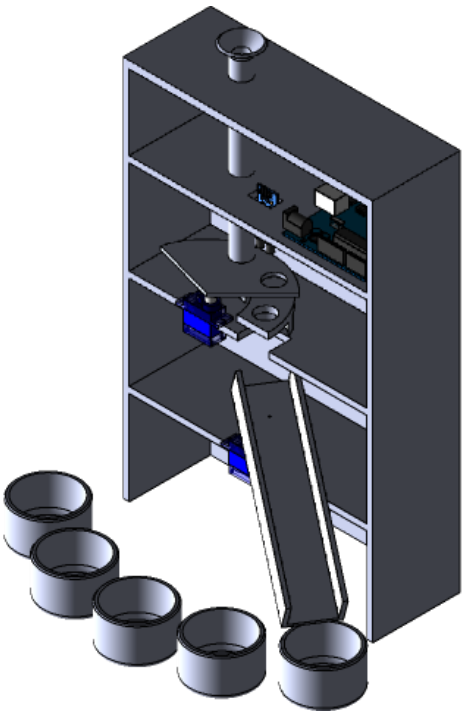
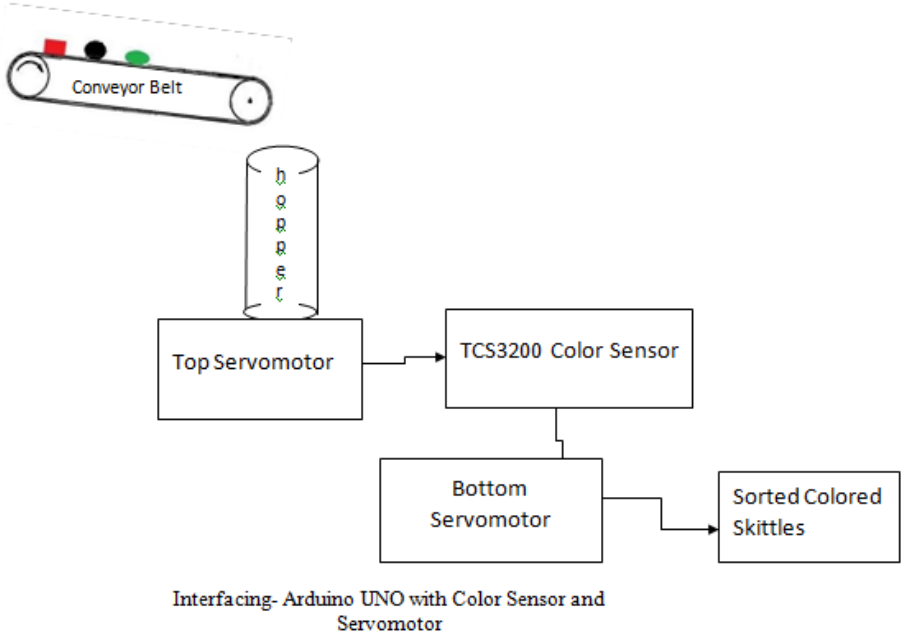
ARDUINO BASED COLOR SORTING MACHINE USING TCS3200 COLOR SENSOR

BAYAN ABO RAZMEH / 16/5/2022



INTRODUCTION

- This prototype is built as a simple digital gadgets like microcontroller for processing, Servo motors for actions and coloration sensor for recognizing exclusive-colored devices.
- This venture makes use of a simplified and not steeply-priced technique for sorting the substances of a unique colored items, it's far sensing the color of the object and kind out the different colored devices. Servo automobiles are used to manipulate the motion of the skittles are amassed on the hopper. A servo motor is used to pressure the skittles to the sensor and the sensor that is interfaced with ARDUINO identifies the shade of the object and the bottom servo is operated as consistent with the deliver code



Interfacing- Arduino UNO with Color Sensor and Servomotor

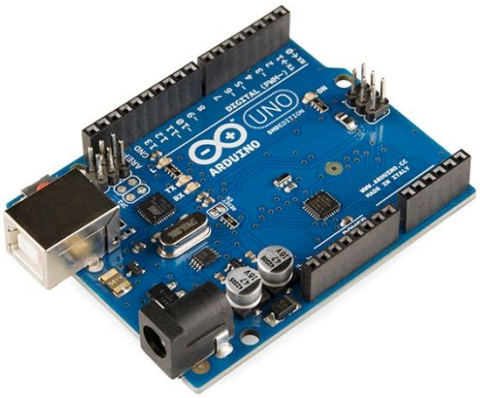
Figure 1: Block Diagram of the System

HARDWARE DESIGN:

COMPONENTS:

ARDUINO UNO:

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output pins that can be interfaced to various expansion boards and other circuits.

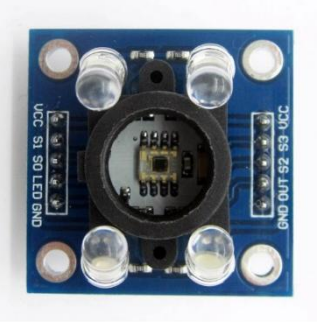


SERVO MOTOR SG90:

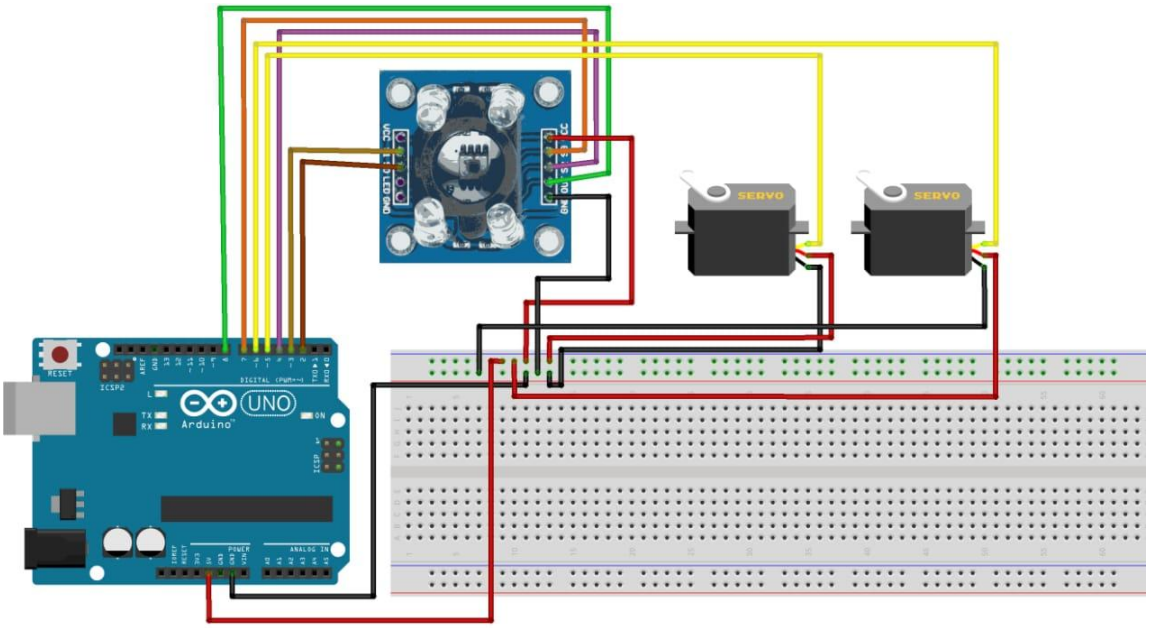
A servomotor (or servo motor) is a rotary or linear actuator that allows for precise control of angular or linear position and of velocity or acceleration. It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, usually a dedicated module designed specifically for the use with servomotors.

COLOR SENSOR TCS230/TCS3200:

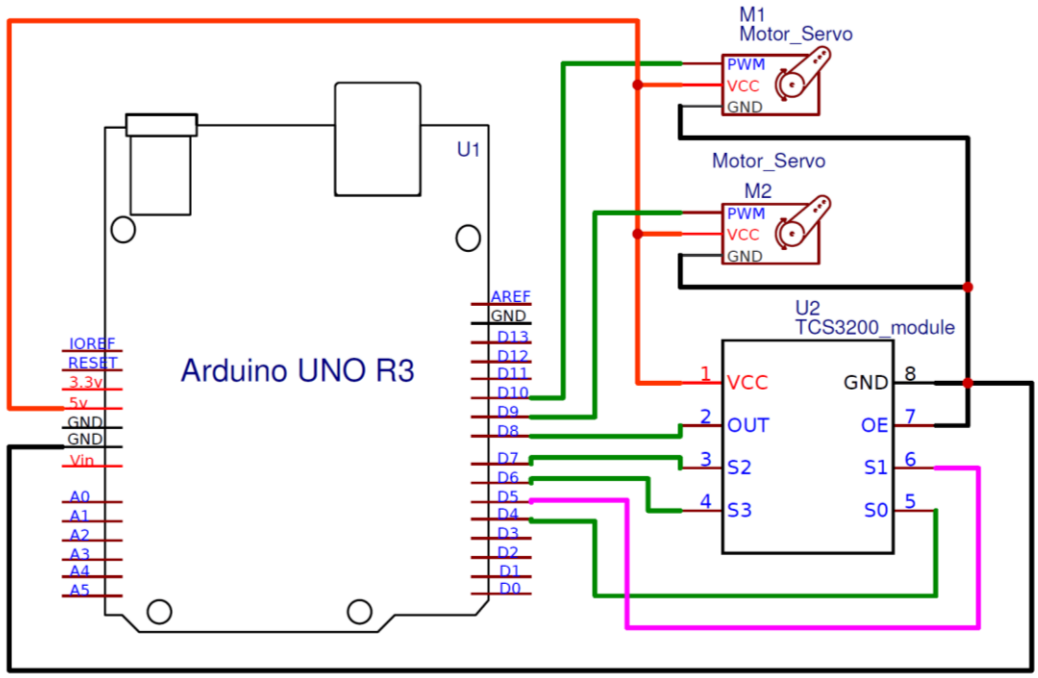
The TCS230 senses color light with the help of an 8 x 8 array of photodiodes. Then using a Current-to-Frequency Converter the readings from the photodiodes are converted into a square wave with a frequency directly proportional to the light intensity. Finally, using the Arduino Board we can read the square wave output and get the results for the color



Hardware Diagram:



SCHEMATIC VIEW:



SOFTWARE DESIGN:

THE CODE:

```
#include <Servo.h>
#define S0 5
#define S1 6
#define S2 3
#define S3 4
#define sensorOut 2
Servo topServo;
Servo bottomServo;
int frequency = 0;
int color = 0;
void setup() {
   Serial.begin(9600);
  pinMode(S0, OUTPUT);
  pinMode(S1, OUTPUT);
  pinMode(S2, OUTPUT);
  pinMode(S3, OUTPUT);
  pinMode(sensorOut, INPUT);
  digitalWrite(S0, HIGH); //20% scalling
  digitalWrite(S1, LOW);
  topServo.attach(9);
  bottomServo.attach(10);

}

void loop() {
  topServo.write(70); //set top servo at the first hole to catch the candy
  delay(500);

for(int i = 70; i >= 20; i--) { //move top servo with the candy to the color sensor position
  topServo.write(i);
  delay(2);
}
delay(500);

color = readColor(); //read the value from color sensor and put the bottom servo in the right path
delay(10);
switch (color) {
  case 1:
  bottomServo.write(22);
  break;
  case 2:
  bottomServo.write(70);
  break;
  case 3:
  bottomServo.write(120);
  break;


  case 0:
  break;
}
delay(300);

for(int i = 20; i >=0; i--) { //move top servo with candy from the color sensor position to the second hole position
    topServo.write(i);
    delay(5);
  }
  delay(200);

  for(int i = 0; i <= 70; i++) { //return top servo to the first position to catch another candy
    topServo.write(i);
    delay(2);
  }
  color=0;
}


int readColor() { //color sensor function to read the color

  // Setting red
  digitalWrite(S2, LOW);
  digitalWrite(S3, LOW);
  frequency = pulseIn(sensorOut, LOW);
  int R = frequency;
  Serial.print("R= ");
  Serial.print(frequency);
  Serial.print("  ");
  delay(50);

  // Setting Green
  digitalWrite(S2, HIGH);
  digitalWrite(S3, HIGH);
  frequency = pulseIn(sensorOut, LOW);
```

```
  int G = frequency;
  Serial.print("G= ");
  Serial.print(frequency);
  Serial.print("  ");
  delay(50);

  // Setting Blue
  digitalWrite(S2, LOW);
  digitalWrite(S3, HIGH);
  frequency = pulseIn(sensorOut, LOW);
  int B = frequency;
  Serial.print("B= ");
  Serial.print(frequency);
  Serial.println("  ");
  delay(50);

  if(R<560 & R>350 & G<600 & G>400&  B<868 & B>500 ){
    color = 1; // YeLLOW 22
  }
  if(G<800 &G>600& B<900 & B>600 & R<600 & R>400){
    color = 2; // orange 70
  }
  if (G>500  & G <700 & B <1000 & B>640 & R<980 & R>640  ){
    color = 3; //GREEN 120
  }


  return color;
}
```

FLOWCHART: