

# sunflower

The sunflower turns with the sun. In a large field, they all point in the same direction.



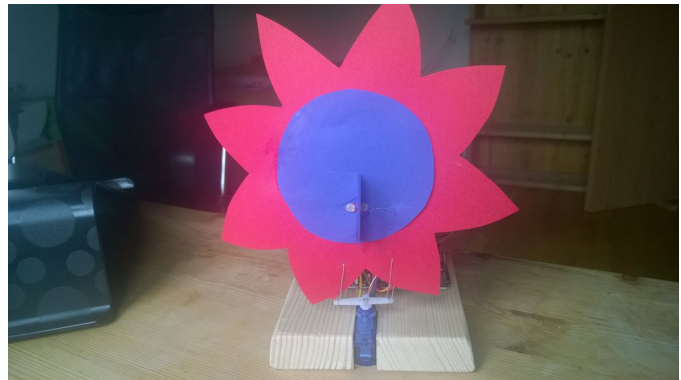
How does it work? Experiments showed that the plants control their alignment via the growth of the stem. Accordingly, the sun-facing,

eastern half grows during the afternoon, so that the tip points to the west. In contrast, the western half grows faster at night - both with the help of the growth hormone auxin.



We can also make a sunflower like that. With wire, paper and a few Arduino components.

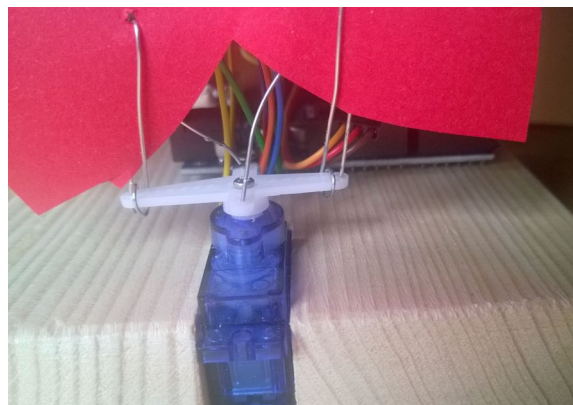
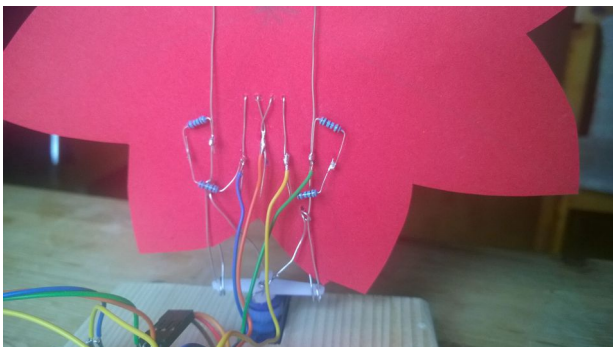
... and this is how it looks like:



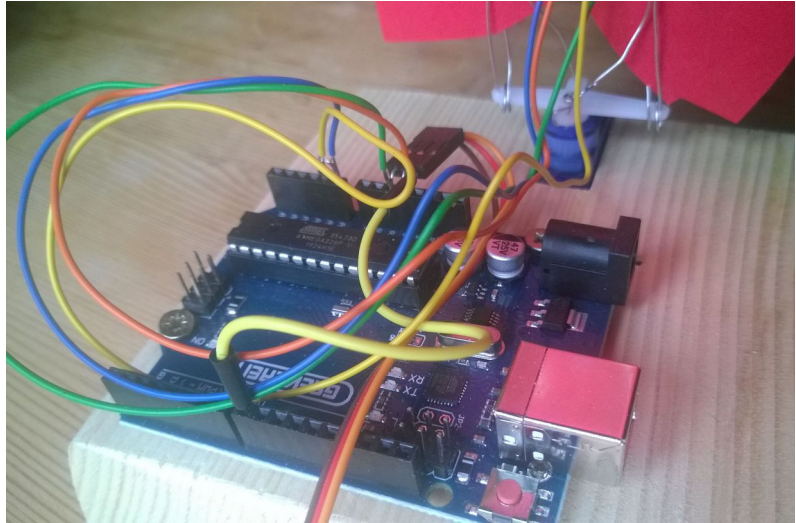
Two photo resistors look ahead. Between the two is a web made of paper, about 1 cm high.

The petal (red) is held by wires attached to the white plastic oar from the servo are attached. When the servo turns, the flower rotates with the photo resistors.

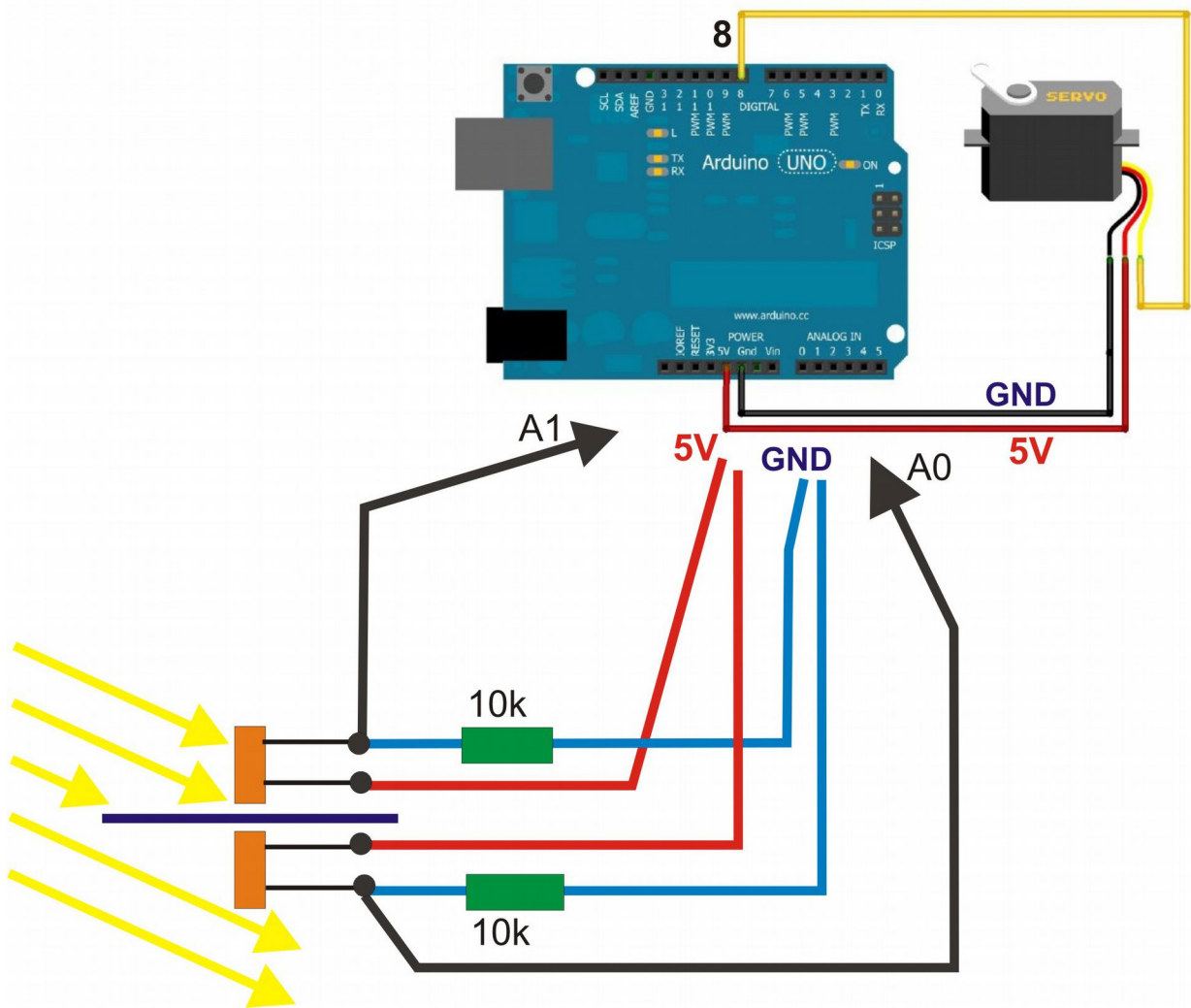
It looks like this from behind.



We see the wiring in the following picture.



In this example the 10k resistors are soldered directly to the back of the flower. But it can also be put on the breadboard.



Das folgende Programm einfach kopieren, testen und zum Arduino hochladen:

```
// Die Sonnenblume dreht sich immer zum Licht hin.

#include <Servo.h>           //Die Servobibliothek wird aufgerufen.
                           //Sie wird benötigt, damit die Ansteuerung des Servos vereinfacht wird.
Servo servoblau;           //Erstellt für das Programm ein Servo mit dem Namen „servoblau“

int Wi;                    // Winkelstellung Servo
int MW1;                   // Messwert vom Fotowiderstand 1
int MW2;
int Licht1;                // Licht vom Fotowiderstand 1 gefiltert
int Licht2;

void setup()
{
  servoblau.attach(8);     //Servo wird an der Steuerleitung (gelb) mit Pin 8 verbunden.
                           // Mit „servoblau.write(Grad)“ wird das Servo angesteuert.
  Wi = 90;
  servoblau.write(Wi);    //Start-Position 1 ansteuern mit dem Winkel 90°
  Serial.begin(9600);
}

void loop()
{
  MW1=0;
  MW2=0;
  for(int index=0; index <20; index++) // nur zur Messwertglaettung
  {
    MW1 = MW1 + analogRead(A0); // Messwert links
    MW2 = MW2 + analogRead(A1); // Messwert rechts
  }

  Licht1 = MW1 / 20;        // Messwertglaettung
  Licht2 = MW2 / 20;

  int Sum = Licht1 + Licht2; // Summe Licht
  int Dif = 0;              // Differenz Licht recht-links
  int Null = 0;
  if (Sum > 1400)           // wenn zu wenig Licht, dann stehen bleiben
  {
    Dif = Licht1 - Licht2; // wenn genug Licht, Differenz bilden
  }

  if (Dif > 10)
  {
    if (Wi < 150) { Wi = Wi + 1; } // Winkelstellung vergroessern
  }
}
```

```

if (Dif < -10)
{
  if (Wi > 30) { Wi = Wi - 1; } // Winkelstellung verkleinern
}

servoblau.write(Wi); // neue Position ansteuern

// Durch die serielle Schnittstelle werden die folgenden Werte zu PC uebertragen
// Mit "Seriell Plotter" koennen sie als Diagramm dargestellt werden
// Darin sieht man, wie sich die Winkelstellung zwischen 30° und 150° bewegt
// Zur Kontrolle kann notfalls auch Licht und Summe dargestellt werden
// Serial.print(Licht1/10); Serial.print(" "); Serial.print(Licht2/10);
// Serial.print(" "); Serial.print(Sum/20);

Serial.print(Wi); Serial.print(" "); // Winkelstellung anzeigen
Serial.print(Dif); Serial.print(" "); // Differenz anzeigen
Serial.print(180); Serial.print(" "); Serial.println(Null); // Hilfslinien anzeigen

delay(100); // eine groessere Zahl bewegt langsamer
}

```

