**Date Submitted:**

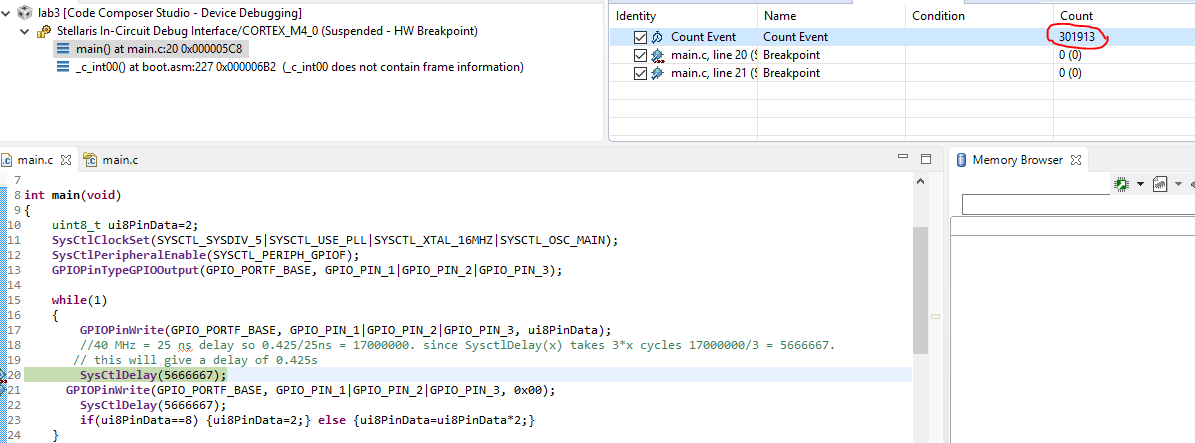
**Task 01:**

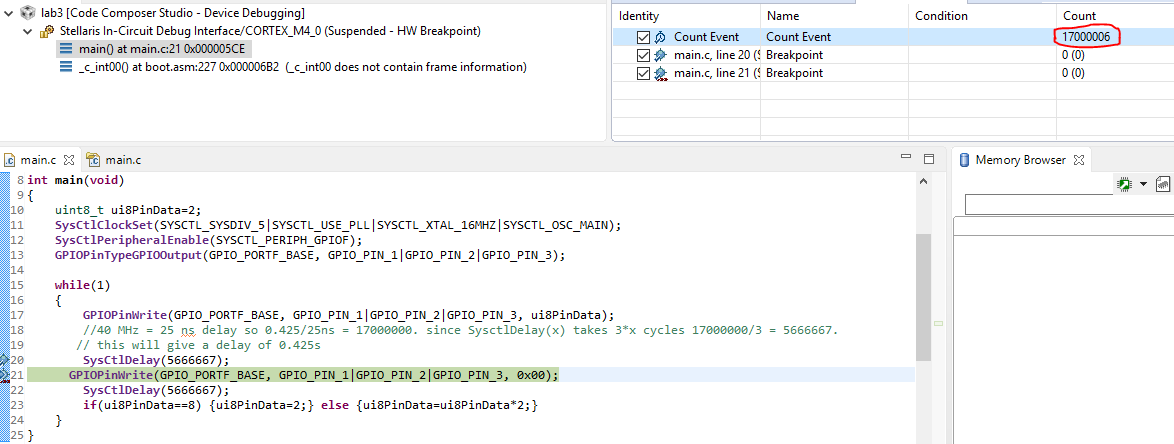
Youtube Link:

<https://youtu.be/wpKVRQnxOE8>

40 MHz = 25 ns delay. The number of clock cycles for a 0.425s delay is 0.425s/25ns = 17000000. since SysctlDelay(x) takes 3\*x cycles; 17000000/3 = 5666667.

in the first picture the clock cycle counter is reset to 0 (note: the value shown is not reset until the next clock cycle, thus the value shown is not the actual value which is 0)

****

****

The total count of 17000006 informs us that based on the 40MHz clock the delay is roughly 0.425s.

**Modified Code:**

**#include <stdint.h>**

**#include <stdbool.h>**

**#include "inc/hw\_memmap.h"**

**#include "inc/hw\_types.h"**

**#include "driverlib/sysctl.h"**

**#include "driverlib/gpio.h"**

**int main(void)**

**{**

**uint8\_t ui8PinData=2;**

**SysCtlClockSet(SYSCTL\_SYSDIV\_5|SYSCTL\_USE\_PLL|SYSCTL\_XTAL\_16MHZ|SYSCTL\_OSC\_MAIN);**

**SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPIOF);**

**GPIOPinTypeGPIOOutput(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3);**

**while(1)**

**{**

**GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, ui8PinData);**

**//40 MHz = 25 ns delay so 0.425/25ns = 17000000. since SysctlDelay(x) takes 3\*x cycles 17000000/3 = 5666667.**

**// this will give a delay of 0.425s**

**SysCtlDelay(5666667);**

**GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 0x00);**

**SysCtlDelay(5666667);**

**if(ui8PinData==8) {ui8PinData=2;} else {ui8PinData=ui8PinData\*2;}**

**}**

**}**

**------------------------------------------------------------------------------------**

**Task 02:**

Youtube Link:

<https://youtu.be/Ul0688uy7O8>

**Modified Code:**

**#include "inc/hw\_memmap.h"**

**#include "inc/hw\_types.h"**

**#include "driverlib/sysctl.h"**

**#include "driverlib/gpio.h"**

**int main(void)**

**{**

**uint8\_t ui8PinData=2;**

**SysCtlClockSet(SYSCTL\_SYSDIV\_5|SYSCTL\_USE\_PLL|SYSCTL\_XTAL\_16MHZ|SYSCTL\_OSC\_MAIN);**

**SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPIOF);**

**GPIOPinTypeGPIOOutput(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3);**

**while(1)**

**{**

**GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, ui8PinData);**

**//40 MHz = 25 ns delay so 0.425/25ns = 17000000. since SysctlDelay(x) takes 3\*x cycles 17000000/3 = 5666667.**

**// this will give a delay of 0.425s**

**SysCtlDelay(5666667);**

**GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 0x00);**

**SysCtlDelay(5666667);**

**//for the sequence R, G, B, RG, RB, GB, RGB, R, G, ...**

**// R = 2, B = 4, G = 8. thus a mapping from blink to the next is made based on the change in value**

**if(ui8PinData==2 | ui8PinData==4 | ui8PinData== 6) {ui8PinData=ui8PinData+6;} // R->G, B->RG, RB->GB: add 6 to current value**

**else if (ui8PinData==8 | ui8PinData==10) {ui8PinData=ui8PinData-4;} // G-> B, RG-> RB: subtract 4 from current value**

**else if (ui8PinData==12) {ui8PinData=ui8PinData+2;} // if GB add 2 to get to RGB**

**else {ui8PinData=2;} // if RGB or any other non specified value occurs, go to R**

**}**

**}**