Arduino Colour Sorter Machine

Project Report submitted in the partial fulfillment

Of

Bachelor of Technology

In

Mechatronics Engineering

By

Ketan Patni (H054)
Rafa Shaikh (H063)
Aparna Sharma (H064)
Akarsh Singh(H069)

Under the supervision of

Prof. Vinod Jain
(HOD, Mechatronics, MPSTME)

SVKM's NMIMS University
(Deemed-to-be University)

MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT & ENGINEERING
Vile Parle (W), Mumbai-56

2018-19

CERTIFICATE

This is to certify that the project entitled **"Arduino Colour Sorter Machine"**, has been done by **Ms. name** under my guidance and supervision & has been submitted in partial fulfillment of degree of Bachelor of Technology in Mechatronics of MPSTME, SVKM's NMIMS (Deemed-to-be University), Mumbai, India.

____

Name of Internal guide

(Internal Guide)

Date

Place: Mumbai

____

Prof. Vinod Jain

(HOD Mechatronics)

ACKNOWLEDGEMENT

Student Name
B.Tech (MTRX)
Roll No.

ABSTRACT

Sorting of products is a very difficult industrial process. Continuous manual sorting creates consistency issues. This paper describes a working prototype designed for automatic sorting of objects based on the color. TCS230 sensor was used to detect the color of the product and the Arduino Nano microcontroller was used to control the overall process. The identification of the color is based on the frequency analysis of the output of TCS230 sensor. Two servo motors were used to facilitate the sorting process. The first motor helps in transporting the object from the cylindrical pipe to the sensor and the second motor effectively works in conjunction with the sensor and distributes the object in separate compartments, in order to separate the products. The experimental results promise that the prototype will fulfill the needs for higher production and precise quality in the field of automation.

(This page is only for reference and should not be included in the final report)

Follow the given instructions while preparing the report:

1.	Font	Times New Roman.
2	Line spacing	1.5".
3.	Margins (left)	1.5”
4.	Remaining all margins	1”.
5.	Header Right corner	Academic Year
6.	Header Left corner	Project Title
7.	Footer: Right Corner	Page Number
8.	Chapter Title	Font size 18 and bold.
9.	Section titles	Font size 14 and bold
10.	Subsection titles	Font size 12 and bold
11.	Figure name (below the figures)	Times new roman , font size 10
12.	Table name (below the table)	Times new roman, font size 10
13.	Report	Font size 12

14. Appendix will include detailed information, lengthy derivations, raw experimental observations etc. preseented in the separate appendices, which shall be numbered in Roman Capitals (e.g. “Appendix IV”).
15. The report (One Copy) should be Spiral bound.
16. The report has to be signed by the respective mentors before the final Presentation / examination.
18. Refer to the following table of contents for the format.
19. Page numers in Table of Contents for List of Figures, List of Tables and Abbreviations has to be in Roman numbers.
20. Page numers from Introduction onward should be in decimal format.

Table of Contents

List of the figures

List of tables

Table No.	**Name of the Table**	**Page No.**

Abbreviations

Table No.	Name of the Table	Page No.

Chapter 1

Introduction

1.1 Background of the project topic

There is a wide usage of many products in our day to day life, and manufacturing of this products are done in many large scale and small scale industries. Arranging makes quality consistency issue. Nowadays the main difficulty that is faced after the production is of sorting Arranging of items in an industry is a dull modern process, which is by and large done physically. Consistent manual the need of this type of machine in the industries will help in sorting the machine according to their weight, size, color, shape, etc. This paper gives a brief information about the sorting of objects according to their color using TCS3200 color sensor, Arduino UNO and servo motors. The identification of color is done using frequency scaling of color detection.

1.2 Motivation and scope of the report (in paragraph form)

The shading sorting machine utilizing Arduino is an intriguing and prestigious venture for techies, who might want to consolidate Electronics, Machine building and programming. The shading Sorting Machines is utilized for sorting mostly RGB hues. This shading sorting machine isolates diverse hued questions and characterizes them into individual compartments/glasses. The shading sorting machine is completely mechanized with the assistance of Arduino. This electronic task made up of Arduino UNO alongside Arduino UNO BOB, RGB shading sensor, two servo engines and some plastic channels and tube parts

1.3 Salient contribution (in point form)

1.4 Advantages, Limitations and Applications

Advantages

- Accurate
- Good repeatability
- Reduce labor cost
- Less human interference
- Fully automatic operation

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 10 (2018) pp. 7383-7387 © Research India Publications. http://www.ripublication.com

Applications

- In food industry to identify rotted fruits and vegetables, in minor scale and big scale productions, to categorize the products established on the several factors.

