GigaDevice Semiconductor Inc.

Arm® Cortex®-M3/4 32-bit MCU

Application Note
AN008

Table of Contents

List of Tables

1. Introduction

There is a limitation when using deep-sleep mode with GD32 series MCUs. If user enable some interrupts which are not the target wake-up source, when using deep-sleep mode, regardless of the use of WFI or WFE instruction, there will be a serious risk, that after entering deep-sleep mode, MCU may never be awakened by the target signal.

2. Workarounds

In order to avoid this risk, the user should replace the font-weight part of the function pmu_to_deepsleepmode in the ***Table 2-1. original function of GD standard lib***. The function could be found in the file named similar as “gd32fxxx_pmu.c” in our standard library.

The user needs to select the corresponding new codes according to MCU product type.

Table 2-1. original function of GD standard lib

```
/*!
    \brief        PMU work at deep sleep mode
    \param[in]    ldo
      \arg          PMU_LDO_NORMAL: LDO normal work when pmu enter deepsleep mode
      \arg          PMU_LDO_LOWPOWER: LDO work at low power mode when pmu enter deepsleep mode
    \param[in]    deepsleepmodecmd:
      \arg          WFI_CMD: use WFI command
      \arg          WFE_CMD: use WFE command
    \param[out] none
    \retval       none
*/
void pmu_to_deepsleepmode(uint32_t ldo,uint8_t deepsleepmodecmd)
{
    /* clear stbmod and ldolp bits */
    PMU_CTL &= ~((uint32_t)(PMU_CTL_STBMOD | PMU_CTL_LDOLP));

    /* set ldolp bit according to pmu_ldo */
    PMU_CTL |= ldo;

    /* set sleepdeep bit of Cortex-M4 system control register */
    SCB->SCR |= SCB_SCR_SLEEPDEEP_Msk;

    /* select WFI or WFE command to enter deepsleep mode */
    if(WFI_CMD == deepsleepmodecmd){
        __WFI();
    }else{
        __SEV();
        __WFE();
        __WFE();
    }
    /* reset sleep deep bit of Cortex-M4 system control register */
    SCB->SCR &= ~((uint32_t)SCB_SCR_SLEEPDEEP_Msk);
}
```

2.1. The corresponding replacement codes for GD32F10x / F20x / F30x / F403 series lib

Table 2-2. Replacement codes for GD32F10x / F20x / F30x / F403

```
{
    static uint32_t reg_snap[4];

    reg_snap[0] = REG32(0xE000E010);
    reg_snap[1] = REG32(0xE000E100);
    reg_snap[2] = REG32(0xE000E104);
    reg_snap[3] = REG32(0xE000E108);

    REG32(0xE000E010) &= 0x00010004;
    REG32(0xE000E180)   = 0XFF7FF83D;
    REG32(0xE000E184)   = 0XBFFFF8FF;
    REG32(0xE000E188)   = 0xFFFFFFFF;

    /* select WFI or WFE command to enter deepsleep mode */
    if(WFI_CMD == deepsleepmodecmd){
        __WFI();
    }else{
        __SEV();
        __WFE();
        __WFE();
    }
    REG32(0xE000E010) = reg_snap[0] ;
    REG32(0xE000E100) = reg_snap[1] ;
    REG32(0xE000E104) = reg_snap[2] ;
    REG32(0xE000E108) = reg_snap[3] ;
}
```

2.2. The corresponding replacement codes for GD32F1x0 / F3x0 series lib

Table 2-3. Replacement codes for GD32F1x0 / F3x0

```
{
    static uint32_t reg_snap[4];

    reg_snap[0] = REG32(0xE000E010);
    reg_snap[1] = REG32(0xE000E100);
    reg_snap[2] = REG32(0xE000E104);
    reg_snap[3] = REG32(0xE000E108);

    REG32(0xE000E010) &= 0x00010004;
    REG32(0xE000E180)   = 0XB7FFEF19;
    REG32(0xE000E184)   = 0XFFFFFBFF;
    REG32(0xE000E188)   = 0xFFFFFFFF;

    /* select WFI or WFE command to enter deepsleep mode */
    if(WFI_CMD == deepsleepmodecmd){
        __WFI();
    }else{
        __SEV();
        __WFE();
        __WFE();
    }

    REG32(0xE000E010) = reg_snap[0] ;
    REG32(0xE000E100) = reg_snap[1] ;
    REG32(0xE000E104) = reg_snap[2] ;
    REG32(0xE000E108) = reg_snap[3] ;
}
```

2.3. The corresponding replacement codes for GD32F40x series lib

Table 2-4. Replacement codes for GD32F40x

```
{
    static uint32_t reg_snap[4];

    reg_snap[0] = REG32(0xE000E010);
    reg_snap[1] = REG32(0xE000E100);
    reg_snap[2] = REG32(0xE000E104);
    reg_snap[3] = REG32(0xE000E108);

    REG32(0xE000E010) &= 0x00010004;
    REG32(0xE000E180)   = 0XFF7FF83D;
    REG32(0xE000E184)   = 0XBFFFF8FF;
    REG32(0xE000E188)   = 0xFFFFFFFF;

    /* select WFI or WFE command to enter deep sleep mode */
    if(WFI_CMD == deepsleepmodecmd){
        __WFI();
    }else{
        __SEV();
        __WFE();
        __WFE();
     }

    REG32(0xE000E010) = reg_snap[0] ;
    REG32(0xE000E100) = reg_snap[1] ;
    REG32(0xE000E104) = reg_snap[2] ;
    REG32(0xE000E108) = reg_snap[3] ;
}
```

3. Revision history

Table 3-1. Revision history

Revision No.	Description	Date
1.0	Initial Release	Nov.30 2021

Important Notice

This document is the property of GigaDevice Semiconductor Inc. and its subsidiaries (the "Company"). This document, including any product of the Company described in this document (the “Product”), is owned by the Company under the intellectual property laws and treaties of the People’s Republic of China and other jurisdictions worldwide. The Company reserves all rights under such laws and treaties and does not grant any license under its patents, copyrights, trademarks, or other intellectual property rights. The names and brands of third party referred thereto (if any) are the property of their respective owner and referred to for identification purposes only.

The Company makes no warranty of any kind, express or implied, with regard to this document or any Product, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Company does not assume any liability arising out of the application or use of any Product described in this document. Any information provided in this document is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Except for customized products which has been expressly identified in the applicable agreement, the Products are designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only. The Products are not designed, intended, or authorized for use as components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, atomic energy control instruments, combustion control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or Product could cause personal injury, death, property or environmental damage ("Unintended Uses"). Customers shall take any and all actions to ensure using and selling the Products in accordance with the applicable laws and regulations. The Company is not liable, in whole or in part, and customers shall and hereby do release the Company as well as it’s suppliers and/or distributors from any claim, damage, or other liability arising from or related to all Unintended Uses of the Products. Customers shall indemnify and hold the Company as well as it’s suppliers and/or distributors harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of the Products.

Information in this document is provided solely in connection with the Products. The Company reserves the right to make changes, corrections, modifications or improvements to this document and Products and services described herein at any time, without notice.