

CATEGORÍAS DEL BLOG

Tutoriales

Noticias

ÚLTIMOS COMENTARIOS



Alberto

en Configuración del
módulo bluetooth HC-
05...



DANIEL PUGLIESI

en Tutorial MPU6050,
Acelerómetro y
Giroscopio



aitor

en Tutorial Módulo
Controlador de servos
PCA9685...



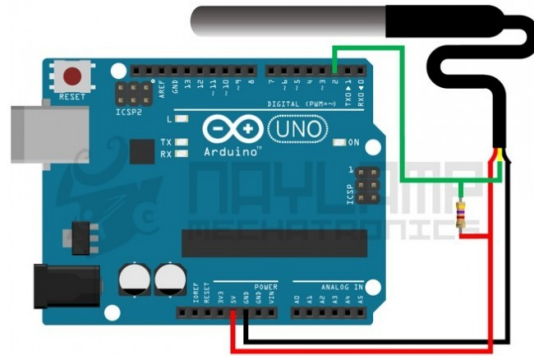
jose

en Tutorial módulo
Lector RFID RC522

BÚSQUEDA DE BLOGS



TUTORIAL SENSOR DIGITAL DE TEMPERATURA DS18B20

 250191


Sensor DS18B20

El DS18B20 es un sensor digital de temperatura que utiliza el protocolo 1-Wire para comunicarse, este protocolo necesita solo un pin de datos para comunicarse y permite conectar más de un sensor en el mismo bus.

El sensor DS18B20 es fabricado por Maxim Integrated, el encapsulado de fábrica es tipo TO-92 similar al empleado en transistores pequeños. La presentación comercial más utilizada por conveniencia y robustez es la del sensor dentro de un tubo de acero inoxidable resistente al agua, con el que trabajemos este tutorial.



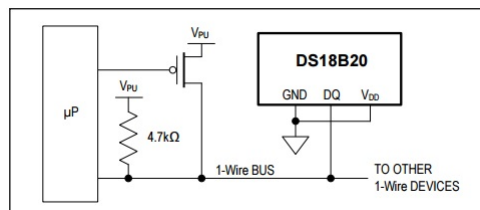
Con este sensor podemos medir temperatura desde los -55°C hasta los 125°C y con una resolución programable desde 9 bits hasta 12 bits.

Cada sensor tiene una dirección única de 64bits establecida de fábrica, esta dirección sirve para identificar al dispositivo con el que se está comunicando, puesto que en un bus 1-wire pueden existir más de un dispositivo.

El sensor tiene dos métodos de alimentación:

Alimentación a través del pin de datos:

De esta forma, el sensor internamente obtiene energía del pin de datos cuando este se encuentra en un estado alto y almacena carga en un condensador para cuando la línea de datos esté en un estado bajo, a esta forma de obtener energía se le llama "Parasite Power" y se usa cuando el sensor debe conectarse a grandes distancias o en donde el espacio es limitado, puesto que de esta forma no se necesita la línea de VDD. El diagrama para su conexión debe ser de la siguiente forma:

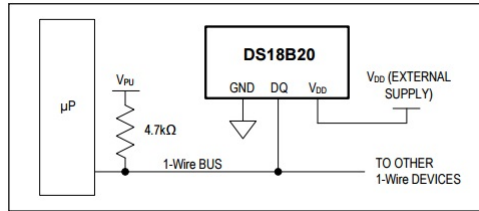


Notar que el pin GND y VDD están ambos conectados a GND, esto es indispensable para que se active el Parasite Power. EL MOSFET en la imagen es necesario para cuando se realicen conversiones de temperatura o copiar datos desde la memoria de circuito de la EEPROM, en estas operaciones la corriente de operación aumenta y si solo se suministra energía a través de la resistencia pueden causar caídas de voltaje en el condensador interno.

Alimentación usando una fuente externa:

De esta forma el sensor se alimenta a través del pin VDD, de esta forma el voltaje es estable e independiente del tráfico del bus 1-wire.

El diagrama de conexión es de la siguiente forma:



Esta forma de alimentación es la más recomendada y es la utilizada en este tutorial.

