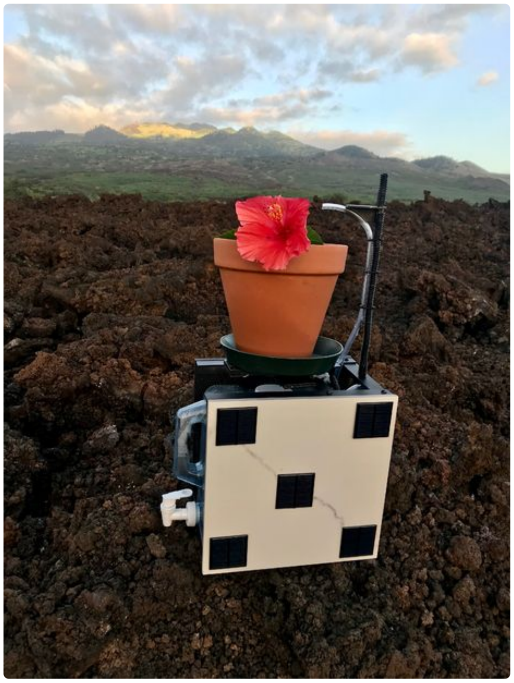
Solar Weight-Based Plant Management With ESP32

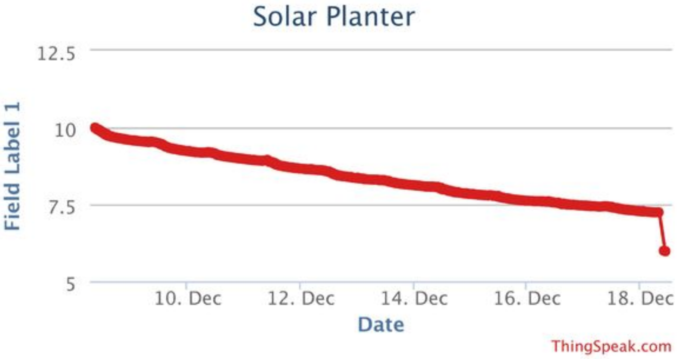
by rabbitcreek

Growing plants is fun and watering and caring for them is not really a hassle. Microcontroller applications to monitor their health are all over the internet and the inspiration for their design comes from plant's static nature and the ease of monitoring something that doesn't run around and sweat. I am relatively new to plant growth and guides on the internet seemed to be written by well meaning but not engineer types. A friend who I asked "how much do I water them..." replied the only way is to heft the plant and if it feels light you water it. He is very good at "growing". Sticking your finger into the soil does not really help much. Most of the Instructables utilize a cheap soil moisture probe which is prone to a variety of failures--the most blatant of which are inaccuracy and corrosion.

Reviewing the literature reveals that dirt can be up to 40% water and measuring this requires fairly expensive instruments. The cheaper probes rely on water conductance which will vary with dissolved salts and other factors. Above is a graph I did of a container of dirt weighed over 2 weeks followed by oven heating to 300 to remove all non attached water. Forty percent of the total soil is water and over ten hot days of direct sun it lost 75% of this water at a relatively linear rate. So what is the correct level of moisture? Depends on a variety of factors but when building this machine a good clue is to carefully water your plant to the level that you think is right and set it on the machine which carefully measures its weight and then within a set limit adds water when needed. The design can be modified for hanging plant baskets and pressurized water systems.

The machine had to run on solar energy, be autonomous with its own water supply, monitor its water supply by notifications to the web, sleep when not in use to minimize power and remember the base weight and how many waterings and other data in between sleep cycles. The new ESP32 seemed a good candidate for the brain.




Solar Planter
12.5
10
7.5
5
Field Label 1
10. Dec
12. Dec
14. Dec
16. Dec
18. Dec
Date
ThingSpeak.com

Step 1: Gather Your Supplies

The machine is made out of two BigBox store 12 inch ceramic tiles in a aluminum channel frame sandwiching a water tank. The electronics are secured in a plastic electrical box on the back. The water tank has an exit hose from a enclosed pump and sensor unit glued to the bottom of the tank that feeds the plant. The load cell cantilevers from a cross beam at the top of the unit.

1. Arrow Home Products 00743 2 Gallon Slimline Beverage Container in Clear

2. uxcell 5Pcs 5.5V 60mA Poly Mini Solar Cell Panel Module DIY

3. Gikfun Metal Ball Tilt Shaking Position Switches for Arduino

4. Uxcell a14071900ux0057 10Kg Aluminium Alloy Electronic Scale Load Cell

5. Adafruit HUZZAH32 – ESP32 Feather Board

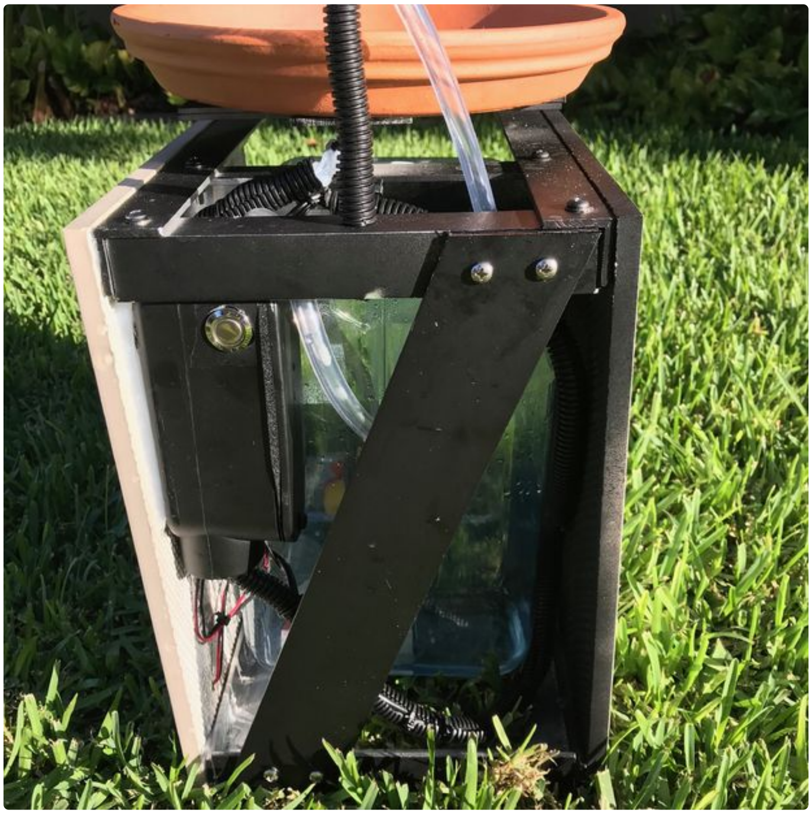
6. HX711 Weight Weighing Load Cell Conversion Module Sensors Ad Module for Arduino

