## **SENSOR LIDAR**

El nombre de LIDAR en realidad es un acrónimo que podría traducirse como sistema de medición y detección de objetos mediante láser. Los LIDAR son sensores que emiten rayos de luz. Cuando ese rayo de luz toca un objeto rebota y vuelve a la posición en la que se encuentra el sensor. Calculando el tiempo que tarda el rayo láser en ir y volver se puede descifrar la distancia que hay entre el sensor y cualquier objeto. Para eso es necesario contar con el emisor de rayos láser infrarrojos, una lente que recoja los haces de luz cuando reboten y un chip o sistema que procese todos esos datos hasta construir un mapa 3D de la escena que hay delante del sensor. El sensor mide el cambio de fase entre las señales transmitidas y reflejadas.

La longitud de onda de la señal de modulación obedece a la ecuación:

$$C = f \cdot \tau$$

Donde  ${\bf c}$  es la velocidad de la luz,  ${\bf f}$  la frecuencia de modulación y  ${\bf \tau}$  la longitud de onda de modulación conocida.

La distancia total D'cubierta por la luz emitida es:

$$D' = B + 2A = B + (\theta * \tau) / 2\pi$$

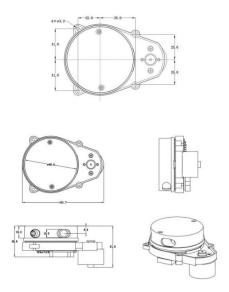
Donde **A** es la distancia medida y **B** es la distancia desde la unidad de medida de fase. Por lo tanto, la distancia requerida **D**, entre el divisor de haz y el objetivo, viene dada por:

$$D = \tau * \theta / 4\pi$$

Donde  ${\bf \theta}$  es la diferencia de fase medida electrónicamente entre los haces de luz transmitidos y reflejados.

El LIDAR que utilizaremos será el YDLIDAR X4. Es un escáner de rango láser bidimensional de 360 grados (LIDAR). Este dispositivo utiliza el principio de triangulación para medir la distancia, junto con el diseño de algoritmo óptico, eléctrico y apropiado, para lograr una medición de distancia de alta precisión.

Su rango va desde los 0.12 metros hasta los 3. Tiene una resolución menor que 0.5 mm para un rango menor a 2 m, y de un 1% mejor que la distancia medida para cifras superiores a 2 m.



El error de distancia es muy pequeño, tiene una gran estabilidad y puntería. Además, cuenta con una gran resistencia a la interferencia de la luz. Su tamaño es pequeño, consume poca batería y tiene una duración de vida larga.

```
#include <Wire.h>
#include <VL53L0X.h>
                                               //Downland it here: https://www.electronoobs.com/eng_arduino_Adafruit_VL53L0X.php
       VL53L0X sensor;
                                               //Define our sensor
       #define LONG_RANGE
#define HIGH_SPEED
//#define HIGH_ACCURACY
                                      //Pin para la dirección del conductor paso a paso.
//Pin para pasos del conductor paso a paso.
//Pin para habilitar el controlador paso a paso.
        //Variables
int Value = 1200;
float angle = 0;
      36
37
       float maxdist = 400;
bool loop_starts = false;
byte last_PIN_state;
                                                  //He establecido la distancia máxima alrededor del sensor a solo 400 mm. Cambiar a cualquier otro valor.
39
40
41
42
44 p void setup() {
           // Declarar pines como saltau:
pinMode(stepPin, OUTPUT);
pinMode(dirPin, OUTPUT);
pinMode(Enable, OUTPUT);
digitalWrite(Enable,LOW);
digitalWrite(dirPin, HIGH);
//Coloque dirPin en ALTO para que giremos en sentido horario
Sarial.begin(9600);
//Iniciar puerto serie
45
            // Declarar pines con
46
47
48
49
50
51
52
53
54
            sensor.init();
55
56
57
            sensor.setTimeout(500);
            Sensor-SetTimeou(1908); //habilite el escaneo PCMSK0 para que podamos usar interrupciones
PCMSK0 |= (1 << PCINT0); //Establecer pin "D8" desencadenar una interrupción en "cualquier" cambio de estado.
//Ver el vector de interrupción debajo del bucle vacío
58
59
60
61
            #if defined LONG_RANGE
62
               // baja el límite de velocidad de la señal de retorno (el valor predeterminado es 0.25~\text{MCPS}) sensor.setSignalRatelimit(0.1);
63
               sensor.setSignalKateLimit(0.1);
// aumentar Los períodos de pulso Láser (Los valores predeterminados son 14 y 10 PCLK
sensor.setVcselPulsePeriod(VL53L0X::VcselPeriodPreRange, 18);
sensor.setVcselPulsePeriod(VL53L0X::VcselPeriodFinalRange, 14);
64
65
66
67
68
            #endif
            #if defined HIGH_SPEED
         // reduce el presupuesto de tiempo a 20 ms (el valor predeterminado es aproximadamente 33 ms)
sensor.setMeasurementTimingBudget(20000);
69
70
71
            #elif defined HIGH_ACCURACY
// // aumentar el presupuesto de tiempo a 200 ms
72
73
74
75
            // sensor.setMeasurementTimingBudget(200000);
            #endif
76
         ]//Fin del bucle de configuración
  // sensor.setMeasurementTimingBudget(200000);
#endif
]//Fin del bucle de configuración
                                                  //Restablecemos el ángulo cuando se detecta el imán en D8
           angle = 0;
loop_starts = false;
}
            digitalWrite(stepPin, HIGH);

delayMicroseconds(Value);

delayMicroseconds(Value);

delayMicroseconds(Value);

int r = sensor.readRangeSingleMillimeters();

f(r) = maxist)

//koer to distancia del sensor

//koer the distance to maximum set distance obove
             r = maxdist;
             }
Serial.print(angle);
Serial.print(",");
Serial.print(r);
Serial.println(",");
                                                   //Imprima los valores en el puerto serie
            angle = angle + angle step; //Aumente el valor del ángulo por el valor del ángulo / bucle establecido anteriormente (en este caso 2.4º por bucle)
         )//end of void Loop
   104
105
```