青风带你玩蓝牙 nRF51822 系列教程

----------作者：青风

出品论坛：　www.qfv8.com　青风电子社区

作者： 青风
出品论坛: www.qfv8.com
淘宝店：http://qfv5.taobao.com
QQ 技术群：346518370
硬件平台：青云 QY-nRF51822 开发板

2.3 按键中断

下面我就来分别介绍下 nRF51822 的按键中断控制方式。

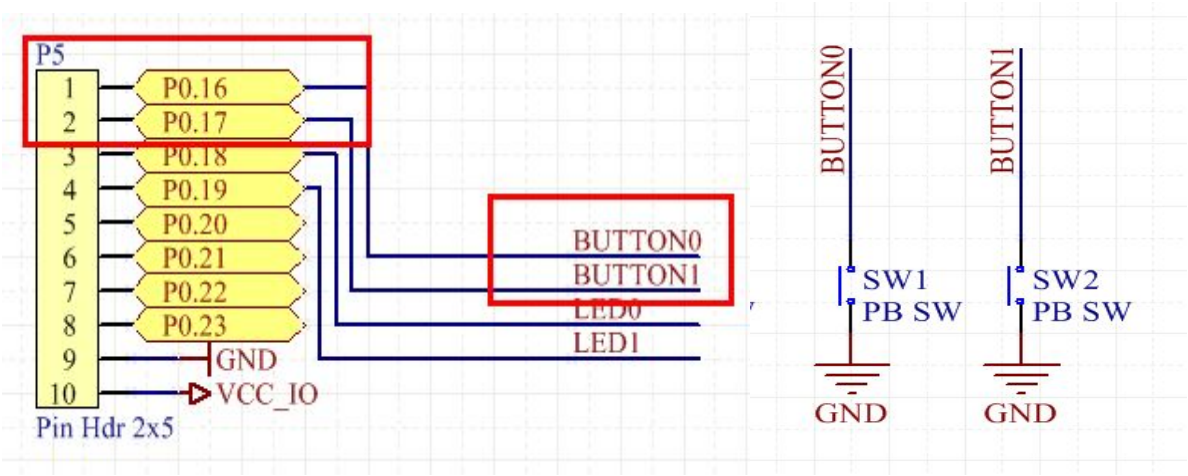
中断控制的效率很高，一旦系统 IO 口出现上升沿或者下降沿电平就会触发执行中断内的程序，这样可以大大节省了 cpu 的占有率。中断是指由于接收到来自外围硬件（相对于中央处理器和内存）的异步信号或来自软件的同步信号，而进行相应的硬件／软件处理。发出这样的信号称为进行中断请求（interrupt request，IRQ）。硬件中断导致处理器通过一个上下文切换（context switch）来保存执行状态（以程序计数器和程序状态字等寄存器信息为主）；软件中断则通常作为 CPU 指令集中的一个指令，以可编程的方式直接指示这种上下文切换，并将处理导向一段中断处理代码。中断在计算机多任务处理，尤其是实时系统中尤为有用，这样的系统，包括运行于其上的操作系统，也被称为“中断驱动的”。简单的来说就比如某个人正在做某事，突然来了个电话，他就要停下手中的事情去接电话，中断相当于这个电话。触发中断后跳出原来运行的程序去执行中断处理。

在使用 nRF51822 完成中断时，需要设置如下几个地方：

第一：中断嵌套的设置。
第二：外部 GPIO 中断函数的设置。

2.3.1 硬件准备：

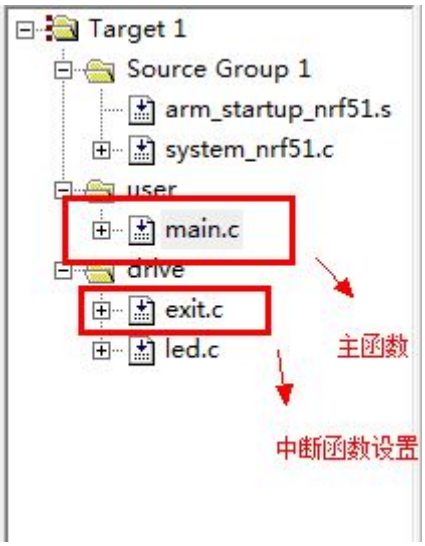
在青云 nRF51822 豪华开发板上设置了 2 个用户按键如下图所示：



SW1 和 SW2 分别和 P0.16 和 P0.17 两个 IO 管脚相连。当 IO 管脚为低的时候可以判断管脚已经按下。通过 key 的按下来控制 led 的亮灭。硬件上设计是比较简单的，这个普通的 MCU 的用法一致。