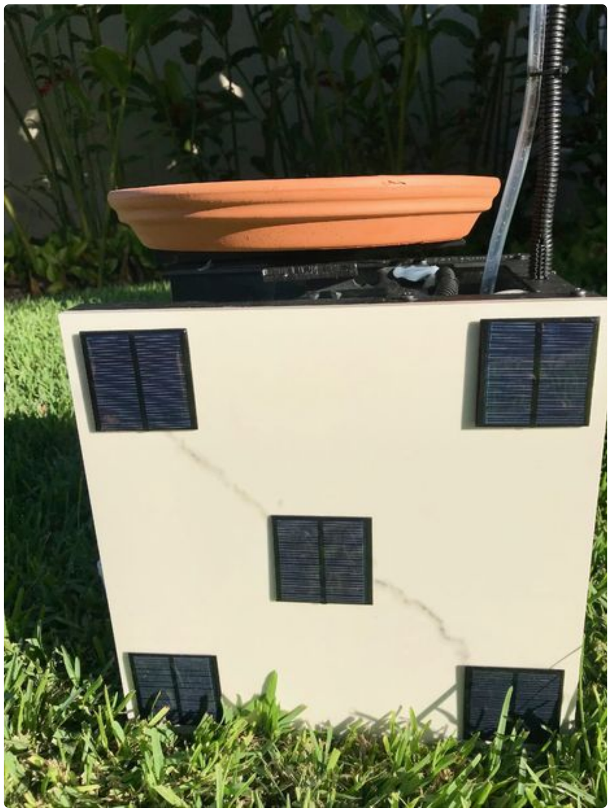
7. Adafruit Latching Mini Relay FeatherWing

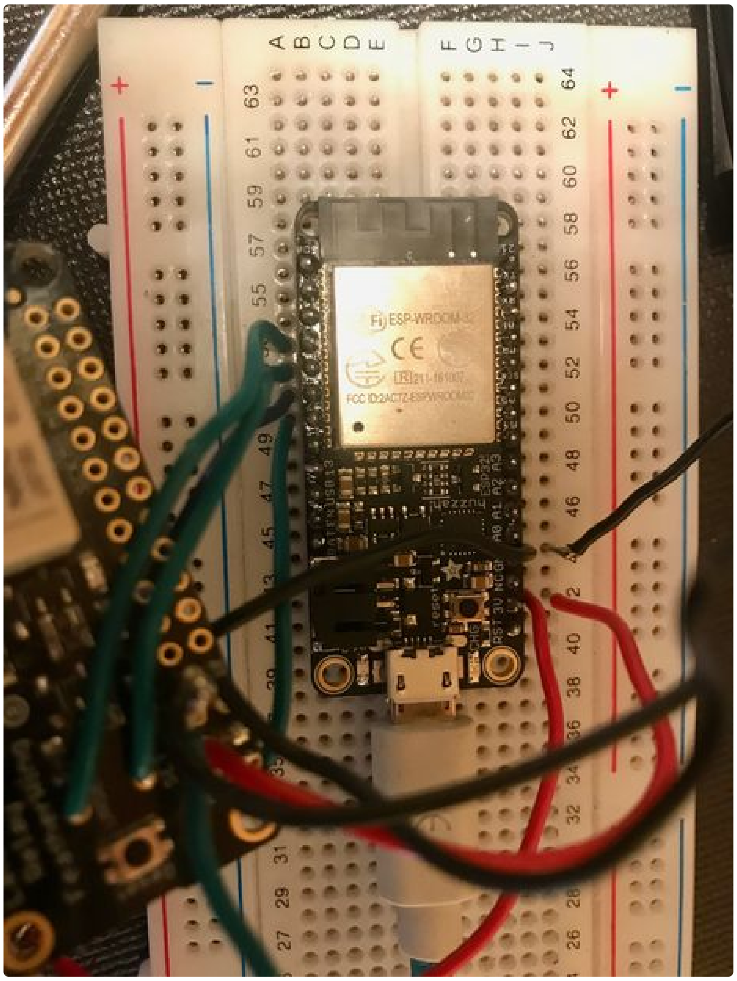
8. TP4056 Lithium Cell Charger Module with Battery Protection

