



INTEGRANTES:

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

PRACTICA No. 2  
ADC con otro pin

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

//variables
uint32_t muestra[1];

int main(void)
{
    // RELOJ GENERAL
    SysCtlClockSet(SYSCTL_SYSDIV_2_5|SYSCTL_USE_PLL| SYSCTL_OSC_MAIN| SYSCTL_XTAL_16MHZ);
    // PERIFERICO ADC
    SysCtlPeripheralEnable(SYSCTL_PERIPH_ADC0);
    // PIN ADC PA7
```

```
SysCtlPeripheralEnable( SYSCTL_PERIPH_GPIOE);
GPIOPinTypeADC(GPIO_PORTA_BASE, GPIO_PIN_7);

// Configura el disparo que habilitará el ADC y su Prioridad
// ADCSequenceConfigure(uint32_t ui32Base, uint32_t ui32SequenceNum, uint32_t ui32Trigger, uint32_t ui32Priority)
ADCSequenceConfigure(ADC0_BASE, 1, ADC_TRIGGER_PROCESSOR, 0);

//Secuencia de conversión para ADC0
// ADCSequenceStepConfigure(uint32_t ui32Base, uint32_t ui32SequenceNum, uint32_t ui32Step, uint32_t ui32Config)
ADCSequenceStepConfigure(ADC0_BASE, 1, 0, ADC_CTL_CH0);

//Habilita la toma de muestras
ADCSequenceEnable(ADC0_BASE, 1);
while(1)
{
    //Permite limpiar interrupciones para poder obtener una nueva.
    ADCIntClear(ADC0_BASE, 1);
    //Dispara la secuencia para poder llegar a obtener una muestra.
    ADCProcessorTrigger(ADC0_BASE, 1);
    // Es la muestra que se está obteniendo.
    ADCSequenceDataGet(ADC0_BASE, 1, muestra);

}
}
```