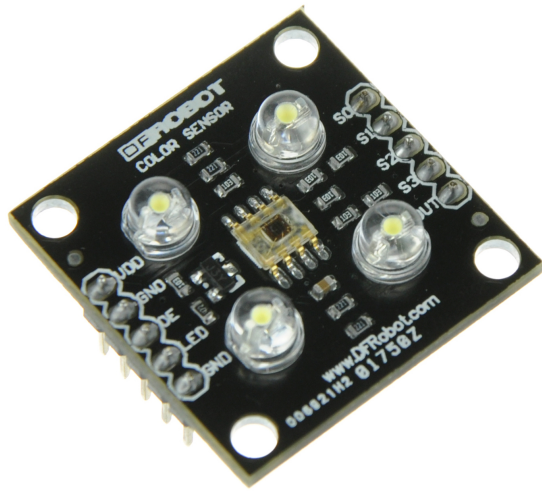


SKU:SEN0101 (<https://www.dfrobot.com/product-540.html>)



(<https://www.dfrobot.com/product-540.html>)

Introduction

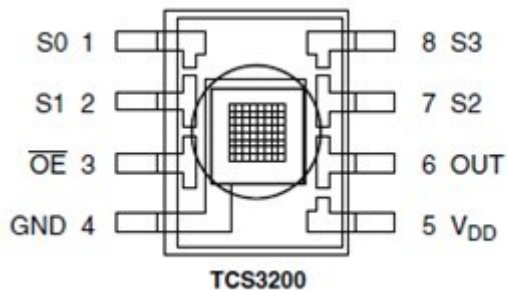
TCS3200 GBB Color Sensor For Arduino (<https://www.dfrobot.com/product-540.html>) is a complete color detector, including a TAOS TCS3200 RGB sensor chip and 4 white LEDs. The TCS3200 can detect and measure a nearly limitless range of visible colors. Applications include test strip reading, sorting by color, ambient light sensing and calibration, and color matching, to name just a few.

The TCS3200 GBB Color Sensor For Arduino has an array of photo detectors, each with either a red, green, or blue filter, or no filter (clear). The filters of each color are distributed evenly throughout the array to eliminate location bias among the colors. Internal to the device is an oscillator which produces a square-wave output whose frequency is proportional to the intensity of the chosen color.

Specifications

- Single-Supply Operation (2.7V to 5.5V)
- High-Resolution Conversion of Light Intensity to Frequency
- Programmable Color and Full-Scale Output Frequency
- Power Down Feature
- Communicates Directly to Microcontroller
- S0~S1: Output frequency scaling selection inputs
- S2~S3: Photodiode type selection inputs
- OUT Pin: Output frequency
- OE Pin: Output frequency enable pin (active low), can be impeding when using
- Support LED lamp light supplement control
- Size: 28.4x28.4mm

PinOut



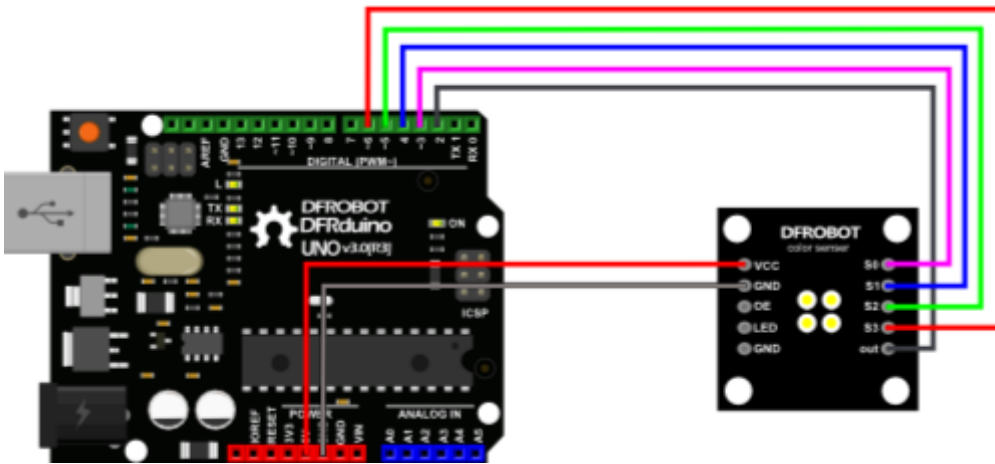
Pin Name	I/O	DESCRIPTION
GND(4)	GND	Power supply ground. All voltages are referenced to GND
OE(3)	I	Enable for fo (active low)

Pin Name	IO	DESCRIPTION
S0, S1 (1, 2)	I	Output frequency scaling selection inputs

S2, S3 (7, 8)	I	Photodiode type selection inputs
VDD (5)	VDD	Supply voltage

Tutorial

Connection Diagram

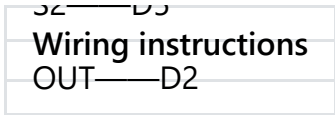


Wiring instructions

VCC — 5V

S0 — D3

S2 — D7



S0, S1, S2, S3

To TCS3002D, when choose a color filter, it can allow only one particular color to get through and prevent other color. For example, when choose the red filter, Only red incident light can get through, blue and green will be prevented. So we can get the red light intensity. Similarly ,when choose other filters we can get blue or green light.

