



# **Actividad 3. Pantalla de Temperatura**

## **Internet de las Cosas**

### **Ingeniería en Desarrollo de Software**

**Tutor: Marco Alonso Rodríguez Tapia.**

**Alumno: Carlos Alberto Fuentes Mendoza**

**Fecha: 17-febrero-2024**

# Índice

Introducción .....	3
Descripción .....	5
Justificación .....	7
Armado del Circuito .....	8
Codificación .....	12
Emulación del Circuito .....	13
Conclusión .....	16
Referencias .....	17

## Introducción

Un LCD Arduino es un display de cristal líquido que se utiliza para mostrar información en proyectos de Arduino. Los LCD Arduino son populares debido a su facilidad de uso y bajo costo. Uno de los tipos más comunes de LCD Arduino es el LCD 16×2, que tiene una resolución de 16 caracteres por 2 líneas. Un LCD es la forma más económica y sencilla de dotar nuestros proyectos de una pantalla que permita visualizar resultados importantes del sistema automatizado.

Las pantallas LCD tienen diferentes características porque son fabricadas en diferentes tamaños y su uso depende de la necesidad que se tenga, los tamaños comunes en el mercado son el LCD 16×2 (2 filas y 16 caracteres), de 20×4, 20×2 y 40×2. A continuación podemos ver el LCD en diferentes tamaños:



Para que el LCD consiga mostrar los caracteres se utiliza un controlador en estos dispositivos donde el más popular es el Hitachi HD44780, el cual trabaja con LCDs monocromáticos de hasta 80 caracteres alfanuméricos y símbolos. Y poseen una luz de fondo la cual puede ser de color azul o verde y adicionalmente poseen un pin con el cual se puede variar el contraste de la pantalla LCD con Arduino utilizando para eso un potenciómetro.

El LCD 16×2 o 20×4 se conecta a la placa Arduino mediante una serie de pines. Estos pines se utilizan para

enviar señales de control y datos al LCD. Una vez conectado, el LCD Arduino puede mostrar cualquier tipo de información que se envíe a través de los pines. Esto puede incluir texto, números, gráficos, etc.

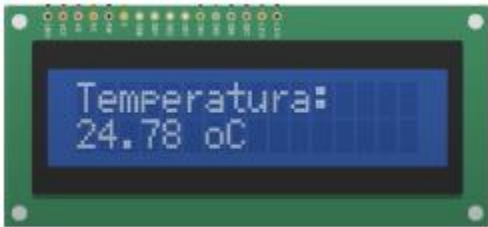
En esta actividad 3, utilizaremos un display de 16x2, a continuación una breve explicación de este tipo de pantallas:

Los displays LCDs de 16x2 tienen 16 pines de los cuales únicamente usaremos 12 pines para la realización de una conexión básica. El pinado del LCD 1602a Arduino con controlador HD44780 se muestra a continuación:

- El pin “RS” controla en que parte de la memoria LCD se están escribiendo los datos.
- El pin de “lectura/escritura” (R/W) selecciona el modo de lectura o de escritura. Si el pin es conectado a +5V el modo de lectura es activado, si el pin es conectado a GND el modo de escritura es activado.
- EL pin para habilitar “enable”, este habilita los registros.
- 8 pines de datos “D00-D07” donde se envían bits para escribir en un registro, también pueden ser usados para leer un registro.
- Pin V0 “de contraste” del display LCD Arduino.
- Pin A y Pin K “de retro-iluminación” (Bklt+ y Bklt-) que le permiten controlar la retroiluminación.
- Pin de alimentación (+5V y GND).

## Descripción

En esta ocasión, se utilizará una pantalla LCD y un sensor de temperatura para crear un circuito. De manera que el sensor de temperatura mida la temperatura y se muestre en grados Celsius en la pantalla. De la siguiente manera:



Para el armado de la alarma, se necesitan los siguientes componentes (están de manera virtual en la plataforma digital Tinkercad):

- Sensor de temperatura
- Pantalla LCD
- Una placa Arduino
- Una placa de pruebas pequeña
- Potenciómetro
- Una resistencia

Es importante recordar que la función LiquidCrystal se utiliza para poder programar la pantalla LCD, por lo que es importante importarla.

Además, se debe crear una variable tipo float para el sensor de temperatura. Dentro de esta, se debe añadir la variable de la pantalla LCD. Entre paréntesis escribir qué pines son los que se utilizan en el Arduino.

En el void setup, poner los píxeles que se pueden utilizar en la pantalla, la cual es de 16x2. Luego mandar a imprimir el mensaje “Temperatura”. Finalmente, en el void loop, averiguar el valor de la temperatura. Para ello, se debe usar la función MAP con analogRead. Después, dividir el valor de temperatura entre 100, y mandar a imprimir a la pantalla LCD.

Actividad:

Utilizando la plataforma online de Tinkercad, crear un circuito donde con un sensor muestre la temperatura exterior en una pantalla LCD.

## Justificación

Las pantallas LCD son una herramienta versátil y útil en proyectos de Arduino, que permiten una presentación clara y efectiva de la información al usuario, mejorando la experiencia general del proyecto.

