

第 19 章 UART

目录

本章包括下列主题：

19.1 简介	19-2
19.2 控制寄存器	19-3
19.3 UART 波特率发生器 (BRG)	19-8
19.4 UART 配置	19-10
19.5 UART 发送器	19-11
19.6 UART 接收器	19-14
19.7 使用 UART 进行 9 位通信	19-18
19.8 接收中止字符	19-19
19.9 初始化	19-20
19.10 UART 的其他特性	19-21
19.11 UART 在 CPU 休眠和空闲模式下的工作	19-21
19.12 与 UART 模块相关的寄存器	19-22
19.13 设计技巧	19-23
19.14 相关应用笔记	19-24
19.15 版本历史	19-25

19.1 简介

通用异步收发器（Universal Asynchronous Receiver Transmitter, UART）模块是 dsPIC30F 系列器件提供的串行 I/O 模块之一。UART 是可以和外设（例如，个人电脑、RS-232 和 RS-485 接口）通信的全双工异步系统。

UART 模块的主要特性有：

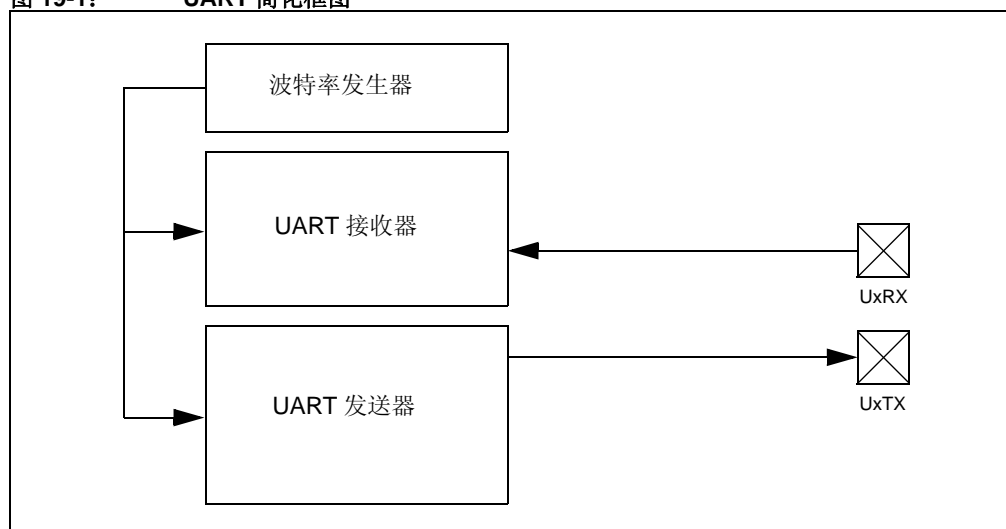
- 通过 UxTX 和 UxRX 引脚进行全双工 8 位或 9 位数据传输
- 偶、奇或无奇偶校验选项（对于 8 位数据）
- 一或两个停止位
- 完全集成的具有 16 位预分频器的波特率发生器
- 当 Fcy 为 30 MHz 时，波特率范围为 29 bps 到 1.875 Mbps
- 4 级深度先进先出（First-In-First-Out, FIFO）发送数据缓冲器
- 4 级深度 FIFO 接收数据缓冲器
- 奇偶、帧和缓冲溢出错误检测
- 支持带地址检测的 9 位模式（第 9 位 = 1）
- 发送和接收中断
- 用于诊断支持的环回模式

注： 不同的 dsPIC30F 器件可以有一个或多个 UART 模块。在引脚、控制 / 状态位和寄存器的名称中，使用“x”表示特定的模块。更多细节请参见特定的器件数据手册。

图 19-1 显示了 UART 的简化框图。UART 模块由以下至关重要的硬件元件组成：

- 波特率发生器
- 异步发送器
- 异步接收器

图 19-1: UART 简化框图



19.2 控制寄存器

寄存器 19-1: UxMODE: UARTx 模式寄存器

高字节:							
R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0
UARTEN	—	USIDL	—	保留	ALTIO	保留	保留
bit 15				bit 8			

低字节:							
R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
WAKE	LPBACK	ABAUD	—	—	PDSEL<1:0>	STSEL	
bit 7				bit 0			

- bit 15 **UARTEN:** UART 使能位
1 = UART 使能。UEN<1:0> 和 UTXEN 控制位定义了 UART 如何控制 UART 引脚。
0 = UART 禁止。UART 引脚由相应的 PORT、LAT 和 TRIS 位控制。
- bit 14 **未用:** 读作 0
- bit 13 **USIDL:** 在空闲模式停止位
1 = 当器件进入空闲模式时，停止运行
0 = 在空闲模式继续运行
- bit 12 **未用:** 读作 0
- bit 11 **保留:** 在该位置写 0
- bit 10 **ALTIO:** UART 备用 I/O 选择位
1 = UART 通过 UxATX 和 UxARX I/O 引脚通信
0 = UART 通过 UxTX 和 UxRX I/O 引脚通信

注： 不是所有的器件都具有备用 UART I/O 引脚。具体细节参见器件数据手册。
- bit 9-8 **保留:** 在这两位写 0
- bit 7 **WAKE:** 在休眠模式期间检测到启动位唤醒使能位
1 = 使能唤醒
0 = 禁止唤醒
- bit 6 **LPBACK:** UART 环回 (Loopback) 模式选择位
1 = 使能环回模式
0 = 禁止环回模式。
- bit 5 **ABAUD:** 自动波特率使能位
1 = 从 UxRX 引脚输入到捕捉模块
0 = 从 ICx 引脚输入到捕捉模块
- bit 4-3 **未用:** 读作 0
- bit 2-1 **PDSEL<1:0>:** 奇偶校验和数据选择位
11 = 9 位数据，无奇偶校验
10 = 8 位数据，奇校验
01 = 8 位数据，偶校验
00 = 8 位数据，无奇偶校验
- bit 0 **STSEL:** 停止选择位
1 = 2 个停止位
0 = 1 个停止位

图注:			
R = 可读位	W = 可写位	U = 未用位，读作 0	
-n = POR 时的值	1 = 置位	0 = 清零	x = 未知

寄存器 19-2: UxSTA: UARTx 状态和控制寄存器

高字节:							
R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R-0	R-1
UTXISEL	—	—	—	UTXBRK	UTXEN	UTXBF	TRMT
bit 15				bit 8			

低字节:							
R/W-0	R/W-0	R/W-0	R-1	R-0	R-0	R/C-0	R-0
URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR	OERR	URXDA
bit 7				bit 0			

