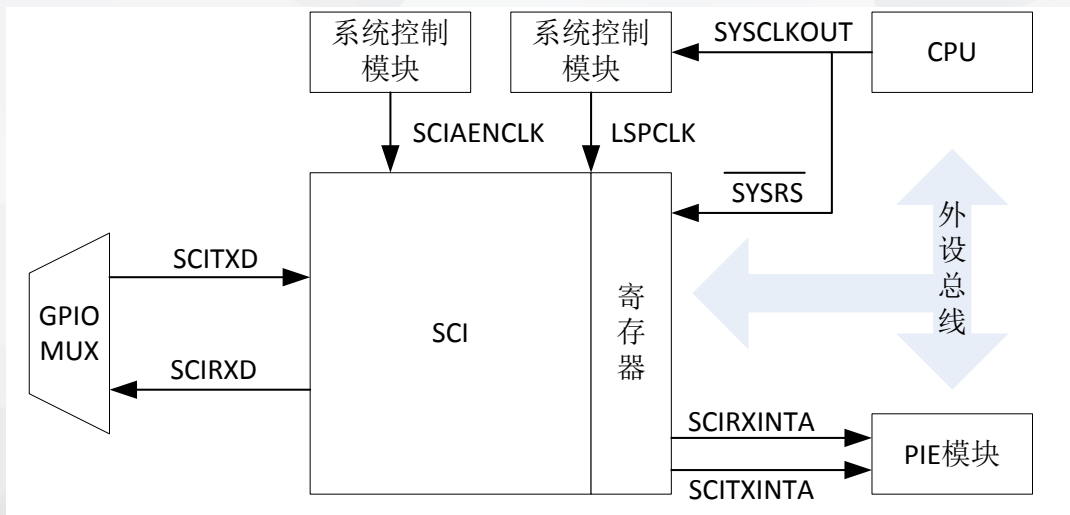


图15-2 SCIA与CPU的接口



SCI模块的概述·SCI模块的特点

- 1.从图15-2可以看到，SCI模块具有两个引脚，发送引脚SCITXD和接收引脚SCIRXD。SCITXD可以实现数据的发送，SCIRXD可以实现数据的接收。
- 2.外部晶振通过F28335的PLL模块倍频之后产生了CPU的系统时钟SYSCLKOUT，然后SYSCLKOUT经过低速时钟预定标器之后输出低速外设时钟LSPCLK提供给SCI模块。要保证SCI的正常运行，系统控制模块必须使能SCI时钟，只有使能了，LSPCLK才供给SCI，也就是在系统初始化函数中需要将外设时钟控制寄存器PCLKCR0的SCIAENCLK位置1。
- 3.SCI模块具有4种错误检测标志，分别是：极性错误(parity)、超时错误(overrun)、帧错误(framing)、间断(break)检测。



SCI模块的概述·SCI模块的特点

4.SCI模块具有双缓冲接收和发送功能，接收缓冲寄存器为SCIRXBUF，发送缓冲寄存器位SCITXBUF。独立的发送器和接收器使得SCI既能工作于半双工模式，也能工作于全双工模式。

5 .从图15-2可以看到，SCI模块可以产生两个中断，SCIRXINT和SCITXINT，即接收中断和发送中断。

6.SCI模块具有独立的发送中断使能位和独立的接收中断使能位。发送和接收可以通过中断方式实现，也可以查询中断方式实现。



SCI模块的概述·SCI模块的特点

7.在多处理器模式下，SCI模块具有两种唤醒方式：空闲线方式和地址位方式。平时在使用SCI的时候很少遇到多处理器的情况，通常就是2个处理器之间进行通信，这时候SCI通信采用空闲线方式。

8.SCIA模块具有13个寄存器，值得注意的是，与前面所学的ePWM、AD的寄存器不同，SCI的这些寄存器都是8位的。当某个寄存器被访问时，数据位于低8位，高8位为0，因此，如果将数据写入高8位将是无效的。

9. F28335的SCI还具有增强的16级深度的发送/接收FIFO以及自动通信速率检测的功能。



SCI模块的概述·SCI模块信号总结

从图15-2可以看到，SCI模块的信号有外部信号、控制信号和中断信号3种，如表15-1所示。

信号分类	信号名称	说明
外部信号	SCIRXD	SCI异步串口接收数据
	SCITXD	SCI异步串口发送数据
控制信号	LSPCLK	低速外设预定标时钟
中断信号	RXINT	SCI接收中断
	TXINT	SCI发送中断

表15-1 SCI模块的信号



SCI模块的工作原理

SCI模块能够工作于全双工模式，主要由于具有以下功能单元：

1. 1个发送器及其相关寄存器

SCITXBUF：发送数据缓冲寄存器，存放由CPU装载的需要发送的数据；

TXSHF：发送移位寄存器，从SCITXBUF寄存器接收数据，然后将数据逐位逐位移到SCITXD引脚上，每次移1位数据。



SCI模块的工作原理

SCI模块能够工作于全双工模式，主要由于具有以下功能单元：

2. 1个接收器及其相关寄存器

RXSHF：接收移位寄存器，从SCIRXD引脚移入数据，每次移1位数据。

SCIRXBUF：接收数据缓冲寄存器，存放CPU要读取的数据。从其他处理器传输过来的数据逐位逐位的移入寄存器RXSHF，当装满RXSHF的时候，将数据装入接收数据缓冲寄存器SCIRXBUF和接收仿真缓冲寄存器SCIRXEMU中。

