

GR551x睡眠模式及功耗测量说明

版本: 1.7

发布日期: 2020-09-25

深圳市汇顶科技股份有限公司

版权所有 © 2020 深圳市汇顶科技股份有限公司。保留一切权利。

非经本公司书面许可,任何单位和个人不得对本手册内的任何部分擅自摘抄、复制、修改、翻译、传播,或将其全部或部分用于商业用途。

商标声明

G@DiX和其他汇顶商标均为深圳市汇顶科技股份有限公司的商标。本文档提及的其他所有商标或注册商标,由各自的所有人持有。

免责声明

本文档中所述的器件应用信息及其他类似内容仅为您提供便利,它们可能由更新之信息所替代。确保应用符合技术规范,是您自身应负的责任。

深圳市汇顶科技股份有限公司(以下简称"GOODIX")对这些信息不作任何明示或暗示、书面或口 头、法定或其他形式的声明或担保,包括但不限于针对其使用情况、质量、性能、适销性或特定用途的 适用性的声明或担保。GOODIX对因这些信息及使用这些信息而引起的后果不承担任何责任。

未经GOODIX书面批准,不得将GOODIX的产品用作生命维持系统中的关键组件。在GOODIX知识产权保护下,不得暗中或以其他方式转让任何许可证。

深圳市汇顶科技股份有限公司

总部地址: 深圳市福田保税区腾飞工业大厦B座2层、13层

电话: +86-755-33338828 传真: +86-755-33338099

网址: <u>www.goodix.com</u>

前言

编写目的

本文档将介绍GR551x系列芯片的睡眠模式以及示例应用详解,以指导用户快速配置睡眠模式。

读者对象

本文适用于以下读者:

- GR551x用户
- GR551x开发人员
- GR551x测试人员
- 开发爱好者
- 文档工程师

版本说明

本文档为第5次发布,对应的产品系列为GR551x。

修订记录

版本	日期	修订内容
1.0	2019-12-08	首次发布
1.3	2020-03-16	更新了"IO管脚配置"章节描述
1.5	2020-05-30	优化了"睡眠模式"章节描述和IO"管脚配置"章节描述
1.6	2020-06-30	删除"IO管脚配置"章节代码中msio的声明
		优化了"睡眠流程验证"章节描述及示意图
		删除了"软件配置"章节中io_table的配置说明
1.7	2020-09-25	"环境搭建"章节,补充开始功耗测量前需先移除J5上跳线帽的说明、增加开发板硬件
		布局图

目录

前言	I
1 简介	1
1.1 睡眠模式	1
1.2 唤醒源	2
1.3 睡眠控制流程	3
1.4 参考文档	4
2 Demo测试以及验证	5
2.1 准备工作	5
2.2 硬件连接	5
2.3 睡眠流程验证	6
2.3.1 配置示例工程	6
2.3.2 编译下载	7
2.3.3 输出波形验证	7
3 模式配置	8
3.1 配置睡眠模式	8
3.2 使能睡眠模式	8
3.2.1 配置Active Mode	9
3.2.2 配置Idle Mode	9
3.2.3 配置Sleep Mode	9
4 IO管脚配置	
4.1 配置原则	
4.2 软件配置	
5 测量功耗	15
5.1 测量原理	15
5.2 环境搭建	
5.3 进行测量	
6 常见问题	
6.1 热启动外设无法正常操作	
6.2 J-Link断开	
6.3 J-Link无法链接	

简介

1 简介

本章介绍了GR551x芯片的睡眠模式、唤醒源、睡眠模式的转换流程。

1.1 睡眠模式

GR551x芯片支持工作模式、空闲模式、睡眠模式。

工作模式(Active Mode)

此模式下GR551x芯片的CPU处于全速运行,用户可以根据需求,合理的配置并使用外设。在此模式下,典型的情况是:

- MCU子系统(包括ARM处理器、SRAM和外设等)一直处于就绪或工作状态。
- 。 蓝牙子系统(包括RF收发机、通信内核等)一直处于就绪或工作状态。
- PMU子系统(包括DC/DC、LDO、RTC等)一直处于工作状态。
- 空闲模式(Idle Mode)

配置Idle模式是通过ARM系列自带的WFI/WFE(Wait For Interrupt/Event)指令实现。当调用此指令时,PC指针停留在WFI/WFE指令处,当系统发生IRQ或者Event来唤醒CPU时,PC指针指向下一条指令并继续执行;由于GR551x芯片的蓝牙业务均是基于中断,GR551x芯片执行蓝牙业务时,为了降低功耗,可使系统进入Idle模式;在没有蓝牙业务的时候,可以让蓝牙子系统进入断电模式,进一步节约功耗。