- bit 15 UTXISEL:** 发送中断模式选择位
 1 = 当一个字符被传输到发送移位寄存器，导致缓冲器空的时候，产生中断
 0 = 当一个字符被传输到发送移位寄存器（发送缓冲器中至少还有一个字符）时，产生中断
- bit 14-12 未用:** 读作 0
- bit 11 UTXBRK:** 发送中止位
 1 = 不管发送器处在什么状态，拉低 UxTX 引脚
 0 = UxTX 引脚正常工作
- bit 10 UTXEN:** 发送使能位
 1 = UART 发送器使能，UART 控制 UxTX 引脚（如果 UARTEN = 1）
 0 = UART 发送器禁止，中止所有等待的发送，缓冲器复位。PORT 控制 UxTX 引脚。
- bit 9 UTXBF:** 发送缓冲器满状态位（只读）
 1 = 发送缓冲器满
 0 = 发送缓冲器未满，至少一个或多个数据字可以写入缓冲器。
- bit 8 TRMT:** 发送移位寄存器空位（只读）
 1 = 发送移位寄存器为空，同时发送缓冲器为空（上一个发送已经完成）
 0 = 发送移位寄存器非空，发送在进行中或在发送缓冲器中排队
- bit 7-6 URXISEL<1:0>:** 接收中断模式选择位
 11 = 接收缓冲器满时（即，有 4 个数据字符），中断标志位置位
 10 = 接收缓冲器 3/4 满时（即，有 3 个数据字符），中断标志位置位
 0x = 当接收到一个字符时，中断标志位置位
- bit 5 ADDEN:** 地址字符检测（接收数据的第 8 位 = 1）
 1 = 地址检测模式使能。如果没有选择 9 位模式，这个控制位将无效。
 0 = 地址检测模式禁止
- bit 4 RIDLE:** 接收器空闲位（只读）
 1 = 接收器空闲
 0 = 正在接收数据
- bit 3 PERR:** 奇偶校验错误状态位（只读）
 1 = 检测到当前字符的奇偶校验错误
 0 = 没有检测到奇偶校验错误
- bit 2 FERR:** 帧错误状态位（只读）
 1 = 检测到当前字符的帧错误
 0 = 没有检测到帧错误

寄存器 19-2: UxSTA: UARTx 状态和控制寄存器 (续)

- bit 1 **OERR:** 接收缓冲器溢出错误状态位 (只读 / 清零)
 1 = 接收缓冲器溢出
 0 = 接收缓冲器没有溢出
- bit 0 **URXDA:** 接收缓冲器中是否有数据位 (只读)
 1 = 接收缓冲器中有数据、有至少一个或多个字符可读
 0 = 接收缓冲器为空

图注:			
R = 可读位	W = 可写位	U = 未用位, 读作 0	C = 可清零位
-n = POR 时的值	1 = 置位	0 = 清零	x = 未知

dsPIC30F 系列参考手册

寄存器 19-3: UxRXREG: UARTx 接收寄存器

高字节:							
U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0
—	—	—	—	—	—	—	URX8
bit 15							bit 8

低字节:							
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
URX<7:0>							
bit 7							bit 0

bit 15-9 未用: 读作 0

bit 8 **URX8**: 接收到的字符的第 8 位数据 (9 位模式下)

bit 7-0 **URX<7:0>**: 接收到的字符的 7-0 位数据

图注:

R = 可读位

W = 可写位

U = 未用位, 读作 0

-n = POR 时的值

1 = 置位

0 = 清零

x = 未知

寄存器 19-4: UxTXREG: UARTx 发送寄存器 (只写)

高字节:							
U-0	U-0	U-0	U-0	U-0	U-0	U-0	W-x
—	—	—	—	—	—	—	UTX8
bit 15							bit 8

低字节:							
W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
UTX<7:0>							
bit 7							bit 0

bit 15-9 未用: 读作 0

bit 8 **UTX8**: 将要发送的字符的第 8 位数据 (9 位模式下)

bit 7-0 **UTX<7:0>**: 将要发送的字符的 7-0 位数据

图注:

R = 可读位

W = 可写位

U = 未用位, 读作 0

-n = POR 的值

1 = 置位

0 = 清零

x = 未知

寄存器 19-5: UxBRG: UARTx 波特率寄存器

高字节:							
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BRG<15:8>							
bit 15						bit 8	

低字节:							
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BRG<7:0>							
bit 7						bit 0	

bit 15-0 BRG<15:0>: 波特率除数位

图注:			
R = 可读位	W = 可写位	U= 未用位, 读作 0	
-n = POR 时的值	1 = 置位	0 = 清零	x = 未知

19.3 UART 波特率发生器 (BRG)

UART 模块包含一个专用的 16 位波特率发生器 (Baud Rate Generator, BRG)。UxBRG 寄存器控制一个自由运行的 16 位定时器的周期。公式 19-1 给出了计算波特率的公式。

公式 19-1: UART 波特率

$$\text{波特率} = \frac{F_{CY}}{16 \cdot (UxBRG + 1)}$$
$$UxBRG = \frac{F_{CY}}{16 \cdot \text{波特率}} - 1$$

注: F_{CY} 表示指令周期时钟频率。

例 19-1 给出了如下条件下的波特率误差计算:

- F_{CY} = 4 MHz
- 目标波特率 = 9600

例 19-1: 波特率误差计算

目标波特率 = F_{CY}/(16 (UxBRG + 1))

UxBRG 值的计算方法:

UxBRG	=	((F _{CY} / 目标波特率)/16)-1
UxBRG	=	((4000000/9600)/16)-1
UxBRG	=	[25.042] = 25

计算出来的波特率 = 4000000/(16 (25 + 1))

= 9615

误差 = (计算波特率 - 目标波特率) / 目标波特率

= (9615-9600)/9600

= 0.16%

最大的波特率是 F_{CY} / 16 (当 UxBRG = 0 时), 最小的波特率是 F_{CY} /(16 * 65536)。

向 UxBRG 寄存器中写新值会导致 BRG 定时器复位 (清零)。这保证了 BRG 在产生新的波特率之前不需要等待定时器溢出。

19.3.1 波特率表

表 19-1 给出了普通器件指令周期频率（Fcy）的 UART 波特率。同时也给出了每个频率下的最小和最大波特率。

表 19-1: UART 波特率

目标 波特率 (kbps)	Fcy = 30 MHz			25 MHz			20 MHz			16 MHz		
	计算 波特率	% 误差	BRG 值 (十进制)	计算 波特率	% 误差	BRG 值 (十进制)	计算 波特率	% 误差	BRG 值 (十进制)	计算 波特率	% 误差	BRG 值 (十进制)
0.3	0.3	0.0	6249	0.3	+0.01	5207	0.3	0.0	4166	0.3	+0.01	3332
1.2	1.1996	0.0	1562	1.2001	+0.01	1301	1.1996	0.0	1041	1.2005	+0.04	832
2.4	2.4008	0.0	780	2.4002	+0.01	650	2.3992	0.0	520	2.3981	-0.08	416
9.6	9.6154	+0.2	194	9.5859	-0.15	162	9.6154	+0.2	129	9.6154	+0.16	103
19.2	19.1327	-0.4	97	19.2901	0.47	80	19.2308	+0.2	64	19.2308	+0.16	51
38.4	38.2653	-0.4	48	38.1098	-0.76	40	37.8788	-1.4	32	38.4615	+0.16	25
56	56.8182	+1.5	32	55.8036	-0.35	27	56.8182	+1.5	21	55.5556	-0.79	17
115	117.1875	+1.9	15	111.6071	-2.95	13	113.6364	-1.2	10	111.1111	-3.38	8
250							250	0.0	4	250	0.0	3
500										500	0.0	1
最小值	0.0286	0.0	65535	0.0238	0.0	65535	0.019	0.0	65535	0.015	0.0	65535
最大值	1875	0.0	0	1562.5	0.0	0	1250	0.0	0	1000	0.0	0