9. ECEEN USB Pump Mini Submersible Water Pumping for Aquarium Hydroponic Powered Via USB DC 3.5-9V

10. 18650 Lipo battery with battery holder





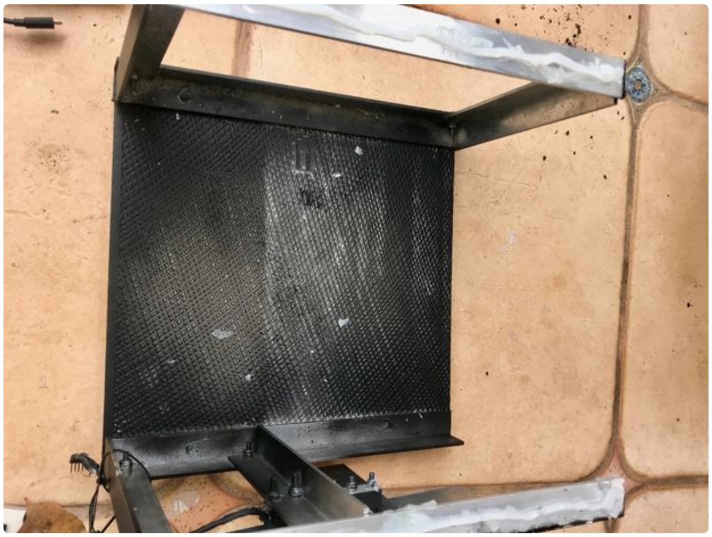

ESP-WROOM-32
CE

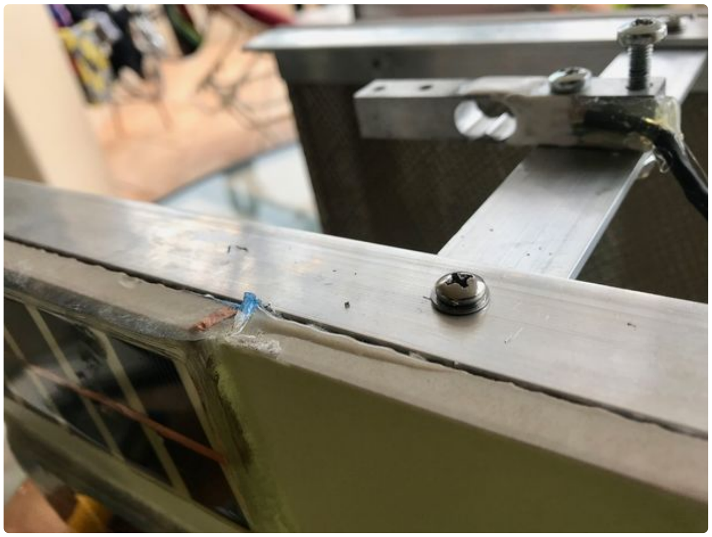
Step 2: Build the Box

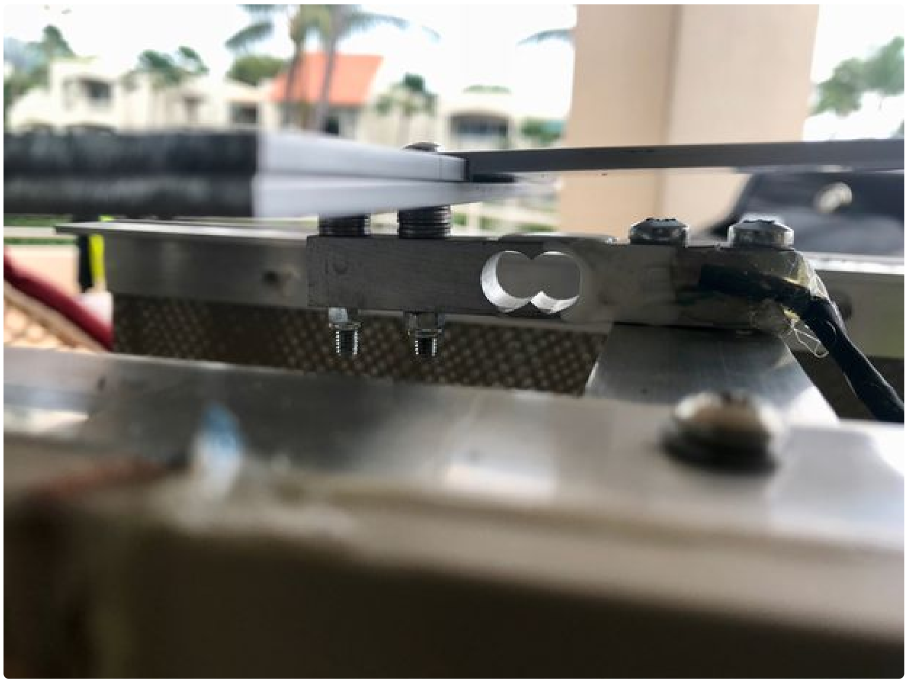
The box frame is made out of BigBox 1 inch aluminum angle. You get the general idea from the pictures and its not too difficult to assemble. The frames are based on the square foot tiles that form the front and back sides of the unit. The tiles are held to the faces of the aluminum frame with silicon glue. The dimension of the center section is dependent on the size of your water tank. The tank opening is designed so that you can easily pull it out of the unit and refill it from the top. Wires and tubes that attach the tank must be long enough and curl at the back.

The solar panel placement is design dependent. I was going to use multiple round panels to give it a 'dice' look but settled on the squares because they gave the best combination of voltage and current. I am not going to go into the details of hooking up multiple solar panels but you need at least 5.5v to make the charger circuit work. These panels were all hooked in parallel to boost the amperage. The holes in the ceramic tile are carefully drilled with a diamond bit--make sure you use water as a coolant to do this or you'll ruin the bit. These holes should only take a couple minutes each. Use liberal amounts of silicon glue to hold the panels and the wires on the inside of the tiles in place.

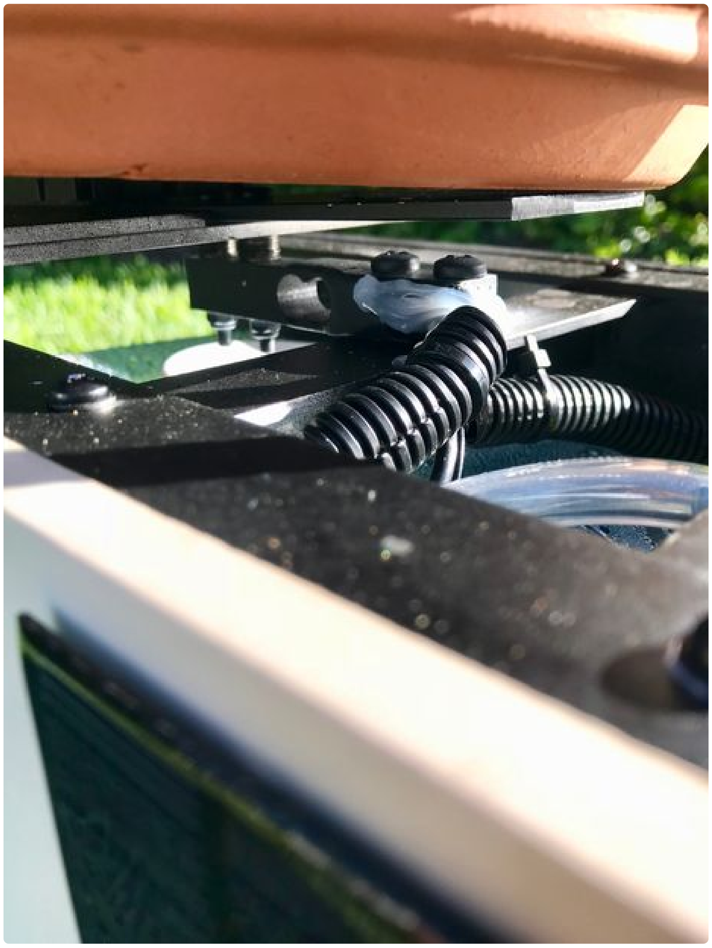
The load cell is very reasonable and comes rated in various weights. I used the 10 kg variety but if you are going heavy-planter plan accordingly. Like my other Instructables: https://www.instructables.com/id/Bike-Power-Pedal-IoT/ these load cells must be cantilevered out from their support side with their 4mm and 5mm taped screw holes. In this case a aluminum cross piece between the two ceramic tile supports holds one end of the load cell. The other supports a platform of flat aluminum bar silicon glued to the plant drainage cup. Be very careful with the wires from these guys--they are very fragile and nearly impossible to repair if broken off near their origin. Goop with lots of hot glue or silicon to maintain their integrity.

