

SENSORES Y ACTUADORES – TST 2024

Tp8-e4 #58. Salida a una pantalla LCD de 16x2 ó 20x4. de este Proyecto

- ☒ 1. Configurar la pantalla LCD en el entorno de simulación.
- ☒ 2. Implementar el código necesario para mostrar los valores de los sensores.
- ☒ 3. Validar la correcta visualización en la pantalla.

Tutorial – LCD + Sensor de Temperatura (Termómetro de ambiente)



El **objetivo** de este tutorial es modelar y programar un termómetro de exteriores trabajando como Sensor de temperatura con Arduino y LCD . Para ello representaremos los datos medidos mediante un **sensor de temperatura LM35** en un **LCD de 16x2**.

Los **componentes** que vamos a utilizar serán:

1 x Protoboard o Breadboard

1 x Arduino UNO (controlador)

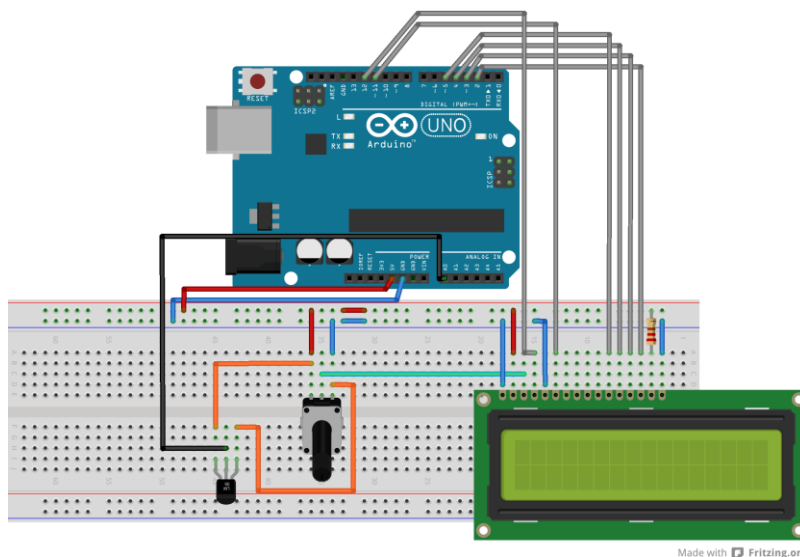
1 x Sensor de temperatura LM35

1 x Potenciómetro (resistencia Variable)

1 x Resistencia de 220 ohmios

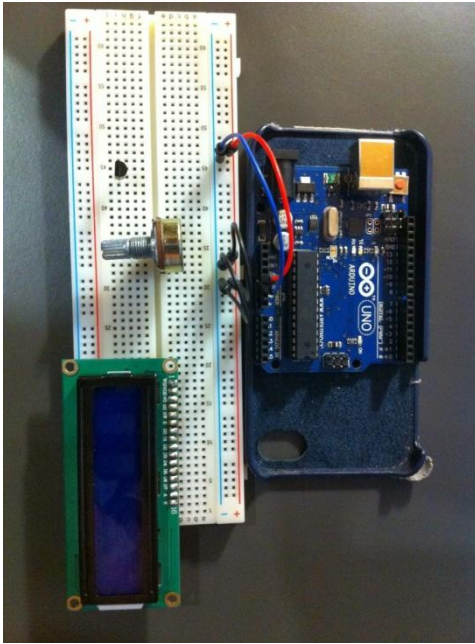
1 x LCD de 16 x 2

Cables



Para comenzar nuestro montaje prepararemos la [protoboard](#). Primeramente conectaremos un cable uniendo nuestro polo positivo de la protoboard con el PIN 5V. El negativo de la protoboard lo conectaremos con GND en nuestro controlador. Por último y para evitar problemas futuros realizaremos dos puentes en la mitad de la board para unir las filas de positivos entre sí y de negativos entre sí.

