



# LPC1700 系列微控制器

## 第 15 章 通用异步收发器 UART0、2、3

用户手册 Rev00.04

**广州周立功单片机发展有限公司**

地址：广州市天河北路 689 号光大银行大厦 12 楼 F4

网址：<http://www.zlgmcu.com>

## 销售与服务网络

### 广州周立功单片机发展有限公司

地址：广州市天河北路 689 号光大银行大厦 12 楼 F4 邮编：510630

电话：(020)38730972 38730976 38730916 38730917 38730977

传真：(020)38730925

网址：<http://www.zlgmcu.com>

### 广州专卖店

地址：广州市天河区新赛格电子城 203-204 室

电话：(020)87578634 87569917

传真：(020)87578842

### 南京周立功

地址：南京市珠江路 280 号珠江大厦 2006 室

电话：(025)83613221 83613271 83603500

传真：(025)83613271

### 北京周立功

地址：北京市海淀区知春路 113 号银网中心 A 座  
1207-1208 室（中发电子市场斜对面）

电话：(010)62536178 62536179 82628073

传真：(010)82614433

### 重庆周立功

地址：重庆市石桥铺科园一路二号大西洋国际大厦  
（赛格电子市场）1611 室

电话：(023)68796438 68796439

传真：(023)68796439

### 杭州周立功

地址：杭州市天目山路 217 号江南电子大厦 502 室

电话：(0571)89719480 89719481 89719482  
89719483 89719448 89719485

传真：(0571) 89719494

### 成都周立功

地址：成都市一环路南二段 1 号数码同人港 401 室（磨  
子桥立交西北角）

电话：(028) 85439836 85437446

传真：(028) 85437896

### 深圳周立功

地址：深圳市深南中路 2070 号电子科技大厦 C 座 4  
楼 D 室

电话：(0755)83781788（5 线）

传真：(0755)83793285

### 武汉周立功

地址：武汉市洪山区广埠屯珞瑜路 158 号 12128 室（华  
中电脑数码市场）

电话：(027)87168497 87168297 87168397

传真：(027)87163755

### 上海周立功

地址：上海市北京东路 668 号科技京城东座 7E 室

电话：(021)53083452 53083453 53083496

传真：(021)53083491

### 西安办事处

地址：西安市长安北路 54 号太平洋大厦 1201 室

电话：(029)87881296 83063000 87881295

传真：(029)87880865

## 目录

第 15 章 通用异步收发器 UART0/2/3.....	1
15.1 基本配置 .....	1
15.2 特性 .....	1
15.3 管脚描述 .....	1
15.4 寄存器描述 .....	1
15.4.1 UARTn 接收器缓冲寄存器 (U0RBR - 0X4000 C000, U2RBR - 0X4009 8000, U3RBR - 0X4009 C000, DLAB=0, 只读) .....	3
15.4.2 UARTn 发送保持寄存器 (U0THR - 0x4000 C000, U0THR - 0x4009 C000, U0THR - 0x4009 C000, DLAB=0, 只写) .....	3
15.4.3 UARTn 除数锁存器 LSB (U0DLL - 0x4000 C000, U2DLL - 0x4009 C000, U3DLL - 0x4009 C000, DLAB=1) 和 UARTn 除数锁存器 MSB 寄存器 (U0DLM - 0x4000 C004, U0DLM - 0x4009 C004, U0DLM - 0x4009 C004, DLAB=1) .....	3
15.4.4 UARTn 中断使能寄存器(U0IER - 0x4000 C004, U2IER - 0x4009 8004, U3IER - 0x4009 C004, DLAB=0) .....	4
15.4.5 UARTn 中断标识寄存器 (U0IIR - 0x4000 C008, U2IIR - 0x4009 8008, U3IIR - 0x4009 C008, 只读) .....	5
15.4.6 UARTn FIFO 控制寄存器 (U0FCR - 0x4000 C008, U2FCR - 0x4009 C008, U3FCR - 0x4009 C008, 只写) .....	7
15.4.7 UARTn 线控制寄存器 (U0LCR - 0x4000 C00C, U2LCR - 0x4009 800C, U3LCR - 0x4009 C00C) .....	8
15.4.8 UARTn 线状态寄存器(U0LSR - 0x4000 C014, U2LSR - 0x4009 8014, U3LSR - 0x4009 C014, 只读) .....	8
15.4.9 UARTn 高速缓存寄存器 (U0SCR - 0x4000 C01C, U2SCR - 0x4009 801C, U3SCR - 0x4009 C01C) .....	10
15.4.10 UARTn Auto-baud 控制寄存器 (U0ACR - 0x4000 C02C, U2ACR - 0x4009 801C, U3ACR - 0x4009 C020) .....	10
15.4.11 IrDA 控制寄存器 (U3ICR - 0x4009 C024, 仅存在于 UART3) .....	12
15.4.12 UARTn 小数分频器寄存器 (U0FDR - 0x4000 C028, U2FDR - 0x4009 8028, U3FDR - 0x4009 C028) .....	13
15.4.13 UARTn 发送使能寄存器 (U1TER - 0x4000 C030, U2TER - 0x4009 8030, U3TER - 0x4009C030) .....	16
15.4.14 UARTn FIFO 水平寄存器 (U0FIFOLVL - 0x4000 C058, U2FIFOLVL - 0x4009 8058, U3FIFOLVL - 0x4009 C058, 只读) .....	17
15.5 结构 .....	17

## 第15章 通用异步收发器UART0/2/3

### 15.1 基本配置

UART0/2/3 外设的配置需要使用下列寄存器：

a) 功率：位于 PCONP 寄存器中，设置 PCUART0/2/3 位。

注：复位时，UART0 会被使能 (PCUART0=1)，且 UART2/3 会被禁能 (PCUART2/3=0)。

b) 外设时钟：位于 PCLK\_SEL0 寄存器中，选择 PCLK\_UART0；在 PCLK\_SEL1 寄存器中，选择 PCLK\_UART2/3。

c) 波特率：位于 U0/2/3LCR 寄存器中 (表 15.11)，将位 DLAB 置 1，从而能够对寄存器 DLL (表 15.5) 和 DLM (表 15.6) 进行访问，设置波特率。同时，若有需要的话，设置小数分频寄存器 (表 15.17) 中的小数波特率。

d) UART FIFO：使用 U0/2/3FCR 寄存器 (表 15.10) 中的 FIFO 使能位 (位 0) 启动 FIFO。

e) 管脚：通过 PINSEL 寄存器选择 UART 管脚，并通过 PINMODE 寄存器选择管脚模式。

注：UART 接收管脚不应使能下拉电阻。

f) 中断：将 U0/2/3LCR 寄存器 (表 15.11) 中的 DLAB 位置 0，使能 UART 中断。从而能够对 U0/2/3IER 寄存器 (表 15.7) 进行访问。中断的使能是通过在 NIVC 中使用相应的中断设置使能寄存器 (Interrupt Set Enable register) 来实现的。

g) DMA：UART0/2/3 的发送和接收可通过 GPDAM 控制器进行操作。

### 15.2 特性

- 16 字节收发 FIFO；
- 寄存器的存储单元符合 '550 工业标准；
- 接收器 FIFO 触发点可为 1、4、8 和 14 字节；
- 内置波特率发生器；
- 用于精确控制波特率的小数分频器，并拥有赖以实现软件流控制的自动波特率检测能力和机制；
- 支持 DMA 发送和接收；
- UART3 还包含一种支持红外通信的 IrDA 模式。

