



www.enjoylinux.cn

ARM综合实例



版权声明：本课件及其印刷物、视频的版权归成都国嵌信息技术有限公司所有，并保留所有权力：任何单位或个人未经成都国嵌信息技术有限公司书面授权，不得使用该课件及其印刷物、视频从事商业、教学活动。已经取得书面授权的，应在授权范围内使用，并注明“来源：国嵌”。违反上述声明者，我们将追究其法律责任。

Contents



www.enjoylinux.cn



S3c2440 GPIO

LED程序设计

ARM中断与异常

按键程序设计

串口程序设计

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



Contents



S3c2440 GPIO

LED程序设计

ARM中断与异常

按键程序设计

串口程序设计

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



S3c2440 GPIO



S3C2440有130个IO口，分为A~J共9组：GPA、GPB、GPC... GPJ，可以通过设置寄存器来确定某个IO口用于输入、输出还是其他特殊功能。比如可以设置GPH6做为一般的输入、输出端口，或者用于串口。对GPIO的操作是所有硬件操作的基础，由此扩展开来可以了解所有硬件的操作，这是嵌入式开发人员必须掌握的。

嵌入式Linux技术咨询QQ号：550491596
嵌入式Linux学习交流QQ群：65212116



操作GPIO



www.enjoylinux.cn

可以通过读取一个IO对应的寄存器来确定该IO口的电平是高还是低；还可以通过其他的寄存器来控制它。对于这几组GPIO，他们对应的寄存器是相似的；**GPxCON**用于选择引脚功能，**GPxDAT**用于读/写引脚数据；另外，**GPxUP**用于确定是否使用内部上拉电阻。其中x为A、B.....H、J等。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



GPxCON寄存器



www.enjoylinux.cn

从寄存器的名字可以看出，它属于配置（Configure）寄存器——选择IO的功能。其中PORTA与PORTB~PORTJ在功能选择方面有所不同。GPACON中每一位对应一个IO口（共23个）。当某位为0时，相应的IO为输出；当某位被置为1时，相应的引脚为地址线或用于地址控制。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



GPxCON寄存器



www.enjoylinux.cn

**PORTB~PORTJ在寄存器操作方面完全相同，
GPxCON中每两位控制一根引脚：00表示输入、01表示输出、10表示特殊功能、11保留不用。**

**嵌入式Linux技术咨询QQ号：550491596
嵌入式Linux学习交流QQ群：65212116**



GPxDAT寄存器



www.enjoylinux.cn

GPxDAT寄存器用于读/写引脚：当引脚被设为输入时，读此寄存器可知相应引脚电平状态是高还是低；当引脚被设为输出时，写此寄存器相应的位可令此引脚输出高电平或低电平。

嵌入式Linux技术咨询QQ号：550491596
嵌入式Linux学习交流QQ群：65212116



GPxUP寄存器



www.enjoylinux.cn

GPxUP: 某位为1时，相应引脚无内部上拉电阻；为0时相应的引脚使用内部上拉电阻。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



Contents



S3c2440 GPIO

LED程序设计

ARM中断与异常

按键程序设计

串口程序设计

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



LED程序设计



LED 是开发中最常用的状态指示设备，本开发板具有**4** 个用户可编程**LED**，它们直接与**CPU** 的**GPIO** 相连接，低电平有效(点亮)，详细的资源占用如下表：

	LED1	LED2	LED3	LED4
GPIO	GPB5	GPB6	GPB7	GPB8
可复用为	nXBACK	nXREQ	nXDACK1	nDREQ1
在原理图中的网络名	nLED_1	nLED_2	nLED_3	nLED_4

