应用指南

32-bit Cortex™-M0 MCU
NuMicro® Family

电源管理及断电/空闲模式

目录

1 简介

空闲/断电模式下功耗是最低的，当系统不工作或者等待一个外部中断/事件的时候，软件使系统进入空闲/断电模式，在这种状态下，时钟源停止了所有的外围设备也进入睡眠模式。当检测到外部中断/事件的时候，系统退出空闲/断电模式。

本文件说明了压缩文件***AN_1001_EN.ZIP***中的示例程序“Smpl_PowerManagement”，这个程序描述了怎样进入系统空闲/断电模式以及如何从空闲/断电模式唤醒系统。

断电模式控制表

寄存器/指令 模式	PWR_DOWN_EN	PD_WAIT_CPU	CPU run WFE/WFI instruction	Clock Gating
正常运行模式	1’b0	1’b0	NO	根据时钟选择寄存器打开所有时钟
空闲模式 **(CPU进入睡眠)**	1’b0	1’b0	YES	只有CPU时钟打开
断电模式	1’b1	1’b0	NO	除10K/32K时钟外所有时钟关闭，一些终端设备时钟仍然工作，例如：RTC/WDT/Timer/PWM/ADC
断电模式 **(CPU进入深度睡眠)**	1’b1	1’b1	YES	除10K/32K时钟外所有时钟关闭，一些终端设备时钟仍然工作，例如：RTC/WDT/Timer/PWM/ADC

1.1 性能

- 支持禁止选定的IP时钟以节省能量
- 支持系统空闲模式，可以通过RTC/GPIO/UART/USB/CAN 中断事件唤醒
- 支持系统断电模式，可以通过RTC/GPIO/UART/USB/CAN 中断事件唤醒

2 代码部分

大部分函数定义在Smpl_PowerManagement.c文件中

详细调用顺序参考调用顺序部分

2.1 主函数

在主函数中，我们设置外部12MHz时钟为系统时钟以方便后续测设。硬件初始化完成后，打开UART0并配置波特率为115200来输出调试信息，然后你就可以输入你要测试的内容了。

```
// Poll and handle USB events.
DrvSYS_SetOscCtrl(E_SYS_XTL12M, 1);
.....
DrvUART_Open(UART_PORT0, &sParam);
……
While (i32Loop)
{
      ……
          printf("| [1] IP Clock Control Test              |\n");
          printf("| [2] Idle Mode Test                     |\n");
          printf("| [3] Power Down Mode Test               |\n");
          switch (u8Item)
          {
                    case '0':
                              i32Loop = 0;
                              break;
                    case '1':
                              IpClockCtrlTest();
                              break;
                    case '2':
                              PowerDownAndIdleModeTest(0);
                              break;
                    case '3':
                              PowerDownAndIdleModeTest(1);
                              break;
          }
}
```

2.2 IpClockCtrlTest 函数

这个函数中，用户可以通过输入字符 ‘a’ ~ ‘z’ 来使能或禁止选定的单个模块的时钟，字符 ‘1’来使能或禁止所有终端模块的时钟

```
……
if (u8Item == '1')
{
        if (u32IPClkCfg == ALL_IP_CLOCK)
                u32IPClkCfg = 0;
        else
                u32IPClkCfg = ALL_IP_CLOCK;
        printf("\r%s all IP ...\n\n", (u32IPClkCfg==0)? "Disable":"Enable");
}else
if ((u8Item >= 'a') && (u8Item <= 'z'))
{
        u8IPIndex = u8Item - 'a';
        u32IPClkCfg = u32IPClkCfg ^ (1<<u8IPIndex);
        printf("\rIndex-[%c] IP is %s ...\n\n", u8Item,
                 ((u32IPClkCfg&(1<<u8IPIndex))==0)? "Disable":"Enable");
}else
{
        continue;
}
 ……
/* Apply the stting */
DrvSYS_SetIPClock(E_SYS_WD_CLK,      ((u32IPClkCfg&WD_CLOCK)? 1:0));
DrvSYS_SetIPClock(E_SYS_RTC_CLK,    ((u32IPClkCfg&RTC_CLOCK)? 1:0));
DrvSYS_SetIPClock(E_SYS_TMR0_CLK,  ((u32IPClkCfg&TMR0_CLOCK)? 1:0));
DrvSYS_SetIPClock(E_SYS_TMR1_CLK,  ((u32IPClkCfg&TMR1_CLOCK)? 1:0));
DrvSYS_SetIPClock(E_SYS_TMR2_CLK,  ((u32IPClkCfg&TMR2_CLOCK)? 1:0));
……
```

