# PM

APT32F171x支持两种低功耗模式，SLEEP、DEEPSLEEP。在csi驱动中，有一个全局的宏CONFIG\_USER\_PM。一旦开启，就必须定义进出低功耗模式前后的自定义处理函数。

## API列表

Table 5‑1 PM CSI接口函数

|  |  |  |
| --- | --- | --- |
| API | 说明 | 函数位置 |
| csi\_pm\_attach\_callback | 设置进出某个低功耗模式前后的用户自定义处理函数。 | pm.c |
| csi\_pm\_config\_wakeup\_source | 设置唤醒源。 |
| csi\_pm\_enter\_sleep | 进入低功耗模式。 |
| csi\_pm\_clk\_enable | 睡眠模式下使能/关闭时钟 |

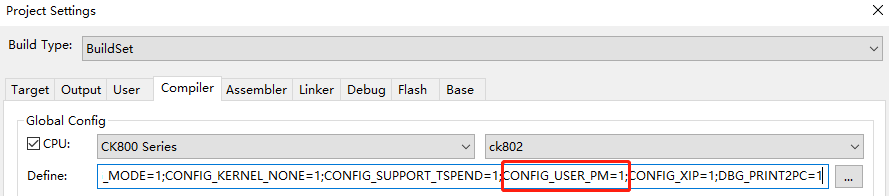
## API详细说明

### csi\_pm\_attach\_callback

void csi\_pm\_attach\_callback(csi\_pm\_mode\_e eMd, void \*pBeforeSlp, void \*pWkup)

#### 功能描述

这个函数受一个宏（CONFIG\_USER\_PM）控制。只有打开这个宏，函数才会加入编译。可以考虑在以下应用场景时打开这个宏：希望在系统进入低功耗模式前做一些状态和数据保存的操作，在系统从低功耗模式中恢复时恢复一些状态。建议在工程设置compiler tab下的Define中加入：CONFIG\_USER\_PM=1。



**Figure**  **5‑1 开启全局宏**

#### 参数/返回值说明

1. 参数：

eMd：低功耗模式。

pfBeforeSlp：进入低功耗模式前需调用的函数名。

pfWkup：从低功耗模式出来后需调用的函数名。

1. 返回值：无。
2. 参数说明表

|  |  |  |
| --- | --- | --- |
| 参数 | 说明 | 位置 |
| eMd | 可选值为csi\_pm\_mode\_t枚举值中的一个。  typedef enum {      PM\_MODE\_RUN  = 0      PM\_MODE\_SLEEP,  PM\_MODE\_DEEPSLEEP,  } csi\_pm\_mode\_e; | pm.h |
| pfBeforeSlp | (void \*)空指针。传入进入低功耗模式前需要调用函数（用户自定义）的名字。 |
| pfWkup | (void \*)空指针。从低功耗模式退出后调用函数（用户自定义）的名字。 | 用户自定义函数位置 |

### csi\_pm\_enter\_sleep

csi\_error\_t csi\_pm\_enter\_sleep(csi\_pm\_mode\_e eMode)

#### 功能描述

调用函数即进入指定的低功耗模式。如果工程打开了宏（CONFIG\_USER\_PM），如Figure 5-1所示，会在进出低功耗模式前分别调用用户自定义函数。此时必须用csi\_pm\_attach\_callback() 指定自定义函数，否则芯片会出现异常。如果不需要自定义函数，则需要关闭全局宏，即Figure 5-1中删除CONFIG\_USER\_PM=1。

#### 参数/返回值说明

1. 参数：

eMd：低功耗模式。

1. 返回值：

csi\_error\_t型数据。当传入低功耗模式不在枚举变量范围时，会返回CSI\_ERROR。

1. 参数说明表

|  |  |  |
| --- | --- | --- |
| 参数 | 说明 | 位置 |
| eMode | 可选值为csi\_pm\_mode\_e枚举值中的一个。  typedef enum {      PM\_MODE\_RUN  = 0      PM\_MODE\_SLEEP,  PM\_MODE\_DEEPSLEEP,  } csi\_pm\_mode\_e; | pm.h |

### csi\_pm\_config\_wakeup\_source

csi\_error\_t csi\_pm\_config\_wakeup\_source(wakeupn\_type\_e eWkupSrc, bool bEnable)

#### 功能描述

设置将系统从DEEP-SLEEP模式中唤醒的源。

注：任何使能的中断都可以将系统从SLEEP模式中唤醒。无需在这里配置。

#### 参数/返回值说明

1. 参数：

eWkupSrc：选择唤醒源。

bEnable：使能/禁止。

1. 返回值：

csi\_error\_t型数据。当传入唤醒源不在枚举变量范围时，会返回CSI\_ERROR。

1. 参数说明表

|  |  |  |
| --- | --- | --- |
| 参数/返回值 | 说明 | 位置 |
| eWkupSrc | 可选值为wakeupn\_type\_e枚举值中的一个。  typedef enum {     WKUP\_EXI0 = 0,  WKUP\_EXI1,  WKUP\_EXI2,  WKUP\_EXI3,  WKUP\_EXI4,  WKUP\_IWDT = 8,  WKUP\_LVD = 11  } csi\_wakeup\_src\_e | pm.h |
| bEnable | ENABLE/DISABLE |  |
| csi\_error\_t | csi\_error\_t 中定义值 | csi\_error\_t |

### csi\_pm\_clk\_enable

void csi\_pm\_clk\_enable(csi\_pm\_clk\_e eOsc, bool bEnable)

#### 功能描述

睡眠模式下使能/禁止某时钟。

#### 参数/返回值说明

1. 参数
2. eOsc：选择某个时钟，枚举定义详见csi\_pm\_clk\_e。
3. bEnable：使能/禁止（ENABLE/DISABLE）
4. 返回值：无。
5. 参数/返回值说明表

|  |  |  |
| --- | --- | --- |
| 参数 | 说明 | 位置 |
| ePower | 可选值为csi\_snooze\_power\_e枚举值中的一个。  typedef enum {       SP\_IDLE\_PCLK = (0x01ul << 8), ///< IDLE\_PCLK for sleep  SP\_IDLE\_HCLK = (0x01ul << 9), ///< DLE\_HCLK for sleep  DP\_ISOSC = (0x01ul << 12), ///< ISOSC for deepslee  DP\_IMOSC = (0x01ul << 13), ///< IMOSC for deepsleep  DP\_EMOSC = (0x01ul << 15) ///< EMOSCfordeepsleep  } csi\_pm\_clk\_e; | pm.h |
| bEnable | Bool 类型数值，ENBALE/DISABLE | common.h |