

# 第十一章 UART

## 10.1 概述

s3c2440A 通用异步接收器和发送器 (UART) 提供了三个独立的异步串行 I/O (SIO) 端口, 每个端口都可以在中断模式或 DMA 模式下操作。换言之, UART 可以生成一个中断或 DMA 请求用于 CPU 和 UART 之间的数据传输。UART 使用系统时钟可以支持最高 115.2K bps 的波特率。如果一个外部设备提供 UEXTCLK 给 UART, UART 可以在更高的速度下工作。每个 UART 通道对于接收器和发送器包括了 2 个 64 位的 FIFO。

s3c2440A UART 包括了可编程波特率, 红外传输接收, 一个或两个停止位, 5 位 6 位 7 位 8 位数据长度和奇偶校验。

每个 UART 包含一个波特率发送器, 发送器, 计数器和一个控制单元, 如图 11-1 所示。其波特率发生器可由 PCLK, FCLK/n 或 UEXTCLK (外部输入时钟) 来锁定。发送器和接收器包含了 64 位 FIFO 和数据移位器。数据写到 FIFO 然后在被传送前拷贝到发送移位器。数据通过发送数据引脚 (TxDn) 被发出。同时, 接收数据通过接收数据引脚 (RxDn) 移入, 然后从移位器拷贝到 FIFO。

## 10.2 特点

- 基于 DMA 或中断操作的 RxD0, TxD0, RxD1, TxD1, RxD2 和 TxD2
- 有红外和 64 位 FIFO 的 UART 通道 0 和 1
- 有 nRTS0, nCTS0, nRTS1 和 nCTS1 的 UART 通道 0 和 1
- 支持握手的发送和接收

## 10.3 模块图

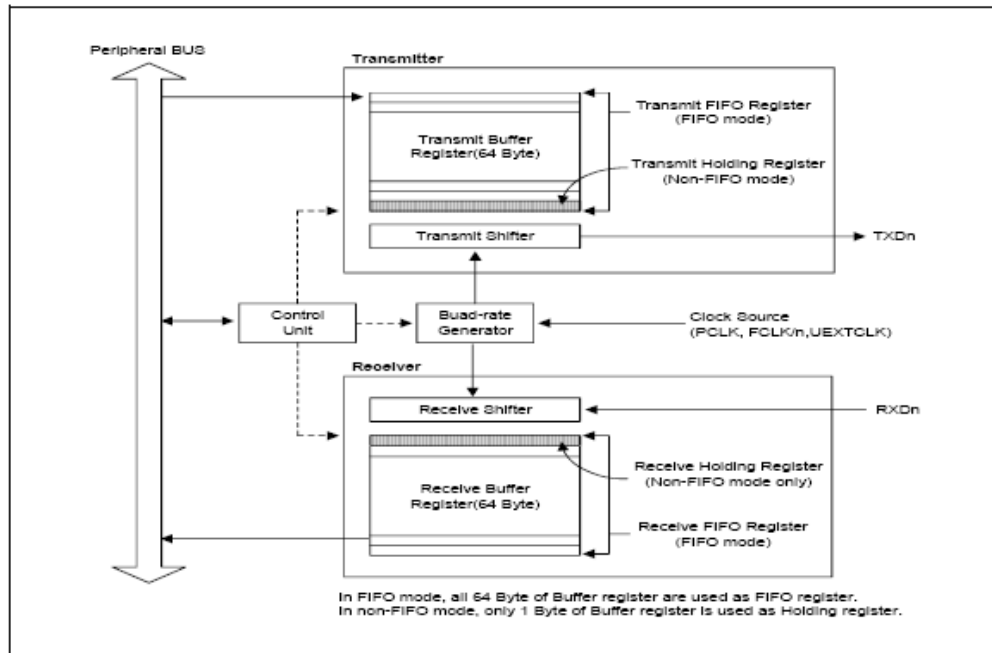


Figure 11-1 UART Block Diagram (with FIFO)

## 10.4 UART 操作

以下章节描述了 UART 操作，包括数据传输，数据接收，中断生成，波特率生，回送模式，红外模式自动流控制。

### 10.4.1 数据发送

发送的数据帧是可编程的。其包括一个开始位，5~8 个数据位，一个可选的奇偶校验位和 1~2 个停止位，其可由线性控制寄存器 ULCONn 来设置。发送器也可以产生一个终止条件，其可以对一个帧发送时间强制串行输出为逻辑 0。在当前发送字被完全传输完以后，该模块发送一个终止信号。在终止信号发送后，其串行发送数据到 Tx FIFO。

### 10.4.2 数据接收

如数据发送，接收的数据帧是可编程的。其包括一个开始位，5~8 个数据位，一个可选的奇偶校验位和 1~2 个停止位，其可由线性控制寄存器 ULCONn 来设置。接收器可以侦测溢出错误，奇偶校验错误，帧错误和终止条件，每个错误都可以设置一个错误标志。

