



INTEGRANTES:

No	NOMBRE	CARNÉ
1	Cindy Melissa Gatica Arriola	201709692
2	Stheeven Adonías, Coc Chán	201700519
3	Rodrigo Sosa Aquino	202012337
4		
5		

PRACTICA No. 3  
 PWM pin PB4

```
#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/pwm.h"
#include "driverlib/pin_map.h"
#include "inc/tm4c123gh6pm.h"

#define PWM_FREQUENCY 55
uint32_t Load;
uint32_t PWMClock;
uint8_t Adjust;

int main(void)
```

```

{
    Adjust = 84;
    SysCtlClockSet(SYSCTL_SYSDIV_2_5|SYSCTL_USE_PLL|SYSCTL_OSC_MAIN|SYSCTL_XTAL_16MHZ);

    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOB);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_PWM0);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);

    SysCtlPWMClockSet(SYSCTL_PWMDIV_64);

    // Configurar PB4 como salida de PWM (M0PWM2)
    GPIOPinTypePWM(GPIO_PORTB_BASE, GPIO_PIN_4);
    GPIOPinConfigure(GPIO_PB4_M0PWM2);

    // Desbloquear el puerto F y configurar los botones (SW1 y SW2)
    GPIO_PORTF_LOCK_R = GPIO_LOCK_KEY;
    GPIO_PORTF_CR_R = 0x1F;
    GPIOPinTypeGPIOInput(GPIO_PORTF_BASE, GPIO_PIN_4 | GPIO_PIN_0); // Configurar F0 y F4 como entrada
    GPIOPadConfigSet(GPIO_PORTF_BASE, GPIO_PIN_4 | GPIO_PIN_0, GPIO_STRENGTH_2MA,
    GPIO_PIN_TYPE_STD_WPU);

    PWMClock = SysCtlClockGet() / 64;
    Load = (PWMClock / PWM_FREQUENCY) - 1;

    PWMGenConfigure(PWM0_BASE, PWM_GEN_1, PWM_GEN_MODE_DOWN);
    PWMGenPeriodSet(PWM0_BASE, PWM_GEN_1, Load);

    PWMPulseWidthSet(PWM0_BASE, PWM_OUT_2, Adjust * Load / 1000);
    PWMOutputState(PWM0_BASE, PWM_OUT_2_BIT, true);
    PWMGenEnable(PWM0_BASE, PWM_GEN_1);

    while(1)
    {
        if(GPIOPinRead(GPIO_PORTF_BASE, GPIO_PIN_4) == 0x00)
        {
            Adjust--;
            if (Adjust < 28) // Evitar que baje demasiado
            {
                Adjust = 28;
            }
            PWMPulseWidthSet(PWM0_BASE, PWM_OUT_2, Adjust * Load / 1000);
        }
    }
}

```

```
}

    if(GPIOPinRead(GPIO_PORTF_BASE, GPIO_PIN_0) == 0x00)
    {
        Adjust++;
        if (Adjust > 140) {
            Adjust = 140;
        }
        PWMPulseWidthSet(PWM0_BASE, PWM_OUT_2, Adjust * Load / 1000);
    }

    SysCtlDelay(100000);
}
}
```

Link del funcionamiento

<https://drive.google.com/drive/folders/1LdESpYN1Vdraoc8ng8Ub1D8IPVsJX5hh?usp=sharing>