TIMER Capture Both Edge

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	This sample code uses the TIMER to capture both rising edge and falling edge of the external signal
BSP Version	M0518 Series BSP CMSIS V3.00.002
Hardware	NuTiny-M0518S-V1.1

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 sample code uses the TIMER to capture both rising edge and falling edge of the external signal and calculates the high level duration time base on TIMER clock.

1.2 Principle

The TIMER interrupt can be triggered by rising edge or falling edge or both edge of external signal. This sample code demonstrates how to use the both edge to trigger the interrupt function to capture the current TIMER counter and further calculate the high level duration with the TIMER clock.

For demonstration purposes, the sample code will generate a pulse signal and output it via GPIO PB.14 instead of a true external signal. Before executing the sample code, please connect PB.14 and PB.15 with jumper or wire. In actual use, you can connect the real external signal directly to PB.15.

1.3 Demo Result

```
CPU @ 50000000 Hz
+----------------------------------------+
|    TIMER0 Capture Counter Sample Code    |
+----------------------------------------+

  * Please connect PB14 and PB15 by wire.
TIMER0 clock @ 25000000 Hz
Capture counter for rising edge: 0x2 (2)
Capture counter for falling edge: 0x61D2 (25042)

Capture counter for high level duration: 0x61D0 (25040)
The high level duration time should be about (TIMER0 clock) * 25040
                                            = 1001 us
```

2 Code Description

Configure TIMER0 to capture mode, TIMER0 clock to HCLK/2, and interrupt trigger source to both rising edge and falling edge:

```
    /*----- Initial TIMER0 -----*/

    SYS_ResetModule(TMR0_RST);
    TIMER_Open(TIMER0, TIMER_CONTINUOUS_MODE, 1);

    /* Configure TIMER0 setting for external counter input and capture function */
    /* Set TIMER0 clock to HCLK/(1+1) */
    TIMER_SET_PRESCALE_VALUE(TIMER0, 1);
    /* Reset TIMER0 counter value */
    TIMER_SET_CMP_VALUE(TIMER0, 0xFFFFFF);
    /* Set TIMER0 to capture both edges */
    TIMER_EnableCapture(TIMER0, TIMER_CAPTURE_FREE_COUNTING_MODE,
TIMER_CAPTURE_FALLING_AND_RISING_EDGE);
    /* Enable TIMER0 capture interrupt */
    TIMER_EnableCaptureInt(TIMER0);
    NVIC_EnableIRQ(TMR0_IRQn);
```

TIMER0 interrupt handler will capture the TIMER0 counter when both rising edge and falling edge:

```
void TMR0_IRQHandler(void)
{
    if (TIMER_GetCaptureIntFlag(TIMER0) == 1)
    {
        /* Clear TIMER0 capture interrupt flag */
        TIMER_ClearCaptureIntFlag(TIMER0);

        if (PB15)
        {
            /* Save the capture counter for PB15 rising edge */
            g_au32TMRCapCountRise = TIMER_GetCaptureData(TIMER0);
        }
        else
        {
            /* Save the capture counter for PB15 falling edge */
            g_au32TMRCapCountFall = TIMER_GetCaptureData(TIMER0);
            g_u32TimerCapFlag = 1;
        }
    }
}
```

Generate a pulse signal on PB.14 to create rising edge and falling edge to trigger TIMER0 interrupt handler. At last, we can calculate high level duration time base on TIMER0 counter and TIMER0 clock.

```
/*---- Begin to test -----*/
    g_au32TMRCapCountRise = g_au32TMRCapCountFall = g_u32TimerCapFlag = 0;

    /* Start TIMER0 counting */
    TIMER_Start(TIMER0);

    /* Generate a pulse with both rising edge and falling edge to PB14 */
    PB14 = 1;   /* Rising edge */

    /* Delay to keep high level of pulse. */
    /* Change the delay counter will get different capture counter value */
    for (u32delayCounter = 0; u32delayCounter < 10000; u32delayCounter++);

    PB14 = 0;   /* Falling edge */

    /* Wait a rising edge and a falling edge from PB15 */
    /* TIMER0 interrupt handler will set g_u32TimerCapFlag to 1 when received */
    /*      external capture signal from PB15 */
    while (g_u32TimerCapFlag == 0);

    /* Got the counter value for external capture signal */
    if (g_au32TMRCapCountFall > g_au32TMRCapCountRise)
        u32CAPValue = g_au32TMRCapCountFall - g_au32TMRCapCountRise;
    else
        /* In the case that TIMER0 counter overflowed */
        u32CAPValue = 0xffffff - g_au32TMRCapCountRise + g_au32TMRCapCountFall;

    printf("Capture counter for rising edge: 0x%X (%d)\n", g_au32TMRCapCountRise,
g_au32TMRCapCountRise);
    printf("Capture counter for falling edge: 0x%X (%d)\n", g_au32TMRCapCountFall,
g_au32TMRCapCountFall);
    printf("\n");
    printf("Capture counter for high level duration: 0x%X (%d)\n", u32CAPValue,
u32CAPValue);
    printf("The high level duration time should be about (TIMER0 clock) * %d\n",
u32CAPValue);
    printf("                                                = %d us\n", u32CAPValue /
(SystemCoreClock / 2 / 1000000));
```

3 Software and Hardware Environment

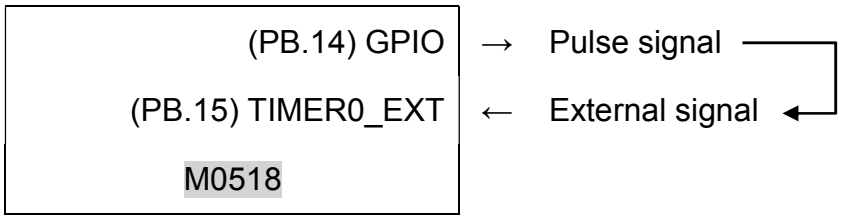
Software Environment

- BSP version
  - M0518 Series BSP CMSIS V3.00.002
- IDE version
  - Keil uVersion 4.74

Hardware Environment

- Circuit components
  - NuTiny-M0518S-V1.1
- Diagram

M0518 uses PB.15 as TIMER0 external signal input pin. The sample code uses PB.14 to generate a pulse signal. Before executing the sample code, please connect PB.14 and PB.15 with jumper or wire as below:



4 Directory Information

- EC_M0518_TIMER_Capture_Both_Edge_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 Example Code

1. Browsing into sample code folder by Directory Information (section 4) and double click M0518_TIMER_Capture_Both_Edge.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
Sep. 17, 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.