**Date Submitted: 9/24**

**Task 00: Execute provided 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"

uint8\_t ui8PinData = 2;

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

{

// Task 00

**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);

// T = 300 mS where LED stays on for 150 mS

**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; // Reset the LED sequence

**else**

ui8PinData = ui8PinData \* 2; // Increment the LED sequence

}

**return** 0;

}

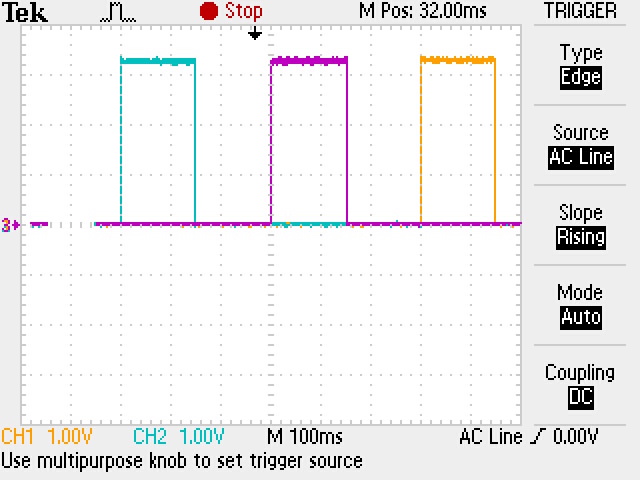
**Youtube Link:**

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

**Task 01:**

Youtube Link:

**Waveform of TASK 00:**

****

**The period of the provided code from TASK 00 is T = 300 mS. The LED turns on for 150 mS.**

**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"

uint8\_t ui8PinData = 2;

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

{

// Task 01

**SysCtlClockSet**(SYSCTL\_SYSDIV\_10 | 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);

// T = 1 s where LED stays on for 0.5 s

**while**(1)

{

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

**SysCtlDelay**(3333333);

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

**SysCtlDelay**(3333333);

**if**(ui8PinData == 8)

ui8PinData = 2; // Reset the LED sequence

**else**

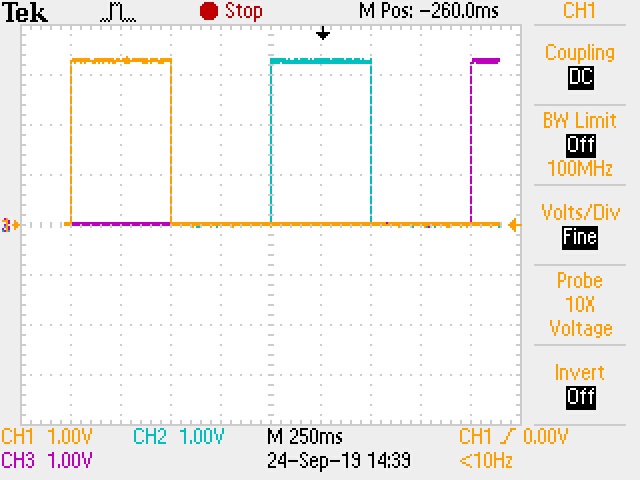
ui8PinData = ui8PinData \* 2; // Increment the LED sequence

}

**return** 0;

}

**Waveform of TASK 01:**



**Modifying the code from TASK 00 to show a wave with T = 1 s. The LED turns on for 500 mS.**

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

**Task 02:**

Youtube Link:

**Modified Code:**

1. **Sequence the LED to blink BGR**

**#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;

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

{

// Task 02

**SysCtlClockSet**(SYSCTL\_SYSDIV\_10 | 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);

// Color Values

// R = 2

// B = 4

// G = 8

// Task A - Blink the sequence BGR

**while**(1)

{

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

**SysCtlDelay**(3333333);

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

**SysCtlDelay**(3333333);

**if**(ui8PinData == 8)

ui8PinData = 2; // Reset the LED to RED

**else**

ui8PinData = ui8PinData \* 2; // Increment the LED sequence

}

**return** 0;

}

1. **Sequence the LED to blink R, G, B, RG, RB, GB, RGB,…**

**#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 = 1;

uint8\_t seqCount = 1;

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

{

**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);

**switch**(seqCount)

{

**case** 1:

**if**(ui8PinData == 0)

ui8PinData = 2; // Set to RED

**else**

{

ui8PinData = 0;

seqCount++;

}

**break**;

**case** 2:

**if**(ui8PinData == 0)

ui8PinData = 8; // Set to GREEN

**else**

{

ui8PinData = 0;

seqCount++;

}

**break**;

**case** 3:

**if**(ui8PinData == 0)

ui8PinData = 4; // Set to BLUE

**else**

{

ui8PinData = 0;

seqCount++;

}

**break**;

**case** 4:

**if**(ui8PinData == 0)

ui8PinData = 2; // Set to RED

**else** **if**(ui8PinData == 2)

ui8PinData = 8; // Set to GREEN

**else**

{

ui8PinData = 0;

seqCount++;

}

**break**;

**case** 5:

**if**(ui8PinData == 0)

ui8PinData = 2; // Set to RED

**else** **if**(ui8PinData == 2)

ui8PinData = 4; // Set to BLUE

**else**

{

ui8PinData = 0;

seqCount++;

}

**break**;

**case** 6:

**if**(ui8PinData == 0)

ui8PinData = 8; // Set to GREEN

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

ui8PinData = 4; // Set to BLUE

**else**

{

ui8PinData = 0;

seqCount++;

}

**break**;

**case** 7:

**if**(ui8PinData == 0)

ui8PinData = 2; // Set to RED

**else** **if**(ui8PinData == 2)

ui8PinData = 8; // Set to GREEN

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

ui8PinData = 4; // Set to BLUE

**else**

{

ui8PinData = 0;

seqCount = 1;

}

**break**;

**default**:

ui8PinData = 0; // Clear LED

seqCount = 1; // Reset sequence

}

}

**return** 0;

}