

## Assignment 2

- a. Scope captures were not saved will be recaptured and uploaded at a later date.
- b.
- c.
- d.

**set\_DCO.h**

```

/*
 * set_DCO.h
 *
 * Created on: Apr 9, 2019
 * Author: Daniel Gutmann
 */

#ifndef SET_DCO_H_
#define SET_DCO_H_

#define FREQ_1_5_MHZ 1
#define FREQ_3_MHZ 3
#define FREQ_6_MHZ 6
#define FREQ_12_MHZ 12
#define FREQ_24_MHZ 24
#define FREQ_48_MHZ 48

void set_DCO(freq)
{
    CS->KEY = CS_KEY_VAL; //Unlock registers
    CS->CTL0 = 0;

    //Set MCLK to DCOCLK input
    CS->CTL1 = CS_CTL1_SELM__DCOCLK;

    //SET DCO FREQUENCY
    if(freq == FREQ_1_5_MHZ)
    {
        CS->CTL0 = CS_CTL0_DCORSEL_0;
    }
    else if(freq == FREQ_3_MHZ)
    {
        CS->CTL0 = CS_CTL0_DCORSEL_1;
    }
    else if(freq == FREQ_6_MHZ)

```

```

{
    CS->CTL0 = CS_CTL0_DCORSEL_2;
}
else if(freq == FREQ_12_MHZ)
{
    CS->CTL0 = CS_CTL0_DCORSEL_3;
}
else if(freq == FREQ_24_MHZ)
{
    CS->CTL0 |= CS_CTL0_DCORSEL_4;
}
else if(freq == FREQ_48_MHZ)
{
    while ((PCM->CTL1 & PCM_CTL1_PMR_BUSY));
    PCM->CTL0 = PCM_CTL0_KEY_VAL | PCM_CTL0_AMR_1;
    while ((PCM->CTL1 & PCM_CTL1_PMR_BUSY));

    FLCTL->BANK0_RDCTL = (FLCTL->BANK0_RDCTL &
                        ~(FLCTL_BANK0_RDCTL_WAIT_MASK)) |
FLCTL_BANK0_RDCTL_WAIT_1;

    FLCTL->BANK1_RDCTL = (FLCTL->BANK0_RDCTL &
                        ~(FLCTL_BANK1_RDCTL_WAIT_MASK)) |
FLCTL_BANK1_RDCTL_WAIT_1;

    CS->CTL0 = CS_CTL0_DCORSEL_MASK & CS_CTL0_DCORSEL_5;

    CS->CTL1 = CS->CTL1 & ~(CS_CTL1_SELM_MASK | CS_CTL1_DIVM_MASK)
|
    CS_CTL1_SELM_3;
}
CS->KEY = 0; //lock registers
}

#endif /* SET_DCO_H_ */

```

## delay\_us.h

```

#include "msp.h"
/*
 * delay_us.h
 *

```

```

*   Created on: Apr 9, 2019
*       Author: Daniel Gutmann
*/

#ifndef DELAY_US_H_
#define DELAY_US_H_

void delay_us(usec)
{
    SysTick -> LOAD = (((CS->CTL0 &
CS_CTL0_DCORSEL_MASK) | CS_CTL0_DCORSEL_1) >> 11) * (usec) / 21;
    SysTick->CTRL |= 1;
    while((SysTick->CTRL & 1<<16) != 1<<16);
}

#endif /* DELAY_US_H_ */

```

## main.c

```

#include "msp.h"
#include "set_DCO.h"
#include "delay_us.h"

#define FREQ_1_5_MHZ 1
#define FREQ_3_MHZ 3
#define FREQ_6_MHZ 6
#define FREQ_12_MHZ 12
#define FREQ_24_MHZ 24
#define FREQ_48_MHZ 48

/**
 * main.c
 */

void main(void)
{
    WDT_A->CTL = WDT_A_CTL_PW | WDT_A_CTL_HOLD;    // stop watchdog
timer

    set_DCO(FREQ_48_MHZ);

    P4 -> SEL0 |= BIT3;

```

```
P4 -> SEL1 &= ~BIT3;
P4 -> DIR |= BIT3;
P4 -> OUT |= BIT3;

P2 -> SEL0 &= ~BIT7;
P2 -> SEL1 &= ~BIT7;
P2->DIR |= BIT7;
P2->OUT |=BIT7;

while (1)
{
    P2->OUT ^= BIT7;
    delay_us(40);        // Delay in microseconds
}
}
```