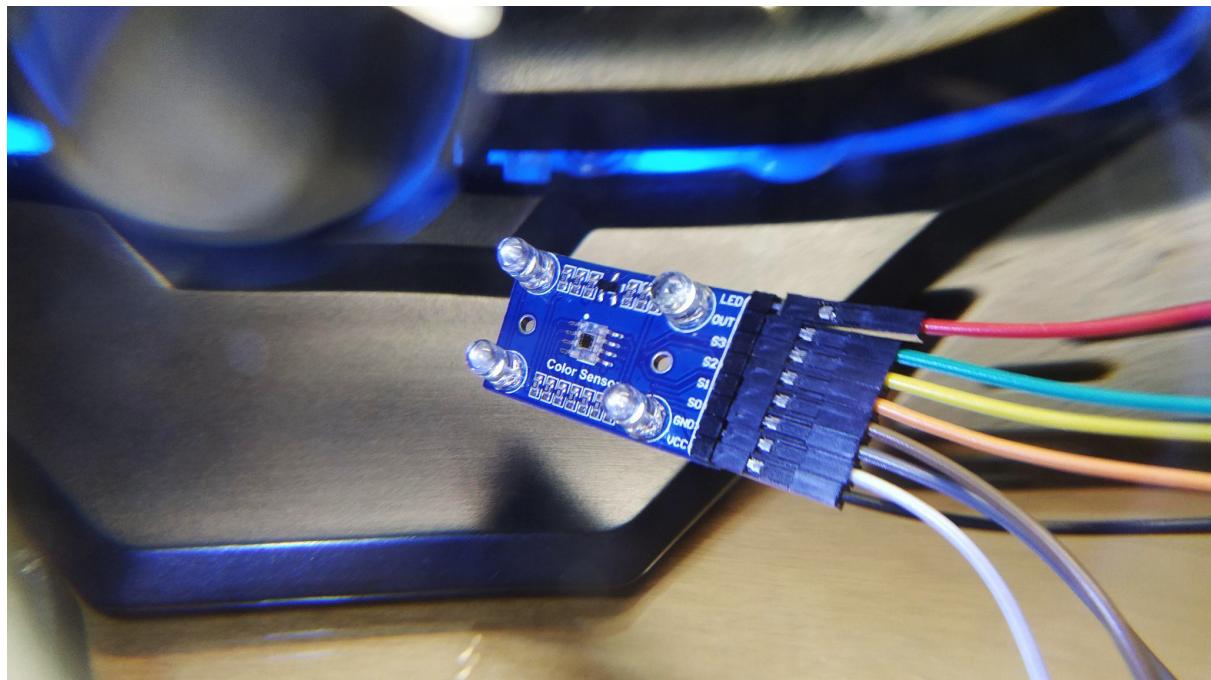


Arduino and color sensor (TCS3200)



Εφαρμογές Τηλεπικοινωνιακών Διατάξεων

Γιαχούδης Χρίστος
ΑΕΜ:

Νομικός Γιώργος
ΑΕΜ: 9857
gnomikos@ece.auth.gr

TABLE OF CONTENTS

| | |
|------------------------------------|-----------|
| Introduction | 3 |
| Schematic | 5 |
| Hardware connections | 6 |
| Filter selection | 7 |
| Frequency scaling | 8 |
| Code | 9 |
| Desirable results | 10 |
| Findings and problems | 10 |
| References and useful links | 11 |

Introduction



Fig 1. TCS3200 sensor

The **TCS3200** has a **grid of photodiodes** with 4 different filters.

Note that a **photodiode** is simply a semiconductor device that **converts light into current**.

The sensor has:

- 16 photodiodes with red filter – sensitive to red wavelength
- 16 photodiodes with green filter – sensitive to green wavelength
- 16 photodiodes with blue filter – sensitive to blue wavelength
- 16 photodiodes without filter

The sensor has a current-to-frequency converter that converts the photodiodes' readings into a square wave with a frequency that is proportional to the light intensity of the chosen color. This frequency is, then, read by the Arduino, as shown in Fig. 2.

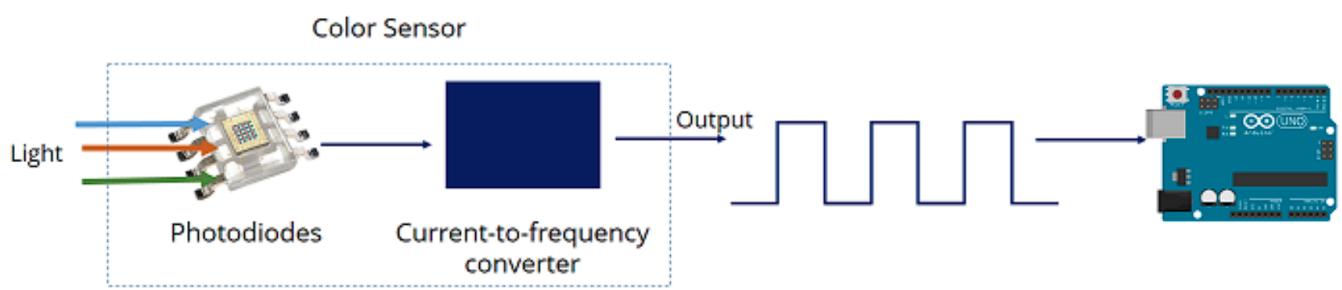


Fig 2. General description of how the sensor works

Schematic

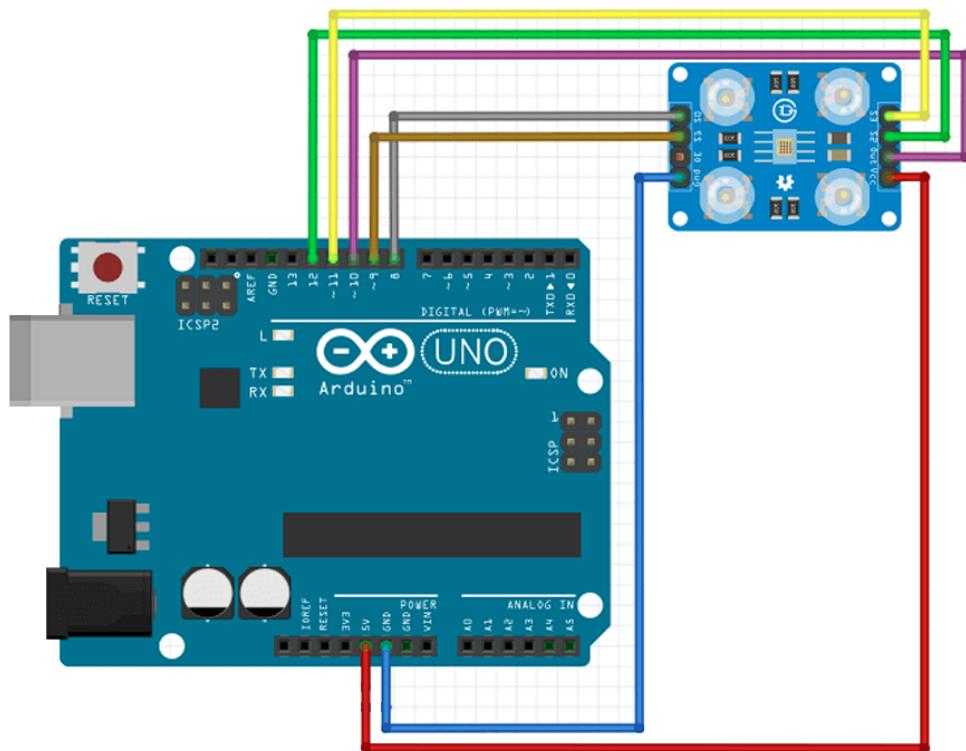


Fig 3. Arduino - TCS3200 connection schematic

Hardware connections

Here's the connections between the TCSP3200 and the Arduino:

- **S0**: digital pin 8
- **S1**: digital pin 9
- **VCC**: power pin 5V
- **S2**: digital pin 12
- **S3**: digital pin 11
- **OUT**: digital pin 10
- **GND**: power pin GND
- **LED**: power pin 3.3V

All the above connections are being achieved with male-to-female jumper wires, which are shown in fig. 4.



Fig 4. Male-to-female jumper wires

Also, a connection with a pc is needed, using a usb-A to usb-B cable.

Filter selection

To select the color read by the photodiode, you use the control pins S2 and S3. As the photodiodes are connected in parallel, setting the S2 and S3 in different states (LOW and HIGH) in different combinations allows you to select different photodiodes(red, green, blue, neutral). Take a look at the table below:

| Photodiode type | S2 | S3 |
|-------------------|------|------|
| Red | LOW | LOW |
| Blue | LOW | HIGH |
| No filter (clear) | HIGH | LOW |
| Green | HIGH | HIGH |

Frequency scaling

Pins S0 and S1 are used for scaling the output frequency. It can be scaled to the following preset values: 100%, 20% or 2%. Scaling the output frequency is useful to optimize the sensor readings for various frequency counters or microcontrollers. Take a look at the table below:

| Output frequency scaling | S0 | S1 |
|--------------------------|----|----|
| Power down | L | L |
| 2% | L | H |
| 20% | H | L |
| 100% | H | H |

Code

The code file of the project can be found at the following address:

<https://github.com/giahou2000/Arduino-color-sensor.git>

Useful notes:

- 1) We choose 100% Frequency scaling, by setting:

S0 = HIGH

S1 = HIGH

- 2) pulseIn() function:

<https://referencearduino.cc/reference/en/language/functions/advanced-io/pulsein/>

Desirable results

It is logical, that when you turn the sensor towards a red (green, blue) surface, the arduino will print at the serial monitor that the detected color is red (green, blue).

```
Red Color
R = 4
G = 6
B = 5

Red Color
R = 4
G = 6
B = 5

Red Color
R = 4
G = 6
B = 5

Red Color
R = 16
G = 22
B = 19
```

Fig 5. Serial Monitor

In case we want to detect more colors, a more sophisticated implementation is needed (different code and maybe different sensor).

Findings and problems

- The results are affected by the surrounding light.
- The results are affected by the distance between the sensor and the object.
- The TCS3200 sensor has limited capabilities. For complex scenarios of color recognition.

References and useful links

- <https://randomnerdtutorials.com/arduino-color-sensor-tcs230-tcs3200/>
- <https://tutorial45.com/arduino-color-sensor-arduino/>
- <https://grobotronics.com/color-sensor-tcs3200.html>
- <https://github.com/giahou2000/Arduino-color-sensor.git>