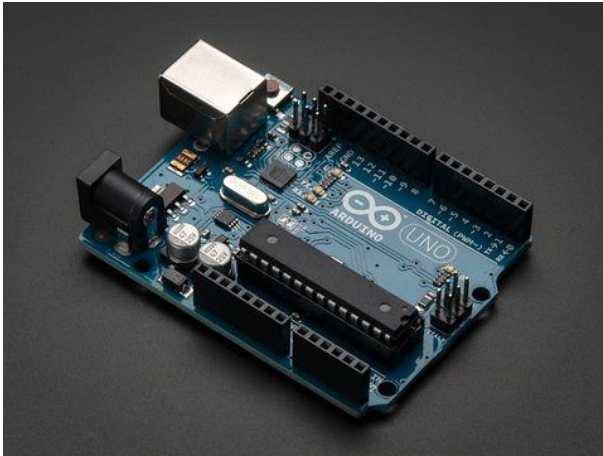


Objective:

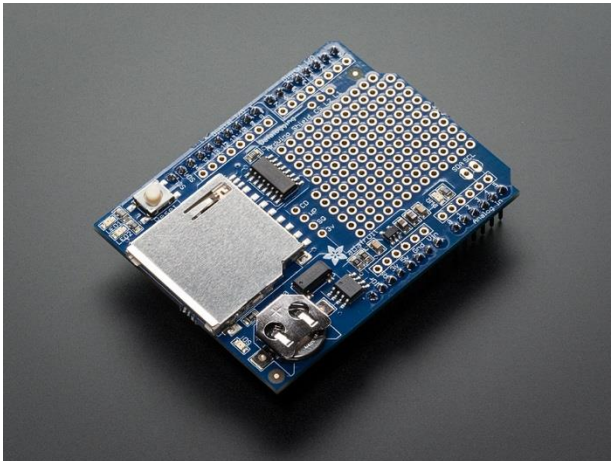
Assemble solar powered system to collect and store temperature and humidity values on SD card.

Contents:

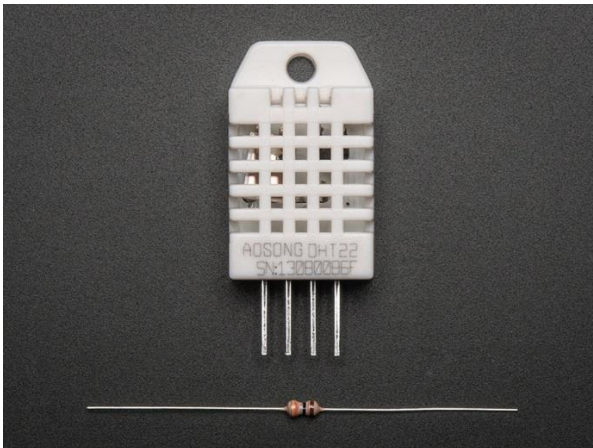
- Arduino Uno - \$25 (<https://www.adafruit.com/products/50>)



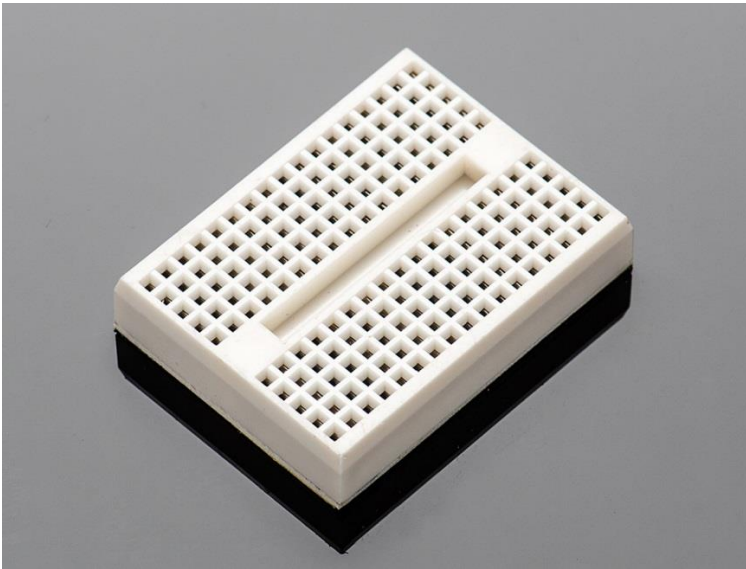
- Data Logging Shield - \$20 (<https://www.adafruit.com/product/1141>)