目标 波特率 (kbps)	Fcy = 12 MHz			10 MHz			8 MHz			7.68 MHz		
	计算 波特率	% 误差	BRG 值 (十进制)	计算 波特率	% 误差	BRG 值 (十进制)	计算 波特率	% 误差	BRG 值 (十进制)	计算 波特率	% 误差	BRG 值 (十进制)
0.3	0.3	0.0	2499	0.3	0.0	2082	0.2999	-0.02	1666	0.3	0.0	1599
1.2	1.2	0.0	624	1.1996	0.0	520	1.199	-0.08	416	1.2	0.0	399
2.4	2.3962	-0.2	312	2.4038	+0.2	259	2.4038	+0.16	207	2.4	0.0	199
9.6	9.6154	-0.2	77	9.6154	+0.2	64	9.6154	+0.16	51	9.6	0.0	49
19.2	19.2308	+0.2	38	18.9394	-1.4	32	19.2308	+0.16	25	19.2	0.0	24
38.4	37.5	+0.2	19	39.0625	+1.7	15	38.4615	+0.16	12			
56	57.6923	-2.3	12	56.8182	+1.5	10	55.5556	-0.79	8			
115			6									
250	250	0.0	2				250	0.0	1			
500							500	0.0	0			
最小值	0.011	0.0	65535	0.010	0.0	65535	0.008	0.0	65535	0.007	0.0	65535
最大值	750	0.0	0	625	0.0	0	500	0.0	0	480	0.0	0

目标 波特率 (kbps)	Fcy = 5 MHz			4 MHz			3.072 MHz			1.8432 MHz		
	计算 波特率	% 误差	BRG 值 (十进制)	计算 波特率	% 误差	BRG 值 (十进制)	计算 波特率	% 误差	BRG 值 (十进制)	计算 波特率	% 误差	BRG 值 (十进制)
0.3	0.2999	0.0	1041	0.3001	0.0	832	0.3	0.0	639	0.3	0.0	383
1.2	1.2019	+0.2	259	1.2019	+0.2	207	1.2	0.0	159	1.2	0.0	95
2.4	2.4038	+0.2	129	2.4038	+0.2	103	2.4	0.0	79	2.4	0.0	47
9.6	9.4697	-1.4	32	9.6154	+0.2	25	9.6	0.0	19	9.6	0.0	11
19.2	19.5313	+1.7	15	19.2308	+0.2	12	19.2	0.0	9	19.2	0.0	5
38.4	39.0625	+1.7	7				38.4	0.0	4	38.4	0.0	2
56												
115												
250												
500												
最小值	0.005	0.0	65535	0.004	0.0	65535	0.003	0.0	65535	0.002	0.0	65535
最大值	312.5	0.0	0	250	0.0	0	192	0.0	0	115.2	0.0	0

19.4 UART 配置

UART 使用标准非归零（Non-Return-to-Zero, NRZ）格式（一个启动位，8 或 9 个数据位，一或两个停止位）。硬件提供奇偶校验，用户可以配置成偶校验，奇校验或者不使用奇偶校验。最普通的数据格式是 8 位，没有奇偶校验位，有一个停止位（用 8、N 或 1 表示），这是缺省的（POR）设置。数据位和停止位的数目，还有奇偶校验在 PDSEL<1:0>（UxMODE<2:1>）和 STSEL（UxMODE<0>）位中指定。片上专用的 16 位波特率发生器可用于根据振荡器产生标准的波特率频率。UART 先发送和接收 LSB。UART 的发送器和接收器从功能上讲是独立的，但使用相同的数据格式和波特率。

19.4.1 使能 UART

UART 模块通过置位 UARTEN（UxMODE<15>）位和 UTXEN（UxSTA<10>）位使能。一旦使能了，UxTX 和 UxRX 引脚被分别配置为输出和输入，改写了相应 I/O 端口引脚的 TRIS 和 PORT 寄存器位的设置。UxTX 引脚在没有传输发生时，状态为逻辑 1。

注： 在 UARTEN 位置位之前，不应该置位 UTXEN 位。否则将无法使能 UART 发送。

19.4.2 禁止 UART

通过清零 UARTEN（UxMODE<15>）位禁止 UART 模块。这是任何复位后的默认状态。如果禁止了 UART，所有的 UART 引脚在相应的 PORT 和 TRIS 位控制下用作端口引脚。

禁止 UART 模块将缓冲器复位为空状态。所有缓冲器中的字符丢失，同时波特率计数器也复位。

当 UART 模块禁止时，所有与之相关的错误和状态标志都复位。URXDA、OERR、FERR、PERR、UTXEN、UTXBRK 和 UTXBF 位被清零而 RIDLE 和 TRMT 被置位。其他的控制位，包括 ADDEN、URXISEL<1:0> 和 UTXISEL，还有 UxMODE 和 UxBRG 寄存器不受影响。

当 UART 处于活动状态时，对 UARTEN 位清零将中止所有有等待的发送和接收，同时如以上定义那样将该模块复位。再次使能 UART 将使用同样的配置重新启动 UART。

