

Procedimiento para la actualización de firmware.

Sensor ams CCS811 medición de gases TVOC y CO₂eq.

Autor. Ernesto Gutiérrez @erguro1973

Basado en la documentación de:

<https://github.com/maarten-pennings/CCS811>.



Foto Amazon

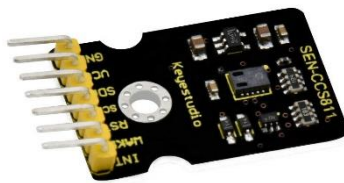


Foto Keyestudio



Foto Adafruit



Foto Sparkfun

DESCARGO DE RESPONSABILIDAD: Una actualización de firmware puede inutilizar para siempre su CCS811. No puedo ser responsable de que algo salga mal durante el proceso de actualización.

El riesgo es bajo, ya que solo actualiza el firmware de la aplicación, no el cargador de arranque (boot loader)

Objeto.

El presente documento es una guía desarrollada por la asociación de Makers Baleares, con la colaboración de Fablab Mallorca, para actualizar el Firmware de los sensores del fabricante ams (en la actualidad [ScioSense](https://www.sciosense.com)) modelo **CSS811** la información completa del sensor se puede encontrar en el siguiente enlace:

<https://www.sciosense.com/products/environmental-sensors/ccs801-gas-sensor/>

La librería para realizar la actualización de Firmware ha sido desarrollada por **Maarten-Pennings** y ha sido usada con éxito por el autor de esta guía, Ernesto Gutiérrez (@erguro1973) la información completa se puede encontrar en el siguiente enlace:

<https://github.com/maarten-pennings/CCS811>

Alcance.

Sensores y breakout boards basadas en CSS811 con versión de firmware 1.0. La actualización se ha probado con éxito en el siguiente hardware.

- NodeMCU (ESP8266)
- Arduino Pro Mini
- Arduino Nano ver [ADVERTENCIA] en descripción.
- ESP32.

Descripción.

El CSS811 es un sensor para la medición de calidad del aire en interiores, puede detectar una amplia gama de VOC (compuestos orgánicos volátiles), Alcoholes, aldehídos, cetonas, ácidos orgánicos, aminas, hidrocarburos alifáticos y aromáticos.

Al no poder medirlos individualmente lo hace de manera total, por eso los indica como TVOC, también realiza mediciones de CO₂eq (CO₂ Equivalente) mediante un algoritmo. Para más información visitar la web de [Sciosense](https://www.sciosense.com).

¡Importante!, este sensor necesita 48h funcionando continuamente para dar medidas fiables a esta fase se le conoce como "Burn in". Una vez superada la fase "Burn in" cada vez que se use necesitará 20 minutos aproximadamente (posiblemente menos) para hacer mediciones no erráticas.

[ADVERTENCIA]

Hay que tener en cuenta que el CCS811 requiere un voltaje de suministro de 1.8V - 3.6V. por lo que 3.3V es correcto, pero no se debe usar una placa con líneas de comunicación I2C a 5V. El Nano tiene un pin de "salida 3v3", pero es sólo un pin de alimentación, el micro impulsa las líneas I2C en 5V. Esto es un riesgo para el CCS811 y en ese caso debería usarse un conversor de niveles (level shifter). También hay que tener cuenta que el voltaje de suministro mínimo del CCS811 es 1.8V y no debe caer por debajo de este valor para un funcionamiento estable del sensor, si bien la mayoría de breakout boards ya incluyen un regulador de tensión.

La mayoría de los microcontroladores parecen tener incorporados pull-ups para I2C. En cualquier caso (si la breakout board no las lleva) se recomienda agregar resistencias de pull-ups de 10k tanto en SDA como en SCL.

Hay que tener en cuenta que a partir de la versión 2.0.0 el fabricante (ams) incluye esta nota de cambio "Se eliminó la funcionalidad NTC. El pin 8 no se mide y se deja sin uso". Por lo tanto, si su aplicación usa el termistor integrado, no actualice de 1.x.x a 2.x.x.

Procedimiento - Prerrequisitos.

Se considera que se tiene instalado el IDE de Arduino.

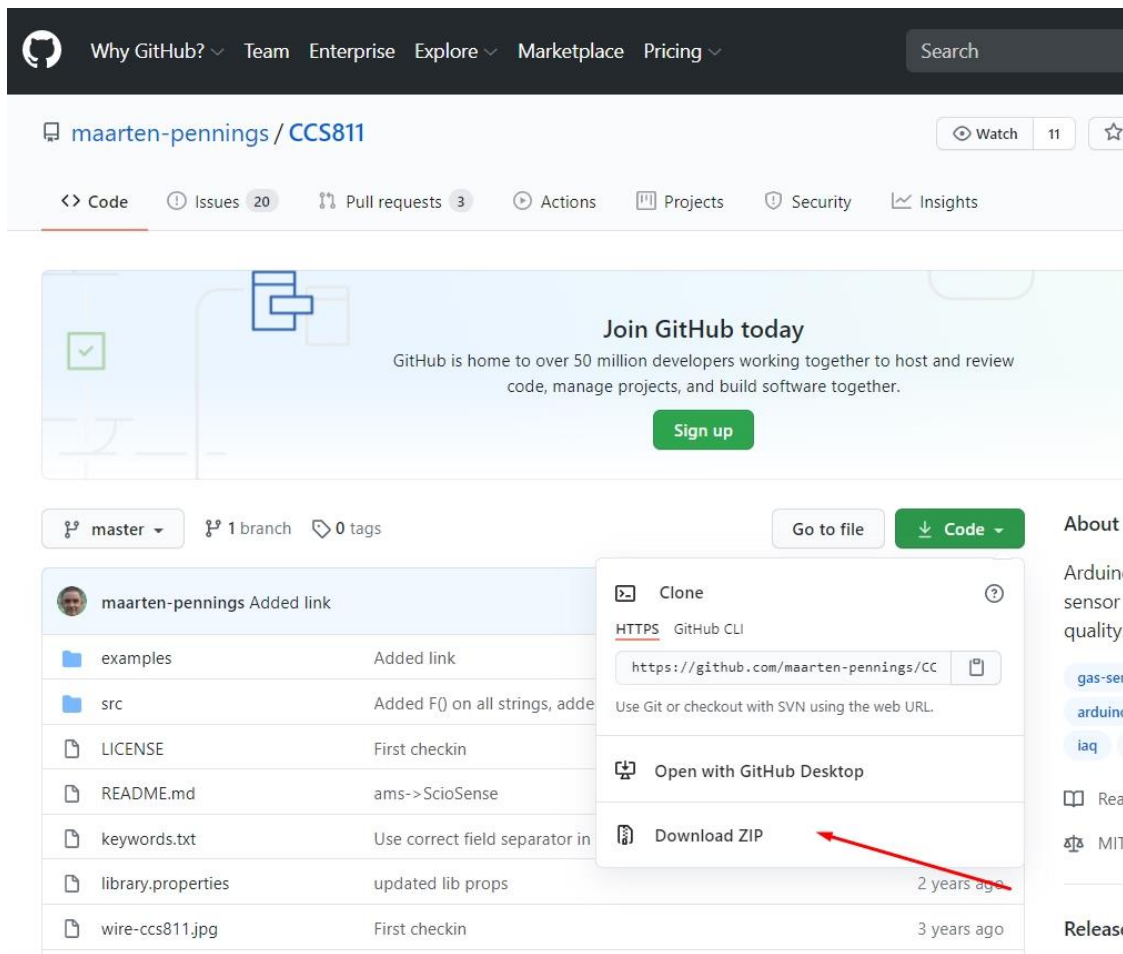
El directorio de la biblioteca Arduino se encuentra en su ubicación predeterminada. En mi caso C: \ Users \ erguro1973 \ Documentos \ Arduino \ libraries.

Instalación.

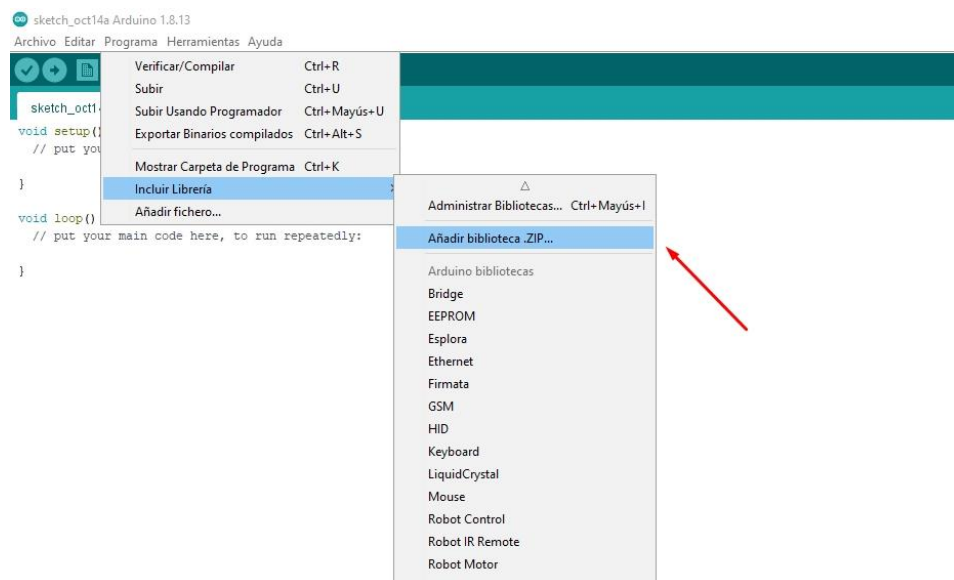
Visite la página del proyecto de la [librería Arduino CCS811](#).

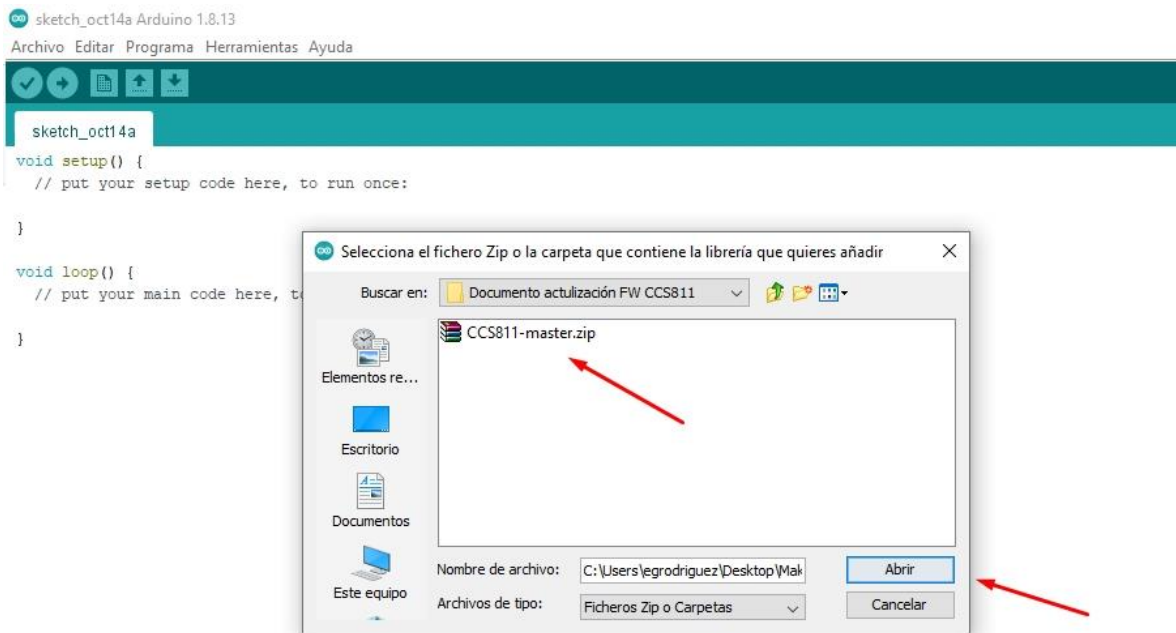
Haga clic en el botón verde Clonar o descargar en el lado derecho.

En la ventana emergente, elija Descargar ZIP.



En Arduino IDE, seleccione Programa> Incluir librería> Administrar bibliotecas> Añadir biblioteca ZIP... y buscas el archivo ZIP que acabas de descargar.



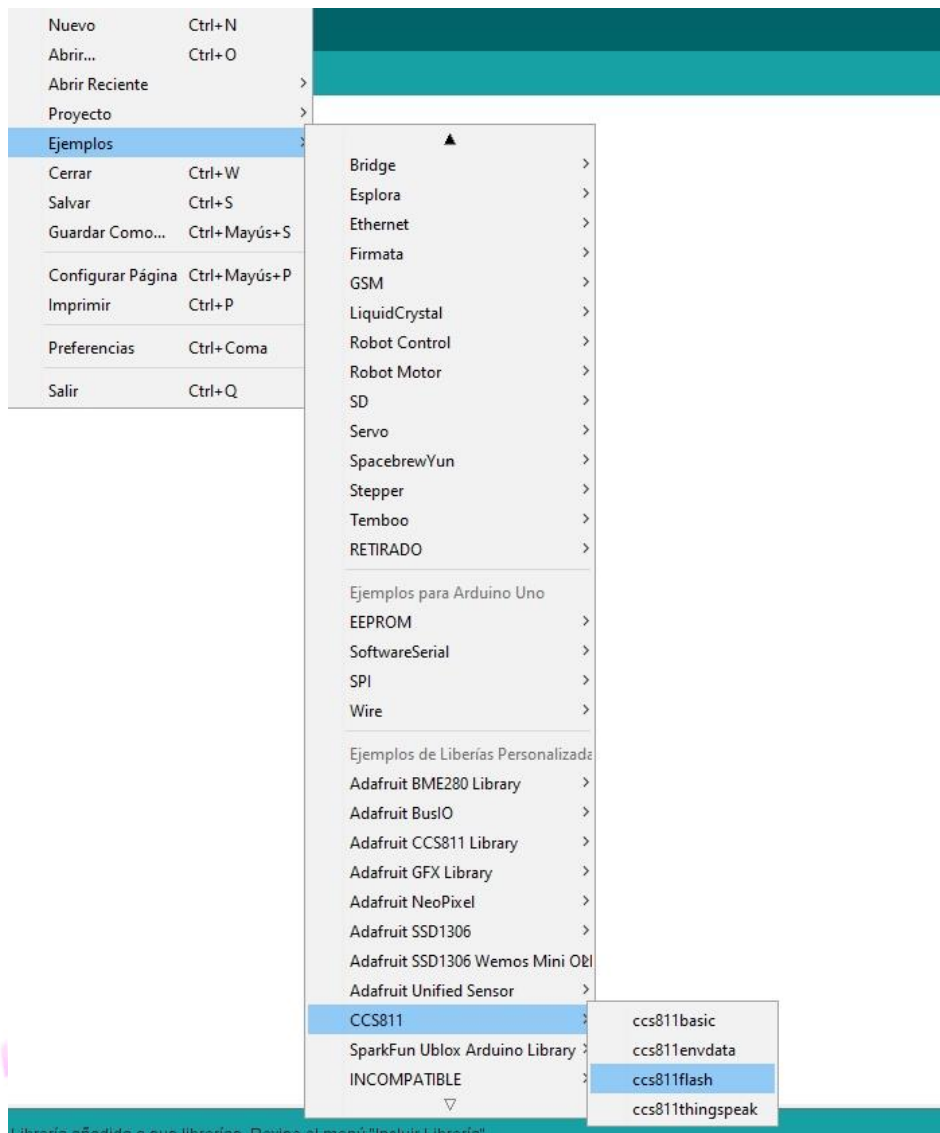


Cargar un ejemplo.

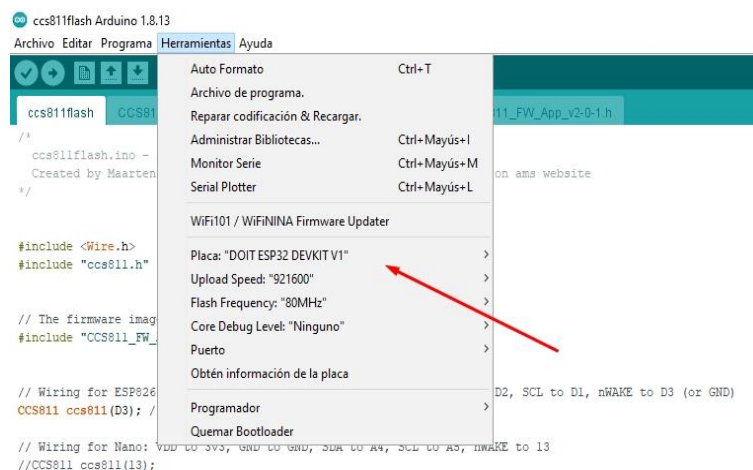
Para cargar un ejemplo de la librería (solo compilar, para grabarlo necesitamos cablearlo, y ese es el siguiente paso):

(Re) iniciar Arduino.

Abrir Archivo> Ejemplo> Ejemplos de librerías personalizadas> CCS811> ccs811flash.



Asegúrese de que en el menú Herramientas> esta seleccionada la placa correcta.



también bastaría con reemplazar D3 por 23 en la línea de código del ESP8266.

Ahora ya podemos compilar y subir el programa al ESP32, si todo ha ido bien iremos a Serial Monitor.

```
COM6

loop: ended ...
loop: ended ...
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:1044
load:0x40078000,len:8896
load:0x40080400,len:5816
entry 0x400806ac

Starting CCS811 flasher
setup: library      version: 10
setup: hardware     version: 12
setup: bootloader   version: 1000
setup: application  version: 1100
setup: comment-out this code line if you want to flash
loop: ended ...
```

Podremos ver la versión actual, en este caso pone 1100, entendemos que será V 1.1 como no hemos des-comentado la línea de código que evita que le actualicemos el Firmware.

¿Qué versión queremos actualizar?

Tenemos 3 opciones:

1. **CCS811_FW_App_V1-0-0.h** para devolverlo al firmware original
2. **CCS811_FW_App_V2-0-0.h** nueva versión si has hecho el procedimiento "Burn In" Tener el sensor trabajando más de 48h
3. **CCS811_FW_App_V2-0-1.h** nueva versión si **no** has hecho el procedimiento "Burn In" Tener el sensor trabajando más de 48h

Según el caso, escribimos ese nombre en el código (ver rectángulo)


```
ccs811flash Arduino 1.8.13
Archivo Editor Programa Herramientas Ayuda

ccs811flash$ CCS811_FW_App_v1-0-0.h CCS811_FW_App_v2-0-0.h CCS811_FW_App_v2-0-1.h

/*
  ccs811flash.ino - Sketch to flash a CCS811 with 2.0.0 firmware.
  Created by Maarten Pennings 2018 Dec 10, updated 2020-03-07 change on ams website
*/

#include <Wire.h> // I2C library
#include "ccs811.h" // CCS811 library

// The firmware image as byte array in C
#include "CCS811_FW_App_v2-0-0.h"

// Wiring for ESP8266 NodeMCU boards: VDD to 3V3, GND to GND, SDA to D2, SCL to D1, nWAKE to D3 (or GND)
//CCS811 ccs811(D3); // nWAKE on D3

// Wiring for ESP32 NodeMCU boards: VDD to 3V3, GND to GND, SDA to 21, SCL to 22, nWAKE to D3 (or GND)
CCS811 ccs811(23); // nWAKE on 23

// Wiring for Nano: VDD to 3v3, GND to GND, SDA to A4, SCL to A5, nWAKE to 13
//CCS811 ccs811(13);

// nWAKE not controlled via Arduino host, so connect CCS811.nWAKE to GND
//CCS811 ccs811;

void setup() {
  // Enable serial
  Serial.begin(115200);
  Serial.println("");
  Serial.println("Starting CCS811 flasher");
  Serial.print("setup: library    version: "); Serial.println(CCS811_VERSION);
}
```

Ahora comentamos las siguientes líneas para poder subir la nueva versión de firmware.

```
ccs811flash Arduino 1.8.13
Archivo Editor Programa Herramientas Ayuda

ccs811flash$ CCS811_FW_App_v1-0-0.h CCS811_FW_App_v2-0-0.h CCS811_FW_App_v2-0-1.h
// nWAKE not controlled via Arduino host, so connect CCS811.nWAKE to GND
//CCS811 ccs811;

void setup() {
  // Enable serial
  Serial.begin(115200);
  Serial.println("");
  Serial.println("Starting CCS811 flasher");
  Serial.print("setup: library    version: "); Serial.println(CCS811_VERSION);

  // Enable I2C
  Wire.begin();

  // Enable CCS811
  ccs811.set_i2cdelay(50); // Needed for ESP8266 because it doesn't handle I2C clock stretch correctly
  bool ok= ccs811.begin();
  if( !ok ) Serial.println("setup: CCS811 begin FAILED");

  // Print CCS811 versions
  Serial.print("setup: hardware    version: "); Serial.println(ccs811.hardware_version(),HEX);
  Serial.print("setup: bootloader  version: "); Serial.println(ccs811.bootloader_version(),HEX);
  Serial.print("setup: application version: "); Serial.println(ccs811.application_version(),HEX);

  // Check if flashing should be executed
  //if( ccs811.application_version()==0x2000 ) { Serial.println("init: already has 2.0.0"); return; } // Do not reflash
  //Serial.println("setup: comment-out this code line if you want to flash"); return; // Extra precaution
}
```

Compilamos > Subimos el programa y volvemos a Serial monitor.

```
COM6
..... 4600
ccs811: writing 4600 ..... 4088
ccs811: writing 4088 ..... 3576
ccs811: writing 3576 ..... 3064
ccs811: writing 3064 ..... 2552
ccs811: writing 2552 ..... 2040
ccs811: writing 2040 ..... 1528
ccs811: writing 1528 ..... 1016
ccs811: writing 1016 ..... 504
ccs811: writing 504 ..... 0
ccs811: app-verify ok
ccs811: status (app-verify) 30 ok
ccs811: reset2 ok
ccs811: status (reset2) 10 ok

loop: ended ...
loop: ended ...
//
//
```

Vemos como se sube el firmware nuevo en diferentes fases, una vez terminado el proceso volvemos a comentar las líneas de código que previamente hemos des-comentado, compilamos, subimos y volvemos a Serial Monitor.

```
COM6

ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:1044
load:0x40078000,len:8896
load:0x40080400,len:5816
entry 0x400806ac

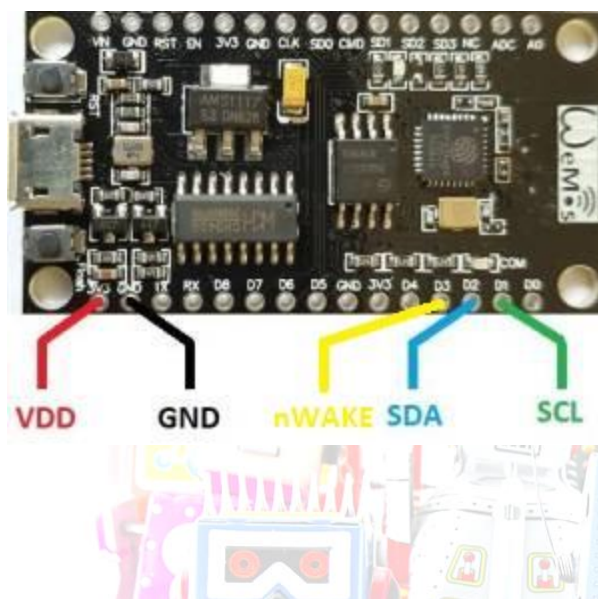
Starting CCS811 flasher
setup: library      version: 10
setup: hardware     version: 12
setup: bootloader   version: 1000
setup: application  version: 2001
setup: comment-out this code line if you want to flash
loop: ended ...
```

Aquí podemos ver que la versión subida (en este caso has sido la 2.01) porque mi sensor todavía no se había hecho el procedimiento “Burn In”

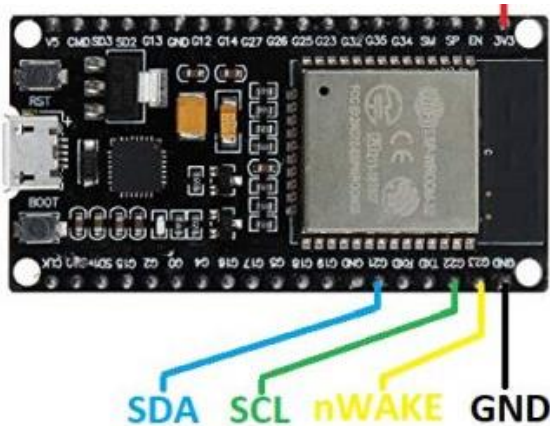
Si queremos subir otro firmware deberemos volver a des-comentar las líneas de código comentadas.

Esquemas de conexión para otras placas.

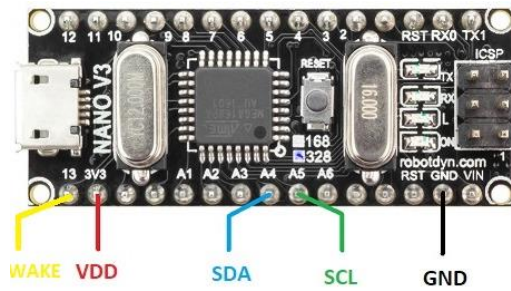
ESP8266 nwake puede ir a D3 o GND



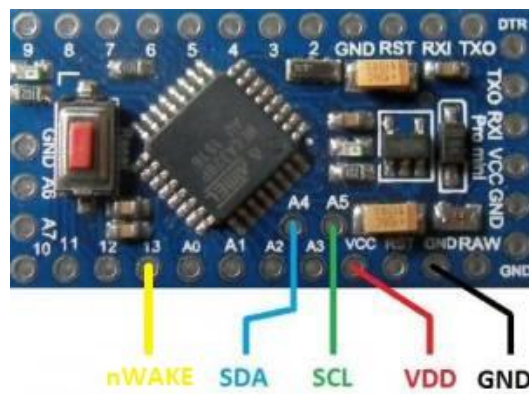
ESP32 nwake puede ir a D23 o GND



Nano V3



Pro Mini



Espero que esta guía sea de utilidad, todos los créditos Maarten Pennings
<https://github.com/maarten-pennings/CCS811>

Autor de la guía: **Ernesto Gutiérrez @erguro1973**