



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA

Laboratorio de Microcomputadoras

Práctica 9. Programación en C
Comunicación serie síncrona, I2C

Grupo de Laboratorio: 4

Sanjuan Aldape Diana Paola
Grupo de Teoría: 5

Esparza Fuentes Jorge Luis
Grupo de Teoría: 4

Fecha de realización: 3 de mayo del 2022



DESARROLLO

1.- El objetivo del siguiente programa será para mayor comprensión de la comunicación I2C y la programación en C, por lo que se pide analizarlo y comentarlo para su reporte; observar en el circuito la conexión de A2, A1 y A0 para generar la dirección del esclavo, así como su uso en el programa.

```
#include <16F877.h>
#fuses HS,NOWDT,NOPROTECT
#use delay(clock=20000000)
//Configura modo Maestro, asigna C4 para SDA y C3 para SCL, así como velocidad baja.
#use i2c(MASTER, SDA=PIN_C4, SCL=PIN_C3,SLOW, NOFORCE_SW)
int contador=0;
void escribir_i2c(){
    i2c_start(); //Inicia la comunicacion
    i2c_write(0x42); //Se envía la dirección del esclavo y se inidica escritura
    i2c_write(contador); //Se manda el valor del contador
    i2c_stop(); //Finaliza la comunicación
}
void main()
{
    while(true)
    {
        escribir_i2c(); //Se llama a la funcion escribir_i2c
        delay_ms(500); //Retardo de 0.5 segundos
        contador++; //Incrementa el contador
    }
}
```

2.- Realizar la modificación al programa para que también muestre el contador en el puerto B.

```
#include <16F877.h>
#fuses HS,NOWDT,NOPROTECT
#use delay(clock=20000000)
//Configura modo Maestro, asigna C4 para SDA y C3 para SCL, así como velocidad baja.
#use i2c(MASTER, SDA=PIN_C4, SCL=PIN_C3,SLOW, NOFORCE_SW)
int contador=0;
void escribir_i2c(){
    i2c_start(); //Inicia la comunicacion
    i2c_write(0x42); //Se envía la dirección del esclavo y se inidica escritura
    i2c_write(contador); //Se manda el valor del contador
    i2c_stop(); //Finaliza la comunicación
}
void main()
{
    while(true)
    {
        escribir_i2c(); //Se llama a la funcion escribir_i2c
        output_b(contador); //Se pone el valor del contador en el puerto b
        delay_ms(500); //Retardo de 0.5 segundos
        contador++; //Se incrementa el contador
    }
}
```



3.- Realizar las modificaciones necesarias para que además de lo resuelto en el ejercicio previo, muestre el contador en un display LCD que funcionará como esclavo I2C.

Consideraciones:

- A. Debe incluir la librería `i2c_LCD.c` a su programa; esta librería contiene el protocolo de comunicación I2C para uso del Display de Cristal Líquido LCD.
- B. La biblioteca `I2C_LCD` permite emplear con los mismos nombres las funciones empleadas para el LCD en formato paralelo.
- C. Es necesario incluir la función que configura la forma de comunicación I2C del LCD. a. `lcd_init(DIRECCION_ESCLAVO,COLUMNAS,RENGLONES);`
 - a. `DIRECCION_ESCLAVO`: es la dirección configurada por los valores fijos de fábrica y los definidos por hardware del módulo PCF8574 que controla al LCD; ubicar en el esquemático la configuración, para obtener la dirección.
 - b. `COLUMNAS`: es la cantidad de columnas de LCD
 - c. `RENGLONES`: es la cantidad de filas disponibles en el LCD
 - d. En la practica se usará LCD de 16x2

```
#include <16F877.h>
#fuses HS,NOWDT,NOPROTECT
#use delay(clock=2000000)
//Configura modo Maestro, asigna C4 para SDA y C3 para SCL, así como velocidad baja.
#use i2c(MASTER, SDA=PIN_C4, SCL=PIN_C3,SLOW, NOFORCE_SW)
#include <i2c_LCD.c>
int contador=0;
void escribir_i2c(){
    i2c_start(); //Inicia la comunicacion
    i2c_write(0x42); //Se envía la dirección del esclavo y se inidica escritura
    i2c_write(contador); //Se manda el valor del contador
    i2c_stop(); //Finaliza la comunicación

//Función para mandar el valor del contador en Decimal y Hexadecimal al LCD
void escribir_lcd(){
    lcd_gotoxy(5,1); //Seleccionamos la columna y las filas a usar
    printf(lcd_putc," DEC: %d\n ",contador); //Se imprime el valor del contador en Decimal
    lcd_gotoxy(5,2); //Seleccionamos la columna y las filas a usar
    printf(lcd_putc," HEX: %x\n ",contador); //Se imprime el valor del contador en Hexadecimal
}
void main()
{
    lcd_init(0x4E,16,2);
    while(true)
    {
        escribir_i2c(); //Se llama a la función escribir_i2c
        output_d(contador); //Se pone el valor del contador en el puerto b
        escribir_lcd(); //Se llama a la función escribir_lcd
        delay_ms(500); //Retardo de 0.5 segundos
        contador++; //Se incrementa el contador
    }
}
```



4.- Realizar un programa de tal forma que obtenga la lectura de la entrada generada por otro dispositivo esclavo y los muestre en los tres periféricos usados en la actividad 3.

```
#include <16F877.h>
#fuses HS,NOWDT,NOPROTECT
#use delay(clock=20000000)
#use i2c(MASTER, SDA=PIN_C4, SCL=PIN_C3,SLOW, NOFORCE_SW)
#include <i2c_LCD.c>
int num=0;
void escribir_i2c(){
    i2c_start();           //Inicia la comunicacion
    i2c_write(0x42);       //Se manda la direccion e indica que será escritura
    i2c_write(num);        //Se manda la información
    i2c_stop();            //Se detiene la comunicacion
}
void leer(){
    i2c_start();           //Inicia la comunicacion
    i2c_write(0x45);       //Se manda la direccion e indica que será lectura
    num=i2c_read();
    num=i2c_read(0);
    i2c_stop();            //Se detiene la comunicacion
}
void main()
{
    lcd_init(0x4E,16,2);
    while(true)
    {
        leer();
        escribir_i2c();
        output_d(num);
        lcd_gotoxy(1,1);
        printf(lcd_putc,"    DEC: %d",num);
        lcd_gotoxy(1,2);
        printf(lcd_putc,"    HEX: %x",num);
        delay_ms(500);
    }
}
```

CONCLUSIÓN

- Sanjuan Aldape Diana Paola

En esta práctica conocí el protocolo I2C de comunicación serial, que brinda la posibilidad de transferir datos de un microcontrolador a diversos periféricos. Con este protocolo y haciendo uso del IDE, PIC C Compiler, se logró mandar información a los 3 periféricos conectados, dos displays de 7 segmentos y un display LCD.

- Esparza Fuentes Jorge Luis

Se experimentó y se reforzó los conocimientos sobre la comunicación serie pero ahora implementando el protocolo de comunicación I2C el cual nos permite entablar comunicación de



forma síncrona, utilizando el circuito PCF8574 como expensor de puertos, de tal forma que se conectó como esclavo y a partir de esta implementación se logró entablar comunicación con varios dispositivos a la vez y de esta forma se pudo lograr obtener los resultados esperados con base en los objetivos planteados y los ejercicios propuestos en esta práctica.