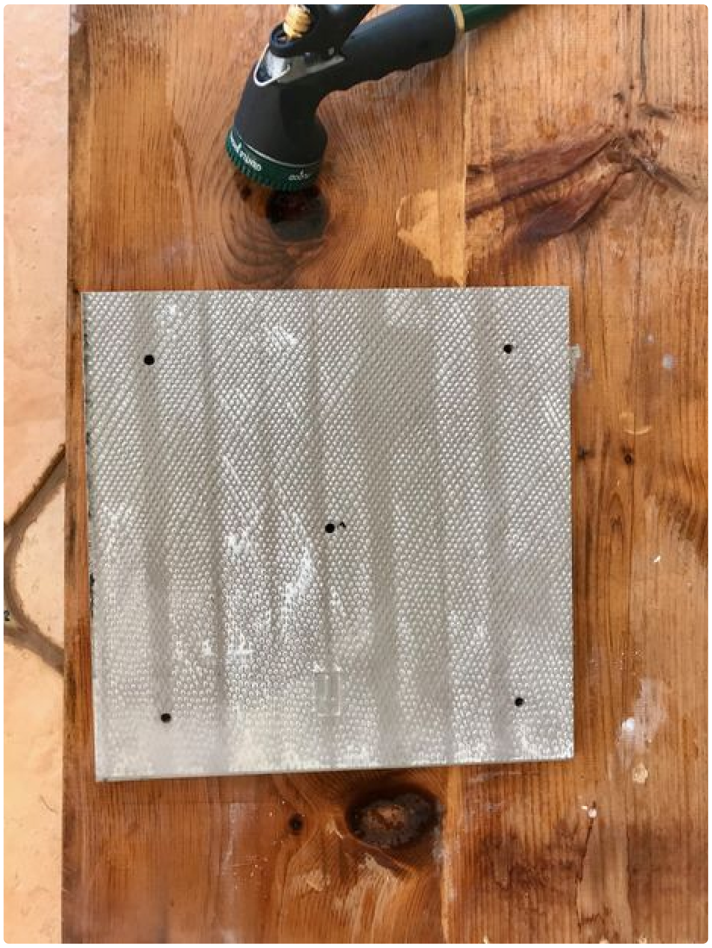






11 • 13 • 15 • 17 • 19

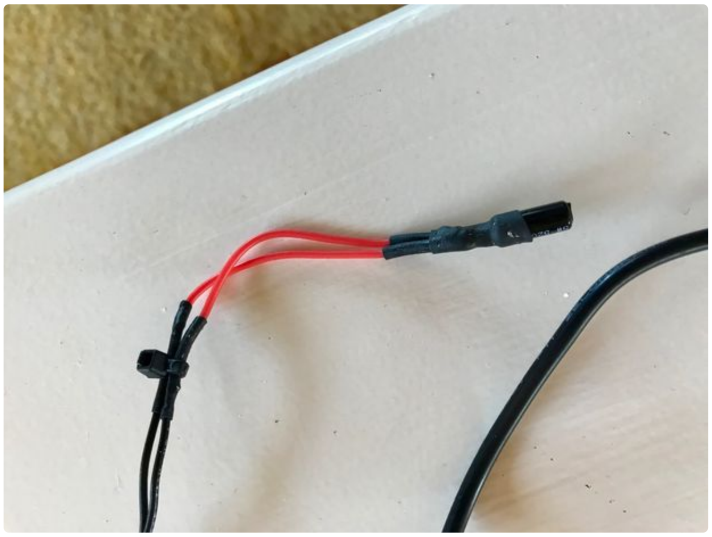


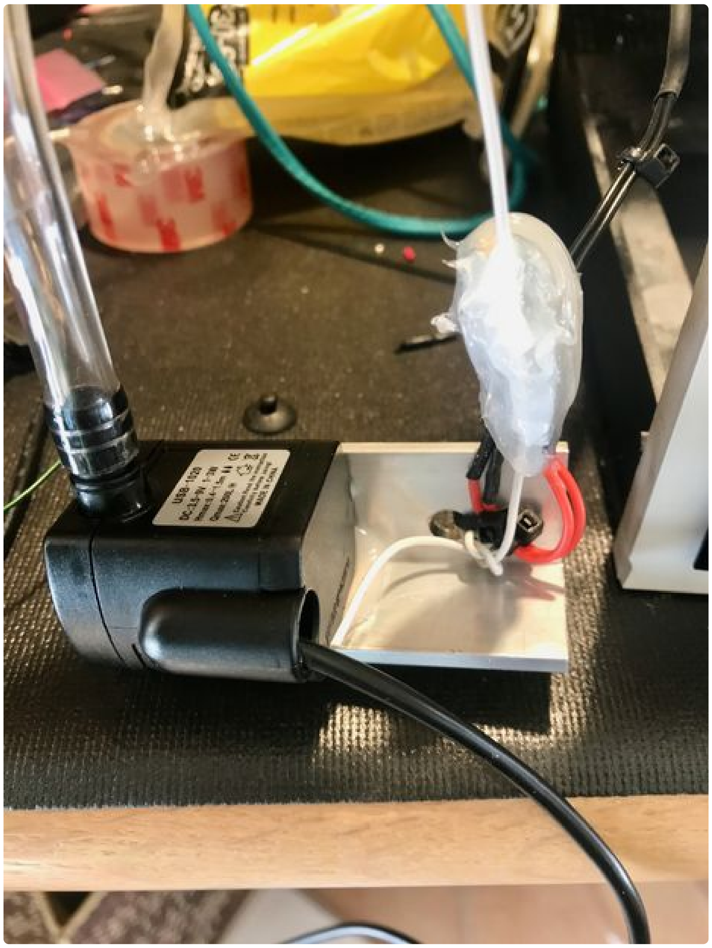


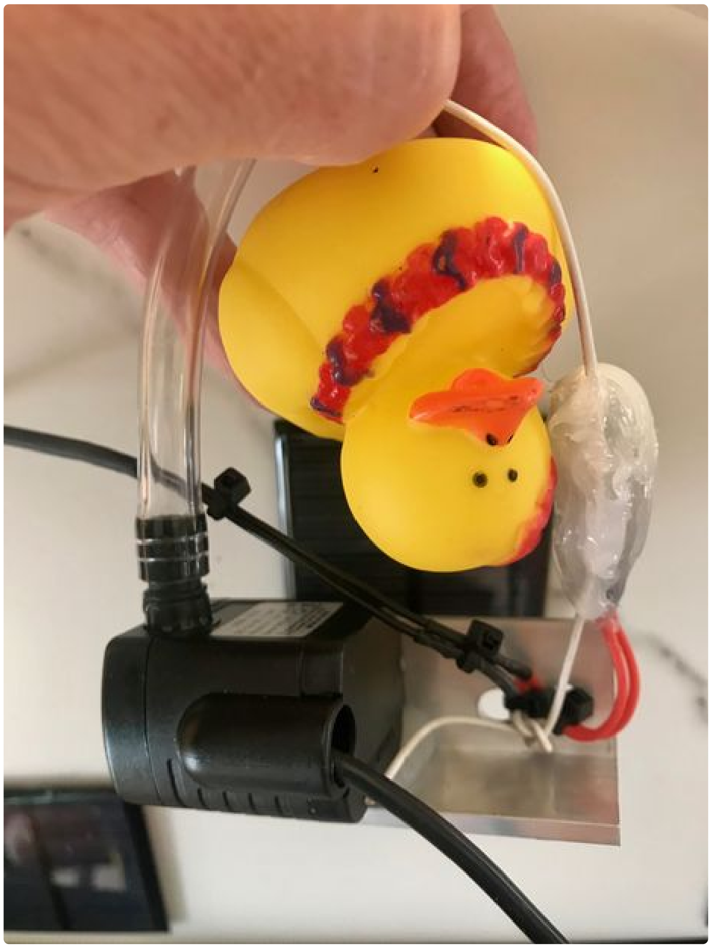
NEW
nouveau • nuevo
3/16
(5mm
LENOX
DIAMOND™
121063DGDS
Upgraded Technology for Longer Life
Tecnología mejorada para mayor duración
Technologie améliorée pour une durée de vie plus longue
Fast, Clean Cutting in Hard Materials
Corte rápido y prolijo en materiales duros
Coupes rapides et nettes sur des matériaux durs

Step 3: Build the Pump/empty Switch Holder

The pump is powered by a relay from the Lipo battery and does ok with the limited voltage, but you can't exceed the height of about 2 feet unless you use a power booster to raise the voltage. The pump is actually a champ, doesn't need priming, waterproof and has a USB plug on one end. Doesnt do well with going dry, however. The reservoir full/empty switch is simply a tilt switch that I swabbed in silicon to waterproof and then tethered to an aluminum bar support for the pump and a floating rubber ducky. The rubber ducky should be directly tethered to the aluminum bar to take traction off of the tilt switch leads. When the reservoir has water in it the ducky floats and tilts the switch--shorting to ground and allowing commands to power the relay and the pump. It also sends this data to the web and will send you a tweet if you need water. The pump is silicon glued to this support structure and than glued to the bottom of the water reservoir.