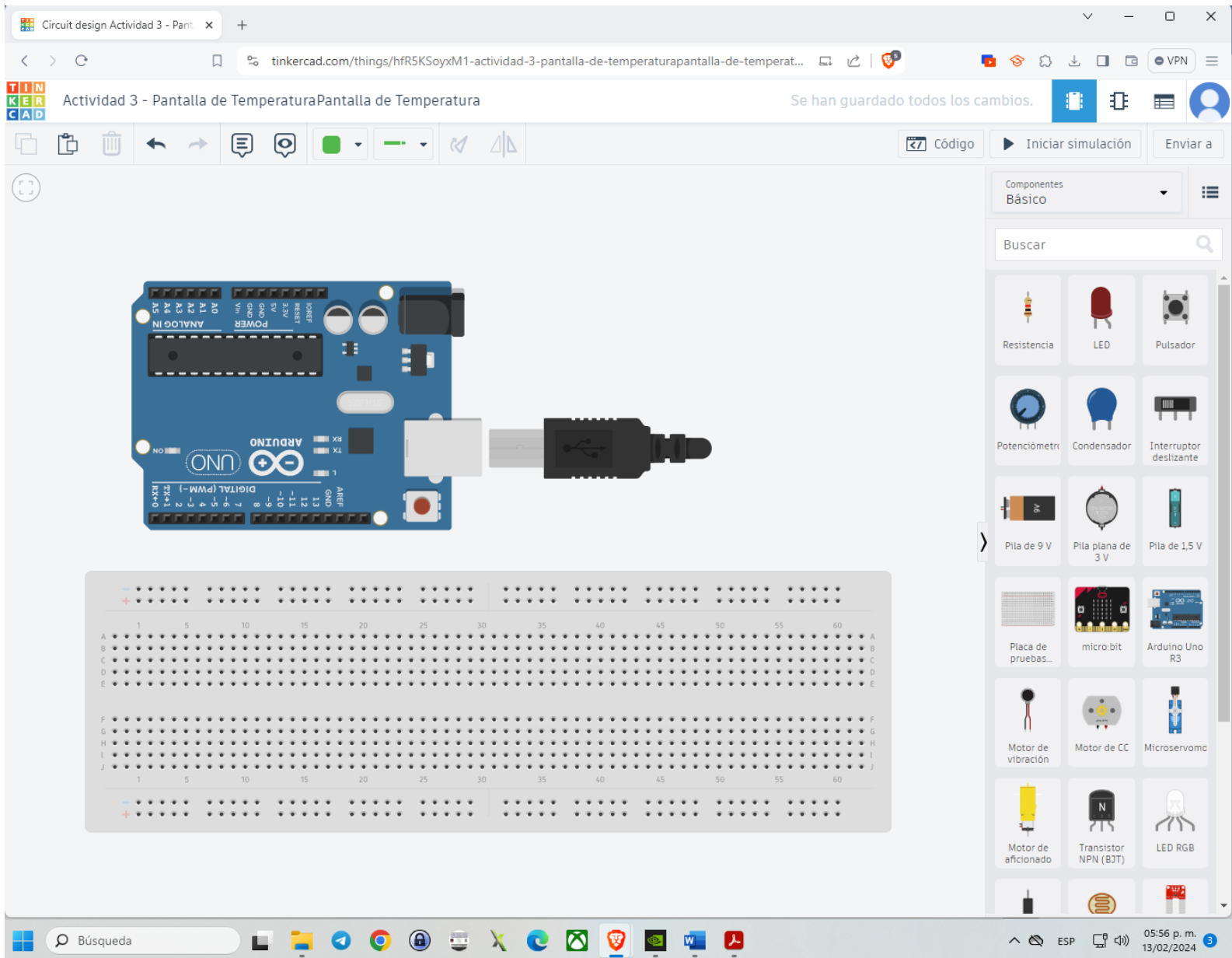
A continuación, se mencionan algunos beneficios significativos:

Las pantallas LCD proporcionan una interfaz de usuario visual que facilita la presentación de datos, mensajes y opciones para interactuar con el dispositivo. Esto permite una experiencia de usuario más intuitiva y agradable. Con una pantalla LCD, puedes mostrar una variedad de datos, como texto, números, gráficos simples e incluso iconos personalizados. Esto permite una presentación más clara y comprensible de la información relevante para el usuario. Las pantallas LCD permiten mostrar información en tiempo real, como lecturas de sensores, estados de dispositivos o cualquier otro dato dinámico que sea importante para el usuario. Esto es especialmente útil en proyectos de monitoreo y control. En comparación con otras formas de visualización, como las pantallas LED, las pantallas LCD tienden a consumir menos energía, lo que las hace más adecuadas para proyectos que requieren eficiencia energética o que funcionan con baterías.

Existen bibliotecas de software disponibles para controlar pantallas LCD con Arduino, lo que simplifica su integración en proyectos y reduce el tiempo de desarrollo. Estas bibliotecas proporcionan funciones para escribir texto, controlar el cursor, cambiar el brillo y más. Variedad de Opciones: Las pantallas LCD vienen en una variedad de tamaños, tipos y características. Desde simples LCDs de caracteres hasta LCDs gráficas a color, OLEDs, y más. Esto te permite elegir la pantalla que mejor se adapte a las necesidades y requisitos de tu proyecto específico.