3. 1个可编程的波特率发生器。

4. 数据存储器映射的控制和状态寄存器。



SCI模块的工作原理·SCI模块发送和接收数据的工作原理

SCI模块的工作原理如图15-3所示，图中的数字8表示8位数据并行传输。SCI模块具有独立的数据发送器和数据接收器，这样能够保证SCI既能同时进行，也能够独立进行发送和接收的操作。

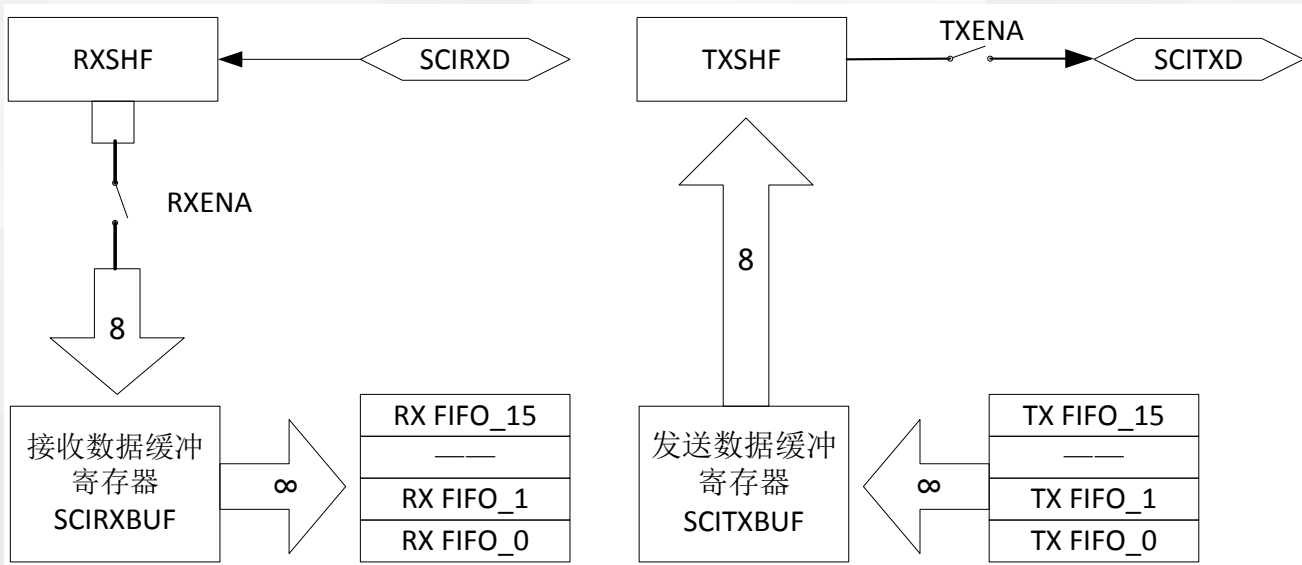


图15-3 SCI模块的工作原理



SCI模块的工作原理·SCI模块发送和接收数据的工作原理

SCI接收数据的过程如图15-3左半部分所示，主要如下：

- (1)当其他处理器发出的数据到达SCIRXD引脚后，SCI开始检测数据的起始位。
- (2)当SCIXD引脚检测到起始位，便开始将随后的数据逐位逐位的移至RXSHF寄存器。
- (3)如果SCI控制寄存器SCICTL1的位RXENA为1，也就是如果使能了SCI的接收操作，当RXSHF寄存器中的数据满后，便将这个8位的数据并行移入接收缓冲寄存器SCIRXBUF，接收缓冲寄存器就绪标志位RXRDY被置位，表示已经接受了一个新的数据，等待CPU来读取，此时还会产生一个SCI的接收中断申请信号。



SCI模块的工作原理·SCI模块发送和接收数据的工作原理

- (4) CPU通过程序读取SCIRXBUF寄存器中的数据后，RXRDY标志位被自动清除。至此，完成了一个数据的读取。
- (5) 如果SCI控制寄存器SCICTL1的位RXENA为0，也就是如果没有使能SCI的接收操作，从图15-3可以看到，当外部数据到达引脚SCIRXD时，数据还是会被逐位移入RXSHF寄存器，但是不会从RXSHF寄存器移入到SCIRXBUF寄存器中。
- (6) 如果使能了SCI的FIFO功能，则RXSHF会将数据直接加载到RX FIFO队列中，CPU再从FIFO队列读取数据，这样减少了CPU的开销，提高了效率。



SCI模块的工作原理·SCI模块发送和接收数据的工作原理

SCI发送数据的过程如图15-3有半部分所示，主要如下：

- (1) CPU通过程序将数据写入SCITXBUF寄存器，这时候发送器不再为空，发送缓冲寄存器就绪标志位TXRDY被清除。
- (2) 如果使能了SCI的FIFO功能，发送移位寄存器TXSHF将直接从TX FIFO队列中获取需要发送的数据。
- (3) SCI将数据从SCITXBUF发送到TXSHF寄存器，这时候，SCITXBUF寄存器为空，可以将下一个数据写入该寄存器了，发送缓冲寄存器就绪标志位TXRDY被置位，并发出发送中断请求信号。
- (4) 当数据移入发送移位寄存器TXSHF后，如果SCI控制寄存器SCICTL1的位TXENA为1，也就是如果使能了SCI的发送操作，则移位寄存器将数据逐位逐位的移到引脚TXRDY上。至此，完成一个数据的发送。



SCI模块的工作原理·SCI模块发送和接收数据的工作原理

仔细回味一下SCI发送和接收数据的原理，不难发现，发送和接收其实就是一个相反的过程。这里需要再提一下的就是当接收缓冲寄存器SCIRXBUF内有数据时，表示接收缓冲寄存器已经就绪，等待CPU来读取数据，其标志位RXRDY为高，当CPU将数据从SCIRXBUF读取后，RXRDY被清除，变为低。当发送缓冲寄存器为空时，表示发送缓冲寄存器就绪，等待CPU写入下一个需要发送的数据，其标志位TXRDY为高，当CPU将数据写入SCITXBUF后，TXRDY被清除，变为低。