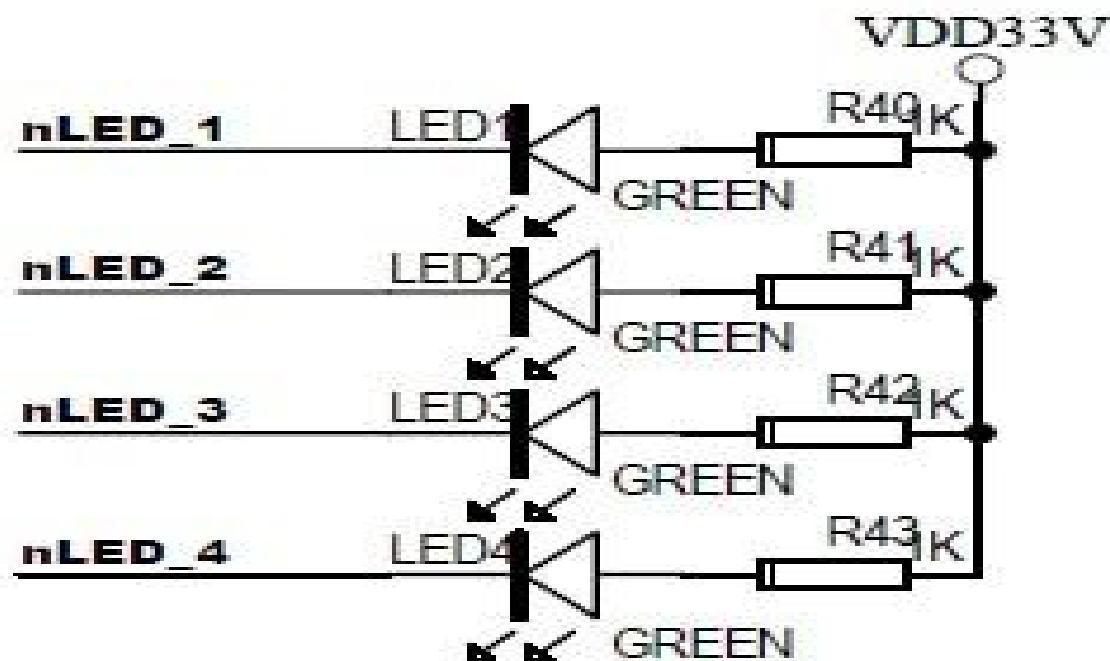
嵌入式Linux技术咨询QQ号：550491596
嵌入式Linux学习交流QQ群：65212116



LED原理图



www.enjoylinux.cn



嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



LED设计思想



先将相应的引脚设置为输出模式，然后相应的引脚输出低电平即可点亮LED灯。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



实验一



www.enjoylinux.cn

实现跑马灯程序

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



实验二



以二进制方式依次显示0-15

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



Contents



S3c2440 GPIO

LED程序设计

ARM中断与异常

按键程序设计

串口程序设计

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



ARM中断机制



CPU运行过程中，如何知道各类外设发生了某些不预期的事件，比如串口收到数据、按键按下等？

嵌入式Linux技术咨询QQ号：550491596
嵌入式Linux学习交流QQ群：65212116



ARM中断机制



www.enjoylinux.cn

1. 查询方式

程序不断地查询各设备的状态，并做出相应的反应。他实现比较简单，常用在比较单一的系统中，比如一个温控系统中可以实用查询方式不断检测温度的变化。缺点是占用**CPU**资源过高，不适应多任务的系统。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



ARM中断机制



www.enjoylinux.cn

2. 中断方式

当事件发生时，硬件会设置某个寄存器；
CPU再每执行完一个指令时，通过硬件查看这个寄存器，如果所关注的事件发生了，则中断当前程序流程，跳转到一个固定的地址处理这个事件，最后返回被中断的程序。它的实现相对复杂，但是效率较高，是常用的方法。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



ARM中断机制



中断处理流程:

(1) 中断控制器汇集各类外设发出的中断信号，然后告诉CPU。

(2) CPU保存当前程序的运行环境，然后调用中断服务程序（ISR）来处理中断。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



ARM中断机制



www.enjoylinux.cn

(3) 在ISR中通过读取中断控制寄存器、外设的相关寄存器来识别是哪个中断，并进行相应的处理。

(4) 清除中断：通过读写相关中断控制寄存器和外设相关寄存器来实现。

(5) 最后恢复被中断程序的环境，继续执行。

嵌入式Linux技术咨询QQ号：550491596

嵌入式Linux学习交流QQ群：65212116



中断控制寄存器



www.enjoylinux.cn

1. SUBSRCPND寄存器 (SUB SOURCE PENDING)
2. INTSUBMSK寄存器 (INTERRUPT SUBMSK)
3. SRCPND寄存器 (SOURCE PENDING)
4. INTMSK寄存器 (INTERRUPT MASK)
5. INTMOD寄存器 (INTERRUPT MODE)
6. PRIORITY寄存器
7. INTPND寄存器 (INTERRUPT PENDING)
8. INTOFFSET寄存器 (INTERRUPT OFFSET)

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



SUBSRCPND寄存器 (SUB SOURCE PENDING)



www.enjoylinux.cn

SUBSRCPND寄存器用来标示INT_RXD0, INT_TXD0等子中断是否发生。S3C2440有这样的15个子中断。每一位对应一个子中断。当这些子中断发生, 并且子中断未被屏蔽, 则它们当中的若干位则汇集在SRCPND上。例如: SUBSRCPND寄存器中的3个中断INT_RXD0、INT_TXD0、INT_ERR0, 只要有一个发生了, 且没有被屏蔽, 则SRCPND的INT_UART0位被置1。清中断时只要往SUBSRCPND寄存器中某位写入1即可。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



INTSUBMSK寄存器 (INTERRUPT SUBMSK)



**INTSUBMSK寄存器用来屏蔽
SUBSRCPND寄存器所标示的中断。
INTSUBMSK寄存器中某位被设为1时，对
应的中断被屏蔽。设为0时，表示中断允
许。**

**嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116**



SRCPND寄存器 (SOURCE PENDING)



www.enjoylinux.cn

SRCPND中每一位用来表示一个（或一类）中断是否已经发生。SRCPND寄存器操作与SUBSRCPND寄存器相似，若想清除某一位，往此位写入1。SRCPND各个位对应哪个中断，

参考S3C2440的数据手册

**嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116**



INTMSK寄存器 (INTERRUPT MASK)



INTMSK寄存器用来屏蔽SRCPND所标示的中断。INTMSK某位被设置为1时，对应的中断被屏蔽，设置为0时，中断允许。INTMSK只能屏蔽设为IRQ的中断，不能屏蔽设为FIQ的中断。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



INTMOD寄存器 (INTERRUPT MODE)



www.enjoylinux.cn

当INTMOD寄存器中某位被设置为1时，它对应的中断被设为FIQ，即此中断发生时，CPU将进入快速中断模式，这通常用来处理特别紧急的中断。当某位被置为0时，表示为IRQ中断。

注意：同一时间里，INTMOD上只能有一位被设置为1.

嵌入式Linux技术咨询QQ号：550491596
嵌入式Linux学习交流QQ群：65212116



INTPND寄存器 (INTERRUPT PENDING)



www.enjoylinux.cn

经过中断优先级仲裁器选出优先级高的中断后，这个中断在INTPND寄存器中的相应位被置1，随后，CPU将进入中断处理模式处理它。同一时间内，此寄存器只有一位被置1；在ISR中可以根据这个位确定是那个中断。清除该中断时，往这个位写入1。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



INTOFFSET寄存器

这个寄存器用来表示INTPND寄存器中哪位被置1了，即INTPND寄存器中位[x]为1时，INTOFFSET寄存器的值为x(x为0~31)。在清除SRCPND、INTPND寄存器时，INTOFFSET寄存器被自动清除。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