### 15.3 管脚描述

表 15.1 UART1 管脚描述

管脚	类型	描述
RXD0、RXD2、RXD3	输入	串行输入管脚。串行接收数据
TXD0、TXD2、TXD3	输出	串行输出管脚。串行发送数据

### 15.4 寄存器描述

UART1 所包含的寄存器，其结构如表 15.2 所述。除数锁存器访问位 (DLAB) 包含在 U1CLR[7] 中，能够使能除数锁存器的访问。

表 15.2 UARTn 寄存器映射

名称	描述	访问	复位值 <sup>[1]</sup>	地址
RBR (DLAB=0)	接收缓冲寄存器。内含下一个要读取的已接收字符	RO	NA	U0RBR - 0x4000 C000 U2RBR - 0x4009 8000 U3RBR - 0x4009 C000
THR (DLAB=0)	发送保持寄存器。在此写入下一个要发送的字符	WO	NA	U0THR - 0x4000 C000 U2THR - 0x4009 8000 U3THR - 0x4009 C000
DLL (DLAB=1)	除数锁存器 LSB。波特率除数值的最低有效字节。	R/W	0x01	U0DLL - 0x4000 C000 U2DLL - 0x4009 8000 U3DLL - 0x4009 C000
DLM (DLAB=1)	除数锁存器 LSM。波特率除数值的最高有效字节。小数分频器是使用整个除数来产生波特率的	R/W	0x00	U0DLM - 0x4000 C000 U2DLM - 0x4009 8000 U3DLM - 0x4009 C000
IER (DLAB=0)	中断使能寄存器。包含 7 个独立的中断使能位对应 7 个潜在的 UART1 中断	R/W	0x00	U0IER - 0x4000 C004 U2IER - 0x4009 8004 U3IER - 0x4009 C004
IIR	中断 ID 寄存器。识别等待处理的中断	RO	0x01	U0IIR - 0x4000 C008 U2IIR - 0x4009 8008 U3IIR - 0x4009 C008
FCR	FIFO 控制寄存器。控制 UART1 FIFO 的使用和模式	WO	0x00	U0FCR - 0x4000 C008 U2FCR - 0x4009 8008 U3FCR - 0x4009 C008
LCR	线控制寄存器。包含帧格式控制和间隔产生控制	R/W	0x00	U0LCR - 0x4000 C00C U2LCR - 0x4009 800C U3LCR - 0x4009 C00C
LSR	线状态寄存器。包含发送和接收的状态标志（包括线错误）	RO	0x60	U0LSR - 0x4000 C014 U2LSR - 0x4009 8014 U3LSR - 0x4009 C014
SCR	高速缓存寄存器。8 位的临时存储空间，供软件使用	R/W	0x00	U0SCR - 0x4000 C01C U2SCR - 0x4009 801C U3SCR - 0x4009 C01C
ACR	自动波特率（auto-baud）控制寄存器。包含了 auto-baud 的特性控制	R/W	0x00	U0ACR - 0x4000 C020 U2ACR - 0x4009 8020 U3ACR - 0x4009 C020
ICR	IrDA 控制寄存器。使能并配置 IrDA 模式，仅适用于 UART3	R/W	0	U3ICR - 0x4009 C024 (只适用于 UART3)
FDR	小数分频寄存器。为波特率分频器产生时钟输入	R/W	0x10	U0FDR - 0x4000 C028 U2FDR - 0x4009 8028 U3FDR - 0x4009 C028
TER	发送使能寄存器。关闭 UART 发送器，使用软件流控制	R/W	0x80	U0TER - 0x4000 C030 U2TER - 0x4009 8030 U3TER - 0x4009 C030

续上表

名称	描述	访问	复位值 <sup>[1]</sup>	地址
U1FIFOLVL	FIFO 水平寄存器。提供了当前发送 FIFO 和接收 FIFO 的填充水平	RO	0x00	U0FIFOLVL - 0x4001 0058 U2FIFOLVL - 0x4001 0058 U3FIFOLVL - 0x4001 0058

[1] 复位值仅指使用位中保存的数据，并不包括保留位的内容。

#### 15.4.1 UARTn接收器缓冲寄存器(U0RBR - 0X4000 C000, U2RBR - 0X4009 8000, U3RBR - 0X4009 C000, DLAB=0, 只读)

UnRBR 是 UARTn RX FIFO 的最高字节。它包含了最早接收到的字符，并且可通过总线接口进行读取。LSB (位 0) 表示最“早”接收的数据位。如果接收到的字符少于 8 位，未使用的 MSB 则用 0 填充。

如果要访问 UnRBR，UnLCR 中的除数锁存器访问位 (DLAB) 必须为 0。UnRBR 为只读寄存器。

由于 PE、FE 和 BI 位与接收 FIFO 顶部的字节 (即下次读 RBR 时获取的字节) 相关，因此，要正确地成对读出有效的接收字节及其状态位，应先读取 U0LSR 的内容，然后再读取 UnRBR 中的字节。

表 15.3 UARTn 接收器缓冲寄存器位描述

位	符号	描述	复位值
7:0	RBR	UARTn 接收器缓冲寄存器包含了 UARTn RX FIFO 当中最早接收到的字节	未定义

#### 15.4.2 UARTn发送保持寄存器(U0THR - 0x4000 C000, U0THR - 0x4009 C000, U0THR - 0x4009 C000, DLAB=0, 只写)

UnTHR 是 UARTn TX FIFO 的最高字节。它是 TX FIFO 中的最新字符，可通过总线接口进行写入。LSB 代表第一个要发送出去的位。

如果要访问 UnTHR，UnLCR 中的除数锁存器访问位 (DLAB) 必须为 0。UnTHR 为只写寄存器。

表 15.4 UARTn 发送保持寄存器位描述

位	符号	描述	复位值
7:0	THR	写 UARTn 发送保持寄存器会使数据保存到 UARTn 发送 FIFO 中。当字节达到 FIFO 的出口并且发送器就绪时，字节就会被发送	NA

#### 15.4.3 UARTn除数锁存器LSB(U0DLL - 0x4000 C000, U2DLL - 0x4009 C000, U3DLL - 0x4009 C000, DLAB=1)和UARTn除数锁存器MSB寄存器(U0DLM - 0x4000 C004, U0DLM - 0x4009 C004, U0DLM - 0x4009 C004, DLAB=1)

UARTn 除数锁存器是 UARTn 波特率发生器的一部分，它与小数分频器一同使用，保持产生波特率时钟的 APB 时钟(PCLK)分频值，波特率时钟必须是波特率的 16 倍。UnDLL 和 UnDLM 寄存器一起构成了一个 16 位除数，其中 UnDLL 包含了除数的低 8 位而 UnDLM 包含了除数的高 8 位。分频值 0x0000 会被作为 0x0001 处理，因为除数不能为 0。如果要访问 UARTn 除数锁存寄存器，UnLCR 中的除数锁存器访问位 (DLAB) 必须为 1。