SCI模块的工作原理·SCI通信的数据格式

处理器在通信的时候，一般都会涉及到协议，所谓协议就是指通信双方预先约定好的数据格式，以及每位数据所代表的具体含义。这就像地下党员做情报工作一样，地下工作人员将一份情报传给了上级，上级可以根据事先约定好的规则进行翻译，获取该份情报的具体内容。如果情报被敌人截获了也不怕，由于敌人不知道情报中每个文字所代表的含义，对于敌人来说，这份情报是无效的。这种事先约定好的规则，在通信中就叫做通信协议。



SCI模块的工作原理·SCI通信的数据格式

在SCI中，通信协议体现在了SCI的数据格式上。通常将SCI的数据格式称之为可编程的数据格式，原因是可以通过SCI的通信控制寄存器SCICCR来进行设置，规定通信过程中所使用的数据格式。

F28335的SCI模块使用的是NRZ数据格式，其包括了：

- (1)1个起始位；
- (2)1~8个数据位；
- (3)1个奇/偶/非极性位；
- (4)1~2个结束位；
- (5)在多处理器通信时的地址位模式下，有1个用于区别数据或者地址的特殊位。



SCI模块的工作原理·SCI通信的数据格式

从图15-4也能看出，SCI的数据帧包括1个起始位、1~8个数据位、1个可选的奇偶校验位和1或2个停止位。每个数据位占用8个SCI的时钟周期SCICLK，也就是LSPCLK，如图15-5所示。

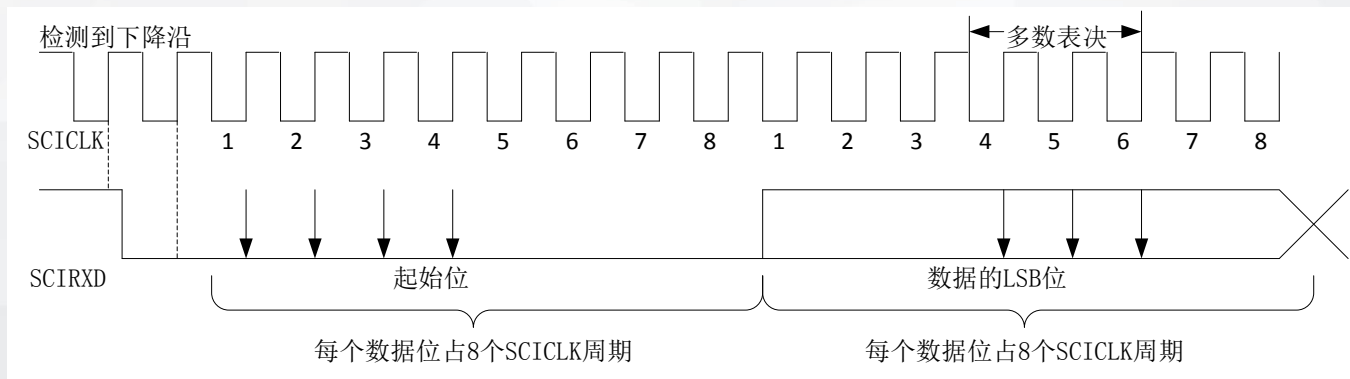


图15-5 SCI异步通信格式



SCI模块的工作原理·SCI通信的数据格式

SCI的接收器在收到一个起始位后开始工作，如图15-5所示，如果SCIRXD引脚检测到连续的4个SCICLK周期的低电平，SCI就认为接收到了一个有效的起始位，否则就需要寻找新的起始位。对于每个帧中起始位后面的数据位，CPU采用多数表决的机制来确定该位的值，具体的做法是，在每个数据位第4、第5、第6个SCICLK周期进行采样，如果3次采样中有2次以上的值相同，那么这个值就作为该数据位的值。



SCI模块的工作原理·SCI通信的波特率

过去寄一封信或者一个包裹通常需要一周甚至更久的时间，而现在使用快递的话有时候只需一到两天，说明运输和派送的速度大大加快了。SCI通信其实也是在运输物品，只不过这些物品是由1或0组成的数字信息，那么SCI是以什么样的速度去运输这些数据呢？这个速度是由SCI的波特率来决定的。所谓的波特率就是指设备每秒所能发送的二进制数据的位数。F28335的每个SCI模块都具有2个8位的波特率寄存器，SCIHBAUD和SCILBAUD，通过编程，可以实现达到64K种不同的速率。

SCI模块通信波特率与波特率选择寄存器之间的关系如式(15-1)所示：

$$BRR = \frac{LSPCLK}{SCI \text{ Asynchronous Baud} * 8} - 1 \quad (15-1)$$

其中，BRR为SCI波特率选择寄存器中的值，从十进制转换成16进制后，其高8位赋值给SCIHBAUD，低8位赋值给SCILBAUD。



SCI模块的工作原理·SCI通信的波特率

需要提醒的是，式(15-1)所示的波特率公式仅仅适用于 时的情况，当BRR=0时，SCI模块通信的波特率为：

$$\text{SCI Asynchronous Baud} = \frac{\text{LSPCLK}}{16} \quad (15-2)$$

下面进行举例说明。假设外部晶振的频率为30MHz，经过锁相环PLL倍频之后SYSCLKOUT为150MHz，然后，假设低速时钟预定标器LOSPCP的值为2，则SYSCLKOUT经过低速时钟预定标器之后产生低速外设时钟LSPCLK的值为37.5MHz，也就是说SCI模块的时钟为37.5MHz。如果需要SCI的波特率为19200，则将LSPCLK和波特率的数值代入式(15-1)，便可得到：BRR=243.15。由于寄存器的值都是正整数，所以忽略掉小数以后可以得到BRR=243。将243用十六进制数表示是0xF3，因此SCIHBAUD的值为0，SCILBAUD的值为0xF3。由于忽略了小数，将会产生0.06%的误差。