• 睡眠模式(Sleep Mode)

Sleep Mode模式指的是,当GR551x芯片没有蓝牙业务以及其他可处理的任务时,可以使系统进入WFI/WFE状态,为了进一步节约功耗,系统进入Sleep Mode;此模式下,XTAL32M时钟停止运行,以下模块处于断电状态:

- 。 MCU子系统(Retention SRAM除外)
- 。 蓝牙子系统

在Sleep Mode下,系统只有Always-on(AON)模块的电源运行,以保证Retention SRAM中存储的数据不丢失,同时为具有唤醒功能的模块(如Bluetooth LE(BLE) Timer、Sleep Timer、Real Time Calendar、AON GPIO)供电。



图 1-1 GR551x系统架构框图

用户可用SDK层的pwr_mgmt_mode_set()接口将GR551x芯片动态切换到所需模式。

void pwr_mgmt_mode_set(pwr_mgmt_mode_t pm_mode);



图 1-2 状态转换图

1.2 唤醒源

唤醒源可将GR551x芯片从低功耗模式中唤醒,包括从Idle Mode唤醒为Active Mode、从Sleep Mode唤醒为Active Mode。

Idle Mode: ARM处理器执行WFE(Wait For Event)或WFI(Wait For Interrupt)指令后进入空闲模式并保持 该状态。从Idle Mode 到Active Mode的唤醒源包括:

- Reset
- NVIC Event
- Debug Event

G@DiX

Sleep Mode: 进入睡眠模式后,只有Always-on域的模块可唤醒系统,并使系统进入热启动流程。从Sleep Mode到Active Mode的唤醒源包括:

- BLE Timer
- Sleep Timer
- Real Time Calendar
- AON GPIO
- LPCOMP
- Reset

用户可以调用SDK的API来配置唤醒源:

void pwr_mgmt_wakeup_source_setup(uint32_t wakeup_source);

唤醒源配置详情,请参考《GR551x Datasheet》。

1.3 睡眠控制流程



图 1-3 睡眠策略流程图

一个典型的睡眠控制流程(比如用户同时使用BLE和外设)描述如下:

简介

- 1. 系统上电后,进入冷启动。
- 2. 当进入Active Mode后,系统将处于全速运行的高功耗状态。在Active Mode下,芯片可执行任何任务,比如处理蓝牙业务或从外设读取数据。
- 3. 当无业务逻辑处理时,系统进入Check BLE Sleep。此时系统根据BLE Core的内部逻辑判断,使系统进入Idle Mode或者Sleep Mode。
- 4. 如果系统处于BLE Idle状态,蓝牙子系统始终保持供电,以减少BLE启动耗时。系统从BLE Idle状态直接进入Idle Mode。中断事件和调试操作可将系统从Idle Mode中快速唤醒,然后进入Active Mode。
- 5. 如果系统处于BLE Sleep状态,蓝牙子系统掉电,以进一步降低功耗。此时判断剩余睡眠时间(睡眠决策时刻与定时器唤醒时刻之间的时间间隔,所述定时器包括BLE Timer、Sleep Timer、RTC)若小于热启动耗时,则系统会进入Idle Mode。剩余睡眠时间若大于热启动耗时,则系统进入Sleep Mode。处于Sleep Mode下的系统被唤醒源唤醒后将进入热启动(Warm Boot)流程。

🛄 说明:

从Sleep Mode进入热启动流程需要一段启动时间且启动电流较大。睡眠时间较短时(如小于2 ms),系统进入Sleep Mode后功耗降低不明显。从系统综合性能考虑,当睡眠时间较长时,系统进入Sleep Mode的平均功耗更低。

1.4 参考文档

在进行操作前,可参考以下文档。

名称	描述		
GR551x开发者指南	GR551x软硬件介绍、快速使用及资源总览		
Keil用户指南	Keil详细操作说明: <u>https://www.keil.com/support/man/docs/uv4/</u>		
Bluetooth GATT Spec	Bluetooth Profile和Service的详细信息: <u>https://www.bluetooth.com/specifications/gatt</u>		
GP551x Datachoot	介绍GR551x芯片引脚分布、内存、PMU、时钟、外设、安全核心、通信子系统和封装		
GUJJIX Datasneet	信息。		

表 1-1 文档参考

2 Demo测试以及验证

本章介绍如何使用和验证带**GR551x**睡眠模式的示例工程,SDK_Folder\projects\ble\ble_periph eral\目录下的示例工程均支持睡眠模式。

🛄 说明:

SDK_Folder为GR551x SDK的根目录。

2.1 准备工作

应用工程示例之前,需要完成以下准备工作。

硬件准备

表 2-1 硬件准备

名称	描述
开发板	GR5515 Starter Kit开发板,简称GR5515 SK板
连接线	Micro USB 2.0数据线、USB 2.0数据线
逻辑分析仪	Saleae逻辑分析仪、KeySight测试仪

软件准备

表 2-2 软件准备

名称	描述
Windows	Windows 7/Windows 10操作系统
J-Link Driver	J-Link驱动程序,下载网址: <u>https//www.segger.com/downloads/jlink/</u>
Keil MDK5	IDE工具,下载网址: <u>https://www.keil.com/download/product/</u>

2.2 硬件连接

使用Micro USB 2.0数据线连接GR5515 Starter Kit开发板与计算机。



图 2-1 硬件连接示意图

2.3 睡眠流程验证

当系统处于Sleep Mode时,一旦产生BLE timer中断,系统将被唤醒并进入热启动流程。在热启动流程代码中,通过添加GPIO初始化模块(比如GPIO翻转),可实现GPIO在系统Sleep Mode和热启动下输出相应的消息,以验证睡眠模式与热启动流程之间转换是否正常。

下面章节将对睡眠流程验证的各步骤进行详细描述。

2.3.1 配置示例工程

本文档以ble_app_pcs示例工程为例进行配置说明。

🛄 说明:

ble_app_pcs开始运行时,系统处于Ultra Deep Sleep状态。复位后,长按开发板上的"OK"键3秒以上,系统将发起持续时长为30秒的广播。

用户可调用SDK层的pwr_mgmt_ultra_sleep()进入Ultra Deep Sleep。该模式下除Always-on(AON)模块的电源运行,其他电源域均关闭电源,Retention SRAM电源也会被关闭,所以通过AON GPIO或Sleep Timer唤醒后,程序将执行冷启动流程。

- 用Keil打开ble_app_pcs工程,路径为: SDK_Folder\projects\ble\ble_peripheral\ble_ap p_pcs\Keil_5\ble_app_pcs.uvprojx
- 2. 在热启动流程的代码(main_init函数)中,添加GPIO初始化函数。

用户可使用GPIO输出信息,该GPIO应该是当前应用工程中未被使用的GPIO(如AON_GPIO3)。

```
void main init(void)
{
    uint32 t boot flag = get wakeup flag();
    if ( COLD BOOT == boot flag )
    {
        extern void ___main(void);
        main();
    }
    else
    {
        ll aon gpio set pin mode(LL AON GPIO PIN 3, LL AON GPIO MODE OUTPUT);
        ll aon gpio disable it(LL AON GPIO PIN 3);
        ll_aon_gpio_toggle_pin(LL_AON_GPIO_PIN_3);
        ll aon gpio toggle pin(LL AON GPIO PIN 3);
        warm boot();
        while (1);
    }
    // Never execute here
}
```

🛄 说明:

名称: system_gr55xx.c

```
路径: SDK_Folder\toolchain\gr551x\source\
```

3. 使能睡眠功能:将pwr_mgmt_mode_set()函数的入参修改为:PMR_MGMT_SLEEP_MODE(进入Sleep Mode)。

```
void app_periph_init(void)
{
    SYS_SET_BD_ADDR(s_bd_addr);
    ble_rf_tx_mode_set(BLE_RF_TX_MODE_ULP_MODE);
    ble_rf_match_circuit_set(BLE_RF_MATCH_CIRCUIT_250HM);
    wkup_key_init();
    pwr_mgmt_mode_set(PMR_MGMT_SLEEP_MODE);
}
```

🛄 说明:

名称: user_periph_setup.c

路径: SDK_Folder\projects\ble\ble_peripheral\ble_app_pcs\Src\platform\

2.3.2 编译下载

使用Keil编译ble_app_pcs工程并下载至GR5515 SK板,具体操作请见《GR551x开发者指南》。

2.3.3 输出波形验证

使用逻辑分析仪器捕捉GPIO口(AON_GPIO3)的输出波形。



图 2-2 逻辑分析仪捕捉GPIO口的输出波形

AON_GPIO3输出周期性的脉冲波(见图 2-2),表示系统的睡眠模式与热启动流程正常配合。其中,两次脉冲之间代表系统进入Sleep Mode。当BLE Timer中断到来即刻唤醒整个系统,使系统进入热启动。热启动流程中,GPIO被初始化,AON_GPIO3翻转为高电平。

🛄 说明:

系统处于Sleep Mode时J-Link会断开,用户无法进行在线调试。推荐使用GPIO方式输出外部信息进行Sleep Mode调试。

3 模式配置

本章主要介绍如何配置GR551x的Active Mode、Idle Mode、Sleep Mode。

3.1 配置睡眠模式

实际应用可根据睡眠策略灵活配置睡眠模式,以实现系统的快速响应、低功耗设计。

以ble_app_pcs工程为例,其main函数基本能满足常规的低功耗应用。用户可通过pwr_mgmt_schedule函数 进行睡眠策略控制。该main函数的代码片段如下所示。

```
int main(void)
{
    // Initialize user peripherals.
   app periph init();
    // Initialize ble stack.
   ble_stack_init(&s_app_ble_callback, &heaps_table);
    if (is enter ultra deep sleep())
    {
        pwr mgmt ultra sleep(0);
    }
    // Loop
    while (1)
    {
        pwr mgmt schedule();
    }
}
```

🛄 说明:

名称: main.c

```
路径: SDK_Folder\projects\ble\ble_peripheral\ble_app_pcs\Src\user\
```

3.2 使能睡眠模式

用户可通过pwr_mgmt_mode_set()函数配置系统的Active Mode、Idle Mode、Sleep Mode。

模式配置

🛄 说明:

名称: gr55xx_pwr.h

路径: SDK_Folder\components\sdk

pwr_mgmt_mode_set()的形参pwr_mgmt_mode_t,用于设置系统的睡眠模式。

表 3-1 pwr	_mgmt_	_mode_	_t的取值
-----------	--------	--------	-------

pm_mode值	说明		
PMR_MGMT_ACTIVE_MODE	系统进入并保持在Active Mode,全速运行。		
PMR_MGMT_IDLE_MODE	允许进入Idle Mode,系统会在Active Mode与Idle Mode之间切换。 • 有待处理任务时,系统处于Active Mode。 • 无待处理任务时,系统处于Idle Mode,等待中断唤醒后进入Active Mode。		
PMR_MGMT_SLEEP_MODE	 允许系统进入Sleep Mode,系统根据睡眠策略,在Active Mode、Idle Mode、Sleep Mode之间切换。 当有待处理任务时,系统处于Active Mode。 无待处理任务时,系统根据可预测的睡眠时间(AON Timer 与BLE Timer均可预测睡眠时间)判断进入Idle Mode或Sleep Mode。 		

3.2.1 配置Active Mode

通过设置pwr_mgmt_mode_set()函数的形参pwr_mgmt_mode_t为PMR_MGMT_ACTIVE_MODE,系统将始终保持在Active mode。Unused memory block在系统初始化时已做自动配置,用户也可对unused memory block进行电源以及时钟控制,以减少Active Mode下的功耗。

3.2.2 配置Idle Mode

设置pwr_mgmt_mode_set()函数的形参pwr_mgmt_mode_t为PMR_MGMT_IDLE_MODE,系统根据是否有待处理任务,在Active Mode与Idle Mode之间切换。在示例工程的main()函数中,pwr_mgmt_schedule()函数会自动根据睡眠策略,关闭相应模块。例如:当蓝牙无业务处理时,ble_core_sleep会使蓝牙子系统下电,MCU子系统在处理完当前任务后,会自动进入Idle Mode。

3.2.3 配置Sleep Mode

通常系统的Sleep Mode与热启动模型需要配合使用。

Sleep Mode

设置pwr_mgmt_mode_set()函数的形参pwr_mgmt_mode_t为PMR_MGMT_SLEEP_MODE,系统 根据睡眠策略,在Active Mode、Idle Mode、Sleep Mode之间切换。在示例工程的main()函数 中,pwr_mgmt_schedule()函数根据睡眠策略,自动关闭除Always-on domain以外的电源,并将部 分RAM设置成retention RAM低功耗状态。

热启动

驱动接口中已经实现热启动恢复,用户不需要关心热启动的流程。

🛄 说明:

名称: system_gr55xx.c

路径: SDK_Folder\toolchain\gr551x\source\

4 IO管脚配置

4.1 配置原则

GR551x芯片提供片上可编程上/下拉电阻,以替代外部电阻,节省空间和成本;还可以防止未被使用的GPIO管脚处于浮空状态,避免额外电流从VDDIO电源泄露以及防止意外。

🛄 说明:

用户不可为GPIO管脚同时配置上拉电阻和下拉电阻。单个电阻的值约为100 kΩ。GR551x上电后,默认为所 有GPIO管脚配置下拉电阻。

使能GR551x片上配置上/下拉电阻应遵循以下规则:

- 1. GPIO管脚处于浮空状态,当引脚上有外部干扰电压时,GPIO管脚则应使能内部上/下拉电阻。
- 2. GPIO管脚未使用,则使能内部下拉电阻。
- 3. GPIO管脚设为输入且该管脚被外部设备拉高/低时,则应禁能内部上/下拉电阻。
- 4. GPIO管脚作为输入且连接的外部设备处于高阻抗状态,则应使能内部上/下拉电阻。
- 5. GPIO管脚设为输出时,则应禁止使能内部上/下拉电阻。

关于遵循原则的部分说明和电路原理如下。

 如果GPIO管脚处于浮空状态,当引脚上有外部干扰电压时可能导致输入门导通,从VDDIO经过输入 门产生额外电流。在某些情况下,从VDDIO流出的电流较大会导致芯片运作异常,因此GPIO管脚不 得处于浮空状态。

GR551x GPIO管脚的简要工作原理如图 4-1所示:



图 4-1 GPIO管脚电路

2. GPIO管脚用作输入且有外部驱动设备连接时,上/下拉电阻通常应被禁用,否则可能会引起VDDIO电 流泄露。

🛄 说明:

但当外部驱动设备进入高阻抗状态时,如果GR551x输入还处于浮空状态,这可能会导致额外电流流出。因此,当外部驱动设备进入高阻抗状态时,需使能相应的上/下拉电阻。

(1) GPIO管脚设为输入,将片上电阻配置为上拉电阻,且该管脚的外部驱动设备处于低电平状态。电流会从VDDIO泄露,流经上拉电阻到达地线(流经此外部设备),如图 4-2所示。



图 4-2 低电平驱动输入时的电流示意图

上拉电阻的电阻值约为100 kΩ,当VDDIO的电压为3.3 V时,如有一个GPIO管脚配置不当,则会 有约33 μA的额外电流从VDDIO泄露;如多个GPIO管脚配置不当,则每个管脚都会有33 μA的漏电 流产生。

(2) GPIO管脚设为输入,使能下拉电阻而非上拉电阻,所连接的外部设备处于高电平,则无电流从VDDIO泄露,但VDDIO会通过外部器件产生电流,如图 4-3所示。



图 4-3 高电平驱动输入时的电流示意图

3. GPIO管脚设为输出时,应禁用上/下拉电阻。否则也可能会引起电流泄露且额外电流会流经该上/下 拉电阻。

🛄 说明:

但当GPIO设为输出时,内部上/拉电阻长期处于禁能状态,GR551x输入门的输入和外部设备的输入都会处于浮空状态,而用户须避免出现这样的浮空状态。因此,则在高阻态期间使能上/下拉电阻。

(1) 将GPIO管脚设为输出且为该管脚配置下拉电阻,当输出处于高电平状态时,电流会从VDDIO流 向该下拉电阻,如图 4-4所示。



图 4-4 GPIO设为输出且输出高电平的电流图

(2) GPIO管脚设为输出且处于低电平时,使能上拉电阻,泄露的电流会流经此上拉电阻和低电平输 出缓冲区,最终到达地线,如图 4-5所示。



图 4-5 GPIO设为输出且输出低电平的电流图

4.2 软件配置

GR551x芯片在初次上电初始化后,三类IO引脚的初始化状态如下:

- GPIO: 输入且使能内部下拉电阻
- AON GPIO: 输入且使能内部下拉电阻
- MSIO: 模拟输入

为防止IO状态不确定而导致漏电,继而影响系统的整体功耗,用户可在实际的外设应用需求中选择合适的上拉或者下拉。例如:

- 对于一般未使用的IO引脚,根据4.1 配置原则中的第2条,可设置为内部下拉。
- I2C模块因设备外部有上拉电阻,根据4.1 配置原则中的第3条,设置SCL/SDA引脚为浮空状态。
- SPI/QSPI模块,可根据通信协议中的CPHA和CPOL配置各引脚初始化状态,如:
 - 。 MODEO: SCLK/CS引脚配置为下拉电阻。
 - 。 MODE3: SCLK/CS引脚配置上拉电阻,其他引脚可配置初始化状态为下拉电阻。

5 测量功耗

本节以GR5515 SK板作为待测板,简述如何使用功耗测试仪(KeySight N6705C DC Power Analyzer)对其进行功耗测试的方式和概要分析测试结果。

5.1 测量原理

使用功耗测试仪器为芯片3.3 V供电时,芯片在运行过程中消耗的电流将经过功耗测试仪器,仪器将电流 信号记录下来。

根据公式:

P=U*I(P为功耗; U为供电电压; I为电流;)

U为电压3.3 V保持不变,P的值与I值成正比,因此功耗仪器记录的电流曲线,便可以认为是功耗的曲线。

5.2 环境搭建

如图 **5-1**,连接GR5515 SK板(已下载2.3.2 编译下载的固件)与KeySight功耗测试仪器(以KeySight N6705C为例)。



图 5-1 SK板和Keysight板连接图

将功耗测试仪与GR5515 SK板进行物理连接。

- 1. 打开KeySight,用KeySight B按钮将通道的电压输出调整到3.3 V,即KeySight C处显示3.3 V。
- 2. KeySight的A+连接到GR5515 SK板的VBAT端(开发板上的J10 Pin 2,如图 5-2所示),为GR5515 SK板 上GR551x芯片供电。



图 5-2 GR5515 SK板的硬件布局(顶层图)

- 3. KeySight的A-连接到GR5515 SK板的GND端,形成供电环路。
- 4. 将GR5515 SK板Micro USB接口接到PC机的任意USB端口,为SK板芯片外围电路供电。
- 5. 通过USB线将KeySight与PC机连接。

🛄 说明:

在GR5515 SK板测量功耗前,需要先将如图 5-2所示J5上Pin 5 ~ Pin 6、Pin 7 ~ Pin 8上的跳线帽移除,并取 下SK板右上方的LCD屏幕。避免额外电流从VDDIO电源泄露,导致测量睡眠功耗异常。详细说明请参考4.1 配置 原则。

5.3 进行测量

基本环境搭建完成后,使用KeySight与PC机的配套软件可以进行功耗的测试。

图 5-3为一个典型的广播业务电流波形图:系统在Sleep状态下唤醒后,进行37、38、39信道的TX/RX业务,业务完毕后,再次进入了Sleep状态。



图 5-3 GR5515 Starter Kit Advertising功耗测试波形

图 5-4为一个典型的连接业务电流波形图,也有同广播业务类似的电流时序。



图 5-4 GR5515 Starter Kit Connection功耗测试波形

6 常见问题

本章列出了系统在热启动、睡眠模式时可能出现的问题和解决办法。

6.1 热启动外设无法正常操作

• 问题描述

热启动唤醒后,外设不能正常运行。

• 问题分析

由于外设属于MCU Subsystem域,睡眠期间会关闭电压,导致原先的寄存器配置丢失。

处理方法

推荐用户采用App Drvier外设驱动接口,当热启动唤醒之后,系统会自动初始化外设模块。如果用户 采用HAL Driver外设驱动接口,需要在hal_pm_resume函数中,增加外设初始化代码,以完成初始化 操作。

6.2 J-Link断开

• 问题描述

使能Sleep Mode后,应用程序会进入Sleep Mode。此时J-Link断开,无法继续调试。

问题分析

Sleep Mode中pmu_sleep会关闭除Always-on域外电源(包括J-Link电源)。J-Link不能保持正常连接,无法调试Sleep Mode。

处理方法

在开发初期,建议选择Active Mode或者Idle Mode进行开发。在开发后期,仅需验证睡眠功能是否正常使用。

6.3 J-Link无法链接

问题描述

将使能Sleep Mode的应用下载到开发板,再次使用J-Link链接时,链接失败。

• 问题分析

由于热启动流程的运行时间较短,开发板大部分时间处于Sleep Mode导致J-Link无法与PC端正常交 互,无法链接。

• 处理方法

使用*custom_config.h*中的BOOT_LONG_TIME配置,使能该字段后,在cold boot阶段,系统会增加1s的delay时间,用户可以利用这一段时间完成J-Link的连接,及时的下载代码刷新固件。