19.4.3 备用 UART I/O 引脚

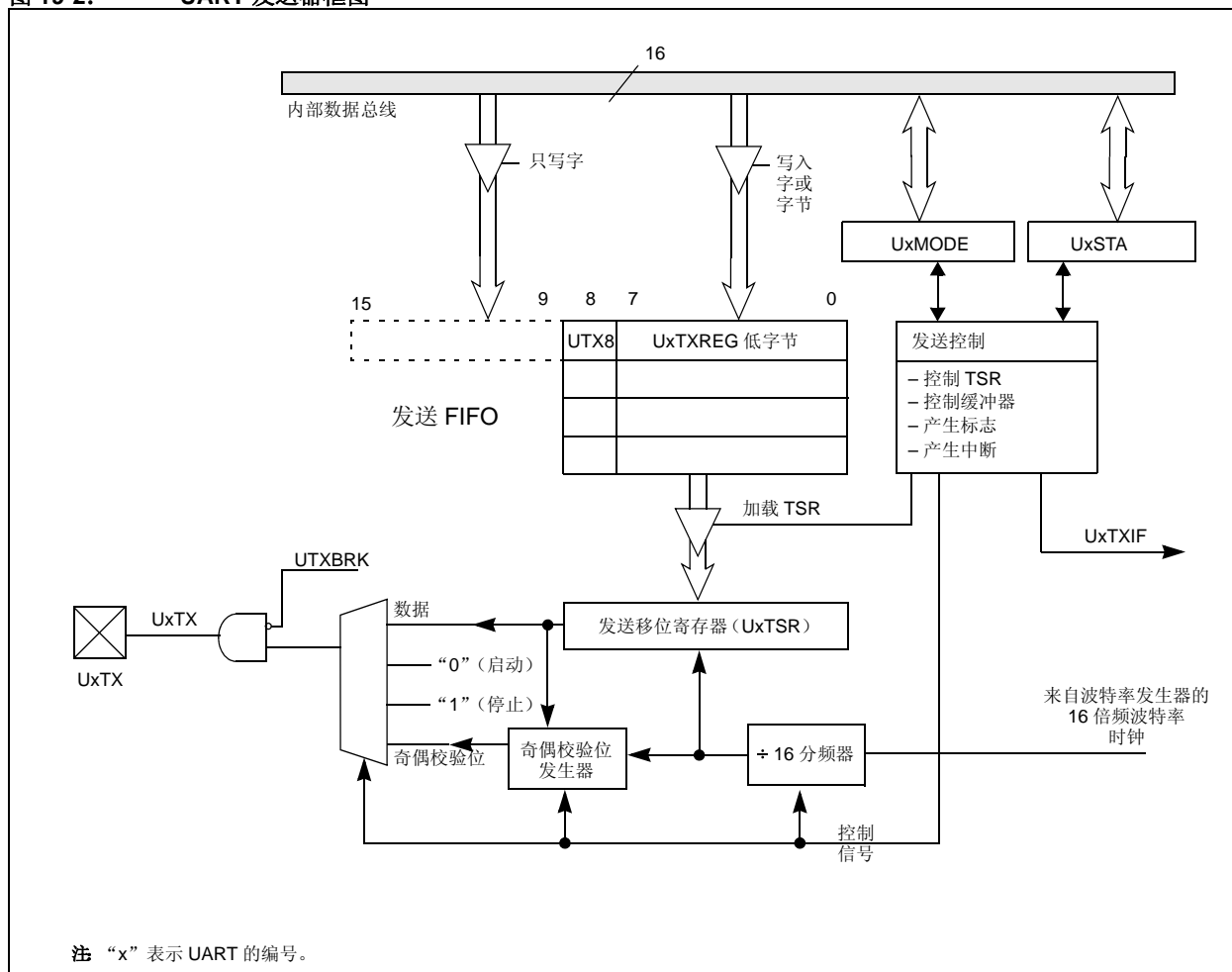
一些 dsPIC30F 器件有可用来通信的备用 UART 发送和接收引脚。当主 UART 引脚和其他外设共用时，可以使用备用 UART 引脚。备用 I/O 引脚通过置位 ALTIO 位（UxMODE<10>）来使能。如果 ALTIO = 1，UART 模块使用 UxTX 和 UxRX 引脚（分别为备用发送引脚和备用接收引脚）代替 UxTX 和 UxRX 引脚。如果 ALTIO = 0，UART 模块使用 UxTX 和 UxRX 引脚。

19.5 UART 发送器

图 19-2 显示了 UART 发送器的框图。发送器的核心是发送移位寄存器 (UxTSR)。移位寄存器从发送 FIFO 缓冲器 UxTXREG 中获得数据。UxTXREG 寄存器使用软件加载数据。直到前一次加载的停止位发送完成后才开始加载 UxTSR 寄存器。停止位一发送, UxTSR 就从 UxTXREG 寄存器加载新的数据 (如果有数据的话)。

注: UxTSR 寄存器没有映射到数据存储空间, 所以用户不能使用它。

图 19-2: UART 发送器框图



通过置位 **UTXEN** 使能位 (**UxSTA<10>**) 使能发送。实际的发送要到 **UxTXREG** 寄存器加载了数据并且波特率发生器 (**UxBRG**) 产生了移位时钟 (图 19-2) 后才发生。还可以先加载 **UxTXREG** 寄存器然后置位 **UTXEN** 使能位来启动发送。一般来说, 第一次开始发送的时候, 由于 **UxTSR** 寄存器为空, 这样传输数据到 **UxTXREG** 会导致该数据立即传输到 **UxTSR**。发送过程中对 **UTXEN** 位清零, 会导致发送中止并复位发送器。因此, **UxTX** 引脚将回复到一个高阻抗状态。

为了选择 9 位传输，PDSEL<1:0> 位 (UxMODE<2:1>) 应该设置为 11 并且第 9 位应该写到 UTX9 位 (UxTXREG<8>)。应该向 UxTXREG 执行一个字写操作，这样可以同时写入所有的 9 位。

注：在 9 位数据发送的情况下，不采取奇偶校验。

19.5.1 发送缓冲器 (UxTXB)

每个 UART 有一个 4 级深、9 位宽的 FIFO 发送数据缓冲器。UxTXREG 寄存器提供对下一个可用的缓冲单元的用户访问。用户最多可在缓冲器中写 4 个字。一旦 UxTXREG 的内容被传送到 UxTSR 寄存器，当前缓冲单元就可以写入新的数据，下一个缓冲单元将成为 UxTSR 寄存器的数据源。无论何时，只要缓冲器满了，UTXBF (UxSTA<9>) 状态位就会置位。如果用户试图写满缓冲器，则新数据将不会被 FIFO 接收。

FIFO 在任何器件复位时复位，但当器件进入省电模式或从省电模式唤醒时，FIFO 不受影响。

19.5.2 发送中断

发送中断标志 (UxTXIF) 在相应中断标志状态寄存器 (IFS) 中。UTXISEL 控制位 (UxSTA<15>) 决定 UART 何时产生一个发送中断。

1. 如果 $UTXISEL = 0$ ，当一个字从发送缓冲器传输到发送移位寄存器 (UxTSR) 时产生中断。这暗示了发送缓冲器中至少有一个空字。由于中断在每个字发送完成后产生，这个模式对于频繁处理中断是很有用的（也就是说，ISR 在下一个字发送前完成）。
2. 如果 $UTXISEL = 1$ ，当将一个字从发送缓冲器传输到发送移位寄存器 (UxTSR) 并且缓冲器为空时产生中断。由于只有在所有 4 个字都发送完后才产生中断，所以如果用户代码不能足够迅速地处理中断（也就是说，ISR 在发送下一个字前完成），这种“块发送”模式是很有用的。

当模块第一次使能时，UxTXIF 位将置位。

用户应该在 ISR 中将 UxTXIF 位清零。

可以在运行时在两个中断模式间切换。

注： 如果 $UTXISEL = 0$ ，当 UTXEN 位置位时，UxTXIF 标志位也置位，因为发送缓冲器尚未满（可以向 UxTXREG 寄存器移入待发送数据）。
--

UxTXIF 标志位指示 UxTXREG 寄存器的状态，而 TRMT 位 (UxSTA<8>) 表明 UxTSR 寄存器的状态。TRMT 状态位是一个只读位，当 UxTSR 寄存器空时置位。没有任何中断逻辑和这个位有关，所以用户必须查询该位来确定 UxTSR 寄存器是否为空。

19.5.3 设置 UART 发送

设置发送时应该遵循的步骤：

1. 初始化 UxBRG 寄存器来获得合适的波特率（第 19.3 节“UART 波特率发生器 (BRG)”）。
2. 通过写 PDSEL<1:0> (UxMODE<2:1>) 和 STSEL (UxMODE<0>) 位来设置数据位数，停止位数和奇偶校验选择。
3. 如果需要发送中断，就要置位相应的中断使能控制寄存器 (IEC) 中的 UxTXIE 控制位。使用相应中断优先级控制寄存器 (IPC) 中的 UxTXIP<2:0> 控制位来指定发送中断的中断优先级。同样，写 UTXISEL (UxSTA<15>) 位来选择发送中断模式。
4. 置位 UARTEN (UxMODE<15>) 位来使能 UART 模块。
5. 置位 UTXEN (UxSTA<10>) 位来使能发送，与此同时将置位 UxTXIF 位。在 UART 发送中断服务程序中，UxTXIF 位应该清零。UxTXIF 位的操作受 UTXISEL 控制位控制。
6. 向 UxTXREG 寄存器加载数据（开始发送）。如果选择了 9 位发送，则加载一个字。如果选择了 8 位发送，则加载一个字节。数据可以加载到缓冲器，直到 UxTXBF 状态位 (UxSTA<9>) 置位为止。

