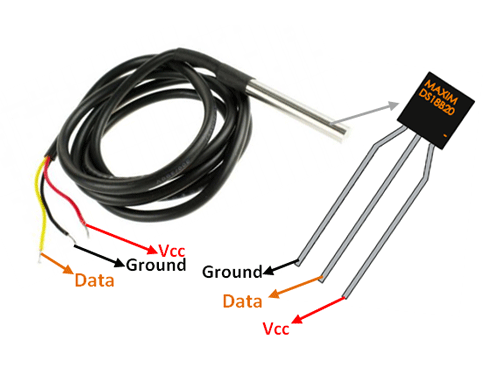
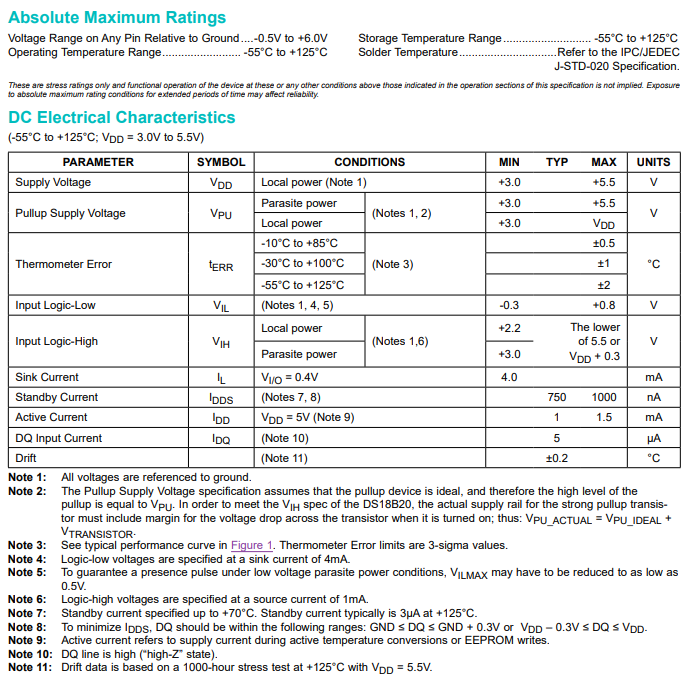
**One Wire (Sensor de to DS18B20)**

**¿Qué es one-wire?**

* **1-Wire** es un **protocolo de comunicaciones** en serie diseñado por **Dallas Semiconductor**. Está basado en un bus, **un maestro** y **varios esclavos** de una sola línea de datos en la que se alimentan. Por supuesto, necesita una referencia a tierra común a todos los dispositivos.
* La línea de datos/alimentación requiere una resistencia de **pull-up** conectada a la **alimentación** y que le proporciona ésta.
* Permite integrar **sensores de temperatura** y **llaves electrónicas** de una forma fácil, fiable y económica.

**DSB18B20**

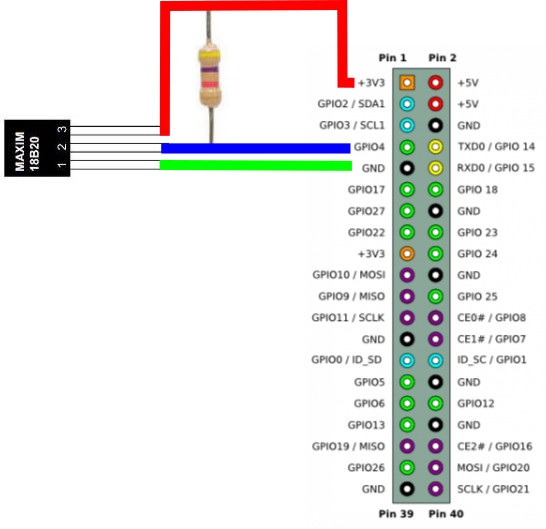
El DS18b20 está disponible en tres versiones: Transistor **individual**, en **módulo** y **sumergible**.

