



PAN2416 低功耗调试说明

Rev 1.1

PANCHIP

Panchip Microelectronics

www.panchip.com

修订历史

版本	修订日期	描述
V1.0	2021-11-11	初始版本创建
V1.1	2024-12-19	添加 RC0 软硬件使用参考

版权所有©

上海磐启微电子有限公司

本资料内容为上海磐启微电子有限公司在现有数据资料基础上慎重编制而成，本资料中所记载的实例以正确的使用方法和标准操作为前提，使用方在应用该等实例时需充分考虑外部诸条件，上海磐启微电子有限公司不担保或确认该等实例在使用方的适用性、适当性或完整性，上海磐启微电子有限公司亦不对使用方因使用本资料所有内容而可能或已经带来的风险或后果承担任何法律责任。基于使本资料的内容更加完善等原因，上海磐启微电子有限公司保留未经预告的修改权，使用方如需获得最新的产品信息，请随时与上海磐启微电子有限公司联系。



目录

第 1 章 概述	4
第 2 章 软件设计参考	5
唤醒源配置	5
软件其它配置	6
软件实现流程	7
第 3 章 硬件设计参考	8

PANCHIP



第 1 章 概述

PAN2416AV、PAN2416AF、PAN2013、PAN2010 均适用于该份操作说明。

PANCHIP

第2章 软件设计参考

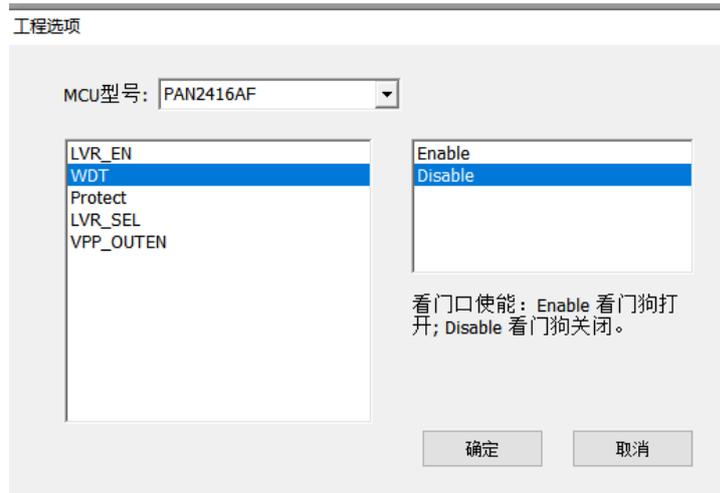
唤醒源配置

1、看门狗唤醒，看门狗唤醒电流会比中断唤醒高 3~5 μ A 左右，具体以实际板子为准。芯片休眠时，看门狗溢出起唤醒作用；芯片工作时，看门狗溢出起复位作用。IDE 里面可以配置 2 种看门狗唤醒方式：

- ①软件里面配置看门狗，可以配置唤醒时间；otp 为准，唤醒时间为 n（n=1、2、4、8、16、32、64、128）*18ms；仿真板/仿真芯片时间会有差异。

```
8
9
10 void WDT_enable(WDT_DIVIDE_RATIO_e WDT_DIVIDE_RATIO)
11 {
12     OPTION_REG &= 0xf0;
13     OPTION_REG |= WDT_DIVIDE_RATIO;
14     WDTCON |= 0x01;           //使能WDT
15     asm("CLRWDTC");
16 }
17
18 typedef enum{
19     WDT_DIVIDE_RATIO_1 = 8+0,           //18*1ms
20     WDT_DIVIDE_RATIO_2 = 8+1,
21     WDT_DIVIDE_RATIO_4 = 8+2,
22     WDT_DIVIDE_RATIO_8 = 8+3,
23     WDT_DIVIDE_RATIO_16 = 8+4,
24     WDT_DIVIDE_RATIO_32 = 8+5,
25     WDT_DIVIDE_RATIO_64 = 8+6,
26     WDT_DIVIDE_RATIO_128 = 8+7,
27 }WDT_DIVIDE_RATIO_e;
```

