

Automated Garden System Built on Raspberry Pi for Outdoors or Indoors - MudPi



by mudpi

Do you like gardening but can't find the time to maintain it? Perhaps you have some houseplants that are looking a little thirsty or looking for a way to automate your hydroponics? In this project we will solve those problems and learn the basics of MudPi by building an automated garden system to help take care of things. MudPi is an open source garden system I made to manage and maintain garden resources built on a Raspberry Pi. You can use MudPi for both indoor and outdoor gardening projects scaled to your needs as it is design to be customized.

Today we will start with a basic setup I used at home to see how MudPi can be deployed to manage an outdoor garden and control the irrigation. In this tutorial you will learn how to deploy a main controller running MudPi. There will be some additional resources near the end for those who want to expand their setups further beyond the basics or who would like to see learn more for different setups like indoors. MudPi can be configured for a variety setups and there is a bunch of [documentation on the project site](#).

Supplies:

Feel free to add/remove any specific sensors or components you may need for your own system as your requirements may vary from mine.

General Supplies

- Raspberry Pi with Wifi (I used Pi 3 B)
 - Debian 9/10
- Monitor/Keyboard/Mouse(for Pi setup)
- SD card for Raspbian (8gb)
- Outdoor rated cable (4 wire)
- Waterproof junction box for outdoors
- Cable glands
- Din Rail (to mount breakers and DC supply)
- PVC tubing
- Drill w/ Spade Bits

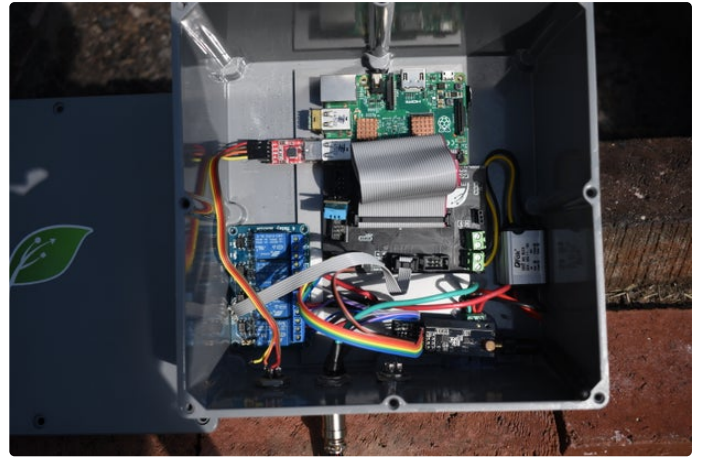
Electronic Supplies

- DHT11 Temperature / Humidity Sensor

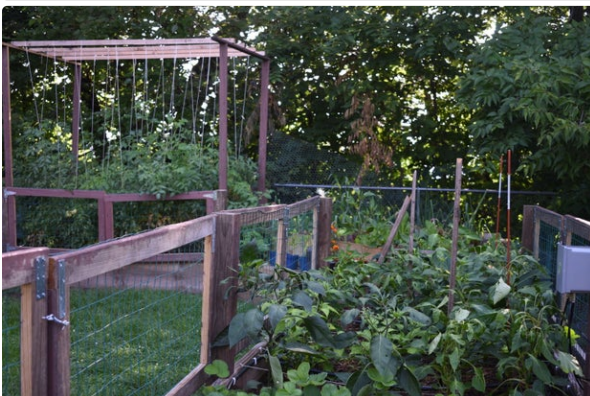
- Liquid Float Level Sensor x2
- 2 Channel relay
- 12v pump (or 120v if your using mains voltage)
 - DC to DC converter if you use 12v
- 5v Power Supply
 - or DC power supply (if powering pi from mains)
- 10k Resistors for pull up/down

Tools

- Screwdriver
- Wire stripper
- Multimeter
- Soldering Iron
- Solder
- Screws (for mounting boxes outside)
- Silicone Calk



<https://youtu.be/tVQ9ePEYinA>



Step 1: Garden & Irrigation Planning

Make sure to have your irrigation planned out if you are establishing a new system. It will be important to have these things already in place when you go to prepare the hardware so you know your component needs. Needs can change over time but it is good practice to prepare for the future. Your two main options of water delivery are either using a pump in a water reservoir or a hose with a solenoid to open and close the line. The choice will be up to you depending on your garden needs. A larger more complex system may make use of both (i.e pumping water through solenoids valves for zone watering). If you plan on using MudPi indoors you'll probably be using a pump if anything. MudPi can control your indoor plant lights using a relay as well.

Maker Tip: Keep in mind you can build your project at any scale. If you just want to try out MudPi for the first time try just something like a water bottle and 3.3v pump to water a houseplant!

Also consider water delivery options. Will you be using drip lines, a soakerhose or sprinklers? Here are a few common methods:

- Sprinkler
- Soakerhose
- Drip lines
- Manual handwater

To keep the scope of this tutorial from growing too large lets assume you already have irrigation in place and just would like to automate it. In my setup I have a tank of water with a pump hooked up to some drip lines. Lets learn how to automate that pump.



Step 2: Sensors & Component Planning

The other important planning aspect to consider is what data are you going to want to get from your garden. Typically temperature and humidity are always useful. Soil moisture and rain detection is great but may not be needed for an indoor setup. It will be your final decision on what conditions are important to monitor for your needs. For our basic outdoor tutorial we will monitor:

- Temperature
- Humidity
- Water levels (float switch x2)

I used 5 water level sensors to determine levels of 10%, 25%, 50%, 75% and 95% in a large tank. In this tutorial we will do 10% for critical low and 95% full for sake of simplicity.

You may also want to control devices in your garden. If you plan to toggle a pump or lights that don't run on 3.3v (the pi GPIO limit) then you will need a relay. A relay allows you to control higher voltage circuits while using a lower voltage to toggle the relay. For our purposes we have a pump that runs on voltages higher than 3.3V so we will need a relay to toggle the pump. Only a single relay is needed to control the pump. Although for future purposes (and because relays are cheap) I installed a 2 channel relay and left the addition slot available for later upgrades.

The most important thing to plan is power supply. How the Pi will be powered and where from. Also you should think about the devices you are using and how they will get their power. Commonly the Pi can be powered from a usb power adapter but that requires a plug on its own. If we are powering other devices with higher voltages a DC to DC power supply can be used to step voltages down to 5v for the Pi. If you plan on getting a power supply to step down voltages I recommend not going with the cheapest option.

Remember the Raspberry Pi can only support digital GPIO by default. This means you can't just hook up a soil sensor which takes analog readings to the Pi GPIO. In order to be compatible with analog components you need to make use of a micro-controller with analog support such as a Arduino or ESP32 (or ESP8266).

Luckily MudPi has support for controlling such devices as slave nodes to issue commands for multiple devices from one main controller (the pi). This makes it possible to have a main controller with multiple sensor units that it can control along with their attached analog components. I used a main controller to monitor the pump area and a sensor unit for each raised garden bed. Today lets continue building the main controller to start.

Step 3: Gather Supplies

It is time for us to gather our materials. The components and tools used in this build are all commercially available off the shelf items to make it easy for others to build their own at home. Most can be found online or at local hardware stores. The exact bill of materials will depend on your specific garden layout. For the sake of this tutorial we will keep things to the essentials as planned in order to get a running unit before going further.

on toggling components that run off mains voltage **please be CAREFUL!** Its is important that you are safe when building electronics and don't tinker with high voltages if you don't know what you are doing. With that said I did use a 120v pump in my home setup. The process is the same for a 12v pump with the main difference being needing a 12v regulator. You can also use relays to toggle lights or other devices.

Note: I would like to make note at this point if you plan



Step 4: Install MudPi on the Raspberry Pi

With a plan ready and supplies at hand it is time to prepare the hardware. To start you should get your raspberry pi prepared to install MudPi. You will need a Raspberry Pi with Wifi capabilities running Debian 9 or higher. If you don't have Raspbian already installed you will need to [download Raspbian from their page here](#).

With the image file downloaded write it to the SD card using an image writer of your choice. Raspberry pi has a [guide for writing the files to an SD card](#) if you need help.

Plug the SD card into your pi and turn it on. Connect your Pi to Wifi using the GUI if you installed Raspbian Desktop or by editing the `/etc/wpa_supplicant/wpa_supplicant.conf` file through the terminal on Raspbian Lite.

The next thing you should do after Wifi is connected is run updates and upgrades on the pi.

To update the Pi login and from the terminal run:

```
sudo apt-get update
sudo apt-get upgrade
```

Once completed reboot

```
sudo reboot
```

After the Pi has booted back up we can now install MudPi. You can do so using the [MudPi Installer](#) with the following command:

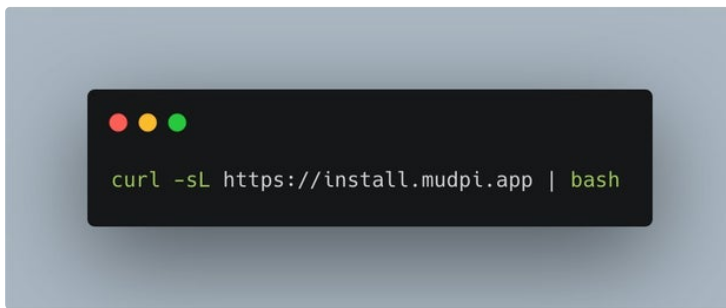
```
curl -sL https://install.mudpi.app | bash
```

The installer will take care of all the needed packages and configurations for MudPi. By default MudPi is installed into the **/home/mudpi** directory with the core located at **/home/mudpi/core**.

You can run MudPi manually with the following command:

```
cd /home/mudpi  
mudpi --debug
```

However MudPi has a supervisor job that will run it for you. Plus you will first need a configuration file before running MudPi. In order to make a configuration file you will need to know what pins you hooked up what components too which is what is done in the next step. Onward!



Step 5: Connect Sensors & Components to the Pi for Testing

The next step is to connect our components to the Pi. (Please note I was testing additional components in the photo) You may be using jumper wires and breadboards for testing which is fine, just remember to upgrade to something more reliable when you build an final unit for the field.

Connect the DHT11/22 sensor DATA pin to GPIO pin 25.

Connect the DHT11/22 power and ground.

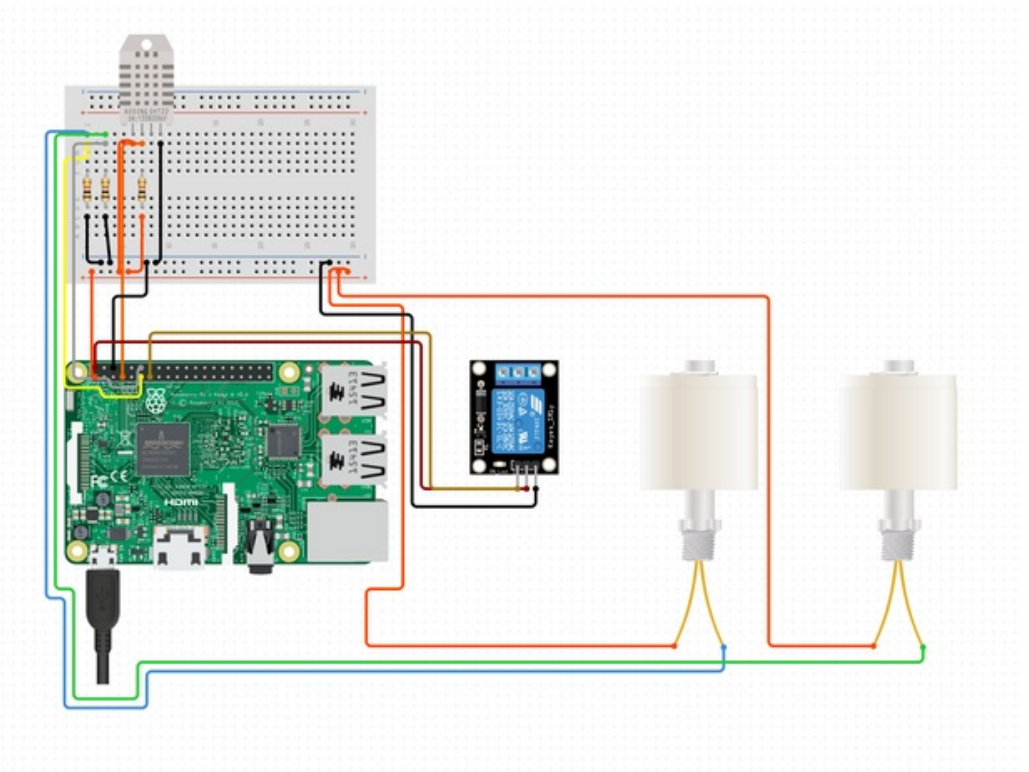
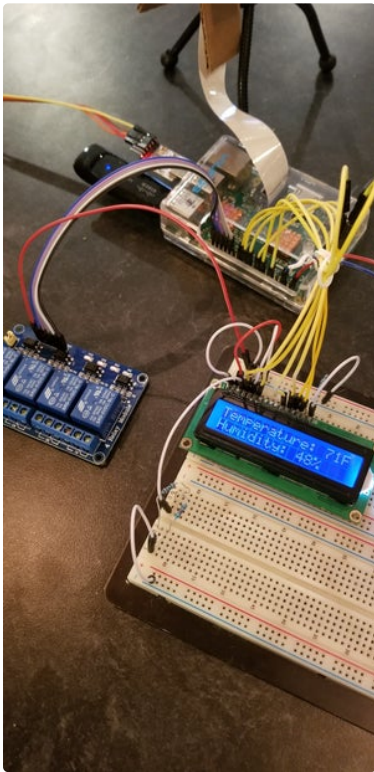
Connect one end of each the 2 liquid float sensors to GPIO pins 17 & 27 respectively with 10k [pull down resistors](#).

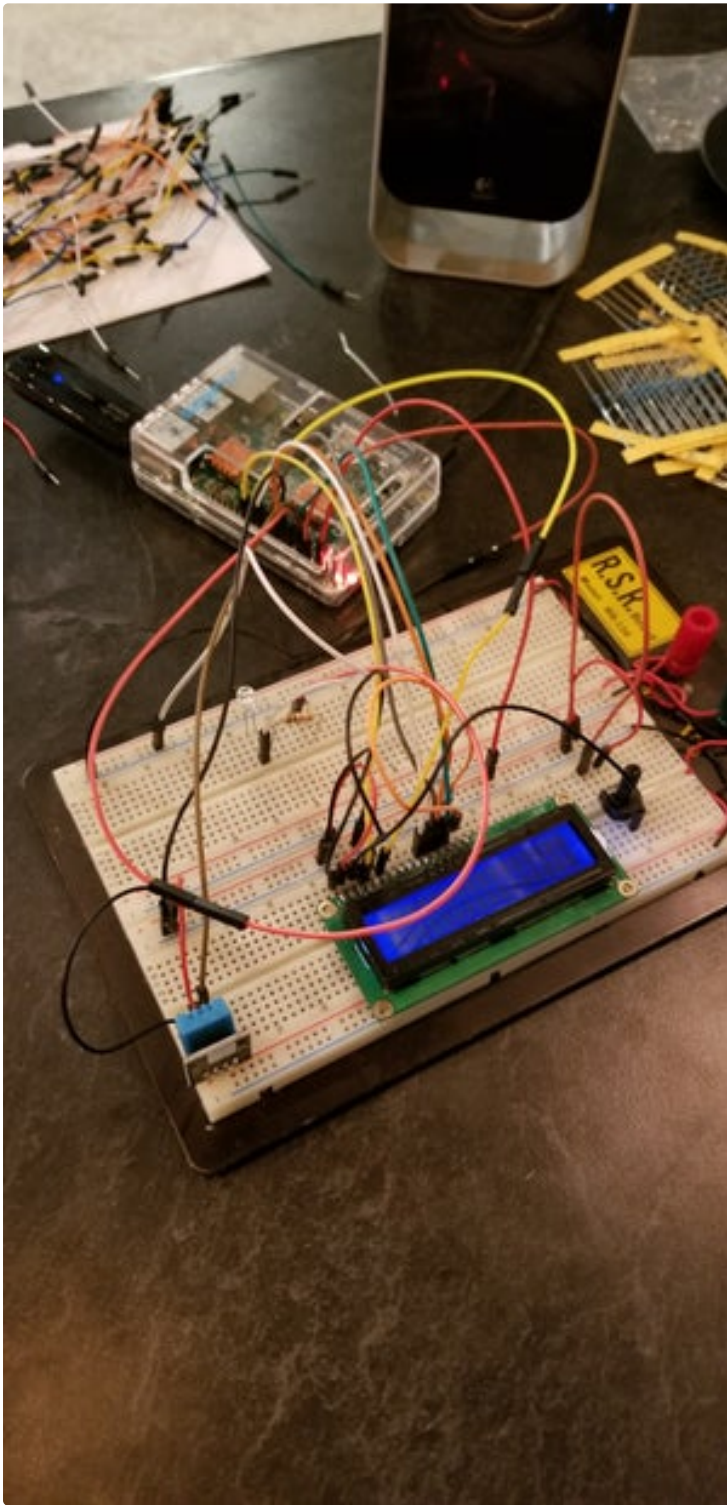
Connect the other ends of the float sensors to 3.3v so that the GPIO will be normally pulled LOW but be HIGH when the float switch closes.

Attach the 2 Channel Relay toggle pins to GPIO pins 13 & 16.

Attach the relay 5V to the power and ground to ground.

We will worry about the high voltage connections of the relay in a later step when we connect the plugs. For now we should be ready to make the MudPi configuration file and test the components.





Step 6: Configure MudPi

With the sensors and components attached you can make the MudPi configuration file and test that everything works before finishing the unit assembly. In order to configure MudPi you will update the **mudpi.config** file located in **/home/mudpi/core/mudpi** directory. This is a JSON formatted file that you can update to fit your component needs. Be sure to check for proper formatting if your having any issues.

If you are following along the following config file will work for the components we connected:

```
{
  "mudpi": {
    "name": "MudPi",
    "debug": false,
    "location": {
      "latitude": 40,
      "longitude": -88
    }
  },
  "sensor": [
    {
      "interface": "dht_legacy",
      "name": "dht",
      "key": "dht",
      "pin": 25
    },
    {
      "interface": "gpio",
      "name": "float1",
      "key": "float1",
      "pin": "D17"
    },
    {
      "interface": "gpio",
      "name": "float2",
      "key": "float2",
      "pin": "D27"
    }
  ],
  "toggle": [
    {
      "interface": "gpio",
      "pin": "D13",
      "name": "Pump",
      "key": "pump",
      "invert_state": false,
      "max_duration": 960
    },
    {
      "interface": "gpio",
      "pin": "D16",
      "name": "Extra",
      "key": "extra",
      "invert_state": false
    }
  ],
  "trigger": [
    {
      "interface": "cron",
      "name": "Daily Pump 12 Hours",
      "key": "turn_on_pump",
      "schedule": "0 */12 * * *",
      "actions": [
        ".pump.turn_on"
      ]
    },
    {
      "interface": "cron",
      "name": "Daily Pump Off",
      "key": "turn_off_pump",
      "schedule": "15 */12 * * *",
      "actions": [
        ".pump.turn_off"
      ]
    }
  ]
}
```

There is a lot going on in the configuration above. I recommend digging into the [configuration docs](#) for more in depth info. We set the DHT11 and floats in the **sensor** array and put the relay settings in the **toggle** array. The automation takes

Automated Garden System Built on Raspberry Pi for Outdoors or Indoors - MudPi: Page 9

place by setting **triggers** and **actions**. A trigger is a way to tell MudPi to listen for certain conditions that we want to take action on like temperature being too high. A trigger is not too useful until we provide it an **action** to *trigger*. In the config above there are two time triggers. A time trigger takes a cron job formatted string to determine when it should activate. The time triggers above are set for every 12 hours (so twice a day). They will trigger the two actions we configured which will just turn on/off our relay with an event emitted by MudPi. The second trigger is offset by 15 minutes so that our pump will turn on and water for 15 minutes before being shut back off. This will happen twice a day everyday.

Now you can reboot MudPi by telling supervisor to restart the program:

```
sudo supervisorctl restart mudpi
```

MudPi should now reload the configurations and be running in the background taking sensors readings and listening for events to toggle the relays. You can check MudPi is running with:

```
sudo supervisorctl status mudpi
```

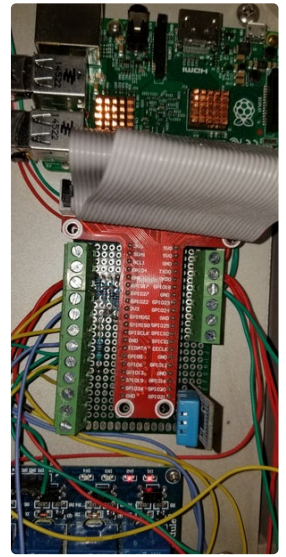
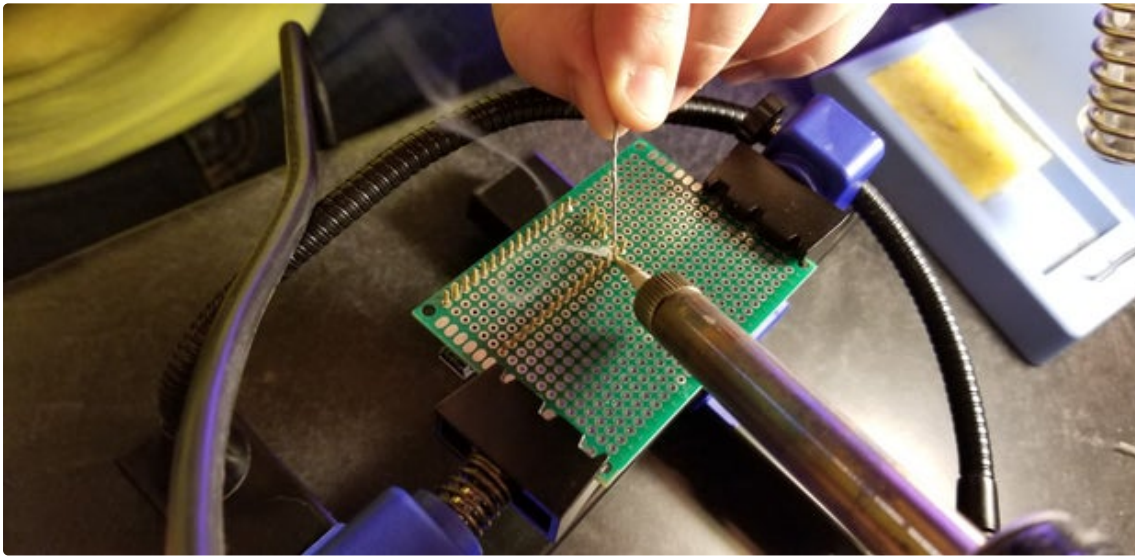
MudPi will also store log files in the **/home/mudpi/logs** directory. If you are encountering issues that is a good place to check first.

If you have verified that MudPi was running it is time to begin final assembly of the unit. Shutdown the Raspberry Pi and lets finish the assembling the hardware.

Step 7: Solder Components to Prototype Board

Now that MudPi is configured you can continue working on the hardware. Components that remain in the box should be soldered to a prototype board for more stability than jumper wires. It is not as nice as a custom circuit board but will work for now. The DHT11 sensor we are using will be external but you could optionally include an another one inside for internal box temperatures.

I soldered a pi breakout cable to a board along with some terminal connectors for easier GPIO connections once we reconnect the sensors and relay. The breakout cable made it nice to be able to disconnect the pi without having to take out the whole module. I also included the needed pull down resistors for the floats as well. With that completed we can put everything inside a nice outdoor junction box to protect it.



Step 8: Start to Put the Electronics in an Outdoor Junction Box

By this point everything has been tested working on MudPi and it's time to assemble the outdoor unit to stand the elements. Your local hardware store will have a selection of junction boxes in the electronics section you can purchase for under 25\$. Look for one that is the right size and has a watertight seal. I spent a little more to get

a fiber reinforced box with spring latches. All you need is something that will keep moisture out and fit all your components. You will be drilling holes in this box to route cables out as well.





Step 9: Connect Plugs to Relay and Install in Junction Box ***Warning High Voltage***

The Pi should be powered off when connecting components. If you are using 120v or 12v for the pump consider the plug to use. Pumps running 12v commonly use a barrel jack connector. Working with 120v you can work with a female extension cord plug. Now don't go cutting an extension cord and messing around with this without proper equipment.

Using a drill or spade bit drill two 3/4in holes in the bottom of the outdoor junction box and put two 3/4in cable glands in. Run the male extension cord through one gland and the female half through the other. If you want to use the other relay channel install another female ended cord in.

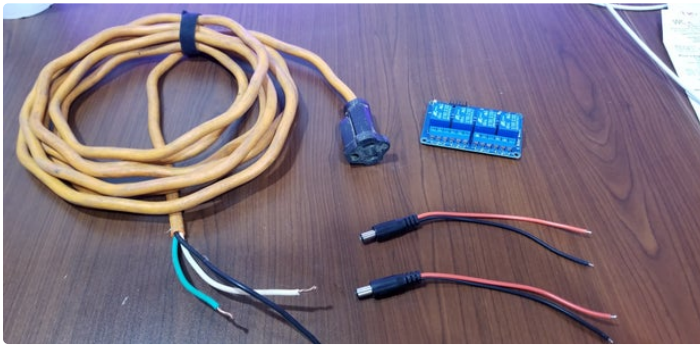
In the box I have installed a small section of din rail. On the rail is a DC Power Supply to step down the 120v to 5v to power the Pi on as well as some safety breakers. I am only using two breakers so that I can shutoff the Pi without shutting off the entire system. One breaker would suffice. Now inside the extension cord there are three colored cables. WHITE is neutral, GREEN is ground, and BLACK is 120v+. The green and white go directly into the DC Power supply. The black first goes into the breakers then to the DC power supply. On the power supply is a small screw which is a potentiometer to trim the voltage right to 5v.

We are going to be using terminal blocks to make

connections between the plugs. Using one block connect all the white neutral cables together. If you don't have terminal blocks electrical tape will suffice. The green ground cables should also be connected together. The relay high voltage side has three connections: COM (common), NC (normally closed), and NO(normally open). Depending on your relay it may only have NC or NO not both. Connect a small bit of extra cable from the breaker that will supply 120v to our relays COM (common) terminal on the high voltage side. Now connect the female extension cords black 120v line to the NC terminal. This will mean the plug will normally be off and not connected but when we toggle the relay on it will supply 120v to the plug thus turning on our pump.

At this point all the extension cables should have their white neutrals tied together and their green grounds tied together. The female cords have their black 120v attached to the relay NC terminal. The male extension cord should have its black live routed to a break on the din rail and then split to the DC power supply and the COMs of the relays.

It is important to install everything in a waterproof box and properly protect/route all your cables. The last thing you want is a fire or someone getting zapped. Also don't mess with high voltage if you are not able to be safe. You can still do quite a bit with 12v and lower components.

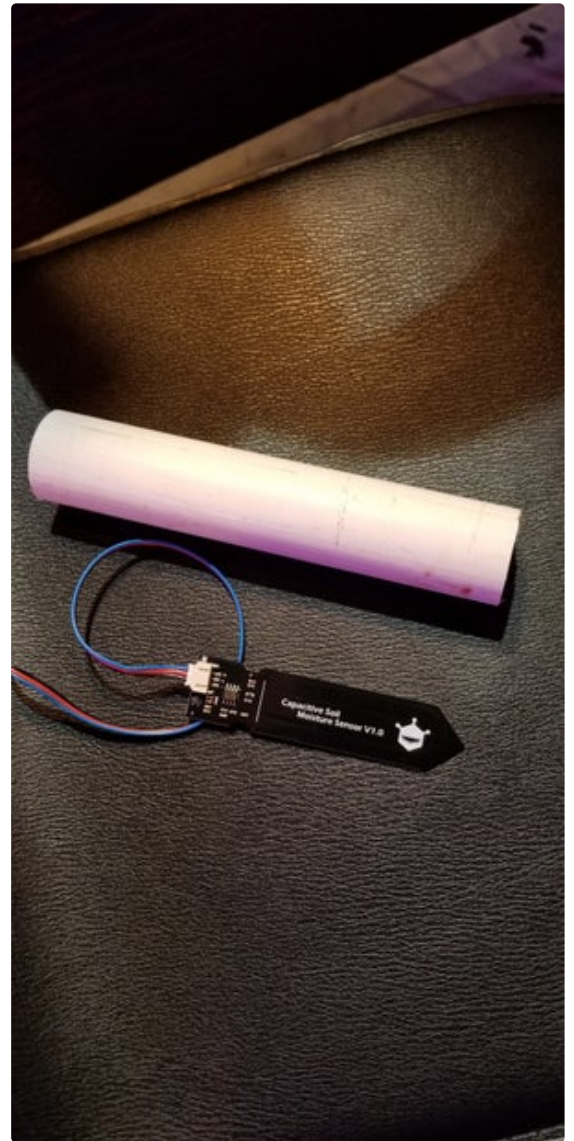


Step 10: Put Sensors in Protective Housing

Nature and moisture is not too friendly to electronics. You have protected the Pi with the outdoor junction box but now you need to protect any external components. You can make some decent housing to protect external components using some PVC pipe or other bits of scrap tubes. I rigged up a simple vented cap for the DHT11 sensor to help protect it from rain and bugs but allow it to breath for accurate outside readings. Use silicone calk to seal around the cables in the next step.

Not the best solution but it works for a cheap 4\$ sensor. (I also made some for soil sensors I was testing at the time as well.) The float sensors will be installed into the water tank and do not require additional housing.

You will also find that the sensors usually only come with some cheap thin gauge wire. This wont last long to some general handling or outside climates. In the next step we address this.





Step 11: Connect Sensors With Outdoor Rated Cable & Plugs

Getting some outdoor rated cable is a must if you want to have external sensors connected to the box. Outdoor rated cable has shielding to help protect the internal wires. I picked up some 4wire cable and plugs. You do not need the plugs and can instead use more cable glands but I wanted to be able to quickly swap out sensors.

Cut some cable to length for your temperature sensor and float sensors. I would give it a few extra feet as it is always nice to have extra to cut if needed. I suggest soldering the cables for best connections and then

wrapping with electrical tape. I suggest using the same color for power and ground with each wire to make things easy to remember. Tuck the cable into the housing with silicone calk seal the rest of the bottom of the housing so only the vented cap is the entry point.

The other end of the cable you can run into the box through cable glands and connect to the Pi on the same pins as before. If you choose to use plugs install the plug ends onto the cable. Drill and install the other ends into the junction box and then connect the internals.



Step 12: Install Float Sensors Into Tank

With the other sensors protected and ready to go its time to install the float sensors into the water tank. Since we are only using two you should install 1 at a critical low level that the pump should not run and one that should mark the tank is full. Find the correct size drill bit and make a hole into the tank at the right levels. Screw the float sensors into the tank with the washer and nut provided. Look inside the tank and be sure the float sensors are orientated so that they are in an off position



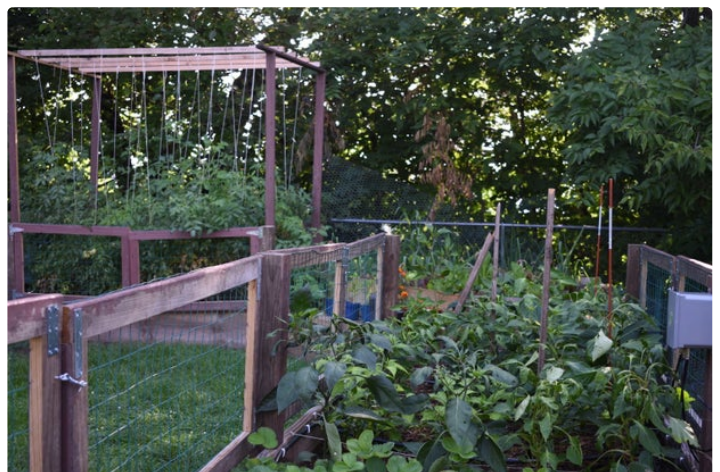
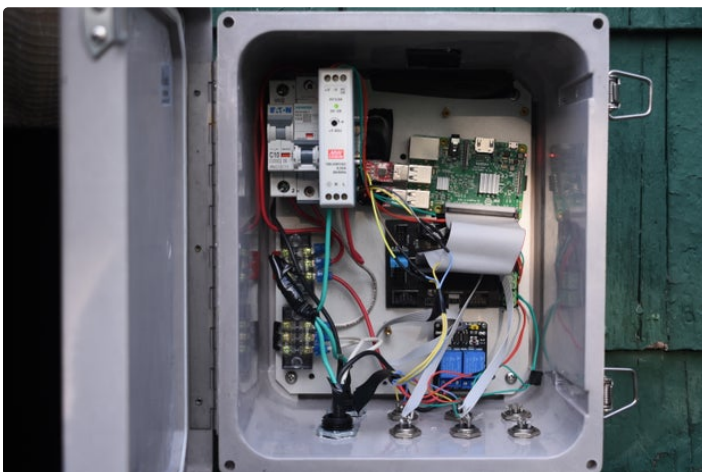
and lift up when the water raises make them close the circuit.

Because of the pull down resistors this means when the water level is met the float sensor at that level with read 1. Otherwise the float sensor will return 0 if the water is not currently lifting the sensor closing the circuit.

Step 13: Deploy the Unit Outside

The MudPi unit is field ready and we can mount it outside in its final location. The outdoor junction box typically comes with a cover to screw down to make the water tight seal. You should also find some mounting holes on the back to use for mounting the unit. I installed my box right next to the water shed outside since the float sensors only had a limited cable run.

You can plug the male extension cord into a socket and flip the breaker to bring MudPi online. Make sure that everything is working before leaving it for an extended period. Test that the sensors are taking readings by looking into redis for stored values or checking the MudPi logs. If everything looks good then its time to let MudPi work while you relax.





Step 14: Monitoring MudPi

Now that MudPi is working you might wonder the ways to monitor your system. The easiest and most direct way is to monitor the MudPi log file:

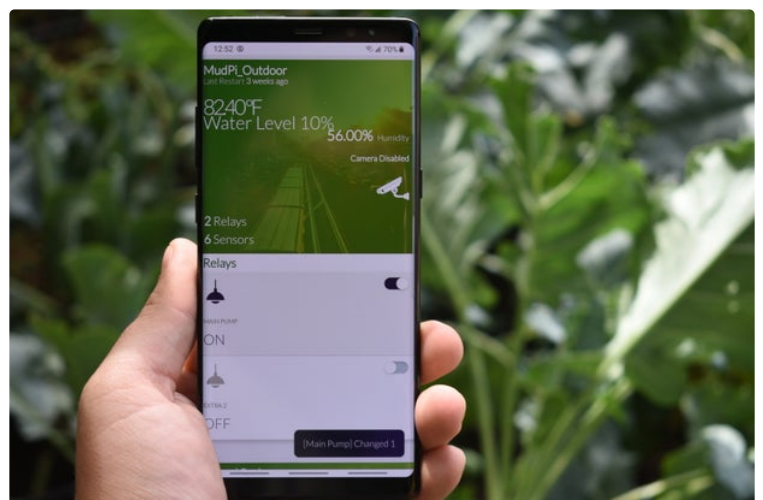
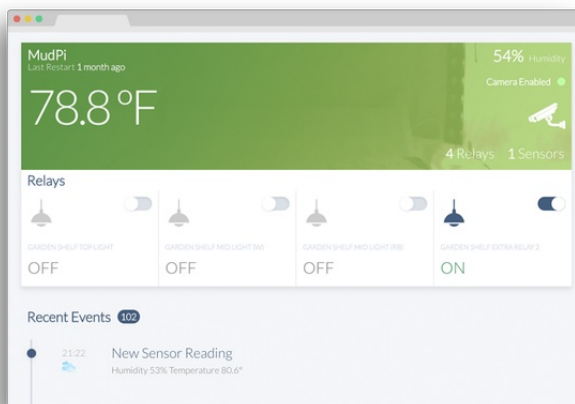
```
tail -f /home/mudpi/logs/output.log
```

Another option is through an interface like a local webpage. I haven't had time to release a public MudPi UI yet but you can easily grab your sensors and component state from redis with PHP. Learn how MudPi stores your data in redis [more in the docs](#).

The latest sensor readings will be stored in redis under the **key** option you set in the config. Using this you can make a simple PHP application to grab the readings on page load and display them. Then just refresh the page for new data.

It is also possible to listen for [MudPi events on redis](#) and this is a better option to get realtime updates from the system. You can read the events directly through the redis-cli

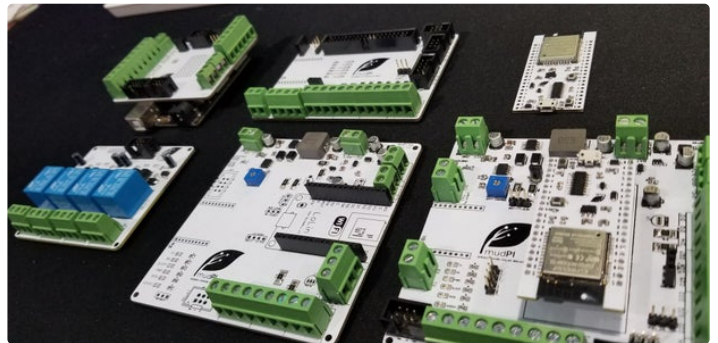
```
redis-cli psubscribe ***
```



Step 15: Replace Prototype Boards With Custom PCBs (Optional)

I have gone a bit farther and made some custom circuit boards as well for MudPi. They help me to speed up the build process with building multiple MudPi units and are far more reliable. I have started replacing my old prototype boards with more reliable PCBs in all the existing units I have. In the future I want to make these boards available for sale in small quantities to help

support my open source work. MudPi does not require any custom circuit boards to run, it just helps reduce the hardware workload with onboard components already installed such as the pull down resistors and temp/humidity sensors.



Step 16: Relax & Watch Your Plants Grow!

Now you have your very own automated garden system that you can expand and scale as you desire. Make more units or expand the one you already built. There is much more you can do with MudPi and a lot of info on the project website at <https://mudpi.app>. My goal was to make MudPi into the resource I was looking for when I started on the garden project. I hope you find great use in MudPi and share the word if you like the work I do. I personally use MudPi both outside and inside at home to manage my plants and have been very happy with the results so far.

MudPi is still being update with more features and developments. You can visit the site for details on what I have been working on and check out some of the links below to guide you to some more resources. I also entered MudPi in the 2020 Raspberry Pi contest. If you like MudPi and want to help me out give me a vote below.

Useful Resources for Going Further

[MudPi Documentation](#)

[MudPi Source Code](#)

[MudPi Guides](#)

Share Your MudPi Build

[@MudpiApp](#)

Support My Work on MudPi

[Support MudPi](#)

Happy growing everyone!

- Eric

Made with♥ from Wisconsin



Hello!

Is this project still supported? I'm hung up on the installation phase of mudpi onto my raspberry pi. The installation keeps quitting near the end, immediately after the nanpy install. It'll say "successfully built nanpy" and then

MudPi Install Error: Problem installing MudPi core python package

I've tried building python 3.7 before starting the mudpi installation, everything is updated and upgraded. What am I missing?

I love your project, I'm nearly done as soon as i can get the software to do what i need it to do! Please help!!



yes it is being worked on every day. In fact the reason you are getting this error is because the project is in the middle of the biggest update yet. The installer is trying to install the master branch but right now the updates are still on the feature. You need to install from the other branch to get the updates until I finish the update and merge the changes to master branch.

You can use this command with the installer: ``curl -sL https://install.mudpi.app | bash -s -- -b feature``

This is mentioned on the bottom of the installer docs too.

That should get you up and running the newest version.



Fantastic!!! Thank you so much! I'll give that go tonight!
I really really appreciate your response! I'm excited for this project!!



I'm still beating my head against the wall with this project. I build houses, this may be beyond my poor monkey brain.

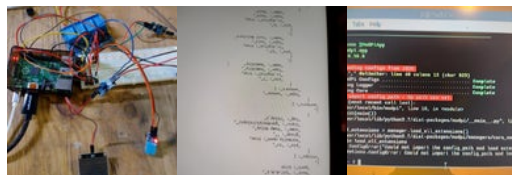
I'm coming into more errors when I try to start mudpi.

"Problem loading configs from JSON"

And

"Could not import config_path - No path was set"

I'm sure these are simple fixes that are escaping me, any help would be appreciated



Hey, we're you able to get the config file to load?



I have updated the article to reflect the new version of MudPi. Much has changed and there have been several big updates. Previously the old config listed here would need to be migrated using the mudpi migrate. ``mudpi --config old.mudpi.config --migrate_config new.mudpi.config``

I have migrated this config and also changed some of the info to reflect the new changes. Hope this helps everyone. Happy growing!



Thanks, got the same error and now it installed, unfortunately, after installation, I don't have a WIFI or CABLE connection anymore. Is this supposed to happen or should I rather wait until the new release?

I've reimaged my Pi 3 times with the same result. Can't wait to check out this project as this is exactly what I'm looking for!



I think you have the Auto hotspot (AutoAP) installed and it possibly started the hotspot for you. This is a script that checks if there is a wifi connection every 5 mins and if not starts up a hotspot so you can still have a connection to the pi. Sometimes on a fresh boot the Wifi isn't fully loaded up so it may miss the first check and start the hotspot. It should stop the hotspot and connect to your saved wifi on the next check.

If you don't need this or have some issues you can try removing the Auto hotspot by editing your crontab and removing the call to auto_hotspot.

In your ``/usr/bin/`` folder you will find the actual start_hotspot and stop_hotspot script you can remove.



Son of a BEEEEEEEEPPPP, How did you get the output to the screen!?!?!? I have been working

on a project based on OSPi (open sprinkler pi) and trying to run 8 relays while also getting some output to the screen. I am totally unaware of how to write programs etc but can follow instructions from multiple sources and tweak and assimilate. The think that stumped me was running the I2c pins so that it could be used for showing which zone was active. Could not find anyone who scratch built it with raspberry and opensprinkler. Wish i knew how. Any pointers? By the way, amazing project, this will probably fit my current needs better. Going to be building a proper garden and want it as automated as possible.



Maybe give MudPi a shot then. Most of what you need to do is just updating a config file. The major heavy lifting is done for you behind the scenes.

I am personally using I2C displays connected to my raspberry pi.



Fabulous. I'm in! Will begin to assemble and build. I am UK based but would order the PCB boards. Let me know how much and please add me to the next order.



I am very new to raspberrypi. I have no idea what I am doing. Could you tell me how to add this app to my raspberrypi. Thanks for any help in advance.



A well deserved win in the Raspberry Pi contest! This is excellent work. Thanks so much for putting this out into the world!



Congratiation!
Amazing project.



I would love to be able to buy the custom boards, or out of the box sourced components for a project like this! Keep us posted.



For sure! I was aiming to start a few small batch sales within the next month or so.



Oh my.. It's really amazing!! Good job. I've voted but I count't see the entire Instructable until now!



Very nice setup :)