2.3.2 软件准备：

在代码文件中，实验三建立了一个演示历程，我们打开看看需要那些库文件。打开 user 文件夹中的 key 工程：



如上图所示：码农只需要自己编写红色框框里的两个文件就 OK 了,因为采用子函数的方式其中 led.c 在上一节控制 LED 灯的时候已经写好，现在我们就来讨论下如何编写 exit.c 这个驱动子文件。

exit.c 文件主要是要起到两个作用：第一：初始化开发板上的按键中断。第二：编写中断执行代码。完成这两个功能够就可以在 main.c 文件中直接调用本驱动了。

首先来认识下如何进行中断设置。那么先来看 GPIOET 的功能说明：

A task can be used for performing the following write operations to a pin:

任务模式

- Set
- Clear
- Toggle

An event can be generated from any of the following input pins using the GPIO DETECT signal:

事件模式

- Rising edge
- Falling edge
- Any change

实际上就两种模式，一个任务模式，一个事件模式。其中任务模式作为输出使用，而事件模式就作为中断触发使用。

任务模式有三种状态：置位，清零，翻转。事件模式三种触发状态：上升沿触发，下降沿触发，任意变化触发。

整个 **GPIOTE** 寄存器的个数也是非常少的，如下图所示：

Registers	Offset	Description	
TASKS			
OUT[0]	0x000	Task for writing to pin specified by PSEL in CONFIG[0].	
OUT[1]	0x004	Task for writing to pin specified by PSEL in CONFIG[1].	→ 任务模式配置
OUT[2]	0x008	Task for writing to pin specified by PSEL in CONFIG[2].	
OUT[3]	0x00C	Task for writing to pin specified by PSEL in CONFIG[3].	
EVENTS			
IN[0]	0x100	Event generated from pin specified by PSEL in CONFIG[0].	
IN[1]	0x104	Event generated from pin specified by PSEL in CONFIG[1].	→ 事件模式配置通道
IN[2]	0x108	Event generated from pin specified by PSEL in CONFIG[2].	
IN[3]	0x10C	Event generated from pin specified by PSEL in CONFIG[3].	
PORT	0x17C	Event generate from multiple input pins.	
REGISTERS			
INTENSET	0x304	Interrupt enable set register.	→ 使能中断寄存器
INTENCLR	0x308	Interrupt enable clear register.	
CONFIG[0]	0x510	Configuration for OUT[0] task and IN[0] event.	
CONFIG[1]	0x514	Configuration for OUT[1] task and IN[1] event.	→ 模式详细配置寄存器
CONFIG[2]	0x518	Configuration for OUT[2] task and IN[2] event.	
CONFIG[3]	0x51C	Configuration for OUT[3] task and IN[3] event.	

我们使用到了按键中断，实际上使用到了事件模式，下面将主要讨论这个模式，任务模式后面有专门的历程进行讨论。在 **CONFIG** 这个寄存器里详细的进行了事件模式的配置，如下图所示，三个红色框框里的寄存器位我们需要进行配置：

Bit number			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID (Field ID)			-	-	-	-	-	-	-	-	-	-	-	D	-	-	C	C	-	-	-	B	B	B	B	B	-	-	-	-	-	-	A	A
Reset value			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	RW	Field	Value ID		Value																													
A	RW	MODE																																
			DISABLED		0																													
			EVENT		1																													
			TASK		3																													
B	RW	PSEL			[0..31]																													
C	RW	POLARITY																																
			LOTOHI		1																													
			HITOLO		2																													
			TOGGLE		3																													
D	RW	OUTINIT																																
			LOW																															
			HIGH																															

