



Prácticas de IoT (telémetro)

Contenido

1	Kit de prácticas (telémetro)	2
2	La tarjeta NodeMCU	2
3	Instalación del IDE de Arduino para NodeMCU	2
4	El telémetro ultrasónico US-100	3
4.1	Conexión entre telémetro y NodeMCU	3
4.2	Funcionamiento del telémetro.....	3
5	Ejemplo de programa para medir distancia enviando la medida por el puerto serie	4
5.1	Para probarlo:.....	4
6	Conexión a una plataforma de IoT	5
6.1	Registro en ThingSpeak.....	5
6.2	Crear un canal	5
6.3	Recopilar datos.....	5
6.4	Ejemplo de programa para el telémetro que publica a través de la API de ThingSpeak	6
7	Ejemplo de visualización de datos desde una App móvil	7

1 Kit de prácticas (telémetro)

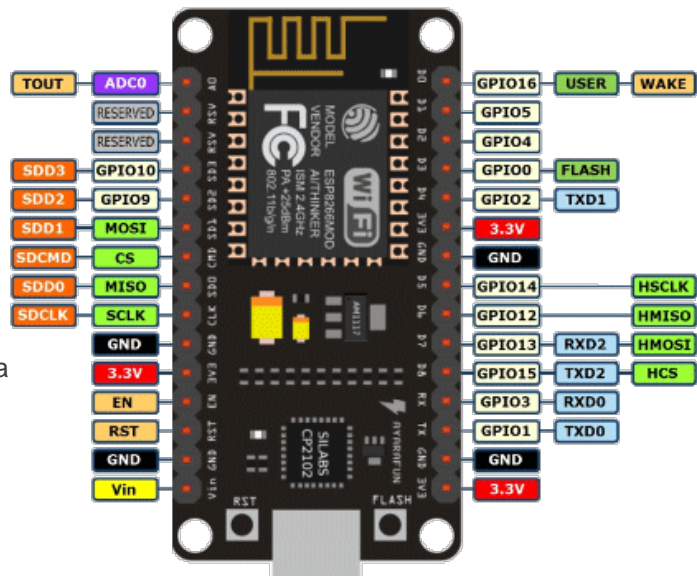
El kit incluye un NodeMCU, un módulo ultrasónico de medida de distancia, un cable USB-uUSB, una placa de pruebas y cables de conexión.

2 La tarjeta NodeMCU

Disponemos de una tarjeta NodeMCU de segunda generación (V1.0), basada en ESP8266 con chip ESP12E. En la figura puede ver un esquema de los pines de la misma. Puede encontrar más información en

<https://www.prometec.net/nodemcu-arduino-ide/>

NodeMCU se puede programar con el IDE de Arduino añadiendo el plugin para ESP8266.



3 Instalación del IDE de Arduino para NodeMCU

1. Instale el IDE de Arduino desde la página oficial:

<https://www.arduino.cc/en/Main/Software>

Más información en: <https://aprendiendoarduino.wordpress.com/2016/03/30/instalacion-del-ide-arduino/>

2. Instalar plugin de ESP8266:

- a. En Archivo>Preferencias, añadir

http://arduino.esp8266.com/stable/package_esp8266com_index.json como "Gestor de URLs Adicionales de Tarjetas".

- b. En Herramientas>Placa:...>Gestor de tarjetas buscar esp8266 e instalar.

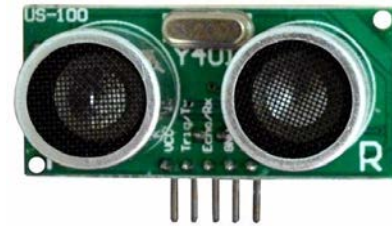
Más información en <https://www.prometec.net/esp8266-pluggin-arduino-ide/>

3. En Herramientas>Placa seleccione la placa "NodeMCU 1.0 (ESP-12 Module)"

4 El telémetro ultrasónico US-100

El US-100 es similar al más conocido HC-SR04, pero con alguna ventaja:

- Es más preciso (1mm frente a 3mm del HC-SR04).
- Admite tensión de alimentación más baja, se puede alimentar entre 2.4V y 5V.
- Además del método de medida por duración de pulso similar al del HC-SR04, tiene un modo de comunicación serie asíncrona que permite medir también la temperatura. Para usarlo como un HC-SR04 hay que quitar el jumper trasero.



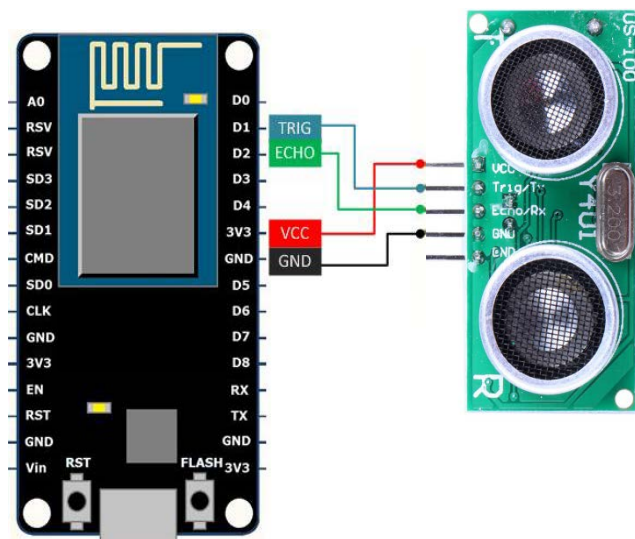
Más información en:

<https://minibots.wordpress.com/2013/10/14/comparacion-de-los-sensores-de-ultrasonidos-us-100-y-hc-sr04/>

<https://minibots.wordpress.com/2014/10/12/control-del-sensor-de-ultrasonidos-us-100-en-modo-serie/>

4.1 Conexión entre telémetro y NodeMCU

En la figura se representa la conexión que debe realizarse entre el telémetro el NodeMCU mediante 4 hilos. El telémetro se alimenta con los 3,3V que provee el NodeMCU. Las dos señales, TRIG es la salida del NodeMCU a través de la cual enviamos un pulso de 10uS para iniciar la medida, y ECHO es la entrada por la que se recibe el pulso de respuesta del telémetro cuya duración se corresponde con el tiempo de ida y vuelta de la señal de ultrasonido.



4.2 Funcionamiento del telémetro

Puede encontrar información de método de medida en diversos enlaces:

<https://www.luisllamas.es/medir-distancia-con-arduino-y-sensor-de-ultrasonidos-hc-sr04/>

<https://create.arduino.cc/projecthub/josemanu/medir-distancias-con-hc-sr04-63f81e>

<https://www.prometec.net/sensor-distancia/>

5 Ejemplo de programa para medir distancia enviando la medida por el puerto serie

```
// Pines del sensor de ultrasonidos
#define trigPin 5
#define echoPin 4
void setup() {
  Serial.begin(9600);
  pinMode ( LED_BUILTIN, OUTPUT );
  pinMode ( trigPin, OUTPUT );
  pinMode(echoPin, INPUT);
  digitalWrite(trigPin, LOW);
}
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);
  delay(500);
  digitalWrite(LED_BUILTIN, LOW);
  delay(500);
  // Medida de la distancia
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  float distancia = 0.0172*pulseIn(echoPin, HIGH);
  // Visualización por el puerto serie
  Serial.print("Distancia: ");
  Serial.println(distancia);
}
```

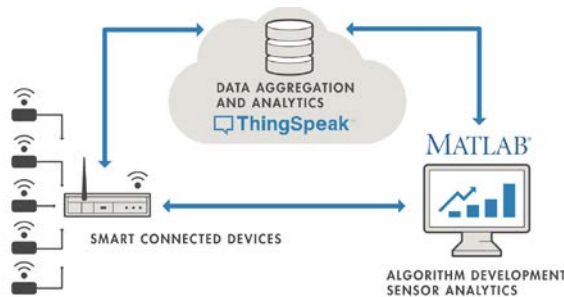
5.1 Para probarlo:

1. Copie el código en un nuevo programa para arduino.
2. Conecte el NodeMCU con el cable USB-uUSB
3. En *Herramientas* seleccione el *Puerto* de comunicación COMn correspondiente
4. Haga click en *Subir*.
5. Desde *Herramientas* abra una ventana de *Monitor Serie*. Ajuste la velocidad de comunicación (9600 baudios)

6 Conexión a una plataforma de IoT

Como primer ejemplo vamos a conectar el telémetro a ThingSpeak.

ThingSpeak es una plataforma que permite recoger, almacenar, analizar, visualizar y actuar sobre los datos de sensores o actuadores basados en módulos microcontroladores tales como Arduino®, Frambuesa Pi™, BEAGLEBONE Black y otro hardware.



6.1 Registro en ThingSpeak

Regístrese en <https://thingspeak.com>. Si ya está registrado en <https://es.mathworks.com> (Matlab) identifíquese en thingspeak con su “usuario” y “contraseña” de mathworks.

6.2 Crear un canal

El elemento principal de la actividad de ThingSpeak es el *canal (channel)*, que contiene campos de datos (*data fields*) y otra información.

Cree un canal con dos campos de datos para el valor de RSSI de la WiFi y para la distancia medida por el telémetro siguiendo las instrucciones de la ayuda:

<https://es.mathworks.com/help/thingspeak/collect-data-in-a-new-channel.html>

6.3 Recopilar datos

En la pestaña “API Key” del canal creado puede ver las claves que necesitará usar para poder escribir (publicar) y leer (suscribir) los datos con la API de ThingSpeak. En “Account > My Profile” puede generar claves para MQTT.

6.4 Ejemplo de programa para el telémetro que publica a través de la API de ThingSpeak

1. Con el gestor de librerías del IDE de Arduino instale la librería de ThingSpeak:
Programa > Incluir Librería > Gestionar Librerías
2. Copie el código en un nuevo programa de arduino
3. Cambie los valores del nombre y clave de la red WiFi por los correspondientes al punto de acceso WiFi que vaya a usar para la conexión a internet, y los valores del número de canal y clave de escritura de ThingSpeak por los del canal que haya creado.
4. Compile y suba el programa al NodeMCU
5. Visualice los datos desde la web de ThingSpeak

```
#include "ThingSpeak.h"
#include <ESP8266WiFi.h>
// Pines del sensor de ultrasonidos
#define trigPin 5
#define echoPin 4
// Punto de acceso wifi al que se conecta el dispositivo
char ssid[] = "YOUR_SSID"; // your network SSID Name
char pass[] = "YOUR_PASS"; // your network password
// Parámetros del cloud
int status = WL_IDLE_STATUS;
WiFiClient cliente;
unsigned long myChannelNumber = 123456; // your ThingSpeak Channel Number
const char * myWriteAPIKey = "0123456789ABCDEF"; // your ThingSpeak Write API Key
void setup() {
  Serial.begin(9600);
  pinMode ( LED_BUILTIN, OUTPUT );
  pinMode ( trigPin, OUTPUT );
  pinMode(echoPin, INPUT);
  digitalWrite(trigPin, LOW);
//Conexión a la WiFi y al cloud
  WiFi.begin(ssid, pass);
  ThingSpeak.begin(cliente);
  delay(500);
}
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);
  delay(500);
  digitalWrite(LED_BUILTIN, LOW);
// Medida de la distancia
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  float distancia = 0.0172*pulseIn(echoPin, HIGH);
// Visualización por el puerto serie
  Serial.print("Distancia: ");
  Serial.println(distancia);
//Medida del rssi de la wifi
  float rssi = WiFi.RSSI();
```

```
//Contrucción del mensaje y envío al cloud
ThingSpeak.setField(1, rssi);
ThingSpeak.setField(2, distancia);
ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
delay(19500); //cada 20 segundos
}
```

7 Ejemplo de visualización de datos desde una App móvil

Instale en su smartphone la aplicación “IoT ThingSpeak Monitor Widget”. Coloque el correspondiente *widget*, introduzca número de canal y clave de lectura y pruébelo.



8 Ejercicios

- ❖ Conecte el telémetro a ThingSpeak usando MQTT (buscar información en ThingSpeak). Un ejemplo de *uplink* a ThingSpeak con MQTT puede encontrarlo en: <https://www.instructables.com/id/Uploading-Data-to-ThingSpeak-With-MQTT/>
- ❖ Conecte el telémetro a otras plataformas IoT (por ejemplo a la plataforma demo de *hivemq* mediante MQTT).
- ❖ Use otras App smartphone para visualizar los datos del telémetro (por ejemplo “IoT MQTT Panel”, “IoT MQTT Dashboard”, “MQTT Dash”, etc).