SOLAR TRACKER PROJECT

Engr 133 - Mechatronics I

Due Week 16 at START of class

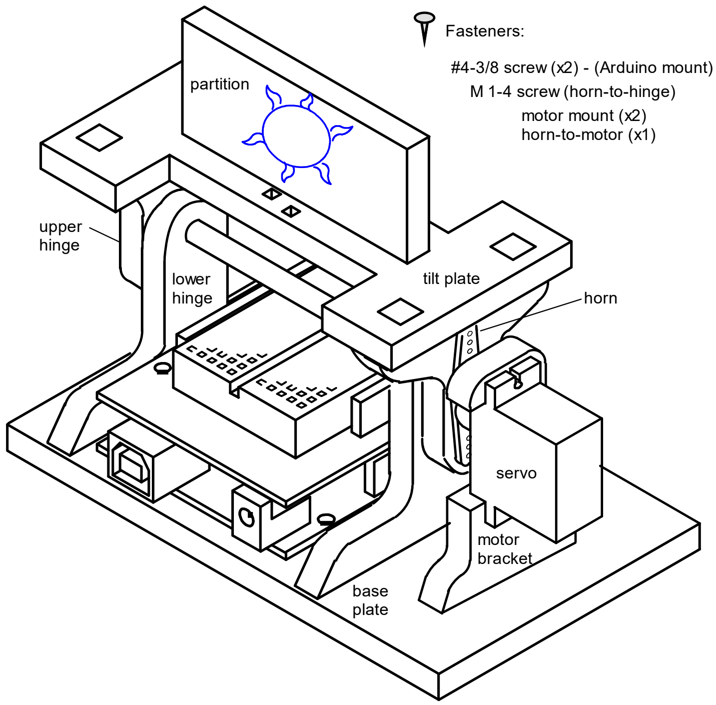


Figure 1 - Solar Tracker completed

1.1 SUMMARY

In this project you will design and fabricate a 1-axis solar tracker controlled using an Arduino micro-controller. The mechanical components can be fabricated using the lab laser cutter. Students will demonstrate their project to the instructor.

1.2 MATERIALS LIST

STUDENTS PROVIDE:

1. Arduino UNO kit - items needed from kit:
   - Arduino UNO R3 or Arduino Mega 2560 microcontroller
   - Breadboard shield
   - micro-servo SG-90 (180 deg, 4.8 - 6v Vin; current (no load): 100 mA, current idle 5 mA)
   - screws for micro-servo (2 to mount, 1 to mount horn to servo, SAVE these from your Arduino kit!)
   - resistors (2k Ohm, 1/4 W) (x2)
2. Wood (Home Depot, 1/4" thick x 5.5" x 24" Poplar) (STUDENTS PROVIDE)

INSTRUCTOR PROVIDES:

3. Photo-transistors (x2) ("photo-Q's") (850 nm T1 3/4, $1.49) (parallax.com/product/350-0029)
   - detects light in the 450 – 1050 nm range (responds to visible light thru low IR range)
   - (use instead of photo-resistors since leads are thicker, so more secure in F-connector of dupont wire)
4. Wood shaft (3/16 dia x 4 inches) - for hinge mechanism
5. Screws, wood, #4 - 3/8 (to mount Arduino board) (x2)

6. Mini crews, wood, M1 x 4 (x2) (to mount horn to hinge) (use 1.0 flat head jeweler's screwdriver)
7. Hook up wire (solid, 22 AWG) - use red for 5V, black for GND, other colors for other connections
8. Dupont wire (short 8 cm, M-F) - to attach sensor leads to breadboard

OPTIONAL ITEMS

1. Solar panel (optional), PRT-13781 from Sparkfun, 2W, 5.3 x 4.4 x 0.2 in, $29;
   Vpeak 6.0V, Ipeak 378 mA, Ppeak 2.27W (www.sparkfun.com/products/13781) (not needed)
2. Power supply (motor shield, DC power supply or 5 AA batteries & holder) (to operate w/o computer)

1.3 PROJECT REQUIREMENTS

1. Complete fabrication of mechanical components (use laser cutter).
   Use provided design (EPL posted) or do your own design in Solidworks (OPTIONAL).
2. Complete circuitry & connections (Arduino, breadboard, wiring, sensors, motors)
3. Program Arduino using the code below. Tune various parameters to optimize performance.
4. Demonstrate system functionality to instructor.
5. NO purchased systems allowed.

Note a solar panel is not needed for the solar tracker to operate.

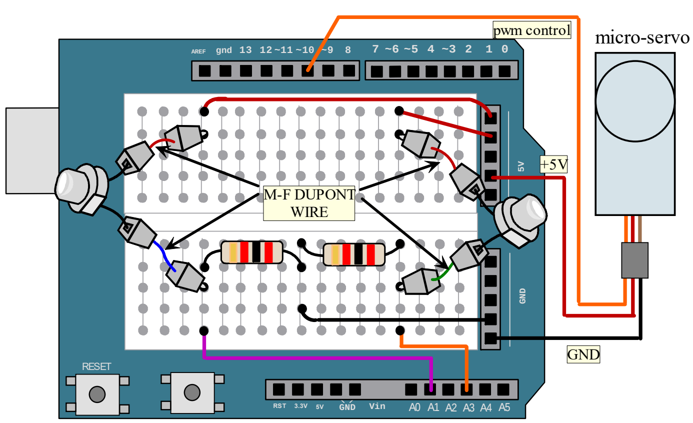
1.4 CONSTRUCTION & CIRCUITRY

You may assemble at home but use the laser cutter to make the mechanical components. Construction may also involve using hand tools and sandpaper. IMPORTANT - work carefully, think about what you are doing, and do NOT GET HURT.

Note – Arduino is powered with 6-12VDC. The hobby servo motors typically require 4.8 - 6V power.

The end of this document provides a guide for assembling the solar tracker.

The breadboard and servo circuit is shown below.



POWER MOTOR WITH ARDUINO
(motor must NOT be loaded)

Figure 2 - Arduino with light-detect circuit and servo connections

1.5 PROGRAMMING

Use the code in this document. It is nearly the same as "Servo PID" code but there is a CONSTRAIN range to prevent the mechanism from "CRASHING". Systems using feedback control can sometimes behave unexpectedly and jump to extreme positions. Such behavior can cause the mechanism to "CRASH". The constrain statement may help prevent this.

Note we found that it is better to MAP FIRST THEN FILTER to get full range of motion.

```
// solar_tracker.ino --------------------
// use PD controller for solar tracker
// ctl variable is DIFF betw 2 lite detectors
// controller uses deadband
// tune performance using: Kp, Kd, w, & loop delay time


#include <Servo.h>
Servo myservo;

int servoPin = 10;
int sensorPin1 = A1;
int sensorPin2 = A3;

void setup() // ------------------------------------------
{
  Serial.begin(9600);
  myservo.attach(servoPin);
}

void loop() // ---------------------------------------------
{
  //int servoAngle;

  int sensor1 = analogRead(sensorPin1);     //0 - 1023
  int sensor2 = analogRead(sensorPin2);
  int diff = sensor2 - sensor1;             // -1000 to +1000, swap if nec (1-2)
  float startAngle = 90.0;

  float u, servoAngleFilt;
  static float servoAngle = startAngle;
  unsigned long tnow = millis();                  // track time for U pause

  u = computePID(diff);

  servoAngle += u;                          // add control to servoAngle
  servoAngleFilt = exponFilter(servoAngle); // smooth sensor data

  // constrain to prevent crash
  servoAngleFilt = constrain(servoAngleFilt, startAngle - 60, startAngle + 60);

  myservo.write((int)servoAngleFilt);             // command motor

  Serial.print(diff);   Serial.print("  ");
  Serial.println(u);

  delay(40);
}
```

```
float computePID (int x) //-------------------------------
{
  float Kp = 0.008;       //P gain - ADJUST THIS
  float Kd = 0.5;         //D gain - ADJUST THIS
  float v, u;
  float xd = 0.0;
  float vd = 0.0;

  unsigned long t = millis();
  static unsigned long to = 0;  // static - remembered, not re-initialized
  static float xo = 0;
  int minerror = 5;             // dead band

  v = (float) (x - xo) / (t - to);          // compute veloc

  xo = x;             //reset old xo,to for next veloc calc
  to = t;

  u = Kp * (xd - x) + Kd * (vd - v);        // PD controller

  // dead band (u=0 if close to desired)
  // OR if t < 1.0 sec - stop jump?
  if ((abs(x - xd) < minerror) || (t <= 1000))
  {
    u = 0.0;
  }

  return u;
}

int exponFilter(int newData) //-----------------------------------
{
  static int lastData = 0;
  float w = 0.80;   // w = filter strength (big w = smooth, slow, low=fast, noisy)
  lastData = w * lastData + (1 - w) * newData;
  return lastData;
}

//end -----------------------------------------------------------
```

1.6 TESTING & EVALUATION

The tracker tilt plate should follow the light of a flashlight. Move the light back and forth and watch the responding movement. The platform should track but it may lag the light movement a bit (ie - it need not be super responsive). Now hold the light still. The platform should stop while facing the light and not oscillate or twitch (too much).

Adjust the Kp gain, Kd gain, filter "w", and possibly the delay on the loop function to obtain: proper tracking, smoothness of movement, and no excessive oscillation. Here are the basic effects of those variables:

Increased:	Effect
Kp	more responsive, more oscillations
Kd	less oscillation but if too high creates instability
w	smooth movement, less responsive, less oscillations
loop delay	smooth movement, less responsive, possibly more "steppy" (move in steps)

DEBUGGING

If the platform avoids light, alter this line of code: int diff = sensor2 - sensor1;
Swap the variables: "sensor2 - sensor1" to "sensor1 - sensor2".

If the tracker does not work, GO BACK. First verify the pins in code match the physical circuit. Ensure the light detectors work using code from the last exercise of the LIGHT DETECTOR LAB (differences smoothed). Use the SERIAL PLOTTER. Also verify the servo works by running the "servo to position" code from the SERVO lab (but limit your commanded angles to 40 to 140 degrees to avoid crashing the upper platform).

1.7 GRADING

Credit for
1. Performance - project completion, performance, demonstration done, done on time

Deductions for:
1. Performance issues (using measures above), plus deduction for oscillations,
2. No in person demonstration of project
3. Sloppy (eg - wiring), incomplete work
4. Wire color coding not followed - red wire (5V), black wire (GND)
5. Hook-up wire not used (except M-F dupont for connecting photo-Qs).
6. Missing artwork on partition.
7. Poor effort, attitude, lab practices (safety issues, not cleaning up, etc.)
8. Poor project planning - poor weekly progress, finishing at last second

An optional nicer solar tracker is a TWO-AXIS tracker. This involves adding a second "axis" so the tilt panel can orient in any direction 3D space. This is a more complicated mechanical mechanism.

1.8 ASSEMBLY GUIDE

1.8.1 Mechanical Design

Students may use the provided design (use the provided EPL file of the laser job).
Students may also design their own tracker if they feel they have the ability.

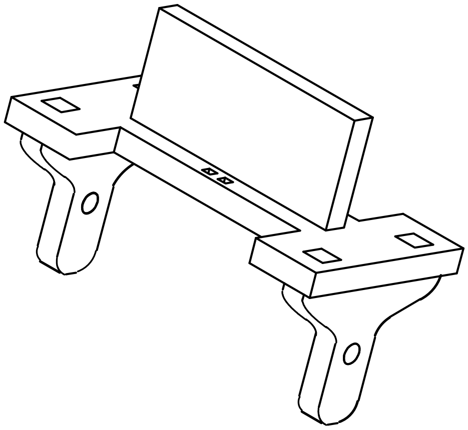
1.8.2 Laser Cut Parts & Make Upper Assembly

Laser cut all parts. Also ADD ARTWORK (raster or vector engrave) on the partition plate.
Dry fit components (no glue) to verify fit. Verify the hinge rotates freely.
Later you will assemble parts with glue. Use thick glue (thick super-glue or wood glue), NOT thin super glue.
Do NOT get glue on desks or computers. Place scrap cardboard or silicone pads below your work.

Glue and assemble the "upper assembly" (tilt plate, upper hinges, partition). See below left.

1.8.3 Build Circuit & Lower Assembly

Mount Arduino without the Bb shield to the lower plate using the 2 provided screws. See below right. While Arduino has 4 holes, we'll only use 2 of them.



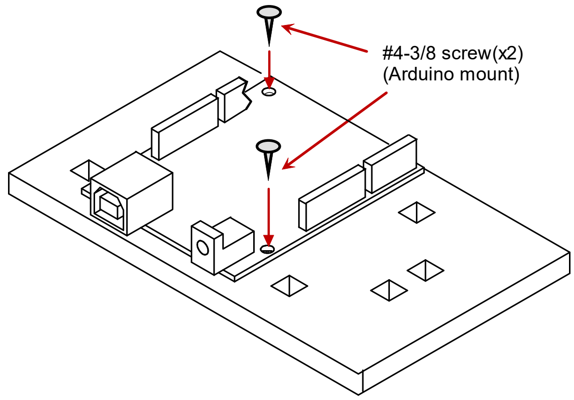


Figure 3 - Upper assembly and mounting Arduino to lower plate

1.8.4 Electrical Assembly

Assemble the Bb shield onto Arduino. Build the light-detector & motor circuit and connections as shown. Use only hook-up (spool) wire that is cut & stripped to length so the wire is FLUSH with the breadboard surface. The exception is to use short M-F (male-female) dupont cable to connect photo-Qs to circuit. Resistors must also be FLUSH to the Bb surface (cut their leads). Use red wire for 5V supply. Use black wire for ground (GND). The other wires can be any color but not red or black. Also color code the wire to the servo (red = 5V, black = GND, orange = PWM signal).

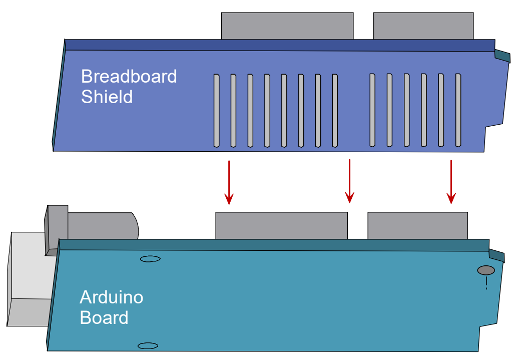


Figure 4 - Assemble breadboard shield to Arduino

The leads of the photo-Q will insert into the female connector of the M-F dupont cable. This connection may not be really secure so ensure the dupont cables are long enough to avoid pulling. Note - photo-resistors with their very thin leads will be even more loose. Bending the photo-R leads into a "wavy" shape using needle-nose may help create a tighter fit. Let the instructor know if the connection is not working. We MAY have to create a solder connection (talk to the instructor before trying to solder in lab).

Verify the light-detector and motor work by running previous exercises.
Then remove the photo-Qs as they'll mount to through the tilt plate.

1.8.5 Servo Assembly & Centering

Mount the motor to the motor mount bracket using 2 screws that came with your motor.
IMPORTANT - mount so the motor flange is on the opposite side of Arduino.
Run screws from that side.

Load the "servo to position" code from the servo motor lab to Arduino. Command the servo (whose range is 0 to 180 degrees) to 90 degrees - its center position. Remove the motor from power. Push the 2-sided horn onto the motor shaft so it aligns with the motor body centerline. Screwing the horn to the motor shaft is not needed. Note the goal here is to have the horn in line with the motor centerline while it is at the 90 degree position. Screw the servo to the motor bracket. This is the servo sub-assembly.

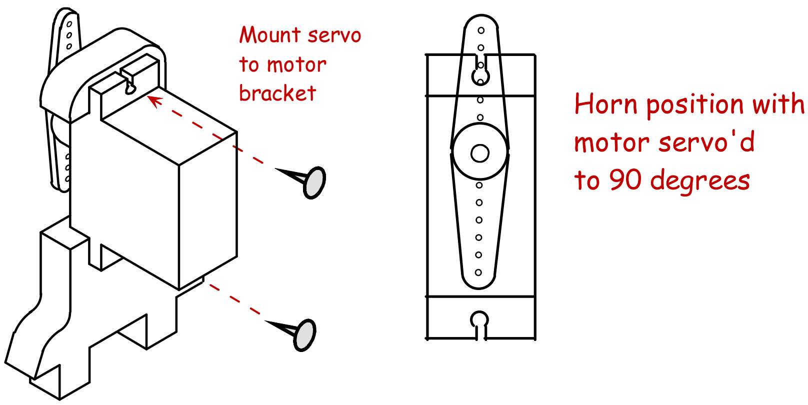


Figure 5 - Servo sub-assembly and servo horn in proper orientation with motor servo'd to 90 degrees

1.8.6 Complete Lower Assembly

Glue the lower hinges and servo sub-assembly (motor bracket, servo, horn) to the bottom plate.
This is the "lower assembly".

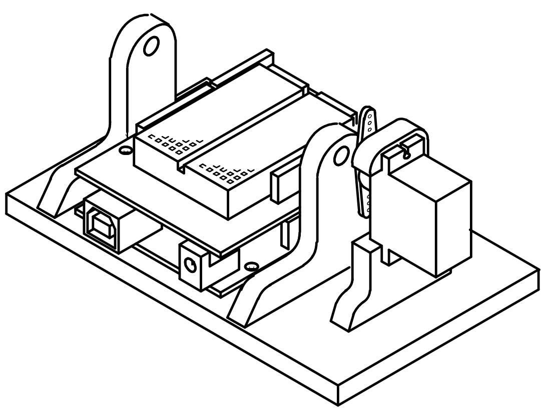


Figure 5 - Lower assembly shown

1.8.7 Assemble Upper & Lower Assemblies

Assemble the lower and upper assemblies using the wood shaft.
Carefully slip the upper hinge past the motor horn.
Move the hinge by hand to verify the hinge assembly rotates freely.
If it binds, you must fix this before continuing.
Also verify the axis of the hinge assembly lines up (co-linear) with the output motor shaft.
They MUST align or the mechanism will bind and overload the motor.

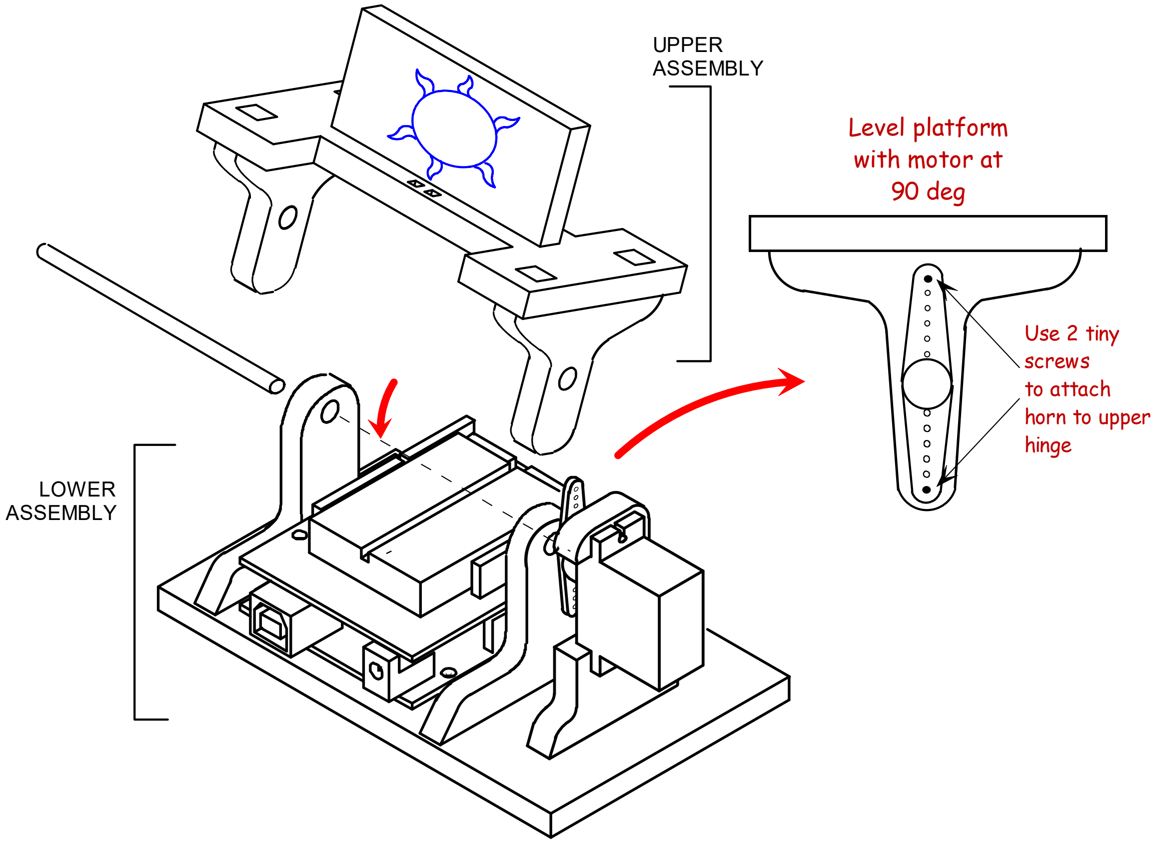


Figure 6 - Assemble upper and lower assemblies. Mounting horn to upper hinge

Set the tilt plate LEVEL and screw the horn to the upper hinge using the 2 mini screws. This assures that the tilt platorm is ~ LEVEL when the servo is at 90 degrees. Rotate the tilt plate SLOWLY (with Arduino off) by hand to verify it moves freely while connected the servo.

1.8.8 Connect Photo-Transistors (Photo-Q's)

Connect photo-Qs to your light detector circuit using M-F (male-female) dupont cables. Slip the female connectors thru the square holes in the tilt plate. Glue the female connector to the tilt plate with one small drop of glue (use very little glue so it's possible to disconnect later if needed). Bend the male connectors on the Bb if needed to make more vertical space.

Trim the leads on the photo-Qs. Keep the longer lead longer after trimming.

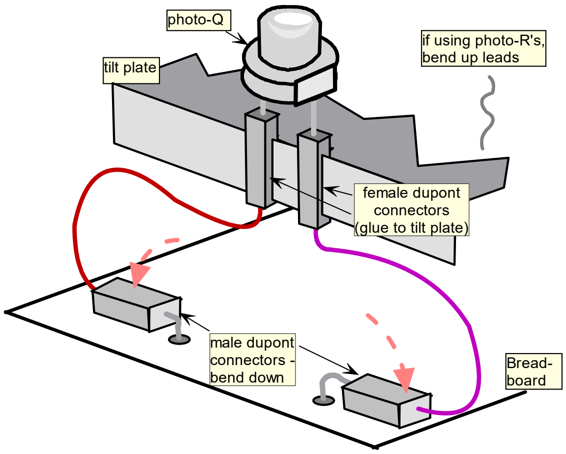


Figure 7 - Wiring of the photo-transistor with dupont wire. Female connectors fit into square holes in tilt plate.

1.8.9 INSTRUCTOR NOTES (students can ignore)

In future - work on RC light sensing circuit - better range, but non-linear