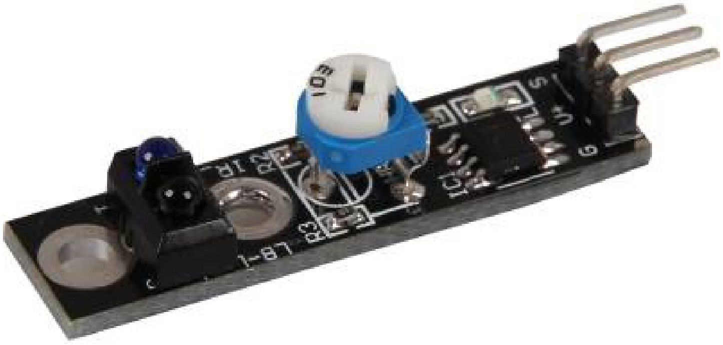
KY-033 Tracking sensor module

Contents

Picture

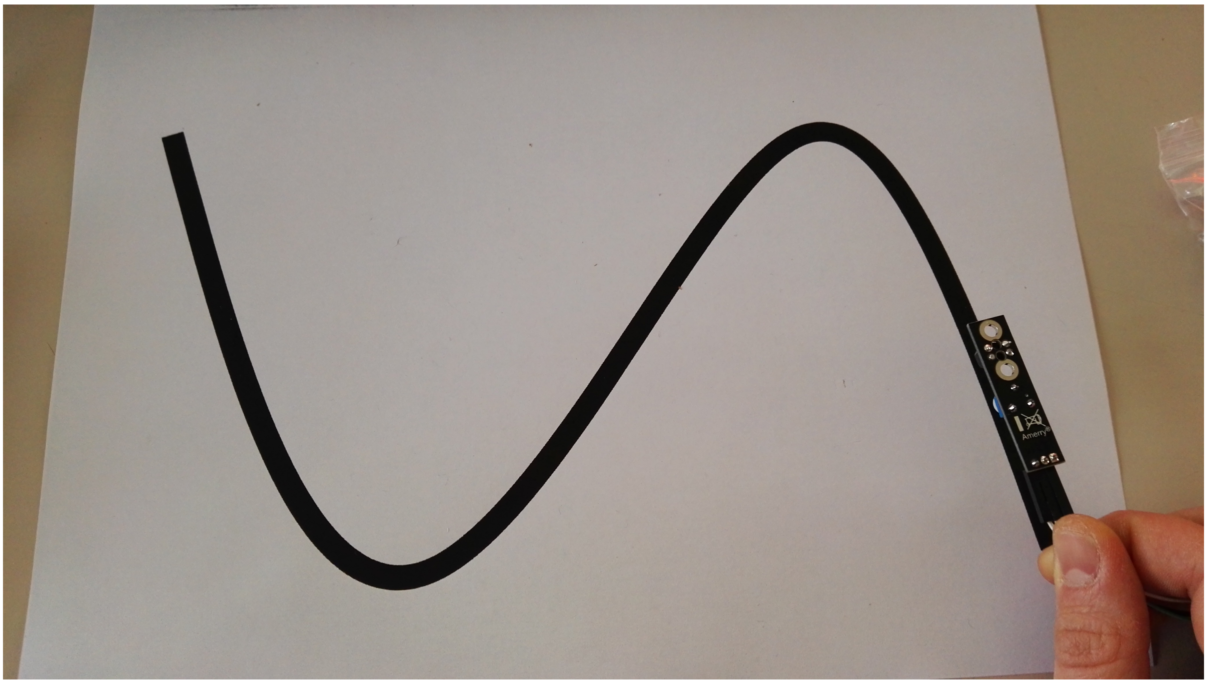


Technical data / Short description

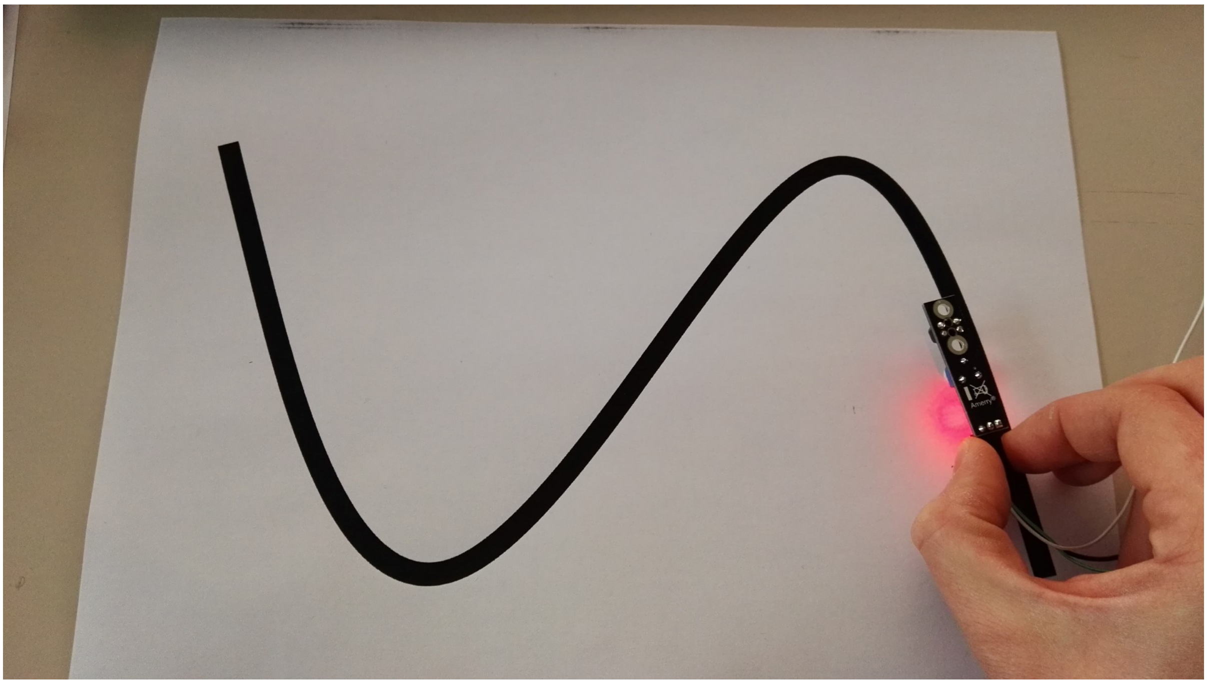
The sensor detects if a light reflecting or absorbing area is in front of it. It shows which of the 2 areas it is via digital output, as you can see in the picture below.
The Sensitivity (minimum range) of the sensor can be adjusted by the controller.

This behavior can be used to automatically follow a line with a robot.

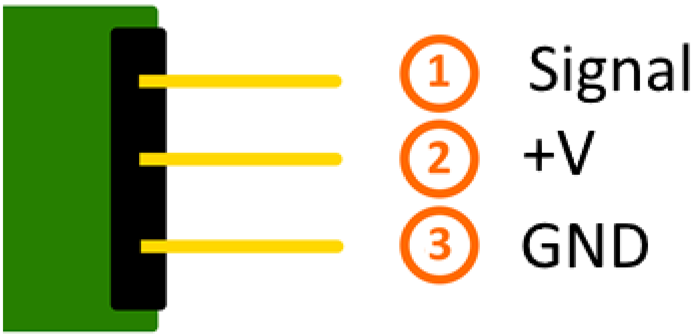
Condition 1: Line Tracker is on a line (not reflecting area) [LED on the module: Off] [Sensor signal= Digital On]



Condition 2: Line Tracker not on a line (reflecting area) [LED on the module: ON] [Sensor signal= Digital Off]



Pinout



Code example Arduino

```
int Sensor = 10; // Declaration of the sensor input pin

void setup ()
{
  Serial.begin(9600); // Initialization serial output
  pinMode (Sensor, INPUT) ; // Initialization sensor pin
}

// The program reads the status of the sensor pins
// shows via serial terminal if the linetracker is on the line or not
void loop ()
{
  bool val = digitalRead (Sensor) ; // The current signal of the sensor will be read

  if (val == HIGH) // If a signal is detected the LED will light up.
  {
    Serial.println("LineTracker is on the line");
  }
  else
  {
    Serial.println("Linetracker is not on the line");
  }
  Serial.println("------------------------------------");
  delay(500); // Break of 500ms between the measurements
}
```

Connections Arduino:

Sensor signal	= [Pin 10]
Sensor +V	= [Pin 5V]
Sensor GND	= [Pin GND]

Example program download

KY-033_Tracking-sensor_ARD

Code example Raspberry Pi

```
# Needed modules will be imported and configured
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)

# Declaration of the input pin which is connected with the sensor
GPIO_PIN = 24
GPIO.setup(GPIO_PIN, GPIO.IN, pull_up_down = GPIO.PUD_UP)

# Break between the results will be defined here (in seconds)
delayTime = 0.5

print "Sensor-Test [press ctrl+c to end]"

# main program loop
try:
        while True:
            if GPIO.input(GPIO_PIN) == True:
                print "LineTracker is on the line"
            else:
                print "Linetracker is not on the line"
            print "---------------------------------------"

            # Reset + Delay
            time.sleep(delayTime)

# Scavenging work after the end of the program
except KeyboardInterrupt:
        GPIO.cleanup()
```

Connections Raspberry Pi:

Signal	=	GPIO24	[Pin 18]
+V	=	3,3V	[Pin 1]
GND	=	GND	[Pin 6]

Example program download

KY-033_Tracking-sensor

To start, enter the command:

```
sudo python KY-033_Tracking-sensor.py
```