International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 3 Issue 5, August 2019 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Colour Detector and Separator Based on Microcontroller

Ni Ni San Hlaing[1], Hay Man Oo[2], Thin Thin Oo[2]

[1]Lecturer, Department of Electronic Engineering, Technological University, Kyaukse, Myanmar
[2]Lecturer, Department of Electronic Engineering, Technological University, Meiktila, Myanmar

How to cite this paper: Ni Ni San Hlaing | Hay Man Oo | Thin Thin Oo "Colour Detector and Separator Based on Microcontroller" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-3 | Issue-5, August 2019, pp.1103-1108, **https://doi.org/10.31142/ijtsrd26633**

IJTSRD26633

ABSTRACT

Sorting of objects is one of the most important tasks in the industries. Sorting machines can be made by using the sensor, camera, microcontroller and image processing system, etc., based on colour, size and shape of the objects. Among these systems, colour sorting machines are widely used in most industries to sort various coloured objects. This thesis describes the design and implementation of colour detection and sorting machine by using Arduino, colour sensor and two servo motors. In this system, a top servo motor is used for placing the object to the colour sensor which detects the RGB values of the object and then sends the information to Arduino. Then bottom servo motor will move according to the angle of respective degree which is command in the code depending on the RGB values of the object and then the object which is over the slider drops into the container. Servo motor is used to substitute the human operator and Arduino is used to controlling the overall process.

KEYWORDS: *Arduino UNO, Servo motor, TCS3200 Colour Sensor, Ball, C programming, Power Supply*

I. INTRODUCTION

Colour sorting machine is a device that can sense the different colours of the objects and place objects into a specific destination based on the colours of the objects by using the colour sensor, camera, etc. The purpose of Arduino based colour detection and the sorting machine is to design and implement a system which automatically sorts objects based on the colours of objects.

Arduino UNO, colour sensor and two servo motors are used in this thesis. The top servo motor is for placing the objects to the colour sensor and then the colour sensor detects the RGB values of the object. The slider which is attached on the bottom servo motor rotates according to the angle of respective degree which is command in the code depending on the RGB values. Then, the object which is over the slider drops into the container. Colour detection techniques have been reported using a variety of methods. The main subject is to implement a system for colour recognition technique by using Arduino C programming. Colour recognition techniques are used for sorting objects. To implement this system, Arduino C programming language, Arduino IDE software is mainly applied.

II. System Block Diagram

Arduino based colour detection and the sorting machine are divided into two steps: colour detection and colour sorting steps. In the colour detection step, TCS3200 colour sensor senses colour light with the help of photodiodes arrays. Then, using a current to frequency converter the readings from the photodiodes are converted into a square wave with frequency directly proportional to the light intensity. The input for the Arduino board is this frequency and the RGB values of the object can be read on the serial monitor. In the colour sorting step, the balls which are held in the charger drop into the platform attached on the top servo motor. The top servothe motor rotates and brings the ball to the colour sensor. After that, the bottom servo motor rotates according to the degree which is command in the code. Then, the ball drops from a slide bar, which is attached on the shaft of the bottom servo motor, to the respective container. After all, the top servo motor rotates again to carry the ball. Figure 1. shows the block diagram of colour detection and sorting machine.

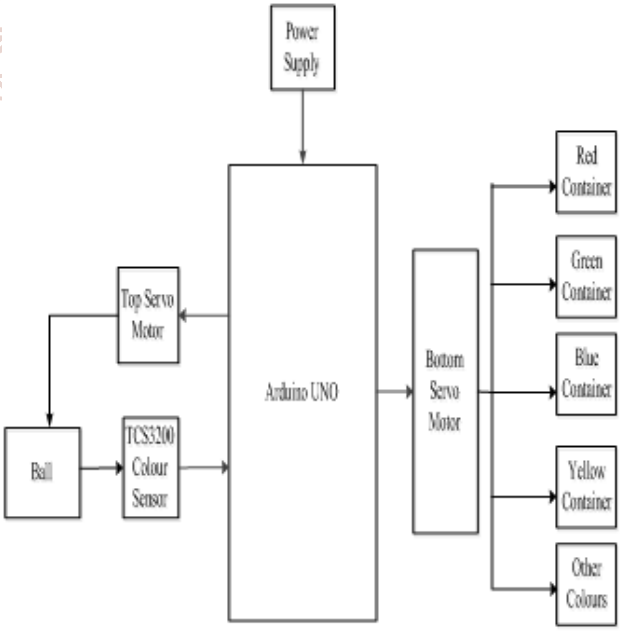


Fig1: Overall Block Diagram of Colour Detector and Separator Based on Microcontroller

