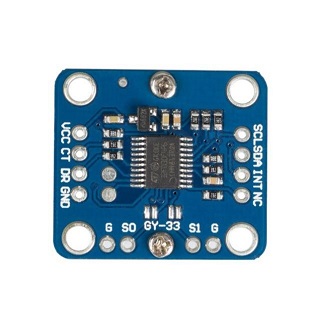
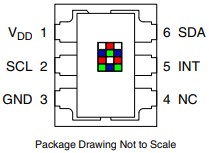
GY-33 Color Recognition Sensor Module

## Introduction

GY-33 color recognition sensor module has these advantages: low cost, small volume, easy to mount, low-power consumption and its working voltage is 3~5V.  
[](http://wiki.sunfounder.cc/index.php?title=File:GY-33_01.jpg) [](http://wiki.sunfounder.cc/index.php?title=File:GY-33_02.jpg)  
**TCS347525**  
[](http://wiki.sunfounder.cc/index.php?title=File:GY-33_03.jpg)  
The TCS3472 device provides a digital return of red, green, blue (RGB), and clear light sensing values. An IR blocking filter, integrated on-chip and localized to the color sensing photodiodes, minimizes the IR spectral component of the incoming light and allows color measurements to be made accurately. The high sensitivity, wide dynamic range, and IR blocking filter make the TCS3472 an ideal color sensor solution for use under varying lighting conditions and through attenuating materials.

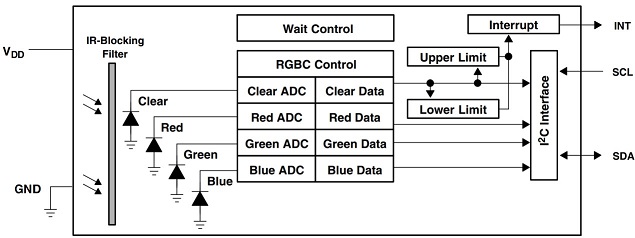
## Features

1. Measure Range: RGB 0~255;
2. Frequency: 10 Hz;
3. Working Voltage: 3~5V;
4. Working Electric Current: 15mA;
5. Working Temperature: -20° ~ 85°;
6. Storage Temperature: -40° ~ 125°;
7. Size: 24.3mm x 26.7mm;
8. Sensor Chip: TCS34725.

## Introduction of Pins

|  |  |
| --- | --- |
| **Pin Introduction** | |
| GND | Connected to the cathode of the power supply |
| DR | Serial port UART\_RX/IIC\_SDA |
| VCC | Connected to the anode of the power supply(3v-5v) |
| NC | Remain, do not connect to anything |
| INT | TCS34725 color chip breaks off s1=0 (wiring up GND, run it) |
| SDA | TCS34725 color chip data wire s1=0(wiring up GND, run it) |
| SCL | TCS34725 clock line of color chip s1 = 0 (wiring up GND, run it) |
| S0 | Serial/MCU\_IIC Mode Choose |
| S1 | Only let sensor chip choose |

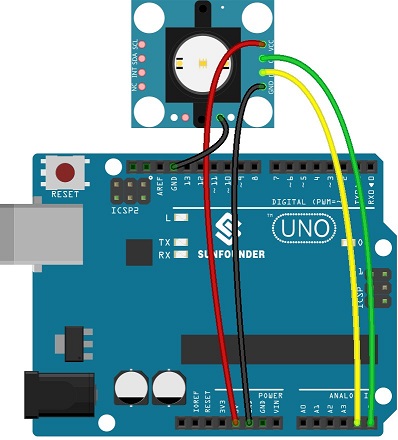
## Principle

LED lights up and illuminates the detected objects. The reflected lights detect the ratio of RGB via the filter and recognize the colors according to the RGB ratio. **TCS347525** The TcS34725 light-to-digital converter contains a 3 X 4 photodiode array, four analog-to-digital converters (ADC) that integrate the photodiode current, data registers, a state machine, and an I2C interface. THE 3 X 4photodiode array is composed of red-filtered, green-filtered, blue-filtered, and clear (unfiltered) photodiodes. In addition, the photodiodes currents use a 16-bit digital value. Upon completion of a conversion cycle, the results are transferred to the data registers, which are double-buffered to ensure the integrity of the data. All of the internal timing, as well as the low-power wait state, is controlled by the state machine.  
[](http://wiki.sunfounder.cc/index.php?title=File:GY-33_04.jpg)

