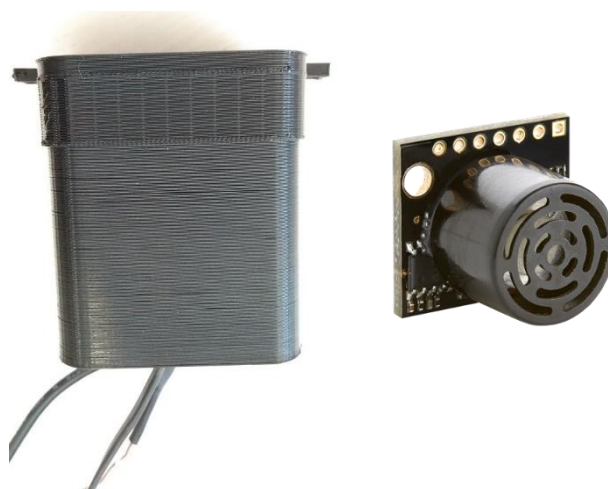




SENSOR PRESENCIAL MIRROR VIDEO WALL



DESCRIPCION:

El sensor es capaz de detectar a una persona u objeto en un rango de 0 – 7.65 m ajustables a la distancia que se dese calibrar, funciona en conjunto del microcontrolador Arduino, su funcionalidad radica en mandar un pulso de 3.3 V al puerto 3 GPIO Brightsign (BUTTON 0).

Básicamente manda un disparo SWITCHEO entre contenidos (presentaciones)

CARACTERISTICAS:

- Sensor Ultrasónico de 42 kHz (Pasivo)
- Dimensiones: 2.2 x 2.0 x 2.5 cm y 6,1 g
- El módulo incluye el sensor, Arduino, cable de instalación y case de apoyo para una fácil utilización
- Rango de detección: 0 m a 7.65 m, ajustable mediante código Arduino.
- Lente 1 zona, ángulo = 43º
- Salida activa alta a 3.3 V
- Consumo de corriente en reposo: < 50 µA
- salida serial: asincrónica, nivel lógico, invertida, 9600 bps 8N1
- salida analógica: 10 mV / pulgada con alimentación de 5V

DIAGRAMA DE CONEXIÓN (interno).

