32-bit Microcontrollers

Stop Mode Precautions for HC 32 F 460 Series

Applicable objects

Series	Product Model
HC32F460	HC32F460JEUA HC32F460JETA HC32F460KEUA HC32F460KETA HC32F460PETB

Table of Contents

Table of Contents

1 Abstract

This application note introduces the stop mode notes of HC32F460 series chips.

2 Stop Mode of HC32F460 Series

2.1 Introduction

Stop (STOP) mode is one of the three low-power modes of HC32F460 series chips. In the stop mode, the CPU, most peripherals and clock sources are stopped, and the chip maintains the CPU internal registers and SRAM data, peripheral status and pin status.

2.2 Register Introduction

1) PWR_PWRC1: Power mode control register, set the drive capability when different modes go into STOP mode.

2) PWR_STPMCR: Stop mode controller to enable STOP mode and set the CLK and FLASH states when STOP mode wakes up.

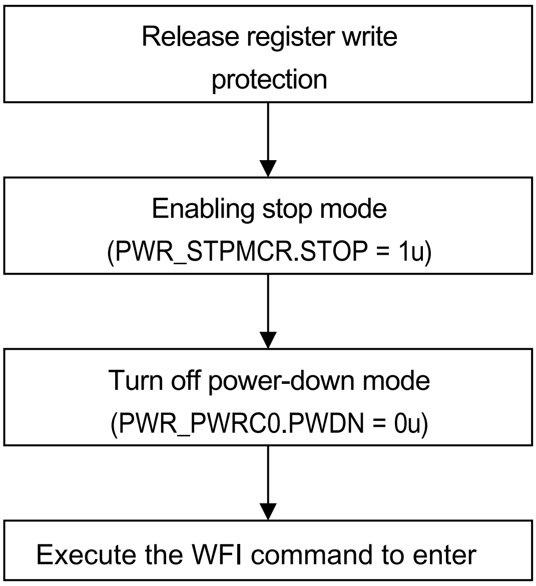
3) INT_WUPEN: Stop mode wake-up event

enable register. Caution:

- You must set INT_WUPEN to enable the corresponding wake-up event before entering the stop mode, otherwise you cannot wake up the stop mode.

2.3 Operation process

2.3.1 Enter stop mode


Release register write protection
Enabling stop mode (PWR_STPMCR.STOP = 1u)
Turn off power-down mode (PWR_PWRC0.PWDN = 0u)
Execute the WFI command to enter

The diagram above lists the steps to enter the stop mode, in which both the peripherals and the CPU of the chip stop working.

2.3.2 Release stop mode

Stop mode can be released by

resetting and interrupting. The resets

that can be used to disengage the stop

mode are

- Pin Reset
- Power-on reset
- Undervoltage Reset (BOR)
- Voltage detection 1/2 reset
- Dedicated watchdog reset

The interrupt events that can be used to release the stop mode are

- Pin interrupt EIRQ0~15
- Voltage detection 1/2 interrupt
- Dedicated watchdog underflow interrupt

- Periodic interrupt of the real time clock

- Alarm clock interruption

- Wake-up timer interrupt
- Comparator 1 Interrupt
- UART1_RXD0 Interrupt
- TIMER01_A Compare

Match Interrupt Attention:

- To select the interrupt event unstop mode, you need to operate the INT_WUPEN register to set the corresponding interrupt event wake-up permission before entering the stop mode.

2.4 Cautions

1) Before executing WFI into stop mode, make sure that the FLASH is not programmed or erased (i.e.

   EFM_FSR.RDY=1), and the oscillation stop monitoring function is invalid, otherwise the chip cannot enter the stop mode.

2) Before executing WFI into stop mode, make sure that the DMA is in stop state (i.e. DMA_EN.EN=0), otherwise the chip may act in an unguaranteed way.

3) Before executing WFI into stop mode, the digital filtering of EIRQ must be set to invalid, otherwise the interrupt cannot be used

   STOP wakes up.