## Experimental Procedures for Arduino

**Step 1:**Connect the circuit:

|  |  |
| --- | --- |
| **GY-33** | **Uno Board** |
| VCC | VCC |
| CT | A5 |
| DR | A4 |
| GND | GND |
| SO | GND |

[](http://wiki.sunfounder.cc/index.php?title=File:GY-33_05.jpg)

**Step 2:** Compile and upload the code.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <i2cmaster.h>

#define uint16\_t unsigned int

typedef struct

{

uint16\_t Red;

uint16\_t Green;

uint16\_t Blue;

uint16\_t Clear;

} RGB;

unsigned char Re\_buf;

unsigned char sign=0;

RGB rgb;

uint16\_t CT=0,Lux=0;

byte color=0,rgb\_data[3]={0};

void setup() {

Serial.begin(115200);

i2c\_init();

delay(1);

}

void loop() {

unsigned char data[9]={0};

if(!sign)

{

iic\_read(0x00,data,8);

rgb.Red=(data[0]<<8)|data[1];

rgb.Green=(data[2]<<8)|data[3];

rgb.Blue=(data[4]<<8)|data[5];

rgb.Clear=(data[6]<<8)|data[7];

Serial.print("Red: ");

Serial.print(rgb.Red);

Serial.print(",Green: ");

Serial.print( rgb.Green);

Serial.print(",Blue");

Serial.print( rgb.Blue);

Serial.print(",Clear");

Serial.println(rgb.Clear);

iic\_read(0x08,data,4);

Lux=(data[0]<<8)|data[1];

CT=(data[2]<<8)|data[3];

Serial.print("CT:");

Serial.print(CT);

Serial.print(",Lux:");

Serial.println( Lux);

iic\_read(0x0c,data,3);

rgb\_data[0]=data[0];

rgb\_data[1]=data[1];

rgb\_data[2]=data[2];

Serial.print("r:");

Serial.print( rgb\_data[0]);

Serial.print(",g:");

Serial.print( rgb\_data[1]);

Serial.print(",b:");

Serial.println( rgb\_data[2]);

iic\_read(0x0f,data,1);

color=data[0];

Serial.print(",color:");

Serial.println( color,HEX);

}

if(sign==1)

{

iic\_read(0x10,&data[8],1);

i2c\_start\_wait(0xb4);

i2c\_write(0x10);

i2c\_write(0x31);

// i2c\_write((data[8]|0x01));

i2c\_stop();

sign=3;

}

delay(200);

}

void iic\_read(unsigned char add,unsigned char \*data,unsigned char len)

{

i2c\_start\_wait(0xb4);

i2c\_write(add);

i2c\_start\_wait(0xb5);

while(len-1)

{

\*data++=i2c\_readAck();

len--;

}

\*data=i2c\_readNak();

i2c\_stop();

}

void serialEvent() {

while (Serial.available()) {

Re\_buf=(unsigned char)Serial.read();

if (Re\_buf=='a')

sign=0;

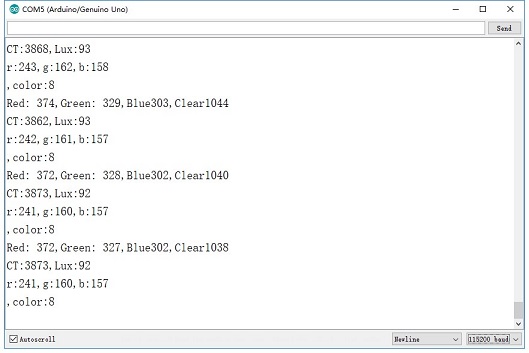
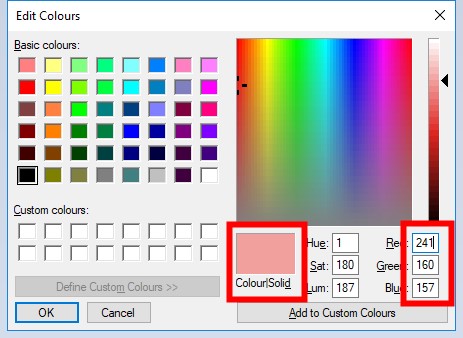
if (Re\_buf=='b')

sign=1;

Re\_buf=0;

}

}

[](http://wiki.sunfounder.cc/index.php?title=File:GY-33_06.jpg)  
Open mspaint and edit colors of motors. Input the values of r, g, b into serial port.  
[](http://wiki.sunfounder.cc/index.php?title=File:GY-33_07.jpg)