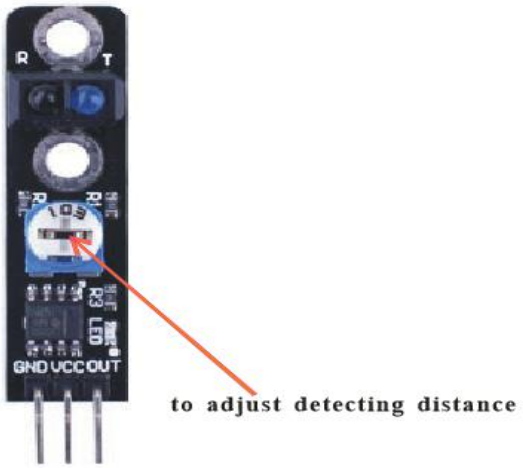
www.kumantech.com

Tracking Sensor



Overview

A tracking sensor combines an infrared emitter and receiver into a package that can detect the presence of dark areas on light surfaces. The emitter shines an (invisible) light which diffusely reflects back to the receiver when the illuminated surface is light, but does not reflect back when that surface is itself light-absorbing (i.e. dark).

In mobile robotics applications, tracking sensors allow robots to follow a painted track (the dark area) on the floor (the light surface), or to stay within some designated perimeter by detecting a dark "fence" painted around a light "work area." They are also used in automatic manufacturing contexts, where a moving object (such as a tape or assembly line) might track past a fixed sensor.

In this experiment, your Raspberry Pi will turn on and off an LED as the tracking sensor tracks onto, and off of, the dark area.

Experimental Materials

Raspberry Pi	x1
Breadboard	x1
Tracking Sensor	x1
LED (3 pin)	x1
Resistor (330Ω)	x1
Dupont jumper wires	

Experimental Procedure

1. If you have not done so already, prepare your development system by installing the Python interpreter, RPi.GPIO library, and wiringPi library as described in READ_ME_FIRST.TXT.

2. Install the tracking sensor and three-pin LED on your breadboard, and use the resistor and Dupont jumper wires as illustrated in the Wiring Diagram below. Note you will connect only two of the three pins on the LED.
3. Execute the sample stored in this experiment's subfolder.
   If using C, compile and execute the C code:

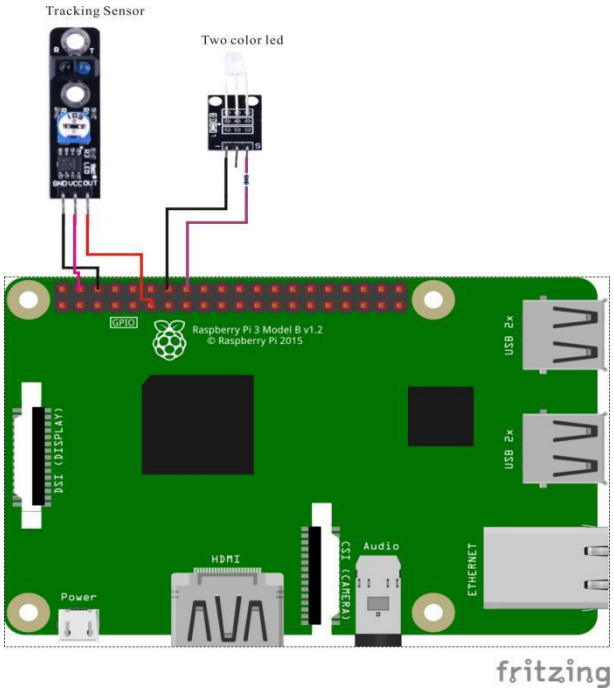
```
cd Code/C
gcc trackingSensor.c -o trackingSensor.out -lwiringPi
trackingSensor.out
```

   If using Python, launch the Python script:

```
cd Code/Python
python trackingSensor.py
```

4. Make experimental observations. Move the tracking sensor over and aware from a dark line on a bright surface. When the sensor tracks the line, the LED illuminates.

Wiring Diagram



Tracking Sensor pin position:

S	↔	Raspberry Pi pin 11
"+"	↔	Raspberry Pi +5V
"-"	↔	Raspberry Pi GND

LED pin position:

"S"	↔	Raspberry Pi pin 16 (through resistor)
"-"	↔	Raspberry Pi GND

Sample Code

Python Code

```
#!/usr/bin/env python
import RPi.GPIO as GPIO

TrackPin = 11
LedPin   = 16

def setup():
    GPIO.setmode(GPIO.BOARD)  # Numbers GPIOs by physical location
    GPIO.setup(LedPin, GPIO.OUT)   # Set LedPin's mode is output
    GPIO.setup(TrackPin, GPIO.IN, pull_up_down=GPIO.PUD_UP)

def loop():
    while True:
        if GPIO.input(TrackPin) == GPIO.LOW:
            GPIO.output(LedPin, GPIO.LOW)  # led on
        else:
            GPIO.output(LedPin, GPIO.HIGH) # led off

def destroy():
    GPIO.output(LedPin, GPIO.HIGH)     # led off
    GPIO.cleanup()                     # Release resource

if __name__ == '__main__':     # Program start from here

    setup()
```

```
    try:
        loop()
    except KeyboardInterrupt:
        destroy()
```

C Code

```
#include <wiringPi.h>
#include <stdio.h>

#define TrackSensorPin    0
#define LedPin           4

int main(void)
{
    if(wiringPiSetup() == -1)
    {
        printf("setup wiringPi failed !");
        return 1;
    }

    pinMode(TrackSensorPin, INPUT);
    pinMode(LedPin,  OUTPUT);

    while(1)
    {
        if(digitalRead(TrackSensorPin) == LOW)
        {
            digitalWrite(LedPin, LOW);
            delay(100);
        }
        else
        {
            digitalWrite(LedPin, HIGH);
            delay(100);
        }
    }

    return 0;
}
```