

LAB 4 Watchdog timer and UART

Goal

- Understand the principle of watchdog timer
- Use watchdog timer in both watchdog mode and interval timer mode
- Use UART to communicate between different microprocessors

Task preparation

This lab contains three tasks. Prepare the code of Task 1 – Task 3 and submit your lab preparation via Lisam before the lab. Codes will be checked by lab assistants during the lab session.

Lab tasks:

Task 1

The following code use timer to generate square wave in Lab 2. In this code, the watchdog timer is disabled by the line:

```
WDTCTL = WDTPW + WDTHOLD;
```

Change the following code so that:

- Enable the watchdog timer
- Configure the watchdog as
 - Watchdog mode
 - WDT clock source from SMCLK
- Find a place to “kick the dog” all the time before the watchdog timer fires and reset the device.

```
#include <msp430g2553.h>
int a = 0;
void main(void)
{
    WDTCTL = WDTPW + WDTHOLD; // Stop WDT
    P1DIR |= 0x01; // P1.0 output
    TACCTL0 = CCIE; // TACCR0 interrupt enabled
    TACCR0 = 50000;
    TACTL = TASSEL_2 + MC_1; // SMCLK, up mode
    for(;;){_BIS_SR(GIE);}
}

// Timer A0 interrupt service routine
#pragma vector=TIMER0_A0_VECTOR
__interrupt void Timer_A (void)
{
    if (a == 10)
        {P1OUT ^= 0x01; a = 0;}
    else{a++;}
}
```

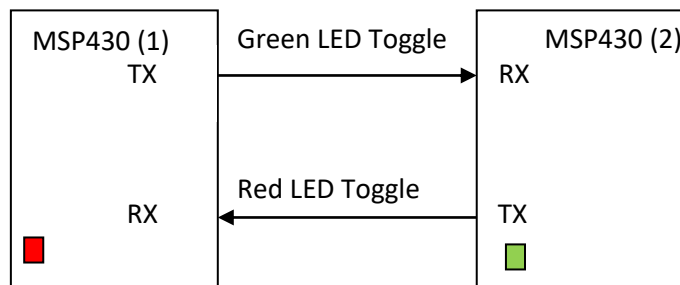
Task 2

Write a program to use watchdog timer in the interval timer mode. Follow the steps below

- Configure the WDT as timer mode
- Select the WDT clock source from SMCLK
- Enable the WDT interrupt
- Handle the WDT interrupt WDT_VECTOR in the interrupt service routine
- Toggle P0_1 (Red led) in the interrupt service routine
- Connect P0_1 with the oscilloscope, change the WDT interval to each of the four intervals and observe the output frequency from P0_1.

Task 3

Develop the code that 2 MSP430 boards can send commands to each other. Once a command is received, the MSP430 board toggles the green or red LED. As shown in the following figure, MSP430 (1) sends a command to MSP430 (2). When the command is received, MSP430 (2) toggles the green LED. Meanwhile, MSP430 (2) sends a command to MSP430 (1). When MSP430 (1) receives the command, it toggles the red LED. A timer is utilized to control the interval between 2 commands.



Important steps:

- Define the commands as a specific character, for example, “Green LED Toggle” defined as ‘g’ and “Red LED Toggle” defined as ‘r’.
- Configure the timer that defines the interval between two commands
- Configure the UART port
 - Utilize the calibrated DCO clock:
 - BCCTL1 = CALBC1_1MHZ;
 - DCOCTL = CALDCO_1MHZ;
 - Configure the TX and RX pin function
 - Configure the UART using SMCLK

- Configure the baud rate as 19200 (the SMCLK is configured as 1MHz)
 - Initialize USCI state machine
 - Enable the UART RX interrupt
- Send the command when TIMER interrupt triggered
 - MSP430 TX buffer name: UCA0TXBUF
 - MSP430(1) sends “Green LED Toggle” while MSP430(2) sends “Red LED Toggle”.
- Handle the received command in the UART RX interrupt service routine with the interrupt vector “USCIAB0RX_VECTOR”
 - Compare the command in the UART receive buffer UCA0RXBUF
 - Toggle the corresponding LED according to the received command
- In this task, one needs to develop two sets of code, one for MSP430 (1) and one for MSP430 (2). The basic logic and the function are same. The only difference is that MSP430 (1) sends out “Green LED Toggle” command and MSP430 (2) sends out “Red LED Toggle” command.