```
//Initialize variables

volatile int p1=0; //Photoresistor 1 value
volatile int p2=0;  //Photoresistor 2 value
volatile int p1Avg=0;
volatile int p2Avg=0;

volatile int timerTime=0;
volatile int timerThreshold = 20; //threshold for timer to go off in
minutes

//Pins into the arduino
volatile int dir=8;  //Direction pin
volatile int pulse=9; //Pulse pin

int checkOn=0; //variable determines whether to take photoresistor
value readings (It's triggered by a timer interrupt)
volatile int timeToCheck=1; //Initially sets time to check to On to
calibrate the solar tracker

volatile int epsilon=0;   //Variable for the threshold value between
photoresitors to trigger the motor
volatile float perc= 5/100;  // Percent of difference between values
to trigger motor (You can modify this till you get a nice response)
volatile int checkTime =10;  //Amount of time in seconds used to
collect sensor data


void setup()
{
  pinMode(0,INPUT);
  Serial.begin(9600);


  pinMode(pulse,OUTPUT);
  pinMode(dir, OUTPUT);
  digitalWrite(dir,LOW);
}

//Main Loop
void loop()
{
  checkTimer();
  if (timeToCheck==1)
  {
    adjustSolarTracker();
    timeToCheck=0; //Turns off "timeToCheck"
  }
}
```

```
void checkTimer()
{
  if (timerTime>= timerThreshold * 60)
  {
    timeToCheck=1;
    timerTime=0;
  }
  else
  {
    timerTime= timerTime + 1;
    delay(1000); //1000 milliseconds ( 1 second) delay
  }
}

void adjustSolarTracker() //This function will run until the
difference between the photoresistor values are within an epsilon
value
{
  takeReadings();
  checkReadings();
}

void takeReadings()
{
    //resets photoresistor values
    int p1sum=0;
    int p2sum=0;

    //Takes readings for the length of "checkTime"
    for( int i =0; i<checkTime; i++)
    {
      p1 = analogRead(A0); //Reads in sensor values
      p2= analogRead(A2);
      p1sum= p1sum +p1; //adds up all the sensor values
      p2sum= p2sum +p2;

      Serial.print("P1sum: ");
      Serial.println(p1sum);

      Serial.print("P2sum: ");
      Serial.println(p2sum);
      delay(1000); //Delays next reading for 1000 milliseconds (1
second)
    }

    //Finds the avgerage of the photoresistor values taken
    p1Avg = p1sum/checkTime;
    p2Avg = p2sum/checkTime;
```

```
    //Determines the threshold, epsilon, for the motor to run
    epsilon = ((p1Avg + p2Avg)/2)*perc;

}

void checkReadings()
{
  if ( p1Avg-p2Avg > epsilon) //Checks whether the difference between
p1 and p2 is too large
     {
      //Runs Motor CW
        digitalWrite(dir,HIGH); //sets motor direction CW
        runMotor();
        adjustSolarTracker(); //Checks photoresistor readings again
      }

  else if ( p2Avg-p1Avg> epsilon)  //Checks the whether the difference
between p2 and p1 is too large
     {
      //Runs Motor CCW
        digitalWrite(dir, LOW); //sets motor direction CCW
        runMotor();
        adjustSolarTracker(); //Checks photoresistor readings again
      }
}

void runMotor()
{
  for (int x=0; 50000; x++)
  {
    digitalWrite(pulse, HIGH);
    delayMicroseconds(20);
    digitalWrite(pulse, LOW);
    delayMicroseconds(20);
  }
}
```