Overview

reef-pi is a DIY reef aquarium controller based on Raspberry Pi. It eases maintaining reef aquariums by automating day to day reef keeping chores. [Reef aquariums](https://adafru.it/CqJ) are saltwater aquariums with live corals. Keeping corals alive in the captive environment requires precise lighting, strong water movement, and more stable water chemistry (salinity, pH, calcium etc) than fish-only marine aquariums.

The controller software runs on Raspberry Pi and uses ancillary electronics for all of its functionality. It's a modular controller and can be configured to run only a small set of features. Due to the open nature of the project, it's also easier to adopt reef-pi electronics to accommodate new or custom hardware (pumps, lights etc). This is the first guide in reef-pi series which provides an overview of the controller software features, installation process and finally testing it with some common electronics (LED dimming, etc.). Though reef-pi projects main goal to automate reef aquariums, it can be useful in freshwater aquariums or even general purpose equipment control (e.g. a web-enabled powerstrip or building a Christmas lighting setup, etc.)

Following is a list of features reef-pi provides (also called as modules):

1. **AC equipment control.** On-demand or periodically, using timers.
2. **LED light control.** On-demand dimming or simulate diurnal cycle with LED lights.
3. **Temperature control.** Monitoring tank temperature and switching on/off heater and chillers to maintain it within a range. Alert if it goes outside the specified temperature range.
4. **Salinity control,** by detecting water level and when it falls below a threshold (due to evaporation) pour fresh water into the tank.
5. **Automated dosing.** Periodically add chemical reagents (e.g. calcium & alkalinity supplements) or liquid food in the tank.
6. **pH monitoring,** continuous monitoring of tank pH and alert if it goes beyond a specified range.

All in one (multiple modules) controller based on reef-pi.
reef-pi dashboard, configured with all the modules, where each module is represented as individual tabs.
**Parts**

You will only need a working Raspberry Pi to play with the software. I highly recommend the kit version as it comes with SD card, power supply and hdmi connector. Pi Zero, 2, 3 and even Pi B will work.

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Item</th>
<th>Price</th>
<th>Stock Status</th>
<th>Add to Cart</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>LED</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td><strong>Resistor</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td><strong>jumper wires</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td><strong>Breadboard, 1/2 size</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Raspberry Pi Zero W Budget Pack - Includes Pi Zero W

$34.50
IN STOCK
ADD TO CART
Configuration and Installation

reef-pi requires a Raspberry Pi running Raspbian or Noobs. Adafruit already has guides on [Preparing SD card](https://adafruit.com/dDL), [First-time configuration](https://adafruit.com/dDM), [Network](https://adafruit.com/dDN) and [SSH](https://adafruit.com/jvB) setup. We'll not repeat those, if you are not familiar with Raspberry Pi, use those guides in exact order to get a working Raspberry Pi with latest Raspbian installed and network configured.

It's always recommended you update Raspbian before installing reef-pi. We'll also enable network time to keep Raspberry Pi time up to date from the Internet.

```bash
sudo apt-get update -y
sudo apt-get upgrade -y
sudo systemctl start systemd-timesyncd
sudo systemctl enable systemd-timesyncd
```

reef-pi uses Raspberry Pi's [hardware timers](https://adafruit.com/CqK) for PWM (used for LED dimming, DC motor control etc.). This allows reef-pi to use up to two PWM channels without any ancillary electronics. This feature is only available in newer Linux kernels (4.9 and above). We have to edit `/boot/config.txt` file and add this line at the end to enable (then reboot)

```bash
dtoverlay=pwm-2chan
```

reef-pi uses a DS18B20 temperature sensor, which requires the one wire protocol to be enabled in the Linux kernel. reef-pi also uses I2C communications protocol for some features. We'll use the `raspi-config` tool from terminal for enabling both one wire and I2C.

```bash
sudo raspi-config
```

From the terminal type `sudo raspi-config`

Select Interfacing Options
Then select 1-Wire

Select Yes to enable, then restart

Update Hostname

You can use the `raspi-config` tool to rename Raspberry Pi to something else. For example, I named one of my reef-pi controllers as `biocube29` (the name of the tank). Doing so allows accessing the reef-pi web UI as `http://biocube29.local` from other devices in the same network.

Disable SPI

`reef-pi` does not use SPI, and disabling it will allow reusing the SPI pins as normal GPIO. The `raspi-config` utility can be used for this as well (disabling SPI).

Reboot and Install reef-pi!

Finally, we’re ready to install the software.

`sudo reboot` the pi to make all the changes active
Next, download the latest (https://adafru.it/FcH) release of reef-pi and install it

```bash
wget -c https://github.com/reef-pi/reef-pi/releases/download/2.4/reef-pi-2.4-pi0.deb
sudo dpkg -i reef-pi-2.4-pi0.deb
```

That's it! reef-pi should be installed and running, you should be able to access the reef-pi UI by visiting the Raspberry Pi IP (or hostname.local, if you have set a hostname) in your browser. You'll get this login prompt:

```
reef-pi
```

Enter `reef-pi` as both username and password and click "Sign In". This is the default credentials reef-pi ships with. You will then redirected to the main dashboard:

```
The error message at the top is benign and expected, and an indicator of a fresh install (if you are curious, the error is about missing data for charting, since there is no data yet). Notice the footer information, it reflects the controller name, current time, reef-pi version, uptime, IP and any error.

Next, let's customize the reef-pi installation for our specific use. For this, head over to the **configuration** tab and change the controller name from `reef-pi` to `adafruit-test` and click on the **update** button to save the setting.
The new name will be reflected upon reloading the controller, which can be done from the admin section under the configuration tab.

Click the reload button and refresh your browser.

After browser refresh, you should see the new reef-pi name changed in footer information.

At this point, reef-pi is running and responding, nice work!

We will next take a brief look at the UI and modules in general and test some of them using simple LEDs.
UI Overview

At the top of reef-pi web UI, there are tabs to represent categories of functionality (also called modules). For example, the temperature controller features are under the "temperature" tab. There are few tabs that are not specific to a module and relevant to the controller as a whole (like "configuration" and "dashboard"). At the bottom is footer data, which provides meta information about reef-pi including version, current time, IP, uptime (how long reef-pi is running) and the number of serious errors (if any).

Most of the time, the first tab or landing page is a dashboard. This is the default configuration that reef-pi ships with. It is possible to disable the dashboard from the configuration module.

Configuration

The Configuration UI holds all the meta information about reef-pi controller. reef-pi configuration shows what modules are active as well as features available in them. This is a brief overview of individual sections as relevant in this guide. It is recommended that every new reef-pi build goes through these steps to ensure appropriate setup and avoid hard to diagnose issues later in the build process.

Admin

This section of the configuration holds administrative buttons that allows reload, reboot, power down and sign off.

Most of the changes in configuration UI (like telemetry, authentication, settings etc.) requires a reload. A reload only restarts the controller software without restarting the whole Raspberry Pi. The Reboot button restarts the whole Raspberry Pi including Raspbian and reef-pi software. The Power Off button shuts down the Raspberry Pi, making it safe to unplug the power supply. It is recommended to power off the reef-pi controller before unplugging the power cable or adding new sensor, etc.

Settings
This is the section where the individual modules are activated or deactivated (under capabilities). Controller name, network interface (ethernet or wifi) that is used to detect the IP address, and various other global configuration (such as https) are enabled here. Any changes in settings require a reload to come in effect.

**Authentication**

By default, reef-pi ships with the username and password set to reef-pi. You should update it to your preferred username and password. I highly recommend using a password manager and set a 32 character random password (generated by a password manager). **Note: As a security posture, reef-pi does not show any password that's previously set in the UI.** So, if you notice the password field is blank, it's normal and expected. Once updated, username and password changes will require a reload of reef-pi for it to come in effect.
Connectors

All equipment is physically connected to the reef-pi controller using connectors. Different modules in reef-pi use different type of connectors and they are all defined here. reef-pi has three types of connectors.

1. Outlets: These are digital output pins, they are used for equipment control, dosing pump direction control etc. Each outlet has a name and a GPIO pin number. **Note: the pin number in outlet definition represents the GPIO pin number not the serial number.** For example, Raspberry pin 11 (based on location) is GPIO pin 17, hence in reef-pi UI 17 is specified as the pin number for the corresponding outlet.

2. Inlets: These are digital input pins, they are used for water level sensors. Just like outlets, they have a name and pin number. **Pin number represents GPIO pin number not serial number.**

3. Jacks: These are groups of PWM output pins. They are used to control light, dc pumps and motors. Each jack can have more than one pin (specified by comma separated pin numbers). reef-pi has the ability to use Raspberry Pi's native hardware timers as well as PCA9685 ([https://adafruit.it/dUG](https://adafruit.it/dUG)) breakout boards to generate PWM output.

Dashboard

Dashboard and charts are special capabilities of reef-pi that can be enabled and disabled from the settings section under the **configuration** tab. When enabled, the dashboard module allows reef-pi users to display a customizable set of charts and visualization.
The exact type of charts available in the dashboard depends on what features are enabled and in-use, since charts are generated from usage data. For example, if the temperature module is enabled, and a temperature probe is configured, then a single temperature chart will be available on the dashboard. If there are multiple temperature sensors configured, the dashboard module can showcase multiple charts, each representing one sensor's data.

reef-pi dashboard can be customized by specifying the grid (rows and columns in the dashboard) and individual chart sizes (height and width in pixels) to fit their specific devices (monitor, tablets etc). The Customize button is located at the bottom right of the dashboard.

Now that we have covered the basic UI components, let's use an LED to test a few modules of reef-pi on the next page.
Testing

Equipment and Timers

Let's test reef-pi's equipment module with some prototype electronics. For this, we'll make an LED blink on-demand and then using timers with reef-pi. Power off reef-pi and prepare the following circuit.

Connect Raspberry Pi GPIO pin 24 to the positive end of a common LED using a jumper wire. Negative end of the LED should be connected to any of the Raspberry Pi's GND pin via a 470-ohm resistor.

Once the prototype circuit is in place, power up reef-pi and head over to configuration tab, under the connectors section. Create an outlet with GPIO pin 24.
Next, head over to the equipment tab and create an equipment with the outlet:

Once the test equipment is created, you can click on the on/off toggle switch and the LED should turn on/off accordingly.
Jacks and Lighting

In reef-pi, jacks refer to PWM output pins. They are used in dimming lights or controlling dosing pumps etc. reef-pi supports native Raspberry Pi hardware timer-based PWM or PCA9685 board based PWM. In this guide, we'll use Raspberry Pi based PWM.

A Raspberry Pi provides two hardware-based timers and, by default, they are on GPIO pins 18 and 19. The Linux kernel uses these two pins to provide two PWM channels, which it calls 0 and 1. Thus in reef-p, jacks numbers 0 and 1 represent Raspberry Pi GPIO 18 or 19 based PWM output pins.
Let's create a jack with pin 0 and type rpi. Make sure to choose rpi as the driver from the dropdown.

Next, create a light and associate the jack with it.
Once the new light is available, click on it, edit it and change set the profile to manual. Set the slider to 50% and click on update.

You should notice the LED wired to GPIO 18 turn on. You can verify the LED is being dimmed as you update the light's value in reef-pi using the slider and click save. We'll use the same circuit with appropriate transistors to make a full blown light controller in a later guide.

Equipment and light prototype circuit on a breadboard:
Alert setup and test email

Next, we'll configure reef-pi to send email alerts when something bad happens (for example, a temperature sensor returning higher than expected readings). Alert configuration is specified in the configuration tab under the telemetry section. It's highly recommended you use an app specific password instead of your main email password. User's email should always be protected by two factor authentication. For gmail users, app passwords can be generated here (https://adafruit.it/CqM). Similar things exist for Hotmail, Yahoo, and most other online email users.

After all the email configuration is saved, click on the update button and reload reef-pi. Once reef-pi is reloaded, click on the "send test message" button. You should receive a test email within a minute or so.
reef-pi can be configured to send alerts as text messages by configuring the To address with appropriate SMS gateway email address. For example, in the US, T-Mobile users can use <Phone number>@tmomail.net email address to send text messages using emails.

Concluding Remarks and Next Steps

We covered all the basic features for a reef-pi controller. This will serve as a foundation for the next set of guides that will incorporate ancillary electronics to control actual equipment and sensors.

reef-pi stores all its data in the file /var/lib/reef-pi/reef-pi.db. This file, along with the specific version of reef-pi which created the database file, can always recreate a reef-pi controller. Older reef-pi versions are available in github. Thus, users only have to make a backup of the database file periodically to backup their controller.