- 溢出错误是指在旧数据被读取前新数据覆盖了旧数据。
- 奇偶校验错误是指接收器侦测到一个不希望的奇偶条件。
- 帧错误是指接收到的数据没有一个有效的停止位。
- 终止条件是指 Rx Dn 输入保持逻辑 0 状态长于一个帧的传输时间。

当接收器在三个字的时间（其间隔根据字长位的设置）没有收到任何数据且 Rx FIFO 在 FIFO 模式下不为空，接收超时条件出现。

### 10.4.3 自动流控制 (AFC)

s3c2440A 的 UART0 和 UART1 支持有 nRTS 和 nCTS 的自动流。这种情况下, 其可以连接到外部 UART。如果用户想连接 UART 到 Modem, 应使 UMCONn 寄存器中的自动流控制位无效且通过软件控制 nRTS。

在 AFC 中, nRTS 依赖于接收器的条件, nCTS 信号控制发送器的操作。UART 的发送器传输数据到 FIFO 仅当 nCTS 信号被激活 (在 AFC 中, nCTS 意思是对方的 UART FIFO 准备好接收数据)。在 UART 接收数据之前, 当其接收 FIFO 有大于 32 个字节的空闲空间, nRTS 必须被激活; 当其接收 FIFO 有小于 32 个字节的空闲空间, nRTS 必须置非激活。

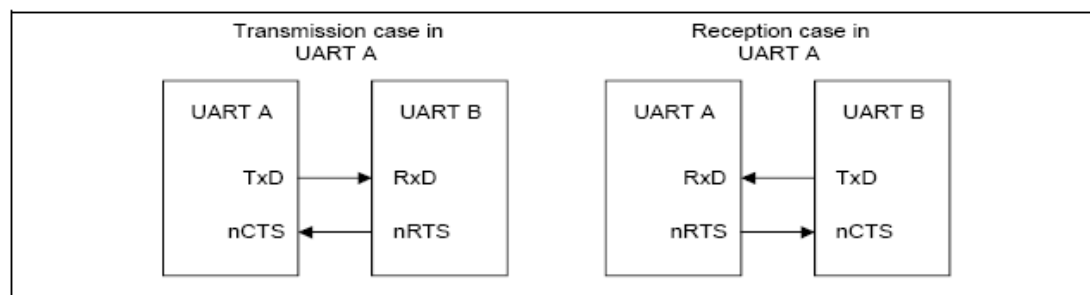


Figure 11-2 UART AFC interface

UART2 不支持 AFC 功能, 因为 s3c2440A 没有 nRTS2 和 nCTS2。

非自动流控制举例 (用软件控制 nRTS 和 nCTS)

使用 FIFO 的接收操作

- (1) 选择接收模式 (中断或 DMA 模式)
- (2) 检查 UFSTATn 寄存器中的 Rx FIFO 计数器。如果其值小于 32, 用户应该将 UMCONn[0] 的值置 1 (激活 nRTS), 如果其等于或大于 32, 用户应该将其置 0。
- (3) 重复步骤 2

有 FIFO 的发送操作

- (1) 选择发送模式 (中断或 DMA 模式)
- (2) 检查 UMSTATn[0] 的值, 如果其值为 1, 用户写数据到发送 FIFO 寄存器。

### 10.4.4 RS-232C 接口

如果用户想连接 UART 到 Modem 接口上, 就需要 nRTS、nCTS、nDSR、nDTR、DCD 和 nRI。在这种情况下, 用户可通过软件使用通用 IO 端口来控制这些信号, 因为 AFC 不支持 RS-232C 接口。

### 10.4.5 中断/DMA 请求生成

s3c2440A 的每个 UART 有七个状态 (Tx/Rx/Error) 信号: 溢出错误, 奇偶校验错误, 帧错误, 终止, 接收缓存数据准备好, 发送缓存空和发送移位器空, 其由相应 UART 的状态寄存器 (UTRSTATn/UERSTATn) 指出。

溢出错误, 奇偶校验错误, 帧错误, 终止条件是作为接收错误状态来参考, 每种错误都可以引起接收错误状态中断请求, 如果在控制寄存器 UCONn 中的接收错误状态中断使能位置 1。

当接收错误状态中断请求被检测到, 该信号引起的请求可以通过读 UERSTSTn 寄存器的值

来鉴别。

当接收器传输接收移位器上的数据到在FIFO模式下的接收FIFO寄存器上且接收到的数据数量达到接收FIFO触发水平，接收中断产生，如果在控制寄存器UCONn中接收模式被置1（中断请求或查询模式）。在非FIFO模式下，传输接收移位器的数据到接收保持寄存器将引起在中断请求或查询模式下的接收中断。

当发送器传输接收移位器上的数据到发送 FIFO 寄存器上且在在发送 FIFO 中剩余的数据数量达到发送 FIFO 触发水平，发送中断产生，如果在控制寄存器 UCONn 中的接收模式被选定为中断请求或查询模式。在非 FIFO 模式下，传输发送保持寄存器的数据到发送移位器将引起在中断请求或查询模式下的发送中断。

