**Date Submitted: 9/24/2019**

**Task 00: Execute provided code**

**Youtube Link:** <https://www.youtube.com/watch?v=AwON6qQNEpo>

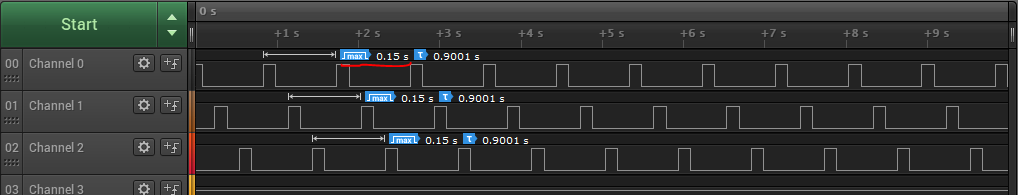
**------------------------------------------------------------------------------------**

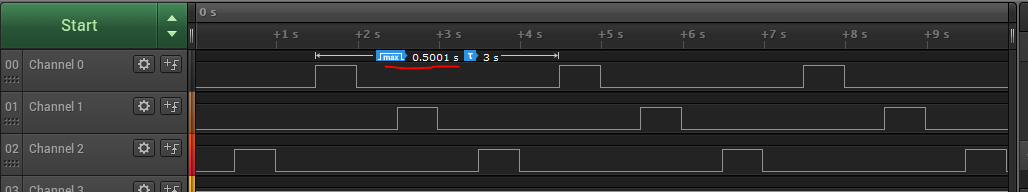
**Task 01:**

Youtube Link: <https://www.youtube.com/watch?v=FZyWAVO8Gsc>

**Modified Schematic (if applicable):**

**ON time for LEDs is around 0.15s**

**Current period (for all LEDs together) is 0.3s**

**To get 0.5s for delay in between LEDs (on and off time of each LED), delay (number) = delay (in seconds) / (1/sysclock) \*3 , and I put that value in the delay function (sysctldelay()). CLK frequency is 40MHz.**

**Modified Code:**

**// Insert code here**

**int** **main**(**void**)

{

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

//using a sysdiv of 5, using PLL, an external crystal of 16MHz, and using the main osc

**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);

**SysCtlDelay**(6666666); //this value will delay for 0.5s high (on)

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

**SysCtlDelay**(6666666); //this value will delay for 0.5s low (off)

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

}

}

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

**Task 02-a:**

Youtube Link: <https://www.youtube.com/watch?v=S0Z94dpUV5I>

**Modified Schematic (if applicable):**

**Modified Code:**

**// Insert code here**

**#include** <stdint.h>

**#include** <stdbool.h>

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

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

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

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

uint8\_t ui8PinData=4; //blue LED will start first

**int** **main**(**void**)

{

**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);

**SysCtlDelay**(2000000);

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

**SysCtlDelay**(2000000);

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

//sequence of colors -> B, G, R (repeat)

}

}

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

**Task 02-b:**

Youtube Link: <https://www.youtube.com/watch?v=uZZLFYt9ThE>

**Modified Schematic (if applicable):**

**Modified Code:**

**// Insert code here**

**#include** <stdint.h>

**#include** <stdbool.h>

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

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

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

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

uint8\_t ui8PinData=2; //start with red LED

**int** **main**(**void**)

{

**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);

**SysCtlDelay**(2000000);

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

**SysCtlDelay**(2000000);

**if** (ui8PinData==14) { //if LED = RGB

ui8PinData = 2; //resets back to R

}

**else** **if** (ui8PinData == 8) {

ui8PinData = 10; //LED = RG

}

**else** **if** (ui8PinData == 10) {

ui8PinData = 6; //LED = RB

}

**else** **if** (ui8PinData == 6) {

ui8PinData = 12; //LED = GB

}

**else** **if** (ui8PinData == 12) {

ui8PinData = 14; //LED = RGB

}

**else** {

ui8PinData \*= 2; //pattern sequence will reset to R, B, G (single pattern)

}

}

}

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