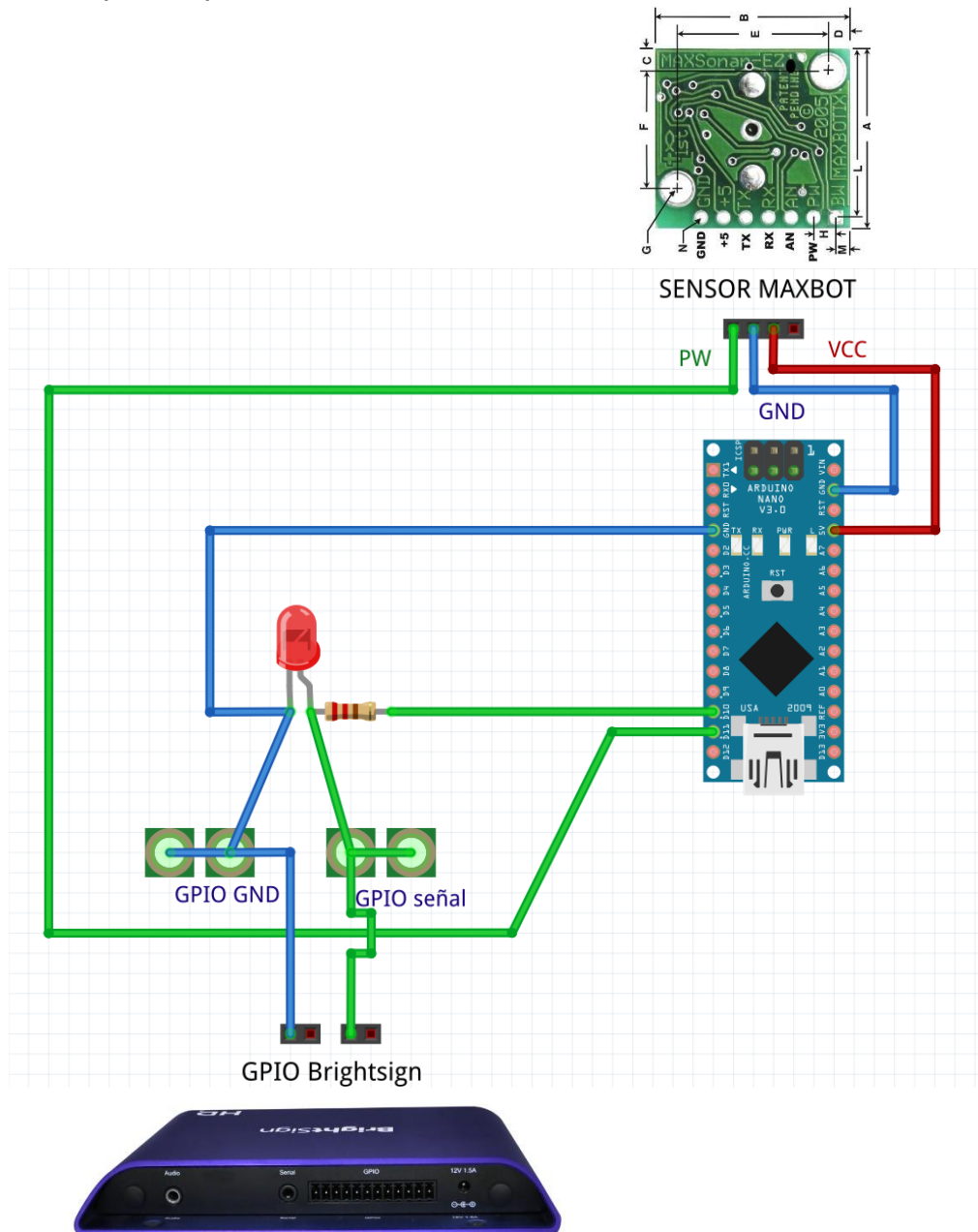
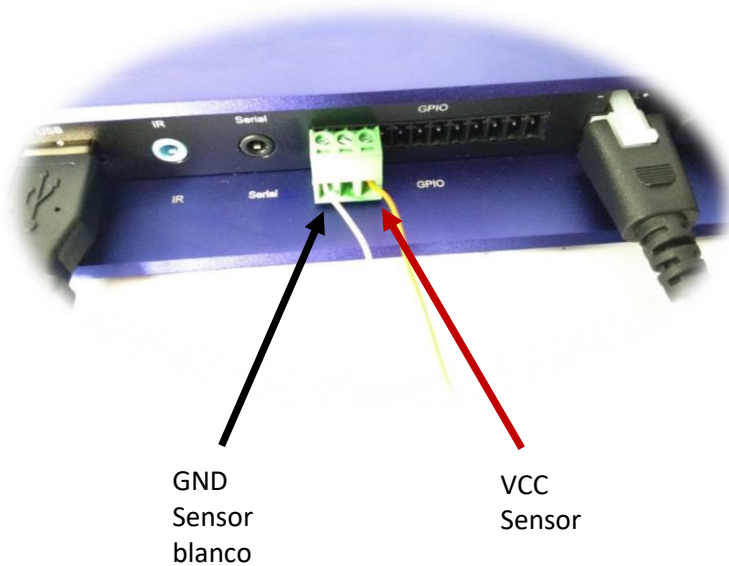


DIAGRAMA DE CONEXIÓN EN SITIO.

Conexión a Brightsign

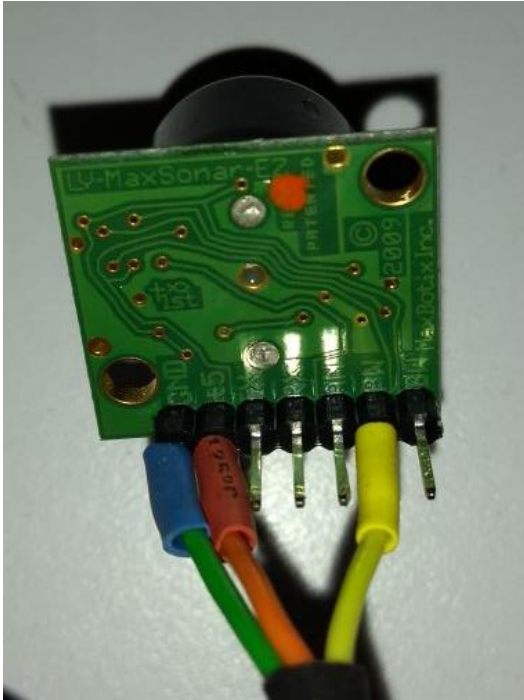
1.- Realice conexión como se muestra en la figura.

- a.- USB (izquierdo)
- b.- GPIO (centro)
- c.- Alimentación (derecho)



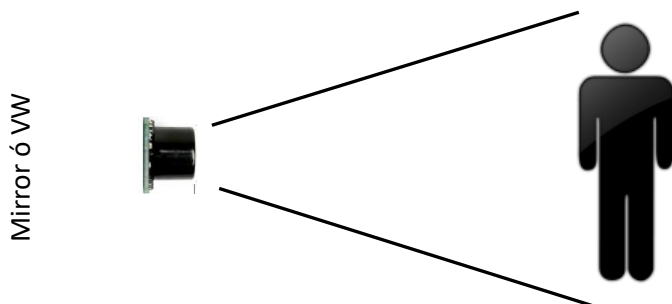
Nota: El cable de color blanco siempre deberá de ir de lado izquierdo y el cable de color lado derecho como se muestra en la figura.

2.- Verifique la conexión del lado del sensor sea correcta



Azul – GND
 Rojo – 5Vcc
 Amarillo – PW

3.- El sensor se debe de colocar de forma que apunte a la ubicación a sensor.



El sensor esta calibrado a una distancia de 90 cm max, sobrepasando dicha distancia no realizará efecto alguno.

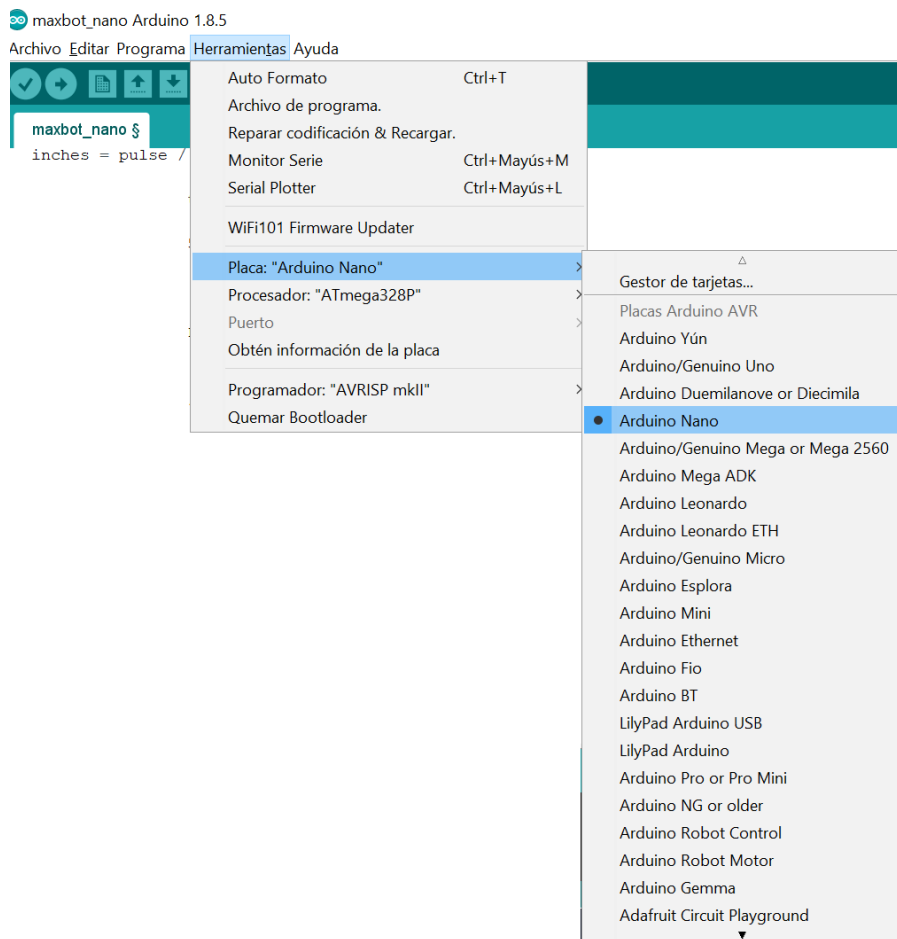
CALIBRACIÓN

El IDE Arduino se puede descargar de forma gratuita en el siguiente enlace.

<https://www.arduino.cc/en/Main/Software>

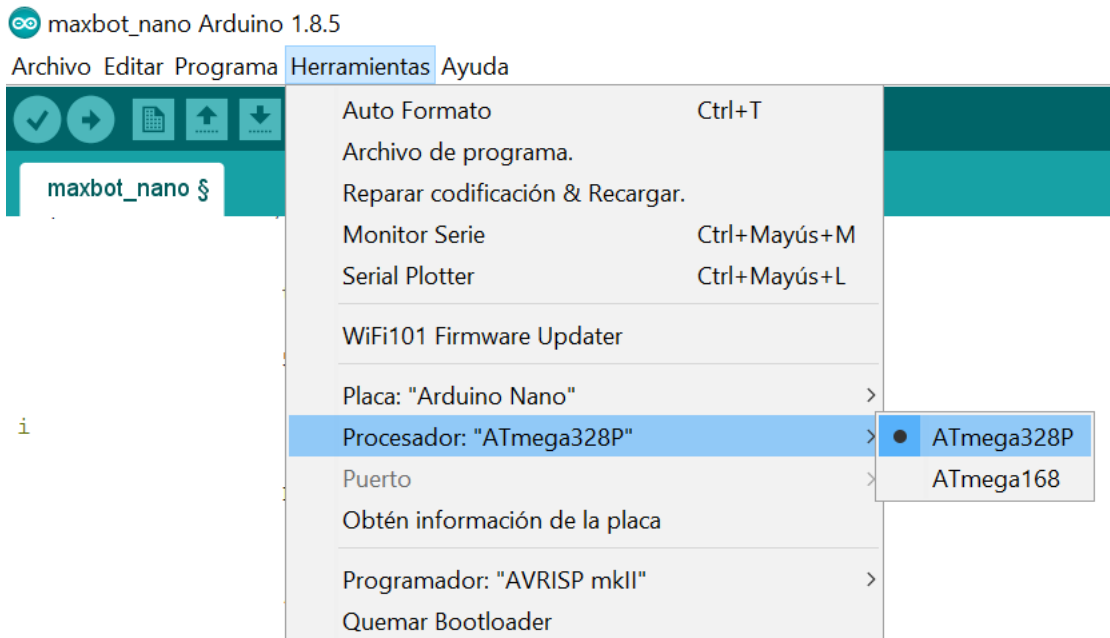
Para poder cargar correctamente el programa al Arduino con la nueva medida se requiere que se validen los siguientes pasos.

- 1.- Conectar el sensor mediante el cable USB a la computadora.
- 2.- Abrir IDE Arduino. (previamente descargado)
- 3.- Abrir Archivo maxbot_nano.ino (adjunto en el mismo ZIP)
- 4.- Dirigirse a la barra de menú en la ruta: **Herramientas\Placa\Arduino Nano**
- 5.- Seleccionamos Arduino Nano



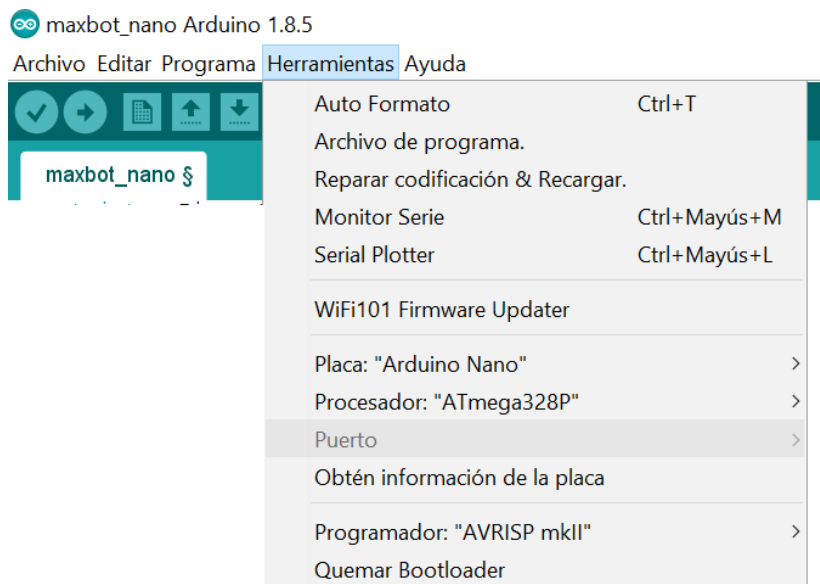
6.- Dirigirse a la barra de menú en la ruta: **Herramientas\Placa\procesador**

7.- Seleccionamos ATmega328P.




8.- Dirigirse a la barra de menú en la ruta: **Herramientas\Placa\puerto**

9.- Seleccionamos el puerto en el cual conectamos nuestro sensor.

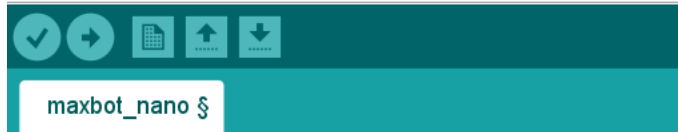


10.- Archivo maxbot_nano.ino

El siguiente código es el usado para la calibración del sensor en unidades CM, en el cuadro rojo se muestra la línea de código a modificar dependiendo la distancia que se desee.

 maxbot_nano Arduino 1.8.5

Archivo Editar Programa Herramientas Ayuda



```
//kolo sensor maxbot.
// Date: 23/07/09
// pin 11 entrada
// pin 9 salida
```

```
const int pwPin = 11;
const int ledPIN = 9;
```

```
long pulse, inches, cm;
```

```
void setup()
```

```
{
```

```
    Serial.begin(9600);
```

```
}
```

```
void loop()
```

```
{
```

```
    pinMode(pwPin, INPUT);
```

```
    pinMode(ledPIN , OUTPUT);
```



```
pulse = pulseIn(pwPin, HIGH);

//147uS per inch
```

```
inches = pulse / 147;

//change inches to centimetres

cm = inches * 2.54;
```

```
if (cm < 90) {
```

Distancia a sensor en [cm]

```
digitalWrite(ledPIN , HIGH);
```

```
Serial.print(cm);
```

```
Serial.print("cm");
```

```
Serial.println();
```

```
}
```

```
else
```

```
{
```

```
digitalWrite(ledPIN , LOW);
```

```
}
```

```
delay(500);
```

```
}
```

11.- una vez que modificamos el código pulsamos el botón verificar, seguido de subir.



12.- Si todo está en orden en la parte inferior izquierda aparecerá una leyenda “SUBIDO” la cual nos indicara que fue programado con éxito.

13.- Ya podremos desconectar el sensor de la computadora y regresar a conexión en Brightsign y funcionara de manera correcta con la distancia configurada previamente.

TABLA DE PINES BRIGHTSING GPIO

Pin	Function	Pin	Function
1	GND	7	GND
2	3.3V	8	3.3V
3	BUTTON 0	9	BUTTON 4
4	BUTTON 1	10	BUTTON 5
5	BUTTON 2	11	BUTTON 6
6	BUTTON 3	12	BUTTON 7

NOTAS:

En conjunto de este documento se adjunta archivo con extensión Arduino (.ino) para la calibración del mismo, así como el IDE Arduino par una arquitectura de 64 bits.