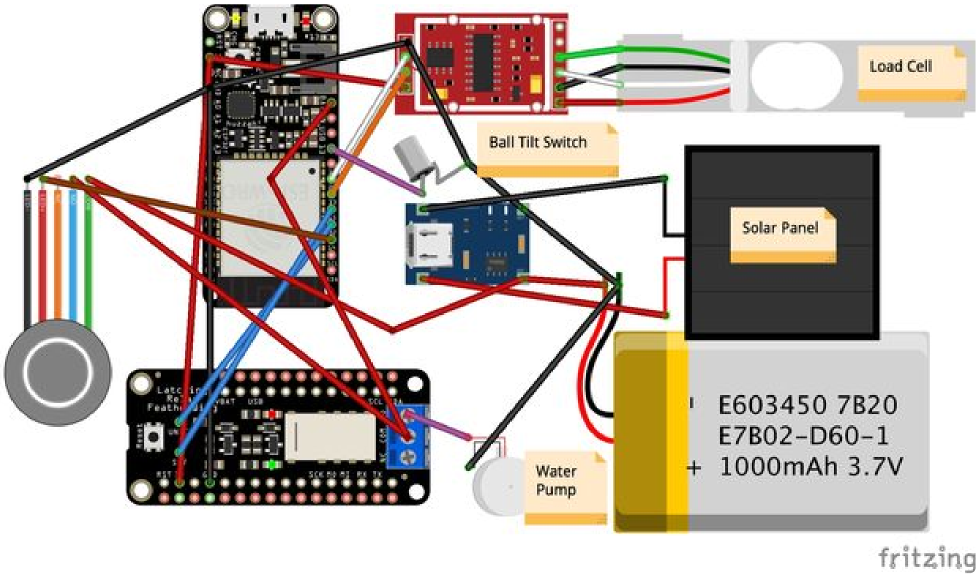


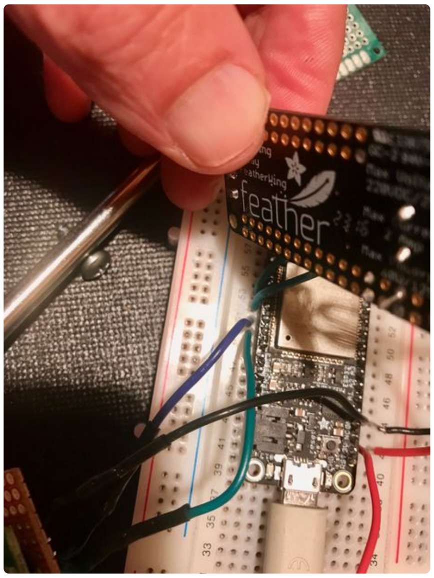
Step 4: Building the Electronics

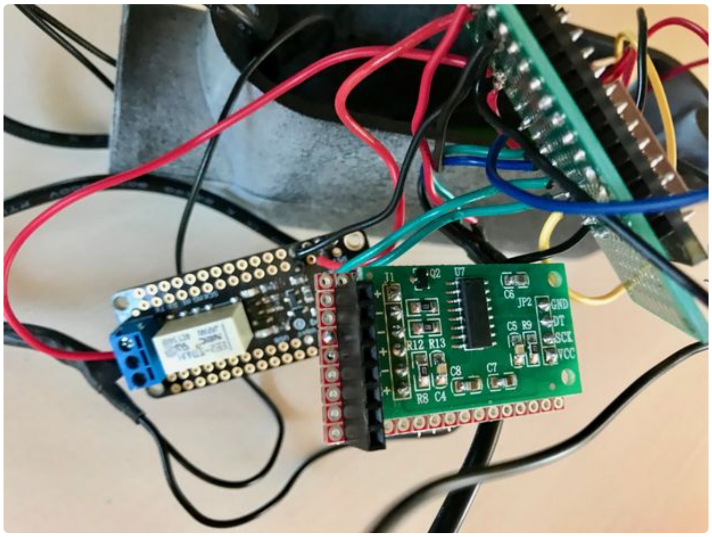
Adafruit HUZZAH32 – ESP32 Feather Board is a relatively new microcontroller and works very well in this brainy plant helper. The advantage of this board over the older 8266 is in its better sleeping ability (supposedly years instead of an hour or so...) its ability to remember what it learned in between naps (the old 8266 reset from ground zero...) and lower power consumption while napping and more pins. The great Youtuber Andreas Spiess details the changes in code to make the ESP32 do a proper job of weighing and you should watch his video if you want to know more about how the details work. The sleep example from the Arduino IDE was also used and modified for this software.

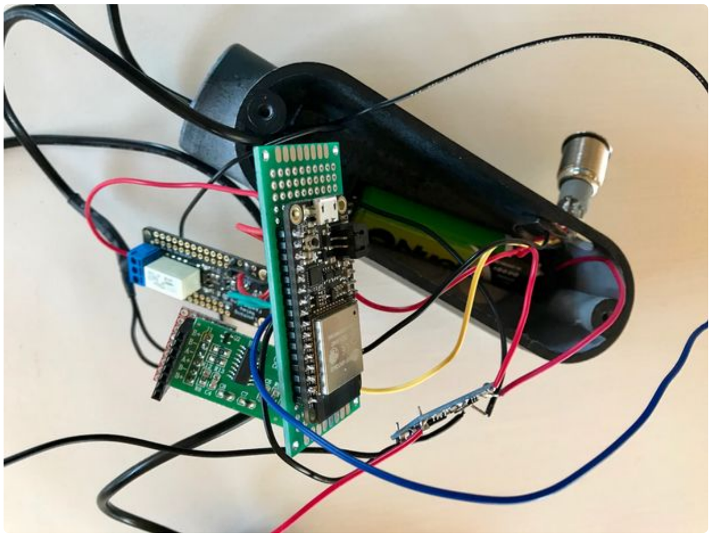
