

## 32 位微控制器

# HC32F460 系列的停止模式注意事项

适用对象

F 系列	HC32F460
------	----------

# 目 录

<b>1</b>	<b>摘要 .....</b>	<b>3</b>
<b>2</b>	<b>HC32F460 系列的停止模式.....</b>	<b>3</b>
2.1	简介 .....	3
2.2	寄存器介绍.....	3
2.3	操作流程 .....	4
2.3.1	进入停止模式 .....	4
2.3.2	解除停止模式 .....	4
2.4	注意事项 .....	6
<b>3</b>	<b>样例代码 .....</b>	<b>7</b>
3.1	代码介绍 .....	7
3.2	代码运行 .....	9
<b>4</b>	<b>版本信息 &amp; 联系方式 .....</b>	<b>11</b>

## 1 摘要

本篇应用笔记主要介绍 HC32F460 系列芯片的停止模式注意事项。

## 2 HC32F460 系列的停止模式

### 2.1 简介

停止（STOP）模式是 HC32F460 系列芯片三种低功耗模式之一，在停止模式中，CPU、大部分外设和时钟源都停止动作，芯片保持 CPU 内部寄存器和 SRAM 数据，外设状态和管脚状态。

### 2.2 寄存器介绍

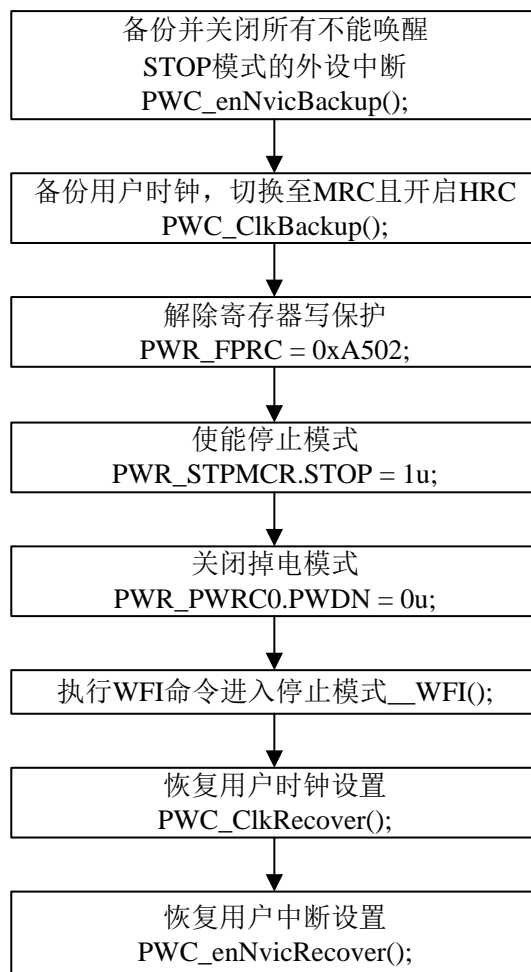
- 1) PWR\_PWRC1：电源模式控制寄存器，设定不同模式进去 STOP 模式时的驱动能力。
- 2) PWR\_STPMCR：停止模式控制器，使能 STOP 模式以及设定 STOP 模式唤醒时 CLK、FLASH 状态。
- 3) INT\_WUPEN：停止模式唤醒事件使能寄存器。

注意：

- 进入 stop 模式之前必须设定 INT\_WUPEN，使能相应唤醒事件，否则无法唤醒 stop 模式。

## 2.3 操作流程

### 2.3.1 进入停止模式



上图列出了进入停止模式的步骤，在停止模式中，芯片的外设和 CPU 都停止工作。

**注意：**当系统时钟为 HRC 和 MPLL 时，需切换系统时钟源至 MRC，其他时钟源只需打开 HRC。

### 2.3.2 解除停止模式

停止模式可以通过复位和中断解除。

可以用于解除停止模式的复位方式有：

- 管脚复位
- 上电复位
- 欠压复位（BOR）

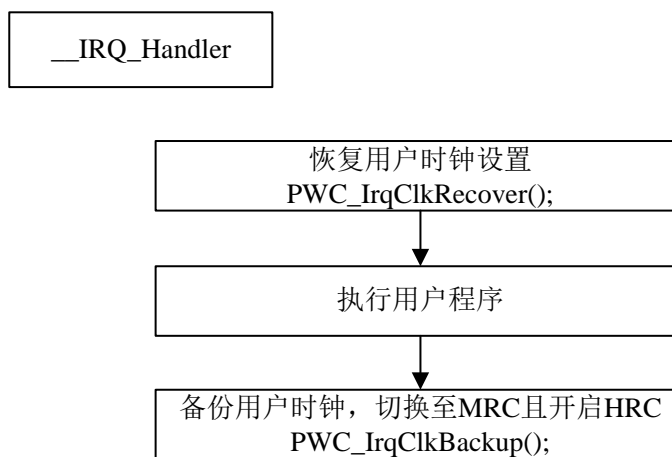
- 电压检测 1/2 复位
- 专用看门狗复位

可以用于解除停止模式的中断事件有：

- 管脚中断 EIRQ0~15
- 电压检测 1/2 中断
- 专用看门狗下溢中断
- 实时时钟的周期中断
- 闹钟中断
- 唤醒定时器中断
- 比较器 1 中断
- UART1\_RXD0 中断
- TIMER01\_A 比较匹配中断