表 15.5 UARTn 除数锁存 LSB 寄存器位描述

位	符号	描述	复位值
7:0	DLLSB	UARTn 除数锁存 LSB 寄存器与 UnDLM 寄存器一起决定 UARTn 的波特率	0x01

表 15.6 UARTn 除数锁存 MSB 寄存器位描述

位	符号	描述	复位值
7:0	DLMSB	UARTn 除数锁存 MSB 寄存器与 U0DLL 寄存器一起决定 UARTn 的波特率	0x00

**15.4.4 UARTn 中断使能寄存器 (U0IER – 0x4000 C004, U2IER – 0x4009 8004, U3IER – 0x4009 C004, DLAB=0)**

UnIER 用于使能 3 个 UARTn 中断源。

表 15.7 UARTn 中断使能寄存器位描述

位	符号	值	描述	复位值
0	RBR Interrupt Enable	0 1	使能 UARTn 的接收数据可用中断。它还控制着字符接收超时中断 禁止 RDA 中断 使能 RDA 中断	0
1	THRE Interrupt Enable	0 1	使能 UARTn 的 THRE 中断。该中断的状态可从 U1LSR[5]中读出 禁止 THRE 中断 使能 THRE 中断	0
2	RX Line Interrupt Enable	0 1	使能 UARTn 的 RX 线状态中断。该中断的状态可从 U1LSR[4:1]中读出 禁止 RX 线中断 使能 RX 线中断	0
7:3	-		保留。用户软件不应对其写入 1。从保留位读出的值未定义	NA
8	ABEOIntEn	0 1	使能 auto-baud 结束中断 禁止 auto-baud 结束中断 使能 auto-baud 结束中断	0
9	ABTOIntEn	0 1	使能 auto-baud 超时中断 禁止 auto-baud 超时中断 使能 auto-baud 超时中断	0
31:10	-		保留。用户软件不应对其写入 1。从保留位读出的值未定义	NA

**15.4.5 UARTn中断标识寄存器 (U0IIR – 0x4000 C008, U2IIR – 0x4009 8008, U3IIR – 0x4009 C008, 只读)**

UnIIR 提供状态代码用于指示一个挂起处理中断的优先级别和中断源。在访问 UnIIR 过程中, 中断被冻结。如果在访问 UnIIR 过程中产生了中断, 该中断会被记录, 下次访问 UnIIR 时可读出该中断。

表 15.8 UARTn 中断标识寄存器位描述

位	符号	值	描述	复位值
0	IntStatus	0 1	中断状态。 注: UnIIR[0]为低电平有效。挂起的中断可通过 UnIIR[3:1]确定 至少有一个中断被挂起 没有挂起的中断	0
3:1	IntId	011 010 110 001	中断标识。UnIER[3:1]指示对应 UARTn RX FIFO 的中断。上面未列出的 UnIER[3:1]的其它组合都为保留值 (000、100、101、111) 1 – 接收线状态 (RLS) 2a – 接收数据可用 (RDA) 2b – 字符超时指示器 (CTI) 3 – THRE 中断	0
5:4	-		保留。用户软件不应对其写入 1。从保留位读出的值未定义	NA
7:6	FIFO Enable		这些位等效于 UnFCR[0]	0
8	ABEOInt		Auto-baud 结束中断。若 auto-baud 已成功结束且中断被使能, 则为真	0
9	ABTOInt		Auto-baud 超时中断。若 auto-baud 已超时且中断被使能, 则为真	0
31:10	-		保留。用户软件不应对其写入 1。从保留位读出的值未定义	NA

位 UnIIR[9:8]由 auto-baud 功能设置, 用于发布超时信号或 auto-baud 结束条件信号。而 Auto-baud 中断条件的清除则是通过设置 Auto-baud 控制寄存器中的相应 Clear (清除) 位来实现。

如果IntStatus位为 1, 表示没有中断被挂起, 此时IntId位为 0。如果IntStatus位为 0, 表示有一个非auto-baud中断被挂起, 此时IntId位会指出中断的类型, 中断的处理见表 15.9。给定了 UnIIR[3:0]的状态, 中断处理程序就能确定中断源以及如何清除激活的中断。在退出中断服务程序之前, 必须读取UnIIR来清除中断。

UARTn RLS 中断 (UnIIR[3:1]=011) 是最高优先级中断, 只要 UARTn Rx 输入产生 4 个错误条件 (溢出错误 OE、奇偶错误 PE、帧错误 FE 以及间隔中断 BI) 中的任意一个, 该位就会被置位。产生中断的 UARTn Rx 错误条件可通过查看 UOLSR[4:1]来得到。当读取 UnLSR 时, 中断就会被清除。

UARTn RDA 中断 (UnIIR[3:1]=010) 与 CTI 中断 (UnIIR[3:1]=110) 共用第二优先级。当 UARTn Rx FIFO 深度到达 U1FCR[7:6]所定义的触发点时,RDA 就会被激活;当 UARTn Rx FIFO 深度低于触发点时, RDA 复位。当 RDA 中断激活时, CPU 可读出由触发点所定义的数据块。

CTI 中断 (UnIIR[3:1]=110) 是一个第二优先级中断, 当 UARTn Rx FIFO 内含有至少一个字符并且在接收到 3.5 到 4.5 字符的时间内没有发生 UARTn Rx FIFO 动作时, 该中断激活。任何 UARTn Rx FIFO 动作 (UARTn RSR 的读取或写入) 将会清除该中断。当接收到的信息不是触发点值的倍数时, CTI 中断将会清空 UARTn RBR。例如: 如果外围设备想要发送一个长度为 105 个字符的信息, 而触发值为 10 个字符, 那么前 100 个字符将使 CPU 接收 10 个 RDA 中断, 而剩下的 5 个字符使 CPU 收到 1 到 5 个 CTI 中断 (取决于服务程序)。

表 15.9 UARTn 中断处理

U0IIR[3:0]值 <sup>[1]</sup>	优先级	中断类型	中断源	中断复位
0001	-	无	无	-
0110	最高	RX 线状态/错误	OE <sup>[2]</sup> 、PE <sup>[2]</sup> 、FE <sup>[2]</sup> 或 BI <sup>[2]</sup>	UnLSR <sup>[2]</sup> 读操作
0100	第二	RX 数据可用	Rx 数据可用或达到 FIFO 的触发点 (UnFCR0=1)	UnLSR <sup>[3]</sup> 读操作或 UARTn FIFO 低于 触发值
1100	第二	字符超时指示	RX FIFO 中至少有一个字符, 并且在一段时间内没有字符输入或移出, 该时间的长短取决于 FIFO 中的字符数以及在 3.5 到 4.5 字符的时间内的触发值 实际时间为: [(字长度)×7-2]×8+[(触发值-字符数)×8+1]RCLK	UnRBR <sup>[3]</sup> 读操作
0010	第三	THRE	THRE <sup>[2]</sup>	UnIIR <sup>[4]</sup> 读 操作 (如果 UnIIR 是中 断源) 或 THR 写操 作

[1] “0000”、“0011”、“0101”、“0111”、“1000”、“1010”、“1011”、“1101”、“1110”、“1111” 均为保留值。

[2] 详细信息, 请参见15.4.8“UARTn线状态寄存器 (U0LSR – 0x4000 C014, U2LSR – 0x4009 8014, U3LSR – 0x4009 C014, 只读)”小节。

[3] 详细信息, 请参见15.4.1“UARTn接收器缓冲寄存器 (U0RBR - 0X4000 C000, U2RBR - 0X4009 8000, U3RBR - 0X4009 C000, DLAB=0, 只读)”小节。

[4] 详细信息, 请参见15.4.5“UARTn中断标识寄存器 (U0IIR – 0x4000 C008, U2IIR – 0x4009 8008, U3IIR – 0x4009 C008, 只读)”小节以及15.4.2“UARTn发送保持寄存器 (U0THR – 0x4000 C000, U0THR – 0x4009 C000, U0THR – 0x4009 C000, DLAB=0, 只写)”小节。

