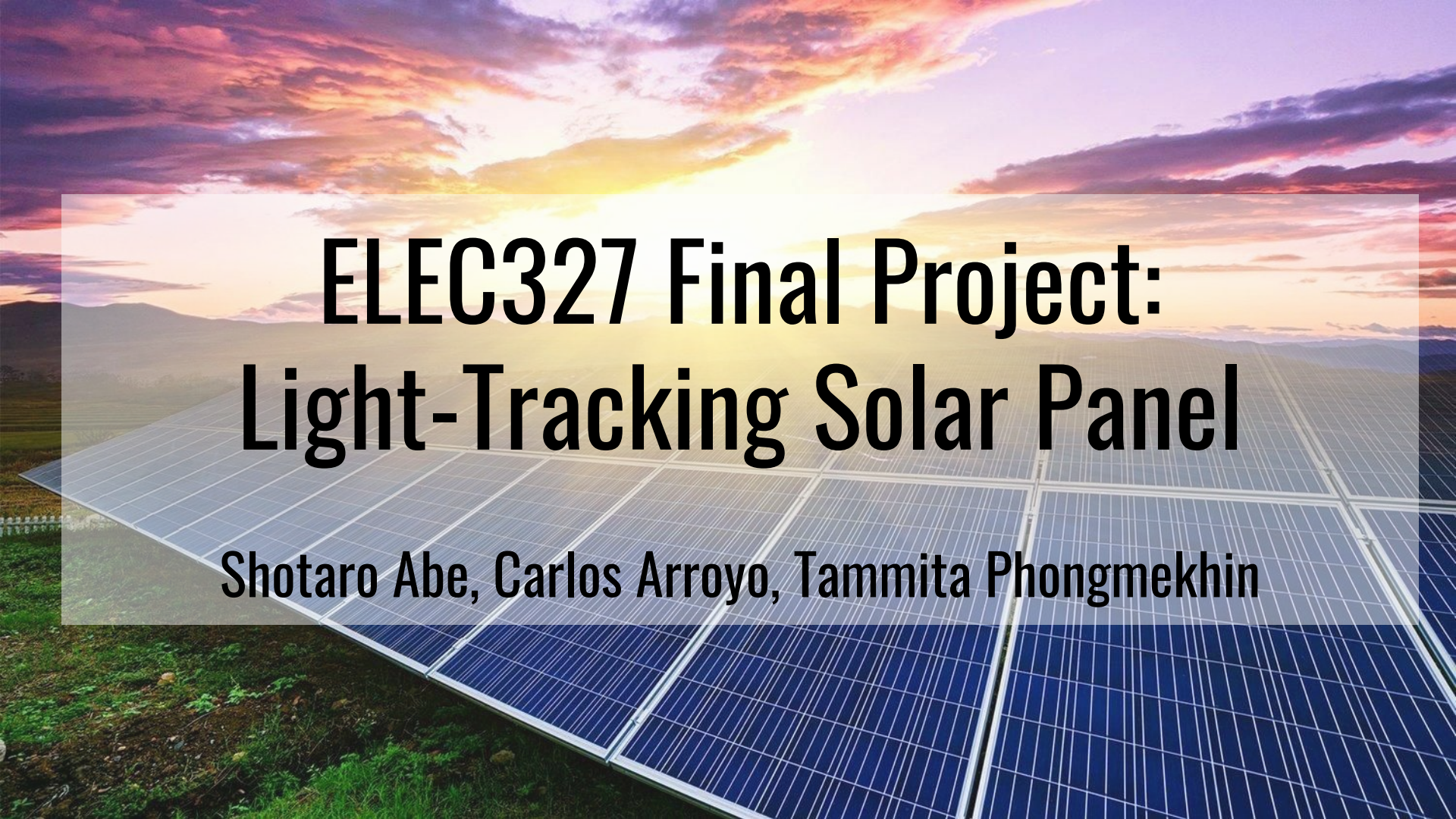


ELEC327 Final Project: Light-Tracking Solar Panel

Shotaro Abe, Carlos Arroyo, Tammita Phongmekhin

Motivation

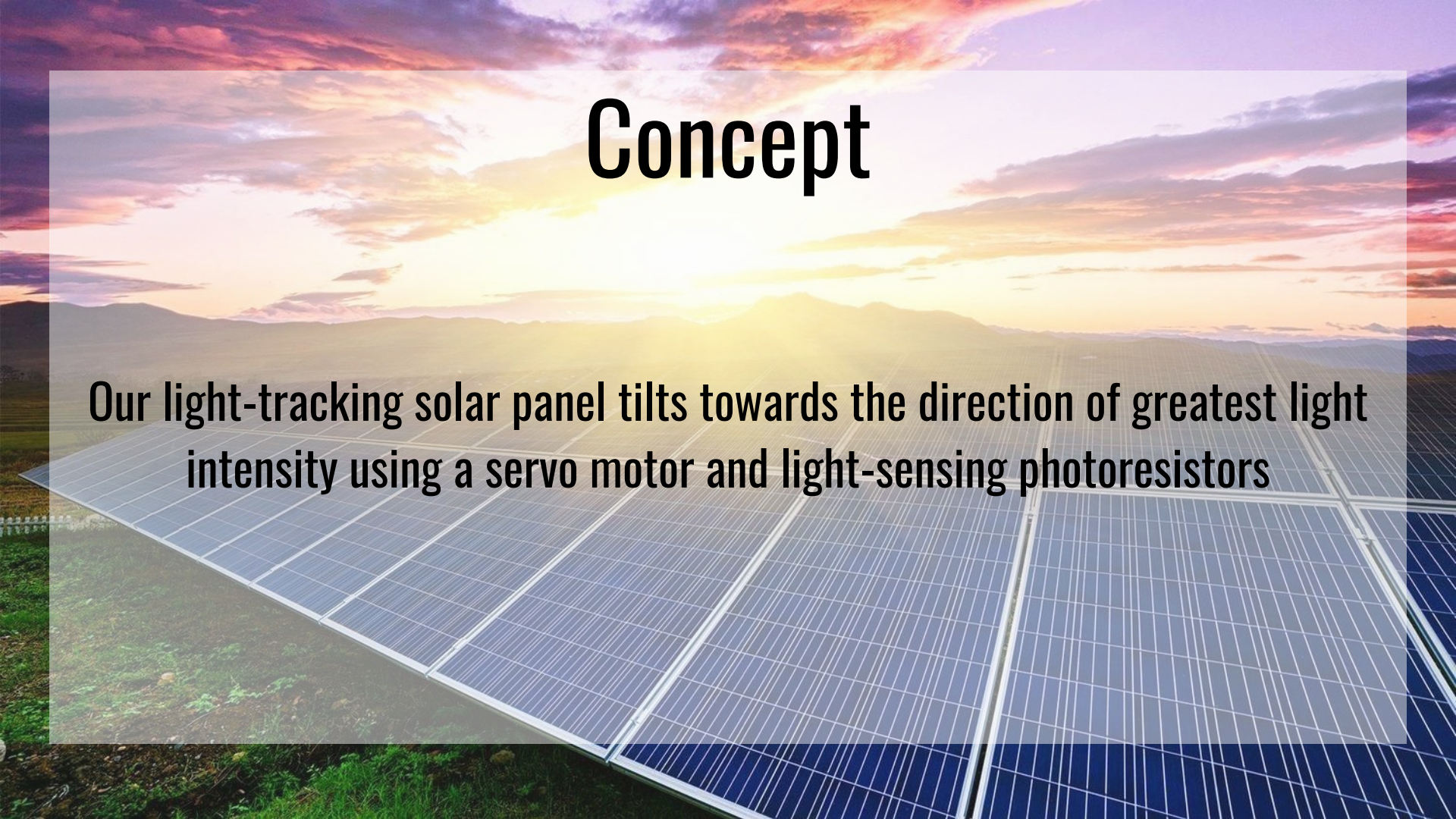
Solar energy is a clean, renewable, and inexhaustible energy source

A current problem is that we are able to harness only 0.001% of the total available solar energy

A solar panel whose surface points in the direction with greatest light allows more solar energy to be harnessed

Concept

Our light-tracking solar panel tilts towards the direction of greatest light intensity using a servo motor and light-sensing photoresistors