如果在控制寄存器中发送模式和接收模式被选定为 DMA<sub>n</sub> 请求模式，这样 DMA<sub>n</sub> 请求将取代上述情形下的接收或发送中断出现。

类型	FIFO 模式	非 FIFO 模式
Rx interrupt	只要接收数据到达接收 FIFO 的触发等级，中断产生。 当 FIFO 中的数据数量没有达到接收 FIFO 的触发等级且在三个字时间内没有接收到任何数据（接收超时），产生中断。该时间根据字长位设置。	只要接收缓存为空，由接收保持寄存器生成。
Tx interrupt	只要发送数据达到发送 FIFO 触发等级，中断产生。	只要发送缓存为空，由接收发送寄存器生成。
Error interrupt	当帧错误，奇偶错误或终止信号被检测到，中断产生。 当达到接收 FIFO 的上限没有读出数据，中断产生。	由错误生成。但是如果其他错误同时产生，仅一个中断产生。

10.4.6 UART 错误状态 FIFO

除了接收FIFO寄存器，UART有一个错误状态FIFO。错误状态FIFO是指出FIFO寄存器中有错的数据。当有错的数据准备读出，错误中断就产生。为了清除错误状态FIFO，有错误的URXH<sub>n</sub>和UERSTAT<sub>n</sub>必须被读出。

举例

假定UART接收FIFO顺序接收到A、B、C、D、E且在接收B时出现帧错误，在接收D时出现奇偶校验错误。因为错误接收的字符没有被读取，所以UART接收错误没有产生错误中断。一旦字符被读取，错误中断产生。

如图 11-3 所示UART接收 4 个字符包括 2 个错误。

时间	顺序流	错误中断	注
#0	当没有字符被读出	-	-
#1	接收到A、B、C、D、E	帧错误（B中）中断出现	‘B’必须被读出
#2	在A被读出后	-	-
#3	在 B 被读出后	-	-
#4	在 C 被读出后	奇偶校验错误（D中）中断出现	‘D’必须被读出
#5	在 D 被读出后	-	-
#6	在 E 被读出后	-	-

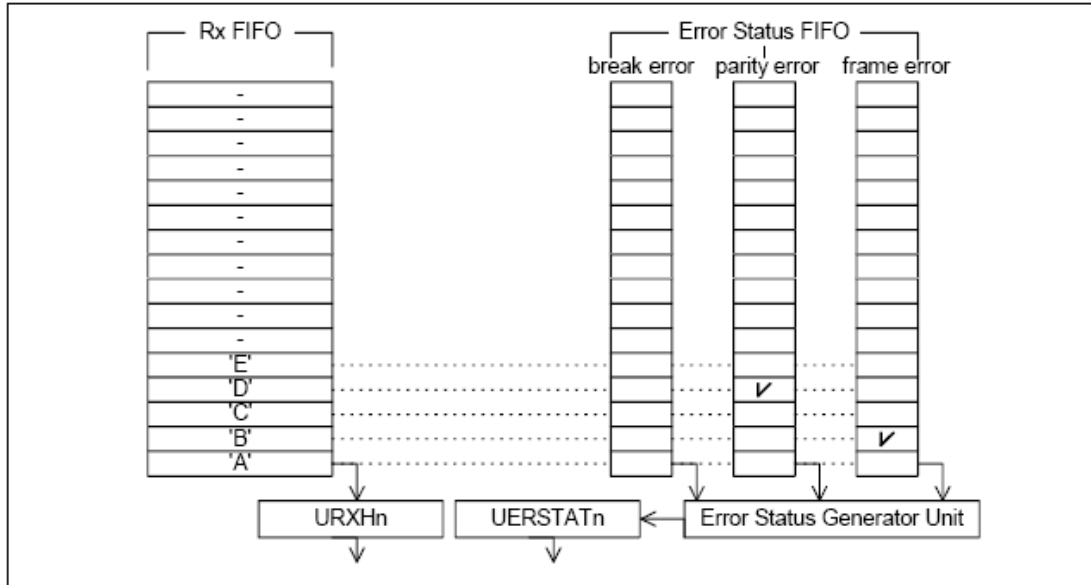


Figure 11-3. Example showing UART Receiving 5 Characters with 2 Errors

### 10.4.7 波特率产生

每个UART波特率发生器为发送器和接收器提供串行时钟。波特率发生器的源时钟可以选择s3c2440A的内部系统时钟或UEXTCLK。换言之，被除数可以设定UCONn中的时钟选择来选择。波特率时钟通过 16 和由UART波特率除数寄存器（UBRDIVn）中定义的 16 位除数来分频源时钟（PCLK, FCLK/n or UEXTCLK）产生。UBRDIVn由以下表达式来决定。

$$\text{UBRDIVn} = (\text{int}) (\text{UART clock} / (\text{baud rate} \times 16)) - 1$$

(UART clock: PCLK, FCLK/n or UEXTCLK)