UARTn THRE 中断 (UnIIR[3:1]=001) 是第三优先级中断, 当 UARTn THR FIFO 为空且满足特定的初始化条件时, 该中断激活。这些初始化条件是为了让 UARTn THR FIFO 有机会填入数据, 以免在系统启动时产生许多 THRE 中断而规定的。当 THRE=1 时, 且在上次 LSR 寄存器的 THRE=1 事件后, THR 中没有出现至少两个字符时, 这些初始化条件就会实现一个字符减去停止位的延时。当没有解码和服务 THRE 中断时, 该延时为 CPU 提供了写数据到 THR 的时间。当 UARTn THR FIFO 中曾经同时出现两个或更多字符, 而当前的 THR 为空时, THRE 中断就会立即被设置。当发生 UnTHR 写操作或 UnIIR 读操作, 并且 THRE 为最高优先级中断 (UnIIR[3:1]=001) 时, THRE 中断复位。

**15.4.6 UARTn FIFO控制寄存器 (U0FCR – 0x4000 C008, U2FCR – 0x4009 C008, U3FCR – 0x4009 C008, 只写)**

UnFCR 控制 UARTn RX 和 TX FIFO 的操作。

表 15.10 UnFCR 控制寄存器位描述

位	符号	值	描述	复位值
0	FIFO Enable	0	UARTn FIFO 被禁止。禁止在应用中使用	0
		1	高电平有效, 使能对 UARTn RX FIFO 和 TX FIFO 以及 UnFCR[7:1]的访问。该位必须置位以实现正确的 UARTn 操作。该位的任何变化都将使 UARTn FIFO 清空	
1	RX FIFO Reset	0	对两个 UARTn FIFO 均无影响	0
		1	写 1 到 UnFCR[1]将会清零 UARTn Rx FIFO 中的所有字节, 并复位指针逻辑, 清除接收 DMA 请求	
2	TX FIFO Reset	0	对两个 UARTn FIFO 均无影响	0
		1	写 1 到 UnFCR[2]将会清零 UARTn Tx FIFO 中的所有字节, 并复位指针逻辑, 清除发送 DMA 请求。该位会自动清零	
3	DMA Mode Select		当 FIFO 使能位 (该寄存器的位 0) 被置位时, 该位选择 DMA 模式	0
5:4	-		保留。用户软件不应对其写入 1。从保留位读出的值未定义	NA
7:6	RX Trigger Level	00	这两个位决定了接收 UARTn FIFO 在激活中断前必须写入的字符数量 触发点 0 (默认 1 字节或 0x01)	0
		01	触发点 1 (默认 4 字节或 0x04)	
		10	触发点 2 (默认 8 字节或 0x08)	
		11	触发点 3 (默认 14 字节或 0x0E)	

通过使用 DMA, 用户可选择操作 UART 的发送和/或接收。DMA 模式由 FCR 寄存器中的 DMA 模式选择位决定。只有在 FCR 寄存器中的 FIFO 使能位将 FIFO 使能时, 该位才会有用。

### UART 接收 DMA

在 DMA 模式中，当接收 FIFO 水平大于或等于触发点时，或者在发生字符超时的情况下，接收 DMA 请求就会生效。请参考上文对 RX 触发点的描述。接收 DMA 请求由 DMA 控制器清除。

### UART 发送 DMA

在 DMA 模式中，当发送 FIFO 变为未滿时，发送 DMA 请求就会生效。发送 DMA 请求由 DMA 控制器清除。

## 15.4.7 UARTn 线控制寄存器 (U0LCR – 0x4000 C00C, U2LCR – 0x4009 800C, U3LCR – 0x4009 C00C)

UnLCR 确定了发送或接收数据字符的格式。

表 15.11 UARTn 线控制寄存器位描述

位	符号	值	描述	复位值
1:0	Word Length Select	00	5 位字符长度	0
		01	6 位字符长度	
		10	7 位字符长度	
		11	8 位字符长度	
2	Stop Bit Select	0	1 个停止位	0
		1	2 个停止位 (如果 U1LCR[1:0]=00 时, 为 1.5 个)	
3	Parity Enable	0	禁止奇偶产生和校验	0
		1	使能奇偶产生和校验	
5:4	Parity Select	00	奇校验。发送字符和添加的校验位所包含的“1”的数量是一个奇数	0
		01	偶校验。发送字符和添加的校验位所包含的“1”的数量是一个偶数	
		10	强制为“1” stick 校验	
		11	强制为“0” stick 校验	
6	Break Control	0	禁止间隔发送	0
		1	使能间隔发送。当 U1LCR[6]为高电平 (有效) 时, 输出管脚 UART1 TXD 强制为逻辑 0	
7	Divisor Latch Access Bit (DLAB)	0	禁止访问除数锁存器	0
		1	使能访问除数锁存器	

## 15.4.8 UARTn 线状态寄存器 (U0LSR – 0x4000 C014, U2LSR – 0x4009 8014, U3LSR – 0x4009 C014, 只读)

UnLSR 是一个只读寄存器，提供 UARTn TX 和 RX 模块的状态信息。

表 15.12 UARTn 线状态寄存器位描述

位	符号	值	描述	复位值
0	Receiver Data Ready (RDR)	0 1	当 UnRBR 包含未读字符时, UnLSR[0]就会被置位; 当 UARTn RBR FIFO 为空时, UnLSR[0]就会被清零 UnRBR 为空 UnRBR 包含有效数据	0
1	Overrun Error (OE)	0 1	溢出错误条件在错误发生后立即设置。读 UnLSR 会清零 UnLSR[1]。当 UARTn RBR 已有新的字符就绪, 而 UARTn RBR FIFO 已满时, UnLSR[1]机会被置位。此时, UARTn RBR FIFO 将不会被覆盖, UARTn RSR 内的字符将会丢失 溢出错误状态未激活 溢出错误状态激活	0
2	Parity Error (PE)	0 1	当接收字符的校验位处于错误状态时, 校验错误就会产生。读 UnLSR 会清零 UnLSR[2]。校验错误检测时间取决于 UnFCR[0] 注: 校验错误与 UARTn RBR FIRO 中读出的字符相关 校验错误状态未激活 校验错误状态激活	0
3	Framing Error (FE)	0 1	当接收字符的停止位为逻辑 0 时, 就会发生帧错误。读 UnLSR 会清零 UnLSR[3]。帧错误检测时间取决于 UnFCR0。当检测到有帧错误时, RX 会尝试与数据重新同步, 并假设错误的停止位实际是一个超前的起始位。但即使没有出现帧错误, 它也无法假设下一个接收到的字符是正确的 注: 帧错误与 UARTn RBR FIRO 中读出的字符相关 帧错误状态未激活 帧错误状态激活	0
4	Break Interrupt (BI)	0 1	在发送整个字符(起始位、数据、校验位以及停止位)过程中, RXDn 如果保持在空闲状态(全“0”), 则产生间隔中断。一旦检测到间隔条件, 接收器立即进入空闲状态, 直到 RXDn 进入标记状态(全“1”)。读 UnLSR 会清零 UnFCR[0] 注: 间隔中断与 UARTn RBR FIRO 中读出的字符相关 间隔中断状态未激活 间隔中断状态激活	0
5	Transmitter Holding Register Empty (THRE)	0 1	当检测到 UARTn THR 已空时, THRE 就会立即被设置。写 UnTHR 会清零 THRE UnTHR 包含有效字符 UnTHR 为空	0

