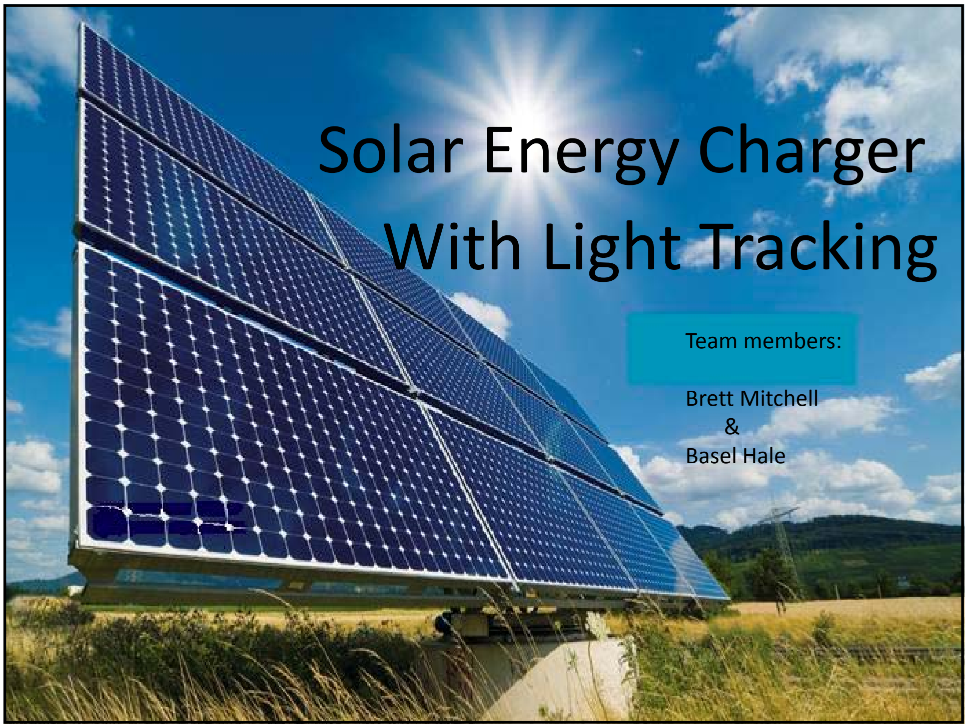


Project Name:

Solar Energy Charger with Light Tracking

Team members Names:

Brett Mitchell & Basel Hale

Contents:

- Motivation
- Risks & Obstacles
- Cost
- Diagrams
- Hardware
- Software
- Testing
- Pictures with Illustrations
- Lessons Learned
- Conclusion
- Questions

Motivation

- The motivation for this project was due to my cell phone's battery going dead by mid day. We thought this project would be ideal for outdoor events, like camping, hunting, hiking, etc. And adding security for a person who would get hurt and cannot call for help because he/she's phone is dead, or a hiker's GPS is dead and he/she is lost.

- This has become a problem that the Solar Tracker will fix.

Risks & Obstacles

- A risk that we ran into was getting the Tracking Circuit smaller to fit into a more portable housing.
- Fix – We fixed this problem by using the Arduino Micro, instead of the Arduino Uno(which is bigger).
- The Micro was having some programming issues.
- Fix – We solved this by downloading a new **FTDI** driver (Future Technology Devices International) for the USB port and a new cable.

Risks & Obstacles

- Another risk we had run into was the servo motors were draining the battery faster than it could charge(When testing).