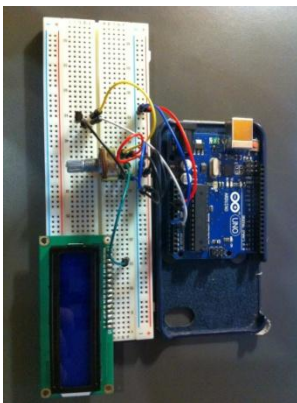
Ahora que ya tenemos la protoboard preparada colocaremos la **LCD**. Para ello la LCD debe prepararse previamente para poder conectarla a una protoboard. Si te perdiste éste tutorial puedes acceder a él [aquí](#) [aquí](#).



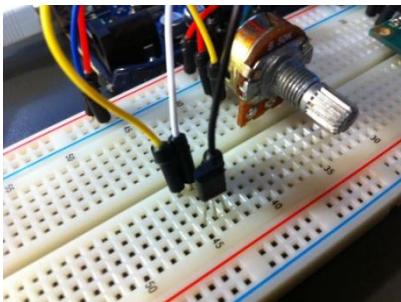
Colocar la **LCD** con los pines como en la imagen ajustándola lo máximo posible a la parte derecha dejando espacio para el resto de componentes. Después, colocaremos nuestro potenciómetro y nuestro sensor más o menos como se muestra en la imagen.

Ya colocado todos los componentes en nuestra protoboard, así que nos dispondremos a comenzar con las conexiones.

Comenzar conectando **el contraste de la pantalla LCD**. Esta parte afecta a los componentes **Potenciómetro y LCD**. Conectaremos la pata de la izquierda del potenciómetro a la fila de positivos de la protoboard (5V). A la pata de la derecha del potenciómetro le conectaremos un cable que una dicha pata con la fila negativa de la protoboard (GND). La pata del centro es la que interacciona con la LCD. Conectaremos la pata central con el PIN tercero de la LCD (empezando por la izquierda mirando el croquis)

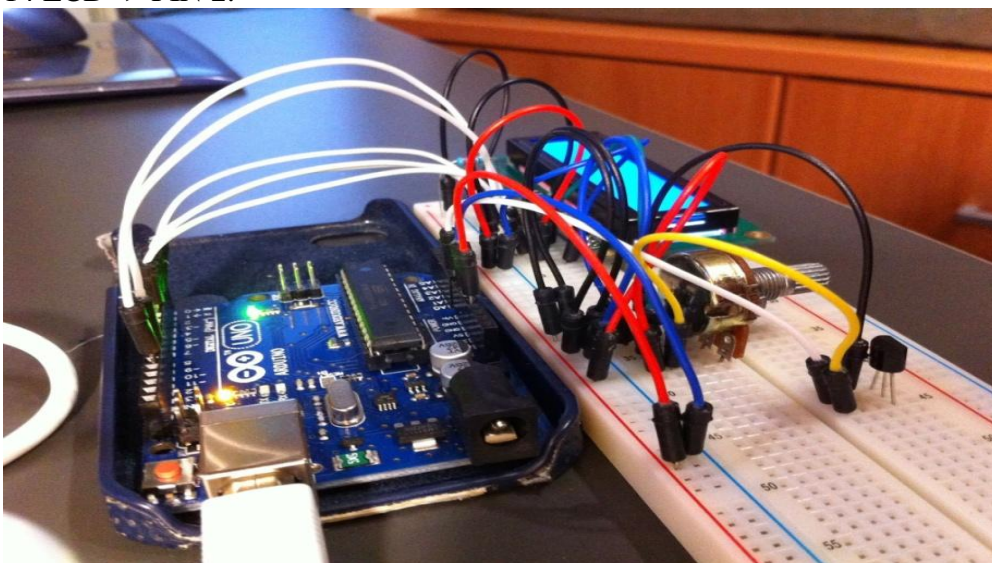


Ahora pasar a conectar el **sensor de temperatura**. Para ello, según la disposición del croquis adjunto, conectaremos la pata de la izquierda del sensor con la izquierda del potenciómetro. De igual forma, conectar la pata de la derecha del potenciómetro con la pata derecha del sensor de temperatura. Ambos componentes se conectan mediante las patas de los extremos al negativo-positivo de la protoboard respectivamente. En cuanto a la pata central del sensor de temperatura la conectaremos al **PIN Analógico de nuestro Arduino UNO A0** dado que el sensor recoge un dato numérico.