SCI模块的工作原理·SCI通信的波特率

当LSPCLK为37.5MHz时，对于SCI模块常见的波特率，其波特率选择寄存器的值如表15-2所示。

理想波特率	BRR(十进制)	SCIHBAUD	SCILBAUD	精确波特率	误差(%)
2400	1952	0x7A	0	2400	0
4800	976	0x3D	0	4798	-0.04
9600	487	0x01	0xE7	9606	-0.06
19200	243	0	0xF3	19211	0.06
38400	121	0	0x79	38422	0.06

在进行串口通信的时候，双方设备都必须以相同的数据格式和波特率进行通信，否则通信就会失败。例如F2812的SCI和计算机上的串口调试软件进行通信时，SCI采用了什么样的数据格式和波特率，那么串口调试软件也需要设定成相同的数据格式和波特率，反之也一样。这是SCI通信不成功时最简单，然而也是最容易忽视的一个问题。



SCI模块的工作原理·SCI模块的FIFO队列

F28335的SCI可以工作在标准SCI模式，也可以工作在增强的FIFO模式。当DSP上电复位时，SCI模块工作在标准SCI模式，此时FIFO功能是被禁止的，相应的，和FIFO功能相关的寄存器SCIFFTX、SCIFFRX和SCIFFCT都是无效的。

通过将SCI FIFO发送寄存器SCIFFTX的位SCIFFEN置1，使能FIFO模式。将SCIFFTX的位SCIRST置1，可以在任何状态下复位FIFO模式，SCI FIFO将重新开始发送和接收数据。



SCI模块的工作原理·SCI模块的FIFO队列

在标准SCI模式下，发送只有发送缓冲器SCITXBUF，接收也只有接收缓冲器SCIRXBUF。在FIFO模式下，发送缓冲器和接收缓冲器都是2个16级的FIFO队列，发送FIFO队列的寄存器是8位宽，而接收FIFO队列的寄存器是10位宽。以发送为例，在标准SCI模式下，8位的SCITXBUF作为发送FIFO和发送移位寄存器TXSHF间的缓冲器，当移位寄存器的最后一位被移出后，SCITXBUF才从FIFO加载CPU写好的需要发送的数据，而在FIFO模式下，SCITXBUF将不被使用，发送移位寄存器TXSHF将直接从FIFO加载需要发送的数据，而且加载数据的速度是可编程的。



SCI模块的工作原理·SCI模块的FIFO队列

通过SCI FIFO控制寄存器SCIFFCT的位FFTXDLY[7 : 0]可以确定TXSHF从FIFO加载数据的速度，或者说是加载数据的延时，就是隔多久加载一个数据。这种延时是以SCI模块波特率的时钟周期为基本单元的，8位的FFTXDLY可以定义最小延时0个波特率时钟周期到最大延时256个波特率时钟周期。如果将延时设定为最小延时，也就是0个波特率时钟周期，则SCI模块的FIFO加载数据没有延时，实现连续的发送数据。如果将延时设定为N个波特率时钟周期，则SCI模块发送完一个数据后，TXSHF将隔N个波特率时钟周期再从FIFO加载数据进行发送。这种可编程延时功能的好处在于可以协调和慢速设备之间的串行通信，同时也减少了CPU的干预。



SCI模块的工作原理·SCI模块的FIFO队列

发送和接收FIFO都有状态位TXFFST和RXFFST。TXFFST位于寄存器SCIFFTX[12:8]，共5位。RXFFST位于寄存器SCIFFRX[12:8]，共5位。这两位的作用是在任何时间可以标识FIFO队列中有效数据的个数。当TXFFST被清零时，发送FIFO队列的复位位TXFIFO RESET也被清零，发送FIFO的指针复位为0，可以通过将TXFIFO RESET置位来重新启动FIFO队列的发送操作。同样的，当RXFFST被清零时，接收FIFO队列的复位位RXFIFO RESET也被清零，接收FIFO的指针复位为0，可以通过将RXFIFO RESET置位来重新启动FIFO队列的接收操作。



SCI模块的工作原理·SCI模块的中断

图15-6是SCI中断标志和中断使能逻辑汇总。从图15-6可以看到，SCI模块可以产生两种中断，接收中断RXINT和发送中断TXINT。由于SCI可以工作在标准的SCI模式下，也可以工作在增强的FIFO模式下，无论工作于哪种模式，SCI都能产生接收中断和发送中断，但是不同的模式下，这两种中断信号产生的情况会有所不同，下面进行一一介绍。