III. Implementation

Colour sorting system is made with various techniques and different sensors. In this system, the colour detection step is needed to know the RGB values of the objects which are going to use in colour sorting machine. In this thesis, the TCS3200 colour sensor is used to detect the colour of the object with the help of photodiode arrays. Then, using a current to frequency converter the readings from the photodiodes are converted into a square wave with frequency directly proportional to the light intensity. The input for the Arduino board is this frequency and the RGB values of the object can be read on the serial monitor. Figure 4.1 shows the circuit diagram of colour detection step and Table 4.2 shows the pin connection of colour detection step.

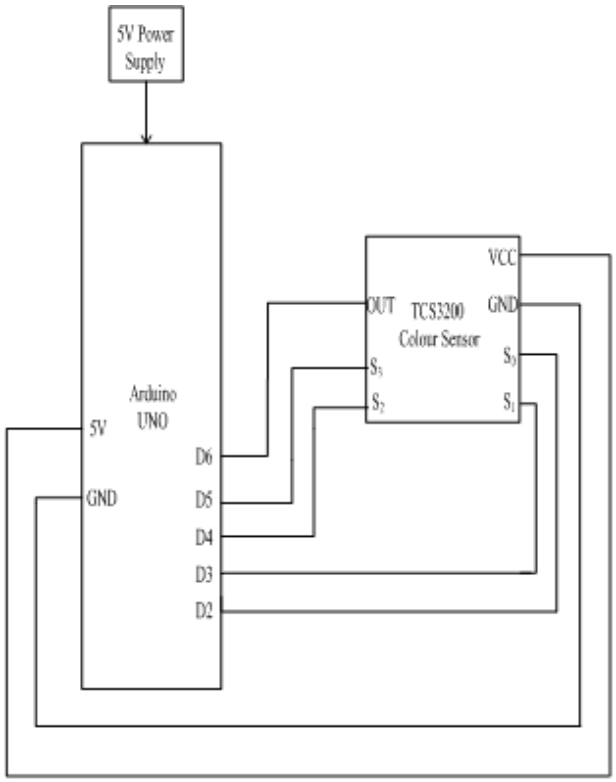


Fig 2: Overall Circuit Diagram of the system

After detecting the colour of the balls, colour sorting system can be started. Initially, the balls which are held in the charger drop into the platform attached on the top servo motor. Top servo motor is used for placing the balls to the colour sensor which detects the RGB values of the ball. Then, the bottom servo motor will move according to the angle of respective degree which is command in the code depending on the RGB values of the ball. And then, the ball drops from slider to the container. After all, the top servo motor rotates again to carry the ball till the ball drops to the guide rail. This machine sorts four colour balls: Red, Green, Blue and Yellow balls to the respective containers and if other colours are detected, this machine sorts other colours balls to other colours container.

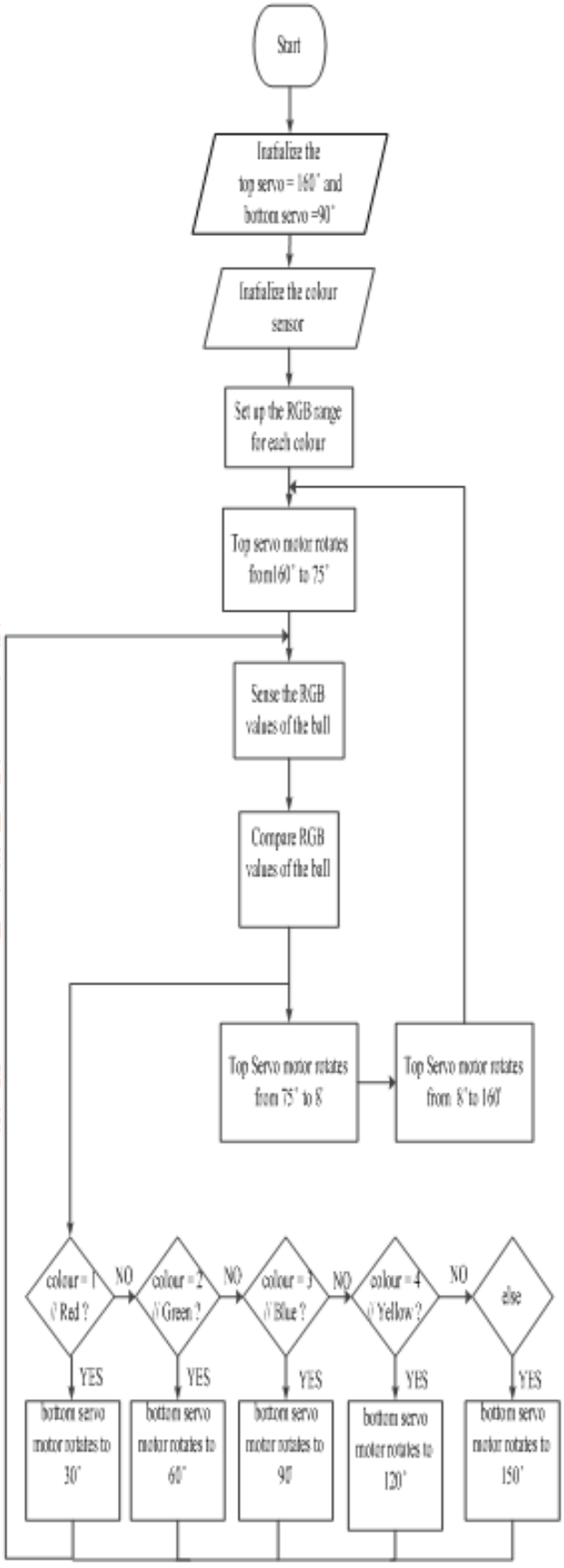


Fig:3 Flowchart of Colour Detector and Separator Based on Microcontroller

A. Working Principle of S_0, S_1, S_2 and S_3 Pins of Colour Sensor

The colour which needs to be sensed by the colour sensor is selected by two pins S_2 and S_3. With these two pins, logic control which colour light intensity is to be measured can be told. If Red colour intensity wanted to be sensed, both S_2 and S_3 pins set to LOW. By setting S_0 to LOW and S_1 to HIGH, Blue colour intensity can be sensed. By setting S_0 to HIGH and S_1 to HIGH, Green colour intensity can be sensed. Once that is done the sensor detects the intensity and sends the value to the control system inside this module. The light intensity measured by the array is sent to current to frequency converter that produces a square wave. The signal frequency sent by the module can be modulated depending on use. Output signal frequency bandwidth can be changed by using S_0 and S_1 pins. For convenience, frequency scaling set to 2% in this thesis. This is done by setting S_0 to LOW and S_1 to HIGH. By setting S_0 to HIGH and S_1 to LOW, frequency scaling can be changed to 20% and by setting S_0 to HIGH and S_1 to HIGH, frequency scaling can be changed to 100%. If frequency scaling set to 2%, the RGB values of the object can be read on the serial monitor as shown in Figure 4.

```
#define sensorOut 6
Servo topServo;
Servo bottomServo;
int frequency = 0;
int color=0;
void setup() {
  pinMode(S0, OUTPUT);
  pinMode(S1, OUTPUT);
  pinMode(S2, OUTPUT);
  pinMode(S3, OUTPUT);
  pinMode(sensorOut, INPUT);
  // Setting frequency-scaling to 2%
  digitalWrite(S0, LOW);
  digitalWrite(S1, HIGH);
  topServo.attach(7);
  bottomServo.attach(12);
  Serial.begin(9600);
}
```

```
COM3 (Arduino/Genuino Uno)
R= 1315   G= 2092   B= 1550
R= 1315   G= 2092   B= 1539
R= 1295   G= 2093   B= 1550
R= 1315   G= 2092   B= 1551
R= 1316   G= 2094   B= 1550
R= 1309   G= 2079   B= 1557
R= 1310   G= 2093   B= 1549
R= 1309   G= 2079   B= 1550
R= 1309   G= 2092   B= 1538
Autoscroll
```

Fig 4. Reading RGB Values by Setting Frequency Scaling to 2%

If frequency scaling set to 20%, the RGB values of the object can be read on the serial monitor as shown in Figure 5.

```
#define sensorOut 6
Servo topServo;
Servo bottomServo;
int frequency = 0;
int color=0;
void setup() {
  pinMode(S0, OUTPUT);
  pinMode(S1, OUTPUT);
  pinMode(S2, OUTPUT);
  pinMode(S3, OUTPUT);
  pinMode(sensorOut, INPUT);
  // Setting frequency-scaling to 2%
  digitalWrite(S0, LOW);
  digitalWrite(S1, HIGH);
  topServo.attach(7);
  bottomServo.attach(12);
  Serial.begin(9600);
}
```

```
COM3 (Arduino/Genuino Uno)
R= 1315   G= 2092   B= 1550
R= 1315   G= 2092   B= 1539
R= 1295   G= 2093   B= 1550
R= 1315   G= 2092   B= 1551
R= 1316   G= 2094   B= 1550
R= 1309   G= 2079   B= 1557
R= 1310   G= 2093   B= 1549
R= 1309   G= 2079   B= 1550
R= 1309   G= 2092   B= 1538
Autoscroll
```

Fig:5. Reading RGB Values by Setting Frequency Scaling to 20%

If frequency scaling set to 100%, the RGB values of the object can be read on the serial monitor as shown in Figure 6.

```
#define sensorOut 6
Servo topServo;
Servo bottomServo;
int frequency = 0;
int color=0;
void setup() {
  pinMode(S0, OUTPUT);
  pinMode(S1, OUTPUT);
  pinMode(S2, OUTPUT);
  pinMode(S3, OUTPUT);
  pinMode(sensorOut, INPUT);
  // Setting frequency-scaling to 2%
  digitalWrite(S0, LOW);
  digitalWrite(S1, HIGH);
  topServo.attach(7);
  bottomServo.attach(12);
  Serial.begin(9600);
}
```

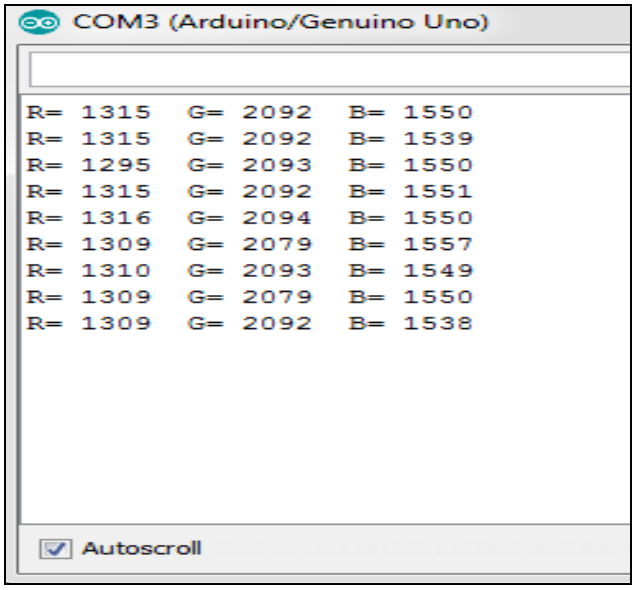


Fig5. Reading RGB Values by Setting Frequency Scaling to 100%

B. Studying Colour Data

Wire the sensor and Arduino. Load the code attached to Arduino board. Open the serial monitor. Now bring the ball in front of the sensor. The RGB values of the object can be seen on the serial monitor. These values have to write down for the colour which is going to use in colour detection and sorting machine. For example, when a red ball is placed in front of sensor R=1300, G=2000 and B=1700 is got. Note that by using colour sensor RGB values of the object vary depends on the surrounding light intensity. And ±10% of at least and at most values of the sensing RGB values is used to compensate fluctuate sensor values.

Even assuming at least and at most values of the sensing RGB values used in the colour sorting sketch, the operation of the machine is not always true. Because the sensing RGB values of the ball is changing if this machine is placed from one place to another. To reduce these errors, external light source is applied near the sensor and black background is applied around the sensor. In this way, the detected RGB values of the balls add in the sketch at this place:

```
if(R<1900 & R>1000 & G<3200 & G>1900 & B<2300 & B>1200) {colour = 1;// Red }
else if(R<2800 & R>2000 &G<2450 &G>1600& B<2500 &B>1700){colour=2; // Green}
else if (R<3300 & R>2100 & G<3000&G>1600& B<1900&B>1190){colour=3; // Blue }
else if (R<1200 &R>500&G<1400 &G>600& B<2020 &B>800){colour = 4; // Yellow}
```

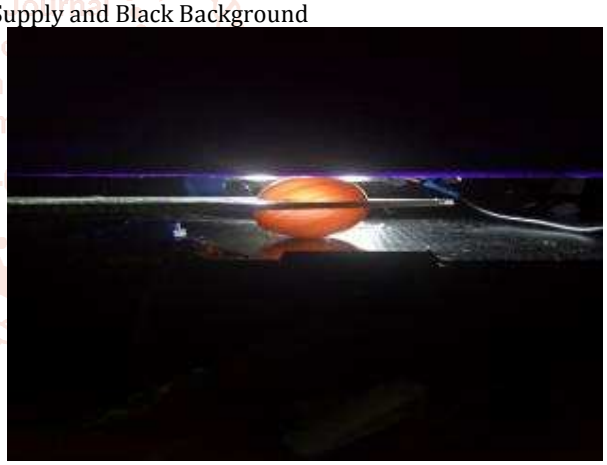
C. Code Explanation of Colour Sorting Machine Sketch

"Servo.h" library is needed to be included, define the pins to which the colour sensor will be connected, create the servo objects and declare some variables needed for the program. In the void setup section, outputs and inputs pins are declared, set the frequency scaling for the colour sensor, define the servo pins and start the serial communication for printing the results of the colour read on the serial monitor. In the loop section, the program starts with moving the top servo motor to the position of the charger. Next using the "for" loop servo motor rotates and brings the coloured ball to the position of the colour sensor. "for" loop is used to control the speed of the rotation by changing the delay time in a loop. Next, after half a second delay, using custom made function "readColour ()", the colour of the ball can be read. Using four control pins and frequency output pin of colour sensor and read colour of the ball. The sensor sense three different values of each ball Red, Green and Blue. Note that these values can vary because the sensor is not always accurate. Therefore, using these "if" statements, "if else" statements and else statements by allowing the sensor an error of around ±10% of the tested value for the particular colour. So for example, if red ball is detected, the first "if" statement will be true and the variable "colour" get the value 1. So that's what the readColour() custom function does and after that using a "switch-case" statement the bottom servo motor rotates to the particular position. In the end, top servo motor rotates again until the ball drops into guide rail and again sends back to the initial position so that the process can repeat.

IV. Results

Arduino based colour detection and sorting machine very depends on surrounding light intensity because of the TCS3200 colour sensor. If this machine is placed in a different place such as different surrounding light, for example, indoor and outdoor, the values of the detected colour values will vary and not properly work because TCS3200 colour is not very accurate. So black background and external light supply is needed to reduce these errors.

Reading the RGB Values of the Red Ball with External Light Supply and Black Background



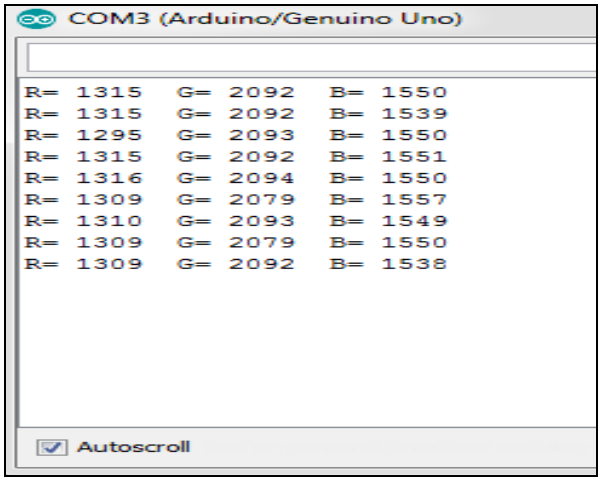


Fig6. Testing the Red Ball with External Light Supply and Black Background

Reading the RGB Values of the Green Ball with External Light Supply and Black Background

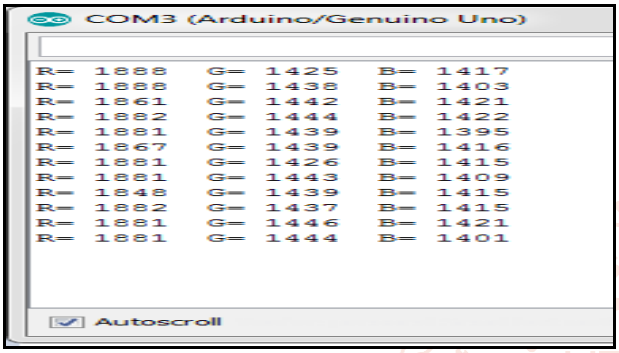
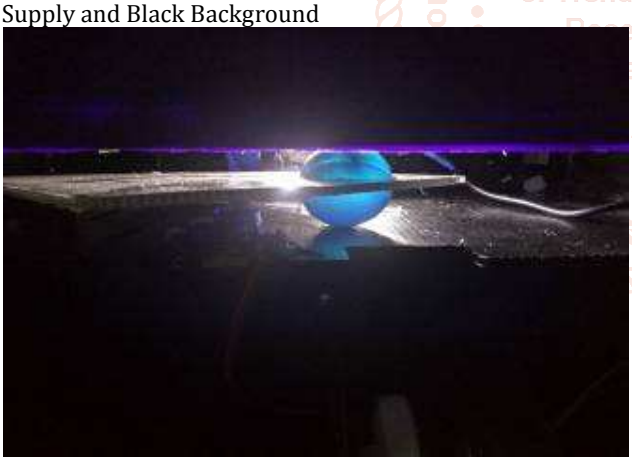


Fig7.Testing the Green Ball with External Light Supply and Black Background

Reading the RGB Values of the Blue Ball with External Light Supply and Black Background



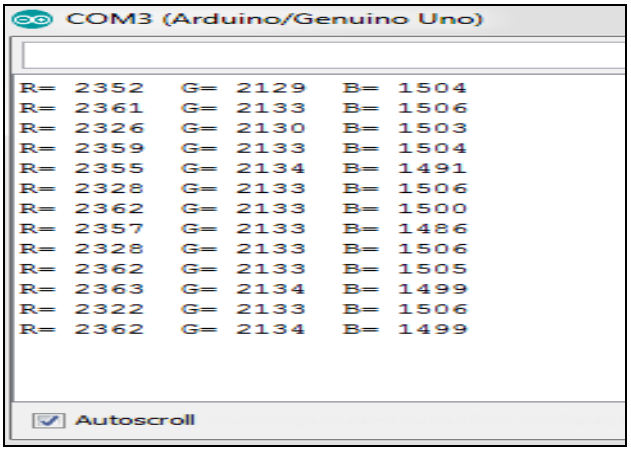
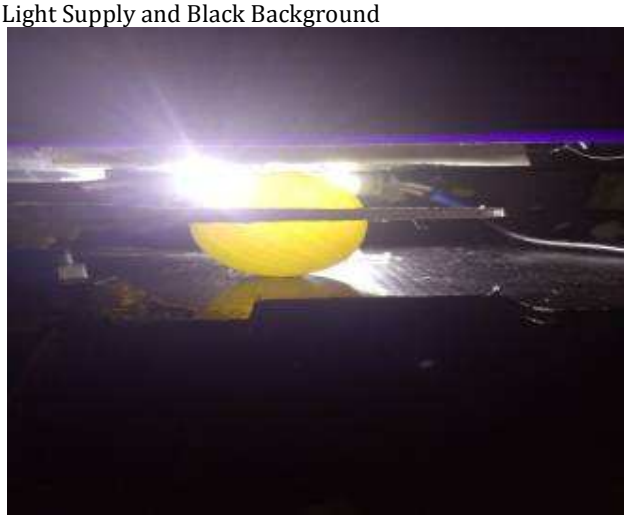


Fig8.Testing the Blue Ball with External Light Supply and Black Background

Reading the RGB Values of the Yellow Ball with External Light Supply and Black Background



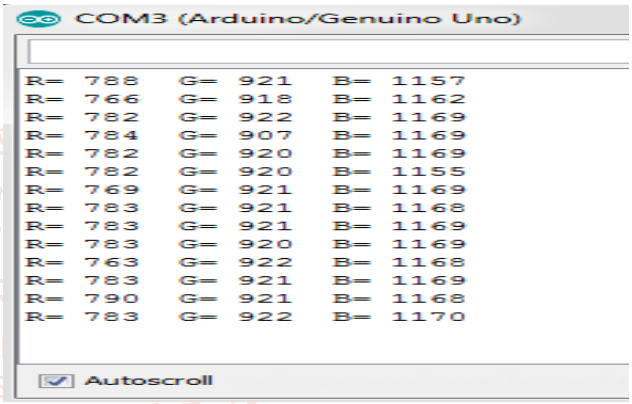


Fig9. Testing the Yellow Ball with External Light Supply and Black Background

Fig10.Colour Detector and Separator Based on Microcontroller

V. Conclusion

This paper is presented colour detection and sorting system, which is detected using RGB values of the objects in order to sort the objects based on the respective colour. So, the system is able to recognize all types of object. In this system, the colours are detected using the TCS3200 colour, servo motors are used to replace human operator and Arduino UNO serves as the main controller. In this thesis, a colour sorting system involves two steps. In the first step, colour detection is included. After the RGB values of the objects are detected, the detected RGB values are used in the colour sorting step. After that, colour sorting system can be started. The top servo motor is for placing the objects to the colour

sensor which detects the RGB values of the object and the bottom servo motor is used for placing the objects to the respective container. By applying the idea of this paper industry can easily sort the products according to the colour of the products. In this system, the colour of the object can be recognized for only 1cm below the sensor because of the TCS3200 colour sensor's limitation and colour sensor's sensing RGB values vary depending on the surrounding light intensity. So external light source and black background should apply to be constant light intensity. If other devices (eg. image processing camera, conveyor belt system, etc) are applied, various types of objects can be recognized above 1 cm and the performance of the system can be achieved higher than by using the colour sensor. Though this system has some limitations, by having done some modification this concept can be implemented in a wide range of applications.

VI. REFERENCES

[1] Apoorve: Servo Motor basic theory and working principle, (2015).

[2] Jim, L.: What is visible light? , April 30, (2015).

[3] https://www. Live science.com /50678-visible-light.html Kunhimohammed C. K.: Automatic Colour Sorting Machine by Using TCS230 Colour Sensor and PIC Microcontroller, (2015). https://www.researchgate.net/publication/29487831_Automatic_Color_Sorting_Machine_Using_TCS230_Color_Sensor_And_PIC_Microcontroller

[4] Anonymous: Learn C programming language, (2014). https://www.turorialspoint.com/cprogramming/indes.htm

[5] Dharmannagari, V. K. R.: Sorting of Objects Based on Colour by Pick and Place Robotic Arm with Conveyor Belt Arrangement, January, (2014).

[6] Chandra, S. N.: An automated machine vision-based system for fruit sorting and grading, December, (2012).

[7] Adna, A.: Introduction to microcontroller, (2018). https://www.theengineeringprojects.com/2018/03/introductionto-microcontrollers.html