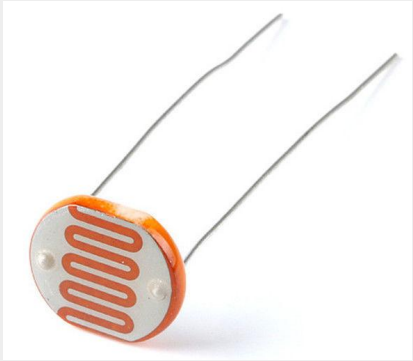
Primary Components

MSP430G2553 microcontroller

Mini solar panel

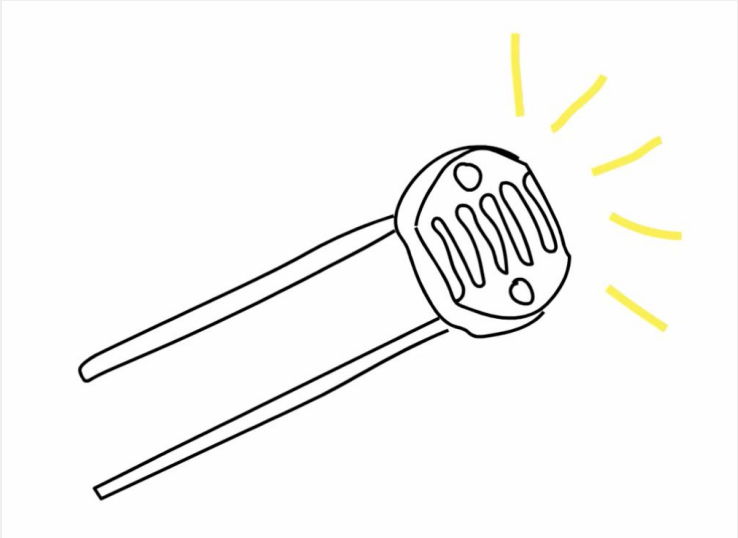
180 degree servo motor

Photoresistors



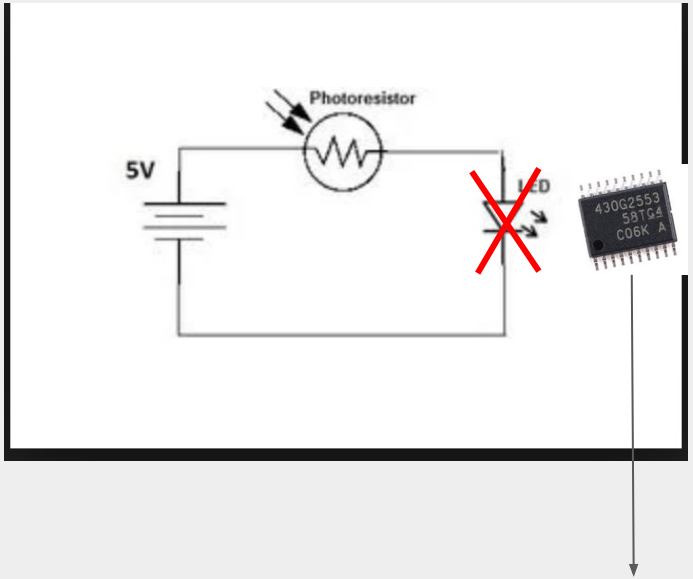
Photoresistor

- A photoresistor is a variable resistor whose resistance changes based on light
- The more light it receives, the lower the resistance
- Used as tool to find intensity of light



Photoresistor

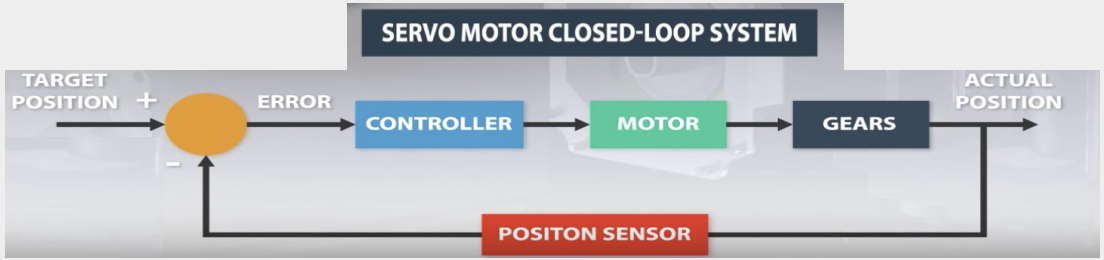
- Often photoresistors are used to control LED's
- Here we use it to control logic of MSP430
- Can add potentiometer to increase sensitivity



Control Servo Based on Input From Photoresistor

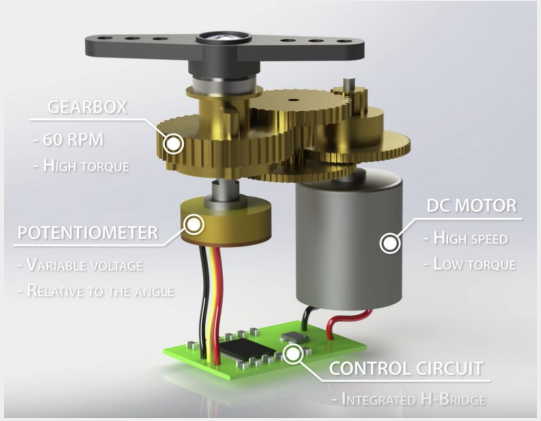
Servo Motor

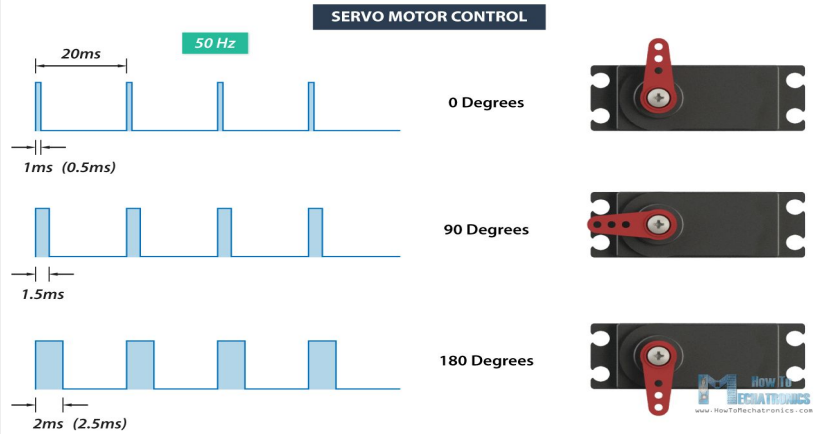
- Often used in projects to control wheels, levers, etc.
- Closed loop system
- Feedback system to control output



Servo Motor

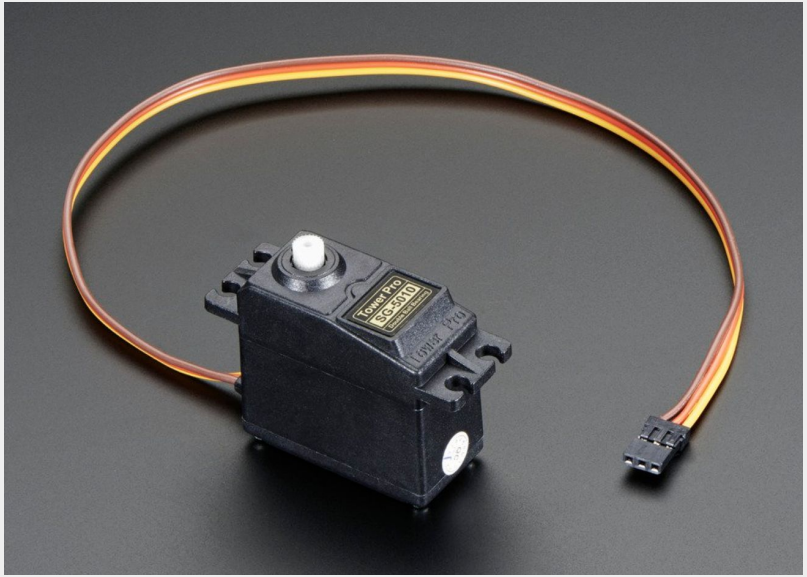
- Parts inside include
  - Circuit
  - Potentiometer
    - Measure error
  - Gears
  - DC Motor
- Input is a PWM Signal



