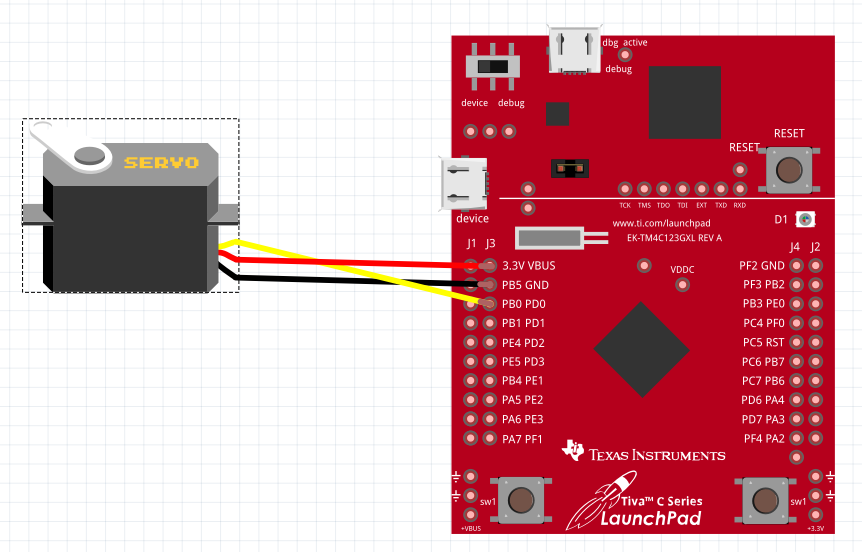
**Date Submitted: 10/6/19**

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

**Task 01:**

Youtube Link: <https://www.youtube.com/watch?v=g5QV3AMOpTw>

**Modified Schematic (if applicable):** 

**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"**

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

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

**#include "driverlib/pin\_map.h"**

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

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

**#define PWM\_FREQUENCY 55**

**int main(void)**

**{**

**volatile uint32\_t ui32Load;**

**volatile uint32\_t ui32PWMClock;**

**volatile uint32\_t ui8Adjust;**

**ui8Adjust = 83;**

**//Set clock to 40MHz**

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

**//Set PWM Clock to 625KHz**

**ROM\_SysCtlPWMClockSet(SYSCTL\_PWMDIV\_64);**

**//Enable PWM1, PortD and PortF**

**ROM\_SysCtlPeripheralEnable(SYSCTL\_PERIPH\_PWM1);**

**ROM\_SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPIOD);**

**ROM\_SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPIOF);**

**//Enable PD0 as output for PWM**

**ROM\_GPIOPinTypePWM(GPIO\_PORTD\_BASE, GPIO\_PIN\_0);**

**ROM\_GPIOPinConfigure(GPIO\_PD0\_M1PWM0);**

**HWREG(GPIO\_PORTF\_BASE + GPIO\_O\_LOCK) = GPIO\_LOCK\_KEY;**

**HWREG(GPIO\_PORTF\_BASE + GPIO\_O\_CR) |= 0x01;**

**HWREG(GPIO\_PORTF\_BASE + GPIO\_O\_LOCK) = 0;**

**ROM\_GPIODirModeSet(GPIO\_PORTF\_BASE, GPIO\_PIN\_4|GPIO\_PIN\_0, GPIO\_DIR\_MODE\_IN);**

**ROM\_GPIOPadConfigSet(GPIO\_PORTF\_BASE, GPIO\_PIN\_4|GPIO\_PIN\_0, GPIO\_STRENGTH\_2MA, GPIO\_PIN\_TYPE\_STD\_WPU);**

**//Set PWM Counter and load value**

**ui32PWMClock = SysCtlClockGet() / 64;**

**ui32Load = (ui32PWMClock / PWM\_FREQUENCY) - 1;**

**PWMGenConfigure(PWM1\_BASE, PWM\_GEN\_0, PWM\_GEN\_MODE\_DOWN);**

**PWMGenPeriodSet(PWM1\_BASE, PWM\_GEN\_0, ui32Load);**

**//Set the initial pulse width and enable PWM**

**ROM\_PWMPulseWidthSet(PWM1\_BASE, PWM\_OUT\_0, ui8Adjust \* ui32Load / 1000);**

**ROM\_PWMOutputState(PWM1\_BASE, PWM\_OUT\_0\_BIT, true);**

**ROM\_PWMGenEnable(PWM1\_BASE, PWM\_GEN\_0);**

**while(1)**

**{**

**//turn clockwise**

**while((ui8Adjust < 120) & (ui8Adjust > 48))**

**{**

**ui8Adjust++;**

**ROM\_PWMPulseWidthSet(PWM1\_BASE, PWM\_OUT\_0, ui8Adjust \* ui32Load / 1000);**

**ROM\_SysCtlDelay(200000);**

**}**

**ui8Adjust = 119;**

**//turn counterclockwise**

**while((ui8Adjust < 120) & (ui8Adjust > 48))**

**{**

**ui8Adjust--;**

**ROM\_PWMPulseWidthSet(PWM1\_BASE, PWM\_OUT\_0, ui8Adjust \* ui32Load / 1000);**

**ROM\_SysCtlDelay(200000);**

**}**

**ui8Adjust = 49;**

**}**

**}**

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

**Task 02:**

Youtube Link: <https://www.youtube.com/watch?v=DHB9-wKv-28>

**Modified Schematic (if applicable): N/A**

**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"**

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

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

**#include "driverlib/pin\_map.h"**

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

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

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

**//about 2ms at 40Mhz**

**#define time 28333**

**//PWM frequency in hz**

**uint32\_t freq = 100000;**

**int main()**

**{**

**//Set clock to 40Mhz**

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

**uint32\_t i;**

**uint32\_t Period;**

**uint32\_t dutyCycle;**

**Period = SysCtlClockGet()/freq ; //set period to 400**

**dutyCycle = Period-2;**

**//Configure PF1 as T0CCP1**

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

**SysCtlDelay(3);**

**GPIOPinConfigure(GPIO\_PF1\_T0CCP1);**

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

**//Configure timer 0 to split pair and timer B in PWM mode**

**//Set period and starting duty cycle.**

**SysCtlPeripheralEnable(SYSCTL\_PERIPH\_TIMER0);**

**SysCtlDelay(3);**

**TimerConfigure(TIMER0\_BASE, TIMER\_CFG\_SPLIT\_PAIR|TIMER\_CFG\_B\_PWM);**

**TimerLoadSet(TIMER0\_BASE, TIMER\_B, Period -1);**

**TimerMatchSet(TIMER0\_BASE, TIMER\_B, dutyCycle); // PWM**

**//Turn on timer0B**

**TimerEnable(TIMER0\_BASE, TIMER\_B);**

**//Start by rising Red LED**

**for(i=Period-2; i > 40 ;i--){**

**TimerMatchSet(TIMER0\_BASE, TIMER\_B, i);**

**SysCtlDelay(time);**

**}**

**while(1)**

**{**

**//Red brightness goes down**

**for(i=1; i < 360; i++){**

**TimerMatchSet(TIMER0\_BASE, TIMER\_B, i);**

**SysCtlDelay(time);**

**}**

**//Red brightness goes up**

**for(i=Period-2; i > 40; i--){**

**TimerMatchSet(TIMER0\_BASE, TIMER\_B, i);**

**SysCtlDelay(time);**

**}**

**}**

**}**

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

**Waiting for DC motor for tasks 3 and 4**

**Task 03:**

Youtube Link:

**Modified Schematic (if applicable):**

**Modified Code:**

**// Insert code here**

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

**Task 04:**

Youtube Link:

**Modified Schematic (if applicable):**

**Modified Code:**

**// Insert code here**

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