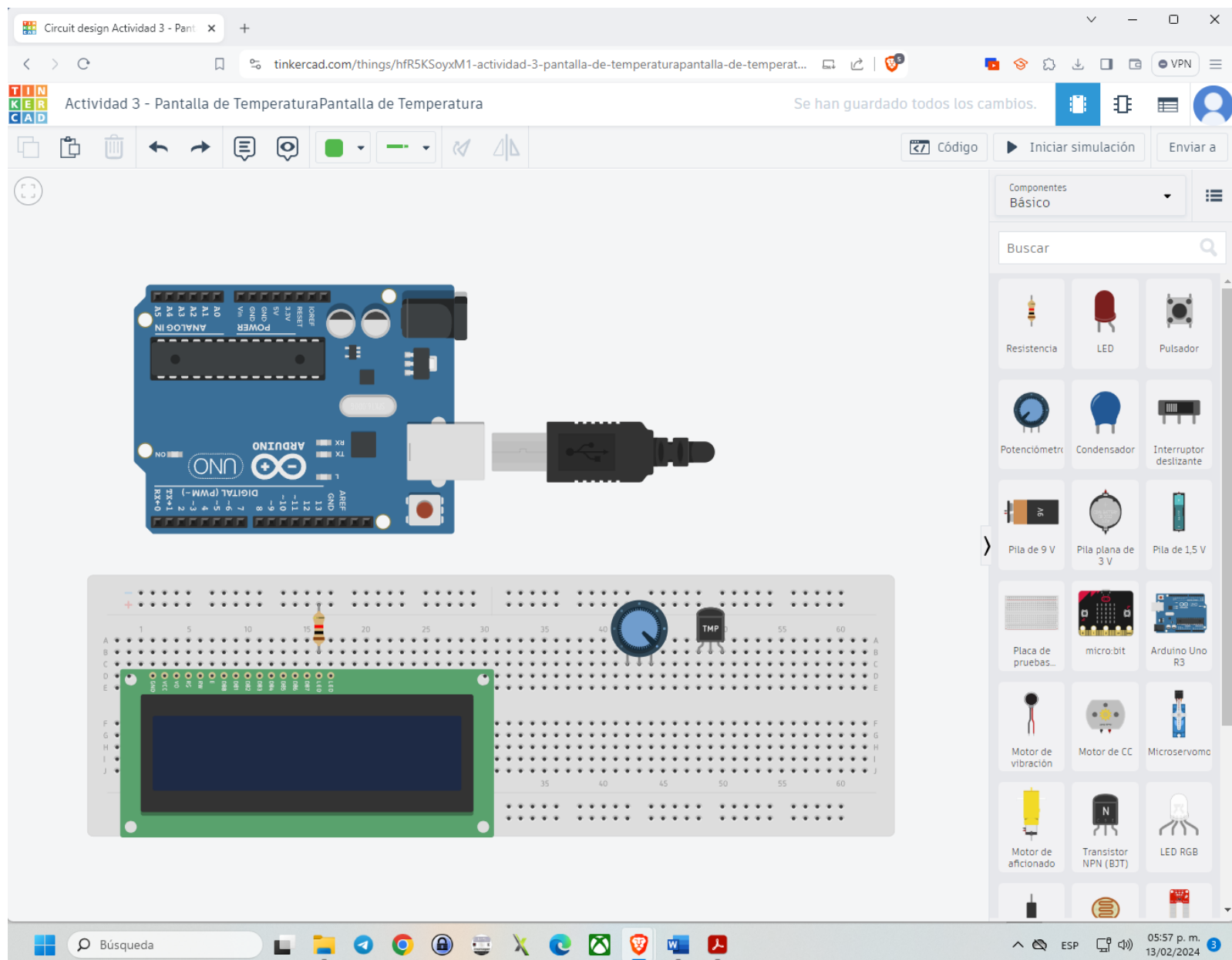
## Armado del Circuito

En la siguiente captura se observa el armado del circuito, comenzando con la placa de Arduino y la placa Protoboard.