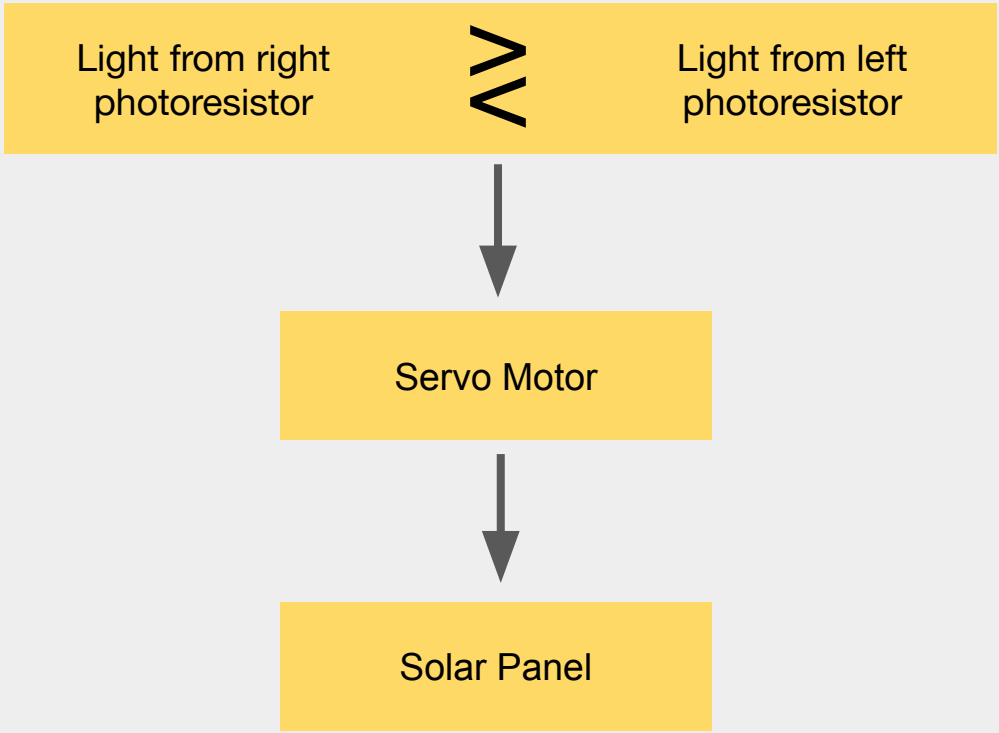


Servo Motor

- We used a TowerPro SG-5010
- Specifications:
  - Can rotate approximately 180 degrees
    - 90 in each direction
  - Power: 4.8V - 6V DC
    - 5V works well
    - Good for MSP430 use
  - Torque:
    - 4.8V: 76.38 oz-in
    - 6.0V: 90.27 oz-in
  - Speed:
    - 4.8V: .19 sec/60°
    - 6.0V: .15 sec/60°
  - 3 Wire connection



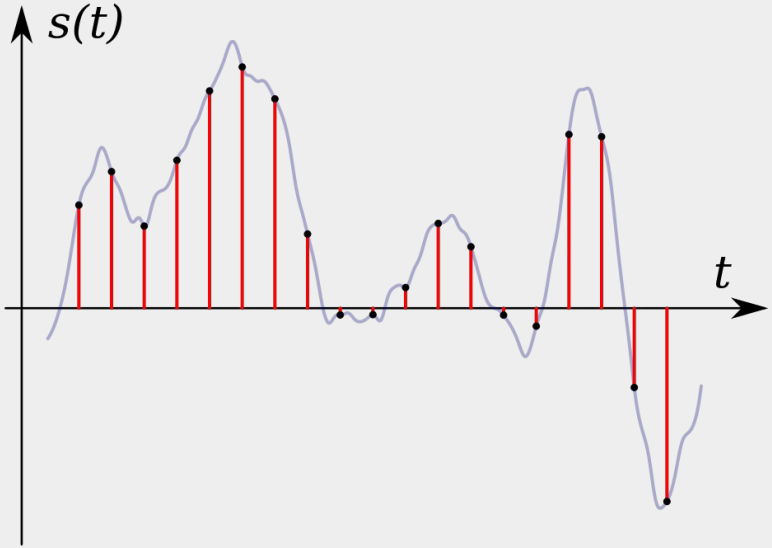
System


Light from right photoresistor
≷
Light from left photoresistor
Servo Motor
Solar Panel

Part I: Light-Sensing

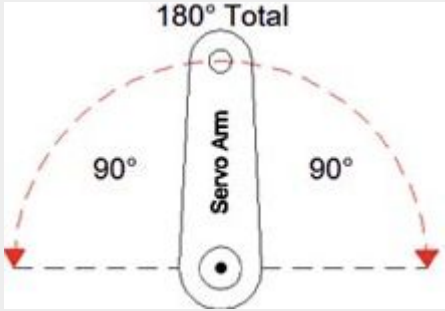
- One photoresistor placed at each end of the board
- Use the ADC10 module to convert the analog light signal detected by each photoresistor into a digital light value
- Compare each value and determine which is larger
- Larger value means more light
- Size of value determine degree of angle