- DHT22 Temperature and Humidity Sensor - \$10 (<https://www.adafruit.com/product/385>)



- Mini solderless breadboard - \$4

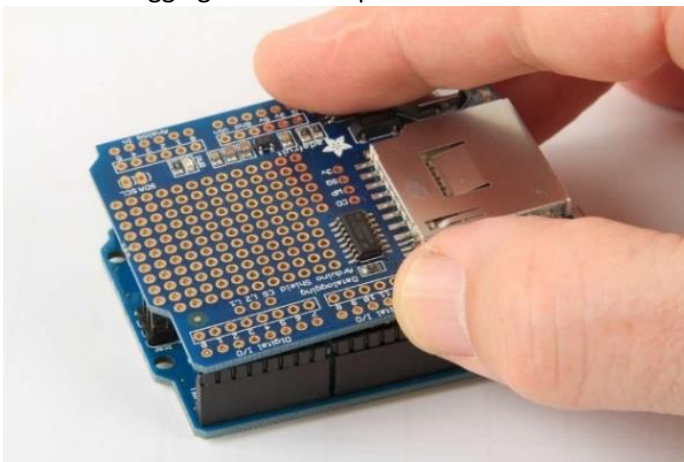


- SD Card - \$10 (<https://www.adafruit.com/products/1562>)
- USB Cord - \$3 (<https://www.adafruit.com/products/900>)
- 6 W Solar Panel - \$50 (<http://www.voltaicsystems.com/6wattkit.php>)
- Voltaic 4,000 mAh battery - \$35 (<http://www.voltaicsystems.com/6wattkit.php>)
- Weatherproof box - \$15 (http://www.bhphotovideo.com/c/product/257883-REG/pelican_1050_025_100_1050_clear_micro_case.html)
- 3 jumper cables
- Connector with wires
- **APX TOTAL COST = \$172 + shipping**

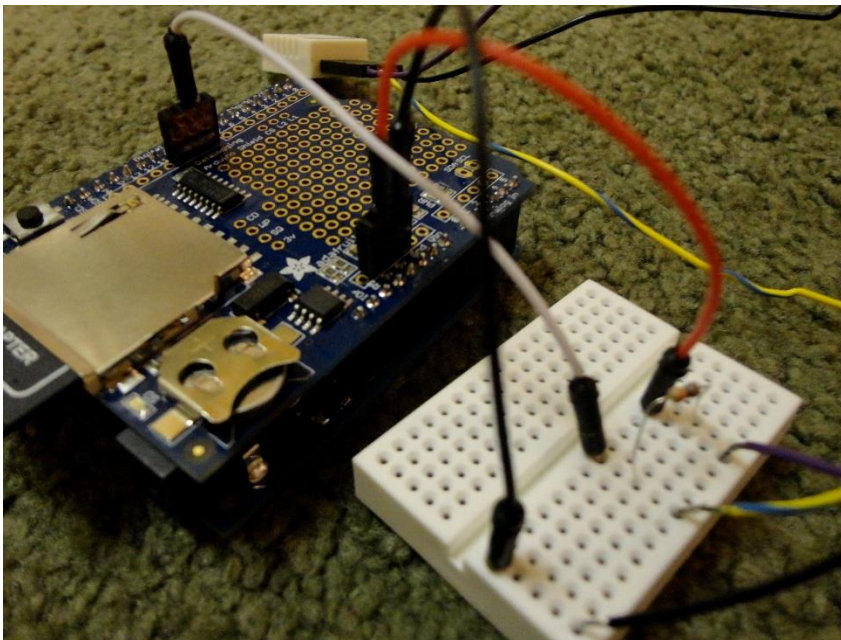
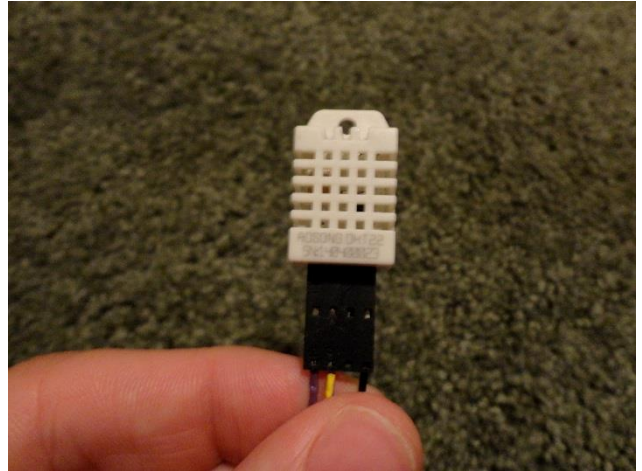
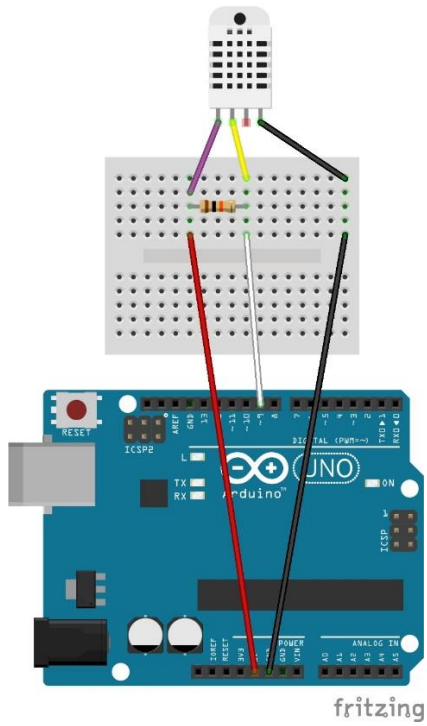
Instructions:

Assemble/Wire Hardware

1. Insert data logging shield on top of Arduino board



2. Wire DHT22 Sensor to data logger shield



Get Software Ready

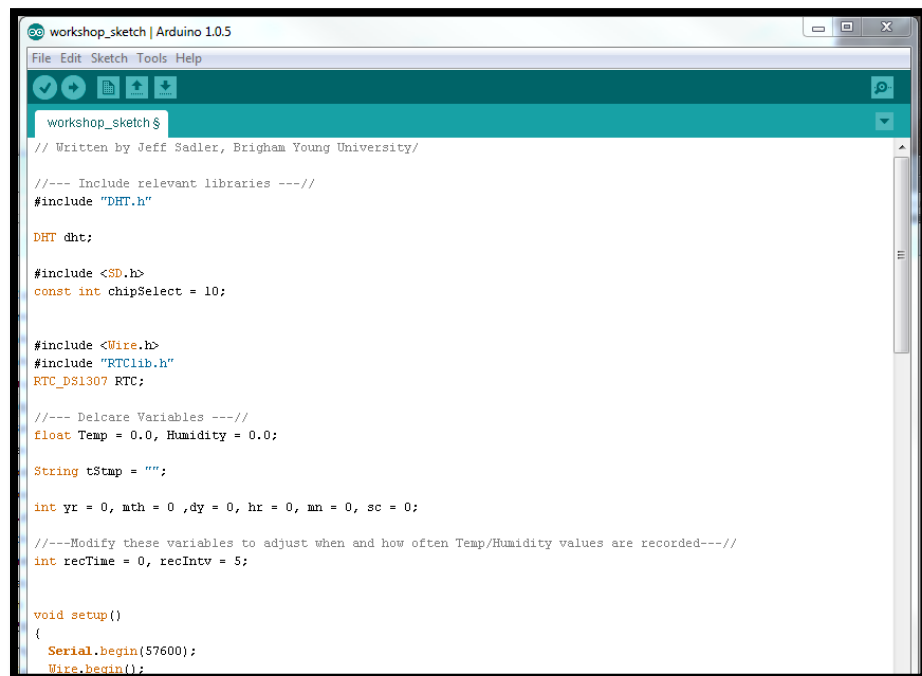
1. Libraries expand the functionality of Arduino. For our purposes we need a library to interact with the RTC and the DHT22.

- RTC Library – <https://github.com/adafruit/RTClib/archive/master.zip>
- DHT22 Library – <https://github.com/markruys/arduino-DHT/archive/master.zip>

The Arduino IDE needs to know where to look for these libraries. Unzip the two folders and put the contents in the “Libraries” folder of the “Arduino” folder.