Por último quedará la **conexión del LCD** a nuestro controlador. Los pines de la LCD 1, 2, 5, 15 y 16 van conectados a la protoboard (empezando por la izquierda). El 1, 5 y 16 van conectado al negativo de la protoboard (GND). El 2 va conectado mediante un cable a la fila positiva de la protoboard (5V). Por último el 15 va conectado mediante la **resistencia de 220 ohmios** a la fila positiva (5V). Los pines 4, 6, 11, 12, 13, 14 irán conectados a nuestro Arduino de la siguiente forma:

- 4 LCD -> PIN 12
- 6 LCD -> PIN 11 (PWM)
- 11 LCD -> PIN 5 (PWM)
- 12 LCD -> PIN 4
- 13 LCD -> PIN 3 (PWM)
- 14 LCD -> PIN 2.



Ahora es el turno del Software:

```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

//Hasta aquí se declara la librería para la LCD y los pines por donde le va a entrar la información.

```
float centi()
```

```
{// Funcion para leer el dato analogico y convertirlo a digital:
```

```
int dato;
```

```
float c;
```

```
dato=analogRead(A0);
```

```
c = (500.0 * dato)/1023;
```

// Esta fórmula sale de la relación del sensor con los grados. Ésta es fácilmente rastreable por la web pero vamos a intentar explicarla un poco: El sensor de temperatura LM35 responde a variaciones de 10 mV por cada grado centígrado. Si el sensor detecta 1 grado centígrado a la salida del sensor obtendríamos 10 mV. Ejemplo: $26,4^{\circ}\text{C} = 264 \text{ mV} = 0.264 \text{ V}$.

Hay que el convertidor de analógico a digital es de 10 bits de resolución, los valores variarán entre 0 y 1023, entonces $V_{out} = (5V * \text{Dato}) / 1023$ siendo $(0 < \text{Dato} < 1023)$ y para ajustar la escala a grados centígrados: $V_{out} = ((5V * \text{Dato}) * 100) / 1023$

```
return (c);
```

```
}
```

```
float kelvin(float cent)
```

```
{
```

```
float k;
```

```
k=cent+273.15;
```

```
return(k);
```

```
}
```

```
float fahren(float cent)
```

```
{
```

```
float f;
```

```
f=cent*1.8+32;
```

```
return (f);
```

```
}
```

```
float rankin(float cent)
```

```
{
```

```
float r;
```

```
r=(cent + 273.15)*1.8;
```

```
return (r);
```

```
}
```

//Hasta aquí hemos declarado las funciones de conversión del dato analógico de entrada del sensor en grados celsius en sus respectivas equivalencias.

```
void setup() {
```

// Definimos la LCD con dimension 2×16 y definimos los caracteres que deben salir en las filas:

```
lcd.begin(16,2);
```

```
lcd.print(«C= K=»);
```

```
lcd.setCursor(0,1);  
lcd.print(«Temperatura»);  
}  
//Hasta aquí hemos definido qué queremos que salga impreso por la pantalla y el  
tamaño de ésta.  
void loop() {  
  // put your main code here, to run repeatedly:  
  float Centigrados = centi();  
  float Fahrenheit = fahren (Centigrados);  
  float Rankin = rankin (Centigrados);  
  float Kelvin = kelvin (Centigrados);  
  lcd.setCursor(2,0);  
  lcd.print(Centigrados);  
  lcd.setCursor(10,0);  
  lcd.print(Kelvin);  
  delay(200);  
}
```

BIBLIOGRAFIA

- El Cajón de Ardu <https://www.elcajondeardu.com/tutorial-lcd-sensor-de-temperatura/>