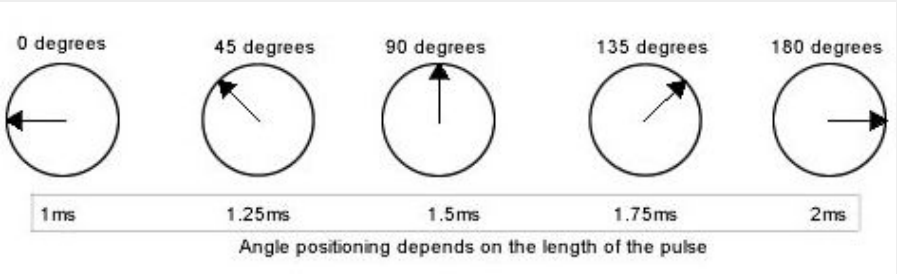
Analog to Digital Conversion



Part II: Servo Motor Control

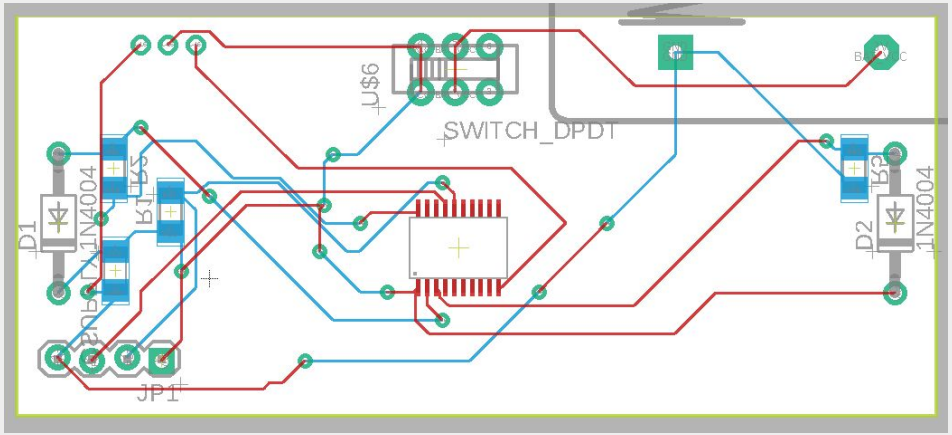
- Digital light values from the 2 photoresistors are compared and used to determine the frequency of a PWM signal generated by the MSP430
- PWM signal controls the angle of rotation of the servo motor
- Solar panel is attached to the motor and tilts according to the motor's rotation





Board Schematic and Wiring

- Parts
  - MSP430G2553 Microcontroller
  - 2 Photoresistors
  - 180 degree servo motor
    - TowerPro SG-5010
  - 3 resistors
  - 10 microfarad capacitor
  - Switch
  - 3.3V battery
- We used pins 1.0 and 1.1 as ADC inputs
  - Each pin reads input from a single photoresistor
- We used pin 2.2 as PWM signal



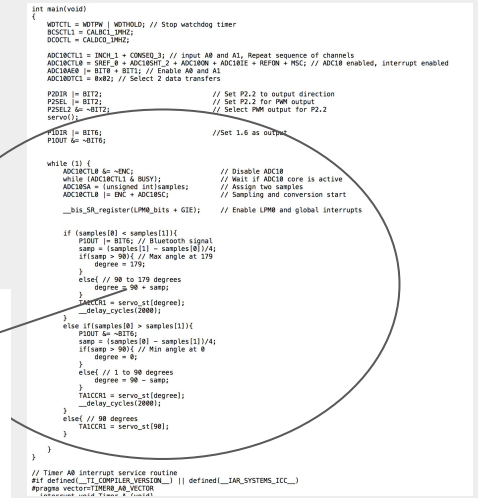
Code

- Our code integrates a variety of topics we learned this year
  - Low Power Mode
  - Analog to Digital Conversion
  - PWMing
  - Interrupt Service Routines
- We broke up our code into two separate parts
  - Reading ADC values and comparing them
  - Controlling the servo based on ADC analysis

Code - Reading and Analysing ADC

- Our main code constantly loops to check the incoming ADC values
- These ADC values are read through an interrupt
  - Two values are read at once
  - One for each photoresistor
- We compare values to see which photoresistor has more light
- Once compared we call function for motor control
- Debugging:
  - To make sure readings are done smoothly
  - We disable ADC10
  - Readings start once flags tell us to start
  - We are in LPM0 until readings come in

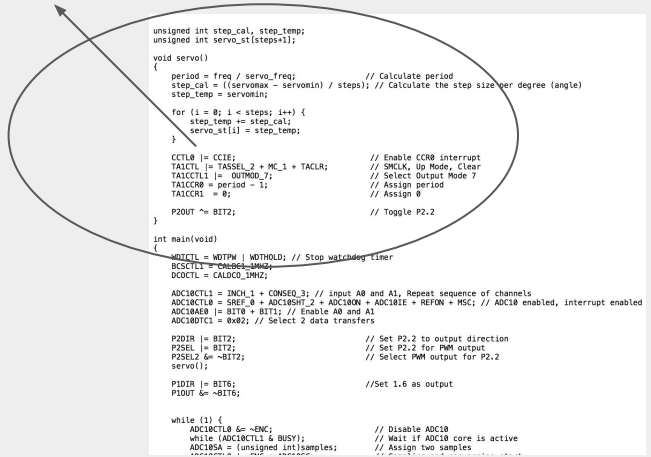
```
while (1) {
    ADC10CTL0 &= ~ENC;                      // Disable ADC10
    while (ADC10CTL1 & BUSY);               // Wait if ADC10 core is active
    ADC10SA = (unsigned int)samples;        // Assign two samples
    ADC10CTL0 |= ENC + ADC10SC;             // Sampling and conversion start

    __bis_SR_register(LPM0_bits + GIE);     // Enable LPM0 and global interrupts


    if (samples[0] < samples[1]){
        P1OUT |= BIT6; // Bluetooth signal
        samp = (samples[1] - samples[0])/4;
        if(samp > 90){ // Max angle at 179
            degree = 179;
        }
        else{ // 90 to 179 degrees
            degree = 90 + samp;
        }
        TA1CCR1 = servo_st[degree];
        __delay_cycles(2000);
    }
    else if(samples[0] > samples[1]){
        P1OUT &= ~BIT6;
        samp = (samples[0] - samples[1])/4;
        if(samp > 90){ // Min angle at 0
            degree = 0;
        }
        else{ // 1 to 90 degrees
            degree = 90 - samp;
        }
        TA1CCR1 = servo_st[degree];
        __delay_cycles(2000);
    }
    else{ // 90 degrees
        TA1CCR1 = servo_st[90];
    }

}
```



Code - Controlling Servo

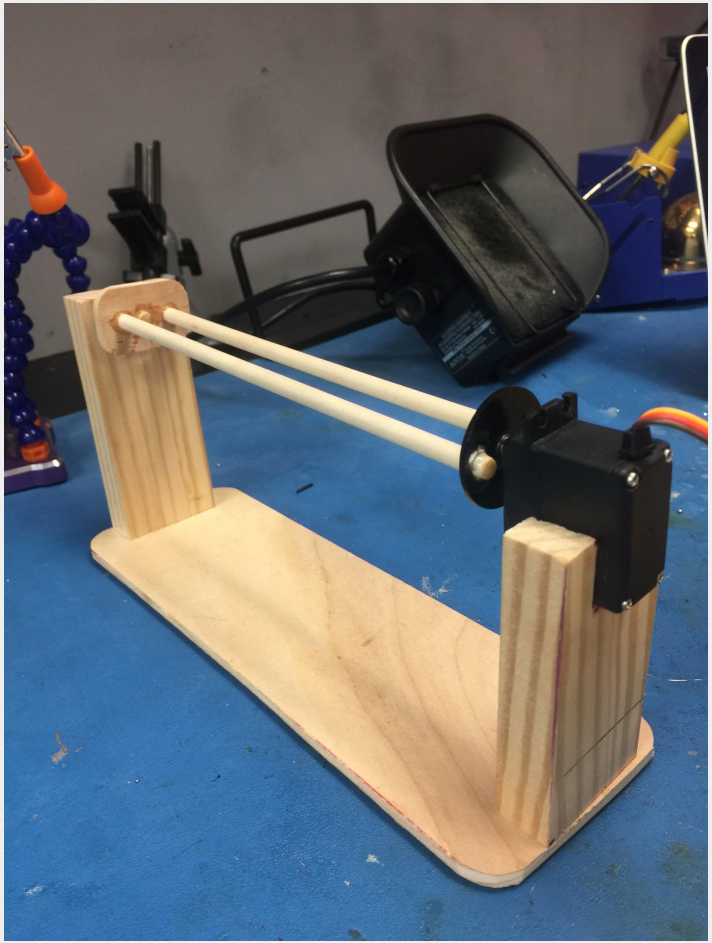
- We have a function to control servo
- Controlling servo is done through PWMing
- Shotaro i need help here

```
void servo()
{
    period = freq / servo_freq;               // Calculate period
    step_cal = ((servomax - servomin) / steps); // Calculate the step size per degree (angle)
    step_temp = servomin;

    for (i = 0; i < steps; i++) {
        step_temp += step_cal;
        servo_st[i] = step_temp;
    }

    CCTL0 |= CCIE;                            // Enable CCR0 interrupt
    TA1CTL |= TASSEL_2 + MC_1 + TACLR;        // SMCLK, Up Mode, Clear
    TA1CCTL1 |=  OUTMOD_7;                    // Select Output Mode 7
    TA1CCR0 = period - 1;                     // Assign period
    TA1CCR1  = 0;                             // Assign 0

    P2OUT ^= BIT2;                            // Toggle P2.2
}
```