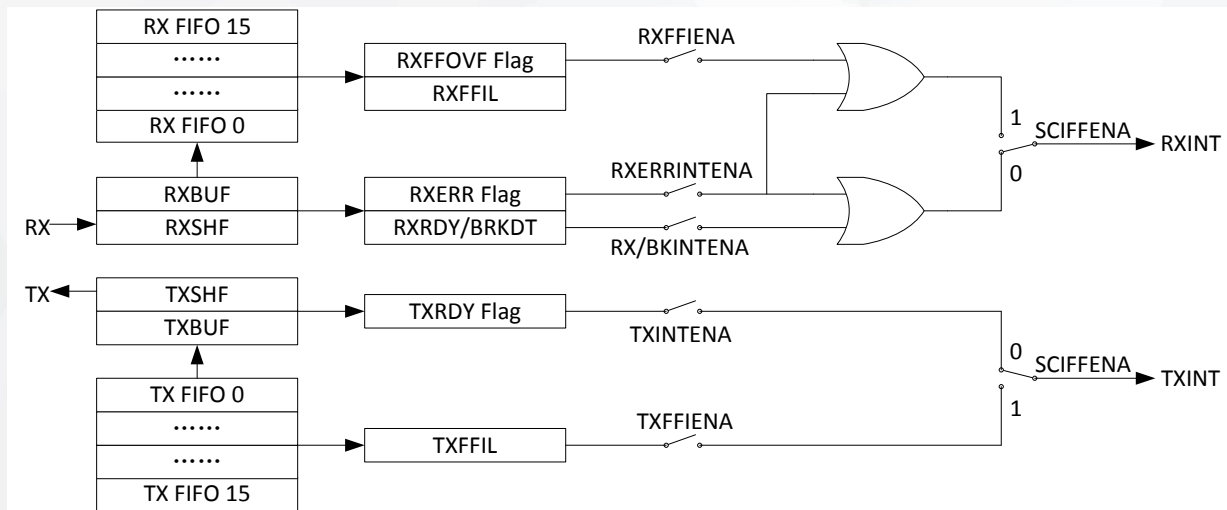


图15-6 SCI中断标志和中断使能逻辑



1.在标准SCI模式下

如图15-6所示，当SCIFFTX寄存器的SCIFFENA位为0时，也就是FIFO功能未使能时，SCI工作于标准SCI模式。对于接收操作，当RXSHF将接收到的数据写入SCIRXBUF，等待CPU来读取时，接收缓冲器就绪标志位RXRDY被置位，表示已经接收了一个数据，同时产生了一个接收中断RXINT的请求信号。如果SCI控制寄存器SCICTL2的位RX/BKINTENA为1，也就是接收中断已经使能，那么SCI将向PIE控制器提出中断请求。

通过接收中断的中断使能位RX/BKINTENA可以看出，其实RXINT是一个复用的中断。当SCI接收出现错误(RX ERROR)时，或者当SCI接收出现间断(RX BREAK)时，都会产生接收中断RXINT的请求信号。



SCI模块的工作原理·SCI模块的中断

当极性错误(parity)、超时错误(overrun)、帧错误(framing)、间断(break)检测这4种错误检测标志位中任何一个标志位被置1，SCI的接收错误标志RX ERROR就会被置1，同时产生一个接收中断RXINT的请求信号。如果SCI控制寄存器SCICTL1的位RX ERR INT ENA为1，也就是接收中断已经使能，那么SCI将向PIE控制器提出中断请求。

当SCI从丢失第一个停止位开始，如果SCIRXD引脚上连续的保持至少10位的低电平，则SCI认为接收产生了一次间断，此时SCI接收状态寄存器SCIRXST的位BRKDT被置位，即间断检测标志位被置位，同时产生一个接收中断RXINT的请求信号。如果SCI控制寄存器SCICTL2的位RX/BKINTENA为1，也就是接收中断已经使能，那么SCI将向PIE控制器提出中断请求。



SCI模块的工作原理·SCI模块的中断

从上面的介绍可以看到，SCI接收时，接收完一个数据、接收出现错误或者接收出现间断，都可以作为SCI接收中断RXINT的中断事件，如果使能相应的中断使能位，当这些事件发生时，都会产生一个RXINT的中断请求。不过，平时使用最多，接触最多的还是当SCI接收完一个数据后产生接收中断的情况。

对于发送操作，当发送缓冲寄存器SCITXBUF将数据写入发送移位寄存器TXSHF后，SCITXBUF为空，发送缓冲器就绪标志位TXRDY被置位，表示CPU可以将下一个需要发送的数据写到SCITXBUF中，同时产生了一个发送中断TXINT的请求信号。如果SCI控制寄存器SCICTL2的位TXINTENA为1，也就是发送中断已经使能，那么SCI将向PIE控制器提出中断请求。



2.在FIFO模式下

如图15-6所示，当SCIFFTX寄存器的SCIFFENA位为1时，也就是FIFO功能被使能时，SCI工作于增强的FIFO模式。对于接收操作，前面已经介绍过，接收FIFO队列有状态位RXFFST，表示接收FIFO中有多少个接收到的数据。同时，SCI FIFO接收寄存器SCIFFRX还有一个可编程的中断级位RXFFIL。当RXFFST的值与预设好的RXFFIL相等时，接收FIFO就会产生接收中断RXINT信号，如果SCIFFRX寄存器的位RXFFIENA为1，也就是FIFO接收中断已经使能，那么SCI将向PIE控制器提出中断请求。比如，假设通过编程，将RXFFIL位设置为8，那么当FIFO队列中接收到8个数据时，RXFFST的值也为8，正好和RXFFIL的值相等，这时候接收FIFO就产生了接收中断匹配事件。复位后，接收FIFO的中断触发级位RXFFIL默认值为0x1111，即16，也就是说FIFO队列中接收到16个数据的时候产生接收中断请求。



SCI模块的工作原理·SCI模块的中断

同工作于标准SCI模式类似，接收FIFO的接收中断RXINT也是复用的，当SCI接收出现错误(RX ERROR)时，也会产生接收中断RXINT的请求信号。前面已经详细介绍过，这里就不再赘述了。