2.3 PowerDownAndIdleModeTest 函数

PowerDownAndIdleModeTest 函数的参数是0表示空闲模式测试，是1代表断电模式测试，选定了测试模式后，用户还必须输入 '1' ~ '5'来选择唤醒类型RTC, GPIO, UART, USB 或CAN中断.

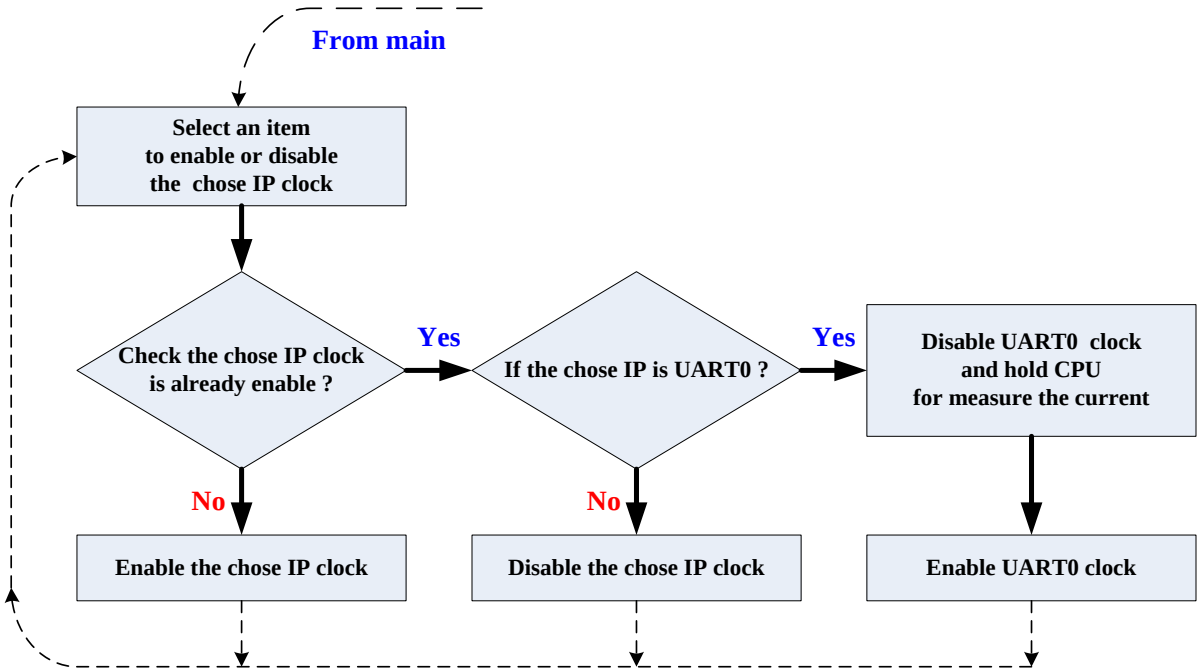
```
……
printf("| [1] RTC wake up Test                    |\n");
printf("| [2] GPIO wake up Test(GPA1 toggle)      |\n");
printf("| [3] UART wake up Test(GPB3 toggle)      |\n");
printf("| [4] USB wake up Test                    |\n");
printf("| [5] CAN0 wake up Test                   |\n");
……
switch (u8Item)
{
        case '1':
                if (InitRTCWakeupFunction() == FALSE)
                        continue;
                break;
        case '2':
                if (InitGPIOWakeupFunction() == FALSE)
                        continue;
                break;
        case '3':
                if (InitUARTWakeupFunction() == FALSE)
                        continue;
                break;
        case '4':
                if (InitUSBWakeupFunction() == FALSE)
                        continue;
                break;
        case '5':
                if (InitCANWakeupFunction() == FALSE)
                        continue;
                break;
        case '0':
```

```
                return;
        default :
                continue;
}
……
if (u8Type == 1)
{
        /* For PowerDown Mode */
        UNLOCKREG();

        DrvSYS_SetPowerDownWaitCPU(1);

        /* Disable crystal to enter power down */
        DrvSYS_EnablePowerDown(1);

        /* wait for interrupt, enter in Power Down mode */
        __WFI();
}else
if (u8Type == 0)
{
        /* For IDLE Mode */
        UNLOCKREG();

        DrvSYS_SetPowerDownWaitCPU(0);

        DrvSYS_EnablePowerDown(0);

        /* wait for interrupt, enter in Idle mode */
        __WFI();
}
 ……
switch (u8Item)
{
        case '1':
```

