



Universidad Veracruzana

UNIVERSIDAD VERACRUZANA

FACULTAD DE INSTRUMENTACIÓN
ELECTRÓNICA



FACULTAD
DE
INSTRUMENTACIÓN
ELECTRÓNICA

1-11-2020

TÓPICOS AVANZADOS DE INSTRUMENTACIÓN ELECTRÓNICA I (SISTEMAS EMBEBIDOS)

Tarea 6. – Aplicación de menú genérico

Alumnos:

Rangel Pulido Julio

Martínez González Sergio David

Ramírez Mendoza Jesús Emiliano

Catedrático:

ÍNDICE

1.	ACTIVIDAD	3
2.	DESCRIPCIÓN	3
3.	LISTA DE COMPONENTES	9
4.	DIAGRAMA.....	11
5.	CONCLUSIONES	11
6.	BIBLIOGRAFÍA	12

1. ACTIVIDAD

La actividad consiste en desarrollar una aplicación de esquema de menú genérico con la incorporación de un LCD y un control mediante teclado/encoder/teclas.

2. DESCRIPCIÓN

Aplicación de menú genérico:

La aplicación de menú se desarrollará en Arduino utilizando como herramienta principal un sistema de matriz para mostrar los textos en el LCD 20x4.

INICIO
ENCENDER LED
ENCENDER LED2
INTEGRANTES

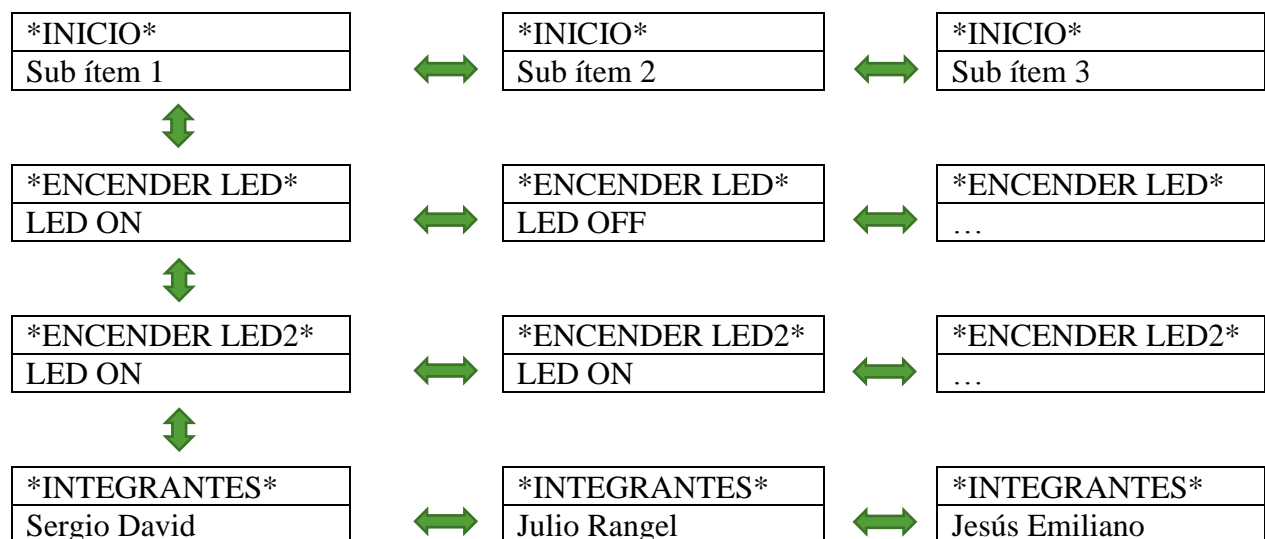
Matriz 4x1

Sub ítem 1	Sub ítem 2	Sub ítem 3
LED ON	LED OFF	...
LED ON	LED OFF	...
Sergio David	Julio Rangel	Jesús Emiliano

Matriz 4x3

Para generar el menú principal se utilizará una matriz 4x1 donde se mostrarán los títulos de cada sección y se utilizará una matriz 4x3 para mostrar los subtítulos de cada sección.

La forma de navegación entre títulos y subtítulos será de la siguiente forma:



Como se puede observar en el menú se podrán realizar acciones específicas como encender un LED conectado a unos pines de salida digital del Arduino.