TCS3002D has four photodiode types. Red , blue, green and clear, reducing the amplitude of the incident light uniformity greatly, so that to increase the accuracy and simplify the optical. When the light project to the TCS3002D we can choose the different type of photodiode by different combinations of S2 and S3. Look at the form as follows.

S0	S1	OUTPUT FREQUENCY SCALING (fo)
L	L	Power down
L	H	2%
H	L	20%
H	H	100%

TCS3002D can output the frequency of different square wave (occupies emptiescompared 50%),different color and light intensity correspond with different frequency of square wave. There is a relationship between the output and light intensity. The range of the typical output frequency is 2HZ~500KHZ. We can get different scaling factor by different combinations of S0 and S1. Look at the form as follows.

S2	S3	PHOTODIODE TYPE
L	L	RED
L	H	BLUE

S2	S3	PHOTODIODE TYPE
H	H	GREEN

Sample Code

```

int s0=3,s1=4,s2=5,s3=6;
int out=2;
int flag=0;
byte counter=0;
byte countR=0,countG=0,countB=0;
void setup()
{
  Serial.begin(115200);
  pinMode(s0,OUTPUT);
  pinMode(s1,OUTPUT);
  pinMode(s2,OUTPUT);
  pinMode(s3,OUTPUT);

}
void TCS()
{
  flag=0;
  digitalWrite(s1,HIGH);
  digitalWrite(s0,HIGH);
  digitalWrite(s2,LOW);
  digitalWrite(s3,LOW);
  attachInterrupt(0, ISR_INT0, CHANGE);
  timer0_init();

}
void ISR_INT0()
{
  counter++;
}
void timer0_init(void)
{
  TCCR0A = 0x00;

```

```

    TCCR2A=0x00;
    TCCR2B=0x07; //the clock frequency source 1024 points
    TCNT2= 100; //10 ms overflow again
    TIMSK2 = 0x01; //allow interrupt
}

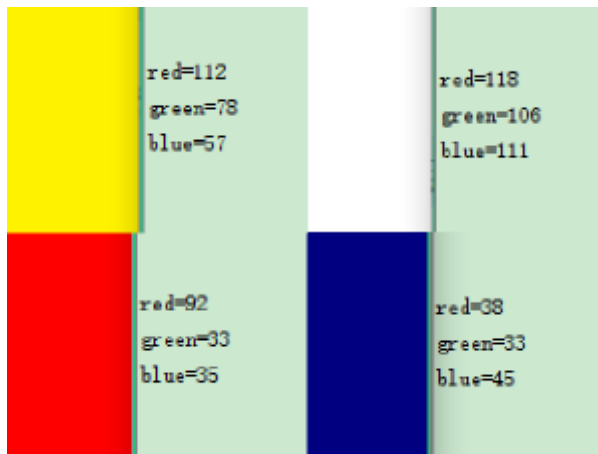
int i=0;
ISR(TIMER2_OVF_vect)//the timer 2, 10ms interrupt overflow again. Internal overflow interrupt executive function
{
    TCNT2=100;
    flag++;
    if(flag==1)
    {
        countR=counter;
        Serial.print("red=");
        Serial.println(countR,DEC);
        digitalWrite(s2,HIGH);
        digitalWrite(s3,HIGH);
    }
    else if(flag==2)
    {
        countG=counter;
        Serial.print("green=");
        Serial.println(countG,DEC);
        digitalWrite(s2,LOW);
        digitalWrite(s3,HIGH);
    }
    else if(flag==3)
    {
        countB=counter;
        Serial.print("blue=");
        Serial.println(countB,DEC);
        Serial.println("\n");
        digitalWrite(s2,LOW);
        digitalWrite(s3,LOW);
    }
    else if(flag==4)

```

```
{
  flag=0;
}
  counter=0;
}

void loop()
{
  TCS();
  while(1);
}
```

Result




Data exists deviation, you need to adjust the program according to environmental parameters

Trouble shooting

More question and cool idea, visit DFRobot Forum (<https://www.dfrobot.com/index.php?route=DFblog/blogs>)

More

- TCS3200 Schematic (<https://www.dfrobot.com/image/data/SEN0101/TCS3200%20Sch.pdf>)
- TCS3200 Data Sheet (<https://www.dfrobot.com/image/data/SEN0101/TCS3200%20TCS3210.pdf>)

 Click to buy TCS3200 GBB Color Sensor For Arduino (<https://www.dfrobot.com/product-540.html>) or DFRobot Distributor List (<https://www.dfrobot.com/index.php?route=information/distributorslogo>)

Category: DFRobot (<https://www.dfrobot.com/>) > Sensors & Modules (<https://www.dfrobot.com/category-156.html>) > Sensors (<https://www.dfrobot.com/category-36.html>) > Light & Imaging Sensors (<https://www.dfrobot.com/category-58.html>) category: Product_Manual (category__Product_Manual) category: DFR_Series (category__DFR_Series) category: Sensors (category__Sensors) category:source (category_source) category:Diagram (category_Diagram)