



S/. 0,00

CATEGORÍAS



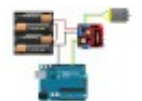
Todos los Posts

Tutorial Magnetómetro HMC5883L

## ARTÍCULOS POPULARES



Tutorial transmisor de celda de carga HX711, Balanza Digital ago 23, 2015



Tutorial de Uso del Módulo L298N feb 19, 2015



Tutorial básico NRF24L01 con Arduino jun 04, 2015

## ARTÍCULOS RECIENTES



Lanzamiento del Arduino IDE Versión 1.6.9 may 13, 2016



Tutorial Magnetómetro HMC5883L may 10, 2016



Tutorial sensor de corriente ACS712 abr 27, 2016

---

## PROMOCIONES ESPECIALES

**Arduino Nano CH340G**

Un pequeño y muy potente Arduino,...

S/. 20,00 ~~S/. 25,00~~[Todas las promociones especiales >](#)

---

## ETIQUETAS



sensor

cnc

Servo

LCD

driver

cable

LED

robot

voltmetro

infrarrojo

---

## NOVEDADES

**Led RGB 5mm**

Agrega colores llamativos a tus proyectos con este led rgb

S/. 0,50

**Módulo Buzzer**

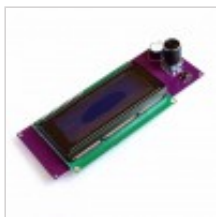
Agrega alertas o efectos de sonido a tus proyectos

S/. 5,00

**Smart controller LCD 12864 para Printer 3D**

Controlador inteligente para impresoras 3D, compatible con Ramps y...

S/. 110,00

**Smart controller LCD 2004 para Printer 3D**

Controlador inteligente para impresoras 3D, compatible con Ramps y...

S/. 75,00

---

**Cargador de batería Li-Po IMAX B3**

Carga tus baterías Li-Po de la forma más rápida y segura

S/. 70,00

---

**Cargador de batería iMax B6AC (compatible)**

Carga tus baterías Li-Po de la forma más rápida y segura

S/. 205,00

---

**Micro Servo MG90S 2.5Kg**

Micro servomotor con engranajes de metal, compacto y sencillo de usar.

S/. 18,00

---

**Medidor Digital AC**

Permite medir voltaje, corriente y la potencia que consume los equipos o...

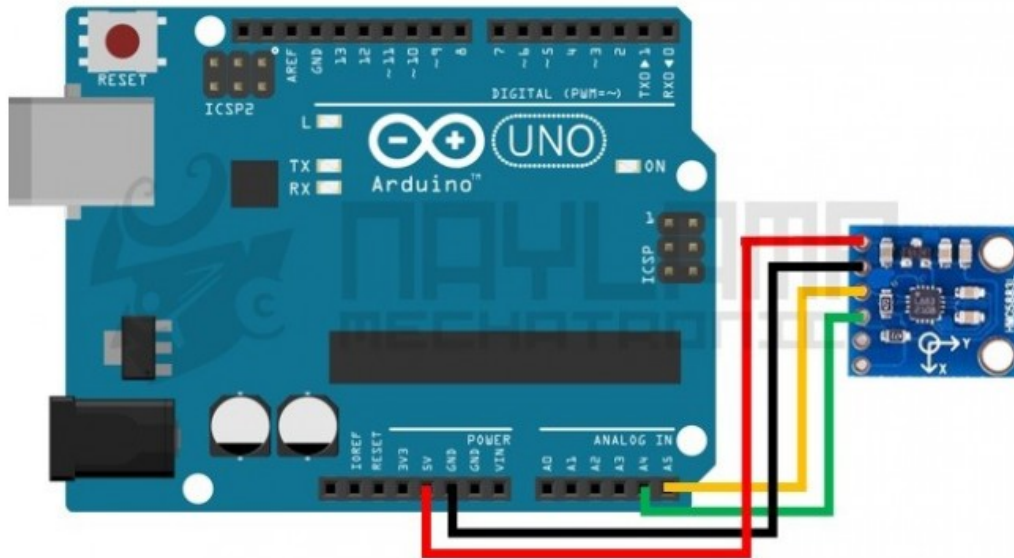
S/. 100,00

---

Todos los nuevos productos >

# Tutorial Magnetómetro HMC5883L

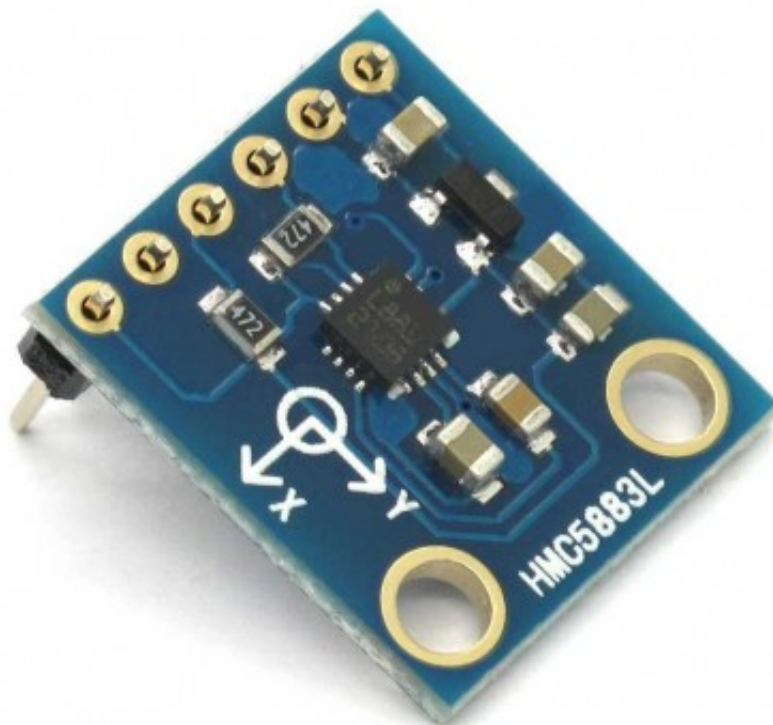
Publicado por Tutoriales 0Comentarios



En este tutorial revisaremos las características de la brújula digital o magnetómetro HMC5883L, explicaremos la conexión y realizaremos ejemplos con Arduino.

EL HMC5883L es un magnetómetro de 3 ejes, con este podemos leer las componentes del campo magnético presente, de esta forma conociendo la dirección del campo magnético terrestre podemos calcular la orientación con respecto al norte magnético de la tierra, esto siempre y cuando nuestro sensor no este expuesto a algún campo magnético externo u algún objeto metálico que altere el campo magnético terrestre.

EL HMC5883L se encuentra comercialmente en módulos, los cuales facilitan su uso, el modelo que usaremos para este tutorial es el que pueden encontrar en nuestra tienda:



El modulo trabaja con 3.3V, pero tiene un regulador interno por lo que se puede alimentar con 5V o con 3.3V en sus pines respectivos.

Su dirección I2C es 0x1E, dirección que no podemos cambiar, por esta razón no se puede conectar otro HMC5883 en el buz I2C, pero si podemos conectar otros dispositivos que tengan dirección distinta.

Tiene dos modos de Funcionamiento, Modo **Continuous-Measurement** (continuo), en donde el magnetómetro está constantemente realizando mediciones y actualizando los registros x,y,z correspondientes a las lecturas. Y el modo **Single-Measurement**, aquí el magnetómetro solo realiza una medida cuando el Arduino le solicite.

Asimismo podemos configurar el rango de medición **desde  $\pm 0.88\text{Ga}$  hasta  $\pm 8.1\text{Ga}$**  a las que se le aplica ganancias de 1370 a 230 respectivamente y de esta forma tener un rango de salida de 12bits.

## Librería para el HMC5883L

En este tutorial trabajaremos con la librería desarrollada por Jeff Rowberg, la librería se descarga en:

<https://github.com/jrowberg/i2cdevlib/tree/master/Arduino/HMC5883L>

Esta librería trabaja con una librería adicional para la comunicación I2C, esta también lo pueden encontrar en el mismo github de la librería anterior:

<https://github.com/jrowberg/i2cdevlib/tree/master/Arduino/I2Cdev>

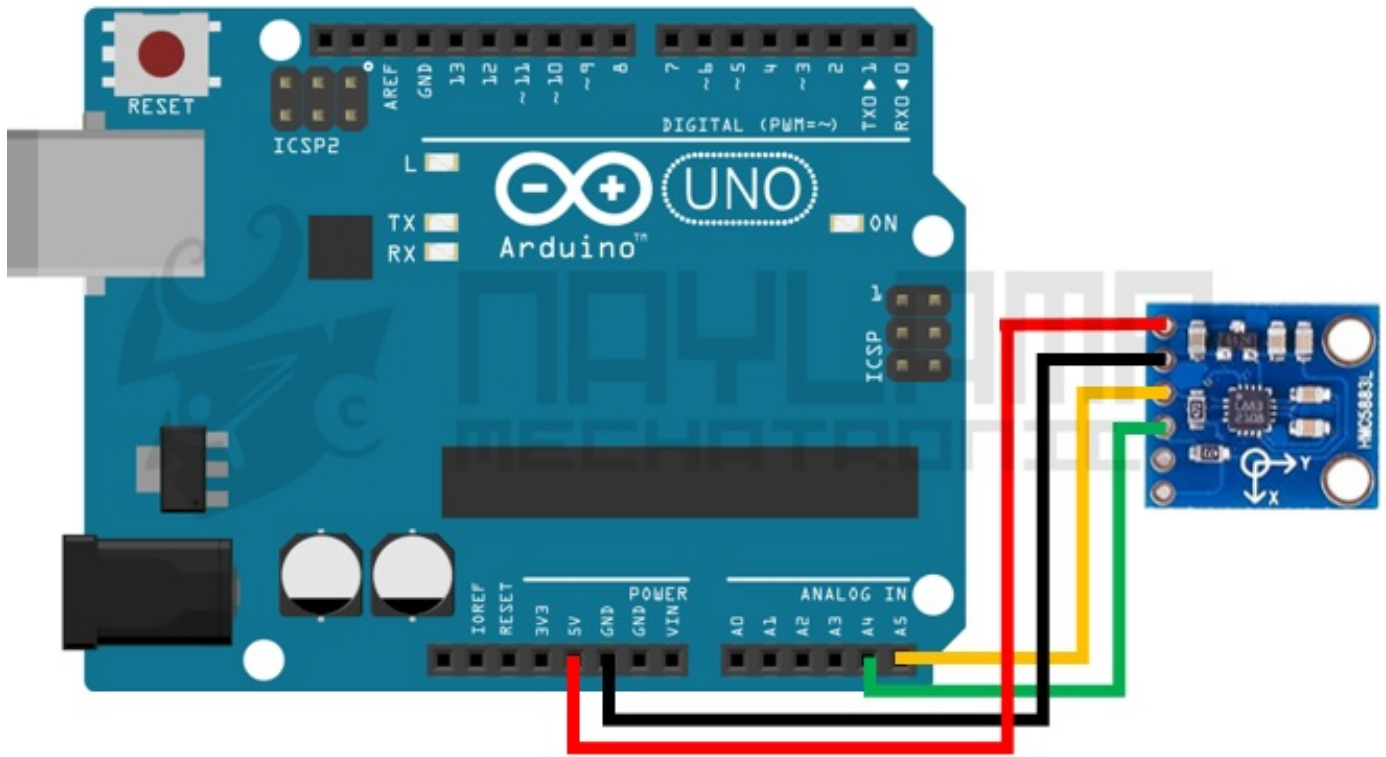
Estas librerías son del mismo desarrollador que las que usamos en el tutorial MPU6050, la librería I2Cdev es la misma que usamos en dicho tutorial.

Para trabajar los siguientes ejercicios es necesario instalar las librerías en el IDE Arduino.

## Conexiones entre HMC5883L y Arduino

Las conexiones son como cualquier conexión I2C:

HMC5883L	Arduino Uno, Nano, Mini.	Arduino Mega , DUE	Arduino Leonardo
VCC	5V	5V	5V
GND	GND	GND	GND
SCL	A5	21	3
SDA	A4	20	2



## Realizando lecturas del campo magnético:

En este ejemplo realizaremos las lecturas del HMC5883L, obtendremos los componentes en X,Y y Z del campo magnético.

El código para este ejemplo es el siguiente:

```
#include "Wire.h"
#include "I2Cdev.h"
#include "HMC5883L.h"

HMC5883L magnetometro;

int16_t mx, my, mz;
void setup() {

    Serial.begin(9600);
    Serial.println("Inicializando Magnetometro...");
    //Inicializamos la comunicación I2C y el magnetómetro
    Wire.begin();
    magnetometro.initialize();
}

void loop() {
    //Obtenemos del magnetometro las componentes del campo magnético
    magnetometro.getHeading(&mx, &my, &mz);

    Serial.print("mx:");
    Serial.print(mx);
    Serial.print("tmy:");
    Serial.print(my);
```

```
Serial.print("tmz:");  
Serial.println(mz);  
delay(100);  
}
```

Como se puede observar el programa es fácil de entender, podemos resumir la explicación a 3 partes de código.

Primero es necesario incluir las librerías y declarar el objeto o variable para el magnetómetro:

```
#include "Wire.h"  
#include "I2Cdev.h"  
#include "HMC5883L.h"  
  
HMC5883L magnetometro;
```

Posteriormente en setup() es necesario inicializar la comunicación I2C y el magnetómetro:

```
Wire.begin();  
magnetometro.initialize();
```

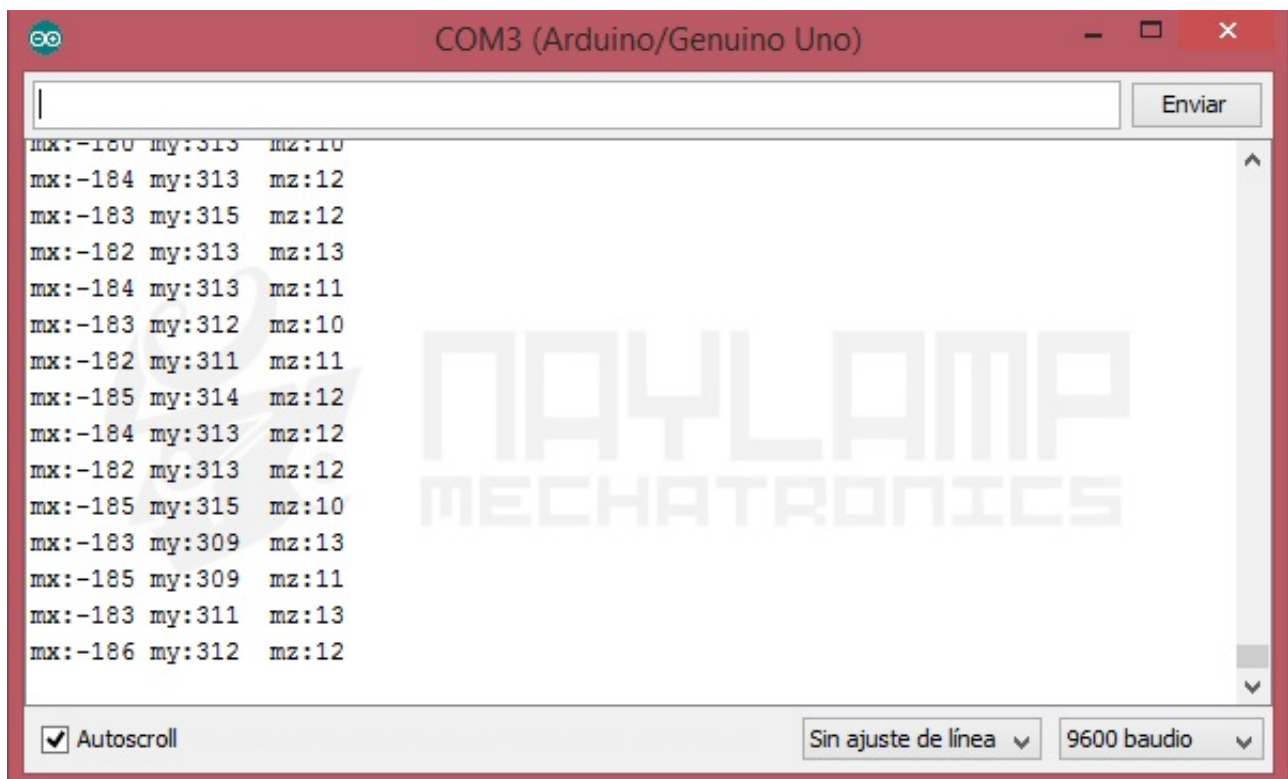
Al inicializar el magnetómetro los valores por defecto para el rango es de  $\pm 1.3\text{Ga}$  y de ganancia 1090.

Posteriormente se realiza la lectura, esto se hace de la siguiente forma:

```
magnetometro.getHeading(&mx, &my, &mz);
```

De esta forma tendremos los valores del campo magnético en las variables mx, my y mz, estos valores están con la ganancia mencionada anteriormente.

Posteriormente enviamos los valores por el puerto serie, esta última parte variará de acuerdo a su aplicación.



Nuevamente mencionamos que estos valores están con ganancia, para tener el valor real hay que dividirlo entre 1090 (ganancia por defecto) y de esta forma obtener la medida en Gauss.

Si queremos modificar el rango y ganancia debemos agregar después de inicializar el magnetómetro la siguiente línea de código:

```
magnetómetro.setGain(value)
```

Donde value corresponde a:

Value	Rango	Ganancia (LSB/Gauss)
0	+/- 0.88 Ga	1370
1	+/- 1.3 Ga	1090 (Default)
2	+/- 1.9 Ga	820
3	+/- 2.5 Ga	660
4	+/- 4.0 Ga	440
5	+/- 4.7 Ga	390
6	+/- 5.6 Ga	330
7	+/- 8.1 Ga	230

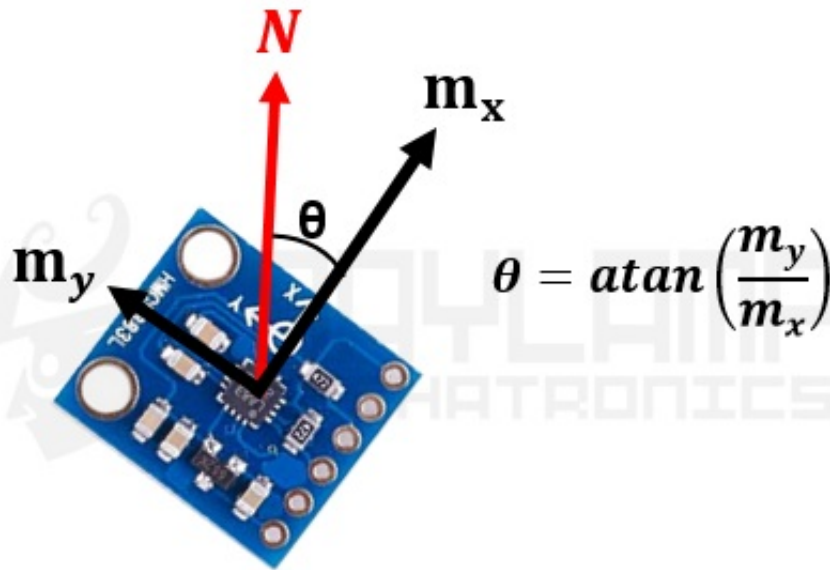
De esta forma podemos establecer un nuevo rango de medición. El campo magnético terrestre dependiendo del lugar puede variar entre 0,25-0,65 Ga, teniendo en cuenta esto es mejor si solo vamos a medir el campo terrestre es mejor trabajar con el rango por defecto, si queremos mayor precisión podemos bajar el rango a 0.88Ga pero se podría saturar la lectura si existe un campo magnético externo.

## Brújula digital con Arduino

En este ejemplo implementaremos una brújula, para esto necesitamos calcular el ángulo de nuestra orientación con respecto al Norte.

Para determinar el ángulo usamos la siguiente forma:





Este ángulo nos determina la orientación del Norte Magnético, pero existe una diferencia entre el norte geográfico y el norte magnético, a esta diferencia se le conoce como declinación magnética.

El valor de la declinación magnética depende de nuestra ubicación y lo pueden obtener en la siguiente página: <http://www.magnetic-declination.com/>, en nuestro caso, para la ciudad de Trujillo-Perú es de 7°2'

El código en donde implementamos lo mencionado anteriormente es el siguiente:

```
#include "Wire.h"
#include "I2Cdev.h"
#include "HMC5883L.h"

HMC5883L magnetometro;

int16_t mx, my, mz;
float declinacion=7.33; //declinación de 7°2'(Trujillo-Perú)
void setup() {

    Serial.begin(9600);

    Serial.println("Inicializando Magnetometro...");
    //Inicializamos la comunicación I2C y el magnetómetro
    Wire.begin();
    magnetometro.initialize();
}

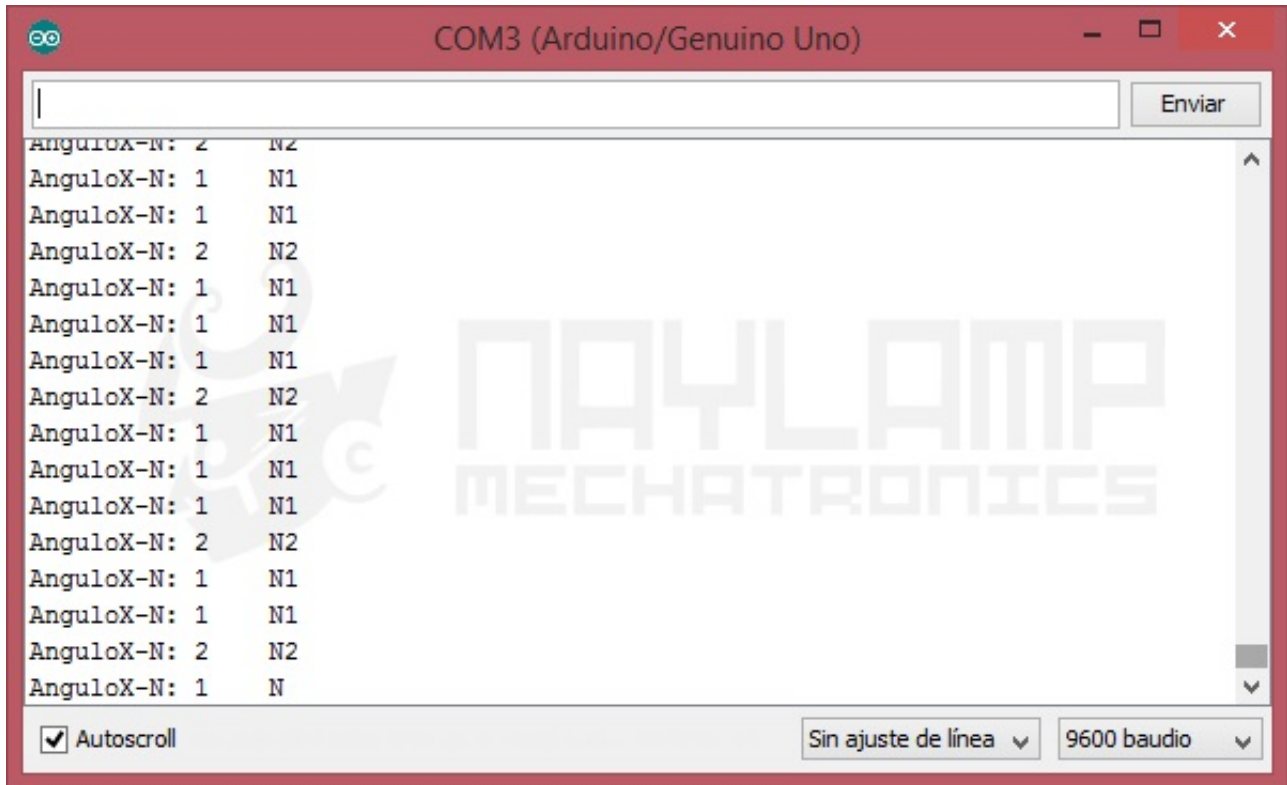
void loop() {
    //Obtenemos del magnetómetro las componentes del campo magnético
    magnetometro.getHeading(&mx, &my, &mz);

    //Calculamos el ángulo del eje X con respecto al norte
    float angulo = atan2(my, mx);
    angulo=angulo*(180/M_PI); //convertimos de Radianes a grados
    angulo=angulo-declinacion; //corregimos la declinación magnética
    //Mostramos el angulo entre el eje X y el Norte
    Serial.print("AnguloX-N: ");
    Serial.print(angulo,0);

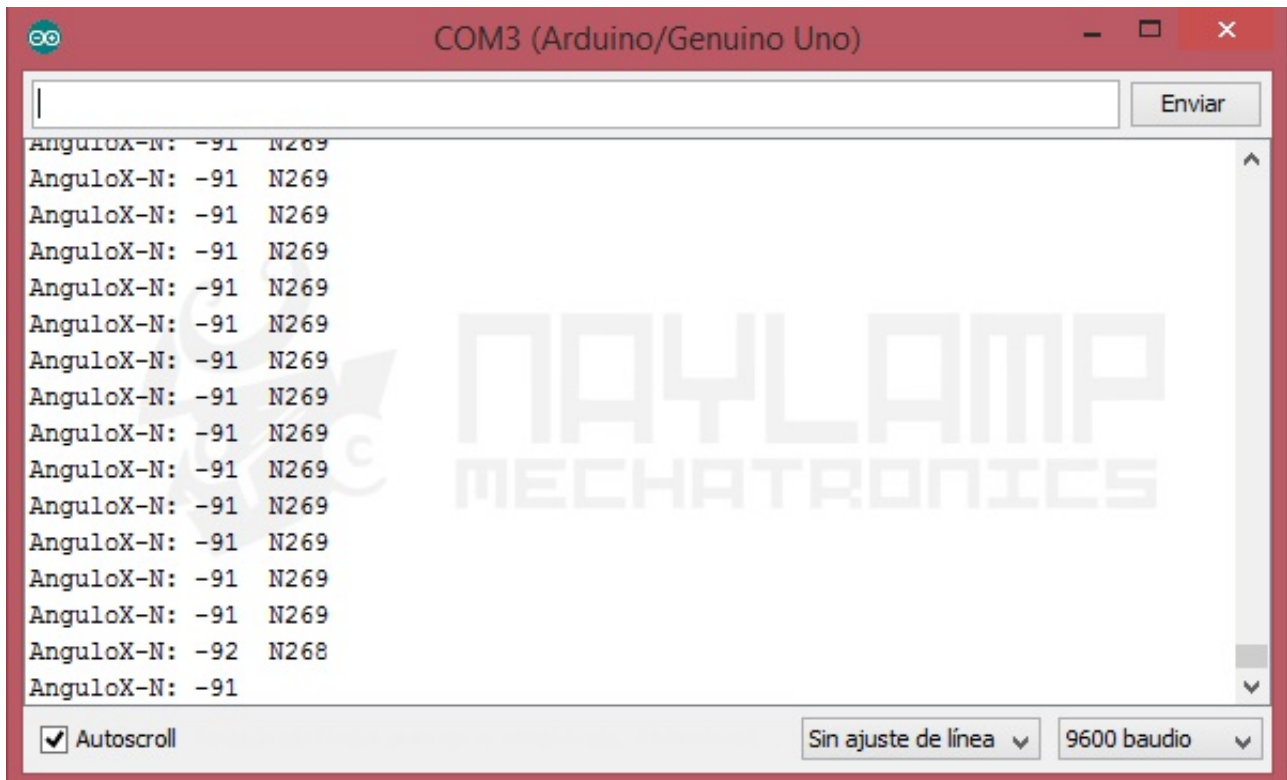
    //calculamos el ángulo equivalente de [-180 180] a [0 360]
    if(angulo<0) angulo=angulo+360;
    Serial.print("tN");
```

```
Serial.println(angulo, 0);  
}
```

En la siguiente imagen mostramos el resultado cuando apuntamos el eje X en la dirección Norte



Y la siguiente imagen es para cuando apuntamos el eje x al Oeste.



Estos valores serán correctos siempre y cuando no existan campos magnéticos externos, o metales cerca que interfieran con el campo magnético. Por ejemplo si acercamos nuestro magnetómetro a nuestra PC o acercamos un objeto metálico al magnetómetro, el valor del ángulo variara sin haber rotado el magnetómetro.

## Pueden adquirir los materiales usados en este tutorial en nuestra tienda:

- Arduino Uno R3
- Módulo Magnetómetro 3-ejes HMC5883L

0      ?      0      0      0

### Leave a Reply

\* **Name:**

\* **E-mail:**

(Not Published)

**Website:**

(Site url withhttp://)

\* **Comment:**

Submit

### Boletín

Introduzca su dirección de correo electróni



### Información



**Contacte con nosotros**

**Entrega**

**Condiciones de uso**

**Nosotros**

**Tutoriales y Proyectos con Arduino**

**Mapa del sitio**

## Mi cuenta



Mis compras

Mis vales descuento

Mis direcciones

Mis datos personales

Mis vales

## Información sobre la tienda



Naylamp Mechatronics, Trujillo Perú



Llámanos ahora: 997646858



Email: [naylamp.mechatronics@gmail.com](mailto:naylamp.mechatronics@gmail.com)