



Universidad Veracruzana

Universidad Veracruzana



FACULTAD
DE
INSTRUMENTACIÓN
ELECTRÓNICA

Experiencia Educativa:

Tópicos Avanzados de Instrumentación Electrónica II
Sistemas embebidos

Actividad n. 7

Menú en aplicación para PC

Profesor:

Equipo:

Fernández Salinas Violeta Margarita
Velázquez Castañeda Mauricio
Salazar Aguilar Luis Gerardo

28 de octubre del 2020

Objetivos:

Desarrollar una aplicación con menú en un lenguaje visual para desplegar información en la PC que represente un menú básico con el cual interactúe el usuario. Debe existir la comunicación entre pc y microcontrolador, contener botones, cajas de texto, listas de selección o algún medio gráfico con el que el usuario interactúe en él.

Marco teórico:

El puerto serie es en la actualidad una de las principales formas de comunicación entre dispositivos externos y ordenadores en la actualidad. Una interfaz que podemos encontrar en absolutamente todos los periféricos que tenemos en nuestro escritorio, así como en el interior de nuestro equipo.

En tecnologías básicas, un puerto serie es una interfaz física de comunicación en serie a través de la cual se transfiere información mandando o recibiendo un bit. A lo largo de la mayor parte de la historia de las computadoras, la transferencia de datos a través de los puertos de serie ha sido generalizada. Se ha usado y sigue usándose para conectar las computadoras a dispositivos como terminales o módems. Los ratones, teclados, y otros periféricos también se conectaban de esta forma.

Metodología

Para el diseño de este menú se utilizó el lenguaje C# por medio del programa de visual estudio.

De esta manera se puede realizar un código en el microcontrolador por medio de arduino, en el cual, se encarga de mandar y recibir datos por el puerto serial para realizar diferentes tareas, entre ellas, tenemos que es el encendido y apagado de leds, uso de una lcd, así como también enviar datos de un sensor de temperatura para mostrarlo en la aplicación en tiempo real.

Desarrollo para programa en Arduino.

Como bien se mencionó anteriormente, el desarrollo del programa en el microcontrolador ATM328P se realizó en el id de arduino. Y consta de varias partes donde envía y recibe datos para realizar diferentes tareas.

```
Interfaz_grafica $
#include <Mouse.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,16,2); // inicializa la interfaz I2C del LCD 16x2

int potenciometro=14;           //Asignamos el numero 14 para declarar el pin
int voltage=0;                  //Guardaremos el valor de registro
String Voltage2;                //Cadena de Caracteres
int leds[4] = {10, 11, 12, 13};
volatile int velocidad = 0;

void setup() {

    pinMode(potenciometro, OUTPUT);
    pinMode(10,OUTPUT);
    pinMode(11,OUTPUT);
    pinMode(12,OUTPUT);
    pinMode(13,OUTPUT);

    Serial.begin(9600);          //Comunicacion

    lcd.init();                  // Se inicializa el LCD 16x2
    lcd.backlight();
    lcd.clear();
}
```

En la parte inicial contamos con las declaraciones de las variables que vamos a tener en el programa, estas las hacemos para nombrar los pines a los que vamos a usar para las diferentes tareas

```
Interfaz_grafica $

int potenciometro=14;           //Asignamos el numero 14 para declarar el pin
int voltage=0;                  //Guardaremos el valor de registro
String Voltage2;                //Cadena de Caracteres
int leds[4] = {10, 11, 12, 13};
volatile int velocidad = 0;

void setup() {

    pinMode(potenciometro, OUTPUT);
    pinMode(10,OUTPUT);
    pinMode(11,OUTPUT);
    pinMode(12,OUTPUT);
    pinMode(13,OUTPUT);

    Serial.begin(9600);          //Comunicacion

    lcd.init();                  // Se inicializa el LCD 16x2
    lcd.backlight();
    lcd.clear();
}
```

Seguido del Void setup().

