

# WEC Network Configuration Notes

---

## IP Table

| Item             | IP Address   | DNS Entry                   |
|------------------|--------------|-----------------------------|
| Cell Gateway     | 134.89.92.1  |                             |
| WebRelay         | 134.89.92.2  | buoy-relay.shore.mbari.org  |
| IFCB             | 134.89.92.7  | buoy-ifcb.shore.mbari.org   |
| RaspberryPi      | 134.89.92.15 | torque.shore.mbari.org      |
| Orin-Nano        | 134.89.92.16 | spec.shore.mbari.org        |
| Dock Web-Relay   | 134.89.92.30 | dock-relay.shore.mbari.org  |
| Dock Term-Server | 134.89.92.31 | dock-ts.shore.mbari.org     |
| Dock Dataq       | 134.89.92.32 | dock-dataq.shore.mbari.org  |
| Dock Camera      | 134.89.92.33 | dock-camera.shore.mbari.org |

## RaspberryPi Buoy Control Computer

You can ssh to the controller, if you are outside the institute, VPN in first.

- ssh using command `ssh buoy@torque.shore.mbari.org`,
- ftp using command `ftp buoy@torque.shore.mbari.org`

## Orin-Nano Buoy Computer

You can ssh to the computer, if you are outside the institute, VPN in first.

- ssh using command `ssh username@spec.shore.mbari.org`

## Wave-Energy Web Relay

With VPN access you can reach this page. Enter user/pass (admin:PowerBuoy) in order to login.

- LOGIN: <http://buoy-relay.shore.mbari.org>
- ADMIN: <http://buoy-relay.shore.mbari.org/setup.html>

## Sierra Wireless MP70 Cellular Modem

**WARNING:** Meddling with the modem during at-sea operations carries the risk of losing remote communication with the buoy if configured incorrectly. You should expect certificate warning from your web-browser when connected.

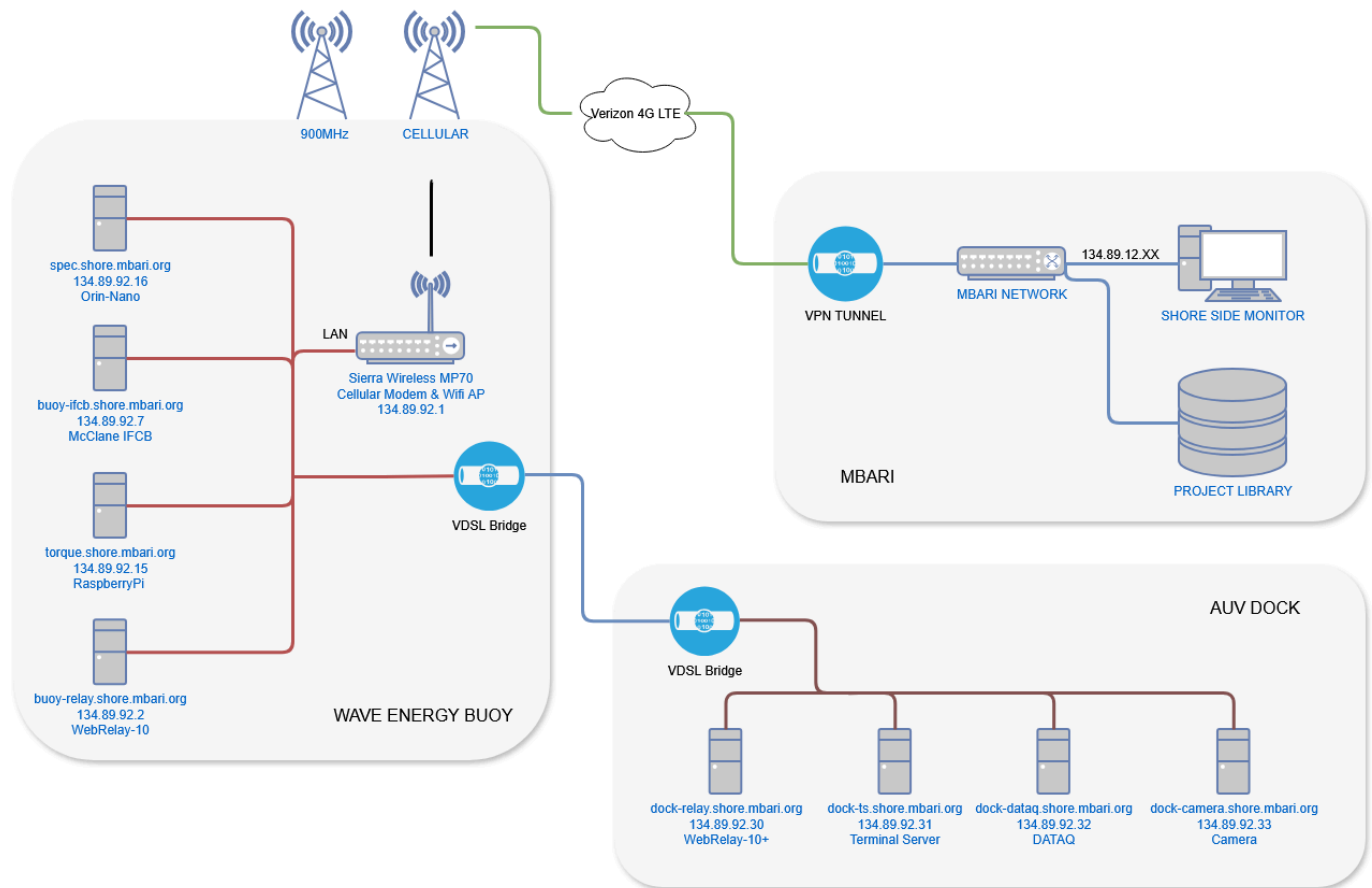
- LOGIN: <https://134.89.92.1:9443>
- USER/PASS: user:WaveEnergyFTW

## Dock Web Relay

With VPN access you can reach this page. Enter user/pass (admin:admin) in order to login.

- LOGIN: <http://dock-relay.shore.mbari.org>
- ADMIN: <http://dock-relay.shore.mbari.org/setup.html>

# Appendix A: Network Diagram



## Appendix B: Buoy Console Commands

---

When logging in you will see this message, and can be reprinted to the shell via `cat ~/helpful.txt`

Some helpful commands and tips:

|                           |                                                           |
|---------------------------|-----------------------------------------------------------|
| <code>pbcmd</code>        | - See a list of commands to the buoy controllers          |
| <code>llogs</code>        | - alias for 'tail -f /logs/latest/latest'                 |
| <code>doorwatch</code>    | - Monitor important trefoil controller data               |
| <code>battwatch</code>    | - Monitor important battery controller data               |
| <code>springwatch</code>  | - Monitor important spring controller data                |
| <code>powerwatch</code>   | - Monitor important power controller data                 |
| <code>ispbcon</code>      | - Check process list for pbcon, aka automator (one match) |
| <code>ispbcon</code>      | - Check process list for pbcon, aka automator (one match) |
| <code>isahrs</code>       | - Check process list for ahrs server (one match)          |
| <code>isdog</code>        | - Check process list for watchdog timers (2 matches)      |
| <code>pblogrestart</code> | - Use systemctl to restart pblog.service                  |
| <code>pblogstart</code>   | - Use systemctl to start pblog.service                    |
| <code>pblogstop</code>    | - Use systemctl to stop pblog.service                     |
| <code>pblogenable</code>  | - Use systemctl to enable pblog.service                   |
| <code>pblogdisable</code> | - Use systemctl to disable pblog.service                  |
| <code>ahrsrestart</code>  | - Use systemctl to restart ahrs_server.service            |
| <code>ahrsstart</code>    | - Use systemctl to start ahrs_server.service              |
| <code>ahrsstop</code>     | - Use systemctl to stop ahrs_server.service               |
| <code>ahrsenable</code>   | - Use systemctl to enable ahrs_server.service             |
| <code>ahrsdisable</code>  | - Use systemctl to disable ahrs_server.service            |
| <code>introhelp</code>    | - Show this list                                          |

These commands do not persist through reboots (pblog will be started when you reboot):

`pblogstop` - stop the service, Linux will not attempt to auto-restart

`pblogstart` - start the service after a stop, Linux will attempt auto-restarts if the service unexpectedly stops

`pblogrestart` - stop then restart the service with auto-restarts (use this when pblog is running, but not all output seems to be written to the log)

These commands persist through reboots:

`pblogdisable` - stop the service from running at boot, Linux will not attempt to auto-restart

`pblogenable` - have the service run at boot, Linux will attempt to auto-restart

## Change Log

- 2026: A few IP devices have been added to the network as follows: An Orin Nano on the buoy, and the AUV dock location (subsurface) a web-relay device, a terminal server for an acoustic modem, an IP-based camera, and a Dataq data-acquisition device. The Garmin radar on the buoy has been removed.
- 2022: The TS7970 has been replaced with a [RaspberryPi](#) for better compatibility with ROS2. There is a [Waveshare isolated CAN hat](#) attached to the RaspberryPi. The cellular modem and wifi router have been replaced with a Sierra Wireless MP70 and now uses Verizon's network. The radar has been changed from a Simrad to [Garmin 24xHD](#)
- 2020: We replaced the beaglebone with a TS-7970 from Technologic Systems, and have installed a simple wifi router for low-power download and diagnostics.
- 2019: We have more tools available for controlling processes on the beaglebone, see [Appendix B](#)
- 2018: We have implemented a VPN tunnel between the buoy and the MBARI network. This allows for an encrypted connection and flat network traversal of all components on the buoy. As a reset, ports are no long routed through the Moxa gateway.