The Fritzing diagram carefully shows you all the wiring connections. Components were assembled perf boards and then wired together. The Lipo battery is your standard cheap 18650 on its own sled. The charger board is a TP4056 which Andreas says is very efficient in this solar charging role. The On/Off button with built-in LED sends power to the whole system as well as the common relay connection which powers the pump. The relay board is a nice Adafruit latching relay feather board which runs on 3 V. The HX711 amp is powered through the Adafruit and is connected up to two pins on its board.

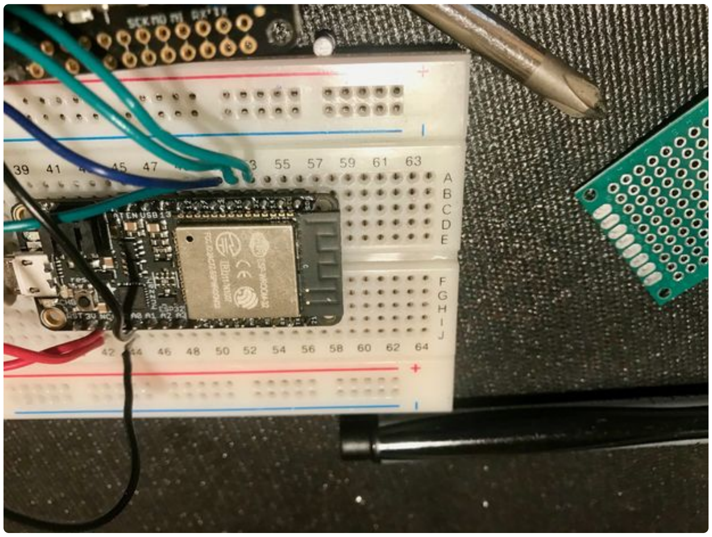
All the components are stacked in a plastic outdoor electrical box open on the lower section to allow air flow but block rain. Place the ESP32 on top to allow programming and serial monitoring with the cover off.

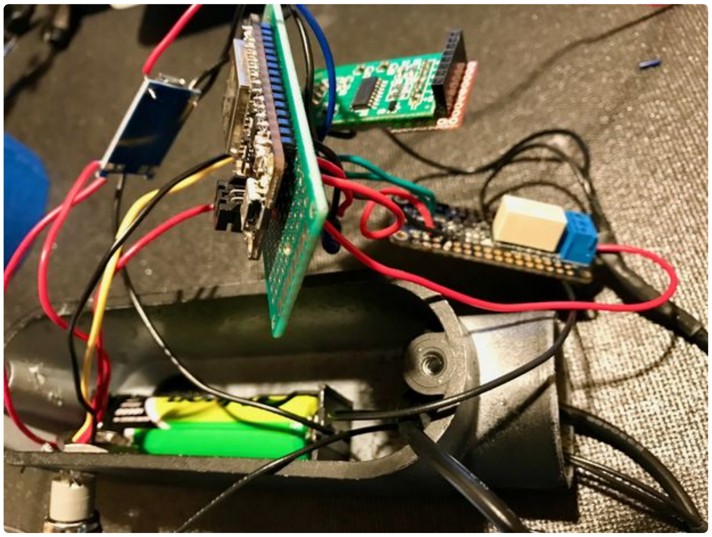

Load Cell
Ball Tilt Switch
Solar Panel
Water Pump
E603450 7B20
E7B02-D60-1
+ 1000mAh 3.7V
fritzing