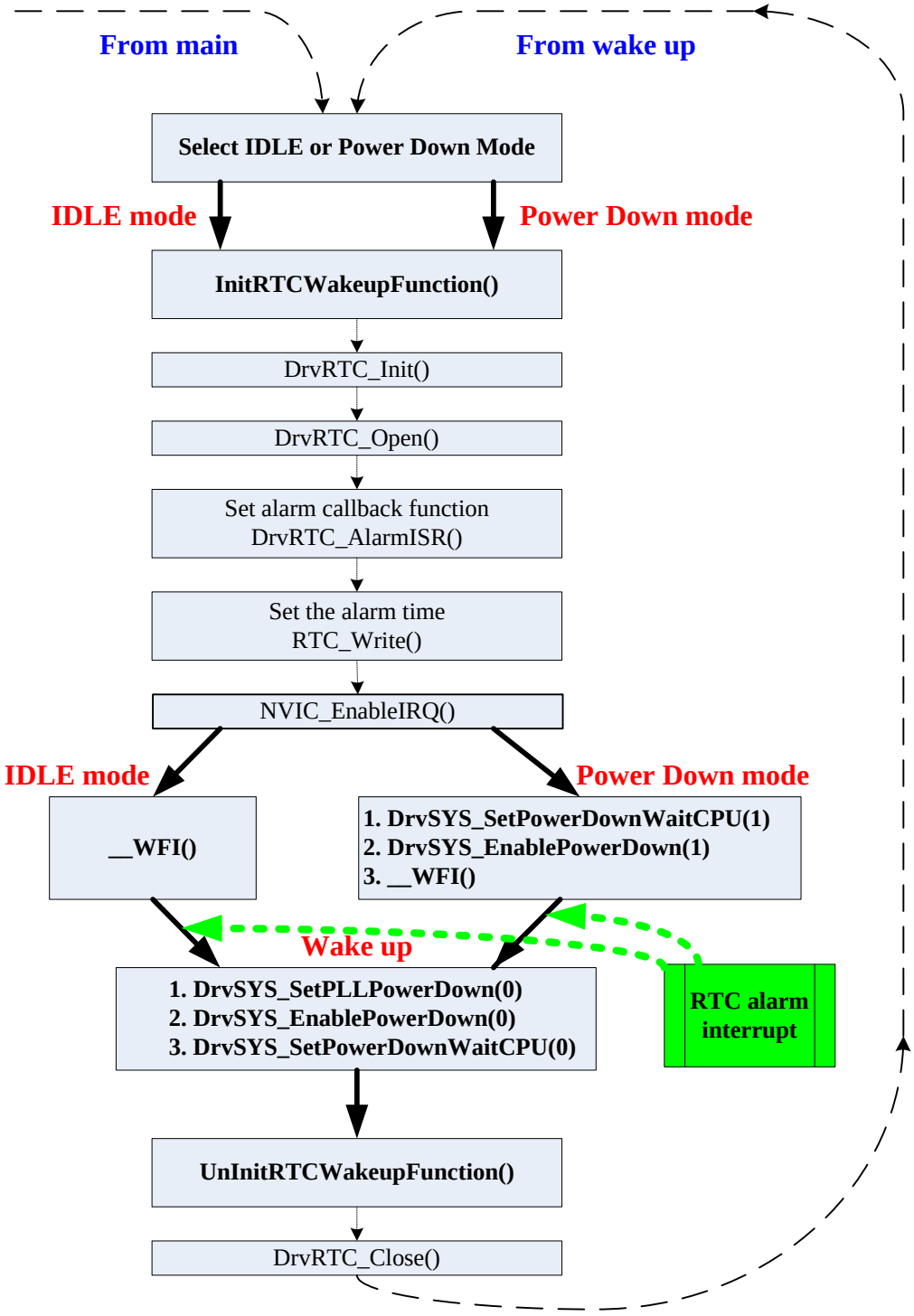
```
                UnInitRTCWakeupFunction();
                break;
        case '2':
                UnInitGPIOWakeupFunction();
                break;
        case '3':
                UnInitUARTWakeupFunction();
                break;
        case '4':
                UnInitUSBWakeupFunction();
                break;
        case '5':
                UnInitCANWakeupFunction();
                break;
        default :
                break;
}
……
```

3 调用顺序

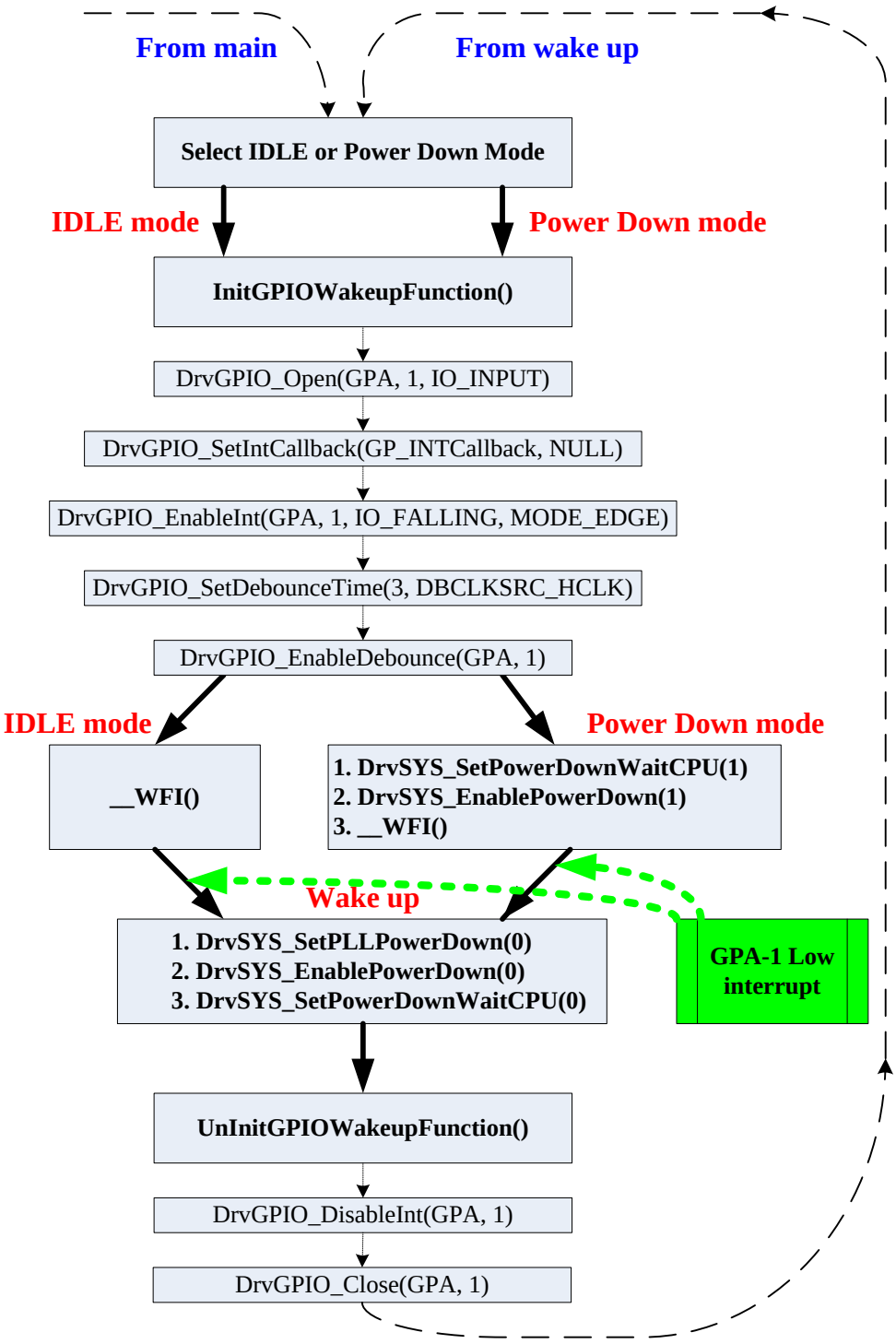
3.1 使能/禁止选定的IP时钟



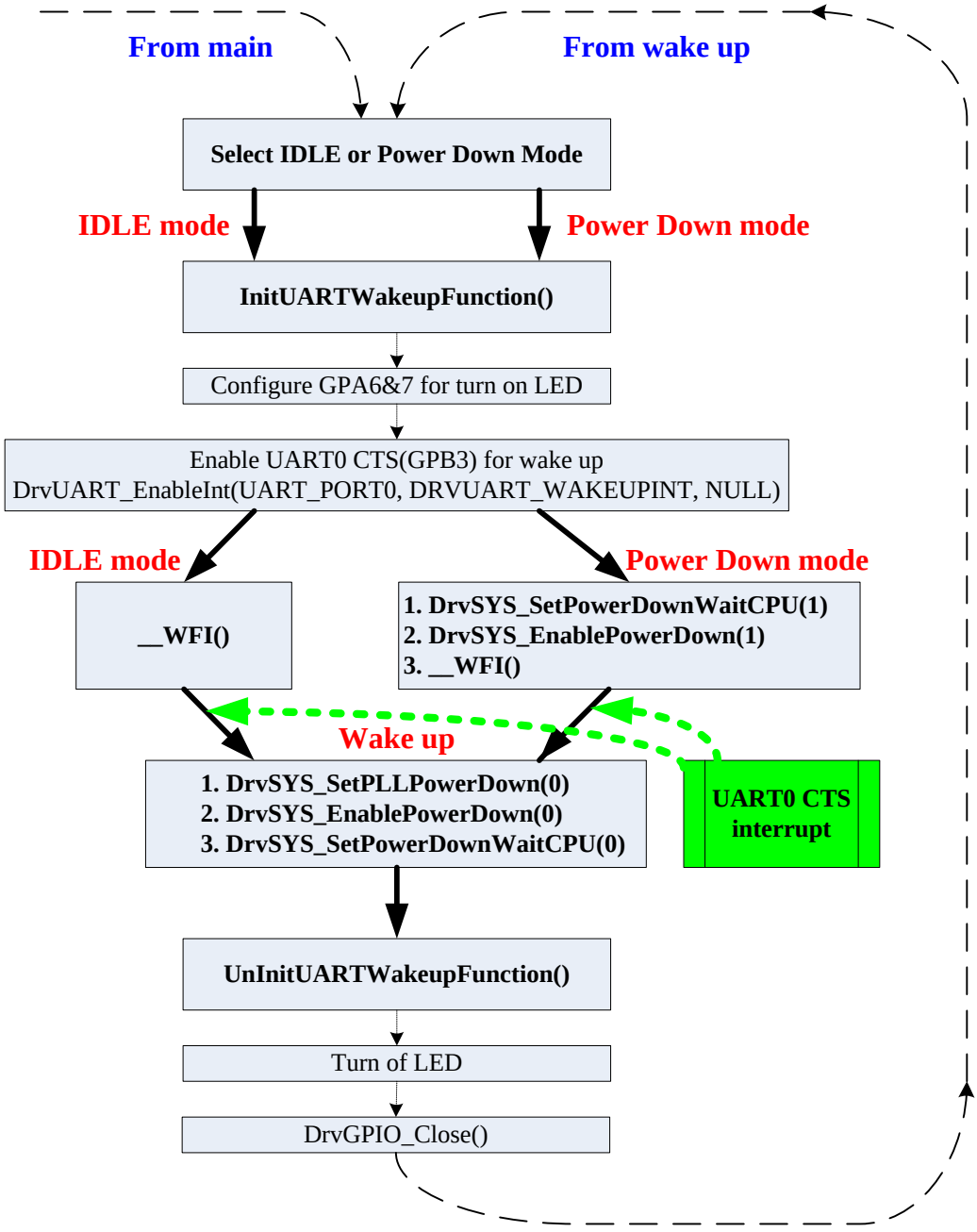
3.2 进入断电/空闲模式并通过RTC唤醒



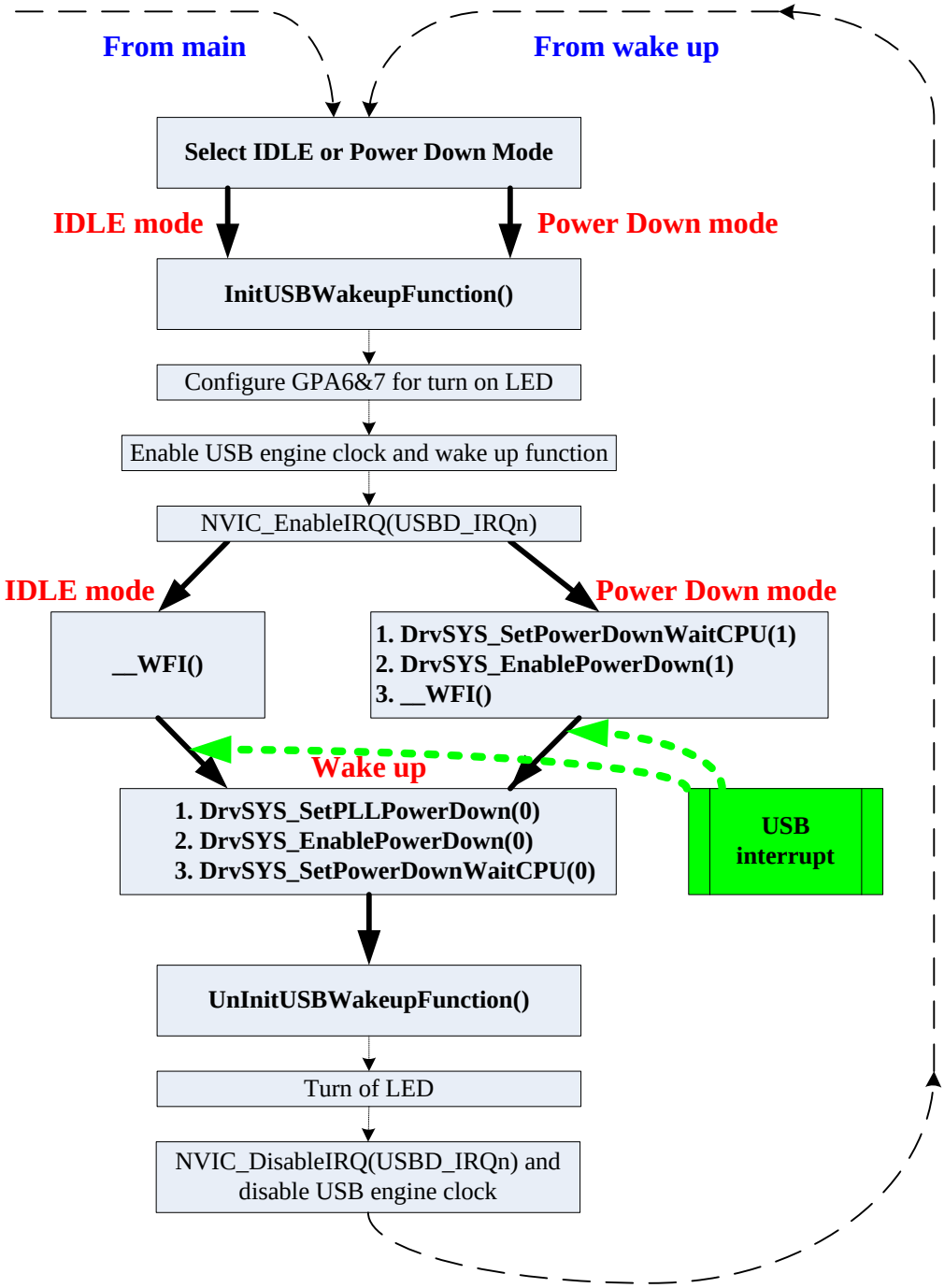
3.3 进入断电/空闲模式并通过GPIO唤醒



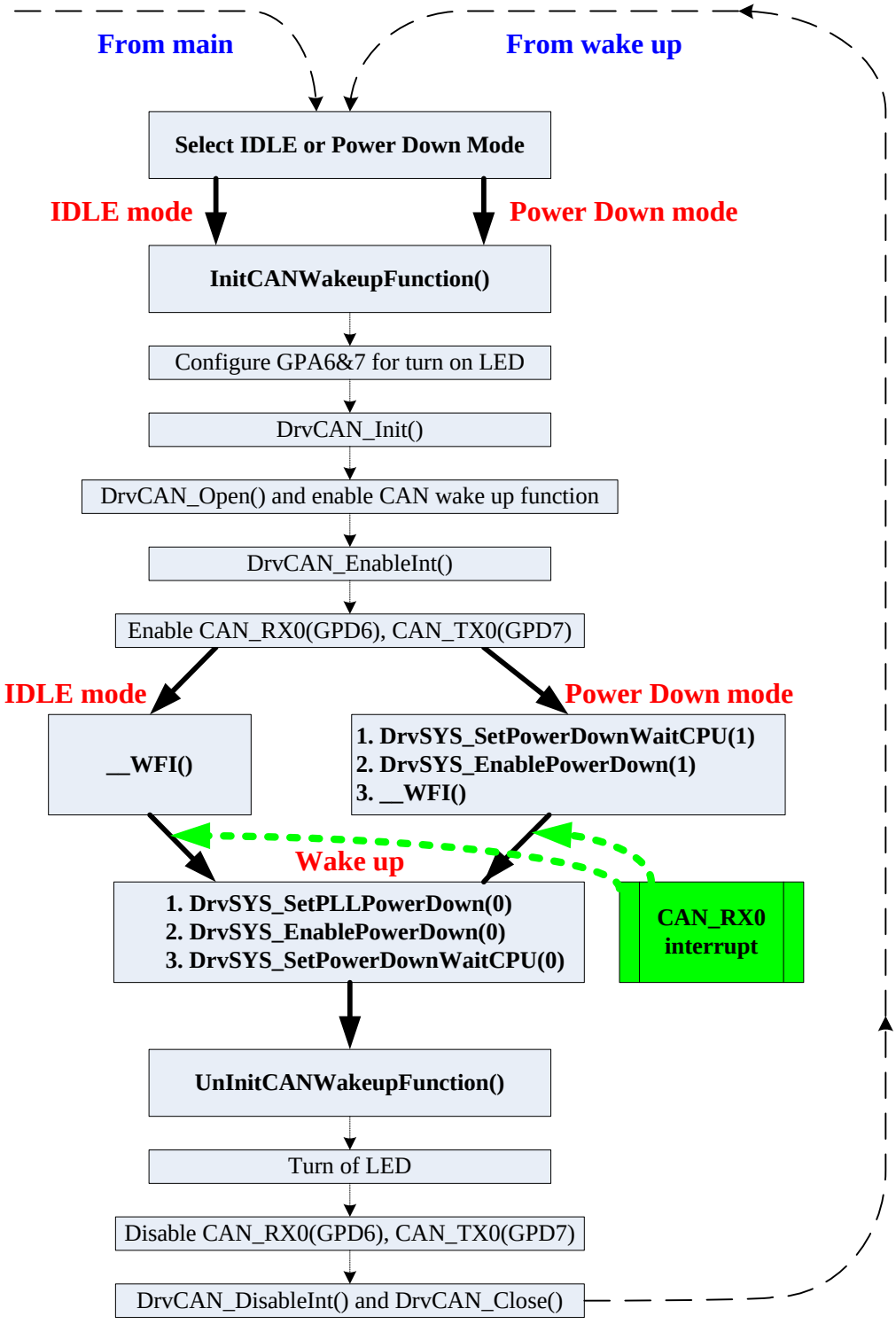
3.4 进入断电/空闲模式并通过UART唤醒



3.5 进入断电/空闲模式并通过USB唤醒



3.6 进入断电/空闲模式并通过CAN唤醒



4 运行环境设置和结果

4.1 电流消耗表

IP Enable/ Disable	电流消耗(毫安)
Watch Dog Timer	0.08
RTC	0.45
Timer 0	0.2
Timer 1	0.2
Timer 2	0.29
Timer 3	0.24
I2C 0	0.29
I2C 1	0.29
SPI 0	0.81
SPI 1	0.84
SPI 2	0.84
SPI 3	0.83
UART 0	0.94
UART 1	0.94
UART 2	x
PWM 10	0.23
PWM 32	0.24
PWM 54	x
PWM 76	x
CAN	1.11
USB	1.94
ADC	1.23
ACMP	0.17
PS2	0.59
PDMA	0.79
Flash ISP	0.4

4.2 断电/空闲模式电流和唤醒方式

使用外部12MHz晶振系统时钟是48MHz@3.3V

系统模式	电流消耗
空闲模式	15.47 毫安
断电模式	15.2 微安
断电模式下使能UART0打印调试信息	34.8 微安

4.2.1 RTC报警唤醒

设置新的RTC日期/时间以及10秒报警时间然后进入断电/空闲模式，10秒后，系统将通过RTC警报中断退出断电/空闲模式然后进入正常模式

4.2.2 GPIO中断唤醒

设置GPIOA-1为输入模式（引脚状态为高）下降沿触发中断，然后进入断电/空闲模式，当GPIOA-1由高变低时，系统将通过GPIO中断退出断电/空闲模式然后进入正常模式

4.2.3 UART CTS 唤醒

使能UART0唤醒功能，配置GPB3为CTS0然后进入断电/空闲模式，当CTS0(GPB3)电平转换时系统将退出断电/空闲模式然后进入正常模式

4.2.4 USB 插入唤醒

使能USB模块时钟和唤醒功能然后进入断电/空闲模式，当USB线插入PC时，系统将退出断电/空闲模式然后进入正常模式

4.2.5 CAN RX 唤醒

使能相关CAN及CAN唤醒功能并设置GPIOD-6为CAN_RX输入脚然后进入断电/空闲模式，当GPIOD-6(CAN_RX)脚电平转换时，系统将退出断电/空闲模式然后进入正常模式

4.2.6 唤醒时间

系统退出断电模式需要2微秒，这段时间指的是从唤醒事件触发到中断事件发生的时间

5 修订历史

版本	日期	描述
0.01	2010-3-9	1. 初次发布

Important Notice

Nuvoton products are not designed, intended, authorized or warranted for use as components in systems or equipment intended for surgical implantation, atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, or for other applications intended to support or sustain life. Further more, Nuvoton products are not intended for applications wherein failure of Nuvoton products could result or lead to a situation wherein personal injury, death or severe property or environmental damage could occur.

Nuvoton customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nuvoton for any damages resulting from such improper use or sales.

Please note that all data and specifications are subject to change without notice. All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.