注： 在 UARTEN 位置位之前不应该置位 UTXEN 位。否则将无法使能 UART 发送。

图 19-3: 发送（8 位或 9 位数据）

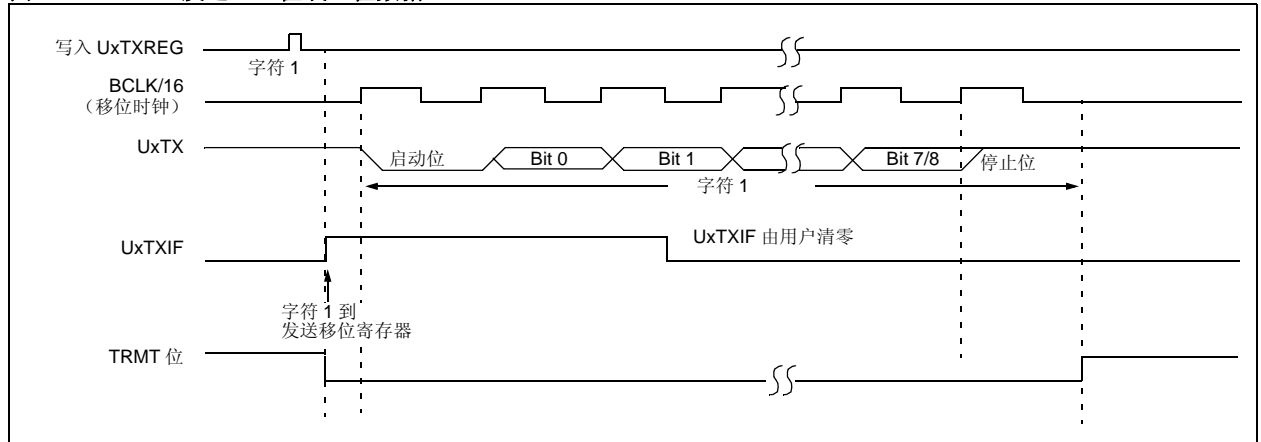
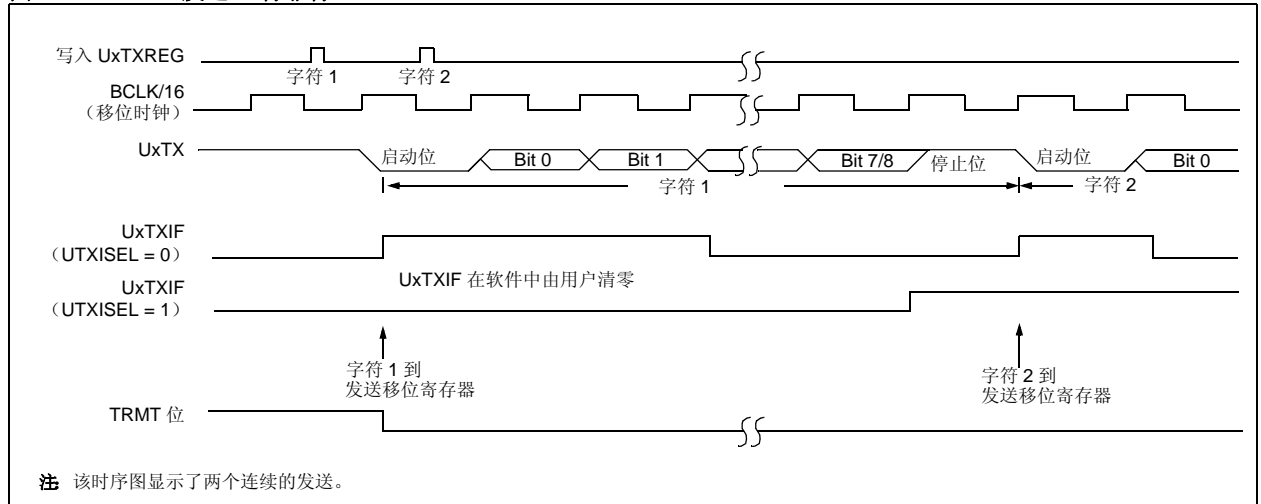


图 19-4: 发送（背靠背）



19.5.4 中止字符的发送

对 **UTXBRK** 位 (**UxSTA<11>**) 置位将强制 **UxTX** 线为 “0”。**UTXBRK** 忽略任何其他的发送器活动。用户应该在置位 **UTXBRK** 之前等待发送器空闲 (**TRMT = 1**)。

为了发送中止字符 (**Break Character**)，必须由软件置位 **UTXBRK** 位，而且保持该位置位至少 13 个波特率时钟。波特率时钟周期在软件中计时。然后通过软件将 **UTXBRK** 位清零，产生停止位。用户在再次加载 **UTXBUF** 或重新开始新的发送活动前，必须等待至少一或两个波特率时钟，来保证产生了有效的停止位。

注： 发送一个中止字符不会产生发送中断。

19.6 UART 接收器

图 19-5 显示了接收器的框图。接收器的核心是接收 (串行) 移位寄存器 (**UxRSR**)。数据在 **UxRX** 引脚上接收，并送到数据恢复区中。数据恢复区以 16 倍波特率运行，而主接收串行移位器以波特率运行。在采集到 **UxRX** 引脚上的停止位后，**UxRSR** 里面的接收到的数据传输到接收 FIFO 中 (如果为空)。

注： **UxRSR** 寄存器没有映射到数据存储空间，所以用户不可使用它。

多数检测电路三次采样 **UxRX** 引脚上的数据，确定该引脚上是高电平还是低电平。图 19-5 显示了采样机理图。

19.6.1 接收缓冲器 (**UxRXB**)

UART 接收器有一个 4 级深、9 位宽的 FIFO 接收数据缓冲器。**UxRXREG** 是一个存储器映射的寄存器可提供对 FIFO 输出的访问。在缓冲器溢出发生以前，可以有 4 个字的数据被接收并传输到 FIFO 中，从第 5 个字开始将数据移位到 **UxRSR** 寄存器中。

19.6.2 接收器错误处理

如果 FIFO 已满 (4 个字符)，而第五个字符已经完全接收到了 **UxRSR** 寄存器，溢出错误位 **OERR** (**UxSTA<1>**) 将会置位。**UxRSR** 中的字得以保留，但是只要 **OERR** 位置位，将禁止向接收 FIFO 传输后续数据。用户必须在软件中将 **OERR** 位清零，以允许更多的数据接收。

如果需要保存溢出前接收到的数据，用户应该先读所有 5 个字符，然后清零 **OERR** 位。如果这 5 个字符可以丢弃，用户只要清零 **OERR** 位即可。这可有效地复位接收 FIFO，同时先前接收到的所有数据都将丢失。

