



## solar tarcker

by **NiranjnV** on May 23, 2016

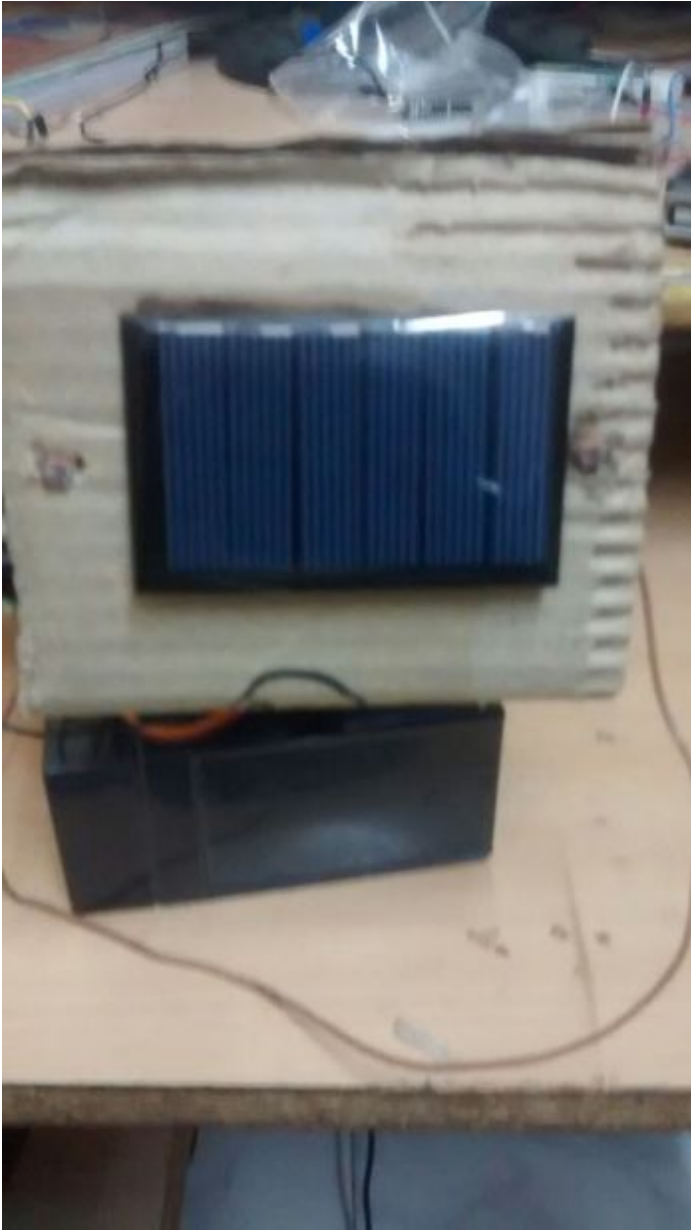
### Table of Contents

solar tarcker .....	1
Intro: Solar tarcker .....	2
Step 1: Steps to build a Sun tracking solar panel: Step-1: .....	2
Step 2: Step 2-Creating the base for solar panel: .....	4
Step 3: Step 3-connecting the circuit: .....	4
Step 4: Step 4-Programming the Arduino: .....	5
File Downloads .....	5
Step 5: Step 5-Testing: .....	5
Advertisements .....	6

## Intro: Solar tarcker

What is a Sun Tracker?

A sun tracker is a automatic platform for solar panel which always turns its face towards the sun throughout the day like a sun flower by using some photo sensors like Photoresistors.

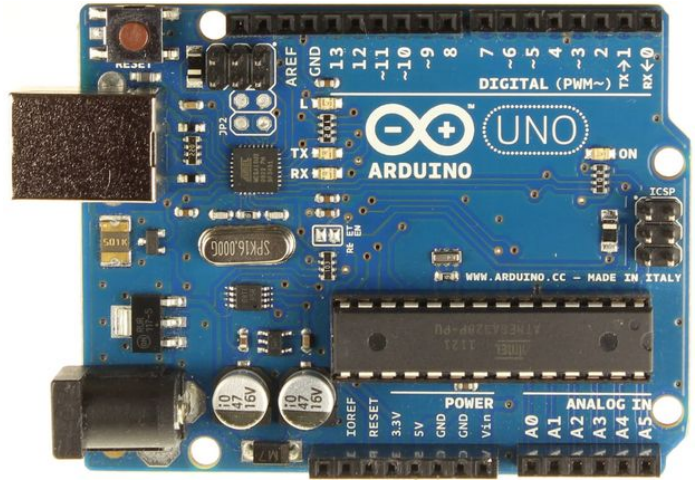


### Step 1: Steps to build a Sun tracking solar panel: Step-1:

First of all you have to gather all the materials required to build the sun tracking system.

Materials required:

1. Solar panel
2. Arduino
3. Servo motor
4. Photoresistors (two)
5. Resistors (2 x 3K ohms)
6. Connecting wires and
7. 9 V battery to power the Arduino
8. cardboard



## Step 2: Step 2-Creating the base for solar panel:

In this step make a platform to place the Solar panel and photoresistors.

Take a cardboard sheet and divide it into three equal parts (don't cut the sheet into three parts just mark some lines) then bend the cardboard at the markings and stick the ends with a tape. The final piece will look like the one in the following picture.

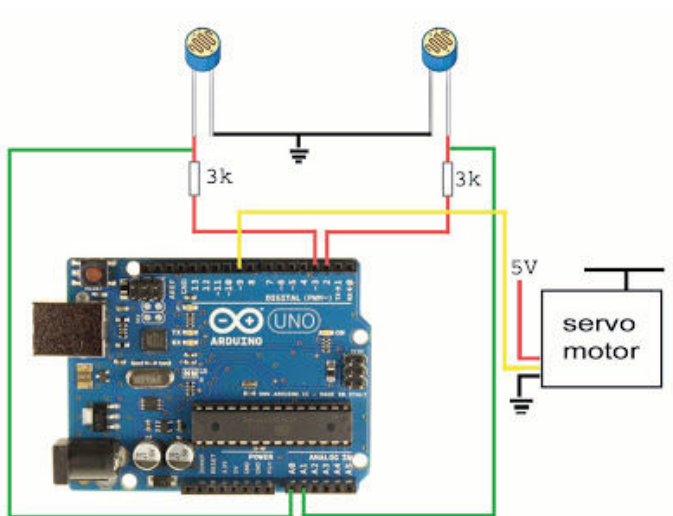
Make two holes at the center of both the sides (right and left) of the cardboard to place the Photoresistors (sensors). After that insert the two legs of the photoresistor into the holes made in the previous step then connect some long wires to the legs of both the photoresistors by soldering. Make sure that the wires connected to the photoresistors are long enough so that they don't come out of the breadboard when the base of Solar panel turns completely towards its right or left.

Now attach the Triangular base on the axis of the Servo motor with some double sided tape. After that place the solar panel on the Triangular base and stick the panel to the base so that it won't fall down when the servo rotates. Now attach the whole setup (Servo motor along with the triangular base and Solar panel) on a stationary base. I had used an cardboard box as its base. left some place on the base to place the breadboard to make the connections for the servo, sensors and Arduino.



## Step 3: Step 3-connecting the circuit:

The following picture shows you the connections for the photoresistors and Servo motor with the Arduino.



#### Step 4: Step 4-Programming the Arduino:

Now its time to write some code for the Arduino to make it work. Here is the simple code for the Arduino microcontroller. I tried to make the code as simple as possible.

Now, upload the code into the Arduino by using the Arduino IDE. It will take only few seconds. Make sure that you have selected the correct Arduino board and the chip used on your Arduino board. My board is Arduino UNO with ATmega328 chip.

How the code works:

In the above program the main if block (if(valueOne<950 || valueTwo<950)) executes only when light is present. The inner if-else is used to know in which side the sun is present (right side or left side). If the sun is present towards the right side of the solar panel the servo moves the solar panel towards the right side in steps of two degrees until the difference of the values from two sensors is less than 100 units. If the sun is present towards the left side of the solar panel then the servo moves towards its left in steps of two degrees until the difference of the values from two sensors is greater than -100 units.

#### File Downloads



sketch.ino (1 KB)

[NOTE: When saving, if you see .tmp as the file ext, rename it to 'sketch.ino']

#### Step 5: Step 5-Testing:

The final step is to test your own Sun tracking solar panel system in the presence of sun or under a light source like touch light. Make sure that the wires connected to the two sensors are long enough to reach when the base of the solar panel moves towards its left and right.