对于发送操作，发送FIFO队列有状态位TXFFST，表示发送FIFO中有多少个数据需要发送。同时SCI FIFO发送寄存器SCIFFTX也有一个可编程的中断级位TXFFIL。当TXFFST的值与预设好的TXFFIL相等时，发送FIFO就会产生发送中断TXINT信号，如果SCIFFTX寄存器的位TXFFIENA为1，也就是FIFO发送中断已经使能，那么SCI将向PIE控制器提出中断请求。比如，假设通过编程，将TXFFIL位设置为8，那么当FIFO队列中还剩8个数据需要发送时，TXFFST的值也为8，正好和TXFFIL的值相等，这时候发送FIFO就产生了发送中断匹配事件。复位后，发送FIFO的中断触发级位TXFFIL默认的值为0x0000，即0，也就是说FIFO队列中数据全部发送完毕后产生发送中断请求。



SCI模块的工作原理·SCI模块的中断

综上所述，SCI的中断如表15-3所示。

工作模式	SCI中断源	中断标志位	中断使能位	SCIFFENA	中断线
标准SCI模式	接收完成	RXRDY	RX/BKINTENA	0	RXINT
	接收错误	RXERR	RXERRINTENA	0	RXINT
	接收间断	BRKDT	RX/BKINTENA	0	RXINT
	发送完成	TXRDY	TXINTENA	0	TXINT
FIFO模式	接收错误和接收间断	RXERR	RXERRINTENA	1	RXINT
	FIFO接收中断	RXFFIL	RXFFIENA	1	RXINT
	FIFO发送中断	TXFFIL	TXFFIENA	1	TXINT

表15-3 SCI的中断



SCI多处理器通信模式

多处理器通信，顾名思义，就是多个处理器之间进行数据通信。一个简单的多处理器通信拓扑示意图如图15-7所示。在图中，处理器A、B、C、D之间都可以实现通信，图中的实线表示处理器A和处理器B、C、D之间的通信。在同一个时刻，处理器A只能和处理器B、C、D之中的一个实现数据传输。当处理器A给处理器B、C、D中的某一个处理器发送数据时，A-B、A-C、A-D这3条通路上都会出现相同的数据，那如何来确保这些数据被正确的处理器接收呢？

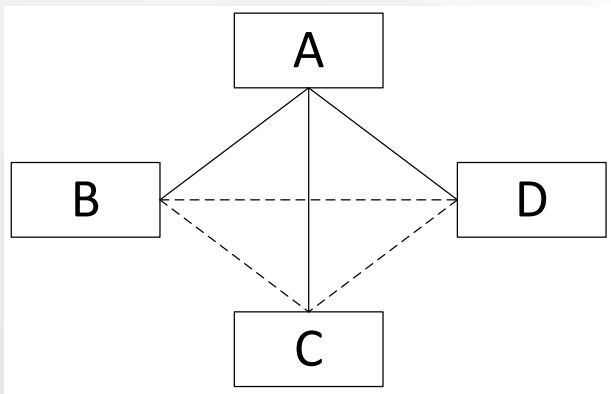


图15-7 一个简单的多处理器通信拓扑示意图



SCI多处理器通信模式

先来思考一下，比如寄了一封信给远方的朋友，那么邮递员是如何准确的将信投递到这位朋友家的邮箱的呢？原因是寄出的信封上清楚的写上了朋友家的地址，邮递员将实际地址和信封上的地址进行核对，两者相符时，就把信投进信箱了。根据这个原理，如果给处理器A、B、C、D都预先分配好地址，然后A发出去的信息里含有接收方的地址信息，接收处理器B或者C或者D，在接收到这个数据信息时，首先进行地址的核对，如果地址不符合，则不予响应；如果地址符合，则立即读取数据。这就是SCI多处理器通信的基本原理。

SCI在进行多处理器通信时，根据地址信息识别方法的不同，多处理器通信方式分为空闲线模式和地址位模式，下面分别进行说明。



SCI多处理器通信模式·地址位多处理器通信模式

图15-8为地址位多处理器通信模式示意图。当处理器A发出一连串数据信息时，将这串数据叫做数据块。数据块是由一个个帧构成的。前面讲过，帧就是带有格式信息的字符数据。从图15-8扩展后的数据格式可以看到，某一个数据块中的第一帧是地址信息，接下去的帧是数据信息，然后，在一些空闲周期之后，又有一个数据块，块中的第一帧也是地址信息，后面帧是数据信息。在块内，第一帧地址信息后面的一个位是1，代表此帧是地址信息，而第2帧数据信息后面的一个位是0，代表此帧是数据信息。这个位就叫做地址位，用于表示某个帧的数据是地址信息还是数据信息。像这样的，在通信格式中加入专门的地址位来判断帧是数据信息还是地址信息的方式叫做多处理器通信的地址位模式。