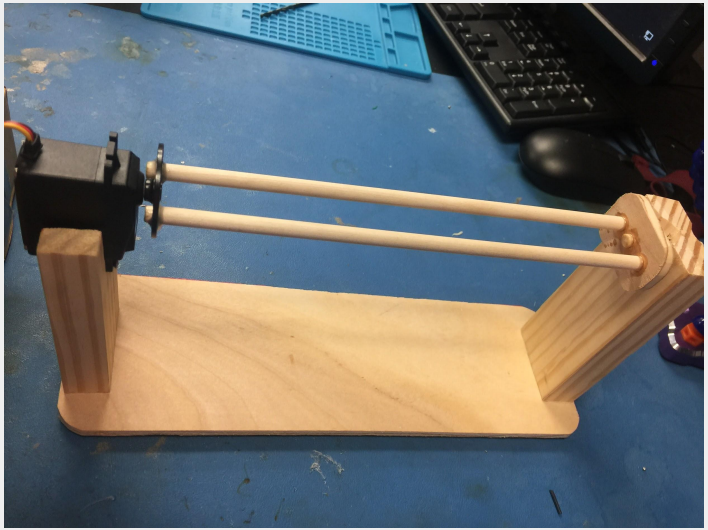


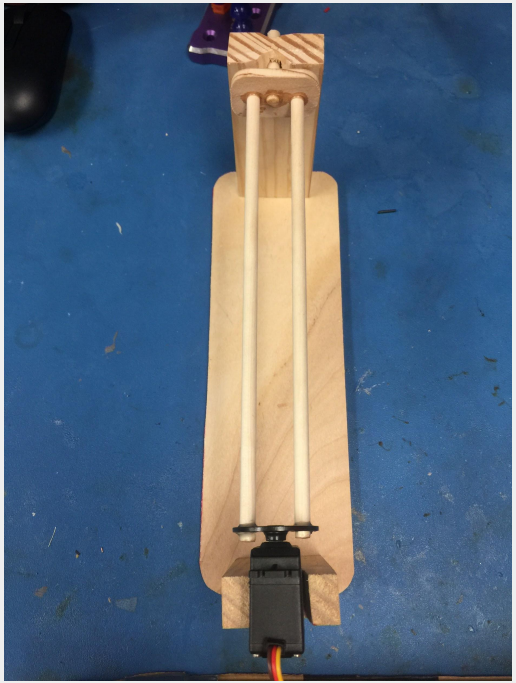
Solar Panel Base

- Built wooden base to hold solar panel
- Maximizes movement and angle capabilities
- Full 180 degree rotation allowed
- Frame for solar panel to lay on
- Sensors sit next to base but not on it
- Single servo motor attached on base



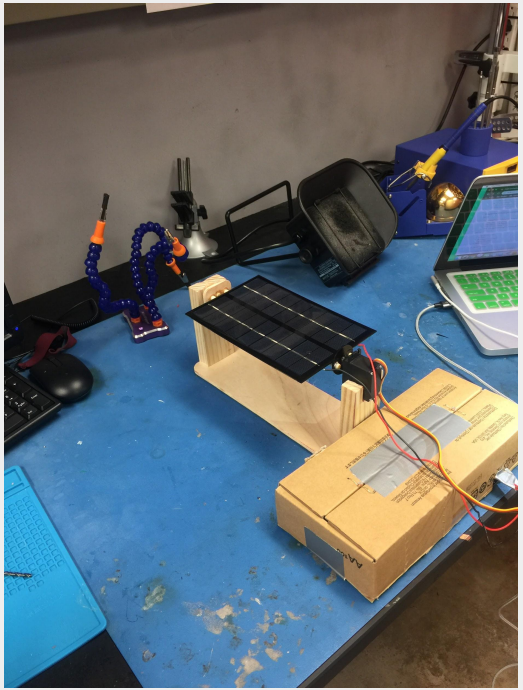
Solar Panel Base - No Solar Panel



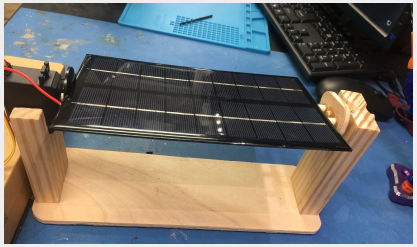


Solar Panel Base - With Solar Panel

Full Base

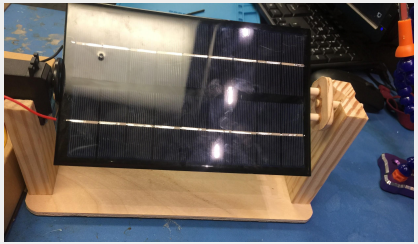


Flat :



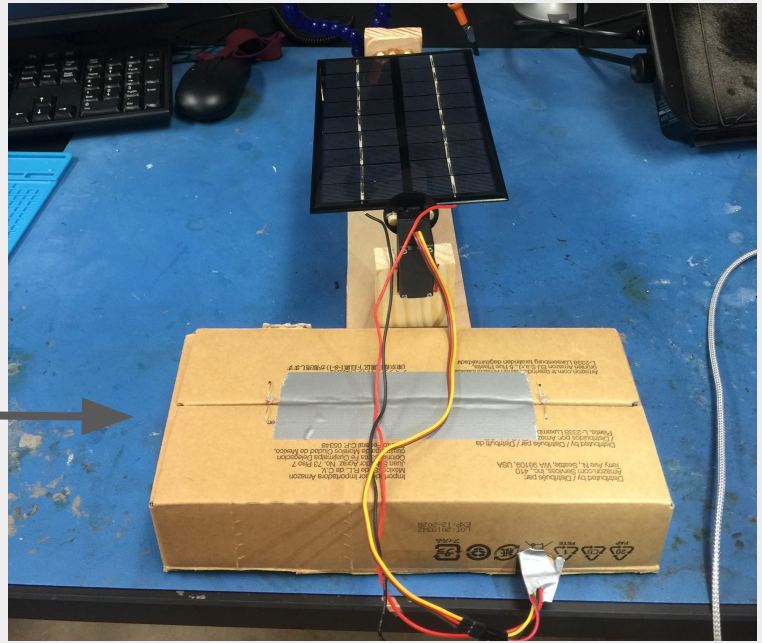
Leaning Left:

Leaning Right:



Free Wiring

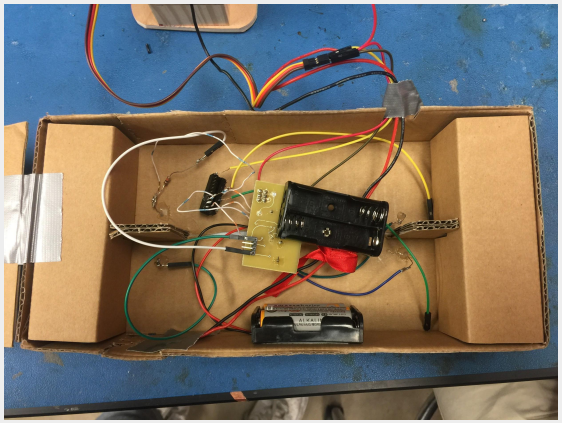
- Due to last minute changes in our design, we our changed PCB schematic
- Couldn't order PCB in time
- Decided to free wire to meet our vision
- Wires and parts sit inside box

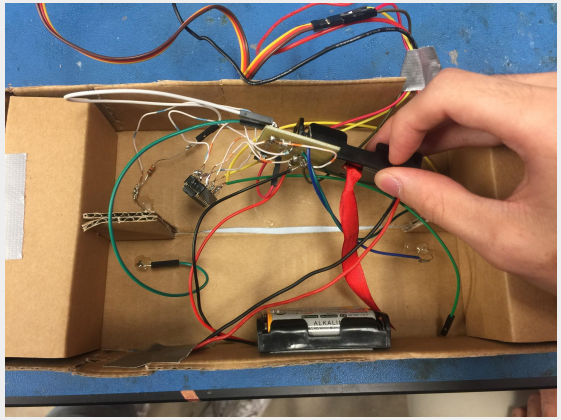


Free Wiring

Photoresistor are on top

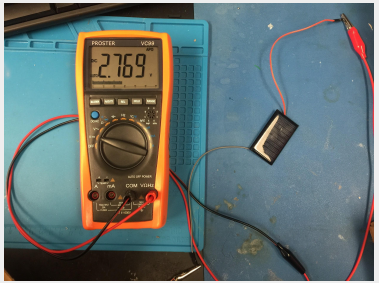
Inside Box:



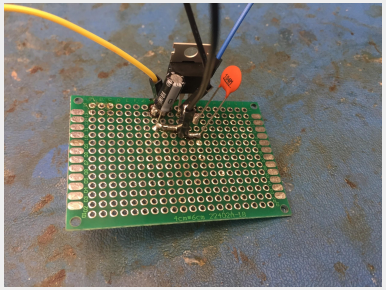


Areas For Improvement

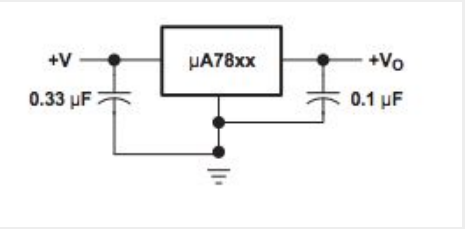
- Have Solar Panel Power Circuit
  - Small Solar Panel generates power
  - Large solar panel doesn't work
  - Order another large Solar Panel
  -
- Create Again With New PCB

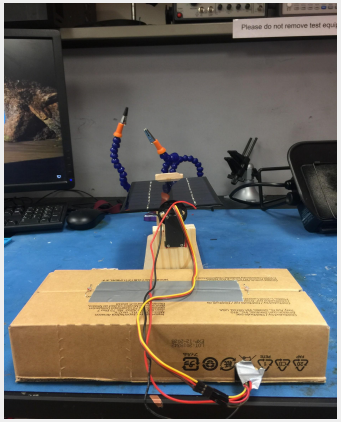


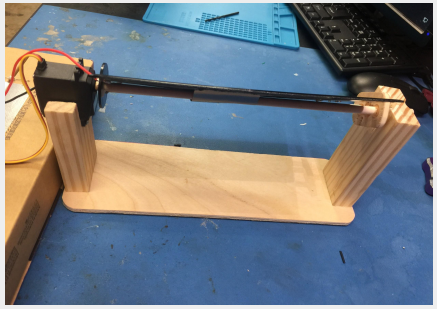
Small solar panel voltage

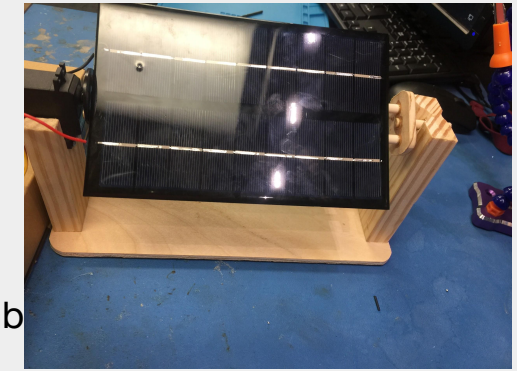


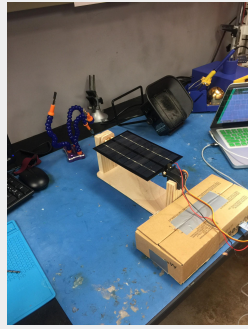
9V → 5V converter for large solar panel











Picture of final product or emb