**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY**

**Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente**

**Microcontrollers Laboratory**

**Practice 08**

**Names:**

**Marcos Rubén Flores Lara A01351360**

**Edgar Alejandro Rodríguez Gallegos A00822028**

**Date: June 2, 2022**

# Report 08 – Timers

Practice 08

ITESM Campus Mty

Marcos Rubén Flores Lara

A01351360

Edgar Alejandro Rodríguez Gallegos A00822028

Horario: 14:30-16:00

**horizontal line**

Instructor: [Matias Vázquez Piñón](https://experiencia21.tec.mx/courses/261838/users/136680)

Due Date: 06/02/2022

**PROCEDURE:**

First, we used the known configuration examples that were given to use during the elaboration of the practice 7, from there we could reuse most of the last code.

For the elaboration of the code we got supported from different colleagues that as well as us are taking the laboratory since we worked better among others.

Finally the code ended up as follows:

#include "device\_config.h"

#include <stdint.h>

#include <math.h>

#include <string.h>

#include <stdio.h>

#include <stdlib.h>

#include <xc.h>

#define \_XTAL\_FREQ 1000000

#define SWEEP\_STEP 1

#define LCD\_DATA\_R PORTD

#define LCD\_DATA\_W LATD

#define LCD\_DATA\_DIR TRISD

#define LCD\_RS LATCbits.LATC2

#define LCD\_RS\_DIR TRISCbits.TRISC2

#define LCD\_RW LATCbits.LATC1

#define LCD\_RW\_DIR TRISCbits.TRISC1

#define LCD\_E LATCbits.LATC0

#define LCD\_E\_DIR TRISCbits.TRISC0

char frequency[3];

void send2LCD(char);

void LCD\_init(void);

void LCD\_cmd(char);

void portsInit(void);

void delay\_counter(void);

void IntegertoString(char\*,int);

void main(void) {

portsInit();

LCD\_RS = 0;

LCD\_RW = 0;

LCD\_E = 0;

LCD\_init();

send2LCD('F');

send2LCD('r');

send2LCD('e');

send2LCD('q');

send2LCD('u');

send2LCD('e');

send2LCD('n');

send2LCD('c');

send2LCD('y');

send2LCD(':');

delay\_counter(); //initializes the timer and counter, and returns the value of the counter

LCD\_cmd(0xC4); //Position the cursor in the fifth space of the second row

send2LCD(' ');

send2LCD('H');

send2LCD('z');

while(1){

delay\_counter();

}

}

void LCD\_init(void){

LATC = 0; // Make sure LCD control port is low

LCD\_E\_DIR = 0; // Set Enable as output

LCD\_RS\_DIR = 0; // Set RS as output

LCD\_RW\_DIR = 0; // Set R/W as output

LCD\_cmd(0x38); // Display to 2x16

\_\_delay\_ms(250);

LCD\_cmd(0x0F); // Display on, cursor on and blinking

\_\_delay\_ms(250);

LCD\_cmd(0x01); // Clear display and move cursor home

\_\_delay\_ms(250);

}

void send2LCD(char xy){

LCD\_RS = 1;

LCD\_RW = 0;

LCD\_E = 1;

LCD\_DATA\_W = xy;

Nop();

Nop();

LCD\_E = 0;

\_\_delay\_ms(250);

}

void portsInit(void){

OSCCON = 0x34;// Set the internal oscillator to 8MHz and stable

ANSELC = 0; // Set port D as Digital for 7 segment anodes

TRISC = 0x00; // for Port D, set all pins as outputs for 7 segment anodes

ANSELD = 0; // Set port D as Digital for 7 segment anodes

TRISD = 0x00; // for Port D, set all pins as outputs for 7 segment anodes

TRISAbits.TRISA4 = 1;

LCD\_DATA\_DIR = 0x00;

void LCD\_cmd(char cx) {

//LCD\_rdy(); // wait until LCD is ready

LCD\_RS = 0; // select IR register

LCD\_RW = 0; // set WRITE mode

LCD\_E = 1; // set to clock data

Nop();

LCD\_DATA\_W = cx; // send out command

Nop(); // No operation (small delay to lengthen E pulse)

LCD\_E = 0; // complete external write cycle

}

}

void delay\_counter(void){

TMR1H = 0x0B; //High part 0x0BDC

TMR1L = 0xDC; //Low part 0x0BDC

T1GCONbits.TMR1GE = 0; //Timer starts via firmware

PIR1bits.TMR1IF = 0; //Clear the TMR1 overflow flag

T1CON = 0b00100001; //Configure and start Timer1

while(PIR1bits.TMR1IF == 0){ //Wait for overflow

T0CON = 0b11111000; //Configure and start Timer0

}

IntegertoString(frequency, TMR0L); //Converts value of counter to a character array

LCD\_cmd(0xC0); //Sends cursor to first position in second row

for(int i=0; i<sizeof(frequency); i++){

if(frequency[i]=='0' | frequency[i]=='1' | frequency[i]=='2' | frequency[i]=='3' | frequency[i]=='4' | frequency[i]=='5' | frequency[i]=='6' | frequency[i]=='7' | frequency[i]=='8' | frequency[i]=='9'){

send2LCD(frequency[i]); //if there is a number (char) in the array it sends it to the LCD

}

else{

send2LCD(' '); //if there isn't a number it sends a blank space

}

}

memset(frequency,' ',sizeof(frequency)); //resets the array

T0CON = 0x00; //Stop Timer0

T1CON = 0x00; //Stop Timer1

PIR1bits.TMR1IF = 0; //Clear the TMR1 overflow flag

TMR0L = 0x00; //Reset counter

}

void IntegertoString(char \* string, int number) { //Integer into string

if(number == 0) { string[0] = '0'; return; };

int divide = 0;

int modResult;

int length = 0;

int isNegative = 0;

int copyOfNumber;

int offset = 0;

copyOfNumber = number;

if( number < 0 ) {

isNegative = 1;

number = 0 - number;

length++;

}

while(copyOfNumber != 0)

{

length++;

copyOfNumber /= 10;

}

for(divide = 0; divide < length; divide++) {

modResult = number % 10;

number = number / 10;

string[length - (divide + 1)] = modResult + '0';

}

if(isNegative) {

string[0] = '-';

}

string[length] = '\0';

}

**RESULTS:**

Once the code was finished, then we proceeded to simulate the laboratory into Proteus since one of our Microcontrollers was out of use, afterwards our schematic ended up as follows:

Diagrama, Esquemático

Descripción generada automáticamente

Ilustración : Proteus schematic view

Imagen de la pantalla de un celular

Descripción generada automáticamente con confianza media

Ilustración : Wave generator

Tabla

Descripción generada automáticamente

Ilustración 3: Frequency recieved from the waveform

GitHub Links:

https://github.com/A01351360/Microcontroladores/tree/main/Laboratorio%20de%20microcontroladores

**CONCLUSIONS:**

Marcos: This project in particular I believed represented a higher degree of complexity since we were not capacitated to use the software Proteus, we´ve never used it before in order to simulate all of the interactions we needed for this practice and that’s what I believe is the most important part of the laboratory, to figure out how to get something working even when you don´t have any idea on how it works.

**Alejandro:**

I believe that the collaborative work is very important now a days since most of thew students find their best ideas while bouncing back and forth their knowledge among people that are involved in the same education challenges.