ID	RW	Field	Value ID	Value	Description
A	RW	MODE			Mode
			DISABLED	0	Disabled. Pin specified by PSEL will not be acquired by the GPIOTE module.
			EVENT	1	Event mode. The pin specified by PSEL will be configured as an input and the IN[n] event will be generated if operation specified in POLARITY occurs on the pin.
			TASK	3	Task mode. The pin specified by PSEL will be configured as an output and triggering the OUT[n] task will perform the operation specified by POLARITY on the pin. When enabled as a task the GPIOTE module will acquire the pin and the pin can no longer be written as a regular output pin from the GPIO module.
B	RW	PSEL		[0..31]	Pin number associated with OUT[n] task and IN[n] event.
C	RW	POLARITY			When in task mode: Operation to be performed on output when OUT[n] task is triggered. When In event mode: Operation on input that shall trigger IN[n] event.
			LOTOHI	1	Task mode: Set pin from OUT[n] task. Event mode: Generate IN[n] event when rising edge on pin.
			HITOLO	2	Task mode: Clear pin from OUT[n] task. Event mode: Generate IN[n] event when falling edge on pin.
			TOGGLE	3	Task mode: Toggle pin from OUT[n]. Event mode: Generate IN[n] when any change on pin.
D	RW	OUTINIT			When in task mode: Initial value of the output when the GPIOTE channel is configured.
					When in event mode: No effect.
			LOW		Task mode: Initial value of pin before task triggering is low.
			HIGH		Task mode: Initial value of pin before task triggering is high.

下面看看代码：

```
01.  NRF_GPIOTE->CONFIG[0] =
02. (GPIOTE_CONFIG_POLARITY_HiToLo << GPIOTE_CONFIG_POLARITY_Pos)
03.                                  | (16 << GPIOTE_CONFIG_PSEL_Pos)
04.                       | (GPIOTE_CONFIG_MODE_Event << GPIOTE_CONFIG_MODE_Pos);
05. //中断配置（详细说明请参看青风教程）
```

上面一段代码的编写严格按照了寄存器要求进行，首先是 MODE，也就是模式设置，我们设置成事件模式。PSEL 设置对应的管脚，我们选择了 SW1 管脚 P0.16 作为触发管脚，POLARIY 极性设置为下降沿触发。

设置好了工作方式后，我们就需要进行中断的使能了：

```
06.    NVIC_EnableIRQ(GPIOTE_IRQn);//中断嵌套设置
07.  NRF_GPIOTE->INTENSET    = GPIOTE_INTENSET_IN0_Set <<
     GPIOTE_INTENSET_IN0_Pos;//  使能中断类型:
```

上面的任务基本上就可以把 IO 管脚中断配置好了，如果你搞清楚寄存器，那么这个配置也是十分简单的。

中断函数的设计，主要任务就是要求判断中断发生后，要对 LED 灯进行翻转，当然你可以加入其它更多的任务。

```
08.   void GPIOTE_IRQHandler(void)
09.  {
10.
```

```
11.       if ((NRF_GPIOTE->EVENTS_IN[0] == 1) &&
12.             (NRF_GPIOTE->INTENSET & GPIOTE_INTENSET_IN0_Msk))
13.       {
14.             NRF_GPIOTE->EVENTS_IN[0] = 0; //中断事件清零.
15.       }
16.      LED_Toggle();//led 灯翻转
17. }
```

那么主函数就是十分的简单了，直接调用我们写好的驱动函数，判断按键按下后就可以翻转 IO 口，LED 灯指示相应的变化。函数如下所示：

```
18.   /******************** (C) COPYRIGHT 2014  青风电子  ********************
19.   * 文件名    ：main
20.   * 描述      ：
21.   * 实验平台：青云 nRF51822 蓝牙开发板
22.   * 描述      ：按键中断
23.   * 作者      ：青风
24.   * 店铺      ：qfv5.taobao.com
25.  *********************************************************************************/
26.  #include "nrf51.h"
27.  #include "nrf_gpio.h"
28.  #include "exit.h"
29.  #include "led.h"
30.
31.  int main(void)
32.  {
33.   LED_Init();
34.       LED_Open();
35.   /*config key*/
36.   EXIT_KEY_Init();
37.   while(1)
38.   {
39.         }
40.  }
```

实验下载到青云 nRF51822 开发板后的实验现象如下：

如图所示，按下复位键后，上方的用户 led 灯不停闪烁