Anotaciones importantes de sensórica.

**Proceso para empezar a usar el microcontrolador ESP32:**

Lo primero es que la placa se llama "Widora ESP32 WiFi and Bluetooth Board"

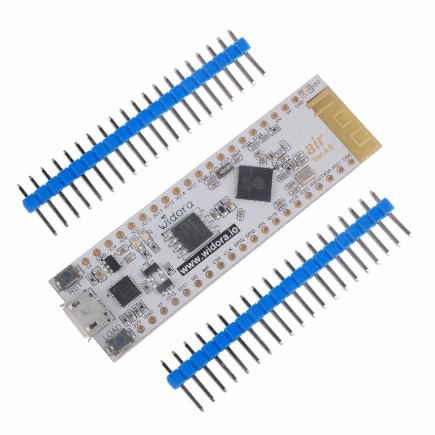
Disponible en el siguiente enlace:

https://www.didacticaselectronicas.com/index.php/comunicaciones/bluetooth/tarjeta-modulo-de-desarrollo-wifi-y-bluetooth-widora-air-v6-0-tarjetas-modulos-de-desarrollo-de-con-wifi-wi-fi-bluetooth-esp32-esp32-widora-air-comunicación-iot-detail

Los pines se muestran en las siguientes páginas:

<https://drive.google.com/file/d/1HEdHKkTQYB1rY_BEUJAKEzjfX1kUFOU1/view>

https://www.smart-prototyping.com/image/data/2\_components/Bluetooth/101766/widora%20air%20sch.pdf



El esquema para conocer las comunicaciones del dispositivo se encuentra en el siguiente enlace:

https://www.smart-prototyping.com/image/data/2\_components/Bluetooth/101766/widora%20air%20sch.pdf

Luego se debe hacer compatible la IDE de Arduino con el ESP32, el tutorial está en el siguiente enlace:

(Si no, no va a funcionar y no lo va a reconocer la IDE):

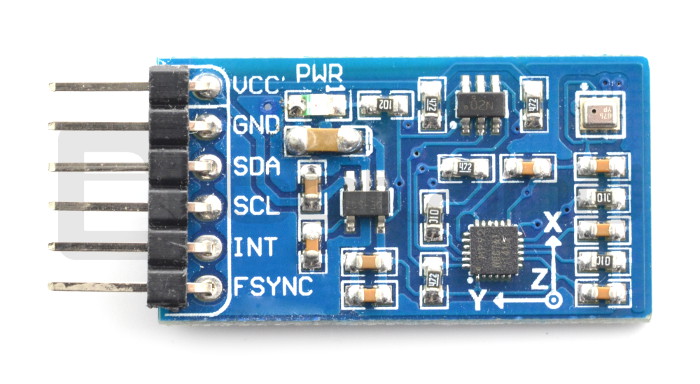
**NOTA:** Dentro del tutorial piden usar un enlace del gestor URL, no usar el de dicho tutorial

si no este “https://dl.espressif.com/dl/package\_esp32\_index.json"

https://solectroshop.com/es/content/31-modulos-wifi-de-arduino-tutorial-esp12e-nodemcu-esp-8266-y-wemos-d1-ch340-

Sensores:

**Acelerómetro MPU-9255**



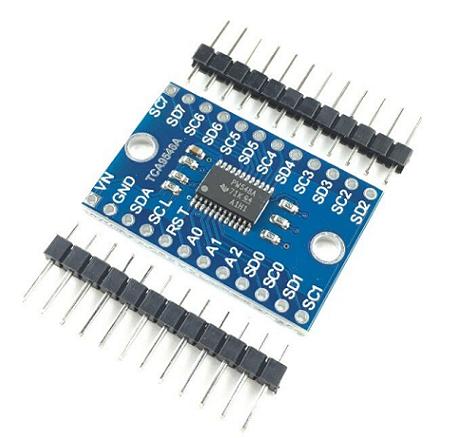
Tiene comunicaciones I2C pero tiene la particularidad de que la dirección física del bus, no puede ser cambiada, lo que significa que para usar varios de estos sensores a la vez, requiere un dispositivo externo que se encarga de conmutar entre las señales, en este caso sería un dispositivo multiplexor del cual se hablará próximamente y tiene referencia **TCA9548a,** el tutorial de uso del sensor se encuentra en el siguiente enlace incluyendo la conexión del dispositivo.

http://arduinolearning.com/code/arduino-and-mpu-9255-sensor-example.php

cabe resaltar que el sensor presentó un ruido de medición el cual puede ser compensado con un filtro pasa bajas que elimina las componentes de alta frecuencia que genera como se puede observar en la siguiente imagen, donde la línea suavizada es el resultado del filtro y la línea con picos es la salida típica del sensor.

Conexión de varios sensores MPU9255, haciendo uso del **multiplexor TCA9548A:**

El multiplexor se encuentra disponible en :  
<https://www.didacticaselectronicas.com/index.php/sistemas-de-desarrollo/compatibles-con-arduino/otros2016-02-07-20-36-39_8/tarjeta-de-expansión-i2c-de-8-canales-tarjetas-de-expansión-extension-de-pines-de-entrada-y-salida-a-pines-i2c-tca9548a-interfaz-i2c-conexion-en-cascada-de-8-canales-tca9548a-detail>



**Nota:** si se requiere cambiar la dirección física del multiplexor, por ejemplo, porque 1 de los sensores tiene la misma dirección I2C que el dispositivo (0x70) , se deben conectar los pines A0 A1 y A2 a Vin o a GND dependiendo del número binario que queramos hacer, partiendo del 70 hasta el 77, donde 70 son todos los pines en la tierra o GND y 77 es todos los pines en la alimentación.

En tutorial de uso y código para detectar la I2C se encuentra en el siguiente enlace:

<https://tronixstuff.com/2019/10/29/tutorial-using-the-tca9548a-1-to-8-i2c-multiplexer-breakout-with-arduino/>

**Sensor de corriente 3144 5V**

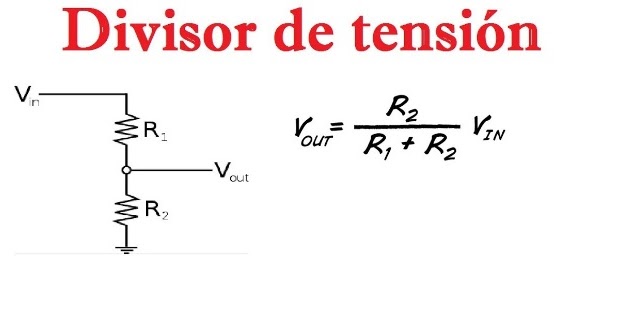


Este sensor es con formato digital, funciona mediante el efecto hall, este sensor debe ser calibrado mediante el Trimmer para elegir el punto en el que un campo magnético generará una salida digital de **5 voltios**, es importante resaltar que este sensor solamente funciona a 5 voltios (no a 3.3 como los pines de entrada y salida del esp32), lo que significa que si tenemos una tarjeta de desarrollo de 3.3 voltios será requerido una etapa de acondicionamiento de la señal mediante un regulador que baje el voltaje de 5 voltios a 3.3 voltios para evitar daños en la tarjeta de desarrollo.

el uso de este sensor no requiere ninguna librería adicional, únicamente la programación mediante dos Estados en los cuales se debe medir el tiempo y haces la resta respectiva para, mediante una ecuación relacionar el radio de la llanta y el tiempo para obtener la velocidad angular o lineal que tenga la moto en ese instante.

Configuración de resistencias para acondicionar la señal digital:

es necesario hacer un arreglo de 2 resistencias en para reducir el voltaje de alimentación de 5 voltios a 3.3 voltios, este arreglo se conoce como divisor de voltaje presente en la siguiente imagen:



Donde requerimos que Vout sea 3.3 V y Vin es un valor conocido de 5 V, para lograr el valor deseado le damos un valor arbitrario a R1 (lo suficientemente grande para no generar una corriente de consumo lo suficientemente alta), despejando de la ecuación del voltaje de salida se encuentra que R2 debe valer el doble de R1.

Para este caso se va a usar un valor de R2 equivalente a 1kΩ y R1 equivalente a 515Ω, haciendo que la corriente generada sea de

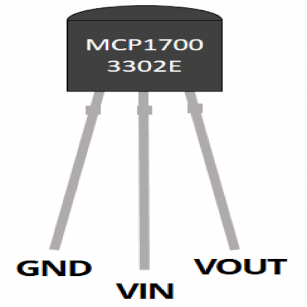
I= V/R

I= 5V/1515 Ω

I= 3.3 mA

Dicha corriente es totalmente tolerable por el microcontrolador (para más información consultar el Datasheet del microcontrolador).

Una forma alternativa de reducir el voltaje sin grandes pérdidas por corrientes es usar un regulador, en este caso el mcp1700 3302e presente a continuación:



En este caso en Vin se conectan 5V y la salida debería presentar 3.3 V,

/////////////////////////////// Interfaz Arduino ///////////////////////////

Aplicación del filtro pasa bajas

Como se explicó anteriormente las señales del sensor son muy bruscas, por lo que se procede a usar un filtro pasa bajas, el cual tiene la función de eliminar los picos de alta frecuencia que entrega el sensor de aceleración, o de otra variable.

Dicho código se encuentra en la siguiente página:

<https://www.luisllamas.es/arduino-paso-bajo-exponencial/>