Contents



S3c2440 GPIO

LED程序设计

ARM中断与异常

按键程序设计

串口程序设计

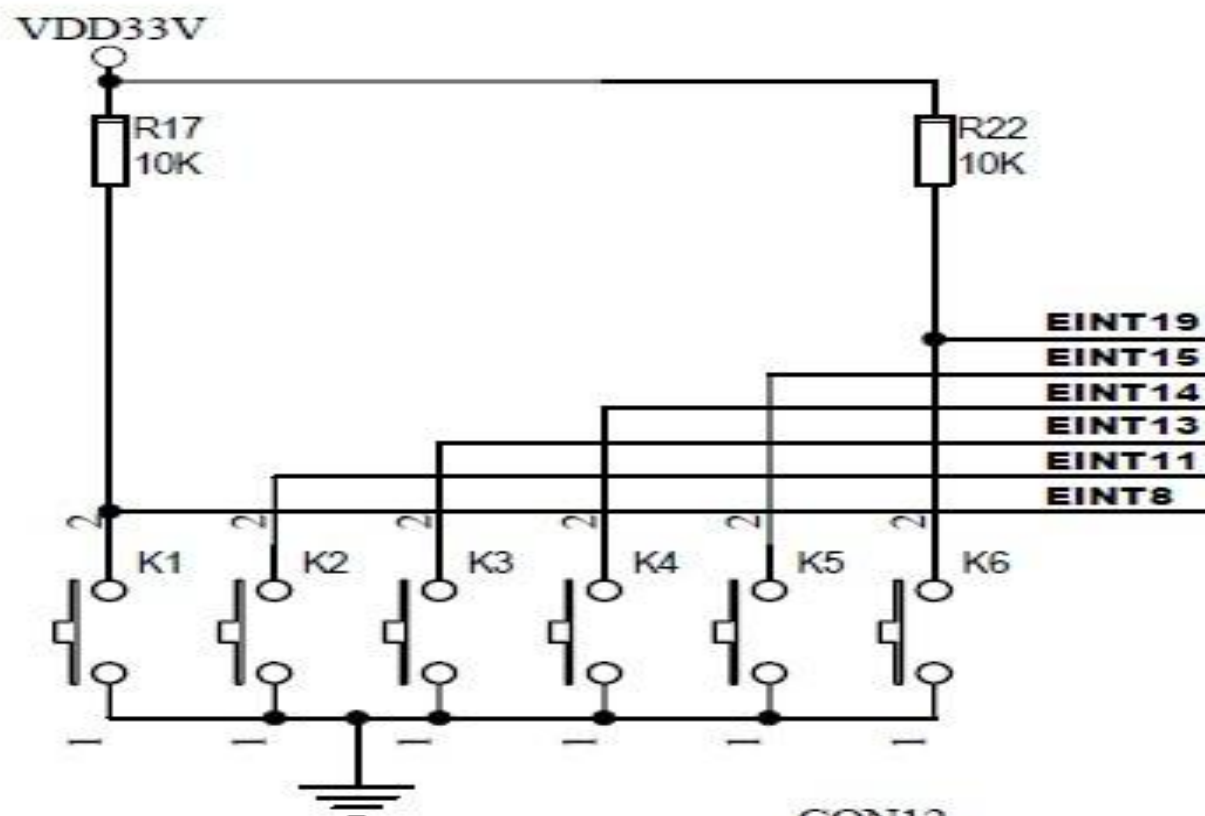
嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



按键原理图



www.enjoylinux.cn



嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



用户按键



www.enjoylinux.cn

本开发板总共有 6 个用户测试用按键，它们均从 CPU 中断引脚直接引出，属于低电平触发，这些引脚也可以复用为 GPIO 和特殊功能口，为了用户把它们引出作为其他用途，这 6 个引脚也通过 CON12 引出，6 个按键和 CON12 的定义如下：

	K1	K2	K4	K4	K5	K6
对应的中断	EINT8	EINT11	EINT13	EINT14	EINT15	EINT19
复用的 GPIO	GPG0	GPG3	GPG5	GPG6	GPG7	GPG11
特殊功能口	无	nSS1	SPIMISO1	SPIMOSI1	SPICLK1	TCLK1
对应的 CON12 引脚	CON12.1	CON12.2	CON12.3	CON12.4	CON12.5	CON12.6

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



实验一



www.enjoylinux.cn

使用 **中断方式** 设计一个按键程序，按一次点亮一个**LED**

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



实验二



www.enjoylinux.cn

使用**查询方式**设计一个按键程序，
按一次点亮一个LED

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



Contents



www.enjoylinux.cn



S3c2440 GPIO

LED程序设计

ARM中断与异常

按键程序设计

串口程序设计

嵌入式Linux技术咨询QQ号: 550491596

嵌入式Linux学习交流QQ群: 65212116



UART基本原理



通用异步收发器简称**UART**，即“**Universal Asynchronous Receiver Transmitter**”，它用来传输串行数据：发送数据时**CPU**将并行数据写入**UART**，**UART**按照一定的格式在一根电线上串行发出；接收数据时，**UART**检测另一根电线上的信号，将串行数据放在缓冲区中，**CPU**可读取**UART**获得的这些数据。