续上表

位	符号	值	描述	复位值
6	Transmitter Empty (TEMT)	0 1	当 UnTHR 和 UnTSR 同时为空时, TEMT 就会被设置; 而当 UnTSR 或 UnTHR 任意一个包含有效数据时, TEMT 就会被清零 UnTHR 和/或 UnTSR 包含有效数据 UnTHR 和 UnTSR 为空	0
7	Error in RX FIFO (RXFE)	0 1	当一个带有 RX 错误 (如: 帧错误、校验错误或间隔中断) 的字符载入到 UnRBR 中时, UnLSR[7] 就会被设置。当 UnLSR 寄存器被读取并且 UARTn FIFO 中不再有错误时, 该位就会清零 UnRBR 中没有 UARTn RX 错误或 UnFCR[0]=0 UARTn RBR 包含至少一个 UARTn RX 错误	0

### 15.4.9 UARTn 高速缓存寄存器 (U0SCR – 0x4000 C01C, U2SCR – 0x4009 801C, U3SCR – 0x4009 C01C)

UnSCR 不会对 UARTn 操作有影响。用户可自由对该寄存器进行读写。不提供中断接口向主机指示 UnSCR 所发生的读或写操作。

表 15.13 UARTn 高速缓存寄存器位描述

位	符号	描述	复位值
7:0	Pad	一个可读、可写的字节	0x00

### 15.4.10 UARTn Auto-baud 控制寄存器 (U0ACR – 0x4000 C02C, U2ACR – 0x4009 801C, U3ACR – 0x4009 C020)

在用户测量波特率的输入时钟/数据速率期间, 整个测量过程就是由 UARTn Auto-baud 控制寄存器 (UnACR) 进行控制的。用户可自由地读写该寄存器。

表 15.14 Auto-baud 控制寄存器位描述

位	符号	值	描述	复位值
0	Start	0 1	在 auto-baud 功能结束后, 该位会自动清零 Auto-baud 功能停止 (auto-baud 功能不运行) Auto-baud 功能启动 (auto-baud 功能正在运行)。Auto-baud 运行位。该位会在 auto-baud 功能结束后自动清零	0
1	Mode	0 1	Auto-baud 模式选择位 模式 0 模式 1	0
2	AutoRestart	0 1	不重新启动 如果超时则重新启动 (计数器会在下一个 UART1 RX 下降沿重新启动)	0
7:3	-	NA	保留。用户软件不应对其写入 1。从保留位读出的值未定义	0

续上表

位	符号	值	描述	复位值
8	ABEOIntClr	0 1	Auto-baud 结束中断清零位（仅可写访问） 写 0 无影响 写 1 将 U1IIR 中相应的中断清除	0
9	ABTOIntClr	0 1	Auto-baud 超时中断清零位（仅可写访问） 写 0 无影响 写 1 将 U1IIR 中相应的中断清除	0
31:10	-	NA	保留。用户软件不应对其写入 1。从保留位读出的值未定义	0

### 1. 自动波特率 (Auto-baud)

UARTn auto-baud 功能可用于测量基于“AT”协议 (Hayes 命令) 的输入波特率。如果 auto-baud 功能被使能, 那么 auto-baud 功能部件将测量接收数据流中的 1 位所消耗的时间, 并根据这个结果来设置除数锁存寄存器 UnDLM 和 UnDLL。

Auto-baud 功能是通过置位 UnACR 起始位来启动的, 并通过清零 UnACR 起始位来停止。Auto-baud 一旦结束, 起始位就将自动清零, 并且对该起始位进行读取将会返回 auto-baud 的状态 (挂起/完成)。

可通过设置 UnACR 模式位来使用两种 auto-baud 测量模式。在模式 0 下, 波特率是通过对 UARTn RX 管脚上两个连续的下降沿进行测量 (起始位的下降沿和第一位的下降沿) 来得到的。而在模式 1 下, 波特率则是通过测量 UARTn RX 管脚上的下降沿和后续的上升沿之间的时间 (起始位的长度) 来得到的。

如果出现超时 (速率测量计数器溢出), UnACR AutoRestart 位可用于自动重启波特率测量。如果该位被置位, 速率测量将会在 UARTn RX 管脚的下一个下降沿重新启动。

Auto-baud 功能会产生两种中断:

- UnIIR ABTOInt 中断 (UnIER ABTOIntEn 置位且自动波特率测量寄存器溢出);
- UnIIR ABEOInt 中断 (UnIER ABEOIntEn 置位且 auto-baud 已经成功完成)。

Auto-baud 中断必须通过置位相应的 UnACR ABTOIntClr 位和 ABEOIntEn 位来清零。

在 auto-baud 期间, 小数波特率发生器通常被禁用 (即 DIVADDVAL = 0)。但是, 如果波特率发生器被启动 (即 DIVADDVAL > 0), 那么它将影响 UARTn Rx 管脚波特率的测量, 但 UnFDR 寄存器的值在速率测量后不会被修改。此外, 当使用 auto-baud 时, 任何对 UnDLM 和 UnDLL 寄存器的写操作都必须在写 UnACR 寄存器之前完成。UARTn 支持的最小和最大波特率受 PCLK、数据的位数、停止位以及校验位的影响。

$$\text{波特率最小值} = \frac{2 \times \text{PCLK}}{16 \times 2^{15}} \leq \text{UARTn 波特率} \leq \frac{\text{PCLK}}{16 \times (2 + \text{数据位} + \text{校验位} + \text{停止位})} = \text{最大波特率}$$

### 2. Auto-baud 模式

当软件执行“AT”命令时, 它采用期望的字符格式对 UARTn 进行配置, 并置位 UnACR 起始位。用户不必关心除数锁存寄存器 UnDLM 和 UnDLL 内的初始值。由于字母“A”或“a”的 ASCII 编码 (“A” = 0x41, “a” = 0x61) 的关系, UART1 Rx 管脚所发送的起始位以及期望字符的 LSB 是由两个下降沿来限定的。当 UnACR 起始位被置位时, auto-baud 协议将执行以下阶段:

a) UnACR 起始位一置位, 波特率测量计数器立即复位, 同时 UARTn UnRSR 复位。UnRSR 波特率切换为最高速率。

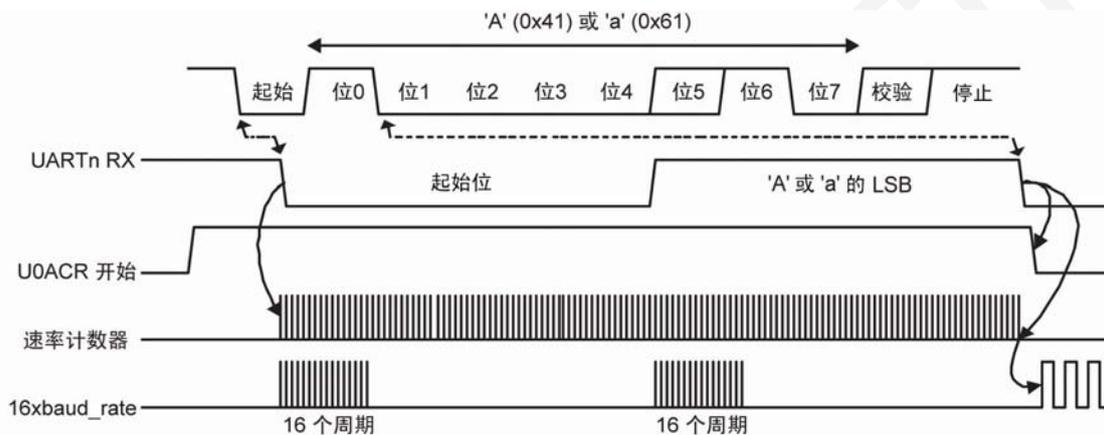
b) UARTn Rx 管脚上的下降沿触发起始位的开始。波特率测量计数器将开始对 PCLK（可选择被小数波特率发生器预分频）进行计数。

c) 在接收起始位的过程中，RSR 波特率输入端会产生 16 个脉冲，脉冲频率和（小数波特率发生器预分频）UARTn 输入时钟相同，这样，保证了起始位存放在 UnRSR 中。