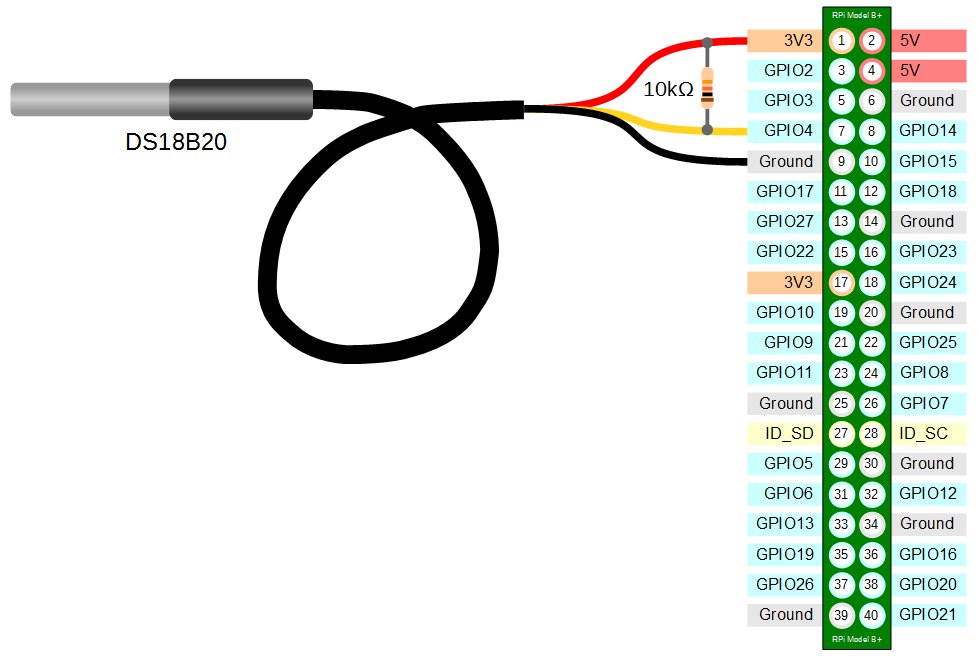
MÓDULO

SUMERGIBLE

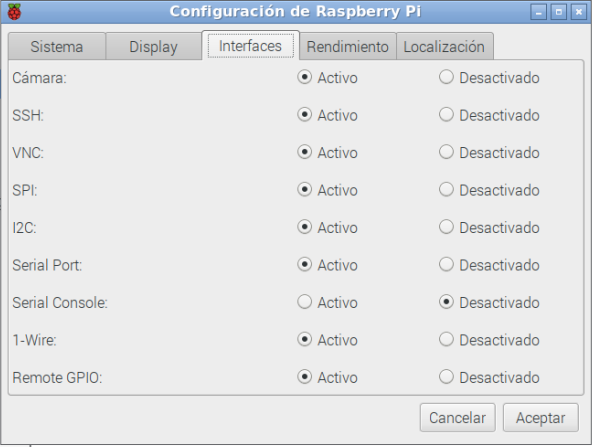
INDIVIDUAL

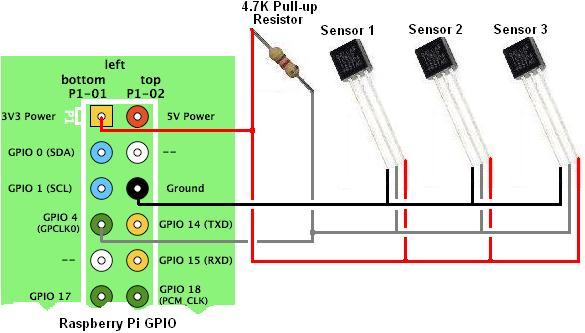
Excepto la versión módulo, es necesario que coloquemos una resistencia «**pull-up**» de 4K7 ohmios, y lo conectemos a la Raspberry Pi según el siguiente esquema:





Una vez conectado, **la lectura de la temperatura** es muy sencilla; **se hace** **leyendo un fichero**. Basta seguir los siguientes pasos:

1. **Habilitamos** el pin **GPIO4**, que es el que trae por defecto la Raspberry para trabajar con **One-Wire**.

Podemos colocar muchos sensores en paralelo, con una sola resistencia para todos ellos. Como cada sensor tiene un código diferente, aparecerán carpetas para cada uno de ellos, y no tendremos más que leer el que queramos.

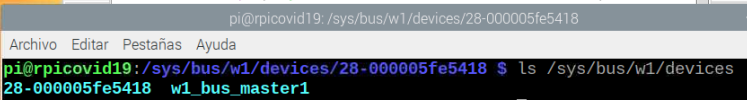
1. Ejecutamos los siguientes comandos:

**$ sudo modprobe w1-gpio**

**$ sudo modprobe w1-therm**

A continuación listamos el contenido de /sys/bus/w1/devices.

**$ ls /sys/bus/w1/devices**

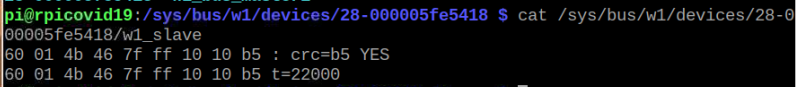


Aparecerán varios directorios, entre ellos uno (o varios, si tenemos varios termómetros conectados), con el nombre 28-XXXXX. 28-XXXXX es el número de serie de nuestro termómetro, diferente para cada transistor. Por ejemplo, este termómetro es **28-000005fe5418**.

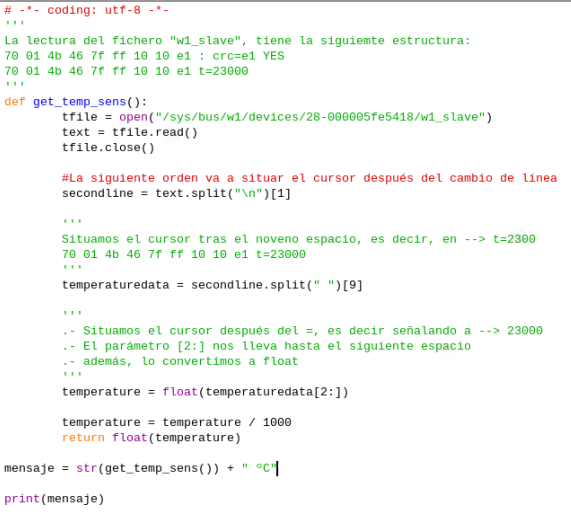
1. Entramos en ese directorio (reemplazando XXXXX por nuestro código), y leemos el fichero **w1\_slave**.

**$ cat /sys/bus/w1/devices/28-000005fe5418/w1\_slave**

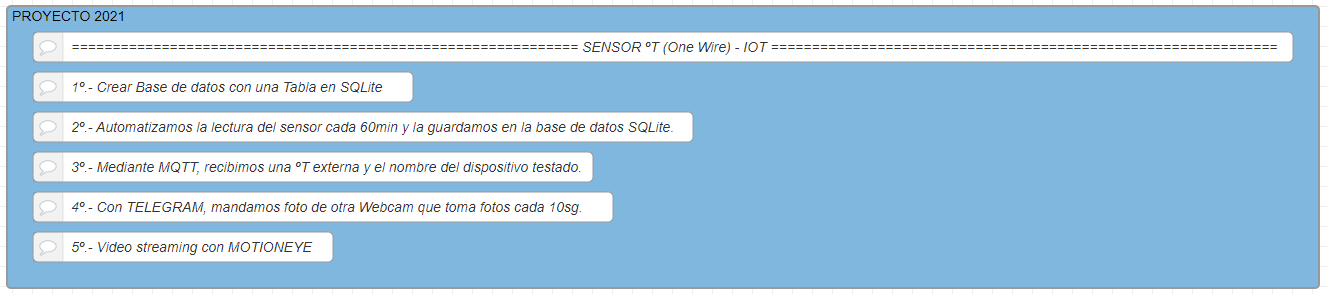
La salida será algo así:



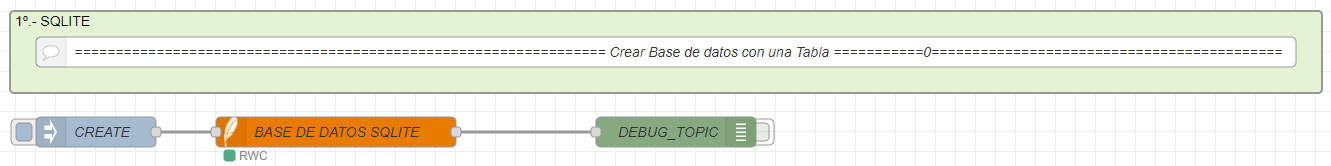
La temperatura viene expresada en moC. En la salida anterior dice t=2200, así que tendremos que dividir por 1000 para obtener la temperatura en oC, es decir: **22.000 oC**.

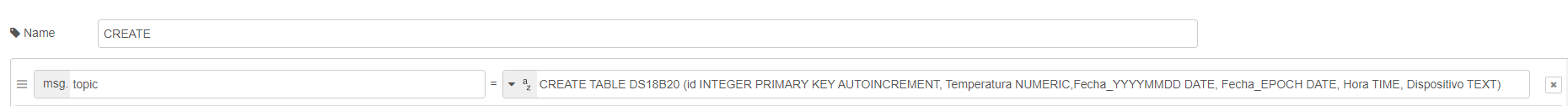
**DESDE PYTHON**

**PROCESO**



1. De forma manual, desde Node-Red, creamos una base de datos en SQLite con los siguientes campos:





1. Desde Node-Red, realizamos la lectura, cada hora, del sensor DS18B20 del siguiente modo:
   * Instalamos el siguiente nodo: “**node-red-contrib-sensor-ds18b20”**



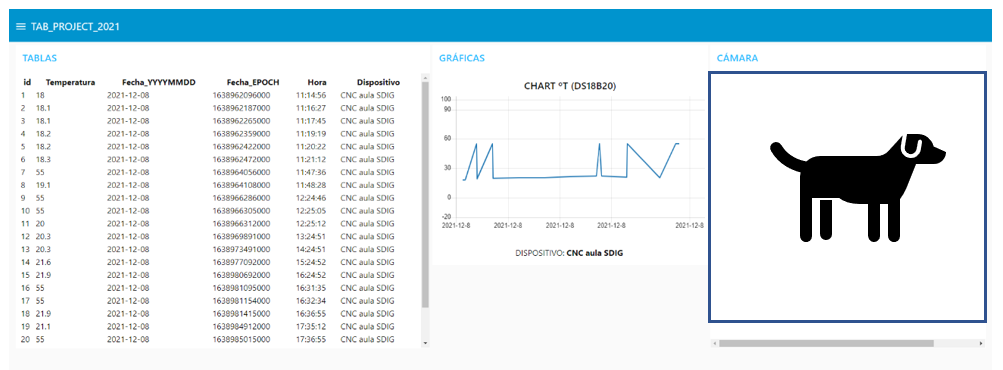
ID del sensor

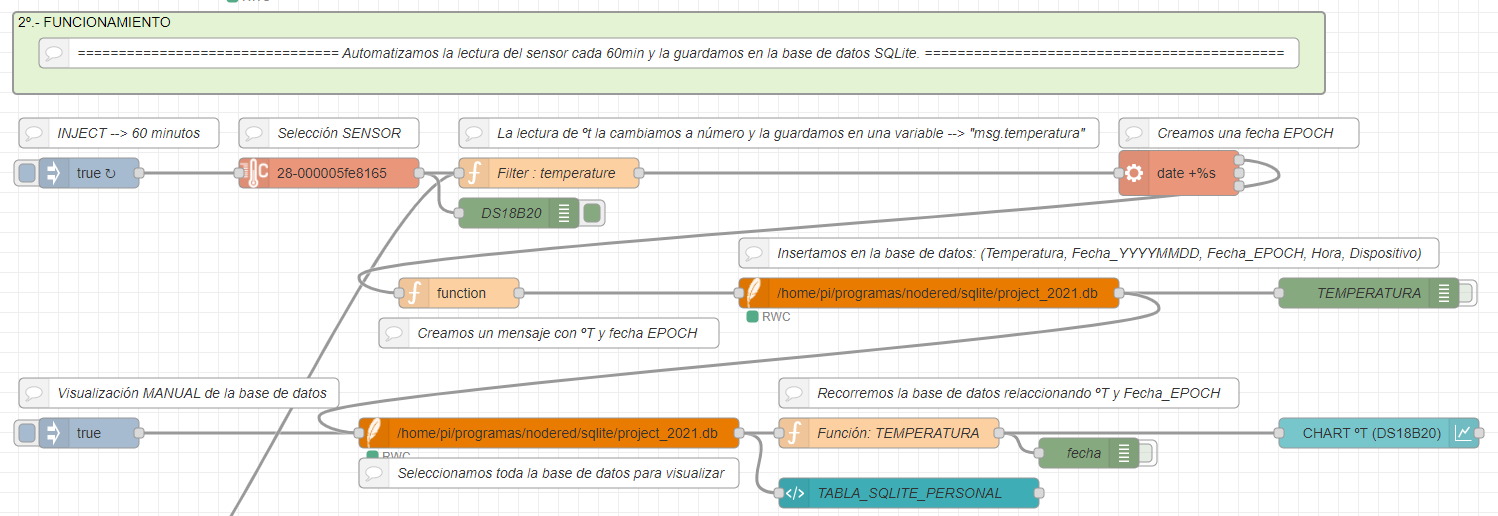
El nodo te devuelve la temperatura en grados centígrados en “**payload**”.

La lectura, será guardada en la base de datos de SQLite y visualizada en una gráfica en el Dashboard, tal y como muestra la figura.

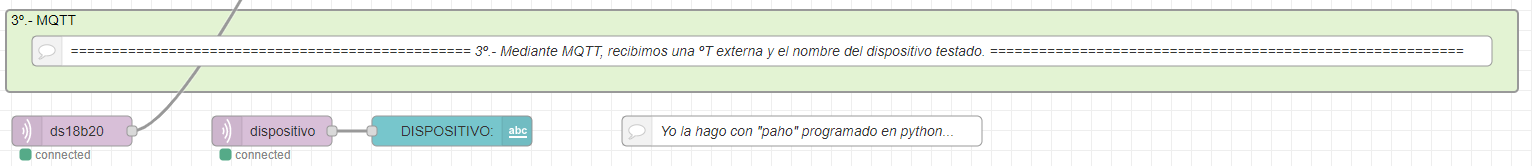
* + Eje Y: ot - Eje X: Fecha

En el dashboard, también se visualizará una tabla de la base de datos y el nombre del dispositivo testado.

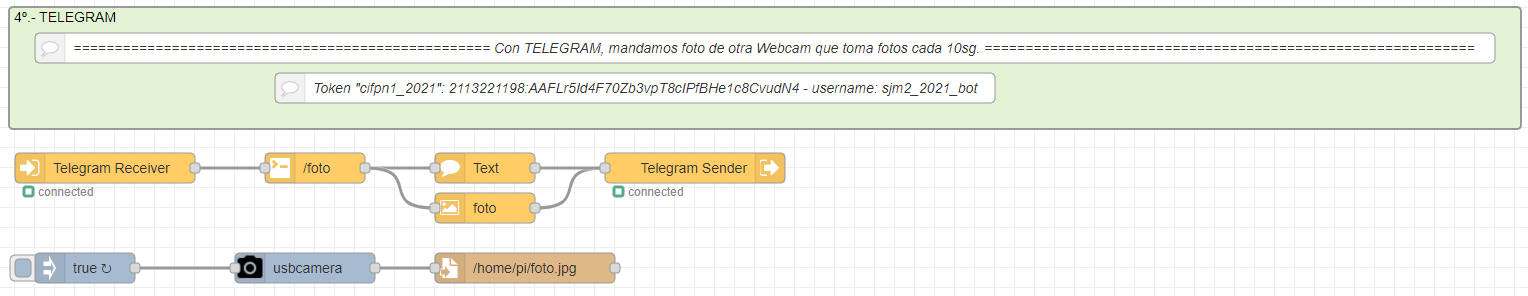




1. Mediante MQTT, recibimos una OT externa y el nombre del dispositivo testado.



1. Con TELEGRAM, mandamos foto de otra Webcam (o Raspicam) que toma fotos cada 10sg.



1. Video streaming con MOTIONEYE