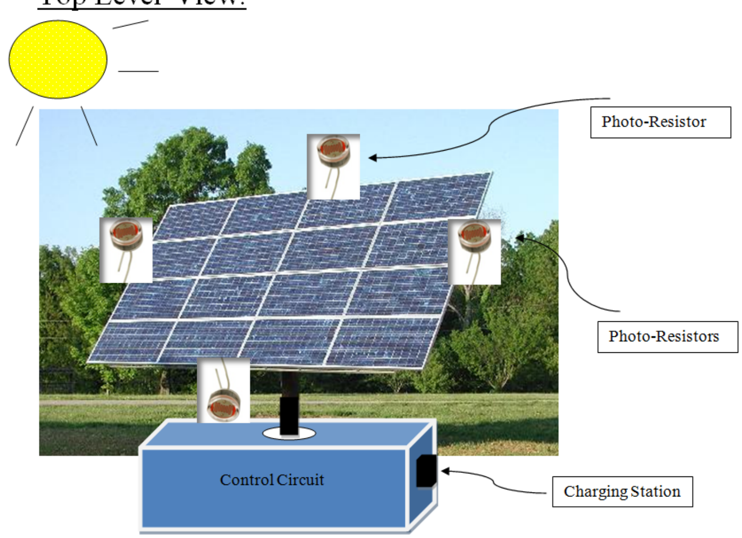
- Fix – We fixed this by purchasing the 9V Solar panels(instead of using our original 6V Panels) and purchasing a better battery pack that would conduct 1600mAh(instead of using our original 1200mAh).

Cost:

Items:	Cost (Dollars, $)
Solar Panel(x3)	22.99
Mounting Bracket	0
40-Pin Socket(Ardiuno Holder)	0.99
Ardiuno Micro Microprocessor	29.99
Photo Resistors	3.99
10k Resistors (x3)	0.97
2n2222 Transistors	.47
1N 4001 Diodes	4.47
Traxxas Servo Motors(x2)	4.99
Circuit Prototype Board	4.49
Charging Circuit Housing	5.49
USB Port	4.29
ON-OFF Rocker Switch	11.49
Battery NiMH (6Volt)	23.99
Misc.	15.48
Total	$134.09

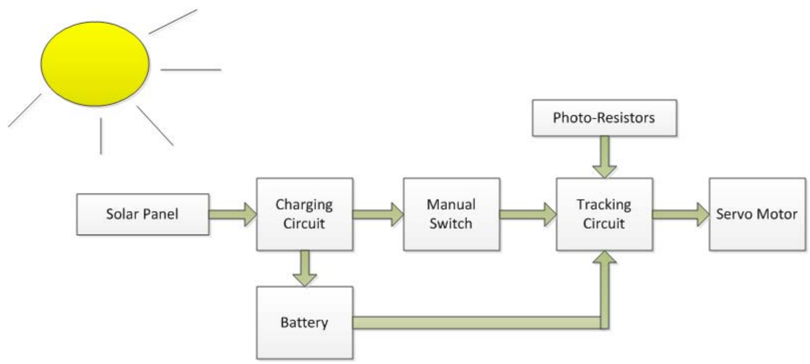
Architecture Designs

Top Level View:



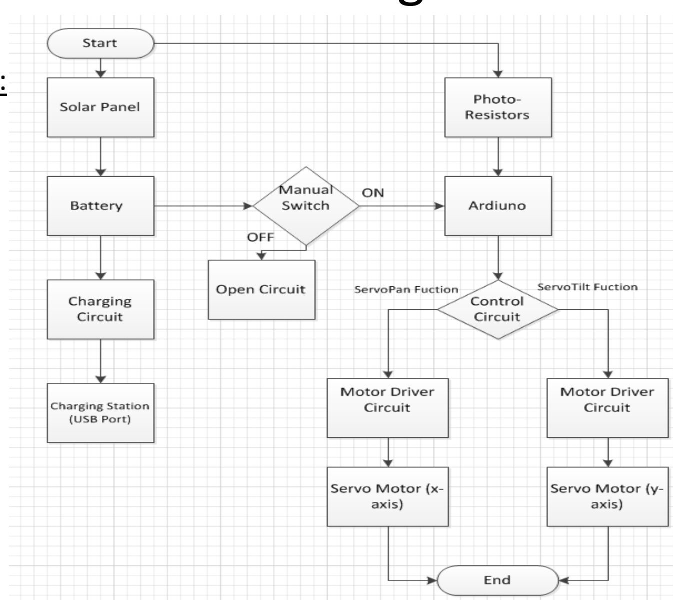
Architecture Designs

Block Diagram View:



Architecture Designs

Control Flow View:



9 Volt Solar Panel 1.5 Watts

RadioShack 1.5W Solar Panel 9V #277- 0053

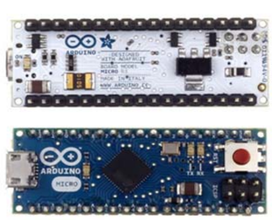
Use this solar panel to power models and science projects

- 1.5W solar panel 9v
- Peak power output (W): 1.5W
- Output voltage: 9V
- Panel dimensions: 5.11x3.34x0.125”
- Output cable length: 7.87”

Arduino Micro

Summary

Microcontroller	ATmega32u4
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	20
PWM Channels	7
Analog Input Channels	12
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega32u4) of which 4 KB used by bootloader
SRAM	2.5 KB (ATmega32u4)
EEPROM	1 KB (ATmega32u4)
Clock Speed	16 MHz



Hi-Energy NiMH Battery

Hi-Energy Receiver Battery 6.0v 1600mAh NiMH Flat JR/Z Conn

Traxxas Servo (built in controller)

Specifications:

Length: 1.59"
Width: 0.77"
Height: 1.58"
Torque: 42.0 oz-in
Transit Time: 0.22 seconds/60 degrees
Weight: 1.5 oz

Servo has built in motor controller.

Arduino Code (library)


```
//By Basel Hale
//Senior Design project
//ECET 491

#include <Servo.h>

Servo ServoTilt;  // create
Servo ServoPan;
```

Arduino Code (Analog Read)

```
void setup()
{
  Serial.begin(9600);   //Begin serial connection between Arduino and PC
  ServoPan.attach(10);  // attaches the servo on pin 10 to the servo object
  ServoTilt.attach(9);  // attaches the servo on pin 9 to the servo object
}


void loop()
{
  cell  = analogRead(0);  //placeholder for the digital value of the ADC conversion of analog photoresistor 1 voltage
  cell2 = analogRead(1);  //placeholder for the digital value of the ADC conversion of analog photoresistor 2 voltage
  cell3 = analogRead(2);  //placeholder for the digital value of the ADC conversion of analog photoresistor 3 voltage
  cell4 = analogRead(3);  //placeholder for the digital value of the ADC conversion of analog photoresistor 4 voltage
```

Arduino Code (Loop)

```
   if(cell > cell2)    //TILT LOOP
 {
posX = ((analogRead(cell) + analogRead(cell))/2);
delay(2200);
pos = constrain(posX, 220, 740);
delay(7500);
int servoPos = map(pos, 220, 740, 255, 0);
int servoDegree = map(servoPos, 255, 0, 179, 0);
ServoTilt.write(servoDegree);

 /*Serial.print("Servo Degree = "); //testing the var
 Serial.println(servoDegree);
Serial.print("Cell = ");
 Serial.println(cell);
Serial.print("Cell2 = ");
 Serial.println(cell2);
 Serial.print("POS = " );
 Serial.println(pos);
  Serial.print("POSX = " );
 Serial.println(posX);*/
 }

if (cell < cell2)
{
pos2X = ((analogRead(cell2)+ analogRead(cell2))/2);
delay(2200);
pos2 = constrain(pos2X, 220, 740);
delay(7500);
int servoPos2 = map(pos2, 220, 740, 255, 0);
int servoDegree2 = map(servoPos2, 255, 0, 0, 179);
ServoTilt.write(servoDegree2);
```

Testing and Analyzing:

Table 4: Traxxas Servo

Traxxas Servo	Operation Volts	Operation Amps
While Turning	5 V	330mA

Testing

Table 1: Solar Panel Output in Bright Sun Light

Panel	1	2	3	1,2,3
Output Voltage	10.45 V	10.47 V	10.5 V	10.45 V
Output Current	158.2mA	162.1mA	167.0mA	307.2mA

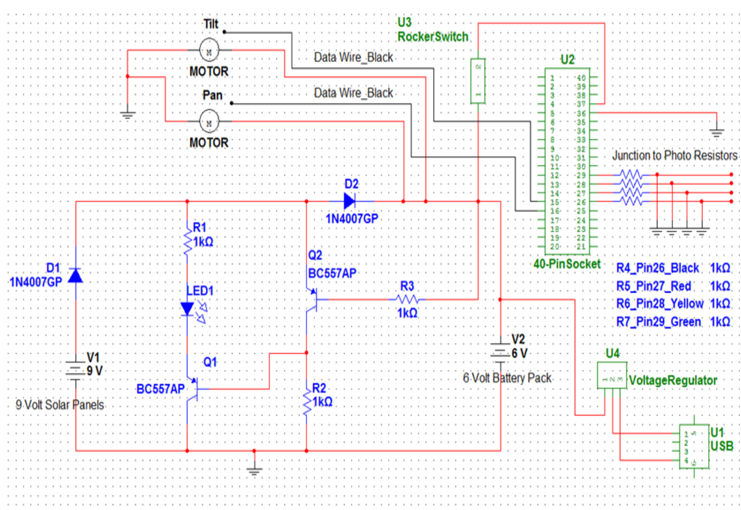
Table 2: Solar Panel Output in Cloudy Light

Panel	1	2	3	1,2,3
Output Voltage	8.28 V	8.30 V	8.21 V	8.21 V
Output Current	78.5mA	80.3mA	75.6mA	198.5mA

Table 3: Solar Panel Ratings

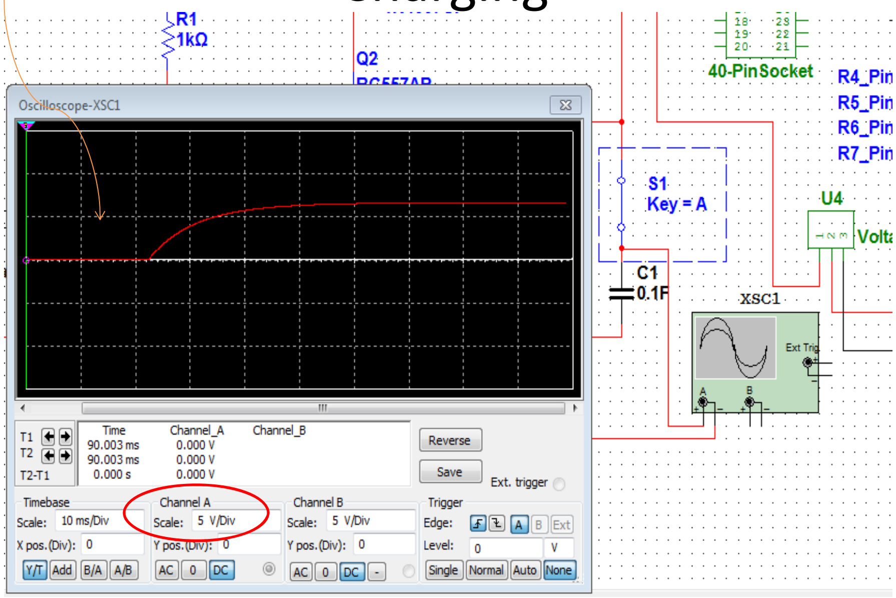
Solar Panel Specifications	Power	Voltage	Current
Solar Panel Output Ratings	1.5 W	9 V	1.5W/9V =0.167A