SCI多处理器通信模式·地址位多处理器通信模式

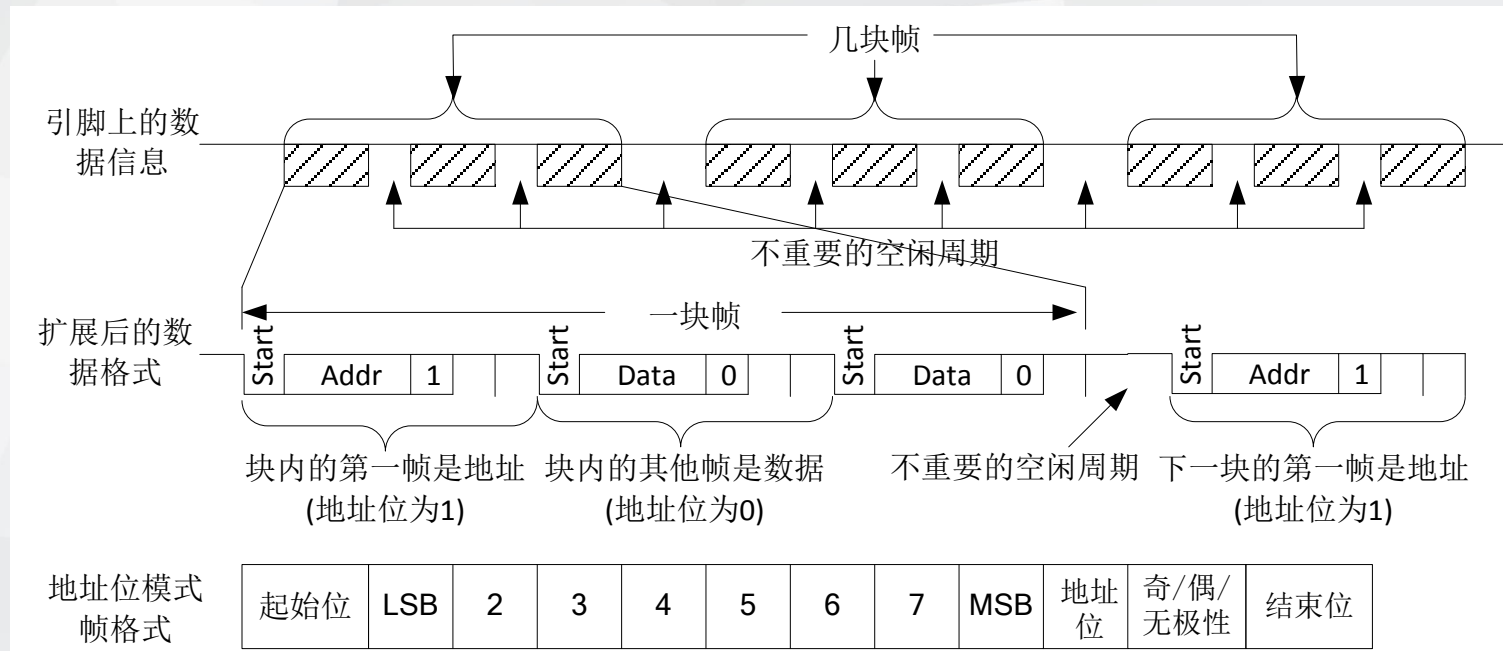


图15-8 地址位多处理器通信模式



SCI多处理器通信模式·空闲线多处理器通信模式

图15-9是空闲线多处理器通信模式示意图。在空闲线模式中，没有专门表示帧是数据或者地址的地址位。块与块之间有一段比较长的空闲周期，这段时间要明显长于块内帧与帧之间的空闲周期。如果某个帧之后有一段10位或者更长的空闲周期，那就表明新的数据块开始了。在某一个数据块中，第一帧代表地址信息，后面的帧代表数据信息。可见，在空闲线模式下，地址信息还是数据信息是通过帧与帧之间的空闲周期来判断的。当帧与帧之间的空闲周期超过10位的时候，就表示新的数据块开始了，而且块中的第一帧是地址信息。