注意：

- 选择中断事件解除停止模式，需在进入停止模式之前操作 INT\_WUPEN 寄存器，设定相应中断事件唤醒许可。



上图说明了唤醒停止模式的步骤。

## 2.4 注意事项

- 1) 执行 WFI 进入停止模式之前，需确保 FLASH 不处于编程或者擦除状态（即 EFM\_FSR.RDY=1），且振荡停止监测功能无效，否则芯片无法进入停止模式。
  - 2) 执行 WFI 进入停止模式之前，需确保 DMA 处于停止状态（即 DMA\_EN.EN=0），否则芯片可能会出现不可保证的动作。
  - 3) 执行 WFI 进入停止模式之前，需将 EIRQ 的数字滤波设置为无效，否则该中断不能用于 STOP 唤醒。
  - 4) 选择中断事件解除停止模式，在执行 WFI 进入停止模式之前，需使能相应中断事件唤醒停止模式（INT\_WUPEN）。
  - 5) 执行 WFI 进入停止模式之前，需确保其他外设中断（非 STOP 模式唤醒中断）都已关闭。否则其他中断的触发可能导致芯片出现不可保证的动作。唤醒后，再恢复其他外设中断状态，以免错过中断事件。
  - 6) 执行 WFI 进入停止模式之前，须确保 HRC 震荡，若系统时钟为 HRC 和 MPLL，还需将系统时钟切换至 MRC，唤醒后再恢复系统时钟，恢复 HRC 状态。
  - 7) 由中断事件将 MCU 从停止模式恢复时，需将系统从 MRC 恢复至用户时钟设定，并还原中断配置
- 其中第 4、5、6、7 点注意事项可通过自行关闭 NVIC 及时钟模块对应寄存器实现，或调用 HDSC 提供的 API 接口：。

➤ `en_result_t enIntWakeupEnable(uint32_t u32WakeupSrc);`（使能中断事件唤醒停止模式）

➤ `void PWC_EnterStopMd(void);`（进入停止模式）

此函数额外占用约 100us@168MHz

➤ `void PWC_IrqClkBackup(void);`（备份用户时钟设置）

此函数额外占用约 50us@168MHz

➤ `void PWC_IrqClkRecover(void)`（恢复用户时钟设置）

此函数额外占用约 50us@168MHz

## 3 样例代码

### 3.1 代码介绍

用户可根据上述的操作流程及注意事项编写自己的代码来学习验证该模块，也可以直接通过华大半导体的网站下载到设备驱动库（Device Driver Library, DDL）的样例代码并使用其中的 LPM 里有关 stop mode 的样例进行验证。

以下部分简要介绍本 AN 基于 DDL 的 LPM 模块样例 lpm\_stop\_wkup 代码所涉及的各项配置。

#### 1) LED 和 PORT 初始化:

本样例使用外部管脚中断唤醒 stop 模式，需对 port 进行初始化，确保对应 port 中断使能。

```
Led_Init();
Port_Init();
```

#### 2) STOP 模式配置:

```
/* Config stop mode. */
stcPwcStopCfg.enStpDrvAbi = StopHighspeed;
stcPwcStopCfg.enStopClk = ClkFix;
stcPwcStopCfg.enStopFlash = Wait;
PWC_StopModeCfg(&stcPwcStopCfg);
```

#### 3) 中断配置:

本样例使用外部管脚 0 中断，上升沿有效，**数字滤波无效**。

```
/* EIRQ0 config. */
stcExintCfg.enExitCh = ExtiCh00;
stcExintCfg.enFilterEn = Disable;
stcExintCfg.enExtiLvl = ExIntRisingEdge;
EXINT_Init(&stcExintCfg);

/* Register EIRQ0.*/
stcPortIrqCfg.enIntSrc = INT_PORT_EIRQ0;
stcPortIrqCfg.enIRQn = PORT_IRQn;
stcPortIrqCfg.pfnCallback = Port_Handle;
enIrqRegistration(&stcPortIrqCfg);
```

#### 4) 设置 STOP 模式中断唤醒源

```
/* Set wake up source EIRQ0. */
enIntWakeupEnable(Extint0WU);
```

## 5) 使能唤醒源中断:

```
/* Enable EIRQ. */  
enIntEnable(Int0);  
NVIC_ClearPendingIRQ(PORT_IRQn);  
NVIC_SetPriority(PORT_IRQn,DDL_IRQ_PRIORITY_DEFAULT);  
NVIC_EnableIRQ(PORT_IRQn);
```