Encontramos que vamos a dar estas decisiones para entradas y salidas de los pines, para que los usaremos e inicializamos la lcd.

```
Interfaz_grafica $

void loop() {

    //if(dato=='I'){

    // while(int a=true){
        voltage=analogRead(potenciometro);    //Cargamos el voltage el valor
        Voltage2=String(voltage);            //Convertimos el entero en una cadena
        Serial.println(Voltage2);            //Lo mandamos al puerto serial
        delay(100);

    ///////////////////////////////////////////////////LCD//////////////////////////////////////

    if(Serial.available()){                  //Comprueba si hay caracter en el puerto serial
        int dato=Serial.read();              //Guardara el caracter leído

    /////////////////////////////////////////////////// ENDENDIDOS DE LEDS//////////////////////////////////////

    //if(Serial.available()){                //Comprueba si hay caracter en el puerto serial
```

Para la primer parte del void loop()

Tendremos que leer el voltaje que se encuentra en el pin 14 donde leerá el voltaje de ese pin y lo convertirá en un valor de 0 a 1024 por la resolución de los 8 bits que tenemos. Este número lo convertirá en una cadena tipo String y lo mandará por el puerto serial.

```
Interfaz_grafica Arduino 1.8.13 (Windows Store 1.8.42.0)
Archivo Editar Programa Herramientas Ayuda

Interfaz_grafica $

if(Serial.available()){                  //Comprueba si hay caracter en el puerto serial
int dato=Serial.read();                  //Guardara el caracter leído

    /////////////////////////////////////////////////// ENDENDIDOS DE LEDS//////////////////////////////////////

    //if(Serial.available()){                //Comprueba si hay caracter en el puerto serial

    //int dato=Serial.read();                //Guardara el caracter leído

    if(dato=='A'){                        //Condicion para verificar si el dato es A
        digitalWrite(10,HIGH);            //Encender led
        delay(10);

    }

    if(dato=='B'){                        //Condicion para verificar si el dato es B
        digitalWrite(10,LOW);              //Apagar el led
        delay(10);

    }

    if(dato=='C'){                        //Condicion para verificar si el dato es C
```

Para las siguientes dos condiciones lo que hará es verificar si hay un dato por el puerto serial y leer ese dato para realizar las diferentes tareas. Ahí entran los encendidos y apagados de leds que dependiendo el carácter que lea el puerto serial realizara el prendido o apagado del tipo led que se

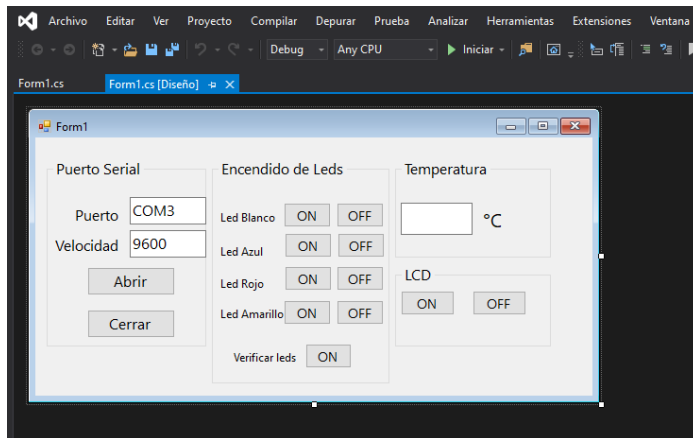
le indique, así como también un carácter en especial para la lcd, y oro para encender los leds de manera continua y verificar que estén funcionando todo.

Básicamente es la la tarea del microcontrolador, que para que reciba los datos y los envíe, necesitaremos de una aplicación de interfaz gráfica que los podamos visualizar y podamos mandar y/o controlar.

Aplicación de menú para pc

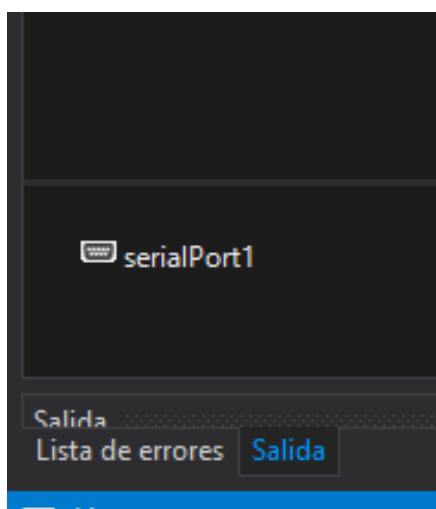
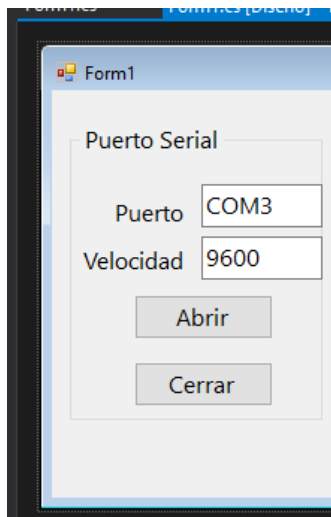
Para esta aplicación usamos el lenguaje C# en visual Studio.

