

Identificación de elementos.

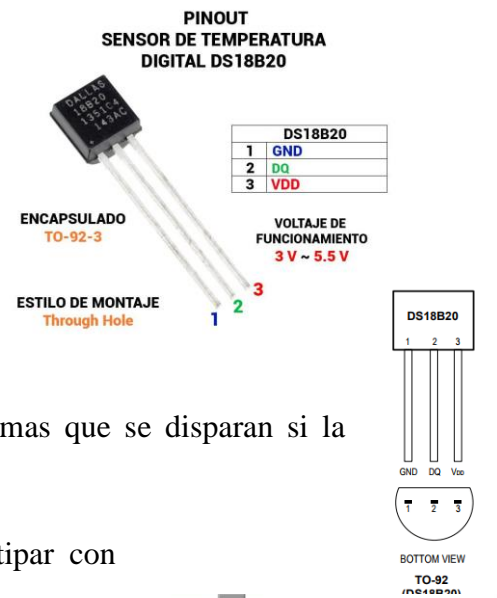
Sensor de temperatura DS18B20

El sensor de temperatura es el componente que recoge la temperatura del exterior y lo transforma en en señal digital o electrónica que envía a la placa de arduino.

El sensor de temperatura DS18B20 comunica de forma digital e incorpora una memoria de 64-bit para almacenar la dirección de cada sensor.

Cuenta con tres terminales:

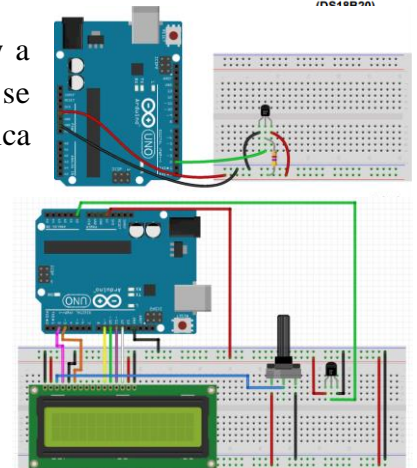
- Vcc: es la tensión de alimentación. (3V-5,5V)
- GND: es el pin conectado a la referencia de 0V.
- pin Data: donde se recibirán todos los datos en el protocolo 1-Wire. Utiliza comunicación por protocolo serial digital OneWire, que permite enviar y recibir datos utilizando solo 1 cable.



El rango de temperaturas es entre -55°C-125°C. Tiene 2 alarmas que se disparan si la temperatura medida es mayor o menor que la máxima o mínima.

Conexión con arduino:

- La disposición de los pines más adecuada para prototipar con Arduino es el TO-92.
- Para alimentar el sensor hay 2 formas: a través del pin Vcc y a través del pin Data, en modo parásito. En los dos casos siempre se pone una resistencia pull-up con el pin Data debido a la electrónica para controlar el bus de comunicación. con un pin en modo INPUT_PULLUP con la instrucción pinMode.
- Una resistencia de 4.7k del pin de datos y Vcc
- Es necesario utilizar las librerías: Dallas Temperature y OneWire, para leer el sensor con el arduino.
- Diagrama de conexiones: con el arduino y la pantalla LCD
- Código:



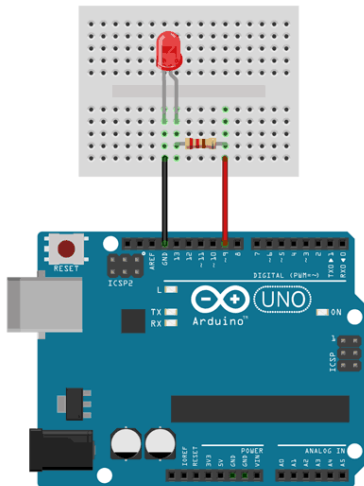
```
#include <OneWire.h> //Se importan las librerías
#include <DallasTemperature.h>
#include <LiquidCrystal.h> //incluimos la libreria Liquidcrystal para poder programar nuestra LCD
#define Pin 2 //Se declara el pin donde se conectará la DATA
OneWire ourWire(Pin); //Se establece el pin declarado como bus para la comunicación OneWire
DallasTemperature sensors(&ourWire); //Se llama a la librería DallasTemperature
void setup() {
    Serial.begin(9600);
    sensors.begin(); } //Se inician los sensores
void loop() {
```

```
sensors.requestTemperatures();    //Prepara el sensor para la lectura

Serial.print(sensors.getTempCByIndex(0)); //Se lee e imprime la temp. en grados Centígrados
Serial.println(" °C");
delay(1000);}                    //Se provoca una parada de 30 mins antes de la próxima lectura
```

Luces LED

- Para indicar si la temperatura está entre los rangos indicados utilizaremos luces LED.
- Diagrama de conexiones: con el arduino.



