



MICROPROCESADORES

INFORME PRÁCTICA 8

JORGE CANDIA

21 de marzo de 2023

main1.c

```
#include <xc.h>
#include <stdint.h>
#include "Pic32Ini.h"

#define PIN_SERVO 15 //RB15 -> OC1
#define PIN_PULSADOR 5 //RB5 -> +20]

#define INC_20_GRADOS 277;

void initOC1(void){ //Conecto OC1 al servo -> genero mi se0al con OC1 y
se la entrego al servo

    //Configuro timer2
    T2CON = 0;
    TMR2 = 0;
    PR2 = 49999; //Inicio en -90] -> 1ms -> 2500
    //IFS0CLR = 1 << 9;
    T2CON = 0x8010;

    //Configuro OC1
    OC1CON = 0;
    OC1R = 2500; //1ms
    OC1RS = 2500; //Aumento este al pulsar
    OC1CON = 0x8006;

}

int main(void){

    //ANSELBCLR = (1<<PIN_SERVO || 1<<PIN_PULSADOR);
    ANSELC = ~0xF;

    TRISA = 0;
    TRISB = 1 << PIN_PULSADOR;
    TRISC = 0;

    LATA = 0;
    LATB = 0; //Servo empieza a 0
    LATC = 0xF; //LEDS apagados

    SYSKEY = 0xAA996655;
    SYSKEY = 0x556699AA;
    RPB15R = 5;
```

```

SYSKEY = 0x1CA11CA1;

initOC1();

int pulsador_ant = (PORTB >> PIN_PULSADOR) & 1;
int pulsador_act;
int t_alto = 2500;

while(1){

    pulsador_act = (PORTB >> PIN_PULSADOR) & 1;
    if((pulsador_act == 0) && (pulsador_ant == 1)){
        t_alto += INC_20_GRADOS;
        LATCINV = 2;

        if(t_alto > 5000)
            t_alto = 2500; //1ms -> -90° -> estado inicial

        OC1RS = t_alto;
    }
    pulsador_ant = pulsador_act;
}
}

```

main2.c

```
#include <xc.h>
#include <stdint.h>
#include "Pic32Ini.h"

#define PIN_SERVO 15 //RB15 -> OC1
#define PIN_PULSADOR 5 //RB5 -> +20]
#define PIN_PULSADOR_DECREMENTO 7 //RB7 -> -10]

#define INC_20_GRADOS 277;
#define DEC_10_GRADOS 138;

void initOC1(void){ //Conecto OC1 al servo -> genero mi se0al con OC1 y
se la entrego al servo

    //Configuro timer2
    T2CON = 0;
    TMR2 = 0;
    PR2 = 49999; //Inicio en -90] -> 1ms -> 2500
    //IFS0CLR = 1 << 9;
    T2CON = 0x8010;

    //Configuro OC1
    OC1CON = 0;
    OC1R = 2500; //1ms
    OC1RS = 2500; //Aumento este al pulsar
    OC1CON = 0x8006;

}

int main(void){

    //ANSELBCLR = (1<<PIN_SERVO || 1<<PIN_PULSADOR);
    ANSELC = ~0xF;

    TRISA = 0;
    TRISB = ((1 << PIN_PULSADOR)|(1 << PIN_PULSADOR_DECREMENTO));
    TRISC = 0;

    LATA = 0;
    LATB = 0; //Servo empieza a 0
    LATC = 0xF; //LEDS apagados

    SYSKEY = 0xAA996655;
```

```

SYSKEY = 0x556699AA;
RPB15R = 5;
SYSKEY = 0x1CA11CA1;

initOC1();

int pulsador_ant = (PORTB >> PIN_PULSADOR) & 1;
int pulsador_decremento_ant = (PORTB >> PIN_PULSADOR_DECREMENTO)&1;
int t_alto = 2500;
int pulsador_act, pulsador_decremento_act;

while(1){

    pulsador_act = (PORTB >> PIN_PULSADOR) & 1;
    pulsador_decremento_act = (PORTB >> PIN_PULSADOR_DECREMENTO)&1;

    //Flanco +20]
    if((pulsador_act == 0) && (pulsador_ant == 1)){
        t_alto += INC_20_GRADOS;

        if(t_alto > 5000)
            t_alto = 2500; //1ms -> -90] -> estado inicial

        OC1RS = t_alto;
    }

    //Flanco -10]

    if((pulsador_decremento_act == 0) && (pulsador_decremento_ant ==
1)){
        t_alto -= DEC_10_GRADOS;

        if(t_alto < 2500)
            t_alto = 5000; //1ms -> -90] -> estado inicial, no baja
m.s

        OC1RS = t_alto;
    }

    pulsador_decremento_ant = pulsador_decremento_act;
    pulsador_ant = pulsador_act;

}
}

```

main3.c

```
#include <xc.h>
#include <stdint.h>
#include "Pic32Ini.h"

#define INC_10_GRADOS 138; //Mejor 138 para que no haya overflow
prematuro
//(139*18 = 2502 -> llega al máximo antes de lo debido)

static int t_alto = 2500;
static int subiendo = 1;

void initOC1(void){ //Conecto OC1 al servo -> genero mi señal con OC1 y
se la entrego al servo

    //Configuro timer2
    T2CON = 0;
    TMR2 = 0;
    PR2 = 49999; //Inicio en -90º -> 1ms -> 2500
    //IFS0CLR = 1 << 9;
    T2CON = 0x8010;

    //Configuro OC1
    OC1CON = 0;
    OC1R = 2500; //1ms
    OC1RS = 2500; //Aumento este al pulsar
    OC1CON = 0x8006;
}

void initT3(void){
    T3CON = 0;
    TMR3 = 0;
    PR3 = 39063; //Cuento 500ms

    IFS0CLR = 1 << 14;
    IEC0SET = 1 << 14;
    IPC3SET = 10; //Prioridad 2 subprioridad 2

    T3CON = 0x8060;

    INTCON |= 1 << 12;
    asm("ei");
}
```

```

__attribute__((vector(12), interrupt(IPL2SOFT), nomips16)) void
InterruccionTimer3(void){
    //IFS0bits.T3IF = 0; //Bajo la flag para que no entre en bucle
    IFS0CLR = 1 << 14; //Bajo la flag

    if(subiendo == 1){
        t_alto += INC_10_GRADOS;

        if(t_alto > 5000){
            t_alto -= 5000 - INC_10_GRADOS;
            subiendo = 0;
        }
    }

    if(subiendo == 0){
        t_alto -= INC_10_GRADOS;

        if(t_alto < 2500){
            t_alto = 2500 + INC_10_GRADOS;
            subiendo = 1;
        }
    }

    OC1RS = t_alto;
}

int main(void){

    ANSEL = ~0xF;

    TRISA = 0;
    TRISB = 0;
    TRISC = 0;

    LATA = 0;
    LATB = 0; //Servo empieza a 0
    LATC = 0xF; //LEDS apagados

    SYSKEY = 0xAA996655;
    SYSKEY = 0x556699AA;
    RPB15R = 5;
    SYSKEY = 0x1CA11CA1;

    initOC1();
    initT3();
}

```

```
while(1){  
  
}  
  
}
```