**Sonoff S20 hack.**

Sonoff S20 es un es un interruptor con control remoto por wifi.

Es la evolución del Sonoff Wi-Fi Smart link.

La diferencia es que no necesita montaje adicional. Viene con un enchufe macho y uno hembra, un pulsador y dos leds. Uno verde y uno azul.



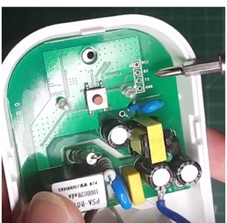
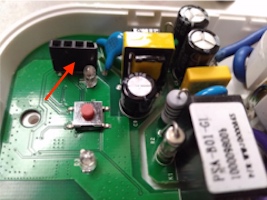
Lo primero que debemos hacer es quitar la pegatina de garantía que cubre el agujero con el único tornillo exterior del dispositivo.

En su interior podemos encontrar un pequeño transformador, un relé para controlar la salida de alto voltaje, los dos leds, el pulsador y un ESP8266 como corazón de la placa.



Identificamos el interruptor/pulsador, que tiene dos leds a sus lados, etiquetados como D1 (azul) y D3 (verde).

Al lado del led D1 tenemos 4 agujeros correspondientes a los pines de comunicación y alimentación.

La etiqueta de los pines no viene escrita en la placa en todas las versiones.

Tomaremos las etiquetas que podemos ver en las fotos.

Si ponemos el S20 en horizontal dejando los agujeros en la parte superior, tenemos de izquierda a derecha los pines de: VCC, RX, TX y GND.

Para poder utilizarlo necesitaremos soldar un “peine” de 4 pines o una tira de 4 pines hembra (como las fotos).

Para ello debemos soltar los dos tornillos que sujetan la placa. Así podremos soldarlo por la otra cara de la placa.



Una vez soldado, volvemos a montarla.

Necesitaremos un FTDI para convertir una salida USB en un puerto serie

Las conexiones serán las siguientes:

|  |  |
| --- | --- |
| **S20** | **FTDI** |
| VCC | 3.3V |
| RX | TX |
| TX | RX |
| GND | GND |

Es importante recordar que el pin VCC sólo acepta un máximo de 3.3V. Si lo conectamos a 5V podríamos quemarlo.

Para poder transmitir un programa necesitaremos ponerlo en modo flash. Para ello, reiniciamos manteniendo el botón pulsado.

Es importante borrar totalmente la memoria del dispositivo. Si no lo hacemos es posible que nuestros desarrollos no se comporten correctamente.

Para ello utilizaremos un script de Python. Por lo tanto necesitaremos tener instalado Python y las herramientas del esp (esptools).

Estas herramientas se instalan desde la línea de comandos:

python -m pip install esptool

Así se descargarán las librerías necesarias.

Sabiendo cuál es el puerto COM al que tenemos conectado el FTDI ejecutamos la siguiente línea:

d:\programas\python27\lib\site-packages\esptool.py --port COM12 erase\_flash

El path de dónde se encuentra la librería lo obtenemos de la instalación

El COM12 debe ser sustituido por el COM correspondiente.

Tarda algo menos de 5 segundos en borrar la memoria.

En el IDE de Arduino se debe configurar de la siguiente manera:

Placa: “Generic ESP8266 Module”

Flash Mode: “DOUT”

Flash Frequency: “40MHz”

Upload Using: “Serial”

CPU Frequency: “80 MHz”

Flash Size: “1M(64K SPIFFS)”

Reset Method: “ck”

Upload Speed: “115200”

Los GPIOs disponibles son los siguientes:

|  |  |
| --- | --- |
| **GPIO** | **Función** |
| 0 | Interruptor (pulsador) |
| 1 | TX |
| 3 | RX |
| 12 | Relé & led azul |
| 13 | Led verde |

El led azul está asociado al relé.

El relé y led azul están encendidos cuando el pin está en HIGH.

El led verde se enciende cuando está en LOW.

El interruptor está en LOW cuando se pulsa.