4) Select the interrupt event to release the stop mode and enable the corresponding interrupt event to wake up the stop mode (INT_WUPEN) before executing the WFI into the stop mode.

5) Make sure all other peripheral interrupts (non-STOP mode wake-up interrupts) are turned off before executing WFI into stop mode. Otherwise, the triggering of other interrupts may lead to unguaranteed actions of the chip. After waking up, restore other peripheral interrupts to avoid missing interrupt events.

Caution:

- The notes for points 4 and 5 can be done by turning off the NVIC corresponding register bits by yourself or by calling the API interfaces provided by HDSC (enIntWakeupEnable (enable interrupt event wakeup stop mode), enNvicBackup (turn off other peripheral interrupts), enNvicRecover (resume other peripheral interrupts)).

3 Sample Code

3.1 Code Introduction

Users can write their own code to learn and verify the module according to the above operation flow and notes, or download the sample code of Device Driver Library (DDL) directly through the website of UW Semiconductors and use the sample of stop mode in LPM to verify.

The following section briefly describes the configuration involved in this sample AN DDL-based LPM module lpm_stop_wkup code.

1) LED and PORT initialization:

This sample uses an external pin interrupt to wake up the stop mode, so the port must be initialized to ensure that the corresponding port interrupt is enabled.

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

2) STOP mode configuration:

```
/* Config s top mode. */
stcPwcStopCfg.enStpDrvAbi = StopHighspeed;
stcPwcStopCfg.enStopClk = ClkFix;
stcPwcStopCfg.enStopFlash = Wait; PWC_
StopModeCfg(&stcPwcStopCfg).
```

3) Interrupt configuration:

This sample uses the external pin 0 interrupt, which is valid on the rising edge and **invalid for digital filtering**.

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

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

4) Set STOP mode interrupt wake-up source

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

5) Enables wake-up source interrupts:

```
/* Enable EIRQ.
*/
enIntEnable(Int0).
NVIC_ClearPendingIRQ(PORT_IRQn);
NVIC_SetPriority(PORT_IRQn,D DL_IRQ_PRIORITY_DEFAULT).
```

6) Peripheral status before entering STOP mode:

Ensure that FLASH is not programmed or erased, that DMA is stopped, and that other peripheral interrupts are turned off.

```
/* 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).

/* NVIC backup and disable before entry from stop mode.*/
enNvicBackup().
PWC_EnterStopMd().
```

7) Resume other peripheral interrupts after wake-up:

```
/* NVIC recover after wakeup from stop mode. */
enNvicRecover().
LED0_TOGGLE().
Ddl_Delay1ms(1000).
```

3.2 Code Run

Users can download the sample code (lpm_stop_wkup) of the HC32F460 DDL from the UW website and run the code with the evaluation board (EV-HC32F460-LQFP100-050-V1.1) to learn to use STOP mode.