嵌入式Linux技术咨询QQ号：550491596
嵌入式Linux学习交流QQ群：65212116



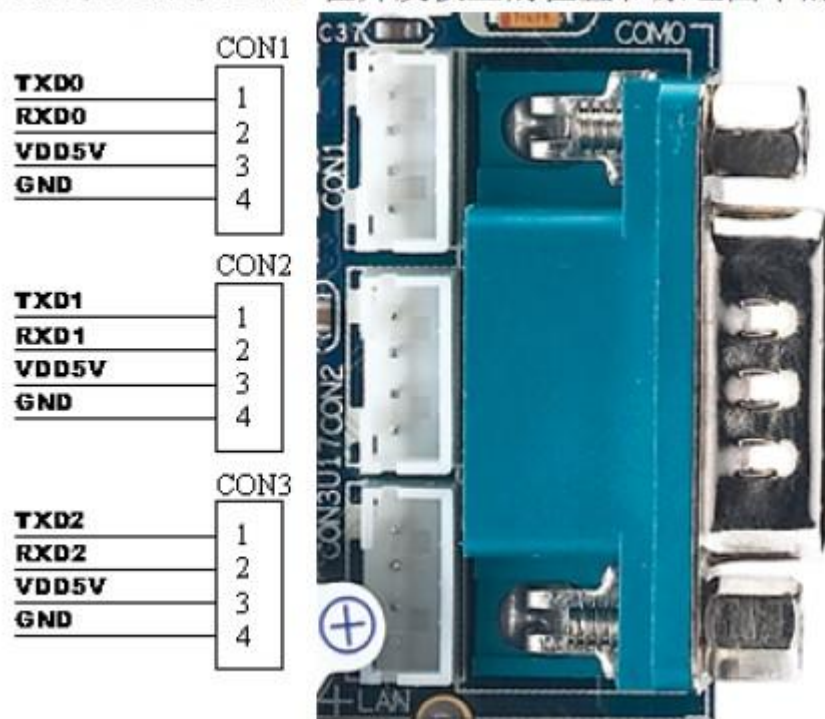
UART基本原理



www.enjoylinux.cn

mini2440的UART:

CON1, CON2, CON3 在开发板上的位置和原理图中的连接定义对应关系如下图所示。



嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



UART初始化



www.enjoylinux.cn

在使用**UART**之前要设置波特率、传输格式（有多少个数据位、是否使用校验位、是奇校验还是偶校验、有多少个停止位、是否使用流量控制）；对于**S3C2440**，还要选择所涉及管脚为**UART**功能、选择**UART**通道的工作模式为中断模式或者**DMA**模式。设置好之后往某个寄存器写入数据即可发送，读某个寄存器即可得到接收的数据，可以通过查询状态寄存器或者设置中断模式来获知数据是否发送或接收完毕。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



UART寄存器



www.enjoylinux.cn

- (1) **UBRDIVn**寄存器
- (2) **ULCONn**寄存器
- (3) **UCONn**寄存器
- (4) **UFCON**寄存器
- (5) **UMCONn**寄存器
- (6) **UTRSTATn**寄存器
- (7) **UERSTATn**寄存器
- (8) **UTXHn**寄存器
- (9) **URXHn**寄存器

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



UBRDIVn寄存器



www.enjoylinux.cn

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 中的值用于决定波特率，如下：

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

例如如果波特率为115200bps 且UART 时钟为40MHz,

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

嵌入式Linux技术咨询QQ号: 550491596

嵌入式Linux学习交流QQ群: 65212116



ULCON寄存器



www.enjoylinux.cn

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

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



UCONn寄存器



www.enjoylinux.cn

该寄存器用于选择UART时钟源、设置
UART中断方式等。详见S3C2440的手册。

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

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



UFCONn寄存器



www.enjoylinux.cn

该寄存器用于设置是否使用**FIFO**，设置各个**FIFO**的触发深度，本次试验中都未使用该寄存器。如果不使用该寄存器，则触发深度默认为1。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



UTRSTATn寄存器



www.enjoylinux.cn

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

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



UART收发缓冲寄存器



www.enjoylinux.cn

UTXHn寄存器:

CPU将数据写入这个寄存器，UART会将它保存到缓冲区中，并自动发出去。

URXHn寄存器:

当UART收到数据时，CPU读取这个寄存器，即可获得数据。

嵌入式Linux技术咨询QQ号: 550491596

嵌入式Linux学习交流QQ群: 65212116



实验一



www.enjoylinux.cn

编写串口收发程序：首先向串口发送十次“hello world”，然后等待键盘输入字符‘R’，然后蜂鸣器连续响5次。

嵌入式Linux技术咨询QQ号：550491596
嵌入式Linux学习交流QQ群：65212116



实验二



www.enjoylinux.cn

改写实验一，实现串口的中断方式
接收数据，然后将接收的数据显示
在超级终端上。

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116

