

## 前言

本应用笔记描述了 PY32 系列产品低功耗特性和配置方法，客户可以按照设计方案和此应用笔记可以快速的选择低功耗模式，使得产品的性能和功耗达到最佳的平衡

重要提示：本文档并不取代 PY32 系列的数据手册，本文给出的数据仅作参考，请参考相关的数据手册来获取有效的最新特性数据

在本文档中，PY32 仅指表 1 中列出的产品系列。

表 1. 适用产品

类型	产品系列
微型控制器系列	PY32F030、PY32F003、PY32F002A

## 目录

<b>1 低功耗模式配置流程</b> .....	<b>3</b>
1.1 低功耗模式开关.....	3
1.2 进入 Sleep mode 配置流程.....	3
1.2.1 WFI 模式进入 Sleep 流程:.....	3
1.2.2 Return from ISR 模式进入 Sleep 流程.....	4
1.2.3 WFE 模式进入 Sleep 流程.....	4
1.3 进入 Stop mode 配置流程.....	4
1.3.1 WFI 模式进入 Stop 流程.....	4
1.3.2 Return from ISR 模式进入 Stop 流程.....	5
1.3.3 WFE 模式进入 Stop 流程.....	5
<b>附录 1: 低功耗模式说明</b> .....	<b>6</b>
1.4 Sleep mode.....	6
1.5 Stop mode.....	6
1.6 各种工作模式下的功能.....	6
<b>附录 2: 低功耗模式下的电流和唤醒时间</b> .....	<b>8</b>
1.7 sleep 模式电流.....	8
1.8 stop 模式电流.....	8
1.9 低功耗模式唤醒.....	9
<b>2 版本历史</b> .....	<b>10</b>

## 1 低功耗模式配置流程

### 1.1 低功耗模式开关

进入低功耗和对应模式唤醒如下表所示，用户可根据自己的唤醒方式来配置为对应的进入低功耗流程

表 3- 1 低功耗模式开关

模式	进入	唤醒源	唤醒时钟	对时钟的影响	Voltage regulator	
					MR	LPR
Sleep (sleep-now or sleep-on-exit)	WFI or Return from ISR	任何中断	与进入 sleep 之前一样	CPU 时钟停止，对其他时钟和时钟源没影响。	开 <sup>(1)</sup>	关
	WFE	唤醒事件				
Stop	SLEEPDEEP bit WFI or Return from ISR or WFE  Note: 系统时钟不能选择 LSI 或 LSE	任何配置为唤醒的 EXTI Line (EXTI 寄存器配置)、IWDG、NRST	HSISYS HSI 保持进入 stop 前的频率配置，不分频	HSI、PLL 关闭; HSE 关闭; LSI 和 LSE 可选择开或者关; LPTIMER、RTC、IWDG: 由软件配置是否工作; 低功耗唤醒和部分 RCC 等模块保持工作; 其余模块的时钟关闭。	软件配置开关	软件配置开关, 如果开, 输出电压 1.2/1.0v 可配置

注 1: 软件要配置 VR 的状态为 MR 模式, 才能进入 sleep 模式。

### 1.2 进入 Sleep mode 配置流程

#### 1.2.1 WFI 模式进入 Sleep 流程:

- 暂停 SysTick 定时器 (如有初始化)
- 清除 SCB->SCR 寄存器的 SLEEPDEEP 位
- 执行函数 \_WFI()

参考代码:

```
HAL_SuspendTick();
HAL_PWR_EnterSLEEPMode(PWR_STOPENTRY_WFI);
HAL_ResumeTick();
```

## 1.2.2 Return from ISR 模式进入 Sleep 流程

- 配置中断源
- 清除 SCB->SCR 寄存器的 SLEEPDEEP 位
- 设置 SCB->SCR 寄存器的 SLEEPONEXIT 位
- 等待到中断，执行中断函数后进入低功耗

参考代码：

```
Configure_EXTI();//配置中断源
CLEAR_BIT(SCB->SCR, ((uint32_t)SCB_SCR_SLEEPDEEP_Msk));
SET_BIT(SCB->SCR, ((uint32_t)SCB_SCR_SLEEPONEXIT_Msk));
while(1)
{
    ;
}
```

