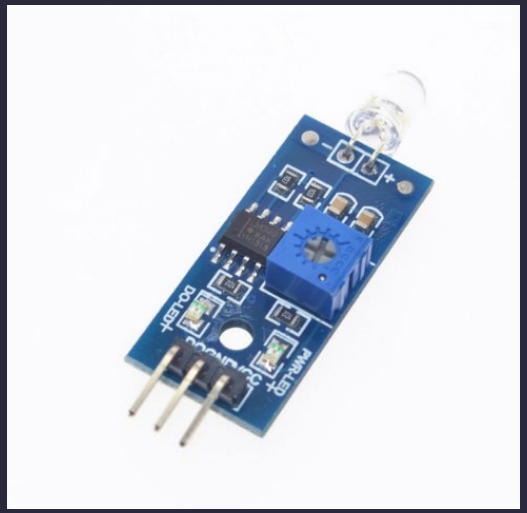
Lighting Sensor

Dr. Jirapipat Thanyaphongphat

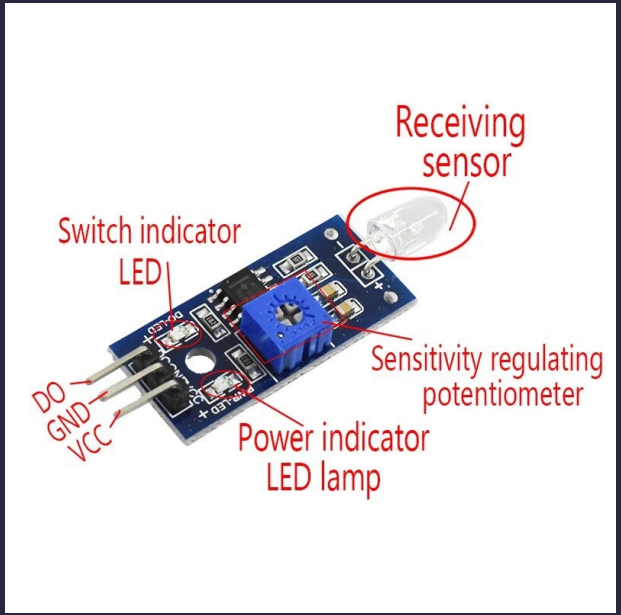
/TABLE OF CONTENTS

/01

/light sensor module LM393



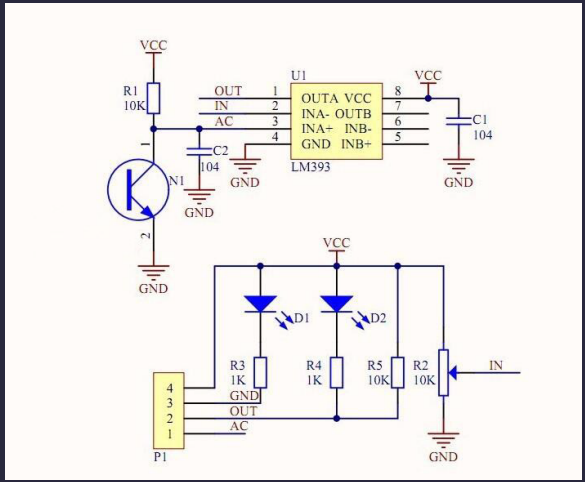
light sensor module LM393



An LDR, or Light Dependent Resistor, is a type of resistor that changes its electrical conductance in response to light. It is made of a semiconductor material that is sensitive to light. Sometimes, this type of sensor is referred to as a "photoresistor" or "photoconductor." It's used in light-sensing modules to detect light levels and is capable of producing output signals in analog (A0) with values ranging from 0 to 1023 and digital (D0) with values of 0 or 1.