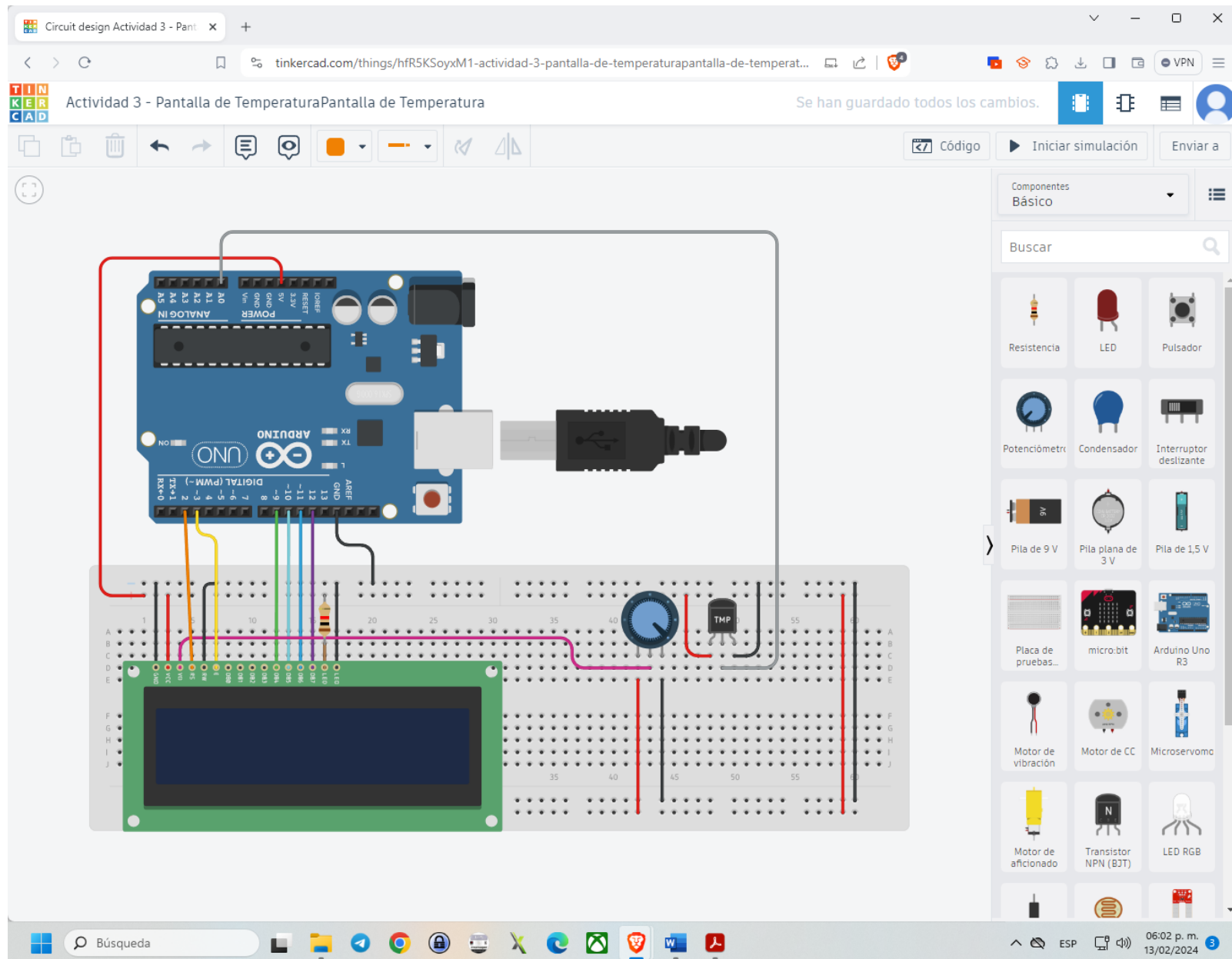




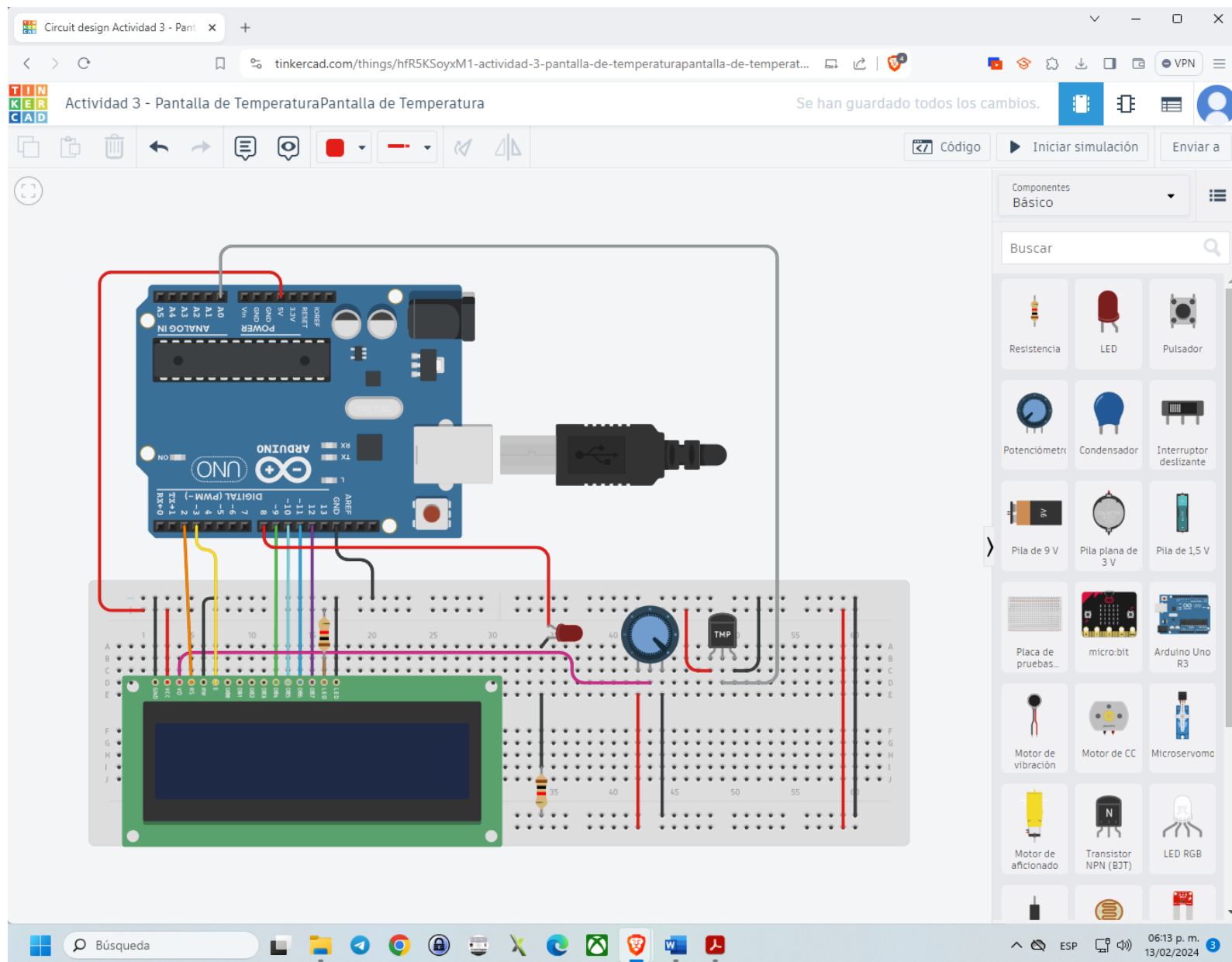
En la siguiente captura se observa que se agregan el sensor de temperatura, pantalla LCD, potenciómetro y resistencia.