SCI多处理器通信模式·空闲线多处理器通信模式

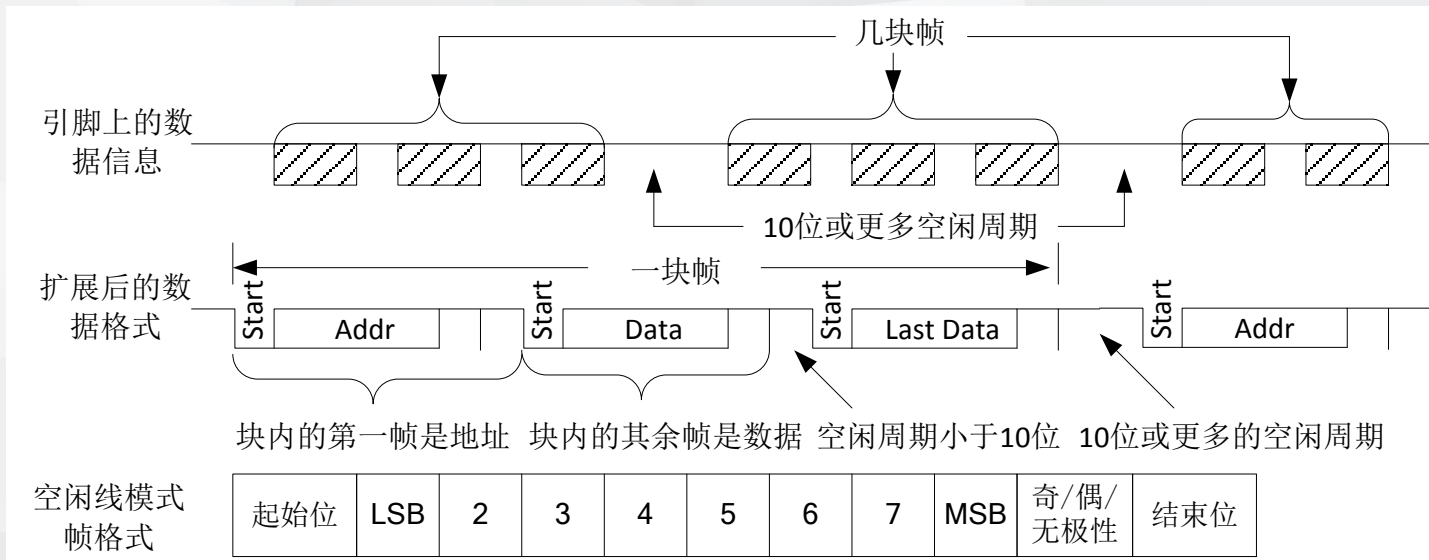


图15-9 空闲线多处理器通信模式



F28335内部的ADC模块·ADC的时钟频率和采样频率

空闲线模式中数据格式里没有提供额外的地址位，在处理10个字节以上的数据块时比地址位模式更为有效，被应用于典型的非多处理器SCI通信场合。而地址位模式由于有专门的位来识别地址信息，所以数据块之间不需要空闲周期等待，所以这种模式在处理一些小的数据块的时候更为有效，当然，当传输数据的速度比较快，而程序执行速度不够快时，很容易块与块之间产生10位以上的空闲，这样其优势就不明显了。平时接触比较多的还是双处理器之间的通信，因此这部分内容了解一下就行。



SCI模块的寄存器

SCI的功能都是可以通过软件进行配置的，可以通过对寄存器的设置来实现SCI通信格式的初始化，包括工作模式和协议、波特率、数据格式和中断使能等。SCI的寄存器如表15-4所示。SCI寄存器具体定义可见C2000助手。

寄存器名	地址范围			尺寸(*16)	说明
	SCIA	SCIB	SCIC		
SCICCR	0x7050	0x7750	0x7770	1	SCI通信控制寄存器
SCICTL1	0x7051	0x7751	0x7771	1	SCI控制寄存器1
SCIHBAUD	0x7052	0x7752	0x7772	1	SCI波特率寄存器高位
SCILBAUD	0x7053	0x7753	0x7773	1	SCI波特率寄存器低位
SCICTL2	0x7054	0x7754	0x7774	1	SCI控制寄存器2
SCIRXST	0x7055	0x7755	0x7775	1	SCI接收状态寄存器
SCIRXEMU	0x7056	0x7756	0x7776	1	SCI接收仿真数据缓冲寄存器
SCIRXBUF	0x7057	0x7757	0x7777	1	SCI接收数据缓冲寄存器
SCITXBUF	0x7059	0x7759	0x7779	1	SCI发送数据缓冲寄存器
SCIFFTX	0x705A	0x775A	0x777A	1	SCI FIFO发送寄存器
SCIFFRX	0x705B	0x775B	0x777B	1	SCI FIFO接收寄存器
SCIFFCT	0x705C	0x775C	0x777C	1	SCI FIFO控制寄存器
SCIPRI	0x705F	0x775F	0x777F	1	SCI优先级控制寄存器

表15-4 SCI寄存器



SCI发送和接收例程

SCI实现数据的接收或者发送可以采用查询的方式，也可以采用中断的方式，下面以查询方式为例进行介绍。HDSP-Super28335上的RS232-1和RS232-2是异步串行通信RS232的接口，是将SCI接口通过MAX3232芯片转换而来的。RS232-1的串口对应于SCIB，RS232-2的串口对应于SCIC。用串口线将RS232-1接口和计算机上的DB9串口连接起来。由于现在的计算机大多自身不带有串口，所以还需要配有USB转232的线，将计算机的USB口通过软件转变成RS232口。还需要注意的是，串口线有两种，一种是直通线，一种是交叉线，要根据硬件设计时采用的方式来定，HDSP-Super28335使用的是交叉线，如果使用的串口线和实际的硬件情况不符，通信将无法实现。



SCI发送和接收例程

这里SCI的程序主要实现的功能是DSP先给计算机上的串口调试软件发送一个字符串，然后串口调试助手向DSP发送用户输入的一串字符串，DSP接收到字符串后再发送给计算机。SCIB通信的数据格式设定为：波特率9600，起始位1位，数据位8位，无校验位，结束位1位。在配置串口调试软件的参数时，上述的所有参数都必须与SCIB设置的完全一致。

SCI无论采用查询方式还是中断方式来发送和接收数据，为了保证数据通信的准确性，必须遵守的原则是如果是接收数据，那么在接收新的数据之前需要将旧的数据读取，否则会产生数据丢失。如果是发送数据，那么必须等旧的数据发送完毕以后，才能发送新的数据，否则也会产生数据丢失。这也是平时编写程序时需要注意的地方。下面以查询方式为例介绍程序的编写。



SCI发送和接收例程

查询方式，就是通过查询发送缓冲器的就绪标志位TXRDY和接收缓冲器的就绪标志位RXRDY来判断SCI是否做好了发送准备或者接收准备。

通过前面的学习已经知道，当发送缓冲寄存器SCITXBUF将数据发送给发送移位寄存器TXSHF后，SCITXBUF为空，这时发送缓冲器的就绪标志位TXRDY被置1，意思是通知CPU可以发送新的数据了，因此，通过不断的查询，当TXRDY为1的时候，就可以发送新的数据了。

当接收移位寄存器RXSHF将接收到的字符发送给接收缓冲寄存器SCIRXBUF后，SCIRXBUF内有数据，这时接收缓冲器的就绪标志位RXRDY被置1，意思是通知CPU已经接收好了一个数据，让CPU赶紧来读取，因此，通过不断的查询，当RXRDY为1的时候，就可以去读取新的数据了。



SCI发送和接收例程

本章首先详细介绍了SCI模块的结构、特点及其工作原理，讲解了SCI通信时的数据格式、波特率的设置以及SCI的各种中断。然后，在标准SCI模式的基础上，介绍了SCI的FIFO功能，并简单介绍了SCI的多处理器通信的地址位和空闲线两种方式。最后以具体的例子来说明如何实现数据的发送和接收。下一章，将详细讲解F28335的串行外设接口SPI。