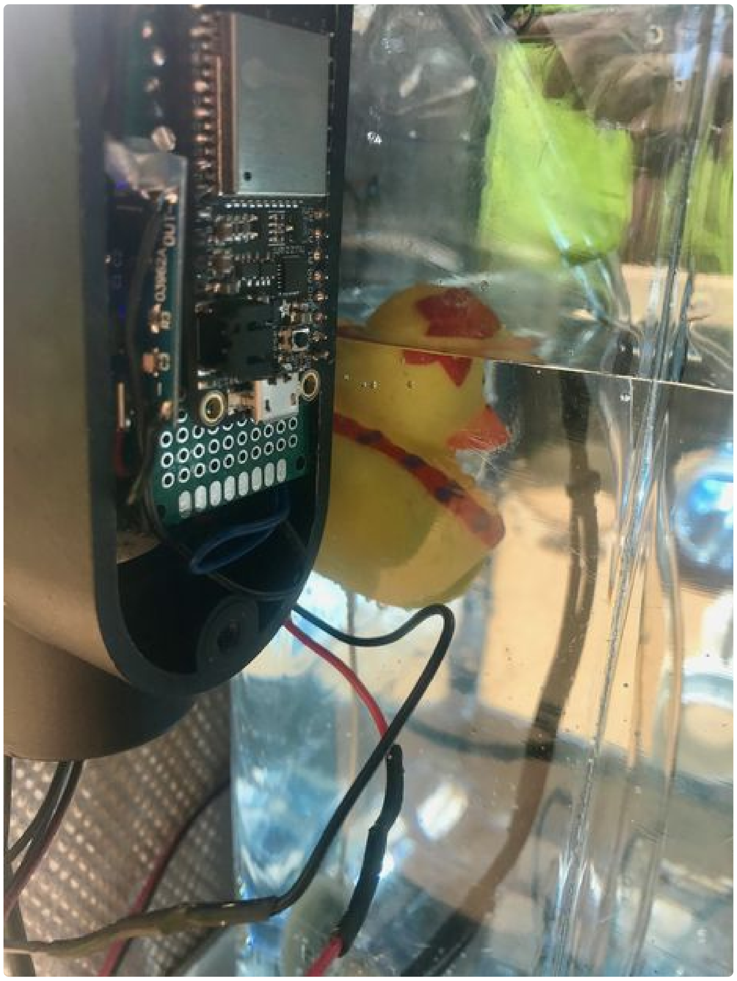








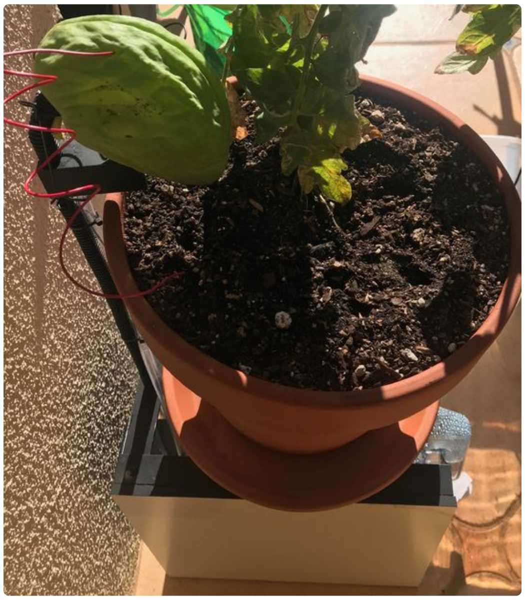




Step 5: Software

The software is pretty easy. It is basically a water-thermostat. The libraries are needed for the wifi connection and the HX711 amplifier as well as "soc/rtc.h" used for reseting the speed of the ESP32 to enable the HX711 to work properly. WaterFactor is the percent you want the water weight to vary at else it will pump and dump every time it checks the weight. Time_to_Sleep is the seconds that you want it to sleep in between measurements--it is currently set for one hour which has worked well. You can also have it turn off at night if you want. Some people have had problems with longer sleeps on the web--the ESP32 is still fairly new. The next section deals with measuring the battery level--important when dealing with solar. Remember to divide by 4096 rather than the older feather library number.

Place your WiFi credentials in the next section for upload to Thingspeak. You don't need to upload all this stuff to the web but it makes it kind of fun to follow. I won't get into the use of Thingspeak and how to upload data--this is really well covered on their website and other Instructables. The software is configured to upload the current weight of your plant, how many times watered, current power of battery, water reservoir full or not and number of times the machine has slept. In Thingspeak you can configure it to send you a tweet on your account if any of these levels goes out of wack. If you get an auxiliary vegetable brain to attach(above) this sometimes helps.