Multisim Circuit

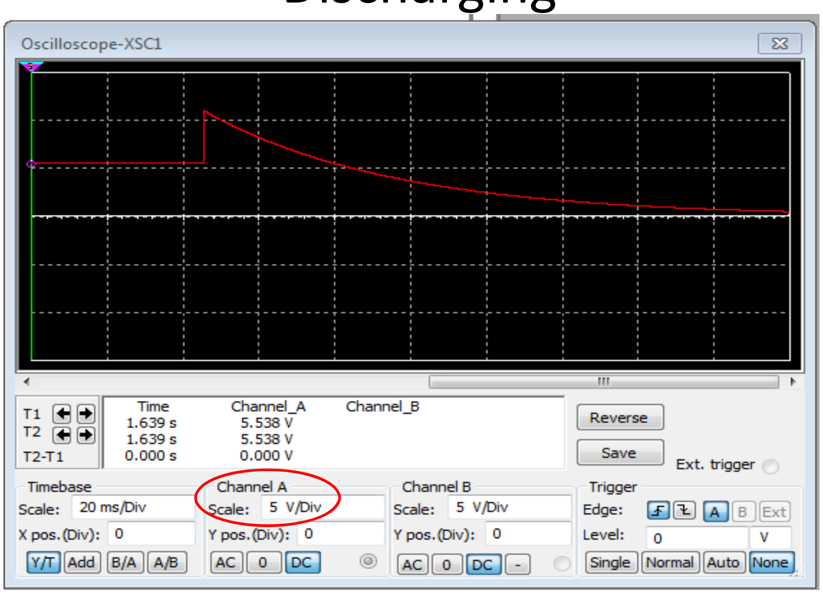


This is a 5 volt scale

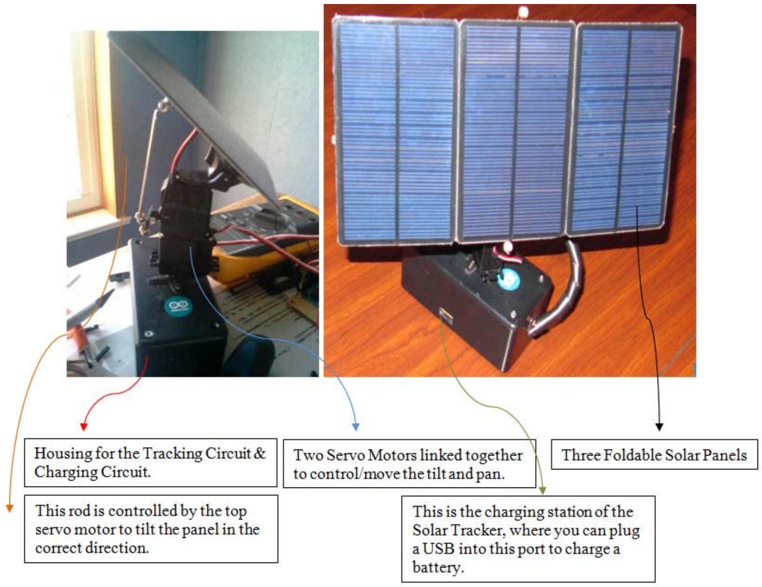
Charging



Discharging



Illustrations



Lessons Learned

- The first lesson learned from this project was to get the tracking portion of the project working properly we had to use programming, instead of IC chips because of the sinking of current.
- The next lesson learned was that the servo motors were using a lot of current to run them and this was cutting down our charging current, we fixed this obstacle by using the 9 volt solar panels instead of the 6 volt panels that we started with. The size of the 6 volt panels and the 9 volt panels are the same, so there were no size or portability issues here.

Conclusion

We consider this project has been a success. We have been able to accomplish our goals. Even though along the way we ran into many problems, our project was flexible enough to adapt to the problems we encountered. We were able to build a successful tracking unit and implement the software to track it. In the process we learned a lot about hardware and software that will enable us to be better electrical engineers.

Questions