- In production units to scan and identify the defects in raw materials.

- In fruits and vegetable farming areas (rural areas) where installation of expensive sorters is very difficult.

- In malls (to segregate and separate different clothes, toys, bags etc.) and in small shop.

The suggested framework will be a demo rendition which gives expense effective, taking less time and technically the easiest way for differentiating objects. This framework utilizes Arduino Uno which makes this model simple to utilize which is more additional effective. The main failure will be caused if the sensing of object according to color is not done. Therefore, it is very important to have proper and checked sensors. Further, making desirable changes it can be used in small scale and large scale industries as well.

Chapter 2

Literature survey

This isn't a special idea, for the execution of object sorting machine based on colour, size, weight, etc. The idea has existed for quite a while, after there has been advancement in technology.

Design And Development of Colour Sorting Robot Lim Jie Shen*, Irda Hassan -This gave us the knowledge of how a robot is used for the sorting process and no manual help or labor was needed.

Automated Object Sorting Using Raspberry Pi N.Aarthi1 , P.Sahithi2 , P.V.Sitaramaih, M.InduVardhani, N. Ranjith Kumar, D. SuneelVarma –This published work gave different ideas in which this sorting mechanism can be taken into consideration.

Sorting Of Objects Based On Colour,Weight And Type On A Conveyor Line Using PLC, S. V. Rautu, A. P. Shinde , N. R. Darda,A, V.Vaghule, C, B.Meshram, S. S. Sarawade –their gave us the knowledge of how different sensors are responsible and helpful for the sorting based on weight, colour and metal.

Chapter 3

Problem Definition

There are various hurdles that we overcome in any given project. The first obstacle to overcome was to find a durable and material material to make the structure of the machine. We ended up choosing mdf as its sustainable as well as can be laser cut. It also makes the machinery portable. We eliminated the use of a bigger microcontroller and substituted it with the nano version of it as we realized it could do the job just as efficiently. Another hurdle that we had to overcome was the fact that the photo sensor is extremely sensitive in not only the colour of the given object, here skittles, but also of the surrounding. It fluctuates in artificial lighting/ natural lighting given that light is composed of different colours of varying wavelength which hampers the result. To overcome this, we set a generic and a larger range for separate colours along with one more criterion to get the best possible result.

Chapter 4

Methodology

4.1 Concept

Colors are light waves that are no different than the electromagnetic waves emitted by cell phones. It is our brain cells that interpret them as real colors . For robot, the simplest way they can use to detect colors is by uses the filters of three main colors, which are red, green and blue and compared the value on the light reflected on it. The value taken will then send signal to Arduino which analyse which color is the object. The color sorting robot code is programmed using Arduino software. Programming code is researched and written in order for the color sorting robot to carry out recognition and sorting mechanisms. The connection is done by connecting wires to connect up Arduino Uno which act as microcontroller, servo as well as color sensor. The servo motor then slides the candy left and right at different angle to different location

4.2 Color Sensor

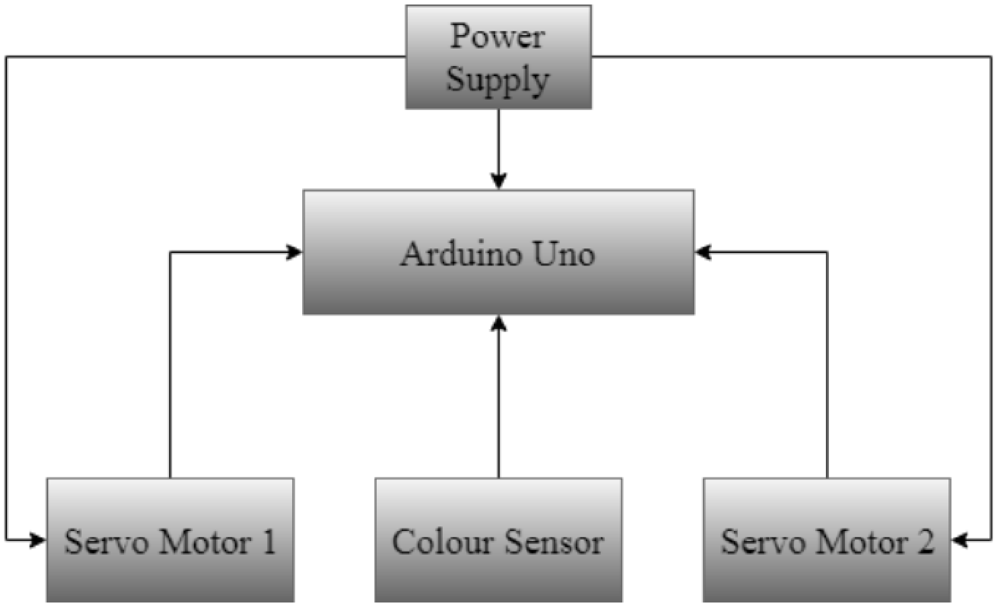
The color of an object we can see in fact is the chromatic light the object reflects in the white light (sunlight) after it absorbs the rest ones. The white color is a mixture of various visible colors, which means it includes each colored light like red (R), yellow (Y), green (G), blue (B), and purple (P). Based on the theory of three primary colors, any color is made by mixing the three primary colors (red, green, and blue) in a certain proportion.

Thus, knowing the proportion you can get the color of the tested object. For TCS3200, when a color filter is selected, it only allows a specific primary color to pass through and blocks the other two colors. With the light intensity value of the three primary colors, by analysis we can know the color of light reflected to TCS3200.

The TCS3200 sensor has 4 types of filter: red filter, green filter, blue filter, and clear with no filter. The filter is selected based on the high/low of pin S2 and S3 on the module

4.3 DESIGN AND ASSEMBLY

The dimensions of are model are made on Solidworks 3D modeling software to perfectly fit our components in a compact way as well as we made sure that all our moving components have a proper degree of freedom.The material we used to make the structure is 3mm mdf board which is strong and durable.The different parts from the software have been cut precisely by a lazer cutting machine and assembled by means of a super glue manually.The components and jumper wires are attached with respect to our circuit schematics presented in the report


Power Supply
Arduino Uno
Servo Motor 1
Colour Sensor
Servo Motor 2

Chapter 5

Mechatronic System Analysis

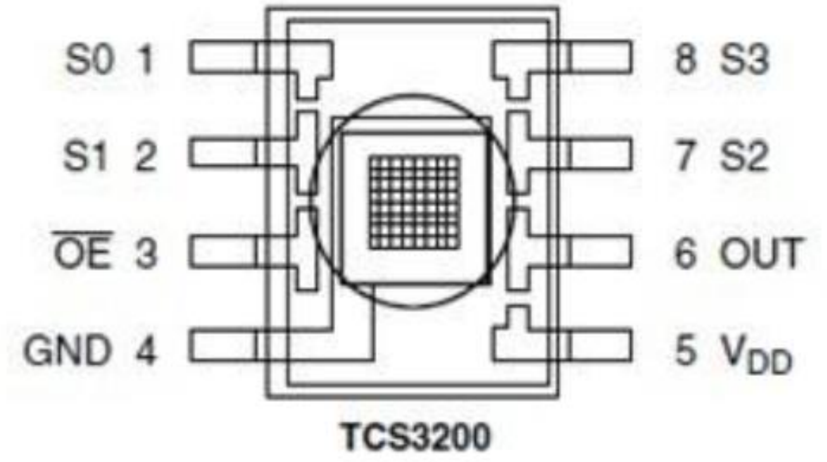


Figure 3: color sensor [6]

1) Colour sensor

A colour sensor depends on a photodiode which measure the power reflected by the protest for a red, green and blue light source. Because of the same it will help in detecting the specified color. There are total 8 pins in a color sensor.pin no 7 and 8 are selection pins, pin no 1 and pin 2 are frequency scaling pin. Pin 6 is output.

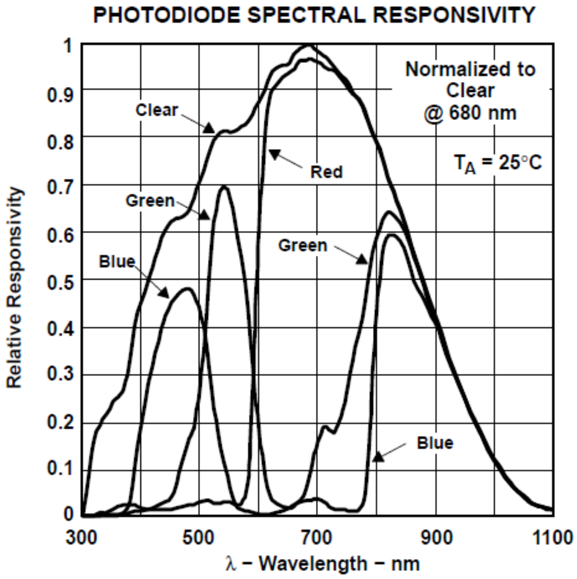
Working of color sensor

Each photodiodes have an 8*8 matrix which results into 16 of green filter, 16 or red filter, 16 of blue filter and the rest 16 for clear filter when no colour will be detected.

Pin description of color sensor
Table 1: Pin description [16]

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

OUT	O	Output frequency (fo).
S0,S1（1,2）	I	Output frequency scaling selection inputs.
S2,S3（7.8）	I	Photodiode type selection inputs
VDD（5）		Supply voltage



1. Apart from colour sensor we have used arduino uno and two servo motors (sg90).

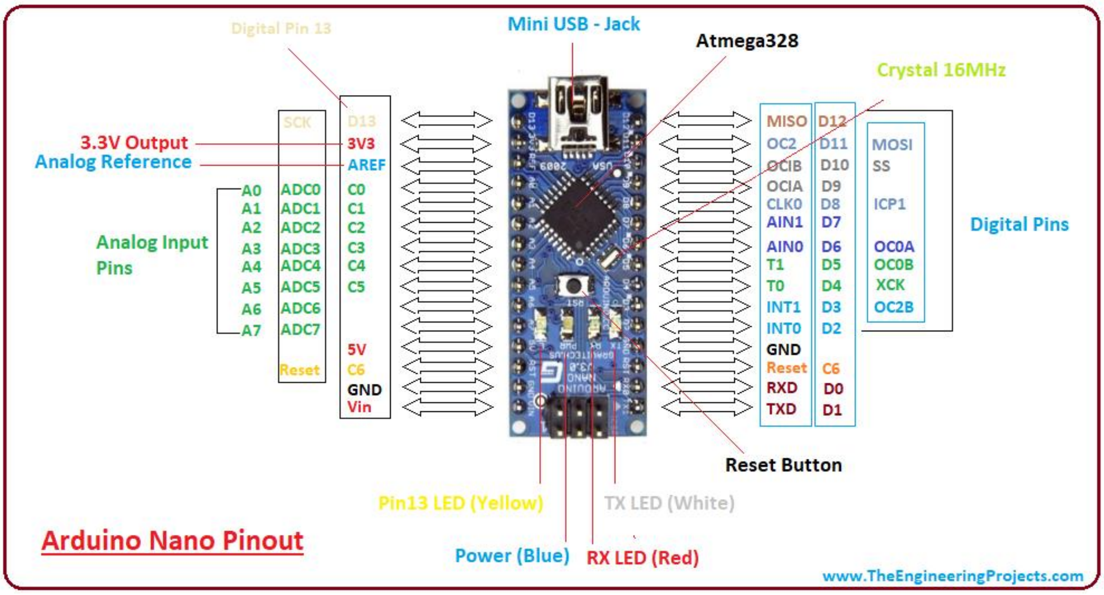
   1) Arduino Uno

   The Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount ofaccessible knowledge that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open- source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide [11].

Figure 5: Arduino Uno [17]



2) Servo Motor

It is tiny and lightweight with high output power. This servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. It comes with a 3 horns (arms) and hardware [12].

Figure 6: Servo Motor [18]

The Specifications of servo motor are mentioned below

- • Operating voltage: 4.8 V (~5V)
- • Operating speed: 0.1 s/60 degree
- • Stall torque: 1.8 kgf·cm
- • Dead band width: 10 µs
- • Temperature range: 0 oC – 55 oC [16].

Chapter 6

Software Description

So, we need to include the “Servo.h” library, define the pins to which the color sensor will be connected, create the servo objects and declare some variables needed for the program. In the setup section we need to define the pins as Outputs and Inputs, set the frequency-scaling for the color sensor, define the servo pins and start the serial communication for printing the results of the color read on the serial monitor.

In the loop section, our program starts with moving the top servo motor to the position of the skittle charger. Note that this value of 115 suits to my parts and my servo motor, so you should adjust this value as well as the following values for the servo motors according to your build.

Next using the “for” loop we will rotate and bring the skittle to the position of the color sensor. We are using a “for” loop so that we can control the speed of the rotation by changing the delay time in loop.

Next, after half a second delay, using the custom made function, readColor() we will read the color of the skittle. Here’s the code of the custom function. Using the four control pins and the frequency output pin of the color sensor we read color of the skittle. The sensor reads 3 different values for each skittle, Red, Green and Blue and according to these values we tell what the actual color is.

Chapter 7

Testing and Results

7.1 Light Intensity Test

The tests are carried out indoor and outdoor to record the results and the values showed that the RGB values taken indoor are more or less higher than the one taken outdoor. This is due to the present of more than single light out door which affected the results. Tests taken indoor and outdoor

It can be seen from Tables 1 and 2 that values of the filter is different when tested outdoor and indoor. The presence of other light has caused the TCS3200D color sensor filters to record RGB (Red, Green, Blue) value that are closer to each other. For example, the most obvious results is when tested with green candy outdoor, the green filter and blue filter showed very close G and B value. This might cause the system to detect it as an error and not carried out sorting task as planned when two filters share the same reading. This means the system fails to detect if it is blue candy or green candy. It can be verified putting both red candy and blue lights together. The mixture resulting from the interaction of a colored object with a colored light source is an example of subtractive mixing. Figure 8 shows that the mixture section of both lights eventually becomes pink. The pink color is getting more and more obvious when the white light is added. The same things happen in this outdoor experiment. The color sensor will detect and record inconsistent RGB value which has high R and B values.

This scenario can be concluded that surrounding light is actually affecting the result and should be avoided. In the situation where more than one light is present, it is advisable to just focus on single light surface. RGB result will be more reliable when tested indoor or in a confined space. Besides preventing the

Design and development of color sorting robot 77

Journal of Engineering Science and Technology Special Issue 1/2015

presence of too many unwanted light, in order to get optimum result, there are another factor needed to be considered throughout the experiment which is listed below. It is the best to choose the objects with high concentration of color, so the surrounding ray will not affected the result too much. For example, dark green, dark blue and dark red is chosen instead of light version of them. This is to increase the accuracy of the RGB values of an object and thus increase the efficiency of the recognising and sorting process by this system.

7.2 Color Selection Test

The test is carried out to determine which color of the candy can be best detected by color sensor TCS3200D. The variety colors of the candys used are blue, red, green, yellow, purple as well as orange. It can be seen from Table 3 that either two or all three of red, green and blue filters showed that the result is not too much differ from one another when the colors tested with secondary or tertiary color. The secondary colors such as yellow, purple and orange cannot be detected as easily as colors such as red, green and blue.

For example, the orange candy is not suitable to be chosen as a tested object as it has R filter and G filter values that are too close to each other.

filters. In the situation when yellow light is break down, it can become red light and green light which are mixed together. It can be observed by human but not the color sensor. For green, blue and red color, they are primary and cannot be breakdown even further. Or in other word, there is no other colors that can form them. Hence, it justified the reason these three primary colors filters are the used in the color sensor nowadays and also act as optimum color of to be chosen as testing object.

Chapter 8

Advantages, Limitatons and Applications

Advantages:

- It helps in sorting of objects based on color approach. It also helps in counting of objects.
- Automated system can be built using color sensors which help in completion of work in less time. Moreover human intervention is not needed.
- Powerful and large memory color sensor ICs are available at low cost. This has driven its use in many applications.
- It is easy to change or modify manufacturing setups without even re-programing the sensor device.
- This is beneficial in low volume manufacturing applications having frequent color variations.
- With the advancement of technology and memory loaded with color intensity data, color sensor controller can store and can make color matching decisions on unlimited number of colors virtually.

Applications:

- Colour sorters are used for food processing industry, such as coffee, nuts, and oil crops.
- The transparency of the diamond is measured by the colour sorter and used as a measurement of its purity, and the diamonds are mechanically sorted accordingly.
- In the recycling industry, colour sorters can distinguish between coloured and colourless PET and coloured and colourless HDPE flakes, as well as being able to separate flakes by colour before re-granulation.

Limitation

- The approach is costly for small scale industries.
- It does color matching or identification in applications requiring only pass/fail output.
- Operating distance range of the color sensors are matter of concern. This needs to be choosen appropriately with rigorous testing in the setup.

Chapter 9

Conclusion and Future Scope

It is very useful in wide varieties of industries, especially in the packaging section. A sorting machine will enable faster distribution in factories as well local grocery departments. Automatic sorting machine enhances efficiency, practicality, and safety of operators. It ensures remarkable processing capacity as well as peerless performance including color detection. Of course we need to add high speed DC motors and sensors with appreciable response to speed up the system for industrial application.

The model can be improved by making some changes in the program and components. Some suggestions are given below.

- We can add a load cell for measurement and control of weight of the product
- We can also add a counter for counting the number of products
- Speed of the system can be increased accounting to the speed of production
- The system can be used as a quality controller by adding more sensors
- The sensor can be changed according to the type of product
- The servo motor can be replaced with stepper motor
- Multiple sensors can be used for the optimum results
- We can make the use of conveyor belts to facilitate bigger objects carrying different materials of varying density

References

- ⇨□ Color sensors applications: http://www.baumerelectric.com/be27.html?L=1
- ⇨□ RGB color space: http://en.wikipedia.org/wiki/RGB_color_space
- ⇨□ HSL color space: http://www.chaospro.de/documentation/html/paletteeditor/colorspace_hsl.htm
- ⇨□ TCS230 color sensor datasheet http://www.taosinc.com/productdetail.aspx?product=3
- [12] Norfazlinda Binti Daud, Application of colors sensor in an automated system , Technical University Malaysia, May 2007.

- [13] Bickman, Josh, Automated Color-Sorting uses optical technology , vol. 13, 1996.

- [6] D.J. Lee and R. S. Anbalagan, High-speed automated colorsorting vision system , in Optical Engineering Midwest 95, pp 573-579, 1995.

- Available resources at www.wikipedia.com

Apendix

Appendix A: Soft Code Flowcharts

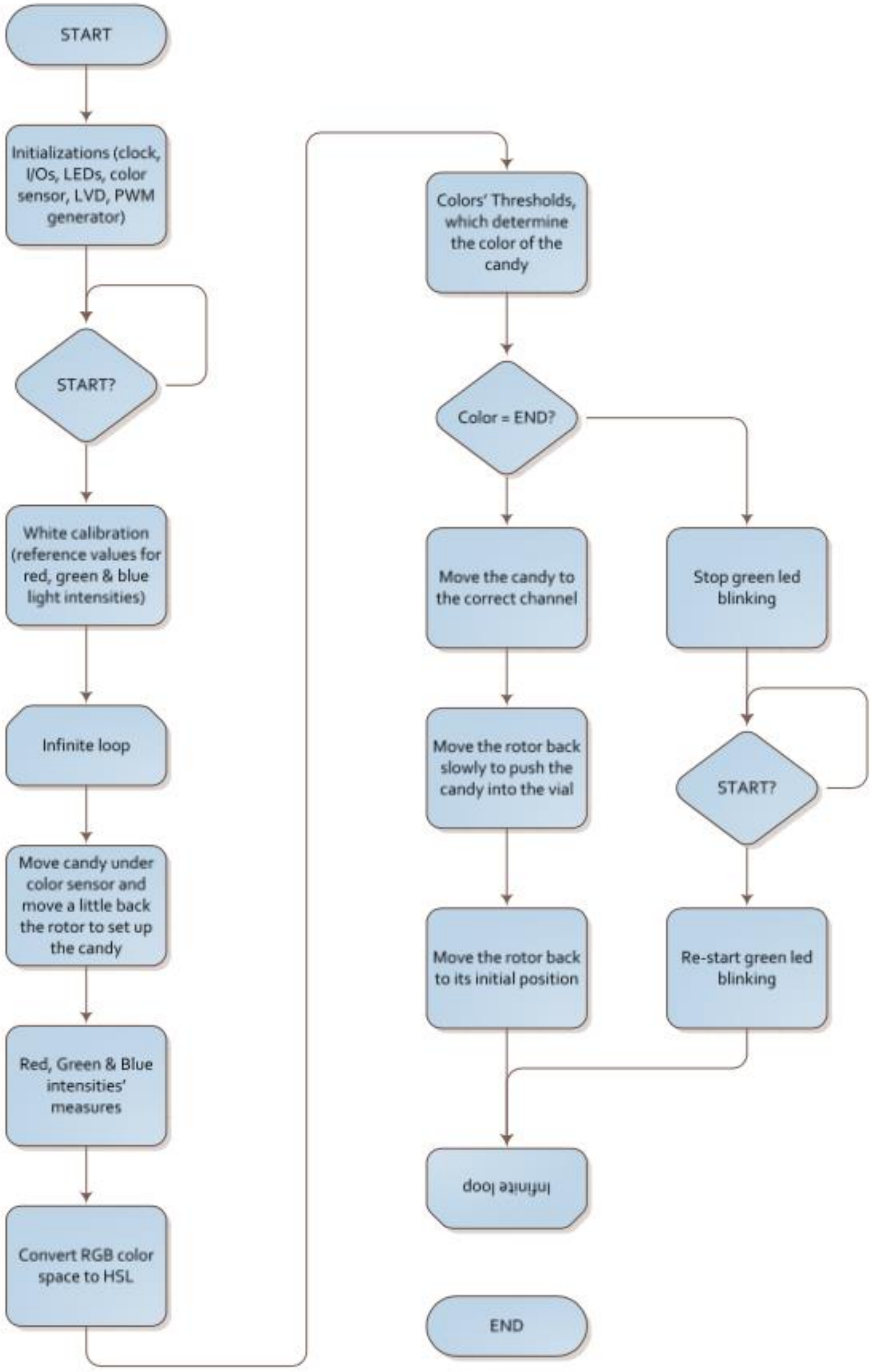
The Software Code

```
#include <Servo.h>

#define S0 2

#define S1 3

#define S2 4

#define S3 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 20%
  digitalWrite(S0, HIGH);
  digitalWrite(S1, LOW);
  topServo.attach(7);
  bottomServo.attach(8);
  Serial.begin(9600);
}
void loop()
{
  topServo.write(115);
  delay(500);

  for(int i = 115; i > 65; i--) {
    topServo.write(i);
    delay(2);
  }
  delay(1000);

  color = readColor();
  delay(10);
  switch (color) {
    case 1:
    bottomServo.write(10);
    break;
    case 2:
    bottomServo.write(45);
    break;

    case 3:

    bottomServo.write(85);

    break;

    case 4:

    bottomServo.write(125);

    break;



    case 0:

    break;

  }

  delay(300);


  for(int i = 65; i > 29; i--) {

    topServo.write(i);

    delay(2);

  }

  delay(200);


  for(int i = 29; i < 115; i++) {

    topServo.write(i);

    delay(2);

  }

  color=0;
```

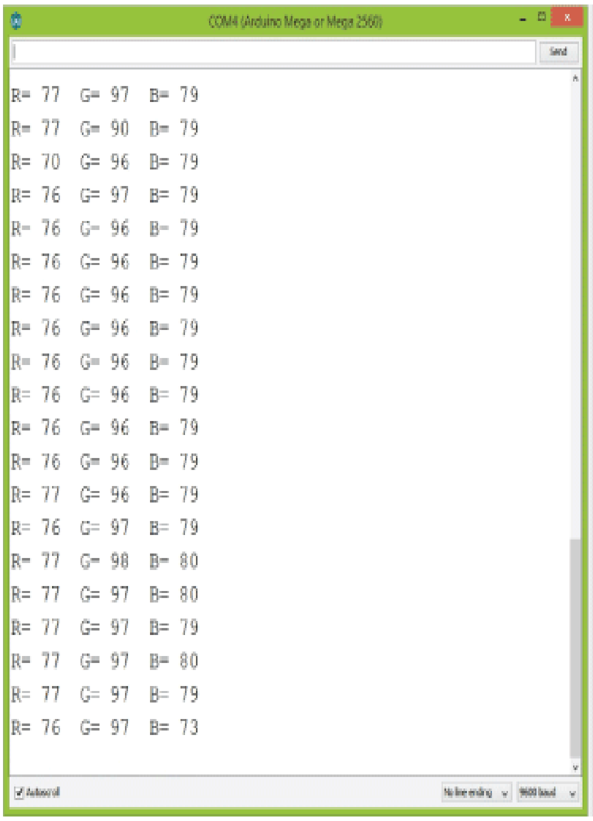
```
}
// Custom Function - readColor()
int readColor() {
  // Setting red filtered photodiodes to be read
  digitalWrite(S2, LOW);
  digitalWrite(S3, LOW);
  // Reading the output frequency
  frequency = pulseIn(sensorOut, LOW);
  int R = frequency;
  // Printing the value on the serial monitor
  Serial.print("R= ");//printing name
  Serial.print(frequency);//printing RED color frequency
  Serial.print("  ");
  delay(50);
  // Setting Green filtered photodiodes to be read
  digitalWrite(S2, HIGH);
  digitalWrite(S3, HIGH);
  // Reading the output frequency
  frequency = pulseIn(sensorOut, LOW);
  int G = frequency;
  // Printing the value on the serial monitor
  Serial.print("G= ");//printing name
  Serial.print(frequency);//printing RED color frequency
  Serial.print("  ");
  delay(50);
```

```
// Setting Blue filtered photodiodes to be read

digitalWrite(S2, LOW);

digitalWrite(S3, HIGH);

// Reading the output frequency

frequency = pulseIn(sensorOut, LOW);

int B = frequency;

// Printing the value on the serial monitor

Serial.print("B= ");//printing name

Serial.print(frequency);//printing RED color frequency

Serial.println("  ");

delay(50);


if(R<145 & R>120 & G<140 & G>120){

  color = 1; // Green

}

if(R<105 & R>85 & G<200 & G>170){

  color = 2; // Red

}

if (G<150 & G>120 & B<95 &B>75){

  color = 3; // Blue

}

if (R<84 & R>60 & G<115 &G>95){

  color = 4; // yellow

}
```

```
  return color;
}
```


START
Initializations (clock, I/Os, LEDs, color sensor, LVD, PWM generator)
START?
White calibration (reference values for red, green & blue light intensities)
Infinite loop
Move candy under color sensor and move a little back the rotor to set up the candy
Red, Green & Blue intensities' measures
Convert RGB color space to HSL
Colors' Thresholds, which determine the color of the candy
Color = END?
Move the candy to the correct channel
Stop green led blinking
Move the rotor back slowly to push the candy into the vial
START?
Move the rotor back to its initial position
Re-start green led blinking
Infinite loop
END

Appendix B: Data Sheets



```
R= 77   G= 97   B= 79
R= 77   G= 90   B= 79
R= 70   G= 96   B= 79
R= 76   G= 97   B= 79
R= 76   G= 96   B= 79
R= 76   G= 96   B= 79
R= 76   G= 96   B= 79
R= 76   G= 96   B= 79
R= 76   G= 96   B= 79
R= 76   G= 96   B= 79
R= 76   G= 96   B= 79
R= 76   G= 96   B= 79
R= 77   G= 96   B= 79
R= 76   G= 97   B= 79
R= 77   G= 98   B= 80
R= 77   G= 97   B= 80
R= 77   G= 97   B= 79
R= 77   G= 97   B= 80
R= 77   G= 97   B= 79
R= 76   G= 97   B= 73
```

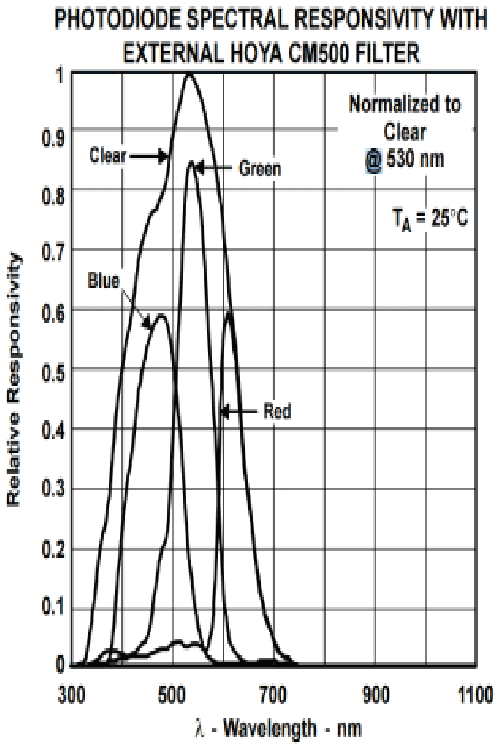
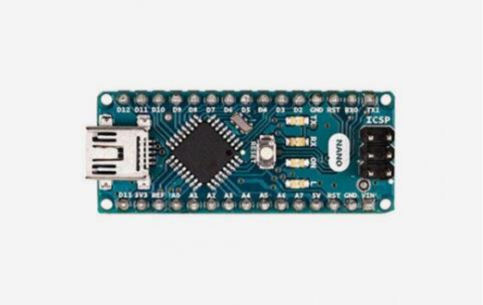


*Figure takan from the TCS230Sensor datasheet

Appendix C: List of Components

Sr No	Component	
1	Arduino Nano	
2	Servo Motor SG90 x2	
3	Colour Sensor TCS230	
4	Jumper Wires	

5	9V battery/Power Source	HW Hi-Watt 6F22 9V BATTERY FOR TRANSISTOR RADIOS GENERAL PURPOSE
7	USB cable	
8	3 mmMDF board	
9	Paper Cups	

10	Candy	
11	Plastic Piper	