M480 Power Mode

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	M480 Power Mode example code
BSP Version	M480 Series BSP CMSIS V3.04.000
Hardware	NuMaker-ETM-M487

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1 Function Description

1.1 Introduction

This example code shows how to set and enter different power down mode.

1.2 Principle

This example shows how to enter different power mode and how to set the wake up source.

1.3 Demo Result

After executing the program, user can set different power down mode and measure the wake up time.

Wake-Up Source	Wake-Up Condition	Power-Down Mode		
		NPD / FWPD / LLPD	SPD0/1	DPD
BOD	Brown-Out Detector Reset / Interrupt	V	-	-
	Brown-Out Detector Reset	-	V	-
LVR	LVR Reset	V	V	-
POR	POR Reset	V	V	-
INT	External Interrupt	V	-	-
GPIO	GPIO Interrupt	V	-	-
GPIO(PA~PD) Wake-up pin	rising or falling edge event, 64-pin	-	V	-
GPIO(PC.0) Wake-up pin	rising or falling edge event , 1-pin	-	-	V
TIMER	Timer Interrupt	V	-	-
Wakeup timer	Wakeup by wake-up timer time-out	-	V	V
WDT	WDT Interrupt	V	-	-
RTC	Alarm Interrupt	V	-	-
	Time Tick Interrupt	V	-	-
	Wakeup by RTC alarm	-	V	V
	Wakeup by RTC tick time	-	V	V
	Wakeup by tamper event	-	V	V
UART	nCTS wake-up	V	-	-
	RX Data wake-up	V	-	-
	Received FIFO Threshold Wake-up	V	-	-
	RS-485 AAD Mode Wake-up	V	-	-
	Received FIFO Threshold Time-out Wake-up	V	-	-

USCI UART	CTS Toggle	V	-	-
	Data Toggle	V	-	-
USCI I2C	Data toggle	V	-	-
	Address match	V	-	-
USCI SPI	SS Toggle	V	-	-
I^2C	Address match wake-up	V	-	-
USBD	Remote Wake-up	V	-	-
ACMP	Comparator Power-Down Wake-Up Interrupt	V	-	-
	ACMPO status change	-	V	-

2 Code Description

Get ADC data and send data out by PWM and DAC

```
/*---------------------------------------------------------------------------------------*/
/*  Main Function
*/
/*---------------------------------------------------------------------------------------*/
int32_t main(void)
{
    uint8_t u8Item;

    /* Unlock protected registers */
    SYS_UnlockReg();

    /* Release I/O hold status */
    CLK->IOPDCTL = 1;

    /* Init System, peripheral clock and multi-function I/O */
    SYS_Init();

    /* ---------- Turn off RTC  -------- */
    CLK->APBCLK0 |= CLK_APBCLK0_RTCCKEN_Msk;
    RTC_WaitAccessEnable();
    RTC->INTEN = 0;
    RTC_Close();
    /* Init UART0 for printf */
    UART0_Init();

    /* Get power manager wake up source */
    //CheckPowerSource();

    /* To check if system has been reset by WDT time-out reset or not */
    if(WDT_GET_RESET_FLAG() == 1) {
        WDT_CLEAR_RESET_FLAG();
        printf("\n*** System has been reset by WDT time-out event ***\n");
    }
    else if(M32(FLAG_ADDR) == SIGNATURE) {
        printf("\nSystem waken-up from SPD0 mode done!\n");
                printf("SRAM retention:%d\n", M32(FLAG_ADDR));
        M32(FLAG_ADDR) = 0;
```

```
    }
         //else
        //printf("\nSystem waken-up from SPD1 mode done!\n");


    CLK_EnableCKO(CLK_CLKSEL1_CLKOSEL_HCLK, 0, 1);

         while(1){

    switch(u8Item) {
    case '1'://Turbo mode.
        Turbo_Function();
        break;
    case '2'://Turbo mode.
        Normal_Function();
        break;
    case '3'://Idle mode Uart.
        IDLE_UARTFunction();
        break;
    case '4'://Fast Wake-up Power Down Wake-up Watch Dag.
        FWUPDWakeUpWDTFunction(CLK_PMUCTL_PDMSEL_FWPD);
        break;
    case '5'://Power Down Wake-up UART2 wake up.
        PDWakeUpUARTFunction(CLK_PMUCTL_PDMSEL_PD);
        break;
    case '6'://Power Down Wake-up UART2 wake up.
        PDFunction_USBD(CLK_PMUCTL_PDMSEL_PD);
        break;
    case '7'://Low Leakage Power Down Wake-up TIMER time-out interval is 1024 OSC10K
clocks.
        LLPDWakeUpTimerFunction(CLK_PMUCTL_PDMSEL_LLPD, CLK_PMUCTL_WKTMRIS_1024);
        break;
    case '8'://SPD0 GPIO Wake-up pin(PA.0) and using rising edge wake up.
        SPD0WakeUpPinFunction(CLK_PMUCTL_PDMSEL_SPD0);
        break;
    case '9': //SPD1 Wake-up by RTC Tamper0, Low level.
        SPD1WakeUpRTCTickFunction(CLK_PMUCTL_PDMSEL_SPD1);
        break;
    case 'a':
        DPDWakeUpPC0Function(CLK_PMUCTL_PDMSEL_DPD, CLK_DPDWKPIN_RISING);
        break;
    case 'b'://Turbo mode.
```

```
        Turbo_Function_basic();
        break;
    case 'c'://Turbo mode.
        Normal_Function_basic();
        break;
    case 'd'://Idle mode Uart.
        IDLE_Function_basic();
        break;
    case 'e'://Fast Wake-up Power Down Wake-up Watch Dag.
        FWUPDFunction_basic(CLK_PMUCTL_PDMSEL_FWPD);
        break;
    case 'f'://Power Down Wake-up UART2 wake up.
        PDFunction_basic(CLK_PMUCTL_PDMSEL_PD);
        break;
    case 'g'://Low Leakage Power Down Wake-up TIMER time-out interval is 1024 OSC10K
clocks.
        LLPDFunction_basic(CLK_PMUCTL_PDMSEL_LLPD, CLK_PMUCTL_WKTMRIS_1024);
        break;
    case 'h'://SPD0 GPIO Wake-up pin(PA.0) and using rising edge wake up.
        SPD0Function_basic(CLK_PMUCTL_PDMSEL_SPD0);
        break;
    case 'i': //SPD1 Wake-up by RTC Tamper0, Low level.
        SPD1Function_basic(CLK_PMUCTL_PDMSEL_SPD1);
        break;
    case 'j':
        DPDFunction_basic(CLK_PMUCTL_PDMSEL_DPD, CLK_DPDWKPIN_RISING);
        break;
    default:
        break;
     }



         printf("\n\nCPU @ %d Hz\n\n", SystemCoreClock);
    printf("+------------------------------------------------------------------------
+\n");
    printf("|    M480 Power-down Mode and Wake-up Sample Code
|\n");
    printf("|    Please Select Power Down Mode and Wake up source.
|\n");
    printf("+------------------------------------------------------------------------
+\n");
    printf("|[1] Turbo mode with Core Freq 192MHz and LDO 1.26V.
|\n");
```

```
    printf("|[2] Normal mode with Core Freq 160MHz and LDO 1.2V.
|\n");
    printf("|[3] IDLE mode with Uart0 transfer ADC5(PIN1) data(8 bits) by PDMA.
|\n");
    printf("|[4] Fast wake up power down mode with LIRC and WDT wakeup after 10s.
|\n");
    printf("|[5] Power down mode with LXT and UART(PIN66) wakeup.
|\n");
         printf("|[6] Power down mode with USB D+,D- wakeup.
|\n");
    printf("|[7] Low Leakage power down mode with LIRC and Timer wakeup after 10s.
|\n");
    printf("|[8] SPD0 Wake-up by GPIO PC.0(PIN82) with data reteantion.
|\n");
    printf("|[9] SPD1 Wake-up by RTC Alarm after 1s.
|\n");
    printf("|[a] DPD1 Wake-up by GPIO PC.0(PIN82).
|\n");
    printf("|[b] Turbo mode with Core Freq 192MHz, LDO 1.26V and other function closed.
|\n");
    printf("|[c] Normal mode with Core Freq 160MHz, LDO 1.2V and other function closed.
|\n");
    printf("|[d] IDLE mode with other function closed.
|\n");
    printf("|[e] Fast wake up power down mode with other function closed.
|\n");
    printf("|[f] Power down mode with other function closed.
|\n");
    printf("|[g] Low Leakage power down mode with other function closed.
|\n");
    printf("|[h] SPD0 with other function closed.
|\n");
    printf("|[i] SPD1 with other function closed.
|\n");
    printf("|[j] DPD1 with other function closed.
|\n");
    printf("+-------------------------------------------------------------------------
+\n");


    /* To check if all the debug messages are finished */
         u8Item = getchar();
             while(IsDebugFifoEmpty() == 0);


    }
}
```

3 Hardware and Software environment

- **Software Environment**
  - BSP version
    - M480 Series BSP CMSIS V3.04.000
  - IDE version
    - Keil uVersion 5.26
- **Hardware Environment**
  - Circuit components
    - NuMaker-ETM-M487
  - Diagram
    - N/A

4 Directory Information

EC_M480_Power_Mode _V1.00

Library	Sample code header and source files
CMSIS	Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
Device	CMSIS compliant device header file
StdDriver	All peripheral driver header and source files
SampleCode	
ExampleCode	Source file of example code

5 How to execute a sample code

1. Enter Keil compile mode
    a. Build
    b. Download
    c. Start/Stop debug session
2. Enter debug mode
    a. Run

6 Revision History

Date	Revision	Description
Nov 14, 2019	1.00	1. Initially issued.