- ②芯片配置里面配置看门狗使能，唤醒时间固定为 1.2S，不可配置时间。



2、中断唤醒，通过 PORTB 中断唤醒。

```

/*****
* icob_value bit1-bit7,置1, 使能相应口电平变化中断
*****/
void PORTB_INT_Init(unsigned char icob_value)
{
    OPTION_REG &= ~0x80;    //RBP0;开启PORTB上拉功能
    ANSELH &= ~icob_value; //数字脚
    TRISB |= icob_value;   //输入
    WPUB  |= icob_value;   //弱上拉
    IOCB  |= icob_value;   //中断使能
    last_PORTB_value = PORTB;
    INTCON |= 0x08;        //RBIE = 1; 开RB中断
}

```

软件其它配置

- 1、芯片管脚若未使用到，都配置成输出低。
- 2、芯片使用 RC0 作为 IO，需要在芯片配置项里面把 VPP_OUTEN 使能。
- 3、RC0 作为 IO 输入口时，芯片内部没有上拉功能。
- 4、RC0 作为 IO 输出口时，芯片只能开漏输出，若要输出高则需要外挂上拉电阻。
- 5、RC0 使用后低功耗电流会到 10ua 左右,若需要降低功耗，需要加 MOS 对 RC0 进行隔离；未使用 RC0&使用看门狗唤醒，低功耗电流 6ua 左右；未使用 RC0&未使用看门狗&使用 IO 中断唤醒，低功耗电流 3ua 左右。

6、低功耗模式下，芯片设置的 LVR 不会生效，在快速上下电过程中，可能无法实现芯片正常复位，此时射频模块会复位，MCU 会在唤醒后继续运行，需要对射频重新进行一遍初始化。

软件实现流程

休眠流程：

RF 休眠->设置休眠唤醒方式（提前设置过则忽略）->MCU 进入休眠

```
while(1)
{
    //设置进入休眠
    //优先设置RF进入休眠
    RF_Sleep();
    //配置唤醒源 看门狗唤醒/PB中断唤醒

#ifdef weekup_wdt
    WDT_enable(WDT_DIVIDE_RATIO_128);
#else
    PORTB_INT_Init(0x02);
#endif

    system_to_sleep();
    //唤醒后由此处继续执行
    RC1^=1;
    RF_Set_TxMode();
    RF_TxData(tx_rx_msg, PAYLOAD_WIDTH);
}
```

唤醒流程：设备进入 sleep->看门狗溢出/PORTB 中断响应->MCU 唤醒，从 sleep 后继续执行

参考 PAN2416_SDK_V1.4 中的软件设计参考 test_sleep

名称	修改日期	类型	大小
test_adc	2022/1/4 17:04	文件夹	
test_gpio&roll_code	2021/11/11 9:49	文件夹	
test_gpio_Interrupt	2021/11/11 9:49	文件夹	
test_pwm	2021/12/7 16:18	文件夹	
test_RF	2021/11/23 13:52	文件夹	
test_RF_enhance	2021/11/11 9:50	文件夹	
test_sleep	2022/1/5 10:15	文件夹	
test_timer	2021/12/22 15:47	文件夹	
test_wdt	2021/11/11 9:50	文件夹	

第 3 章 硬件设计参考

- 1、低功耗应用尽量规避 RC0 的使用,RC0 未使用情况下, RC0 硬件接地。
- 2、外部中断唤醒, 只有 RB (RB1~RB7) 口才能唤醒, 其它 IO 无法做唤醒源。
- 3、低功耗模式下, LVR 会无效, 所以低功耗情况下可能会存在重新上下电后芯片不会触发复位, 需要加入放电电阻, 或者软件部分唤醒后对射频重新初始化。