Note: For library titles to be compatible with Arduino, they cannot contain certain characters and “-” is one of them. To make the above libraries work, you’ll need to either delete the “-”s or change them to “_”s.

2. Arduinos are very customizable and learning how to take advantage of that isn’t too hard. However, for the workshop, I’ve written a program to do the following,
 - i. Take timestamp from real time clock (RTC)
 - ii. Read temperature and humidity values every 15 seconds
 - iii. Record timestamp and readings on SD card
- Download the main file at this link:
 - i. Primary Sketch “workshop_sketch” – see github “workshop sketch”
 - Open the sketch in the Arduino IDE



```
workshop_sketch | Arduino 1.0.5
File Edit Sketch Tools Help

workshop_sketch $
// Written by Jeff Sadler, Brigham Young University/

//--- Include relevant libraries ---//
#include "DHT.h"

DHT dht;

#include <SD.h>
const int chipSelect = 10;

#include <Wire.h>
#include "RTClib.h"
RTC_DS1307 RTC;

//--- Declare Variables ---//
float Temp = 0.0, Humidity = 0.0;

String tStamp = "";

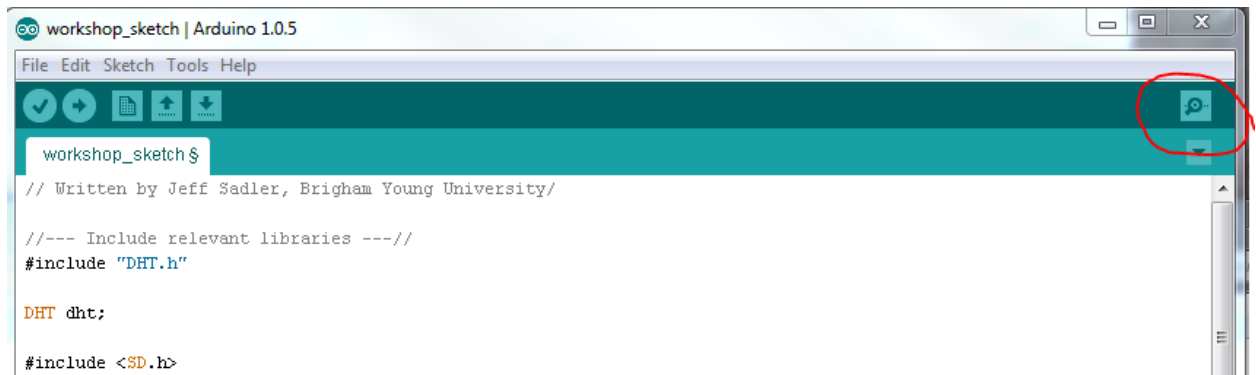
int yr = 0, mth = 0, dy = 0, hr = 0, mn = 0, sc = 0;

//---Modify these variables to adjust when and how often Temp/Humidity values are recorded---//
int recTime = 0, recIntv = 5;

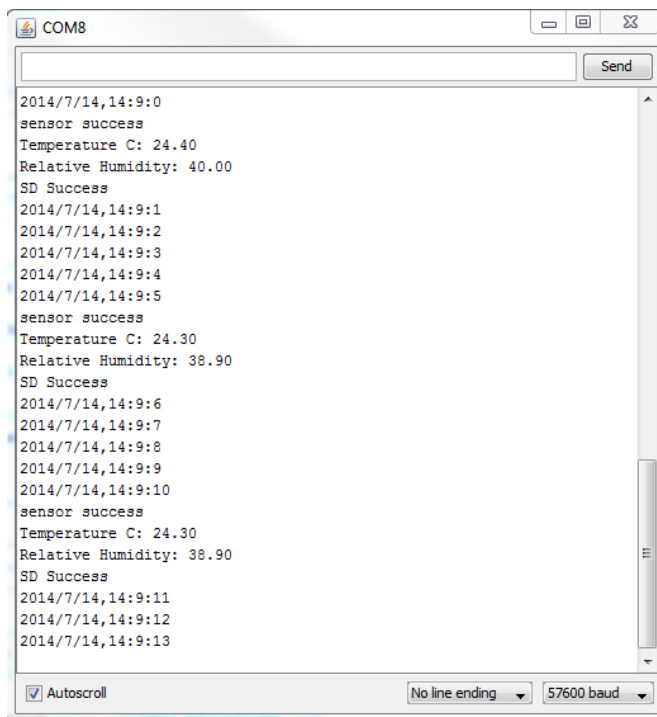
void setup()
{
  Serial.begin(57600);
  Wire.begin();
```

Note: If you want to change when and how frequently, adjust the ‘recTime’ and ‘recIntv’ variables. The values are currently set so that the first measurement is taken at ‘0’ seconds and then every 5 seconds.