## 6) 进入 STOP 模式前外设状态:

确保 FLASH 不处于编程或者擦除状态, 确保 DMA 处于停止状态, 关闭其他外设中断。

```
/* Ensure DMA disable */  
while((0 != M4_DMA1->EN_f.EN) && ((0 != M4_DMA2->EN_f.EN)));  
/* Ensure FLASH is ready */  
while(1 != M4_EFM->FSR_f.RDY);  
  
PWC_EnterStopMd();
```

## 7) 唤醒后中断处理流程:

```
__IRQ_Handler:  
  
PWC_IrqClkRecover();    // in pairs use  
LED0_TOGGLE();         // user application  
Ddl_Delay1ms(1000);  
PWC_IrqClkBackup();     // in pairs use
```

非停止模式唤醒后的中断无需进行时钟相关配置处理。

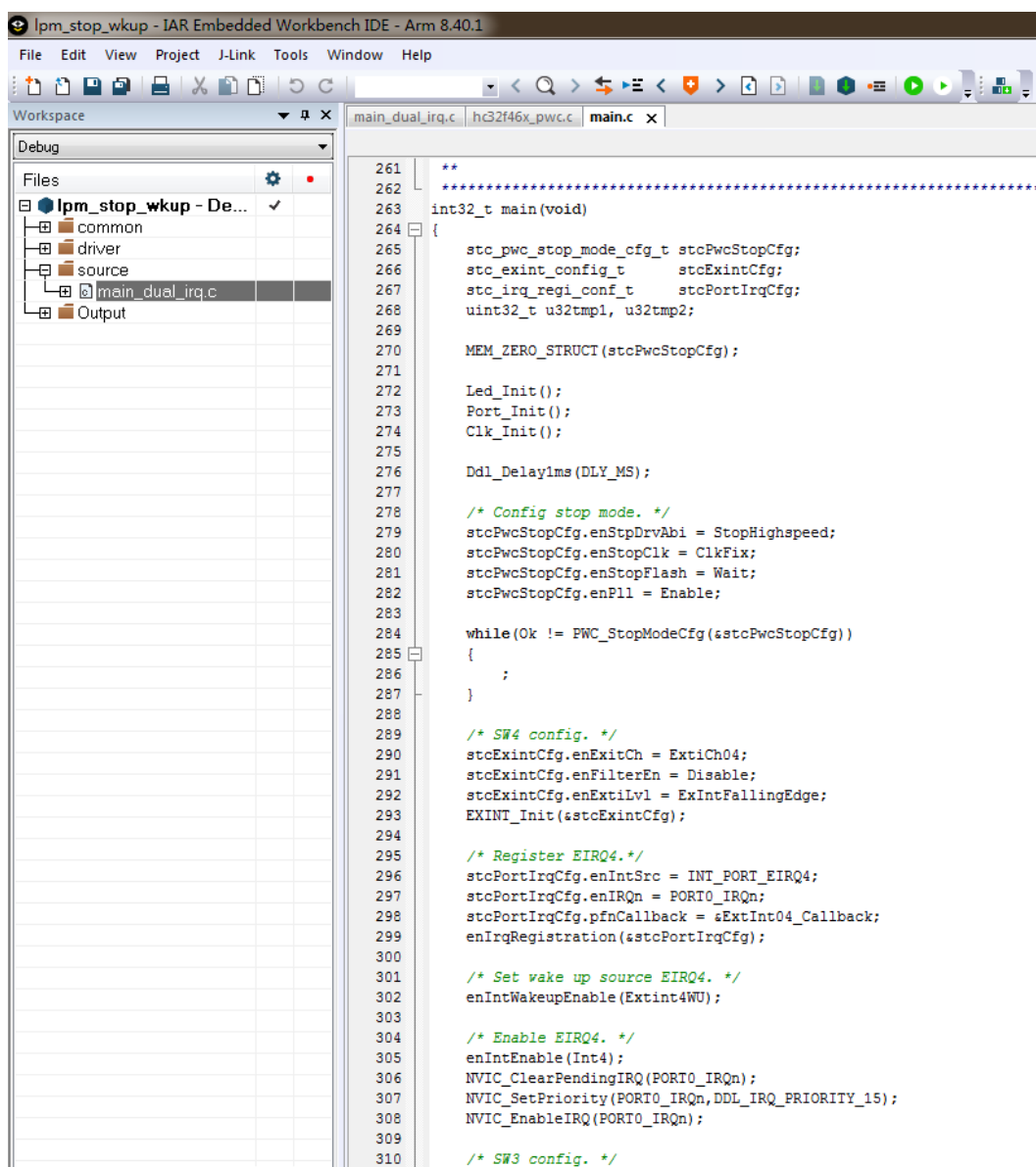


## 3.2 代码运行

用户可以通过华大半导体的网站下载到 HC32F460 的 DDL 的样例代码（lpm\_stop\_wkup），并配合评估用板（EV-HC32F460-LQFP100-050-V1.1）运行相关代码学习使用 STOP 模式。

以下部分主要介绍如何在评估板上运行 STOP 样例代码并观察结果：

- 确认安装正确的 IAR EWARM v7.7 工具（请从 IAR 官方网站下载相应的安装包，并参考用户手册进行安装）。
- 从华大半导体网站下载 HC32F460 DDL 代码。
- 下载并运行 lpm\lpm\_stop\_wkup\中的工程文件：
  - 1) 打开 lpm\_stop\_wkup\工程，并打开‘main.c’如下视图：





The screenshot shows the IAR Embedded Workbench IDE interface. On the left, the 'Workspace' pane displays the project structure for 'lpm\_stop\_wkup - De...'. The 'Files' pane shows a tree view with folders 'common', 'driver', 'source', and 'Output'. The 'main.c' file is selected under the 'source' folder. The main editor window displays the C code for 'main.c', which includes initialization functions like 'Lcd\_Init()', 'Port\_Init()', and 'Clk\_Init()', followed by a loop that configures the STOP mode and registers the EIRQ4 interrupt.

```

261  /**
262  ****
263  int32_t main(void)
264  {
265      stc_pwc_stop_mode_cfg_t stcPwcStopCfg;
266      stc_extint_config_t stcExtIntCfg;
267      stc_irq_regi_conf_t stcPortIrqCfg;
268      uint32_t u32tmp1, u32tmp2;
269
270      MEM_ZERO_STRUCT(stcPwcStopCfg);
271
272      Lcd_Init();
273      Port_Init();
274      Clk_Init();
275
276      Ddl_Delay1ms(DLY_MS);
277
278      /* Config stop mode. */
279      stcPwcStopCfg.enStpDrvAbi = StopHighspeed;
280      stcPwcStopCfg.enStopClk = ClkFix;
281      stcPwcStopCfg.enStopFlash = Wait;
282      stcPwcStopCfg.enPll = Enable;
283
284      while(Ok != PWC_StopModeCfg(&stcPwcStopCfg))
285      {
286          ;
287      }
288
289      /* SW4 config. */
290      stcExtIntCfg.enExitCh = ExtiCh04;
291      stcExtIntCfg.enFilterEn = Disable;
292      stcExtIntCfg.enExtiLvl = ExIntFallingEdge;
293      EXINT_Init(&stcExtIntCfg);
294
295      /* Register EIRQ4. */
296      stcPortIrqCfg.enIntSrc = INT_PORT_EIRQ4;
297      stcPortIrqCfg.enIRQn = PORT0_IRQn;
298      stcPortIrqCfg.pfnCallback = sExtInt04_Callback;
299      enIrqRegistration(&stcPortIrqCfg);
300
301      /* Set wake up source EIRQ4. */
302      enIntWakeUpEnable(Extint4WU);
303
304      /* Enable EIRQ4. */
305      enIntEnable(Int4);
306      NVIC_ClearPendingIRQ(PORT0_IRQn);
307      NVIC_SetPriority(PORT0_IRQn, DDL_IRQ_PRIORITY_15);
308      NVIC_EnableIRQ(PORT0_IRQn);
309
310      /* SW3 config. */

```

- 2) 点击  重新编译整个项目。
- 3) 点击  将代码下载到评估板上,全速运行。
- 4) 按下 SW2, 芯片进入 stop 模式。
- 5) 按下 SW4, 芯片被唤醒, 红灯闪烁。芯片继续进入 stop 模式。
- 6) 按下 SW3, 芯片被唤醒, 绿灯闪烁。芯片继续进入 stop 模式。

可用示波器观察时钟、红灯、绿灯波形图, 其中 SW3 中断优先级高于 SW4, 在 SW4 中断执行过程中, SW3 中断触发, SW4 中断会被抢占, 此时可观察到红灯绿灯同时亮, 然后绿灯、红灯相继熄灭。

## 4 版本信息 & 联系方式

日期	版本	修改记录
2019/3/20	Rev1.0	初版发布
2020/8/26	Rev1.1	增加进入和退出停止模式的时钟切换说明；更新支持型号



---

如果您在购买与使用过程中有任何意见或建议，请随时与我们联系。

Email: [mcu@hdsc.com.cn](mailto:mcu@hdsc.com.cn)

网址: <http://www.hdsc.com.cn/mcu.htm>

通信地址: 上海市浦东新区中科路 1867 号 A 座 10 层

邮编: 201203

---