Librerías para el DS18B20 en Arduino

Para poder trabajar el DS18B20 en Arduino necesitamos dos librerías:

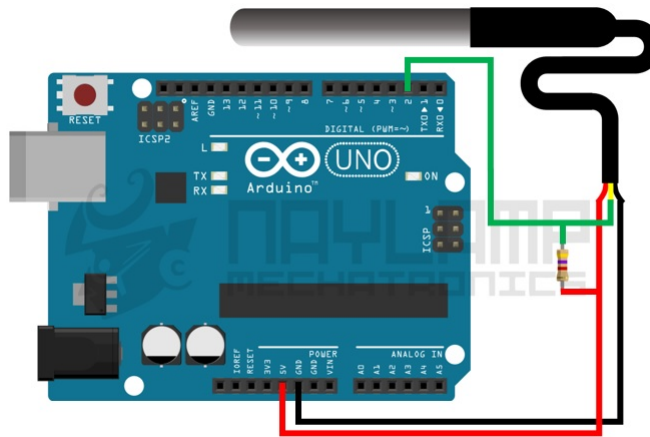
- Librería **OneWire**, descarga: <https://github.com/PaulStoffregen/OneWire>

En esta librería está implementado todo el protocolo del bus 1-wire. Y puede usarse tanto para el DS18B20 como para otros dispositivos 1-wire, para mayor información sobre la librería: http://www.pjrc.com/teensy/td_libs_OneWire.html

-Librería **DallasTemperature**, descarga: <https://github.com/milesburton/Arduino-Temperature-Control-Library>

En esta librería están implementadas las funciones necesarias para poder realizar las lecturas o configuraciones del DS18B20, para mayor información de la librería revisar: http://www.milesburton.com/Dallas_Temperature_Control_Library

Conexiones del DS18B20 con Arduino:



Notar que es necesario poner una resistencia Pull-Up de 4.7K, y pueden usar un protoboard para facilitar la conexión.

Ej.1: Realizando lecturas de temperatura con el DS18B20

Después de instalar las librerías y realizar la conexión anterior podemos realizar las lecturas de temperatura, para eso usamos el siguiente sketch:

```
#include <OneWire.h>
#include <DallasTemperature.h>

OneWire ourWire(2); //Se establece el pin 2 como bus OneWire

DallasTemperature sensors(&ourWire); //Se declara una variable u objeto para nuestro sensor

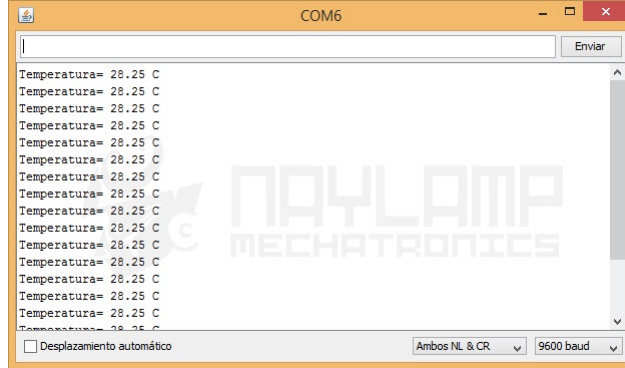
void setup() {
  delay(1000);
  Serial.begin(9600);
  sensors.begin(); //Se inicia el sensor
}

void loop() {
  sensors.requestTemperatures(); //Se envía el comando para leer la temperatura
  float temp= sensors.getTempCByIndex(0); //Se obtiene la temperatura en °C

  Serial.print("Temperatura= ");
  Serial.print(temp);
  Serial.println(" °C");
  delay(100);
}
```

Como se observa medir la temperatura es simple, son necesarias solo dos líneas en el void loop() para realizar esta tarea.

El resultado es el siguiente:



Conectando varios sensores de temperatura:

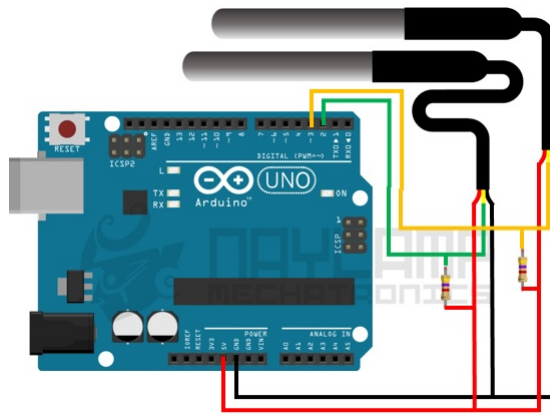
Tenemos dos opciones o métodos que podemos usar cuando necesitamos leer más de un sensor de temperatura.

El primer método es manejando cada sensor con un pin diferente del Arduino. De esta forma sí tenemos 3 sensores, necesitaremos usar 3 pines digitales del Arduino.