Principalmente optamos por, de manera gráfica, darle el diseño a nuestra interfaz.

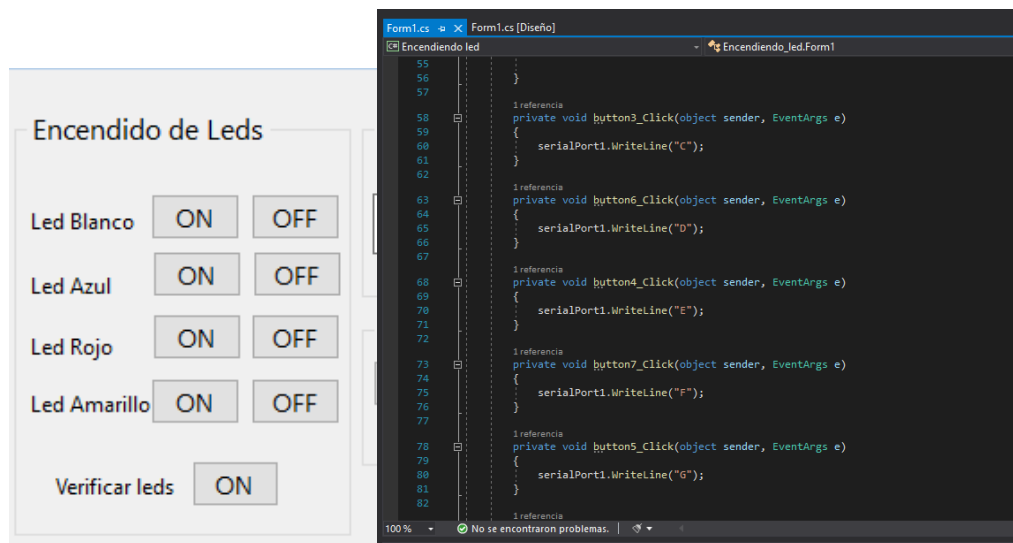


En la cual, dividíamos los apartados para tener un mejor diseño y mejor clasificación.

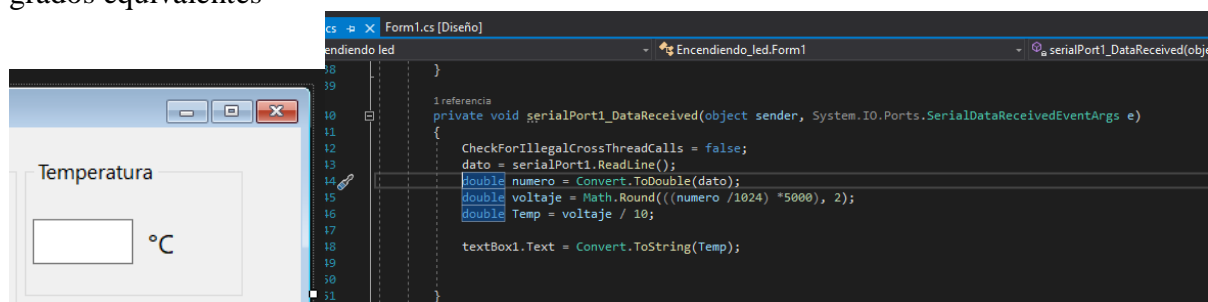
En primera instancia tenemos el puerto serial, que para controlarlo y tenemos dentro de nuestro código tenemos que agregar el Serial Port. Se agregaron varios text box y text button para que, por medio de un código, el usuario pueda cambiar estos puertos de entrada y de salida y la velocidad a su gusto



Se hizo lo mismo con los diferentes apartados, donde para el encendido y apagado de leds por medio de los button que estaban a su lado enviaran un carácter que el cual previamente se había puesto, para que realice la acción deseada.



Para la parte de la visualización de la temperatura, tenemos que se estarán mostrando los datos en un text box, que ya, con su propio código en el cual hace la conversión de los datos recibidos que son del 0 al 1024, los convierta a voltajes y de ahí los pueda convertir de nuevo pero ahora a los grados equivalentes



Resultados:

Los resultados de esta actividad fueron como los esperábamos, aunque al principio costaba un poco de trabajo llegar a entenderle bien o saber que hacer exactamente para que funcionara, gracias algunas páginas de internet y libros aprendimos hacerlo.