Además, se tendrá un sistema de detección de la inactividad en el menú, es decir, cuando el usuario no realice ninguna acción sobre el menú, este cambiará a un salvapantallas que mostrará el estado actual de los leds, para saber si están prendidos o apagados.

Líneas del código.

```

1 //Equipo 6_Sistemas Embebidos
2 //Declaración de las librerías a utilizar
3 #include <Wire.h>
4 #include <LiquidCrystal_I2C.h>
5
6 //Se configura la forma de conexión de la LCD
7 LiquidCrystal_I2C lcd(0x27,20,4);
8
9 //Se declaran contantes que indican las dimensiones del menú
10 const byte NumMenuP = 4;
11 const byte NumMaxSubM = 3;
12 const byte NumBtnn = 5;
13
14 //Arreglo que indica los valores asignados a cada botón
15 byte Btnn[NumBtnn] = {
16     2, //UP
17     3, //RIGHT
18     4, //DOWN
19     5, //INT
20     6 //LEFT
21 };
22
23 //Arreglo que designa los títulos de cada elemento del menú
24 String MenuP[NumMenuP] = {
25     "INICIO",
26     "ENCENDER LED",
27     "ENCENDER LED2",
28     "INTEGRANTES",
29 };
30
31 //Arreglo que designa los títulos de cada sub-elemento del menú
32 String SubMenus[NumMenuP][NumMaxSubM] = {
33     {"Sub item 1", "Sub item 2", "Sub item 3"},
34     {"LED ON", "LED OFF", "..."},
35     {"LED ON", "LED OFF", "..."},
36     {"Sergio David", "Julio Rangel", "Jesus Emiliano"},
37 };
38
39 //Variables utilizadas para el funcionamiento del menú
40 int tecla = -1;
41 int teclaAnt = -1;
42 boolean cursorAct = false;
43 unsigned long tiempo;
44 int x = -1;
45 int y = 0;
46 int LED = 8;
47 int LED1 = 9;
48 int cuenta = 0;
49 int cuenta2 = 0;
50 int estado1 = 0;
51 int estado2 = 0;
52
53 void setup() {
54     //Velocidad de comunicación con el puerto serial
55     Serial.begin(9600);
56     //Configura como entrada los pines físico de cada botón
57     for (int i = 0; i < NumBtnn; i++) {
58         pinMode(Btnn[i], INPUT);
59     }

```

```

60 //Configuraciones iniciales del LCD y despliegue de mensaje de inicio
61 lcd.init();
62 lcd.backlight();
63 lcd.clear();
64 lcd.setCursor(0,0);
65 lcd.print("Menu LCD 20x4");
66 lcd.setCursor(3,1);
67 lcd.print("Arduino");
68 delay(1000);
69 lcd.clear();
70 lcd.setCursor(0,0);
71 lcd.print(" MENU PRINCIPAL");
72 lcd.setCursor(0,2);
73 lcd.print(" Presione cualquier");
74 lcd.setCursor(0,3);
75 lcd.print(" tecla...");
76 //Variable que nos permite controlar el tiempo de inactividad
77 tiempo = millis();
78 //Configuración de pines de los LEDS como salida
79 pinMode(LED, OUTPUT);
80 pinMode(LED1, OUTPUT);
81 }
82
83 void loop() {
84 /*Condición para preguntar si ha estado inactivo el menú y
85 despliegue del salvapantallas*/
86 if(millis() - tiempo > 9000){
87     cuenta2 = cuenta2 + 1;
88     if(cuenta2 == 1){
89
90         lcd.clear();
91         lcd.setCursor(0,0);
92         lcd.print(" MENU PRINCIPAL");
93         if(estad01 == 0){
94             lcd.setCursor(10,1);
95             lcd.print("LED1 = OFF");
96         }else{
97             lcd.setCursor(10,1);
98             lcd.print("LED1 = ON");
99         }
100         if(estad02 == 0){
101             lcd.setCursor(10,2);
102             lcd.print("LED2 = OFF");
103         }else{
104             lcd.setCursor(10,2);
105             lcd.print("LED2 = ON");
106         }
107         lcd.setCursor(0,3);
108         lcd.print(" Presione una tecla");
109     }
110     tecla = get_Key(Buttn);
111
112 //Chequeo de si se ha tocado alguna tecla
113 if(tecla != teclaAnt){
114     delay(50);
115     tecla = get_Key(Buttn);
116     if(tecla != teclaAnt){
117         tiempo = millis();

```




```




118     cuenta2 = 0;
119     teclaAnt = tecla;
120     if(tecla >= 0){
121         lcd.blink();
122     }
123 //Acciones del menú según la tecla presionada
124     if(tecla == 2){ //DOWN
125         x++;
126         y = 0;
127         if(x > NumMenuP - 1) x = NumMenuP - 1;
128     }
129     if(tecla == 0){ //UP
130         x--;
131         if( x < 0) x = 0;
132         y = 0;
133         Serial.println(x);
134     }
135     if(tecla == 1){ //RIGTH
136         y++;
137         if( y > NumMaxSubM - 1) y = 0;
138     }
139     if(tecla == 4){ //LEFT
140         y--;
141         if(y < 0) y = NumMaxSubM - 1;
142     }
143     if(tecla == 3){
144         cuenta = cuenta + 1;
145         if (cuenta == 1){
146             x++;
147         }
148         if(x == 1 && y == 0){
149             digitalWrite(LED, HIGH);
150             estado1 = 1;
151         } else if (x == 1 && y == 1){
152             digitalWrite(LED, LOW);
153             estado1 = 0;
154         }
155         if(x == 2 && y == 0){
156             digitalWrite(LED1, HIGH);
157             estado2 = 1;
158         } else if (x == 2 && y == 1){
159             digitalWrite(LED1, LOW);
160             estado2 = 0;
161         }
162     }
163
164     lcd.clear();
165     lcd.print(MenuP[x]);
166     lcd.setCursor(0,1);
167     lcd.print(SubMenus[x][y]);
168 }
169 }
170 delay(50);
171 }
172
173 int get_Key(byte args[]) {
174     int k;

```

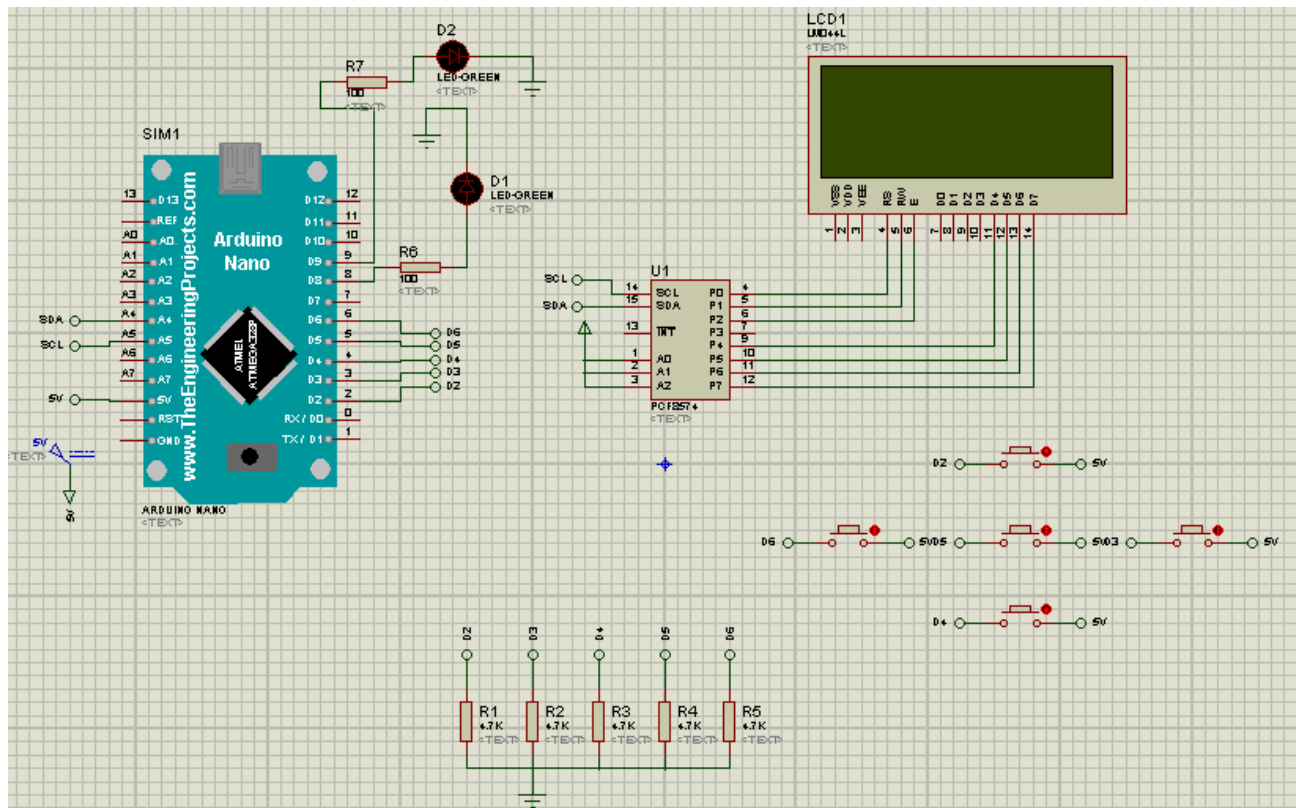
```
175 | if (digitalRead(args[0]) == HIGH) {
176 |     k = 0;
177 | } else if (digitalRead(args[1]) == HIGH) {
178 |     k = 1;
179 | } else if (digitalRead(args[2]) == HIGH) {
180 |     k = 2;
181 | } else if (digitalRead(args[3]) == HIGH) {
182 |     k = 3;
183 | } else if (digitalRead(args[4]) == HIGH) {
184 |     k = 4;
185 | } else {
186 |     k = -1;
187 | }
188 | return k;
189 | }
```


3. LISTA DE COMPONENTES

COMPONENTE	IMAGEN	DESCRIPCIÓN
LCD 20x4		El LCD (Liquid Crystal Display) o pantalla de cristal líquido, un componente empleado para la visualización de contenidos o información de una forma gráfica, mediante caracteres y símbolos.
Arduino Nano		Arduino Nano es una placa de desarrollo de tamaño compacto, completa y compatible con protoboards, basada en el microcontrolador ATmega328P. Tiene 14 pines de entrada/salida digital (de los cuales 6 pueden ser usando con PWM), 6 entradas analógicas, un cristal de 16Mhz, conexión Mini-USB, terminales para conexión ICSP y un botón de reseteo.
PCF8574		Expansor de entradas y salidas digitales por bus I2C

Resistencias de 4.7k y 100 ohms		Componente electrónico diseñado para introducir una resistencia eléctrica determinada entre dos puntos de un circuito eléctrico.
2 LEDs		Fuente de luz constituida por un material semiconductor dotado de dos terminales.
Push Button		Dispositivo utilizado para realizar cierta función. Los botones son de diversas formas y tamaños y se encuentran en todo tipo de dispositivos, aunque principalmente en aparatos eléctricos y electrónicos.

4. DIAGRAMA



5. CONCLUSIONES

Con la realización de esta práctica pudimos comprender con mayor facilidad como se desarrollan este tipo de menús en sistemas embebidos, además con el manejo del LCD y de la interfaz I2C aprendimos la utilidad de este protocolo de comunicación que facilita la conexión de múltiples dispositivos.

La implementación del menú genérico nos facilitará en un futuro la realización de otros sistemas similares que requieran de estos apartados, con ello podremos elaborar sistemas más completos que faciliten al usuario su utilización.

6. BIBLIOGRAFÍA

- [1] L. Llamas. (2016, Abr 28). QUÉ SON Y CÓMO USAR INTERRUPCIONES EN ARDUINO [Online]. Disponible en: <https://www.luisllamas.es/que-son-y-como-usar-interrupciones-en-arduino/>
- [2] R. Rivas. (2017, May 3). Interrupciones [Online]. Disponible en: <https://aprendiendoarduino.wordpress.com/2016/11/13/interrupciones/>
- [3] 5 Hertz Electrónica (2018, Sep 10). Introducción a las LCD [Online] Disponible en: https://www.5hertz.com/index.php?route=tutoriales/tutorial&tutorial_id=9