## 1.2.3 WFE 模式进入 Sleep 流程

- 暂停 SysTick 定时器（如有初始化）
- 清除 SCB->SCR 寄存器的 SLEEPDEEP 位
- 执行函数 \_SEV()
- 执行函数 \_WFI()
- 执行函数 \_WFI()

参考代码：

```
HAL_SuspendTick();//失能 SysTick(SysTick 产生中断会直接退出 SLEEP 模式)
HAL_PWR_EnterSLEEPMode(PWR_STOPENTRY_WFE);//进入睡眠模式
HAL_ResumeTick();//使能 SysTick
```

## 1.3 进入 Stop mode 配置流程

## 1.3.1 WFI 模式进入 Stop 流程

- 设置 SCB->SCR 寄存器的 SLEEPDEEP 位
- 开启 PWR 模块的时钟
- 根据应用需求，设置/清除 PWR->CR1 寄存器的 LPR 位选择供电电压
- 执行函数 \_WFI()

参考代码：

```
__HAL_RCC_PWR_CLK_ENABLE();//开启 PWR 模块的时钟
```

```
HAL_PWR_EnterSTOPMode(PWR_LOWPOWERREGULATOR_ON, PWR_STOPENTRY_WFI);
```

### 1.3.2 Return from ISR 模式进入 Stop 流程

- 配置中断源（中断源为 EXTI line）
- 设置 SCB->SCR 寄存器的 SLEEPDEEP 位
- 设置 SCB->SCR 寄存器的 SLEEPONEXIT 位
- 开启 PWR 模块的时钟
- 根据应用需求，设置/清除 PWR->CR1 寄存器的 LPR 位选择供电电压
- 等待到中断，执行中断函数后进入低功耗

参考代码：

```
Configure_EXTI();
SET_BIT(SCB->SCR, ((uint32_t)SCB_SCR_SLEEPDEEP_Msk));
SET_BIT(SCB->SCR, ((uint32_t)SCB_SCR_SLEEPONEXIT_Msk));
while(1)
{
    ;
}
```

### 1.3.3 WFE 模式进入 Stop 流程

- 设置 SCB->SCR 寄存器的 SLEEPDEEP 位
- 开启 PWR 模块的时钟
- 根据应用需求，设置/清除 PWR->CR1 寄存器的 LPR 位选择供电电压
- 执行函数 \_SEV()
- 执行函数 \_WFI()
- 执行函数 \_WFI()

参考代码：

```
__HAL_RCC_PWR_CLK_ENABLE();
HAL_PWR_EnterSTOPMode(PWR_LOWPOWERREGULATOR_ON, PWR_STOPENTRY_WFE);
```

## 附录 1: 低功耗模式说明

### 1.4 Sleep mode

Sleep 模式下会关闭 CPU 的时钟 (NVIC,) CPU 时钟关闭 (NVIC, SysTick 等工作), 外设可以配置为保持工作。(建议只使能必须工作的模块, 在模块工作结束后关闭该模块)。

### 1.5 Stop mode

在该模式下 SRAM 和寄存器的内容保持, 高速时钟 PLL、HSI 和 HSE 关闭, VDD 域下大部分模块的时钟都被停掉。

### 1.6 各种工作模式下的功能

表 1- 1 各工作模式下的功能

Peripheral	Run	Sleep	Stop	
			VR@LPR or VR@MR	Wakeup ability
CPU	Y	-	-	-
Flash memory	Y	Y	-(2)	-
SRAM	Y	O(3)	-(4)	-
Brown-out reset (BOR)	Y	Y	O	O
PVD	O	O	O	O
DMA	O	O	-	-
HSI	O	O	-	-
HSE	O	O	-	-
LSI	O	O	O	-
LSE	O	O	O	-
PLL	O	O	-	-
HSE Clock Security System (CSS)	O	O	-	-
LSE Clock Security System (CSS)	O	O	O	O
RTC	O	O	O	O
USART1	O	O	-	-
USART2	O	O	-	-
I2C	O	O	-	-

SPI1	O	O	-	-
SPI2	O	O	-	-
ADC	O	O	-	-
COMP1/COMP2	O	O	O	O
Temperature sensor	O	O	-	-
Timers(TIM1/TIM3 /TIM14/TIM16/TIM17)	O	O	-	-
LPTIM	O	O	O	O
IWDG	O	O	O	O
WWDG	O	O	-	-
SysTick timer	O	O	-	-
CRC	O	O	-	-
GPIOs	O	O	O	O