https://www.instructabl…

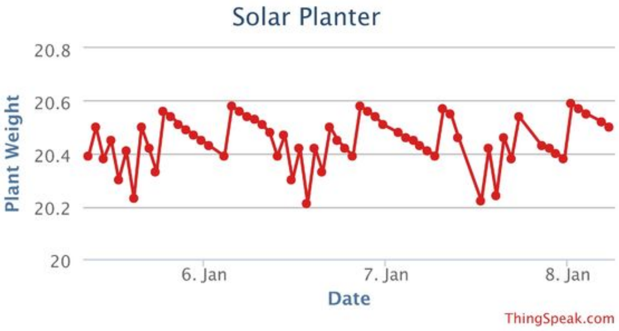
Download

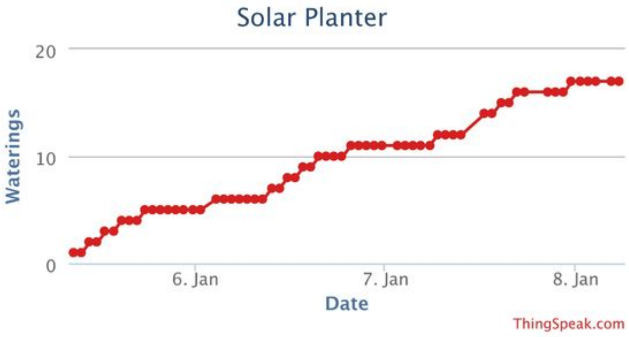
Step 6: Using It

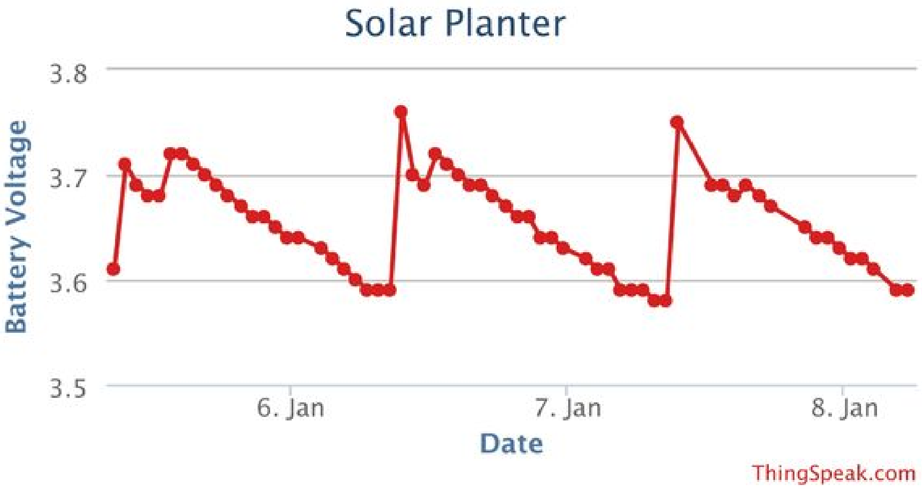
The device is simple to use. When powered up the LED on the power switch blinks until a potted plant that has been watered to a level you want to maintain is placed on the platform. After weight stabilization the computer remembers this initial weight and every hour or set interval compares the plants new weight and either corrects it with additional pumped water or reports the new weight and all other information to Thingspeak and then goes to sleep. The graphs above reflect the output over a three day time period for a tomato plant that is about 2 feet tall growing in full sun. The growth of the plant over time will obviously effect the weight of the pot and should be compensated for by redoing the initialization after a time determined by the bulking of the plant growth.

Additional software adaptations would allow for automated analysis of the plants maximum and minimum water tolerance and requirements by flooding the pot until the weight no longer changed and then measuring the slope of water weight loss over time. This will be dependent on soil type, weather and plant and root structure. Additional watering algorithms based on Thingspeak data evaluations could then be adapted. The disadvantages of weight instead of conductive sensor plant maintenance is the need for a confined watered area to weigh, but smart planters like this one are cheap, easily networked and controlled and in a weird OCD way fun to follow on the internet.

https://youtu.be/OTmJj90ZRbc

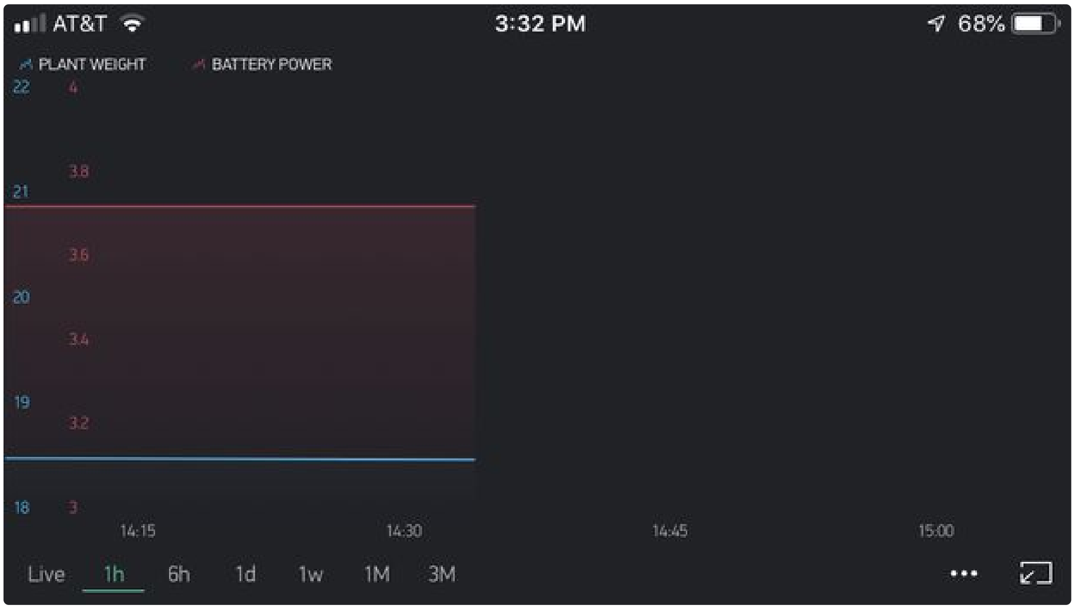


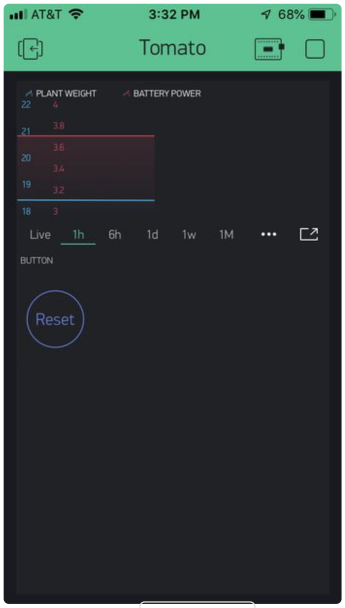




Step 7: Redo

Yea, well as designed the machine worked well for a week or so and then would have a tendency for the the ESP32 to go into a weird loop and not boot correctly and drain its battery overnight. No amount of software changing could effect this so I gave up and added an Adafruit TPL5111 to control the energy cycling of the ESP but since I could no longer use the memory as before I wrote to use the EEPROM and changed from Thingspeak to Blynk which I find more fun on your phone and a really good system. The hardware change is just a matter of connecting the TPL 5111 to power and ground, a done pin to the ESP and the Enable out to the EN pin. Make sure you put a toggle switch in-between the EN-out and the EN on the board so you can change programs and upload. I set the sleep cycle for every two hours. To clear out the EEPROM and reset the unit for a new plant or for additional weight I set up a switch in Blynk to clear out the memory and restart the weight process. The program for the new software is included above and the program on Blynk is obvious to set up. This machine really works great and produces some dandy produce. I am actually impressed with how fun the thing turned out to be---solar cells work easily and it never runs out of power.





https://www.instructabl...

Download

Nice clickbait

.........the new Make Magazine cover............