



Manual de uso: Termistor Wifi TCP

19/12/2018

Fecha	19/12/2018
Autor	R.Mir
Versión	2.0
Revisado por	

## **TABLA DE CONTENIDO**

1. Introducción .....	2
2. Requisitos .....	2
2.1. Hardware .....	2
2.2. Software .....	2
3. Conexiones .....	3
4. Ejemplo de uso .....	3
4.1. Objetivo .....	3
4.2. Notas previas .....	3
4.1. Procedimiento .....	3
4.2. Resultados .....	6
4.3. Procedimiento de reseteo .....	6
4.4. Configuración avanzada .....	6
5. Proceso de ensamblado .....	8
5.1. Material necesario .....	8
5.2. Conexión de la Resistencia .....	8
5.3. Montaje .....	8
6. Disclaimer .....	9

## **1. INTRODUCCIÓN**

El dispositivo se conecta automáticamente a la red WLAN. Envía por TCP una trama con la temperatura y el ID de dispositivo. El intervalo de tiempo en el que se transmiten los datos es configurable. También es configurable un offset para la calibración de las mediciones.

## **2. REQUISITOS**

### **2.1. HARDWARE**

- Dispositivo Termistor Wifi TCP
- Cargador USB (opcional, para cargar la batería)
- Red WLAN
- PC

### **2.2. SOFTWARE**

- Navegador de internet

- Arduino IDE (opcional, si se desean ver tramas de debug).

### 3. CONEXIONES

- Conectar el cable macho USB a un transformador si se desea cargar la batería



Ilustración 1 - Alimentador externo USB

### 4. EJEMPLO DE USO

#### 4.1. OBJETIVO

Realizar la configuración previa. Conectar el dispositivo a nuestra red Wlan. Recibir los valores de temperatura medidos en nuestro servidor.

#### 4.2. NOTAS PREVIAS

**Nota 1:** Durante todo el procedimiento se van a mostrar tramas de debug. Dicha información no es necesaria para utilizar este dispositivo, pero si que es recomendable para entender su funcionamiento. Para poder visualizar estas tramas, es necesario conectar la PCB a un PC por medio del puerto USB, abrir el IDE de Arduino, seleccionar el puerto correspondiente y presionar Ctrl+Mayus+M.

**Nota 2:** El dispositivo tiene 2 modos de funcionamiento.

- Modo AP: El dispositivo crea una red Wlan a la que podemos conectarnos. Este modo sirve para realizar las configuraciones previas.
- Modo normal: El dispositivo se conecta a una red Wlan disponible y transmite los datos obtenidos a un servidor mediante TCP.

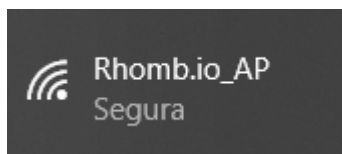
#### 4.1. PROCEDIMIENTO

Encender el dispositivo. Cuando el dispositivo se enciende intenta conectarse automáticamente a una red Wifi existente y previamente configurada. En caso de que la red no esté accesible o que no se haya configurado todavía, el dispositivo arranca en modo punto de acceso (AP).

```
*WM: AutoConnect
*WM: Connecting as wifi client...
*WM: Using last saved values, should be faster
*WM: Connection result:
*WM: 0
*WM:
*WM: Configuring access point...
*WM: Rhomb.io_AP
*WM: rhombio2018
*WM: AP IP address:
*WM: 192.168.4.1
*WM: HTTP server started
```

*Ilustración 2 - Proceso de arranque en modo AP*

Desde un PC, podemos ver que existe una nueva conexión Wifi disponible. Nos conectamos a ella.



*Ilustración 3 - Red Wifi a la que deberemos conectarnos para configurar el dispositivo*

- SSID: Rhomb.io\_AP

- Password: rhombio2018

Una vez conectados, abrimos un navegador y accedemos a la IP:

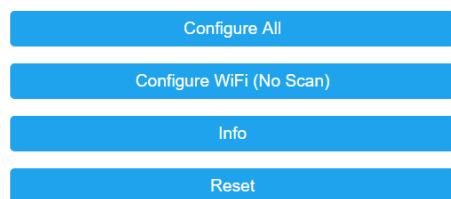
192.168.4.1

Deberá abrirse una página web con el portal de configuración.

#### **Rhomb.io device configuration portal**

### **Rhomb.io\_AP**

#### **WiFiManager**



*Ilustración 4 - Portal de configuración, página inicial*

Presionamos el botón "Configure All".

*Nota: Si en este punto la web se queda cargando indefinidamente, refrescamos la página y volvemos a presionar en el botón "configure all".*

Se deben mostrar las redes wifi disponibles, junto a la calidad de la señal. Seleccionamos la adecuada. El campo SSID se rellenará automáticamente con el nombre de la red.

### Rhomb.io device configuration portal

<a href="#">imasd</a>	🔒 100%
<a href="#">imasd_invitado</a>	🔒 100%
<a href="#">MOVISTAR_3DBC</a>	🔒 20%
<a href="#">SInvitados</a>	🔒 18%
<input type="text" value="SSID"/>	
<input type="text" value="password"/>	
<input type="text" value="intervalo tiempo"/>	
<input type="text" value="offset temperatura"/>	
<input type="button" value="save"/>	
<a href="#">Scan</a>	

Ilustración 5 - Parámetros a configurar

Introducir el password de la red Wifi

Introducir el intervalo de tiempo con los segundos que debe haber entre cada envío de los datos. Por ejemplo 30.

Introducir el offset de temperatura. Se pueden introducir tanto valores decimales, como negativos. Se recomienda un valor de 6.5. Esta opción sirve para que el dispositivo se pueda calibrar contrastando sus valores con los de un termómetro de referencia, sumando o restando este valor a la medida obtenida.

Presionar el botón “save”.

No será necesario volver a realizar esta configuración, aunque el dispositivo se reinicie o se quede sin alimentación. Quedará almacenada en la memoria interna del dispositivo (SPIFFS).

### Rhomb.io device configuration portal

Parameters Saved  
Trying to connect Rhomb.io device to network.  
If it fails reconnect to AP to try again

Ilustración 6 - Parámetros guardados correctamente en el dispositivo

En este momento el dispositivo se conecta a la red Wifi y comienza a enviar tramas al servidor. Cada nueva trama se envía cuando transcurre el intervalo de tiempo configurado.

```
Actual configuration: interval(sec.)/offset(°C): 30/6.5
Actual temperature: 21.25 °C
Connecting to IoT Server 45.76.37.219
TCP CMD: 837289382|s|temp:21.25
Values Sended OK! Closing connection
Waiting period...
```

Ilustración 7 - Información del envío

A través del puerto serie se muestra la siguiente información:

- Actual configuration: Configuración actual del dispositivo. Intervalo en segundos, offset en grados Celsius.

- Actual temperatura: Temperatura de la medición actual
- Connected to IoT Server: Dirección IP del servidor destino
- TCP CMD: Trama que se envía por TCP al servidor

## 4.2. RESULTADOS

Debemos contar con un servidor TCP con el que conectar. Dicho servidor será el encargado de recibir y procesar los datos que se le envíen desde el dispositivo.

Los datos se enviarán con el siguiente formato de trama:

837289382|s|temp:21.25

- La parte señalada en rojo: Identificador del dispositivo
- La parte señalada en verde: Flag “sensing” indica que la información procede de sensores
- La parte señalada en azul: Medición actual de temperatura

## 4.3. PROCEDIMIENTO DE RESETEO



El dispositivo cuenta con un jumper dispuesto para esta operación.

Ilustración 8 - Jumper

Es posible que queramos volver a acceder al portal de configuración en algún momento. Bien para modificar las credenciales de la Wlan, bien para modificar el intervalo de tiempo o el offset de calibración. Para ello necesitaremos poner el dispositivo en modo AP.

- El modo AP se activa automáticamente si el dispositivo no puede realizar satisfactoriamente la conexión con la red.
- El modo AP se puede activar bajo demanda de la siguiente forma. Tenemos para ello un jumper.
  - Poner el jumper
  - Reiniciar el dispositivo
  - Quitar el jumper

Tras un reinicio con el jumper puesto, el dispositivo nos avisa de que debemos quitarlo para que no se resetee en futuros reinicios. También nos avisa de la activación del modo AP.

```
Reseteando parametros
*WM: settings invalidated
*WM: THIS MAY CAUSE AP NOT TO START UP PROPERLY.
Reiniciando en modo AP
```

Ilustración 9 - Resetear los parámetros

## 4.4. CONFIGURACIÓN AVANZADA

Estos pasos van más allá del uso básico del dispositivo. No es necesario modificarlos por el cliente final ya que previsiblemente deben estar previamente configurados. Será Tecnofingers S.L.

necesario reflashear el dispositivo mediante el IDE de Arduino. Será necesario seleccionar la PCB "Generic ESP8266 module" tras haber instalado las dependencias necesarias para ESP8266. Será necesario contar con las librerías apropiadas.

#### ***Establecer valores por defecto***

Tras un reseteo via jumper, los valores de offset e intervalo se conservan.

Si se desea que tras un reseteo via jumper, los valores de offset e intervalo se restablezcan a unos valores por defecto, en lugar de conservar los valores previos, asignamos un valor a las siguientes dos variables:

```
40 //Here you can define default values, if you want.
41 char intervalo_tiempo[40];
42 char offset_temperatura[6];
```

Ej.: `char intervalo_tiempo[40] = 20;`

#### ***SSID y password para el modo AP (Portal de configuración)***

Modificar el SSID y el password cambiando los textos que aparecen entre comillas

```
152 //Config the AP name and password to access configuration portal
153 if (!wifiManager.autoConnect("Rhomb.io_AP", "rhombio2018")) {
```

#### ***Reset configuration pin***

El reset pin definido para resetear la configuración y activar el modo AP es el 4, que equivale a la IO1 en la PCB Deimos.

```
44 //If this pin is LOW during boot process, configurations are reset and the device will reboot in AP mode. IO1 at Deimos Board.
45 int rpin = 4;
```

#### ***Reseteo de la memoria interna SPIFFS***

En caso de querer formatear la memoria interna, y así borrar cualquier configuración previa, descomentar, esta línea, flashear y reiniciar. Tras ello, volver a comentar y reflashear para que no se formatee continuamente en cada reinicio.

```
66 //Uncomment if you want to format SPI Filesystem
67 // SPIFFS.format();
```

#### ***Establecer una IP Estática***

Si deseamos que el dispositivo disponga de una IP estática en lugar de recibirla automáticamente por medio del DHCP de la Wlan, podemos establecerla descomentando la siguiente línea.

```
117 //Descomentar si se desea IP estática
118 // wifiManager.setSTAStaticIPConfig(IPAddress(10,0,1,99), IPAddress(10,0,1,1), IPAddress(255,255,255,0));
```

#### ***Calidad mínima de la señal Wifi***

Si se desea, el dispositivo puede descartar cualquier red Wlan que no disponga de una calidad de señal mínima.

```
138 //set minimum quality of signal so it ignores AP's under that quality
139 //defaults to 8%
140 //wifiManager.setMinimumSignalQuality();
```

#### ***Valor resistivo***

El termistor NTC de 10K requiere para su correcto funcionamiento de una resistencia con un valor resistivo conocido. El valor recomendable son 10K, sin embargo, se puede configurar el valor exacto para aumentar la precisión. Cambiar 10020 por el valor resistivo adecuado (ohmios).

```
195 Temp = log(((40960000/RawADC) - 10020)); //Rango analógico Mega = 1024 ESP32 = 4096 // 10020 es el valor de la resistencia
```

### Rango de la entrada analógica

Según el módulo master que se emplee, el rango de valores de una entrada analógica varía. Por ejemplo.

- Valores Duino Uno-Mega: 0 ~ 1024
- Valores ESP32-ESP8266: 0 ~ 4096

Podemos modificar dichos valores aquí, en función de que PCB utilizemos:

```
195 Temp = log(((4096000/RawADC) - 10020)); //Rango analógico Mega = 1024 ESP32 = 4096 // 10020 es el valor de la resistencia
```

### Cambiar dirección IP y puerto del servidor TCP de destino

Dichos datos, vienen preconfigurados, pero si se desea, se pueden modificar aquí

```
234 //IoT Server IP and Port
235 const uint16_t port = 4444;
236 const char * host = "45.76.37.219"; // ip or dns
```

### Cambiar identificador de dispositivo

Si se desea, se puede modificar el identificador del dispositivo que va asociado a cada trama. Modificar la parte numérica de la siguiente línea.

```
230 //Forming the TCPCmd
231 String StrUno = "837289382|s|temp:";
```

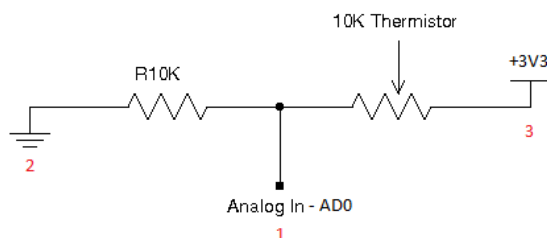
## 5. PROCESO DE ENSAMBLADO

Si deseamos montar el dispositivo desde cero vamos a necesitar el siguiente material.

### 5.1. MATERIAL NECESARIO

- 1x Termistor NTC 10K
- 1x Resistencia 10K
- 1x Jumper
- 1x Batería
- 1x Cable USB
- 1x Deimos
- 1x ESP8266 Rhomb.io Master module

### 5.2. CONEXIÓN DE LA RESISTENCIA



### 5.3. MONTAJE

Realizar las siguientes conexiones

TERMISTOR	DEIMOS
1 (Amarillo)	ADO



2 (Naranja)	GND
3 (Violeta)	3V3

Cable USB	DEIMOS
Positivo	Vbus
Negativo	GND

Jumper	DEIMOS
1	IO1
2	GND

Batería	DEIMOS
Positivo	B+
Negativo	B-

## 6. DISCLAIMER

Todos los derechos reservados. Ninguna parte de este documento puede ser fotocopiada, reproducida, almacenada en un sistema en la nube, o transmitido, en cualquier forma o por cualquier medio, ya sea electrónico, mecánico o de otro tipo, sin el previo consentimiento o permiso por escrito de Tecnofingers S.L.

No se garantiza la exactitud del contenido de la información contenida en esta publicación. En la medida en que lo permita la ley, no se aceptará ninguna responsabilidad (incluida la responsabilidad frente a cualquier persona por negligencia) por parte de Tecnofingers o cualquiera de sus subsidiarias o empleados por cualquier pérdida o daño directo o indirecto causado por omisiones de o de inexactitudes en este documento. Tecnofingers S.L. se reserva el derecho de cambiar los detalles de esta publicación sin previo aviso. Nombres de productos y empresas el presente documento puede ser una marca comercial de sus respectivos propietarios.