UBRDIVn应该是从 1 到  $2^{16}-1$ ，当使用小于PCLK的外部时钟UEXTCLK时应该设置 0。

例如如果波特率是 115200bps且UART时钟是 40MHz, UBRDIVn是

$$\begin{aligned} \text{UBRDIVn} &= (\text{int})(40000000 / (115200 \times 16)) - 1 \\ &= (\text{int})(21.7) - 1 \text{ [round to the nearest whole number]} \\ &= 22 - 1 = 21 \end{aligned}$$

### 波特率错误公差 Baud-Rate Error Tolerance

UART帧错误应该少于 1.87%(3/160).

$$t_{\text{UPCLK}} = (\text{UBRDIVn} + 1) \times 16 \times 1\text{Frame} / \text{PCLK} \quad t_{\text{UPCLK}} : \text{Real UART Clock}$$

$$t_{\text{UEXACT}} = 1\text{Frame} / \text{baud-rate} \quad t_{\text{UEXACT}} : \text{Ideal UART Clock}$$

$$\text{UART error} = (t_{\text{UPCLK}} - t_{\text{UEXACT}}) / t_{\text{UEXACT}} \times 100\%$$

注:

1. 1Frame = start bit + data bit + parity bit + stop bit.
2. 在特定条件下，我们支持波特率最高达到 921.6K bps。例如，当PCLK为 60MHz, 你可以在 1.69UART错误公差下使用 921.6K bps under UART error of 1.69%.

### 10.4.8 回送模式

s3c2440A UART提供了一个参考作为回送模式的测试模式，以帮助孤立通讯连接中的错误。这个模式结构上使能在UART中的RXD和TXD连接。在此模式下发送的数据通过RXD接收到接收器。该特点是允许处理器验证内部传输和每个SIO通道的接收数据通道。该模式通过设置UART控制寄存器UCONn的回送位来选定。

### 10.4.9 红外模式

S3c2440A UART模块支持红外发送和接收，其通过设置UART线性控制寄存器ULCONn的红外模式位来选定。如图 11-4 所示如何执行一个红外模式。

在红外发送模式，发送脉冲是  $3/16$  的正常串行发送率（当发送数据位为 0 时）。在红外接收模式下，接收器必须检测  $3/16$  的脉冲期间来识别一个 0 值（看如图 11-7 和 11-6 所示的帧时序图）。

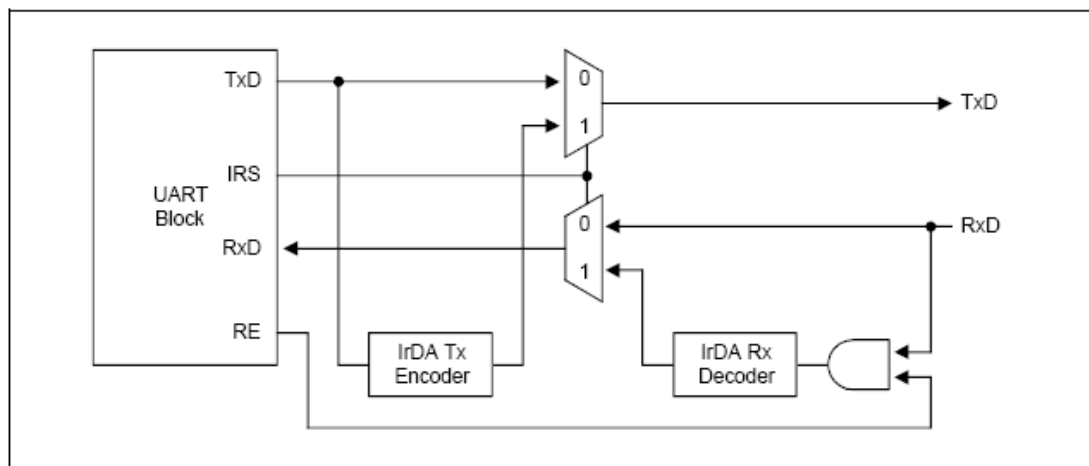


Figure 11-3. IrDA Function Block Diagram

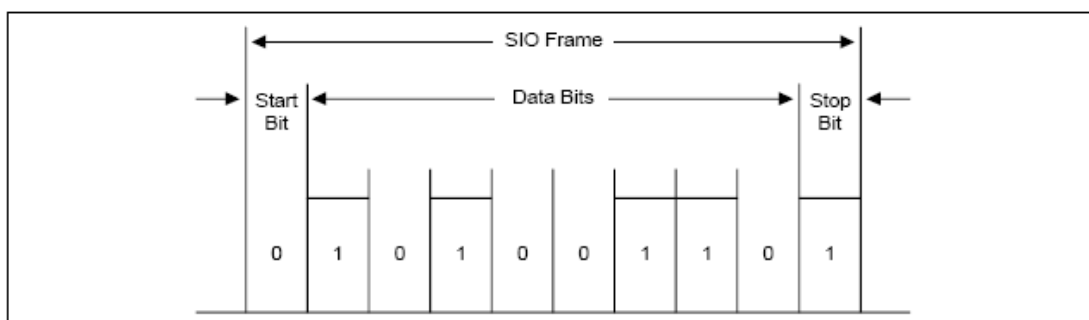


Figure 11-4. Serial I/O Frame Timing Diagram (Normal UART)

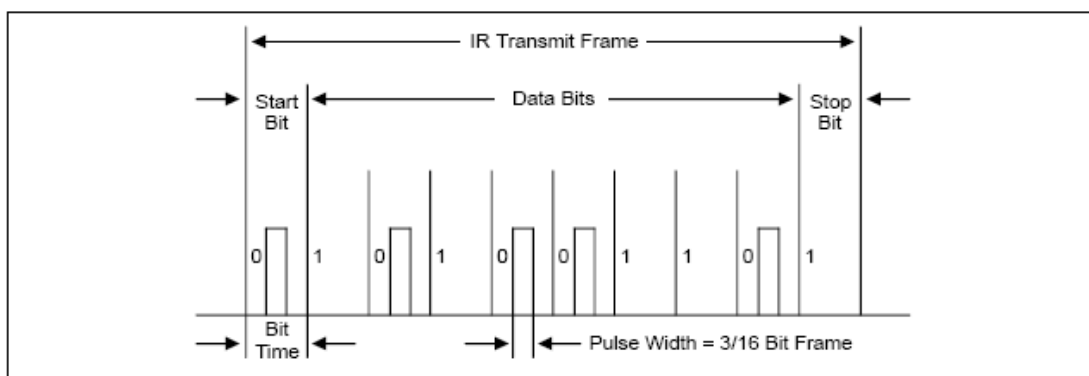


Figure 11-5. Infrared Transmit Mode Frame Timing Diagram

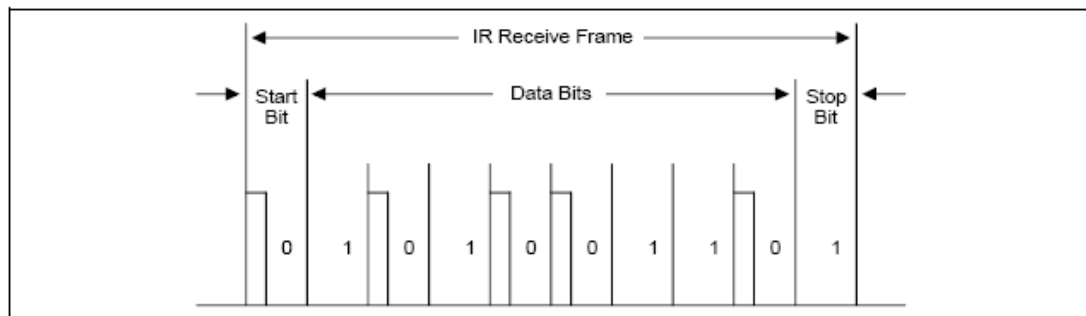


Figure 11-6. Infrared Receive Mode Frame Timing Diagram

## 10.5 UART 特殊寄存器

- (1) UART 线性控制寄存器 (ULCONn)
- (2) UART 控制寄存器 (UCONn)
- (3) UART FIFO 控制寄存器(UFCONn)
- (4) UART MODEM 控制寄存器(UMCONn)
- (5) UART 接收发送状态寄存器(UTRSTATn)
- (6) UART 错误状态寄存器(UERSTATn)
- (7) UART FIFO 状态寄存器(UFSTATn)
- (8) UART MODEM 状态寄存器(UMSTATn)
- (9) UART 发送缓存寄存器(UTXHn)
- (10) UART 接收缓存寄存器(URXHn)
- (11) UART 波特率除数寄存器 (UBRDIVn)

### 10.5.1 UART 线性控制寄存器

UART LINE CONTROL REGISTER (ULCONn)

寄存器	地址	读写	描述	复位值
ULCON0	0x50000000	R/W	UART通道 0 线性控制寄存器	0x00
ULCON1	0x50004000	R/W	UART通道 1 线性控制寄存器	0x00
ULCON2	0x50008000	R/W	UART通道 2 线性控制寄存器	0x00

ULCONn	位	描述	初始值
Reserved	[7]		0
Infrared Mode	[6]	决定是否使用红外模式 0 = 正常模式操作 1 = 红外接收发送模式	0
Parity Mode	[5:3]	在UART发送接收操作中定义奇偶码的生成和检验类型 0xx = No parity 100 = Odd parity 101 = Even parity 110 = Parity forced/checked as 1 111 = Parity forced/checked as 0	000
Number of Stop Bit	[2]	定义度搜按个停止位用于帧末信号 0 = 每帧一个停止位 1 = 每帧两个停止位	0
Word Length	[1:0]	指出发送接收每帧的数据位数 00 = 5-bits 01 = 6-bits 10 = 7-bits 11 = 8-bits	00



## 10.5.2 UART 控制寄存器

### UART CONTROL REGISTER (UCONn)

寄存器	地址	读写	描述	复位值
UCON0	0x50000004	R/W	UART通道 0 控制寄存器	0x00
UCON1	0x50004004	R/W	UART通道 1 控制寄存器	0x00
UCON2	0x50008004	R/W	UART通道 2 控制寄存器	0x00

UCONn	位	描述	初始值
FCLK divider	[15:12]	<p>当UART时钟源选为FCLK/n时的分频器值。 n由UCON0[15:12], UCON1[15:12], UCON2[14:12]来决定。</p> <p><b>UCON2[15]是FCLK/n 时钟使能位。</b> 设置n从 7 到 21, 使用UCON0[15:12], 设置n从 22 到 36, 使用UCON1[15:12], 设置n从 37 到 43, 使用UCON2[14:12],</p> <p><b>UCON2[15]:</b> 0 = 无效FCLK/n时钟    1 = 使能FCLK/n时钟 对于UCON0, UART时钟= FCLK / (divider+6), 其 divider&gt;0, UCON1, UCON2 必须为 0。 ex) 1: UART clock = FCLK/7, 2: UART clock = FCLK/8 3: UART clock = FCLK/9, ... , 15: UART clock = FCLK/21 对于UCON1, UART时钟= FCLK / (divider+21), 其 divider&gt;0, UCON0, UCON2 必须为 0 ex) 1: UART clock = FCLK/22, 2: UART clock = FCLK/23 3: UART clock = FCLK/24, ... , 15: UART clock = FCLK/36 对于UCON2, UART时钟= FCLK / (divider+36), 其 divider&gt;0, UCON0, UCON1 必须为 0。 ex) 1: UART clock = FCLK/37, 2: UART clock = FCLK/38 3: UART clock = FCLK/39, ... , 7: UART clock = FCLK/43 如果UCON0/1[15:12]和UCON2[14:12] 都是 0, 分频器为 44, 则UART时钟= FCLK/44 总的除数范围是从 7 到 44。</p>	0000
Clock Selection	[11:10]	<p>为波特率选择PCLK, UEXTCLK或FCLK/n。 <math>UBRDIVn = (int)(selected\ clock / (baudrate \times 16)) - 1</math> 00, 10 = PCLK, 01 = UEXTCLK, 11 = FCLK/n (如果你选择FCLK/n,你应该在选择或取消选择FCLK/n加上 注释代码)</p>	0
Tx Interrupt Type	[9]	<p>发送中断请求类型 0 = Pulse (在非FIFO模式下一旦发送缓存变空或在FIFO模式 下发送缓存达到发送FIFO触发水平, 则中断请求) 1 = Level (当非FIFO模式下发送缓存变空或在FIFO模式下发 送缓存达到发送FIFO触发水平, 则中断请求)</p>	0
Rx Interrupt Type	[8]	<p>接收中断请求类型 0 = Pulse (在非FIFO模式下一旦接收缓存接收到数据, 或在</p>	0

		FIFO模式下, 发送缓存达到发送FIFO触发水平, 则中断请求) 1 = Level (在非FIFO模式下接收缓存接收数据, 或在FIFO模式下发送缓存达到发送FIFO触发水平, 则中断请求)	
Rx Time Out Enable	[7]	在UART FIFO有效时, 使能接收超时中断。该中断是一个接收中断。 0 = 无效 1 = 有效	0
Rx Error Status Interrupt Enable	[6]	使能UART对异常产生中断, 例如在接收期间终止信号, 帧错误, 奇偶校验错误和溢出错误。 0 = 不产生接收错误状态中断 1 = 产生接收错误状态中断。	0
Loopback Mode	[5]	对该位置位将引起UART进入回送模式。该模式仅用于测试目的。 0 = 正常操作 1 = 回送模式	0
Send Break Signal	[4]	对该位置位将引起UART在一个帧时间内发送一个终止信号。在发送终止信号后该位自动清零。 0 = 正常发送 1 = 发终止信号	0
Transmit Mode	[3:2]	决定哪种功能用来写数据到UART发送缓存寄存器 (UART Tx Enable/Disable)。 00 = 无效 01 = 中断请求或查询模式 10 = DMA0 请求 (仅对UART0), DMA3 请求 (仅对UART2) 11 = DMA1 请求 (仅对UART1)	00
Receive Mode	[1:0]	决定哪种功能用来读取UART接收缓存寄存器中的数据 (UART Rx Enable/Disable)。 00 = 无效 01 = 中断请求或查询模式 10 = DMA0 请求 (Only for UART0), DMA3 请求 (Only for UART2) 11 = DMA1 请求 (Only for UART1)	00

### 10.5.3 UART FIFO 控制寄存器

#### UART FIFO CONTROL REGISTER (UFCONn)

寄存器	地址	读写	描述	复位值
UFCON0	0x50000008	R/W	UART通道 0FIFO控制寄存器	0x00
UFCON1	0x50004008	R/W	UART通道 1FIFO控制寄存器	0x00
UFCON2	0x50008008	R/W	UART通道 2FIFO控制寄存器	0x00

