M480 FreeRTOS LED Flash

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	This code is a basic application for FreeRTOS task creating to flash LED.
BSP Version	M480 Series BSP CMSIS V3.04.000
Hardware	NuMaker-PFM-M487

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

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

www.nuvoton.com

1 Function Description

1.1 Introduction

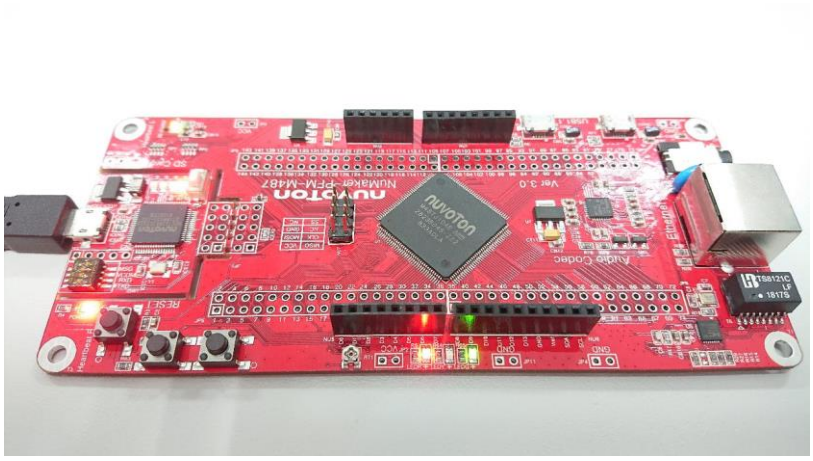
This code is a basic application for FreeRTOS task creating to flash LED.

1.2 Principle

In FreeRTOS, each thread of execution is called a ‘task’. task.c and task.h implement API functions for establishment, scheduling and maintenance tasks.

This FreeRTOS sample code creates three tasks, each of which flash an LED at a different rate. The first LED flashes every 333ms, the second every 666ms, the third every 999ms. The LED flash tasks provide instant visual feedback. The LEDs show that the FreeRTOS scheduler is still operational.

1.3 Demo Result



2 Code Description

Create three tasks for LED flash：

```
void vStartLEDFlashTasks( UBaseType_t uxPriority )
{
BaseType_t xLEDTask;

    /* Create the three tasks. */
    for( xLEDTask = 0; xLEDTask < ledNUMBER_OF_LEDS; ++xLEDTask )
    {
        /* Spawn the task. */
        xTaskCreate( vLEDFlashTask, "LEDx", ledSTACK_SIZE, NULL, uxPriority,
        ( TaskHandle_t * ) NULL );
    }
}
```

Set LED flash rate in the task function：

```
static portTASK_FUNCTION( vLEDFlashTask, pvParameters )
{
TickType_t xFlashRate, xLastFlashTime;
UBaseType_t uxLED;

    /* The parameters are not used. */
    ( void ) pvParameters;

    /* Calculate the LED and flash rate. */
    portENTER_CRITICAL();
    {
        /* See which of the LED's we should use. */
        uxLED = uxFlashTaskNumber;

        /* Update so the next task uses the next LED. */
        uxFlashTaskNumber++;
    }
    portEXIT_CRITICAL();

    xFlashRate = ledFLASH_RATE_BASE + ( ledFLASH_RATE_BASE * ( TickType_t ) uxLED );
    xFlashRate /= portTICK_PERIOD_MS;
```

```
    /* We will turn the LED on and off again in the delay period, so each
    delay is only half the total period. */
    xFlashRate /= ( TickType_t ) 2;

    /* We need to initialise xLastFlashTime prior to the first call to
    vTaskDelayUntil(). */
    xLastFlashTime = xTaskGetTickCount();

    for(;;)
    {
        /* Delay for half the flash period then turn the LED on. */
        vTaskDelayUntil( &xLastFlashTime, xFlashRate );
        vParTestToggleLED( uxLED );

        /* Delay for half the flash period then turn the LED off. */
        vTaskDelayUntil( &xLastFlashTime, xFlashRate );
        vParTestToggleLED( uxLED );
    }
}
```

3 Software and Hardware Environment

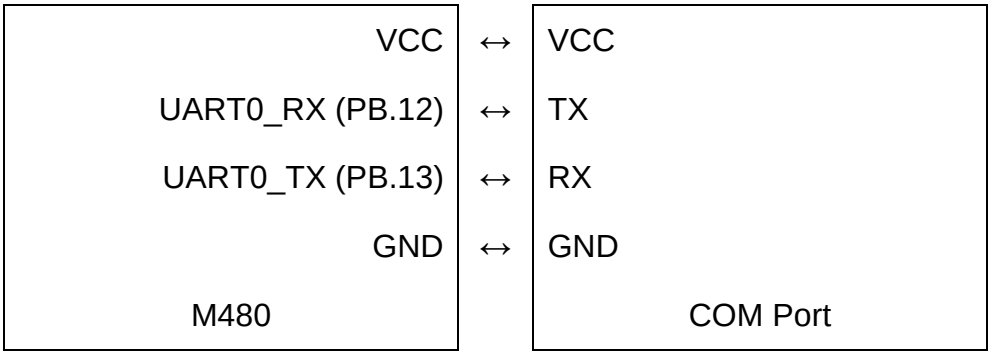
Software Environment

- BSP version
  - M480 Series BSP CMSIS V3.04.000
- IDE version
  - Keil uVersion 5.26

Hardware Environment

- Circuit components
  - NuMaker-PFM-M487 v3.0
- Diagram

Using printf function to output message and the UART0 TX RX pin are PB.13 and PB.12 respectively.



4 Directory Information

- EC_M480_FreeRTOS_LED_Flash_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
  - ThirdParty
    - FreeRTOS — A real time operating system available for free download. Its official website is: http://www.freertos.org/

5 How to Execute Example Code

1. Browsing into sample code folder by Directory Information (section 4) and double click M480_FreeRTOS_LED_Flash.uvproj.
2. Enter Keil compile mode
   a. Build
   b. Download
   c. Start/Stop debug session
3. Enter debug mode
   a. Run

6 Revision History

Date	Revision	Description
Jul. 11, 2019	1.00	1. Initially issued.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.