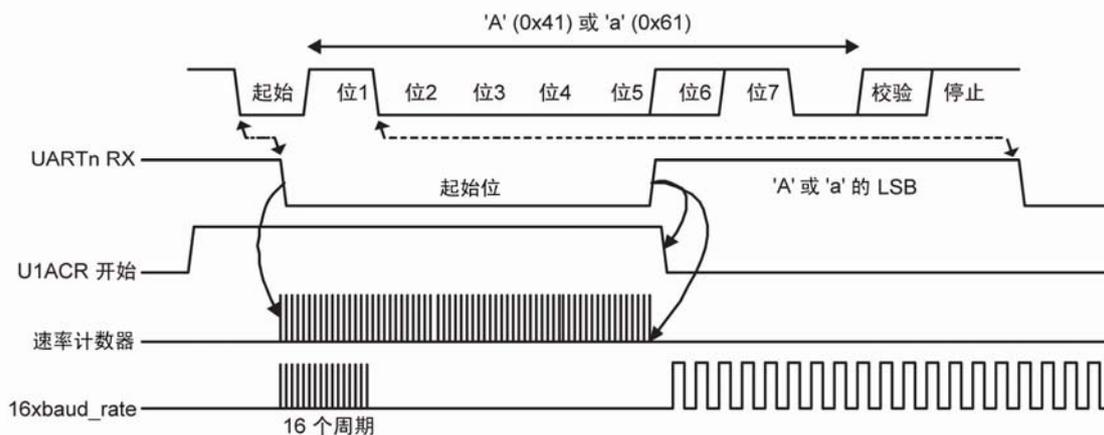
d) 在接收起始位的过程中（以及模式 0 下的字符 LSB），速率计数器将随着被预分频的 UART1 输入时钟（PCLK）递增。

e) 如果是模式 0，那么速率计数器将会在 UARTn RX 管脚的下一个下降沿停止。如果是模式 1，那么速率计数器将会在 UARTn RX 管脚的下一个上升沿停止。

f) 速率计数器的值被载入到 UnDLM/UnDLL 中，并且波特率将会切换到正常操作模式。在设置完 UnDLM/UnDLL 后，如果 auto-baud 结束中断被使能，UnIIR ABEOInt 将会被置位。接着 UnRSR 继续接受“A/a”字符剩下的其它位。



a. 模式 0（起始位和 LSB 均用于 auto-baud 功能）



b. 模式 1（只有起始位是用于 auto-baud 功能）

图 15.1 自动波特率 a) 模式 0 和 b) 模式 0 的波形图

### 15.4.11 IrDA 控制寄存器（U3ICR – 0x4009 C024，仅存在于 UART3）

IrDA 控制寄存器用于将 UART3 配置成 IrDA 模式。在发送或接收数据时，不应改变 U3ICR 的值，否则会造成数据丢失或损坏。

表 15.15 IrDA 控制寄存器

位	符号	值	描述	复位值
0	IrDAEn	0	禁能 UART3 的 IrDA 模式, UART3 作为标准的 UART 使用	0
		1	使能 UART3 的 IrDA 模式	
1	IrDAInv		为 1 时, 串行输入反相。为 0 时, 串行输入不会反相	0
2	FixPulseEn		为 1 时, 使能 IrDA 固定脉冲宽度模式	0
5:3	PulseDiv		当 FixPulseEn=1 时, 对脉冲进行配置。详细信息参考下文	0
31:6	-	NA	保留。用户软件不应向保留位写 1。从保留位读出的值为未定义	0

在 IrDA 模式下, 若启用了固定脉冲宽度模式 (IrDAEn=1 和 FixPulseEn=1), U3ICR 的 PulseDiv 位用于选择脉冲宽度。用户必须对这些位的值进行设置, 使得结果脉冲宽度至少为 1.63μs。给出了可能会使用到的脉冲宽度。

表 15.16 IrDA 脉冲宽度

FixPulseEn	PulseDiv	IrDA 发送脉冲宽度 (μs)
0	X	3/ (16×波特率)
1	0	2×T <sub>PLCK</sub>
1	1	4×T <sub>PLCK</sub>
1	2	8×T <sub>PLCK</sub>
1	3	16×T <sub>PLCK</sub>
1	4	32×T <sub>PLCK</sub>
1	5	64×T <sub>PLCK</sub>
1	6	128×T <sub>PLCK</sub>
1	7	256×T <sub>PLCK</sub>

#### 15.4.12 UARTn 小数分频器寄存器 (U0FDR - 0x4000 C028, U2FDR - 0x4009 8028, U3FDR - 0x4009 C028)

UART0/2/3 小数分频寄存器 (U0/2/3FDR) 控制产生波特率的时钟预分频器, 并且用户可以自由对该寄存器进行读写操作。预分频器使用 APB 时钟并根据指定的小数要求产生输出时钟。

注: 如果激活了小数分频器 (DIVADDVAL > 0) 且 DLM = 0, DLL 寄存器的值必须大于或等于 2。

表 15.17 UARTn 小数分频寄存器位描述

位	符号	值	描述	复位值
3:0	DIVADDVAL	0	产生波特率的预分频除数值。如果该字段为 0, 小数波特率产生器将不会影响 UARTn 的波特率	0
7:4	MULVAL	1	波特率预分频乘数值。不管是否使用小数波特率发送器, 为了让 UARTn 正常运作, 该字段必须大于或等于 1	1
31:8	-	NA	保留。用户软件不应对其写入 1。从保留位读出的值未定义	0

该寄存器控制产生波特率的时钟预分频器。寄存器的复位值会让 UART0/2/3 的小数功能保持在禁用状态，从而确保 UART1 完全软件和硬件兼容没有配备该特性的 UART。

UART0/2/3 波特率的计算 (n =0/2/3) :

$$\text{UARTn}_{\text{波特率}} = \frac{\text{PCLK}}{16 \times (256 \times \text{UnDLM} \times \text{UnDLL}) \times \left(1 + \frac{\text{DivAddVal}}{\text{MulVal}}\right)}$$

其中 PCLK 为外设时钟，U1DLM 和 U1DLL 为标准 UART1 波特率分频器寄存器，而 DIVADDVAL 和 MULVAL 为 UART1 小数波特率发生器的制定参数。

ULVAL 和 DIVADDVAL 的值应符合以下条件：

- a)  $0 < \text{MULVAL} \leq 15$ ;
- b)  $0 \leq \text{DIVADDVAL} < 15$ ;
- c)  $\text{DIVADDVAL} < \text{MULVAL}$ 。

U0/2/3FDR 寄存器的值在发送/接收数据的过程中不应进行更改，否则可能会导致数据丢失或损坏。

如果 U0/2/3FDR 寄存器值不满足上述两个要求，小数分频器输出则为未定义。如果 DIVADDVAL 为 0，小数分频器将被禁能，并且不会对时钟进行分频。

#### 波特率计算：

UART 可以与小数分频器一起工作，也可以不使用小数分频器。在实际应用中，使用几种不同的小数分频器设置很可能会获得目标波特率。下面的算法给出了一个得到 DLM、DLL、MULVAL 以及 DIVADDVAL 参数集的方法。该参数集允许实际波特率与目标波特率之间存在小于 1.1% 的相对误差。

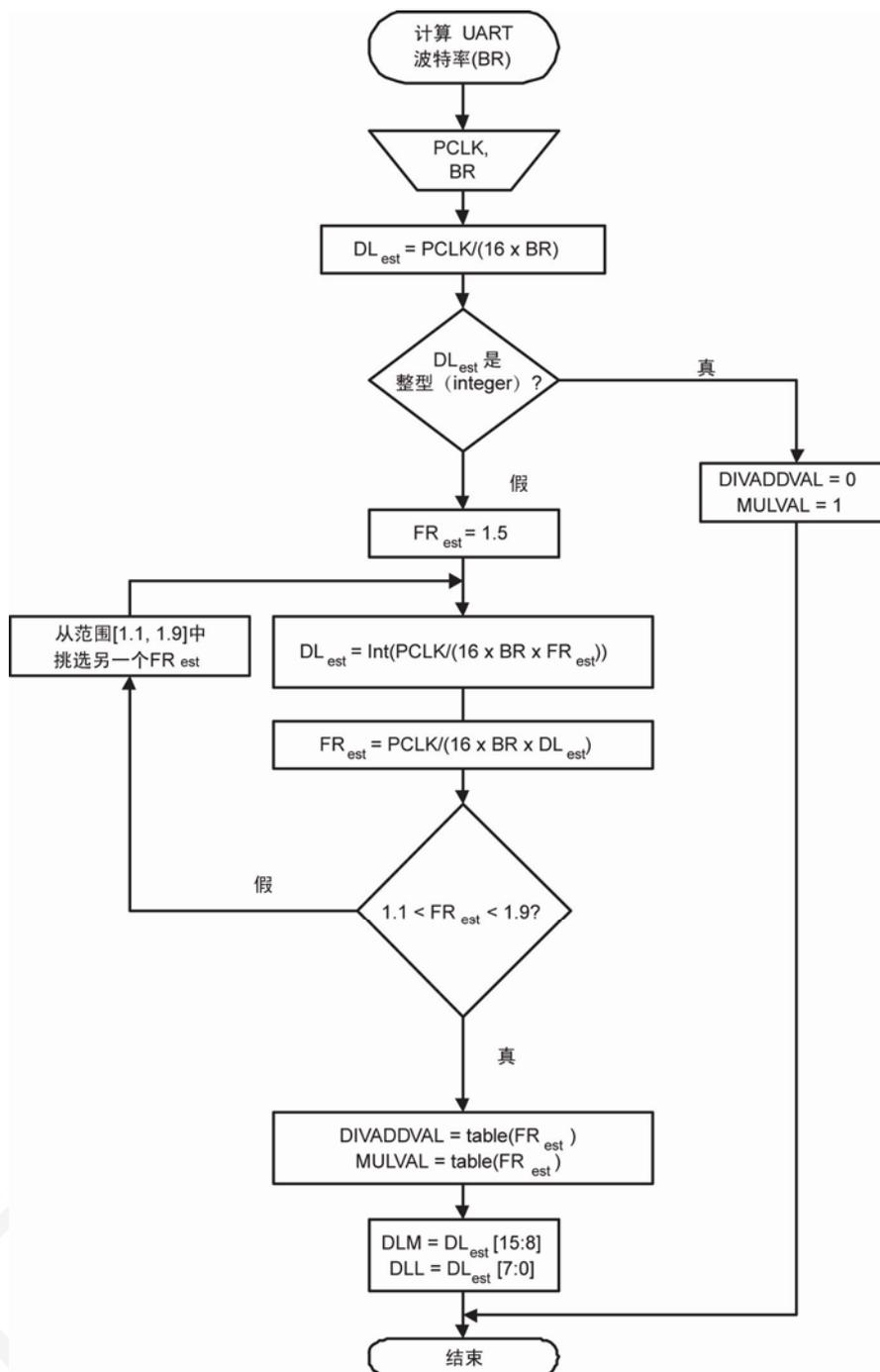


图 15.2 设置 UART1 分频器的算法

表 15.18 小数分频器设置查找表

FR	DivAddVal/ MulVal	FR	DivAddVal/ MulVal	FR	DivAddVal/ MulVal	FR	DivAddVal/ MulVal
1.000	0/1	1.250	1/4	1.500	1/2	1.750	3/4
1.067	1/15	1.267	4/15	1.533	8/15	1.769	10/13
1.071	1/14	1.273	3/11	1.538	7/13	1.778	7/9
1.077	1/13	1.286	2/7	1.545	6/11	1.786	11/14
1.083	1/12	1.300	3/10	1.556	5/9	1.800	4/5
1.091	1/11	1.308	4/13	1.571	4/7	1.818	9/11
1.100	1/10	1.333	1/3	1.583	7/12	1.833	5/6
1.111	1/9	1.357	5/14	1.600	3/5	1.846	11/13
1.125	1/8	1.364	4/11	1.615	8/13	1.857	6/7
1.133	2/15	1.375	3/8	1.625	5/8	1.867	13/15
1.143	1/7	1.385	5/13	1.636	7/11	1.875	7/8
1.154	2/13	1.400	2/5	1.643	9/14	1.889	8/9
1.167	1/6	1.417	5/12	1.667	2/3	1.900	9/10
1.182	2/11	1.429	3/7	1.692	9/13	1.909	10/11
1.200	1/5	1.444	4/9	1.700	7/10	1.917	11/12
1.214	3/14	1.455	5/11	1.714	5/7	1.923	12/13
1.222	2/9	1.462	6/13	1.727	8/11	1.929	13/14
1.231	3/13	1.467	7/15	1.733	11/15	1.933	14/15

(1) 范例 1: PCLK = 14.7456 MHz, BR = 9600

根据所提供的算法,  $D_{Lest} = PCLK / (16 \times BR) = 14.7456MHz / (16 \times 9600) = 96$ 。因为这里的  $D_{Lest}$  是一个整型数, 所以  $DIVADDVAL = 0$ ,  $MULVAL = 1$ ,  $DLM = 0$  且  $DLL = 96$ 。

(2) 范例 2: PCLK = 12 MHz, BR = 115200

根据所提供的算法  $D_{Lest} = PCLK / (16 \times BR) = 12MHz / (16 \times 115200) = 6.51$ 。该算式中的  $D_{Lest}$  并不是整型数, 因此下一步就要对 FR 参数进行估算。使用  $F_{Rest} = 1.5$  进行首次估算, 得到新的  $D_{Lest} = 4$ , 然后使用  $F_{Rest} = 1.628$  再进行计算。由于  $F_{Rest} = 1.628$  是在 1.1 到 1.9 的指定范围之内, 因此  $DIVADDVAL$  和  $MULVAL$  的值可通过附带的查找表获得。

在**错误!未找到引用源。**中, 最接近  $F_{Rest} = 1.628$  的值为  $FR = 1.625$ 。也就是说  $DIVADDVAL = 5$  而  $MULVAL = 8$ 。

基于这些查找结果, 建议 UART 设置为  $DLM = 0$ ,  $DLL = 4$ ,  $DIVADDVAL = 5$  和  $MULVAL = 8$ 。根据公式 12-2, UART 的波特率为 115384。该速率与原来指定的 115200 之间存在 0.16% 的相对误差。

### 15.4.13 UARTn发送使能寄存器 (U1TER - 0x4000 C030, U2TER - 0x4009 8030, U3TER - 0x4009C030)

$UnTER$  寄存器还可以实现软件流控制。当  $TxEn=1$  时, 只要数据可用,  $UARTn$  发送器就会一直发送数据。一旦  $TxEn$  变为 0,  $UARTn$  就会停止数据传输。

$UnTER$ 可实现软件和硬件流控制。当 $TxEn=1$ 时, 只要数据可用,  $UARTn$ 发送器就会一直发送数据。一旦 $TxEn$ 变为0,  $UARTn$ 就会停止数据传输。表 15.19描述了如何利用 $TxEn$ 位来实现软件流控制。

表 15.19 UART1 发送使能寄存器的位描述

位	符号	描述	复位值
6:0	-	保留。用户软件不应对其写入 1。从保留位读出的值未定义	NA
7	TxEN	该位为 1 时（复位后），一旦先前的数据都被发送出去后，写入 THR 的数据就会在 TxD 管脚上输出。如果在发送某字符时该位被清零，那么在将该字符发送完毕后就不再发送数据，直到该位被置“1”。也就是说，该位为 0 时会阻止字符从 THR 或 TX FIFO 传输到发送移位寄存器。当检测到硬件握手 TX-permit 信号（CTS）变为假时，或者在接收到 XOFF 字符（DC3）时，软件通过执行软件握手可以将该位清零。当检测到 TX-permit 信号变为真时，或者在接收到 XON 字符（DC1）时，软件又能将该位重新置位	1

**15.4.14 UARTn FIFO水平寄存器 (U0FIFOLVL - 0x4000 C058, U2FIFOLVL - 0x4009 8058, U3FIFOLVL - 0x4009 C058, 只读)**

UnFIFOLVL 寄存器是一个只读寄存器，允许软件读取当前的 FIFO 水平状态。发送和接收 FIFO 的水平均存放在该寄存器中。

表 15.20 UART1 FIFO 水平寄存器的位描述

位	符号	描述	复位值
3:0	RXFIFOLVL	反映 UART 接收 FIFO 的当前水平 0=空, 0xF=FIFO 为满	0x00
7:4	-	保留。从保留位读出的值未定义	NA
11:8	TXFIFOLVL	反映 UART 发送 FIFO 的当前水平 0=空, 0xF=FIFO 为满	0x00
31:12	-	保留。从保留位读出的值未定义	NA

**15.5 结构**

UART0/2/3 的结构如图 15.3所示。

APB 接口提供了 CPU 或主机与 UART1 之间的通信连接。

UARTn 接收器模块 (UnRX) 监控串行输入线 RXD1 的有效输入。UARTn RX 移位寄存器 (UnRSR) 通过 RXDn 接收有效字符。当 UnRSR 接收到一个有效字符时，它将该字符传送到 UARTn RX 缓冲寄存器 FIFO 中，等待 CPU 或主机通过通用主机接口进行访问。

UARTn 发送器模块 UnTX 接收 CPU 或主机写入的数据，并且将数据缓存到 UARTn TX 保持寄存器 FIFO (UnTHR)中。UARTn TX 移位寄存器 (UnTSR) 读出存放在 UnTHR 中的数据，并对数据进行汇编，通过串行输出管脚 TXDn 发送出去。

UARTn 波特率发生器模块 UnBRG 产生 UARTn TX 模块所使用的时序。UnBRG 时钟输入源为 APB 时钟 (PCLK)。主时钟与 UnDLL 和 UnDLM 寄存器中所指定的除数相除得到 Tx 模块所使用的时钟。该时钟为过采样时钟 NBAUDOUT 的 16 倍。

中断接口包括 UnIER 和 UnIIR 寄存器。中断接口接收若干个由 UnTX 和 UnRX 模块发出的单时钟宽度的使能信号。

UnTX 和 UnRX 所发送的状态信息会被存放到 UnLSR 中。UITX 和 UnRX 的控制信息会被存放到 UnLCR 中。

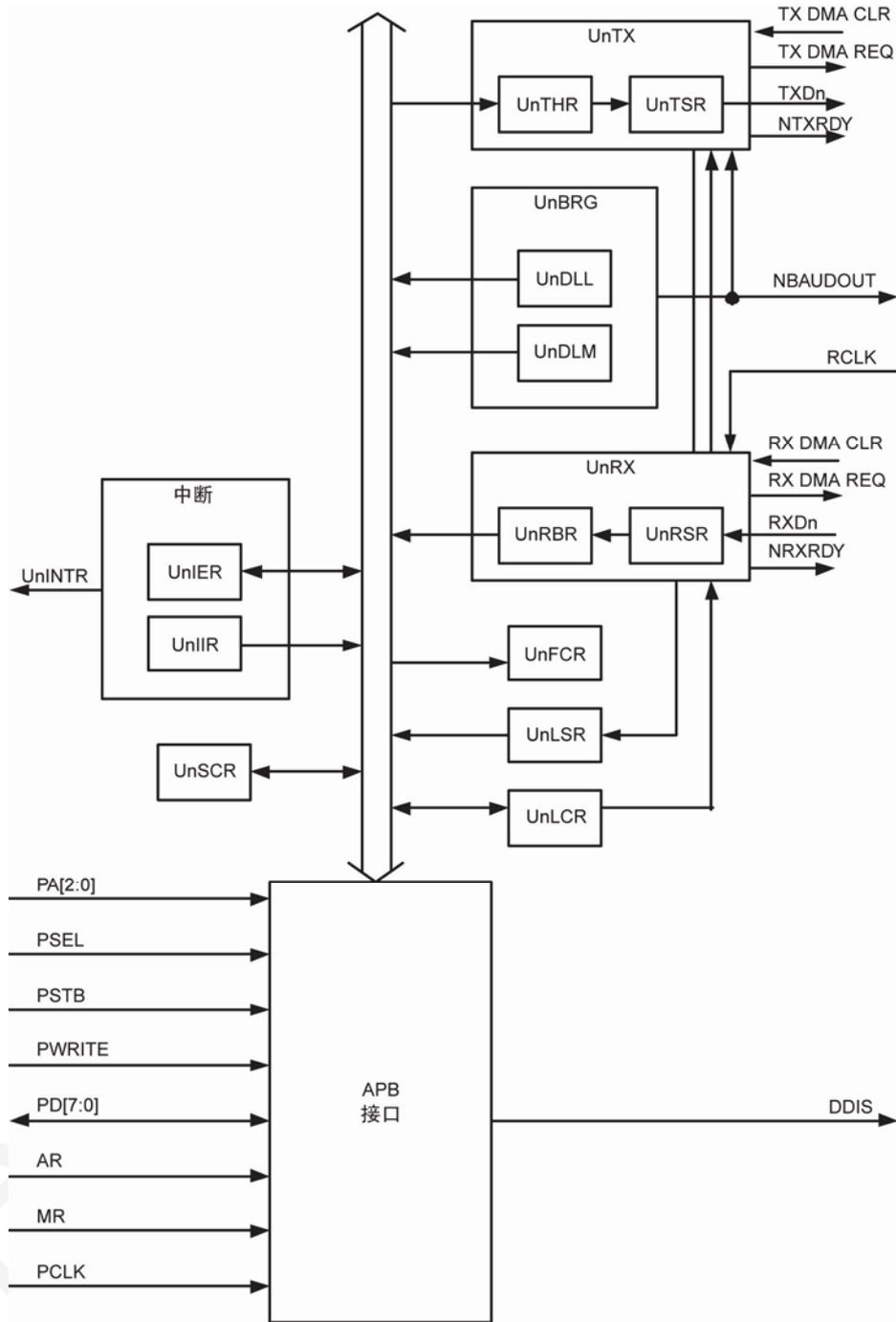


图 15.3 UART0/2/3 结构框图