-Código:

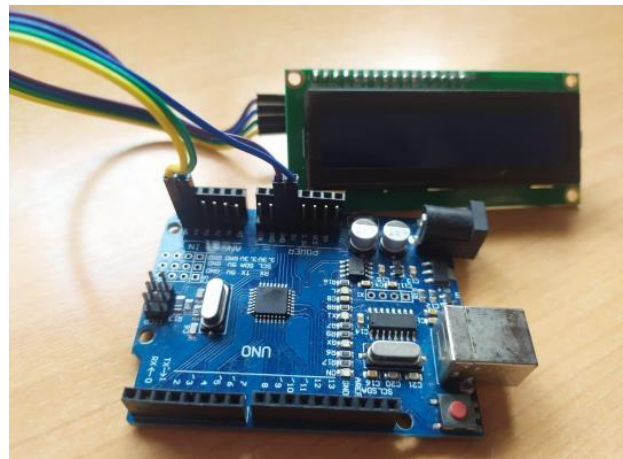
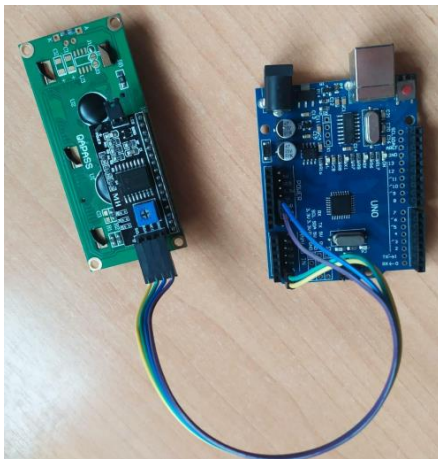
```
const int LED=13;
void setup()
{
  pinMode(LED, OUTPUT);
}
void loop()
{
  digitalWrite(LED, HIGH);
  delay(1000);
  digitalWrite(LED, LOW);
  delay(1000);
}
```

Pantalla LCD 16x2

- Para enseñar los datos a través del arduino, se va a usar un display de 16 caracteres alfanuméricos por 2 columnas, la LCD HD44780, como esta pantalla tiene una configuración de pines bastante compleja, decimos usar el adaptador de pantalla I2C PCF8574, quedándose las conexiones reducidas a tierra (GND), fuente de voltaje (Vcc 5V), y dos pines analógicos (A4 y A5).

- Para poder usar este adaptador ha sido necesaria descargarse una librería que será adjuntada a Github. El código para poner la pantalla en marcha es el siguiente:

```
void setup()
{
  lcd.begin(16, 2);
  lcd.clear();
  lcd.print("Temperatura=");
  sensor=analogRead(A0); //indica el sensor de donde se lee
  lcd.setCursor(0,1);
  lcd.print("temperatura");
  delay(1000);
}
void loop()
{
  // en función de las mediciones de los sensores la pantalla tendrá determinadas respuestas
}
```



Placa ARDUINO UNO R3

Arduino es una placa basada en un microcontrolador ATMEGA con circuitos integrados donde se pueden grabar instrucciones programadas en el entorno Arduino IDE. Este microcontrolador posee tanto una interfaz de entrada como de salida, donde podemos conectar distintos tipos de periféricos. Por ejemplo, en nuestro proyecto conectaremos a la interfaz de entrada un sensor de temperatura DS18B20 y a la interfaz de salida tanto la pantalla LCD como las luces leds que transmitirán al exterior la información detectada por el sensor.

La placa Arduino UNO R3 tiene 14 pines digitales de entrada y salida, entre los cuales 6 se pueden usar como salidas PWM y otros 6 como entradas analógicas. Esta placa se diferencia del resto en que cuenta con el Atmega8U2 programado como convertidor de USB a serie. Arduino posee convertidores ADC (analógico a digital) que nos permiten realizar lecturas analógicas del mundo exterior, imprescindibles en la inmensa mayoría de los proyectos.

El problema de los Arduino es que tienen una precisión y una estabilidad muy justa. Los ADC internos de Arduino no tienen una mala resolución, el problema es que en algunos proyectos pueden ser un tanto imprecisos.

Las placas de Arduino UNO poseen 10 bits, es decir, $2^{10}=1024$ valores de resolución y si comparamos estos con los 5V que nos suministra Arduino UNO, nos encontramos con la siguiente resolución:

$$5 \text{ V}/1024 = 4.8 \text{ mV de resolución.}$$

Esta precisión puede ser más que suficiente en la mayoría de los proyectos simples, aunque en otros se queda corta.

Otro problema muy común es que el Arduino UNO no está bien aislado de interferencias que pueden existir a la hora de realizar la conversión y son demasiado sensibles al ruido externo.