Otra forma es usar el mismo pin para todos los sensores, en otras palabras todos los sensores se conectan al mismo bus 1-Wire, y como cualquier bus, cada elemento o dispositivo posee una identificación o dirección. En el caso del DS18B20 necesitamos averiguar su dirección que es única y viene establecida de fábrica.

Ej.2: Usando varios DS18B20 en diferentes pines del Arduino:

Para este caso las conexiones son las siguientes:



Cada sensor trabaja con un pin diferente y necesita su propia resistencia Pull-Up de 4.7K.

El código para realizar las lecturas es el siguiente:

```
#include <OneWire.h>
#include <DallasTemperature.h>

OneWire ourWire1(2);           //Se establece el pin 2 como bus OneWire
OneWire ourWire2(3);           //Se establece el pin 3 como bus OneWire

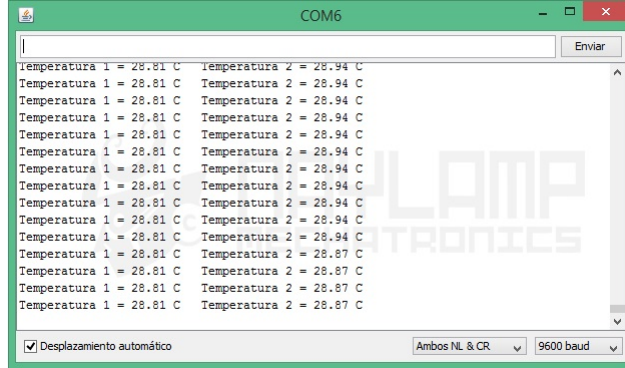
DallasTemperature sensors1(&ourWire1); //Se declara una variable u objeto para nuestro sensor1
DallasTemperature sensors2(&ourWire2); //Se declara una variable u objeto para nuestro sensor2

void setup() {
  delay(1000);
  Serial.begin(9600);
  sensors1.begin(); //Se inicia el sensor 1
  sensors2.begin(); //Se inicia el sensor 2
}

void loop() {
  sensors1.requestTemperatures(); //Se envía el comando para leer la temperatura
  float temp1= sensors1.getTempCByIndex(0); //Se obtiene la temperatura en °C del sensor 1

  sensors2.requestTemperatures(); //Se envía el comando para leer la temperatura
  float temp2= sensors2.getTempCByIndex(0); //Se obtiene la temperatura en °C del sensor 2

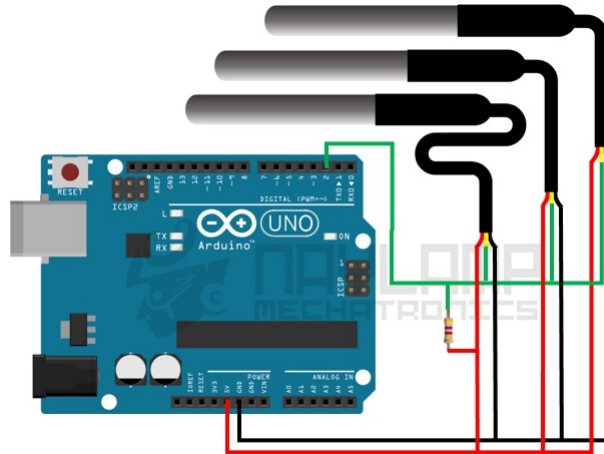
  Serial.print("Temperatura 1 = ");
  Serial.print(temp1);
  Serial.print(" C");
  Serial.print("    Temperatura 2 = ");
  Serial.print(temp2);
  Serial.println(" C");
  delay(100);
}
```



Esta forma de conectar dos o más sensores es fácil de entender e implementar y es útil cuando son pocos sensores o simplemente tenemos pines disponibles para conectar más DS18B20 como en un Arduino Mega.

Ej.3: Usando varios DS18B20 con un solo pin del Arduino:

En este caso conectamos todos los sensores al mismo bus 1-Wire.



En caso sea necesario es posible conectar más sensores al mismo pin de datos.

La diferencia acá, es que al tratarse de un bus, necesitamos averiguar la dirección de cada sensor para poder identificarlo.

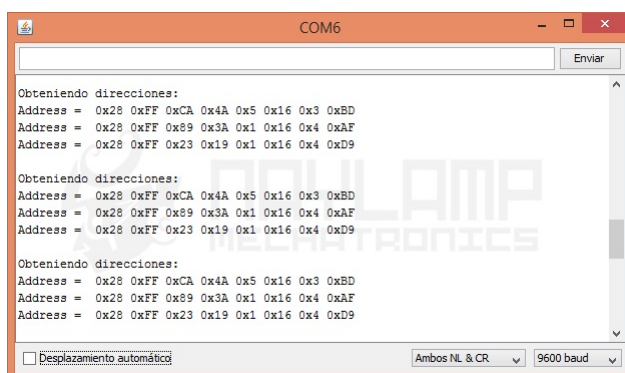
El siguiente sketch solo se utiliza para obtener la dirección de los dispositivos conectados en el bus 1-wire:

```
#include <OneWire.h>

OneWire ourWire(2);          //Se establece el pin 2 como bus OneWire

void setup(void) {
  Serial.begin(9600);
}

void loop(void) {
  byte addr[8];
  Serial.println("Obteniendo direcciones:");
  while (ourWire.search(addr))
  {
    Serial.print("Address = ");
    for(int i = 0; i < 8; i++) {
      Serial.print(" 0x");
      Serial.print(addr[i], HEX);
    }
    Serial.println();
  }
  Serial.println();
  ourWire.reset_search();
  delay(2000);
}
```



El código anterior nos sirve para obtener las direcciones de los sensores, en nuestro caso hemos obtenido las direcciones de los tres sensores que tenemos conectados, pero pueden ejecutar el código anterior para cada sensor de forma individual para saber con exactitud la dirección de su sensor.

Una vez obtenida la lectura podemos identificar la lectura del sensor que deseemos medir.

Para realizar las lecturas usamos el siguiente sketch:

```
#include <OneWire.h>
#include <DallasTemperature.h>

OneWire ourWire(2); //Se establece el pin 2 como bus OneWire

DallasTemperature sensors(&ourWire); //Se declara una variable u objeto para nuestro sensor

DeviceAddress address1 = {0x28, 0xFF, 0xCA, 0x4A, 0x5, 0x16, 0x3, 0xBD};//dirección del sensor 1
DeviceAddress address2 = {0x28, 0xFF, 0xB9, 0x3A, 0x1, 0x16, 0x4, 0xAF};//dirección del sensor 2
DeviceAddress address3 = {0x28, 0xFF, 0x23, 0x19, 0x1, 0x16, 0x4, 0xD9};//dirección del sensor 3

void setup() {
  delay(1000);
  Serial.begin(9600);
  sensors.begin(); //Se inicia el sensor
}

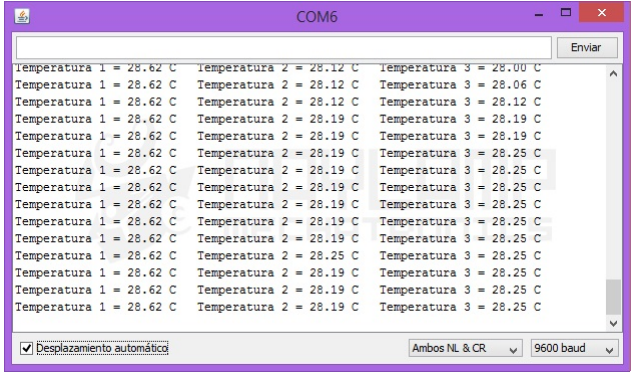
void loop() {

  sensors.requestTemperatures(); //envia el comando para obtener las temperaturas
  float temp1= sensors.getTempC(address1);//Se obtiene la temperatura en °C del sensor 1
  float temp2= sensors.getTempC(address2);//Se obtiene la temperatura en °C del sensor 2
  float temp3= sensors.getTempC(address3);//Se obtiene la temperatura en °C del sensor 3

  Serial.print("Temperatura 1 = ");
  Serial.print(temp1);
  Serial.print(" C");
  Serial.print(" Temperatura 2 = ");
  Serial.print(temp2);
  Serial.print(" C");
  Serial.print(" Temperatura 3 = ");
  Serial.print(temp3);
  Serial.println(" C");
  delay(100);
  delay(100);
}
```

Se debe tener en cuenta que se deben reemplazar la direcciones en el sketch con las direcciones correspondientes a los sensores que se tenga disponibles.

A continuación mostramos nuestros resultados:




Las temperaturas sensadas son similares puesto que los sensores se encontraban en el mismo ambiente:

Para cambiar la resolución del sensor a: 9, 10, 11 o 12 bits. solo se debe usar la función:

```
sensors.setResolution(Address, 9); // resolución de 9 bits
```

Normalmente la resolución se configura en el void setup() después de inicializar los sensores. A menor resolución el tiempo de lectura será menor.

PRODUCTOS RELACIONADOS




Agotado

Sensor De Temperatura Digital
DS18B20

SKU: 000016

S/ 11,00

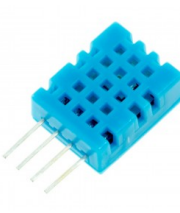


En stock

Arduboard Uno R3

SKU: 000008

S/ 65,00




En stock

Sensor De Temperatura Y
Humedad Relativa DHT11

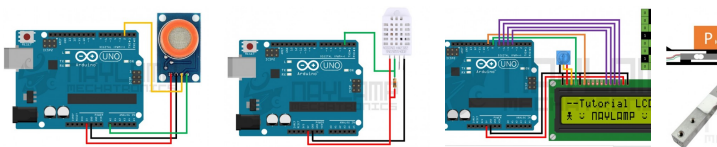
SKU: 000057

S/ 10,00


















Sen

ARTÍCULOS RELACIONADOS



57 COMENTARIOS

	Da**** *va 24/01/2023 Responder
	Hola! . Tienes la librería del sensor Ds18b20 para el micro LPC1769 (librería LPCopen)?
	Saludos desde Argentina!
	Ar*** 31/01/2022 Responder
	Hola, alguien sabe cuál MOSFET lleva, ya que en la hoja de datos no especifica nada... Gracias
	Ju** 11/01/2021 Responder
	Estimado, tengo 4 sensores con un resistwncia de 2k, y solo me funciona dejando el voltaje sin conexion, para mi es raro. Si conecto el voltaje a gnd arroja 0. Hice la prueba cn un sensor viejo q tenia y ese me funciona correctamente, dejando el voltaje en 5v.(no parasito) q puedo hacer
	Pe*** 26/12/2020 Responder
	Buenas noches. Como hago para poder cambiar un sensor sin tener que reprogramar de nuevo el microcontrolador con la nueva direccion del nuevo sensor ?.....hay algun codigo que detecte lea la nueva direccion y la reemplace por la antigua ?.....esto es debido a que si se daña un sensor al reemplazarlo se tendría que reprogramar de nuevo el microcontrolador. Agradezco la respuesta
	Ar*****ta 04/10/2020 Responder
	Buenas noches Tengo que conectar un Sensor DS18B20 a dos arduinos, realizamos un Temp Datalogger con el Shield V1.0, por alguna razón las librerías no deja disponibles las salidas digitales y hay que realizar tres tareas más en este Arduino, se pensó en otro Arduino para las estas tareas, el problema que conectar un solo Sensor a dos puertas estas entran en conflicto y o sabemos como hacerlo. las tareas son: Sensor la Temp, mostrar esta en un dispaly QAPASS serial, tener una salida por una valor de Temp peligroso, y una salida en función de una entrada que es otra protección. Agradecido por vuestro soporte.
	To*** *****ez 27/08/2020 Responder
	Hola, como andas? No se si seguirás respondiendo pero tenía una duda. Estoy pensando utilizar un arduino nano para sensar 16 ds18b20 y 2 dht11, así medir temperatura y humedad. Estos datos se comunican con un software en C# y se guardan en un texto. Hace no mucho estoy en esto del Arduijo, aunque en mi cabeza suene lógico, puede que haya alguna limitación, como puede ser el espacio del arduino. Que opinas sobre esto? Muchas gracias de antemano
	da**** 08/07/2020 Responder
	serviría para tomar la temperatura de una persona?
	Sa***** 10/06/2020 Responder
	Buenas al desconectar u conectar Arduino me manda lectura de 85° cual seria la solucion?
	Al** *****ta 22/03/2022 Responder
	Hola me paso lo mismo, pudiste resolverlo?
	Em*** *****do 09/06/2020 Responder
	Hola gracias por el tutorial , el software que usaste para el diseño de las imágenes también sirve para hacer simulaciones con el DS18B20? Cual es el nombre del programa?
	Ja**** 30/05/2020 Responder
	Hola, compre hace unos días el sensor, lo estoy usando con Arduino 1. Como lectura de temperatura me devuelve -127C, que significa sensor desconectado. He revisado el código y el conexionado varias veces y no encuentro error. Alguna sugerencia? Gracias y saludos
	ro***** 11/02/2020 Responder
	ahora lo interesante es poner la informacion en display LCD o display de 7 segmentos
	Mo***** *****es 28/01/2020 Responder
	Como es la conexion con gps. La instalacion sera en frigorificos.
	Mo**** *****el 17/01/2020 Responder
	Hola buenas tardes. esto lo puedo haer con sensores infrarrojos?
	Ed***** *****da 20/12/2019 Responder
	Hola. Muchísimas gracias por el excelente tutorial. ¿Como podría hacer para que una vez detectados, almacene sus direcciones en un array o algo al que recurrir para medirlos uno por uno cuando se quiere? Sin tener que escribirlos manualmente en un

nuevo sketch.
Perdona mi ignorancia, estoy empezando, he estado buscando como convertir la serie de bytes que entran y unirlos, y no hay manera, siempre me da algún error.



Nj*****ki 19/11/2019 Responder

Sos un capo. Muy util todo
Muchas gracias



Nj***** 14/11/2019 Responder

Quisiera saber el principio fisico del sensor, es decir, como lee la temperatura, variación en la resistencia o algo por ese estilo?



ma** 12/11/2019 Responder

Hola muy bueno el tutorial, estoy por hacer el proyecto con dos sensores pero a su vez quisiera agregar un modulo de wifi y otro modulo para tarjeta memoria, necesito este proyecto para controlar heladeras de dos frios y tener un reporte para mostrarle al cliente, si me pudieran ayudar se lo agradezco,.



Ar****ri 09/11/2019 Responder

hola excelente el tutorial. te queria preguntar, en la conexión con alimentación externa, el vpu al que se conecta la resistencia es la tensión del Arduino? o sea los 5v que suministra o es también la alimentación externa? muchas gracias!!



Ju***** 25/08/2019 Responder

El tutorial será bueno pero las mediciones son una mentira ,tengo ¡cuatro! sensores midiendo temperatura simultáneamente Y TODOS JUNTOS, y NO COINCIDEN EN LAS MEDICIONES entre el que mide la mayor temeperatura y el de la menor hay ¡¡¡ 4°C !!! de diferencia (estan configurados TODOS en 12 bits), ademas los programas de lectura que indican NO SIRVEN porque hay que indicar en el encabezamiento del programa LA DIRECCIÓN DE CADA SENSOR, para que indiquen la temperatura. En definitiva pésima ayuda para los que se inician .



Ju** *****ez 21/08/2019 Responder

¿cual es la distancia maxima del cableado entre el arduino y el sensor?



Jo*** 17/06/2019 Responder

¡Increible tutorial! Había estado buscando información sobre el sensor DS18B20 y la verdad está muy clara y completa la información.

Gracias por compartir :-D



Ju** 11/06/2019 Responder

Buenas, excelente tutorial!!
quisiera saber si es posible usar la informacion de un sensor y transmitirla a dos arduinos.
agradecería cualquier respuesta.



na*** 19/03/2019 Responder

Que buen tuto! te felicito gracias por compartir tu experiencia



Ma*****no 10/03/2019 Responder

Excelente! Todas las variantes, muy útil muchas gracias por compartir!



Jo** 01/02/2019 Responder

Gracias por el tutorial pero sabe como se podría realizar la medicion de temperatura con un sensor pt100 de dos hilos, es que tengo difcultades al momento de obtener los valores. Gracias



Ma*** 24/01/2019 Responder

Gracias por el tutorial.
Tengo una duda, realizando las conexiones ocmo indicas (de un solo sensor), me entrega valores muy superiores a los normales estoy a 25 grados y me marca 89 (estoy usando dos resistencias de 10K en paralelo, seria equiv a 5k).
Alguna idea de donde puede estar el problema?
Gracias!



ar****ri 19/11/2019 Responder

Lo pudiste solucionar? te cuento que hace poco instalando un sistema de control de temperaturas me paso que los sensores empezaron a calentar luego de un rato de estar funcionando, los alimente con los 5V de una fuente de PC ya que al prolongar los cables la alimentación del Arduino se caía.

Lo solucione cambiando la alimentación a 3.3V de esa misma fuente, ya hace casi 72hs que están (al momento son 3 sensores y serán 5 en breve) funcionando correctamente.

Ojala te sirva o le sirva a alguien mi experiencia.



Dj***an 15/01/2019 Responder

buenas noches tengo un problema con el codigo que estoy tratando de hacer tengo un congelador al que le quiero implentar un sensor DS18D20 de tal manera que cuando suba la temperatura prenda el motor del congelador y me presente a que temperatura esta en la pantalla pero el rele que acciona el motor del congelador se activa sin que la temperatura a la que quiero que se ensienda lo comande que debo hacer? te envio el codigo para que porfavor corrijas el error y me expliques que es lo que hice mal
#define ONE_WIRE_BUS 3

#include

```
#include

LiquidCrystal lcd(12,11,7,6,5,4);
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

int sensor = 0;//este pin lee la temperatura
int umbral = -2;//indica la temperatura en la que enciende el ventilador
const int pin = 9;

void setup()
{
  sensors.begin(); //Inicializa los sensores
  lcd.begin(16, 2); //Inicializa la librería
  lcd.clear();
  lcd.setCursor(0,0); //Coloca el cursor en el 0,0
  lcd.print("Diego Román"); //Imprime este mensaje
}

void loop()
{
  pinMode(pin, OUTPUT); //definir pin como salida
  lcd.setCursor(0,1); // Mueve el cursor al principio de la segunda línea
  lcd.print("T"); //Imprime este mensaje
  lcd.setCursor(3,1); // Mueve el cursor a la segunda línea, caracter 3
  sensors.requestTemperatures(); // Mide la temperatura
  lcd.print(sensors.getTempCByIndex(0),1); //Imprime el valor recogido
  lcd.print(" C"); //Imprime este mensaje

  digitalWrite(pin, HIGH); // poner el Pin en HIGH
  delay umbral = 28; // esperar 10 segundos
  digitalWrite(pin, LOW); // poner el Pin en LOW
  delay umbral 25; // esperar 90 segundos
}
```



Lu** 29/11/2018 Responder

Hola, en primer lugar felicitarlos por el tutorial que es excelente y ahora la pregunta, se puede leer lo que entrega cada sensor barriendo todos los sensores mediante un array? y abuso de su amabilidad con otra pregunta, es posible registrar esta cadena de datos en una SD?
Gracias!



An*** **ha 23/11/2018 Responder

HOLA BUENAS TARDES ES POSIBLE CONECTAR ESTE SENSOR CON UN GPS MODELO GV300 SALUDOS Y GRACIAS DE ANTE MANO



Ju** ***** 31/10/2018 Responder

Exelente artículo, me sirvió de mucha ayuda, una consulta habrá alguna forma de imprimir estos datos de temperatura en una LCD?



SA***** *C 22/08/2018 Responder

Que tal
Actualmente estoy utilizando este sensor(ds18b20) y no he encontrado como hacer un control con este sensor, por ejemplo: Cuando mi sensor registre una temperatura =>28 °C, que se active una salida(digital) de la placa arduino, y activar un relay para activar ventiladores.
Alguien que pueda ayudarme.



PA*** ***** **IN 27/07/2018 Responder

Hola buenas tardes jose y joan para que no marque -127 hay que colocar la resistencia pull-up de 4,7k .yo en mi caso lo corregi de esa manera.por ahi la resistencia rota o abierta y marca -127.saludos gracias



Jo** *****el 23/02/2018 Responder

Muy buenas.
Quería daros las gracias por tan interesantes tutoriales que compartís con toda la comunidad. Quería haceros una consulta:
Tengo en funcionamiento desde hace 2 años una aplicación con un arduino Mega y, entre otros, 10 sensores de temperatura de este tipo conectados por separado que controlan un sistema de calefacción, caldera de pellets y unos paneles termosolares. Los datos se guardan en una tarjeta SdD. Muy ocasionalmente (entre 1 y 5 veces por día) y coincidiendo cuando la caldera funciona, algún sensor (aleatoriamente) lee un dato de -127°C. Por software estos datos los elimino y no afectan al control, pero siempre he querido corregir este problema. He hecho algún análisis de estos datos y observo que tiene que ver con interferencias eléctricas, que no se cómo se transmiten. Me pueden informar de cuál puede ser la causa y cómo corregirlo?



Ts***** 10/12/2017 Responder

Simplemente gigante profesor master de master. Muchas gracias :D



be**** 01/12/2017 Responder

slm, merci pour ce tuto vous êtes bien travaille, mais pour quoi il affiche -127C° j'ai utilise EXP2:deux capteur



mj**** *****do 14/11/2017 Responder

cordial saludo sr naylamp es posible que hablemos en privado necesito comentarle algo con respecto casi a ese programa que monto cualquier cosa escribame por mi correo personal gracias en espera a su respuesta.
milton ortega colombia

th***** 30/10/2017 Responder

	Boa tarde, como eu conseguiria medir temperatura corporal com este sensor?
	An**** ***** ****jo 05/09/2017 Responder Gracias por el tutorial. Me ha sido de mucha ayuda
	Ne**** ***** 30/07/2017 Responder hola que tal es muy buena informacion, y abria una forma de almacenar datos en una sd para poder exportarlos a excel
	Na***** 22/08/2017 Responder Hola Nelson, en nuestro Tutorial SD explicamos cómo exportar datos de una SD a Excel.
	Ju***** 07/06/2017 Responder Hola Naylamp. Una consulta. quiero medir la temperatura de motores trifasico de 50 hp en la carcasa colocando tres de estos en diferentes puntos del motor. El sensor DS18B20 es inmune al ruido electromagnético para estos motores. gracias
	Na***** 12/04/2017 Responder Hola Facundo, si puedes usar los pines analógicos, pues estos se pueden usar también como pines digitales.
	fa***** 08/04/2017 Responder Hola una consulta se puede utilizar los pines analogicos?
	Na***** 12/04/2017 Responder Hola Facundo, si puedes usar los pines analógicos, pues estos se pueden usar también como pines digitales.
	Ed***** 18/01/2017 Responder Gracias por compartir tu conocimiento con los que iniciamos en arduino, he tenido algunos inconvenientes con las distancias de cables y me han recomendado un mosfet para dar mas potencia al onewire que opinas de esto y cual me recomiendas? y de nuevo muchísimas gracias
	Na***** 01/02/2017 Responder Hola Eduardo, con que longitud de cable estas trabajando? el MOSFET es necesario para cuando alimentas el DS18B20 de la línea onewire, si estas usando una alimentación externa (VCC) como en nuestro tutorial no es necesario el MOSFET. Puedes intentar bajar el valor de la resistencia pull-up, por ejemplo para una distancia 20 metros usa una resistencia de 2K. Saludos
	ju*** ***** 21/05/2019 Responder buenas noches, gracias por compartir tus conocimientos de arduino, usado el ejemplo 2, lo hice en mi protoboard, pero lo quiero aplicar a una cisterna con una distancia de cerca de 100 metros también sería viable el reducir el valor de la resistencia, también he estado intentando mezclar el programa con un lcd en vez de la pantalla de la computadora pero me está costando mucho
	jo** 26/11/2016 Responder Les recomiendo este software para visualizar los datos que se envían vía serial es muy bueno permite enviar hasta 9 datos que serán mostrados en tiempo real. https://www.youtube.com/watch?v=L1fVNvpPco&t=396s
	Na***** 07/01/2018 Responder Muchas gracias por el aporte, vamos a agregarlo ;)
	Ti** 29/10/2016 Responder Hola! muy bueno el tutorial, mi duda es, se puede conectar otro tipo de sensor en el bus oneWire, como por ejemplo sensor de PH o sensor de Turbiedad????
	Na***** 05/11/2016 Responder Hola Tina, Si se puede y se tiene que trabajar con las direcciones de cada dispositivo, como en el ejemplo 3
	Pa*** 23/09/2016 Responder El tutorial que hiciste, es excelente, no me quedo ninguna duda, realmente completo, mostraste todos los tipos de conexiones que necesitaba, te lo agradezco que hicieras el tutorial.
	ed***** 27/08/2016 Responder Excelente tutorial me ha ayudado demasiado, también felicitarlo por la página y el contenido de esta, una consulta, hasta cuántos sensores es recomendable colocar?
	Na***** 31/08/2016 Responder Hola Eduardo, gracias por el comentario, estaremos publicando más tutoriales. Con respecto a la cantidad de sensores depende de cómo los conectes. Si usas la conexión del ej.2 puedes conectar los sensores que desees mientras tengas pines suficientes, si conectas en un mismo bus como en el ej.3 puedes trabajar hasta más de 10. Puedes mezclar también las dos formas de conexión, dos o más buses con varios sensores en cada bus.

DEJAR UN COMENTARIO

Nombre (obligatorio)

Correo electrónico

☐

No soy un robot

reCAPTCHA
Privacidad - Términos

Publicar Comentario

BOLETIN

Regístrate en nuestro boletín para recibir las últimas ofertas y novedades.

☐ Acepto los términos y condiciones y la política de privacidad



Tu correo electrónico

SOPORTE

- Contactanos
- Mapa del sitio
- Envíos y Entregas
- Métodos de Pago
- Condiciones de uso

CATÁLOGO

- Promociones especiales
- Nuevos productos
- ¡Lo más vendido!

MI CUENTA

- Mi cuenta
- Mis ordenes
- Mis direcciones

INFORMACIÓN DE CONTACTO

- Urb. Ingeniería Mz-E Lt-31, Trujillo, Trujillo, La Libertad, Peru
- contacto@naylampmechatronics.com
- 997646858
- Horario de atención: Lun-Sab: 10am-7pm