注1. Y = Yes (使能); O = Optional (默认关闭, 可以软件使能); - = Not available

注2. Flash 不下电, 但无时钟提供, 进入最低功耗状态。

注3. SRAM 的时钟可以被开或者关。

注4. SRAM 不下电, 但无时钟提供, 进入最低功耗状态。

注5. 进入 stop 模式之前, 如果使能了 LSE CSS, 则当 LSE CSS 出现问题时, 会唤醒系统, 并进入 NMI 中断。

## 附录 2：低功耗模式下的电流和唤醒时间

## 1.7 sleep 模式电流

表 2- 1 sleep 模式电流

符号	条件				典型值 <sup>(1)</sup>	最大值	单位	
	系统时钟	频率	外设时钟	FLASH sleep				
I <sub>DD</sub> (sleep)	HSI	48MHz	ON	DISABLE	1.8	-	mA	
			OFF	DISABLE	1.1	-	mA	
		24MHz	ON	DISABLE	1	-	mA	
			OFF	DISABLE	0.6	-	mA	
		16MHz	ON	DISABLE	0.75	-	mA	
			OFF	DISABLE	0.5	-	mA	
		8MHz	ON	DISABLE	0.5	-	mA	
			OFF	DISABLE	0.35	-	mA	
		4MHz	ON	DISABLE	0.4	-	mA	
			OFF	DISABLE	0.35	-	mA	
		LSI	32.768kHz	ON	DISABLE	170	-	uA
				OFF	DISABLE	170	-	uA
		LSI	32.768kHz	ON	ENABLE	95	-	uA
				OFF	ENABLE	96	-	uA

## 1.8 stop 模式电流

表 2- 2 stop 模式电流

符号	条件					典型值 <sup>(1)</sup>	最大值	单位
	VCC	VDD	MR/LPR	LSI	外设时钟			
I <sub>DD</sub> (stop)	1.7~5.5V	1.2V	MR	-	-	70	-	uA
		1.2V	LPR	ON	RTC+IWDG+LPTIM	6	-	
					IWDG	6	-	
					LPTIM	6	-	



					RTC	6	-	
				OFF	No	6	-	
		1.0V		ON	RTC+IWDG+LPTIM	4.5	-	
			IWDG		4.5	-		
			LPTIM		4.5	-		
			RTC		4.5	-		
				OFF	No	4.5	-	

## 1.9 低功耗模式唤醒

表 2- 3 低功耗模式唤醒时间

符号	参数 <sup>(1)</sup>		条件		典型值 <sup>(2)</sup>	最大值	单位
$T_{WUSLEEP}$	Sleep 的唤醒时间		-		1.65		us
$T_{WUSTOP}$	Stop 的唤醒时间	MR 供电	Flash 中执行程序, HSI(24Mhz) 作为系统时钟		3.5		us
		LPR 供电	Flash 中执行程序, HSI 作为系统时钟	VDD=1.2V	6	us	
				VDD=1.0V	6		

唤醒时间的测量是从唤醒源触发开始至用户程序读取第一条指令。

## 2 版本历史

版本	日期	更新记录
V1.0	2022.06.10	初版
V1.1	2023.08.16	增加 002A 内容
V1.2	2023.08.24	更新声明



Puya Semiconductor Co., Ltd.

### 声 明

普冉半导体(上海)股份有限公司 (以下简称: “Puya” ) 保留更改、纠正、增强、修改 Puya 产品和/或本文档的权利, 恕不另行通知。用户可在下单前获取产品的最新相关信息。

Puya 产品是依据订单时的销售条款和条件进行销售的。

用户对 Puya 产品的选择和使用承担全责, 同时若用于其自己或指定第三方产品上的, Puya 不提供服务支持且不对此类产品承担任何责任。

Puya 在此不授予任何知识产权的明示或暗示方式许可。

Puya 产品的转售, 若其条款与此处规定不一致, Puya 对此类产品的任何保修承诺无效。

任何带有 Puya 或 Puya 标识的图形或字样是普冉的商标。所有其他产品或服务名称均为其各自所有者的财产。

本文档中的信息取代并替换先前版本中的信息。

普冉半导体(上海)股份有限公司 - 保留所有权利