



INTEGRANTES:

No.	NOMBRE	CARNÉ
1	Rodrigo Sosa Aquino	202012337
2	Cindy Melissa Gatica Arriola	201909692
3	Stheeven Adonías Coc Chán	201700519
4		
5		

PRACTICA No. 1
CONTROL DE ENTRADAS Y SALIDAS

OBJETIVOS

- Que el estudiante se familiarice con el entorno Code Composer Studio.
- Implementar el conocimiento adquirido en una aplicación práctica.

CONOCIMIENTOS PREVIOS

- Configuración inicial y relojes.
- Habilitación de entradas y salidas
- Debug.

FORMA DE ENTREGA

- La práctica se entrega una semana después de su publicación.
- Subir los archivos a la plataforma en formato PDF.
- Agregar programa como carpeta comprimida, con el nombre Práctica_01_Micro.

DESCRIPCIÓN

La práctica consiste en habilitar todo el puerto B (B0-B7) como salida. Creando un efecto cascada, cada led debe estar encendido 1 segundo antes de saltar al siguiente. Esto deberá activarse mediante un pulsador externo, cualquier pin que no sea PF0 ni PF4, cuando el pulsador se suelte, deberá permanecer el último led activo encendido, al presionarlo nuevamente continuar a partir de donde se quedó.

En formato PDF subir el código utilizado. La semana siguiente a la publicación deberán presentar el programa.

Código

```
#include <stdint.h>
#include <stdbool.h>
#include "inc/tm4c123gh6pm.h"
#include "inc/hw_memmap.h"
#include "driverlib/sysctl.h"
#include "driverlib/gpio.h"

int pulsador = 0;
int ledA = GPIO_PIN_0;
int Prev = 0;
int contador = 0;

int main(void) {

    SysCtlClockSet(SYSCTL_OSC_MAIN | SYSCTL_USE_PLL | SYSCTL_XTAL_16MHZ | SYSCTL_SYSDIV_2_5);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF); //inicialización puerto F
    GPIOPinTypeGPIOInput(GPIO_PORTF_BASE,GPIO_PIN_1); // declarando pin 1 entrada
    GPIOPadConfigSet(GPIO_PORTF_BASE,GPIO_PIN_1,GPIO_STRENGTH_2MA,GPIO_PIN_TYPE_STD);

    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOB); //inicialización puerto B
    GPIOPinTypeGPIOOutput(GPIO_PORTB_BASE,0xFF);

    while(1){

        pulsador = GPIOPinRead(GPIO_PORTF_BASE,GPIO_PIN_1);

        if((pulsador== 2) && (Prev == GPIO_PIN_1)){
            GPIOPinWrite(GPIO_PORTB_BASE,0xFF , ledA );
            SysCtlDelay(80000000/3);
            ledA <=<1;

            if(ledA == GPIO_PIN_7){
                ledA = GPIO_PIN_0;
            }

        }

        Prev = pulsador;

    }
```

}