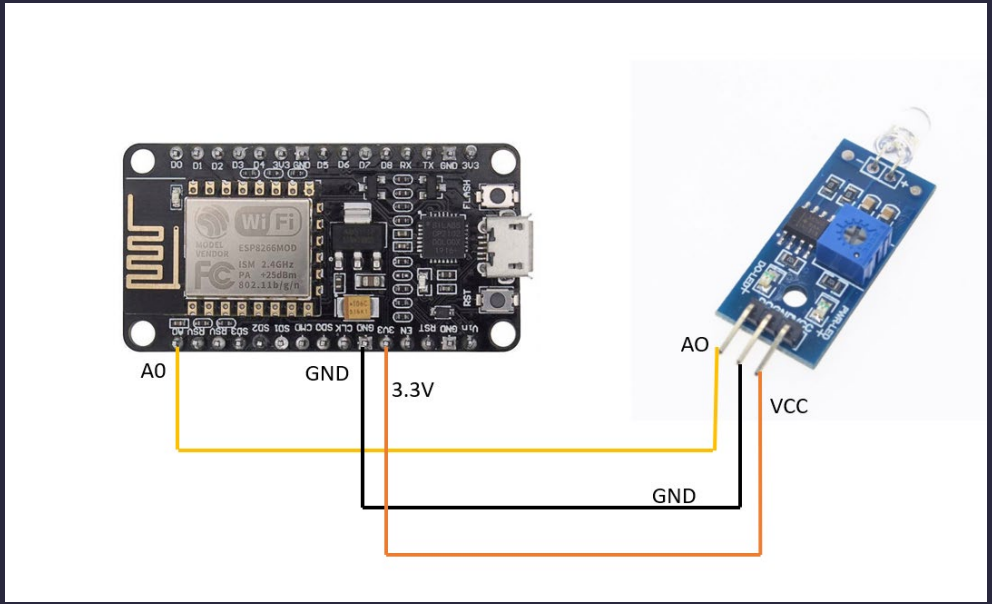
You can adjust the sensitivity of the LDR by using a potentiometer (VR) on the board. To operate, it requires an input voltage of 3.3-5V. The board typically includes LEDs to indicate power (PWR LED) and the signal level (D0 LED).

light sensor module LM393



The internal circuitry of this LDR module is such that when a supply voltage, VCC, and ground, GND, are applied to the board, the resistance of the LDR is converted into a voltage signal via a voltage divider circuit. This signal is then fed into the analog signal pin A0 for comparison. It's then compared with a reference signal generated by another voltage divider circuit, which is controlled by the adjustable resistor VR. An operational amplifier (LM393) is used to compare the signals and then sends the digital signal as D0 output. You can observe the D0 signal level via the D0 LED, which is connected to the output pin of the operational amplifier.

Wiring



LM393	ESP8266
VCC	3.3V
GND	GND
AO	A0

Coding

```
int analogPin = A0;
int val = 0;
void setup() {
  Serial.begin(9600);
}

void loop() {
  val = analogRead(analogPin);
  Serial.print("val = ");
  Serial.println(val);
  delay(500);
}
```

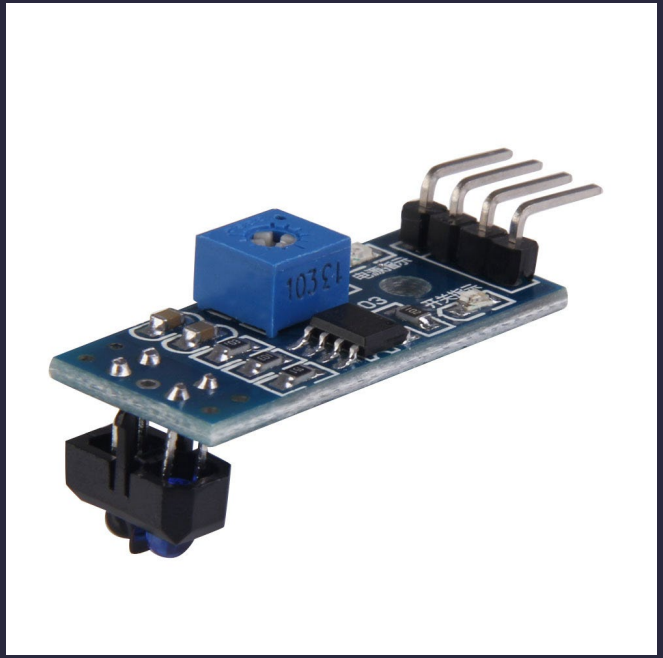
Test

```
14:38:28.162 -> val = 42
14:38:28.659 -> val = 61
14:38:29.158 -> val = 939
14:38:29.628 -> val = 944
14:38:30.127 -> val = 943
14:38:30.657 -> val = 943
14:38:31.155 -> val = 906
14:38:31.656 -> val = 52
14:38:32.124 -> val = 62
14:38:32.666 -> val = 52
14:38:33.163 -> val = 51
14:38:33.660 -> val = 777
14:38:34.169 -> val = 920
14:38:34.667 -> val = 920
14:38:35.167 -> val = 924
```

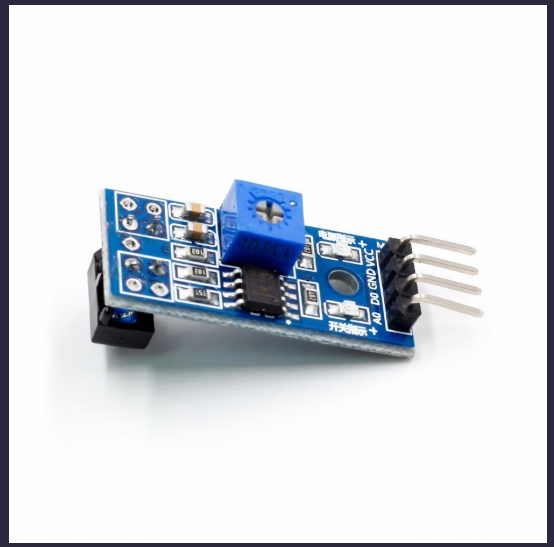
LAB:

/02

/IR Line Tracking TCRT5000

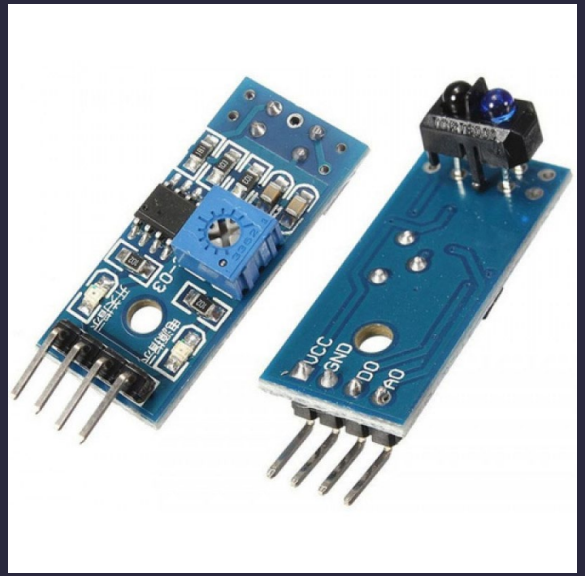


IR Line Tracking TCRT5000

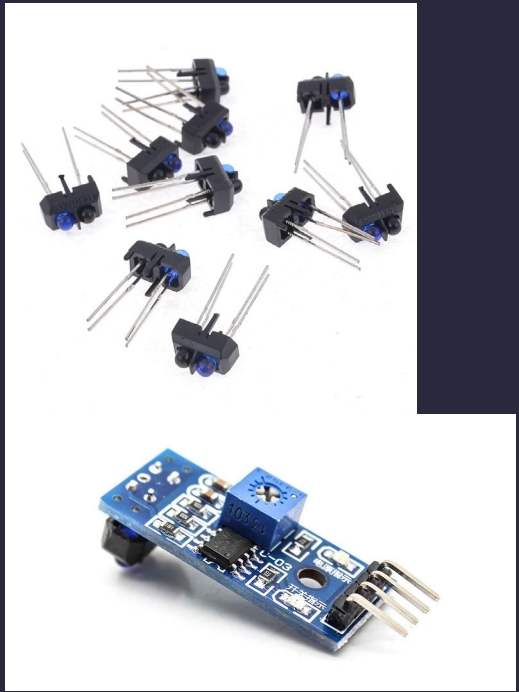


A line sensor is commonly used with line-following robots or obstacle-avoidance systems. Its operating principle involves emitting IR light, which reflects off objects. If there's a significant reflection, it results in a high intensity of light, indicating that it has detected a white line. Conversely, if there's little to no reflection (because black objects absorb light), it signifies that it has encountered a black line.

IR Line Tracking TCRT5000



This sensor typically operates on a voltage range of 3.3V to 5V and provides an output port that gives either a high or low signal. The signal is high when it detects a value above the predefined threshold set using the potentiometer on the board. It can also receive signals as analog values, which allows for more precise measurements.

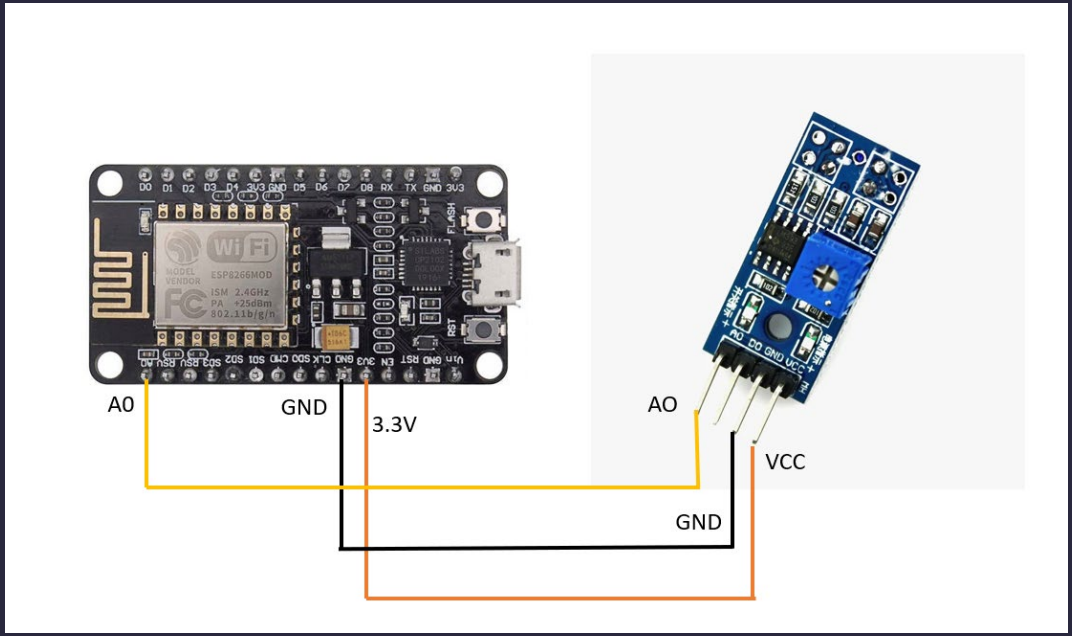


IR Line Tracking TCRT5000

Module functions is introduced:

- TCRT5000 sensor of infrared emission diode constantly launch infrared,
- the infrared ray is emitted when the intensity of reflected back has not been reflected or but is not big enough,
- the light activated triode has been in the off state,
- the output end of the module at this time for the high level,
- indicating diode has been put out state;
- detected objects appear within the scope of testing,
- infrared light is reflected back and strength is enough big,
- light activated triode saturation,
- at this time the output of the module for the low level, instructs the diode lights.

Wiring



TCRT5000	ESP8266
VCC	3.3V
GND	GND
AO	A0

Coding

```
int analogPin = A0;
int val = 0;
void setup() {
  Serial.begin(9600);
}

void loop() {
  val = analogRead(analogPin);
  Serial.print("val = ");
  Serial.println(val);
  delay(500);
}
```

Test

```
16:45:15.796 -> val = 778
16:45:16.289 -> val = 773
16:45:16.786 -> val = 766
16:45:17.287 -> val = 759
16:45:17.787 -> val = 727
16:45:18.286 -> val = 85
16:45:18.783 -> val = 63
16:45:19.282 -> val = 61
16:45:19.796 -> val = 60
16:45:20.294 -> val = 60
16:45:20.790 -> val = 53
16:45:21.261 -> val = 53
16:45:21.759 -> val = 56
16:45:22.258 -> val = 67
16:45:22.757 -> val = 762
16:45:23.256 -> val = 762
16:45:23.786 -> val = 54
16:45:24.283 -> val = 54
16:45:24.797 -> val = 54
16:45:25.298 -> val = 765
```

LAB:

Q/A