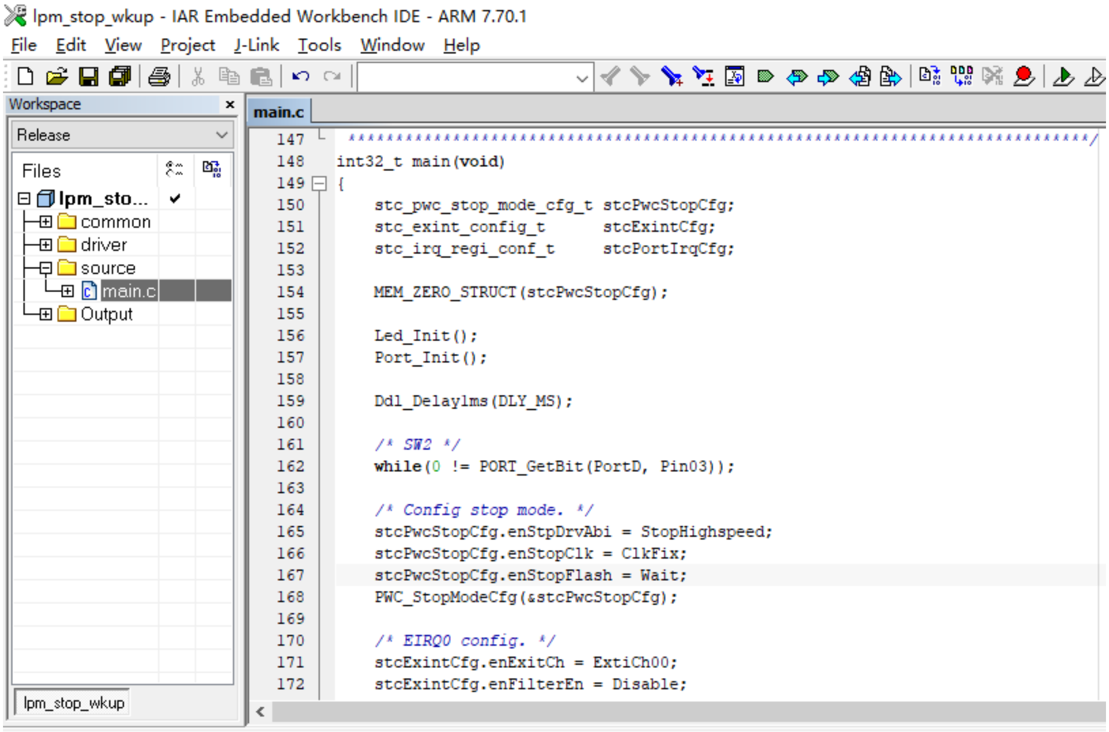
The following section describes how to run the STOP sample code on the evaluation board and observe the results:

— Verify that the correct IAR EWARM v7.7 tool is installed (please download the appropriate installation package from the official IAR website and refer to the user manual for installation).

— Download the HC32F460 DDL code from the UW Semiconductors website.

— Download and run the project file in lpm\ lpm_stop_wkup\ at

1) Open the lpm_stop_wkup \ project and open the 'main.c' view as follows:


lpm_stop_wkup - IAR Embedded Workbench IDE - ARM 7.70.1
File Edit View Project J-Link Tools Window Help
Workspace
Release
Files
lpm_sto...
common
driver
source
main.c
Output
lpm_stop_wkup
main.c

```
 ***************************************************************************/
int32_t main(void)
{
    stc_pwc_stop_mode_cfg_t stcPwcStopCfg;
    stc_exint_config_t      stcExintCfg;
    stc_irq_regi_conf_t     stcPortIrqCfg;

    MEM_ZERO_STRUCT(stcPwcStopCfg);

    Led_Init();
    Port_Init();

    Ddl_Delay1ms(DLY_MS);

    /* SW2 */
    while(0 != PORT_GetBit(PortD, Pin03));

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

    /* EIRQ0 config. */
    stcExintCfg.enExitCh = ExtiCh00;
    stcExintCfg.enFilterEn = Disable;
```

2) Click to recompile the entire project.

3) Click Download the code to the evaluation board and run it at full speed.

4) Press SW2, you can observe the red light blinking and the chip enters the stop mode (debug state, IDE cannot connect to the chip).

5) Connect PA0 pin to VCC (gives PA0 rising edge to wake up the chip).

6) A flashing red light can be observed and the chip is woken up;

7) Continuing to give PA0 a rising edge, the chip falls into sleep and the next rising edge, the chip is woken up.

4 Version Information & Contact

Date	Versions	Modify records
2019/3/20	Rev1.0	Initial Release

If you have any comments or suggestions in the process of purchase and use, please feel free to contact us.

Email: mcu@hdsc.com.cn

Website: http://www.hdsc.com.cn/mcu.htm

Address: 39, Lane 572, Bibo Road, Zhangjiang Hi-

Tech Park, Shanghai, 201203, P.R. China