UFCONn	位	描述	初始值
Tx FIFO Trigger Level	[7:6]	决定发送FIFO的触发等级 00 = Empty 01 = 16-byte 10 = 32-byte 11 = 48-byte	00
Rx FIFO Trigger Level	[5:4]	决定接收FIFO的触发等级 00 = 1-byte 01 = 8-byte 10 = 16-byte 11 = 32-byte	00
Reserved	[3]		0
Tx FIFO Reset	[2]	在重置FIFO后自动清除 0 = Normal 1= Tx FIFO reset	0
Rx FIFO Reset	[1]	在重置FIFO后自动清除 0 = Normal 1= Rx FIFO reset	0
FIFO Enable	[0]	0 = Disable 1 = Enable	0

## 10.5.4 UART MODEM 控制寄存器

### UART MODEM CONTROL REGISTER (UMCONn)

寄存器	地址	读写	描述	复位值
UMCON0	0x5000000C	R/W	UART通道 0MODEM控制寄存器	0x0
UMCON1	0x5000400C	R/W	UART通道 1MODEM控制寄存器	0x0
保留	0x5000800C	-	-	-

UMCONn	位	描述	初始值
Reserved	[7:5]	这些位必须为 0	000
Auto Flow Control (AFC)	[4]	0 = 无效 1 = 有效	0
Reserved	[3:1]	这些位必须为 0	000
Request to Send	[0]	如果AFC位有效, 则该值被忽略。在这种情况下 S3C2440A将自动控制nRTS。 如果AFC位无效, nRTS必须由软件控制。 0 = 'H' level (去激活nRTS) 1 = 'L' level (激活nRTS)	0

## 10.5.5 UART 接收发送状态寄存器

### UART TX/RX STATUS REGISTER (UTRSTATn)

寄存器	地址	读写	描述	复位值
UTRSTAT0	0x50000010	R/W	UART通道 0 接收发送状态寄存器	0x00
UTRSTAT1	0x50004010	R/W	UART通道 1 接收发送状态寄存器	0x00
UTRSTAT2	0x50008010	R/W	UART通道 2 接收发送状态寄存器	0x00

UTRSTATn	位	描述	初始值
Transmitter empty	[2]	当发送缓存寄存器中没有有效值且发送移位寄存器空, 则自动置 1。 0 = 非空 1 = 发送器空(发送缓存和移位寄存器)	1
Transmit buffer empty	[1]	当发送缓存寄存器为空, 则自动置于 1 0 = 发送缓存寄存器不为空 1 = 发送缓存寄存器 为空 (在非FIFO模式下, 中断或DMA被请求。在FIFO模式下当发送FIFO触发等级设为 00 (空) 时, 中断或DMA被请求) 如果UART使用FIFO, 用户应该检查寄存器UFSTAT中的 Tx FIFO Coun位和Tx FIFO Full 位取代对此位的检查。	1
Receive buffer data ready	[0]	只要接收缓存寄存器保留通过RXDn端口接收的有效值, 则自动置 1。 0 = 缓存寄存器为空 1 = 缓存寄存器接收到数据 (在非FIFO模式下, 请求中断或DMA) 如果UART使用FIFO, 用户应该在UFSTAT中的Rx FIFO Count位和Rx FIFO Full位取代对此位的检查。	0

## 10.5.6 UART 错误状态寄存器

### UART ERROR STATUS REGISTER (UERSTATn)

寄存器	地址	读写	描述	复位值
UERSTAT0	0x50000014	R/W	UART通道 0 错误状态寄存器	0x00
UERSTAT1	0x50004014	R/W	UART通道 1 错误状态寄存器	0x00
UERSTAT2	0x50008014	R/W	UART通道 2 错误状态寄存器	0x00

UTRSTATn	位	描述	初始值
Break Detect	[3]	自动置 1 来指出一个终止信号已发出 0 = No break receive 1 = Break receive (已请求中断)	0
Frame Error	[2]	只要在接收操作中帧错误出现则自动置 1 0 = 接收过程中无帧错误 1 = 帧错误(已请求中断)	0
Parity Error	[1]	只要在接收操作中出现奇偶校验错误则自动置 1 0 = 接受过程中无奇偶校验错误 1 = 奇偶校验错误 (已请求中断)	0
Overrun Error	[0]	只要在接收过程中出现溢出错误则自动置 1 0 = 接收过程中无溢出错误 1 = 溢出错误 (已请求中断)	0

## 10.5.7 UART FIFO 状态寄存器

### UART FIFO STATUS REGISTER (UFSTATn)

寄存器	地址	读写	描述	复位值
UFSTAT0	0x50000018	R/W	UART通道 0FIFO状态寄存器	0x00
UFSTAT1	0x50004018	R/W	UART通道 1FIFO状态寄存器	0x00
UFSTAT2	0x50008018	R/W	UART通道 2FIFO状态寄存器	0x00

UFSTATn	位	描述	初始值
Reserved	[15]		0
Tx FIFO Full	[14]	只要在接收操作中接收FIFO满, 则自动置 1。 0 = 0-byte ≤ Tx FIFO data ≤ 63-byte 1 = Full	0
Tx FIFO Count	[13:8]	发送FIFO中的数据数量	0
Reserved	[7]		0
Rx FIFO Full	[6]	只要在接收操作中接收FIFO满, 则自动置 1。 0 = 0-byte ≤ Rx FIFO data ≤ 63-byte 1 = Full	0
Rx FIFO Count	[5:0]	接收FIFO中的数据数量	0

## 10.5.8 UART MODEM 状态寄存器

### UART MODEM STATUS REGISTER (UMSTATn)

寄存器	地址	读写	描述	复位值
UFSTAT0	0x5000001C	R/W	UART通道 0MODEM状态寄存器	0x00
UFSTAT1	0x5000401C	R/W	UART通道 1MODEM状态寄存器	0x00
保留	0x5000801C	R/W	保留	-

UMSTATn	位	描述	初始值
Delta CTS	[4]	Indicate that the nCTS input to the S3C2440A has changed state since the last time it was read by CPU. (参考图 11-8.) 0 = 未变 1 = 已变	0
Reserved	[3:1]		0
Clear to Send	[0]	0 = CTS信号未激活 (nCTS引脚为高电平) 1 = CTS信号激活 (nCTS引脚为低电平)	0

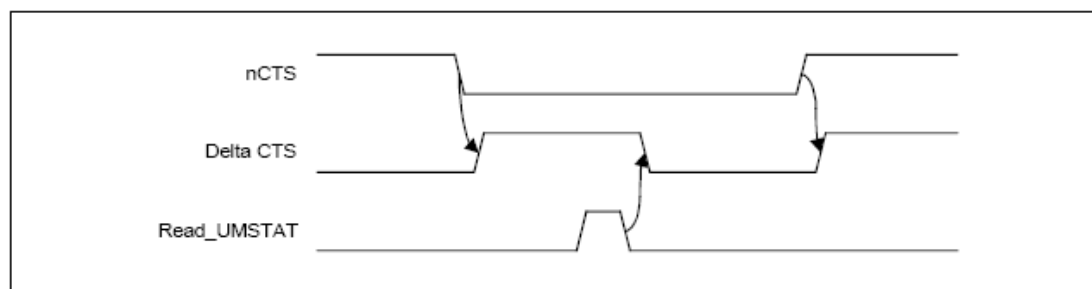


Figure 11-7. nCTS and Delta CTS Timing Diagram

## 10.5.9 UART 发送缓存寄存器

### UART TRANSMIT BUFFER REGISTER (HOLDING REGISTER & FIFO REGISTER)

寄存器	地址	读写	描述	复位值
UTXH0	0x50000020(L) 0x50000023(B)	W (byte)	UART通道 0 发送缓存寄存器	-
UTXH1	0x50004020(L) 0x50004023(B)	W (byte)	UART通道 1 发送缓存寄存器	-
UTXH2	0x50008020(L) 0x50008023(B)	W (byte)	UART通道 2 发送缓存寄存器	-

UTXHn	位	描述	初始值
TXDATAn	[7:0]	UARTn的发送数据	-

### 10.5.10 UART 接收缓存寄存器

#### UART RECEIVE BUFFER REGISTER (HOLDING REGISTER & FIFO REGISTER)

寄存器	地址	读写	描述	复位值
URXH0	0x50000024(L) 0x50000027(B)	R (byte)	UART通道 0 接收缓存寄存器	-
URXH1	0x50004024(L) 0x50004027(B)	R (byte)	UART通道 1 接收缓存寄存器	-
URXH2	0x50008024(L) 0x50008027(B)	R (byte)	UART通道 2 接收缓存寄存器	-

URXHn	位	描述	初始值
RXDATAn	[7:0]	UARTn的接收数据	-

### 10.5.11 UART 波特率除数寄存器

#### UART BAUD RATE DIVISOR REGISTER (UBRDIVn)

在 UART 模块中有三个 UART 波特率除数寄存器 UBRDIV0, UBRDIV1 和 UBRDIV2。存储在波特率除数寄存器 UBRDIVn 中的值用于决定串行发送接收时钟率（波特率）如下：

$$\text{UBRDIVn} = (\text{int}) (\text{UART clock} / (\text{baud rate} \times 16)) - 1$$

(UART clock: PCLK, FCLK/n or UEXTCLK)

UBRDIVn 应该从 1 到  $(2^{16}-1)$ ，仅当使用小于 PCLK 的 UEXTCLK 时可以设为 0。

例如如果波特率为 115200bps 且 UART 时钟为 40MHz, UBRDIVn 是：

$$\text{UBRDIVn} = (\text{int}) (40000000 / (115200 \times 16)) - 1$$

$$= (\text{int}) (21.7) - 1 \text{ [round to the nearest whole number]}$$

$$= 22 - 1 = 21$$

寄存器	地址	读写	描述	复位值
UBRDIV0	0x50000028	R/W	波特率除数寄存器 0	-
UBRDIV1	0x50004028	R/W	波特率除数寄存器 1	-
UBRDIV2	0x50008028	R/W	波特率除数寄存器 2	-

UBRDIVn	位	描述	初始值
UBRDIV	[15:0]	波特率分频值 UBRDIVn > 0 使用 UEXTCLK 作为输入时钟, UBRDIVn 可以置 0	-