En la siguiente captura se observa que se agrega a la protoboard el cable y las conexiones del circuito entre la placa Arduino, el sensor de temperatura, pantalla LCD, potenciómetro y la resistencia.



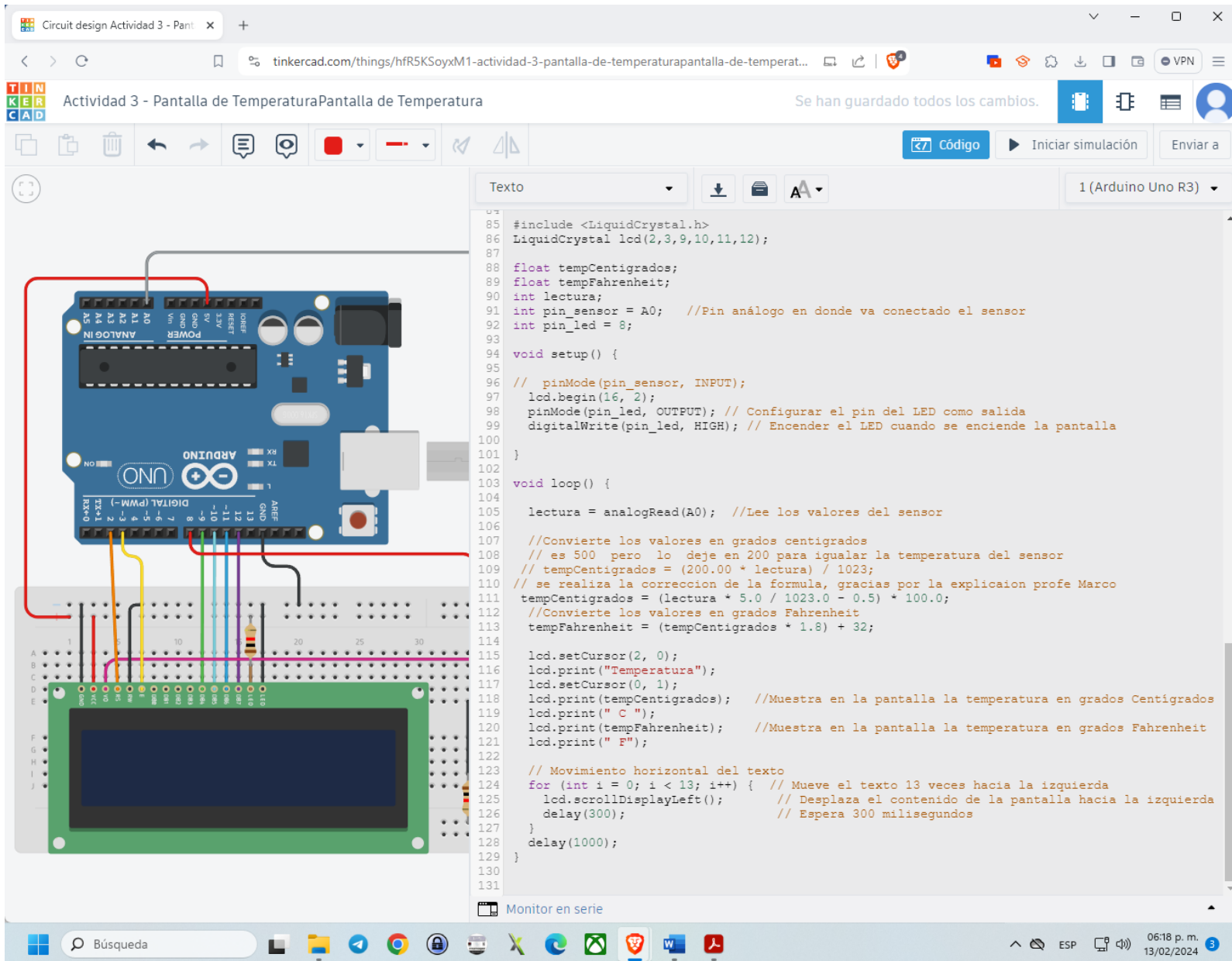
En la siguiente captura se observa que se agrega a la protoboard y a la placa Arduino, un foco led y resistencia que se enciende cuando la pantalla esta encendida.