3. Upload “workshop_sketch” to Arduino
4. Check the serial monitor by clicking the icon circled in red below.

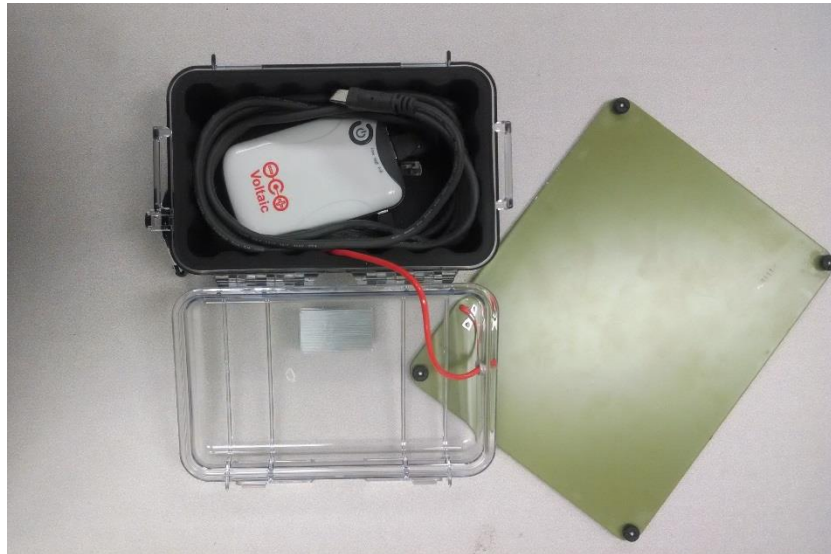


If everything is working right, you should get measurement readings starting when the second value is “0.”

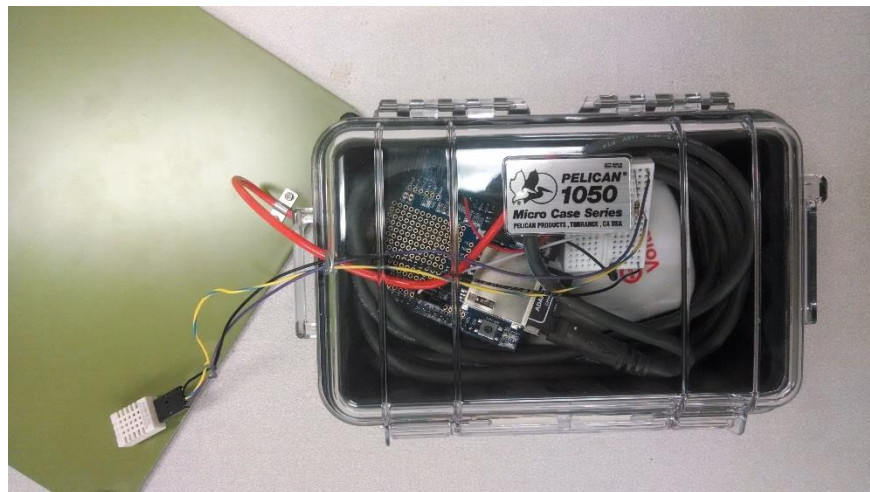


Attach to Solar Power System

1. Unplug from computer, attach Solar Panel to Voltaic battery through hole in case



2. Detach wires from mini breadboard, insert wires through hole and reattach. Then connect Arduino to the free end of the USB cable



3. Press and hold battery button for six seconds and the battery lights should blink three times. Now the battery is set to "Always On" mode. The power light on the data logger shield should be on and the red "SD" light should start flashing every "recIntv" (e.g. every five seconds) soon.
4. The data can be downloaded from the SD card. To view the data in the serial monitor download this sketch: (see "ReadSdCard" sketch on github) then upload it to the board and open the serial monitor as before.