注： 接收 FIFO 中的数据应该在清零 **OERR** 位之前读出。当 **OERR** 清零时，FIFO 复位，这将导致缓冲器中所有的数据丢失。

如果停止位被检测为逻辑低电位，帧错误位 **FERR** (**UxSTA<2>**) 将置位。

如果检测到缓冲器顶部的数据字 (也就是说当前的字) 有奇偶校验错误，奇偶校验错误位 **PERR** (**UxSTA<3>**) 将置位。举例来说，如果奇偶校验设置为偶，但检测出数据中 1 的总数为奇，就产生了奇偶校验错误。**PERR** 位在 9 位模式中是无关的。**FERR** 和 **PERR** 位和相应的字一起被缓冲，并且应该在读取数据字之前读出。

19.6.3 接收中断

UART 接收中断标志 (UxRXIF) 位于相应中断标志状态寄存器 (IFS) 中。URXISEL<1:0> (UxSTA<7:6>) 控制位决定 UART 接收器何时产生中断。

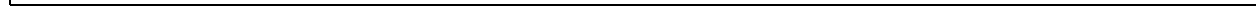
- a) 如果 URXISEL<1:0> = 00 或 01, 每当一个数据字从接收移位寄存器 (UxRSR) 传输到接收缓冲器后就会产生中断。接收缓冲器中可以有一个或多个字符。
- b) 如果 URXISEL<1:0> = 10, 当一个字从移位寄存器 (UxRSR) 传输到了接收缓冲器使得缓冲器中有 3 或 4 个字符时产生中断。
- c) 如果 URXISEL<1:0> = 11, 当一个字从移位寄存器 (UxRSR) 传输到了接收缓冲器使得缓冲器中有 4 个字符 (即, 缓冲器满) 时产生中断。

运行时可以在三个中断模式间切换。

URXDA 和 UxRXIF 标志位指示 UxRXREG 寄存器的状态, 而 RIDLE 位 (UxSTA<4>) 表明 UxRSR 寄存器的状态。RIDLE 状态位是一个只读位, 当接收器空闲 (即, UxRSR 寄存器空) 时置位。没有任何中断逻辑和这个位有关, 所以用户必须查询该位来确定 UxRSR 寄存器是否为空闲。

URXDA 位 (UxSTA<0>) 指示了接收缓冲器有数据还是为空。只要接收缓冲器中至少有一个可以读出的字符, 该位就将置位。URXDA 是只读位。

图 19-5 显示了 UART 接收器的框图。



19.6.4 设置 UART 接收

设置接收时应该遵循的步骤：

- 1. 初始化UxBRG寄存器来获得合适的波特率（第19.3节“UART 波特率发生器（BRG）”）。
- 2. 通过写 PDSEL<1:0>（UxMODE<2:1>）和 STSEL（UxMODE<0>）位来设置数据位数，停止位数和奇偶校验选择。
- 3. 如果需要中断，就要置位相应的中断使能控制寄存器（IEC）中的 UxRXIE 位。使用相应中断优先级控制寄存器（IPC）中的 UxTXIP<2:0> 控制位来指定该中断的优先级。同时，通过写 URXISEL<1:0>（UxSTA<7:6>）位选择接收中断的模式。
- 4. 通过置位 UARTEN（UxMODE<15>）位，使能 UART 模块。
- 5. 接收中断取决于 URXISEL<1:0> 控制位的设置。如果没有允许接收中断，用户可以查询 URXDA 位。UxRXIF 位应该在 UART 接收中断服务程序中清零。
- 6. 从接收缓冲器中读取数据。如果选择 9 位传输，读一个字。否则，读一个字节。无论何时，只要缓冲中有数据可读，URXDA 状态位（UxSTA<0>）就将置位。

图 19-6: UART 接收

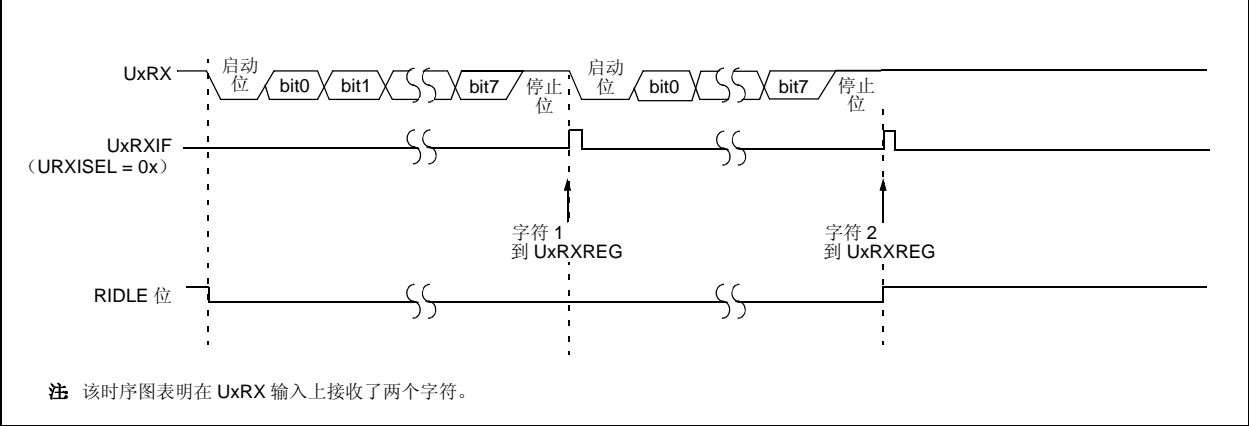
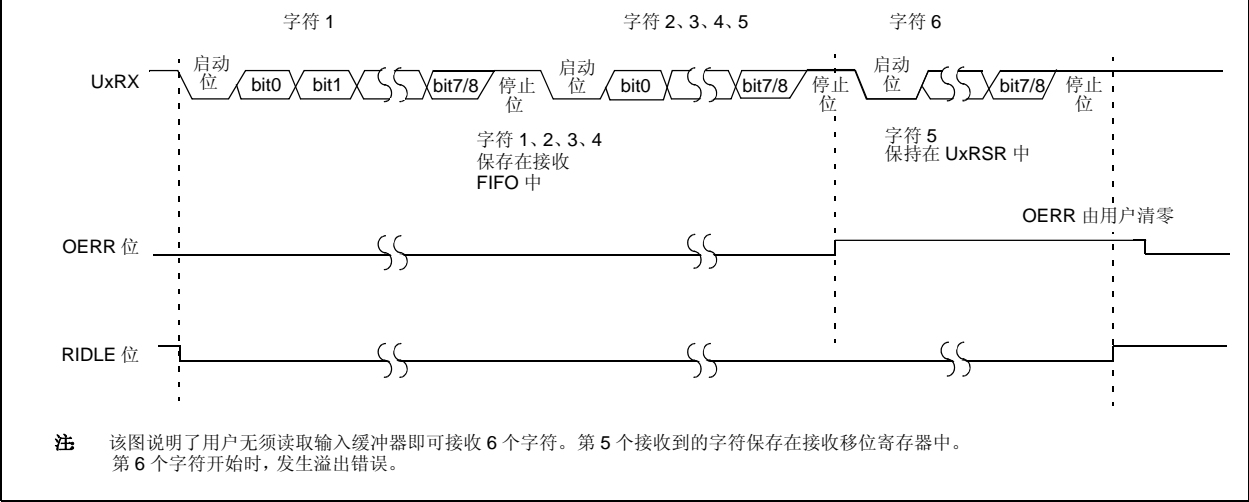


图 19-7: UART 在接收溢出下的接收



19.7 使用 UART 进行 9 位通信

典型的多处理器通信协议会区别数据字节和地址 / 控制字节。一般的方法是使用第 9 个数据位来识别数据字节是地址还是数据信息。如果第 9 位置位，数据就作为地址或控制信息处理。如果第 9 位清零，接收到的数据字就作为和前面的地址 / 控制字节相关的数据处理。

协议操作如下：

- 主器件发送一个第 9 位置位的数据字。数据字包含从器件的地址。
- 通信链中的所有从器件接收地址字并检查从地址值。
- 被寻址的从器件将接收和处理主器件发送的后续数据字节。所有其他的从器件将丢弃后面的数据字节，直到接收到新的地址字（第 9 位置位）。

19.7.1 ADDEN 控制位

UART 接收器有一个地址检测模式，该模式允许接收器忽略第 9 位清零的数据字。这降低了中断开销，因为第 9 位清零的数据字不被缓冲。这个功能通过置位 ADDEN 位（UxSTA<5>）来使能。

使用地址检测模式时，UART 必须配置为 9 位数据。当接收器配置为 8 位数据模式时，ADDEN 位无效。

19.7.2 设置 9 位发送

除了 PDSEL<1:0>（UxMODE<2:1>）应该设为 11 外，设置 9 位发送的过程几乎和设置 8 位发送模式一样（参见第 19.5.3 节“设置 UART 发送”）。

应当对 UxTXREG 寄存器执行写字操作（开始发送）。

19.7.3 设置使用地址检测模式的 9 位接收

除了 PDSEL<1:0> (UxMODE<2:1>) 应该设为 11 外, 设置 9 位接收的过程和设置 8 位接收模式类似 (参见第 19.6.4 节 “设置 UART 接收”)。

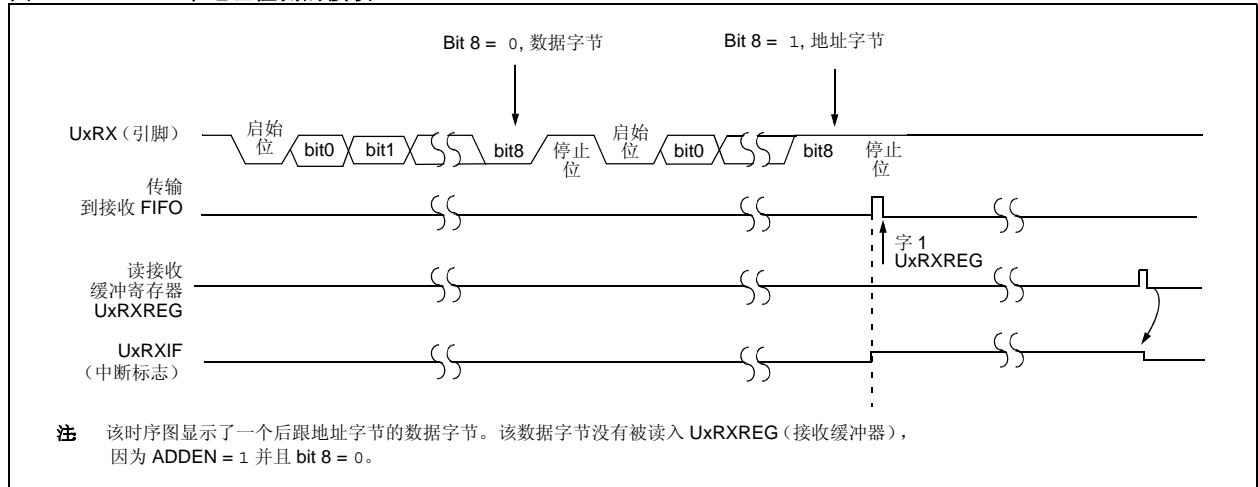
接收中断模式应该通过写 URXISEL<1:0> (UxSTA<7:6>) 位来配置。

注: 如果地址检测模式使能 (ADDEN = 1) 了, URXISEL<1:0> 控制位应该配置成接收到每个字后就产生中断。每个接收到的数据字在接收后必须立即在软件中进行检查, 看是否地址匹配。

使用地址检测模式的过程如下所述:

1. 置位 ADDEN (UxSTA<5>) 位来使能地址检测。必须保证 URXISEL 控制位配置成每接收一个字就产生一个中断。
2. 读 UxRXREG 寄存器, 检查每个 8 位地址, 确定器件是否被寻址。
3. 如果该器件没有被寻址, 就丢弃接收到的字。
4. 如果器件被寻址, 对 ADDEN 位清零, 允许后来的数据字节可以读进接收缓冲器, 并中断 CPU。如果希望长数据包, 则需要改变接收中断模式, 以使中断之间可以缓冲多于一个的数据字节。
5. 如果最后的数据字节接收到了, 置位 ADDEN 位使得只有地址字低功耗被接收。同样, 必须保证 URXISEL 控制位配置成每接收一个字就产生一个中断。

图 19-8: 带地址检测的接收 (ADDEN = 1)



19.8 接收中止字符

接收器会根据编程到 PDSEL (UxMODE<2:1>) 和 STSEL (UxMODE<0>) 位的值, 计数并等待一定的位时间数。

如果中止 (Break) 超过 13 个位时间, 在 PDSEL 和 STSEL 指定的位时间之后, 认为接收完成。URXDA位置位, FERR置位, 接收FIFO中装载零, 同时产生中断 (若允许中断) 且RIDLE位置位。

如果模块接收到了中止信号, 同时接收器检测到了启动位、数据位和无效停止位 (将 FERR 置 1), 接收器在找寻下一个启动位前必须等待有效停止位。接收器不能假定线上的中止条件就是下一个启动位。FERR 位置位且字符包含了全零被认为是中止。中止字符被加载到缓冲器中。只有接收到停止位后才可以接收更多的字符。注意当接收到停止位时 RIDLE 变高。

19.9 初始化

例 19-2 是 8 位模式的发送器 / 接收器的初始化程序。例 19-3 给出了 9 位地址检测模式下可寻址 UART 的初始化。在两个例子中，UxBRG 寄存器中加载的值取决于目标波特率和器件的频率。

注： 在 UARTEN 位置位之前不应该置位 UTXEN 位。否则将无法使能 UART 发送。

例 19-2: 8 位发送 / 接收 (UART1)

```
MOV    #baudrate,W0      ; Set Baudrate
MOV    W0,U1BRG

BSET   IPC2,#U1TXIP2     ; Set UART TX interrupt priority
BCLR   IPC2,#U1TXIP1     ;
BCLR   IPC2,#U1TXIP1     ;
BSET   IPC2,#U1TXIP2     ; Set UART TX interrupt priority
BCLR   IPC2,#U1TXIP1     ;
BCLR   IPC2,#U1TXIP1     ;

CLR    U1STA

MOV    #0x8800,W0        ; Enable UART for 8-bit data,
                        ; no parity, 1 STOP bit,
                        ; no wakeup

MOV    W0,U1MODE

BSET   U1STA,#UTXEN      ; Enable transmit

BSET   IEC0,#U1TXIE      ; Enable transmit interrupts
BSET   IEC0,#U1RXIE      ; Enable receive interrupts
```

例 19-3: 8 位发送 / 接收 (UART1)，地址检测使能

```
MOV    #baudrate,W0      ; Set Baudrate
MOV    W0,U1BRG

BSET   IPC2,#U1TXIP2     ; Set UART TX interrupt priority
BCLR   IPC2,#U1TXIP1     ;
BCLR   IPC2,#U1TXIP0     ;
BSET   IPC2,#U1RXIP2     ; Set UART RX interrupt priority
BCLR   IPC2,#U1RXIP1     ;
BCLR   IPC2,#U1RXIP0     ;

BSET   U1STA,#ADDEN      ; Enable address detect

MOV    #0x8883,W0        ; UART1 enabled for 9-bit data,
                        ; no parity, 1 STOP bit,
                        ; wakeup enabled

MOV    W0,U1MODE

BSET   U1STA,#UTXEN      ; Enable transmit

BSET   IEC0,#U1TXIE      ; Enable transmit interrupts
BSET   IEC0,#U1RXIE      ; Enable receive interrupts
```

19.10 UART 的其他特性

19.10.1 环回模式下的 UART

置位 LPBACK 位来使能这个特殊的模式，在该模式下 UxTX 输出在内部被连到 UxRX 输入。当配置为环回模式时，UxRX 引脚从内部 UART 接收逻辑断开。但是，UxTX 引脚仍然正常工作。要选择该模式：

- 1. 将 UART 配置为所需的工作模式。
- 2. 设置 LPBACK = 1 使能环回模式。
- 3. 像第 19.5 节 “UART 发送器” 定义的那样，使能发送。

环回模式取决于 UEN<1:0> 位的设置，如表 19-2 所示。

表 19-2: 环回模式引脚功能

UEN<1:0>	引脚功能， LPBACK = 1
00	UxRX 输入连接到 UxTX； UxTX 引脚工作； UxRX 引脚忽略； UxCTS/UxRTS 没有使用
01	UxRX 输入连接到 UxTX； UxTX 引脚工作； UxRX 引脚忽略； UxRTS 引脚工作， UxCTS 没有使用
10	UxRX 输入连接到 UxTX； UxTX 引脚工作； UxRX 引脚忽略； UxRTS 引脚工作， UxCTS 输入连接到 UxRTS； UxCTS 引脚忽略
11	UxRX 输入连接到 UxTX； UxTX 引脚工作； UxRX 引脚忽略； BCLK 引脚工作； UxCTS/UxRTS 没有使用

19.10.2 自动波特率支持

为了让系统确定接收字符的波特率，UxRX 输入应该在内部连到所选择的输入捕捉通道。当 ABAUD 位（UxMODE<5>）置位时，UxRX 引脚在内部被连接到输入捕捉通道。ICx 引脚从输入捕捉通道断开。

自动波特率支持所使用的输入捕捉通道是由具体器件决定的。更多的细节请参见器件数据手册。这种模式只有在 UART 使能（UARTEN = 1），同时禁止环回模式（LPBACK = 0）的情况下有效。而且，用户必须对捕捉模块进行设置，使其检测启动位的上升沿和下降沿。

19.11 UART 在 CPU 休眠和空闲模式下的工作

UART 在休眠模式下不工作。如果在发送进行期间进入休眠模式，则将中止发送并且 UxTX 引脚被驱动为逻辑 “1”。同样的，如果接收进行期间进入休眠模式，接收将中止。

UART 可以用来在检测到启动位时选择性地将 dsPIC 器件从休眠模式唤醒。如果 WAKE 位（UxSTA<7>）置位、器件处于休眠模式同时允许接收中断（UxRXIE = 1），则 UxRX 引脚上的下降沿将产生一个接收中断。接收中断选择模式位（URXISEL）对该功能没有影响。为了产生唤醒中断，UARTEN 位必须置位。

USIDL 位（UxMODE<13>）选择当器件进入空闲模式时模块是停止工作，还是在空闲模式下继续正常工作。如果 USIDL = 0，模块将在空闲模式期间继续正常工作。如果 USIDL = 1，模块将在空闲模式下停止工作。正在进行的任何发送或接收操作将中止。

19.12 与 UART 模块相关的寄存器

表 19-3: 与 UART1 相关的寄存器

SFR 名称	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
U1MODE	UARTEN	—	USIDL	—	保留	ALTIO	保留	保留	WAKE	LPBACK	ABAUD	—	—	PDSEL<1:0>		STSEL	0000 0000 0000 0000
U1STA	UTXISEL	—	—	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0000 0001 0001 0000
U1TXREG	—	—	—	—	—	—	—	UTX8	发送寄存器								0000 0000 0000 0000
U1RXREG	—	—	—	—	—	—	—	URX8	接收寄存器								0000 0000 0000 0000
U1BRG	波特率发生器预分频器																0000 0000 0000 0000
IFS0	CNIF	MI2CIF	SI2CIF	NVMIF	ADIF	U1TXIF	U1RXIF	SPI1IF	T3IF	T2IF	OC2IF	IC2IF	T1IF	OC1IF	IC1IF	INT0	0000 0000 0000 0000
IEC0	CNIE	MI2CIE	SI2CIE	NVMIE	ADIE	U1TXIE	U1RXIE	SPI1IE	T3IE	T2IE	OC2IE	IC2IE	T1IE	OC1IE	IC1IE	INT0IE	0000 0000 0000 0000
IPC2	—	ADIP<2:0>			—	U1TXIP<2:0>			—	U1RXIP<2:0>			—	SPI1IP<2:0>			0100 0100 0100 0100

注： 与 UART1 相关的寄存器如上表所示供用户参考。与其他 UART 模块相关的寄存器，请参见器件数据手册。

19.13 设计技巧

问 1: *我用 UART 发送的数据不能正确的接收。这是什么原因啊？*

答: 最普遍的接收错误的原因是为 UART 波特率发生器计算了一个错误的值。确保写进 UxBRG 寄存器的值是正确的。

问 2: *尽管 UART 接收引脚上的信号看上去是正确的，但我还是得到了帧错误。可能是什么原因？*

答: 确保下列控制位已被正确设置：

- UxBRG: UART 波特率寄存器
- PDSEL<1:0>: 奇偶校验和数据大小选择位
- STSEL: 停止位选择

19.14 相关应用笔记

本节列出了与手册的本章内容相关的应用笔记。这些应用笔记可能并不是专为 dsPIC30F 系列产品而编写的，但是概念是相关的，通过适当修改即可使用，但使用中可能会受到一定限制。目前和 UART 模块相关的应用笔记有：

标题	应用笔记编号
Serial Port Utilities Implementing Table Read and Table Write	AN547

<p>注：如需获取更多 dsPIC30F 系列器件的应用笔记和代码示例，请访问 Microchip 网站（www.microchip.com）。</p>
--

19.15 版本历史

版本 A

这是本文档的初始发行版。

版本 B

版本 B 已经过扩充，包含了对 dsPIC30F UART 模块的全面描述。

版本 C

此版本加入了本文档更新时已知的所有勘误表。

注: