ARDUINO SOLAR TRACKER

Review 2

Navya Saxena- 18BCE0770
Rishabh Kumar- 18BCE0816
Devina Varshney – 18BCE0741
Saloni Uday Parekh – 18BCI0212

Krishna Agarwal – 17BIT0186
Hrithik Sarda – 17BEC0078
Prithvi Taladiya – 17BCE0364
Nithish Kanna - 17BEC0919

Components Required

1. Two Servo motors
2. Solar panel
3. Cardboard
4. Four LDRs
5. Straight perforated metal strip
6. 10KΩ resistor
7. Arduino board
8. Connecting wires
9. Breadboard Circuit Diagram

Set-Up

01

- Take cardboard. Make a hole in the middle and four holes on four sides so that LDR fit into that.
- Stick the solar panel to the cardboard and bring two wires of the panel out.

02

- Now cut one of the two leads of the LDR so that one lead is shorter and other is longer.
- Insert these four LDRs into four holes.
- Bend the straight Perforated metal strip.
- Place the bent metal strip on the back side of the cardboard
- Apply glue to the LDR to fix them firmly.

03

- Solder the two leads of LDR as shown
- To the other ends of LDR Solder resistors of 10k ohm
- Join the four leads of the 4 LDRs by connecting with a wire.

04

- Now take a bus wire. This is used to connect the Outputs of four LDRs to Arduino board.
- Insert it into metal strip as shown in the image.
- Now Solder the four wires to four LDRs at any point between LDR and resistor.

05

- Insert another two-wire bus into the perforated metal strip as shown. This is used for supplying Vcc and GND to LDR circuit.
- Solder one wire to the leads of LDRs which are connected to resistors and other wire to other leads.
- Short the leads of LDRs connected to resistors using a wire.

06

- Now connect a servo motor to the Perforated metal strip using Screw.
- Apply glue to the servo to fix it firmly.

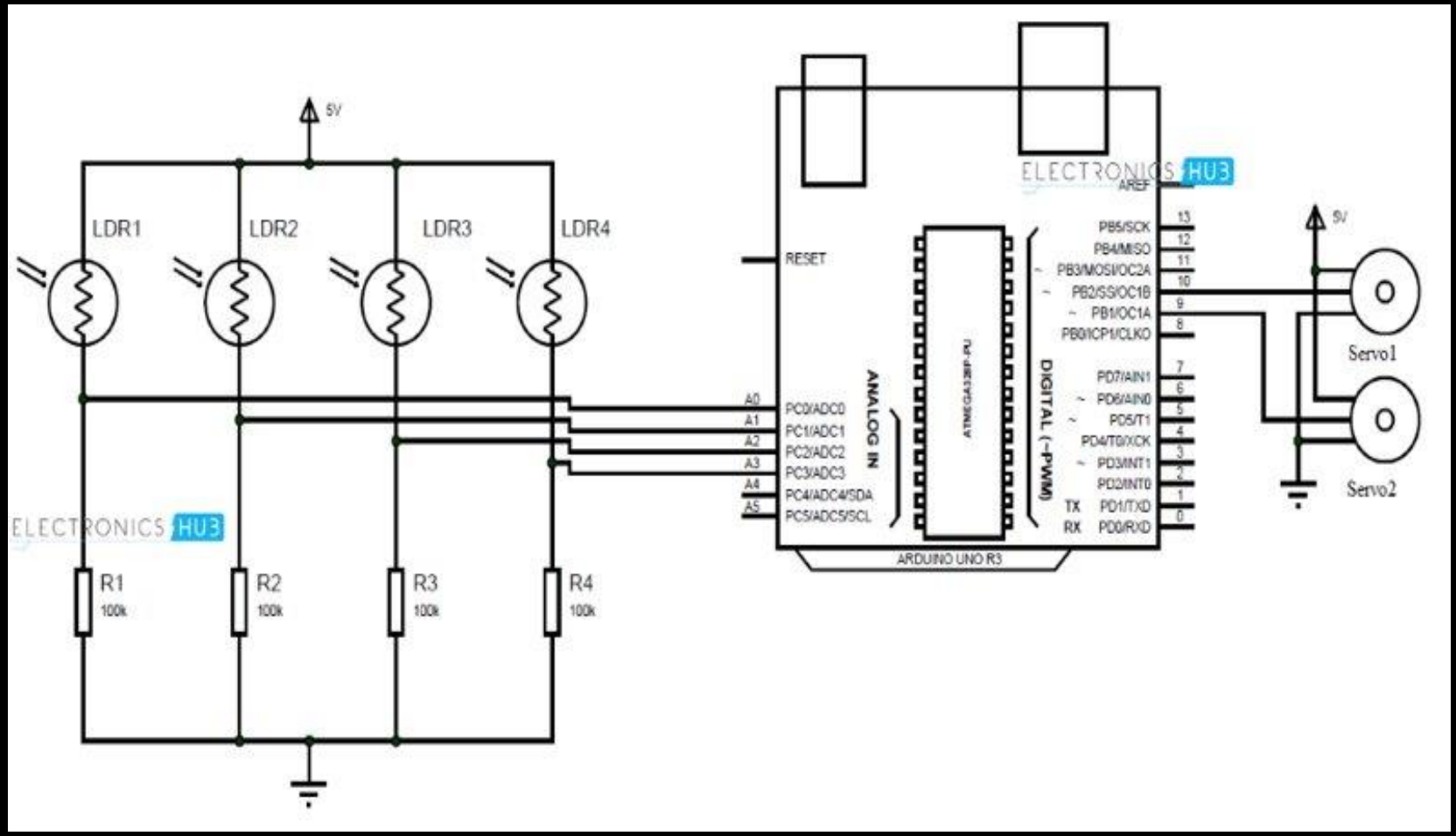
07

- Take another straight Perforated metal strip and bend it.

08

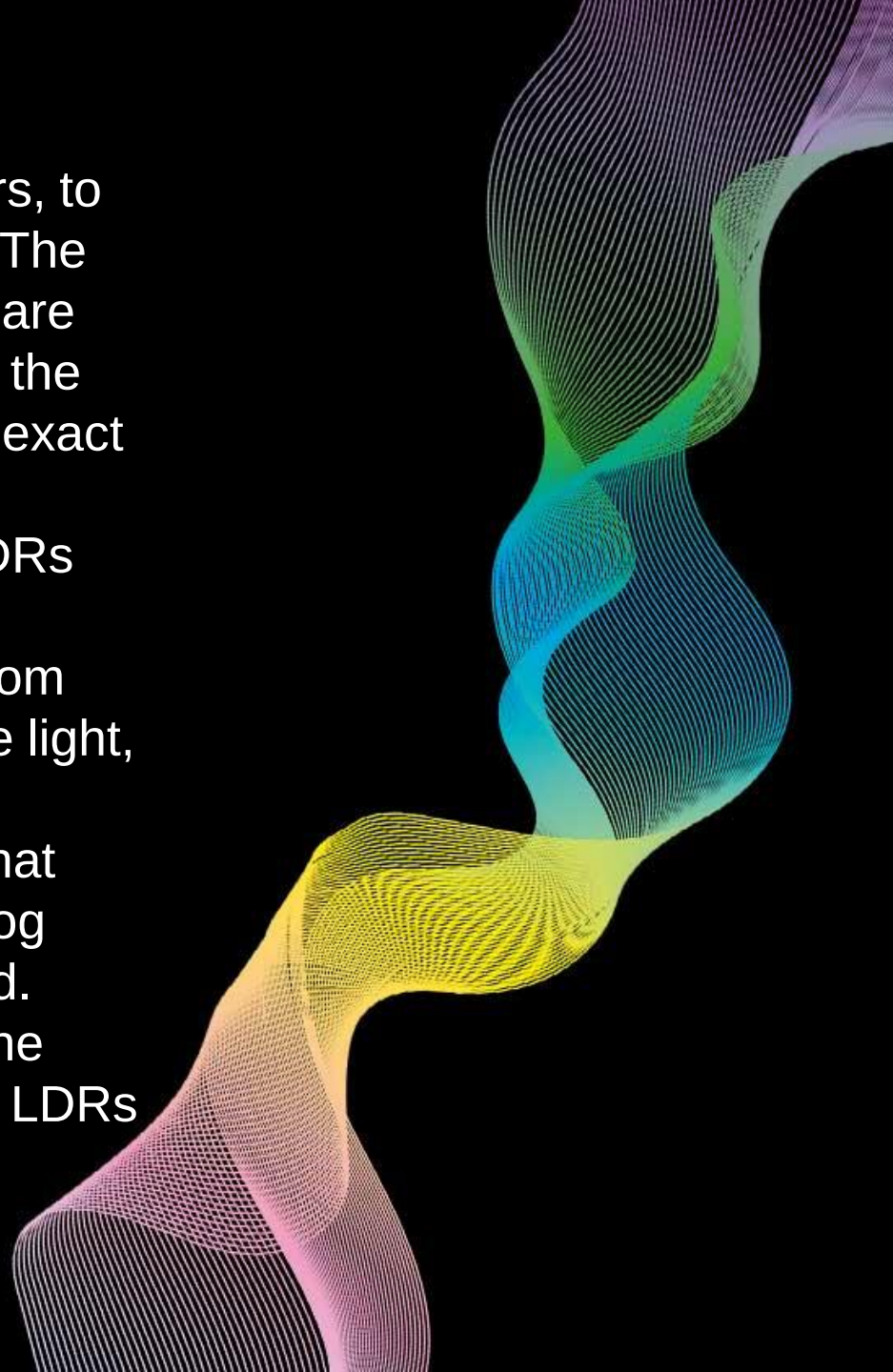
- Now place the setup of solar panel and first servo motor to the metal strip of second servo motor.

CIRCUIT DIAGRAM

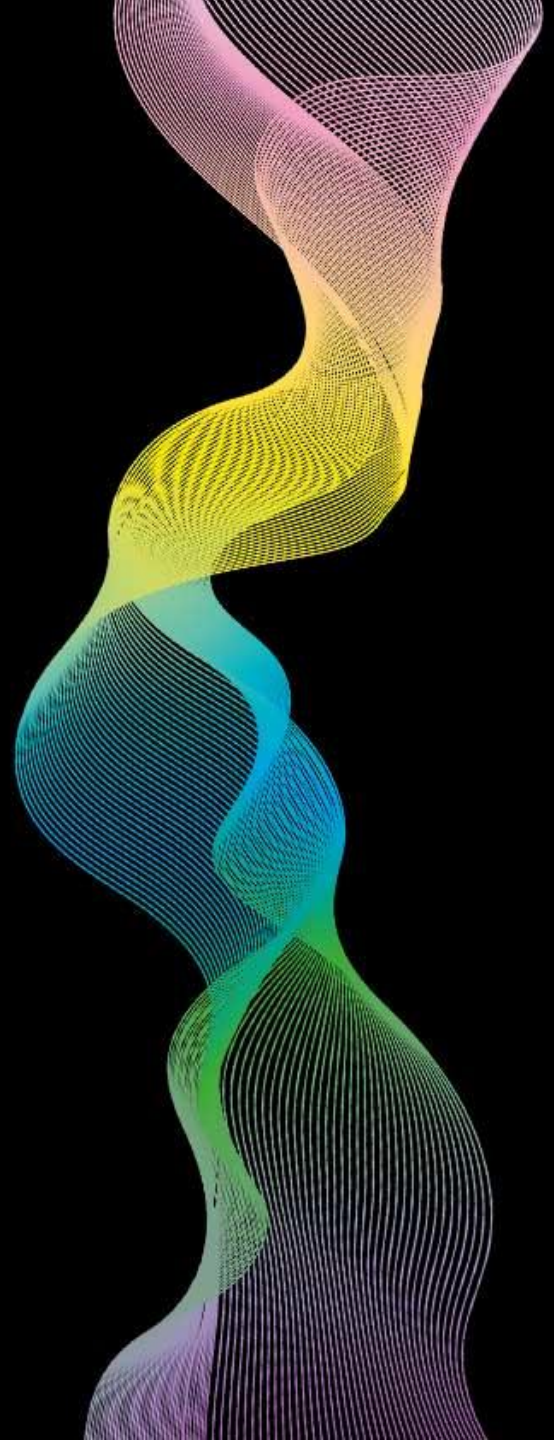


WORKING

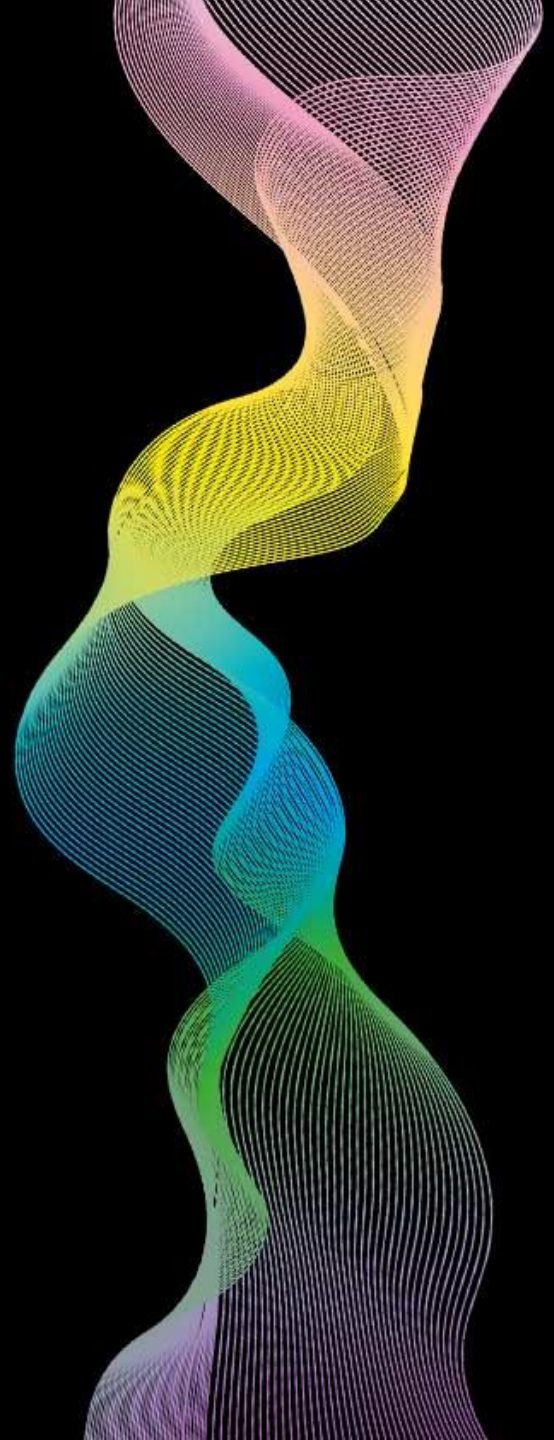
- Light Dependant resistors are used as the main light sensors, to sense which direction and orientation the light comes from. The basic structure of the module is such that two servo motors are fixed to the structure that holds the solar panel. After which, the program for Arduino is uploaded to the microcontroller. The exact working of the project is depicted below.
- LDRs sense the amount of sunlight falling on them. Four LDRs are divided into top, bottom, left and right. For east – west tracking, the analog values from two top LDRs and two bottom LDRs are compared and if the top set of LDRs receive more light, the vertical servo will move in that direction.
- If the bottom LDRs receive more light, the servo moves in that direction. For angular deflection of the solar panel, the analog values from two left LDRs and two right LDRs are compared.
- If the left set of LDRs receive more light than the right set, the horizontal servo will move in that direction. If the right set of LDRs receive more light, the servo moves in that direction.



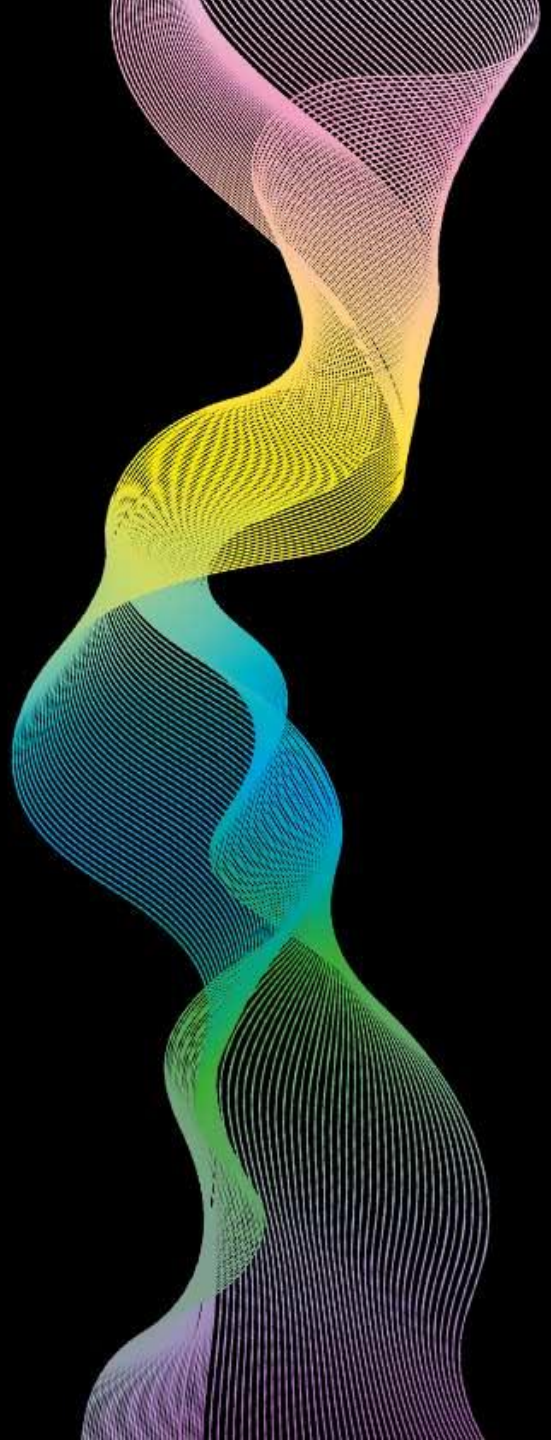
CODE



```
#include <Servo.h>
   //defining Servos
Servo servohori;
int servoh = 0;
int servohLimitHigh = 160;
 int servohLimitLow = 20;
 Servo servoverti;
int servov = 0;
int servovLimitHigh = 160;
int servovLimitLow = 20; //Assigning LDRs
int ldrtopl = 2;
int ldrtopr = 1;
int ldrbotl = 3;
int ldrbotr = 0;
 //top left LDR green //top right LDR yellow
// bottom left LDR blue // bottom right LDR orange
     void setup () {
servohori.attach(10);
 servohori.write(0);
 servoverti.attach(9);
 servoverti.write(0);
 delay(500);
       }
```

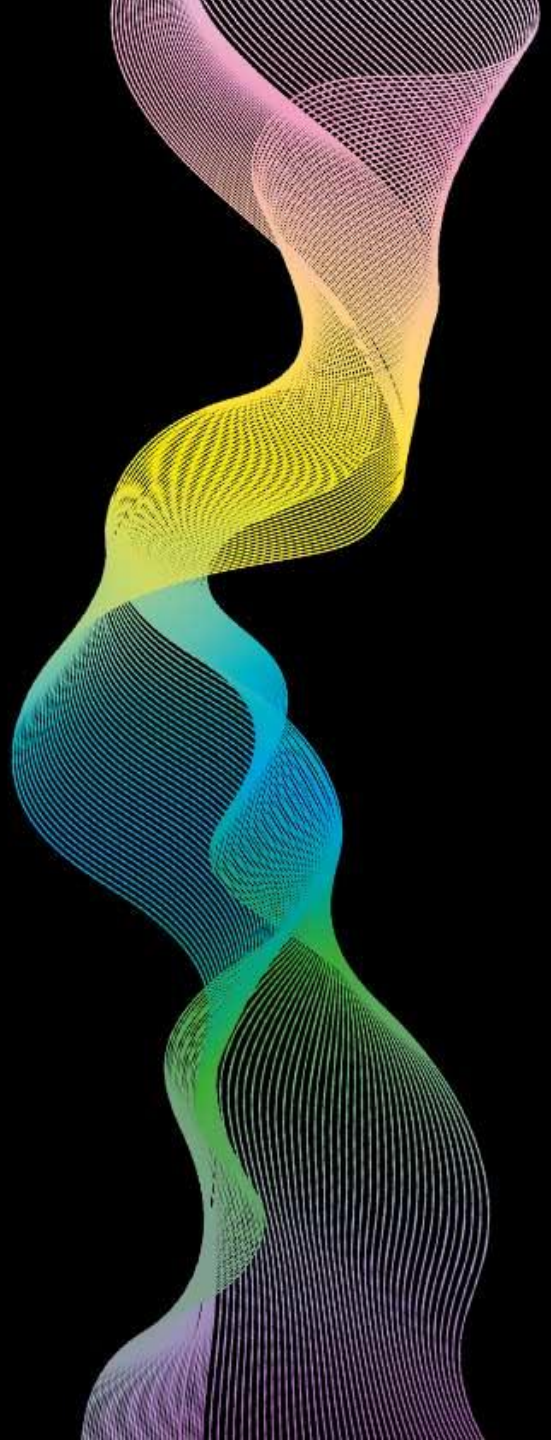


```
void loop() {
servoh = servohori.read();
servov = servoverti.read();
//capturing analog values of each LDR
int topl = analogRead(ldrtopl);
int topr = analogRead(ldrtopr);
int botl = analogRead(ldrbotl);
int botr = analogRead(ldrbotr);
// calculating average
int avgtop = (topl + topr) / 2;
int avgbot = (botl + botr) / 2;
int avgleft = (topl + botl) / 2; //average of left LDRs int
avgright = (topr + botr) / 2; //average of right LDRs
  if (avgtop < avgbot)
  {
servoverti.write(servov +1); if (servov >
servovLimitHigh)
{
servov = servovLimitHigh;
} delay(10);
 }
```



```
  else if (avgbot < avgtop)
  {
servoverti.write(servov -1);
    if (servov < servovLimitLow)
  {
servov = servovLimitLow; }
delay(10); }
else {
servoverti.write(servov); }
  if (avgleft > avgright)
  {
        //average of top LDRs
//average of bottom LDRs

 servohori.write(servoh +1); if (servoh >
servohLimitHigh) {
servoh = servohLimitHigh;
}
```



```
delay(10);
  }
  else if (avgright > avgleft)
  {
servohori.write(servoh -1); if
(servoh < servohLimitLow)
{
servoh = servohLimitLow; }
delay(10); }
else {
servohori.write(servoh); }
delay(50); }
```


THANK YOU