## Codificación

En la siguiente captura se muestra el código utilizado para realizar el funcionamiento de la Pantalla de Temperatura, adicionalmente, se agrega el código para que la información se desplace hacia la izquierda.

Nota: También se adjunta el código de Arduino a la plataforma para una mejor visualización

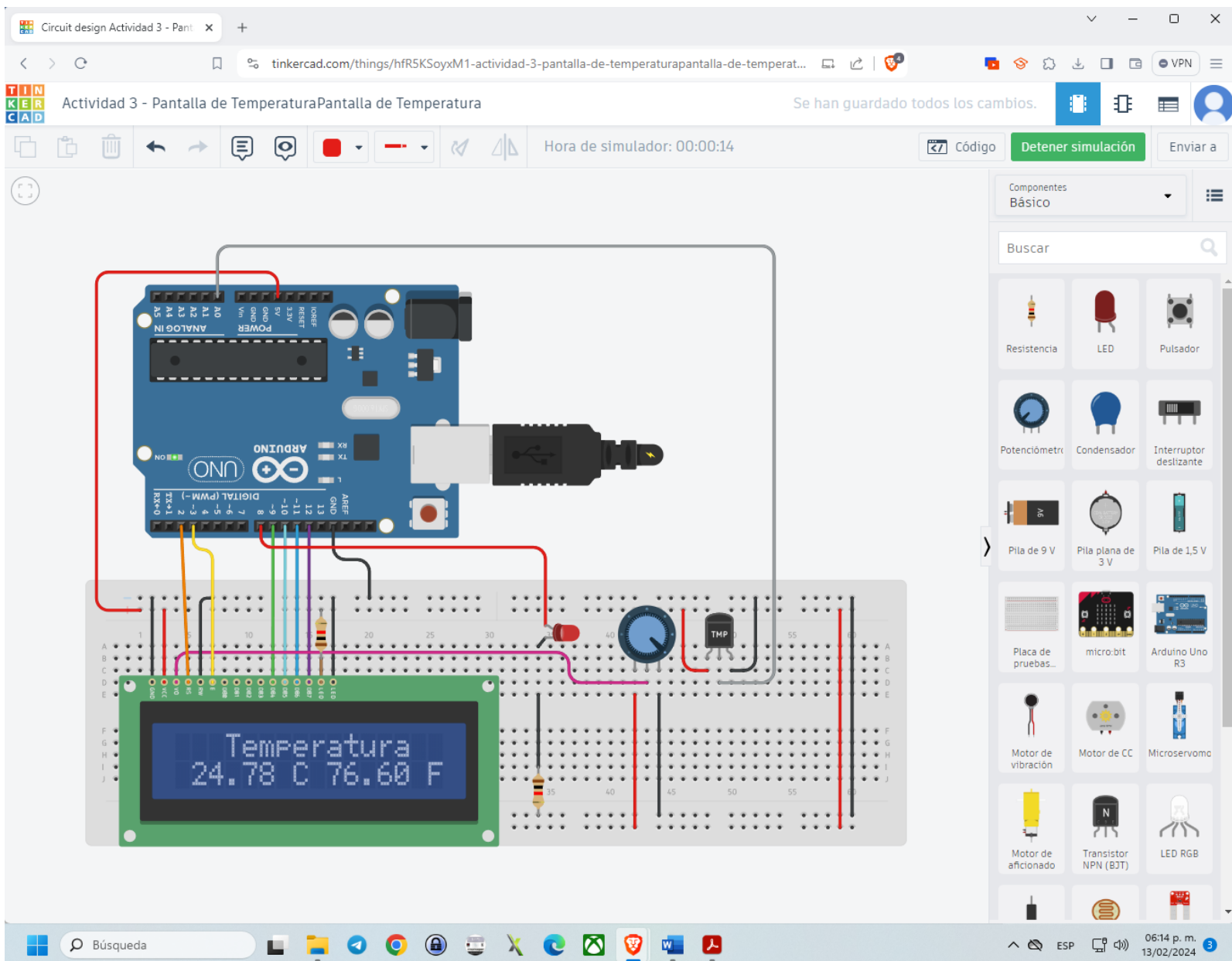


The screenshot displays the Tinkercad web interface for a project titled "Actividad 3 - Pantalla de Temperatura". On the left, a circuit diagram shows an Arduino Uno R3 connected to a breadboard. A temperature sensor (DS18B20) is connected to the Arduino's analog pin A0. An LCD display is connected to the Arduino's I2C pins (SCL to A5, SDA to A4). The code editor on the right contains the following Arduino sketch:

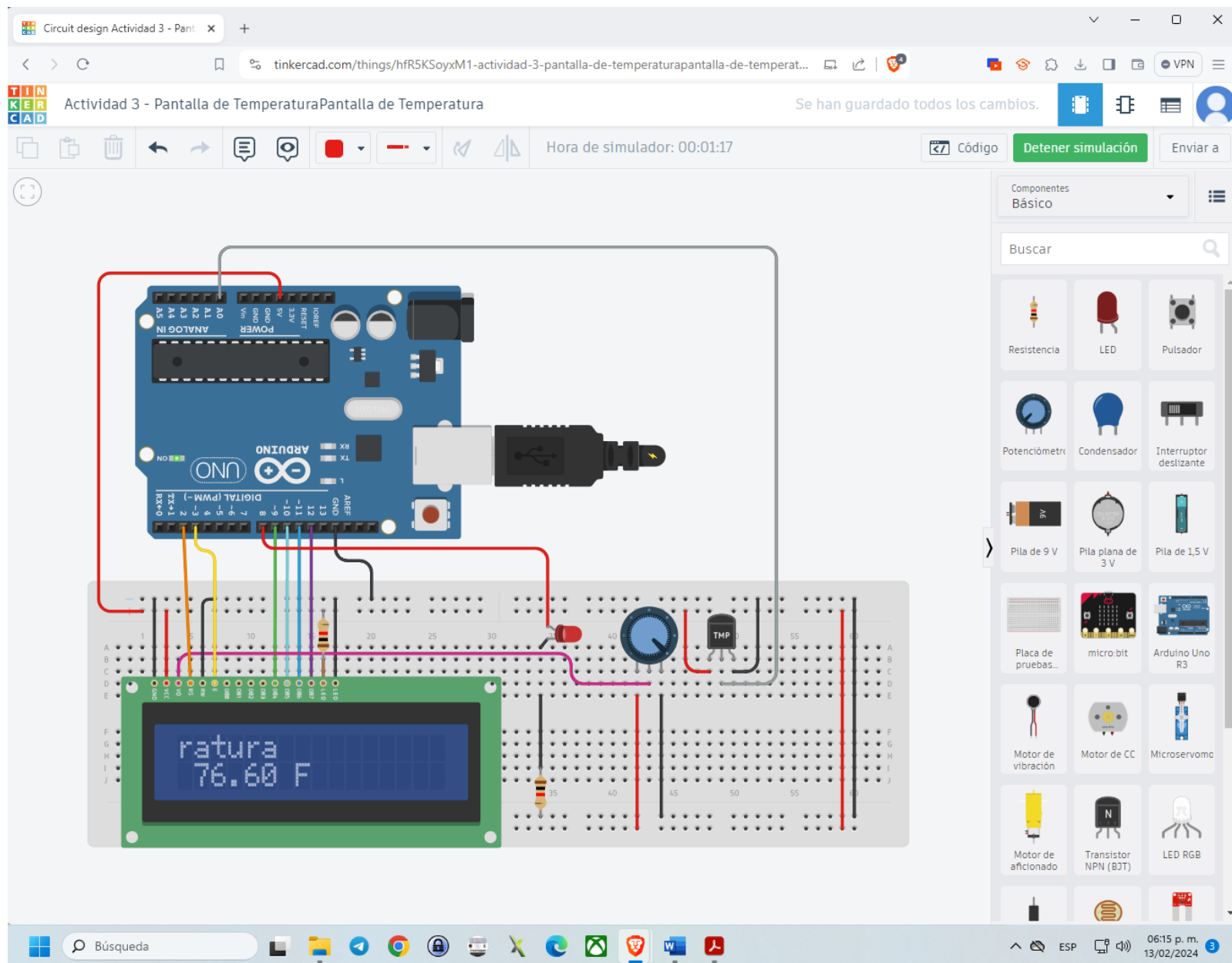
```
85 #include <LiquidCrystal.h>
86 LiquidCrystal lcd(2,3,9,10,11,12);
87
88 float tempCentigrados;
89 float tempFahrenheit;
90 int lectura;
91 int pin_sensor = A0; //Pin analógico en donde va conectado el sensor
92 int pin_led = 8;
93
94 void setup() {
95
96 // pinMode(pin_sensor, INPUT);
97 lcd.begin(16, 2);
98 pinMode(pin_led, OUTPUT); // Configurar el pin del LED como salida
99 digitalWrite(pin_led, HIGH); // Encender el LED cuando se enciende la pantalla
100 }
101
102 void loop() {
103
104 // Lee los valores del sensor
105 lectura = analogRead(A0); //Lee los valores del sensor
106
107 //Convierte los valores en grados centigrados
108 // es 500 pero lo deje en 200 para igualar la temperatura del sensor
109 tempCentigrados = (200.00 * lectura) / 1023;
110 // se realiza la correccion de la formula, gracias por la explicacion profe Marco
111 tempCentigrados = (lectura * 5.0 / 1023.0 - 0.5) * 100.0;
112 //Convierte los valores en grados Fahrenheit
113 tempFahrenheit = (tempCentigrados * 1.8) + 32;
114
115 lcd.setCursor(2, 0);
116 lcd.print("Temperatura");
117 lcd.setCursor(0, 1);
118 lcd.print(tempCentigrados); //Muestra en la pantalla la temperatura en grados Centigrados
119 lcd.print(" C ");
120 lcd.print(tempFahrenheit); //Muestra en la pantalla la temperatura en grados Fahrenheit
121 lcd.print(" F");
122
123 // Movimiento horizontal del texto
124 for (int i = 0; i < 13; i++) { // Mueve el texto 13 veces hacia la izquierda
125   lcd.scrollDisplayLeft(); // Desplaza el contenido de la pantalla hacia la izquierda
126   delay(300); // Espera 300 milisegundos
127 }
128 delay(1000);
129 }
130
131
```

## Emulación del Circuito

En la siguiente captura se muestra como al iniciar la simulación, se enciende el foco led y comienza el funcionamiento de la pantalla de temperatura, en donde se puede observar que se muestra la temperatura que se solicita en la actividad 3, 24.78 C y 76.60 F.



En la siguiente captura se observa como se desplaza hacia la izquierda la información mostrada en la pantalla LCD hasta finalizar el ciclo.



The image is a screenshot of a Tinkercad web interface. At the top, the browser address bar shows the URL 'tinkercad.com/things/hfR5KSoym1-actividad-3-pantalla-de-temperaturapantalla-de-temperat...'. The main workspace displays a simulation of an Arduino Uno microcontroller board connected to a breadboard. The breadboard circuit includes a 9V battery, a 10k potentiometer, a 100k resistor, a 100 ohm resistor, a 100 ohm LED, a 100 ohm TMP sensor, and a 100 ohm LED. The LCD screen on the breadboard displays 'Temper 24.78 C'. The top toolbar contains various simulation controls, including a 'Detener simulación' button. The right sidebar shows a list of components available for the simulation, such as 'Resistencia', 'LED', 'Pulsador', 'Potenciómetro', 'Condensador', 'Interruptor deslizante', 'Pila de 9 V', 'Pila plana de 3 V', 'Pila de 1,5 V', 'Placa de pruebas...', 'micro:bit', 'Arduino Uno R3', 'Motor de vibración', 'Motor de CC', 'Microservo', 'Motor de aficionado', 'Transistor NPN (BJT)', and 'LED RGB'. The bottom taskbar shows the Windows Start button, a search bar, and several application icons, including a web browser, file explorer, and system clock showing '06:15 p.m. 13/02/2024'.

[https://www.tinkercad.com/things/hfR5KSoyxM1-actividad-3-pantalla-de-temperaturapantalla-de-temperatura/editel?returnTo=%2Fdashboard%3Ftype%3Dcircuits%26collection%3Ddesigns&sharecode=-WuWu0h2WQDxboHhXsPb\\_aIS1FPXkuUBvxS9nldeWBQ](https://www.tinkercad.com/things/hfR5KSoyxM1-actividad-3-pantalla-de-temperaturapantalla-de-temperatura/editel?returnTo=%2Fdashboard%3Ftype%3Dcircuits%26collection%3Ddesigns&sharecode=-WuWu0h2WQDxboHhXsPb_aIS1FPXkuUBvxS9nldeWBQ)

## Conclusión

En esta actividad aprendí muchas cosas importantes, entre las cuales destacan que aprendí que hay varios tipos de pantallas LCD disponibles para su uso con Arduino y que estas pueden ser LCDs de caracteres (como 16x2, 20x4, etc.), LCDs gráficas (que pueden mostrar gráficos personalizados), TFT LCDs a color, OLEDs, y más. También aprendí que las pantallas LCD se conectan a Arduino utilizando diferentes protocolos de comunicación, como paralelo, I2C (Inter-Integrated Circuit) o SPI (Serial Peripheral Interface), dependiendo del tipo de pantalla y del modelo de Arduino. En lo personal una gran ayuda para su conexión fue que las instrucciones se encuentran en su mismo sitio web y/o en otras más, compartidas por la comunidad de desarrolladores. También aprendí que para controlar una pantalla LCD, generalmente se utiliza una biblioteca específica en el entorno de desarrollo de Arduino. Fue muy satisfactorio aprender que estas bibliotecas proporcionan funciones para inicializar la pantalla, escribir texto en ella, controlar el cursor, y más. Otra cosa que aprendí fue que la resolución de una pantalla LCD se especifica en términos de la cantidad de caracteres o píxeles que puede mostrar. Por ejemplo, en esta actividad se utilizó una pantalla LCD de 16x2 lo cual significa que esta pantalla LCD puede mostrar hasta 16 caracteres en dos líneas.

Concluyo haciendo mención que, en lo general, aprendí que las pantallas LCD son una herramienta, que permiten una presentación clara y efectiva de la información al usuario, mejorando así la experiencia general del proyecto, y que comúnmente, estas pantallas LCD son utilizadas en una amplia gama de proyectos de Arduino, como en sistemas de monitoreo y control, dispositivos de medición, relojes, termostatos, estaciones meteorológicas, y mucho más.

A continuación, se anexa el enlace al repositorio GitHub en donde se aloja esta actividad.

[https://github.com/charlyfu/Internet\\_de\\_las\\_Cosas](https://github.com/charlyfu/Internet_de_las_Cosas)

---



## Referencias

Arduino para todos. (2017, April 28). Sensor de temperatura LM35 con Arduino + código [Video].

YouTube. [https://www.youtube.com/watch?v=6anOeEcX\\_Io](https://www.youtube.com/watch?v=6anOeEcX_Io)

Techmake, E. (2020, May 15). Empezando con Arduino - 4B: LCD + Sensor de Temperatura. Techmake Solutions. <https://techmake.com/blogs/tutoriales/empezando-con-arduino-4b-lcd-sensor-de-temperatura>

C, S. (2023, January 26). LCD arduino. Control Automático Educación.

<https://controlautomaticoeducacion.com/arduino/lcd/>

Del Valle Hernández, L. (2021, March 23). LCD con Arduino texto en movimiento paso a paso.

Programarfacil Arduino y Home Assistant. <https://programarfacil.com/blog/arduino-blog/texto-en-movimiento-en-un-lcd-con-arduino/>