Control Dashboard for XR Stage/VP Volume

Documentation

Miami University x 4wall

# Getting Started

- Make sure Node-RED is installed and running on a machine on the same network as the devices you are interacting with. (https://nodered.org/docs/getting-started/local)
    - After installation, to run, go to the directory where Node-RED is downloaded in CMD or PowerShell and type node-red (by default, the instructions are to download to the root Wins Sys 32, which requires running CMD or PowerShell as administrator to access Node-RED).

- Installed all the required Palletes/Plugins listed below:
    - node-red-contrib-calc
    - node-red-contrib-osc
    - node-red-contrib-slip
    - node-red-dashboard
    - node-red-disguise-http
    - node-red-node-ui-table
    - node-red-contrib-reusable-flows
    - node-red-contrib-fusion
    - node-red-contrib-telnet-client

- Import the json file provided, deploy, then go to the local port in a browser, followed by “/ui”
    - Default example: “http://127.0.0.1:1880/ui”

# Table of Contents

# Megapixel

Figure 1: Brightness and Gamma Paths

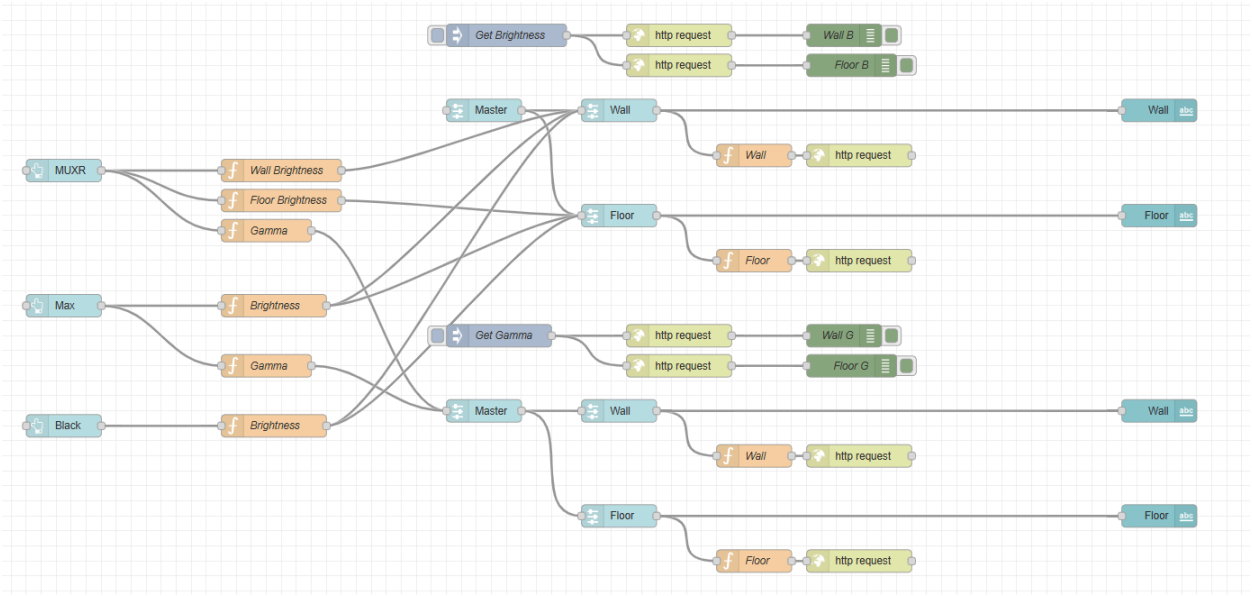
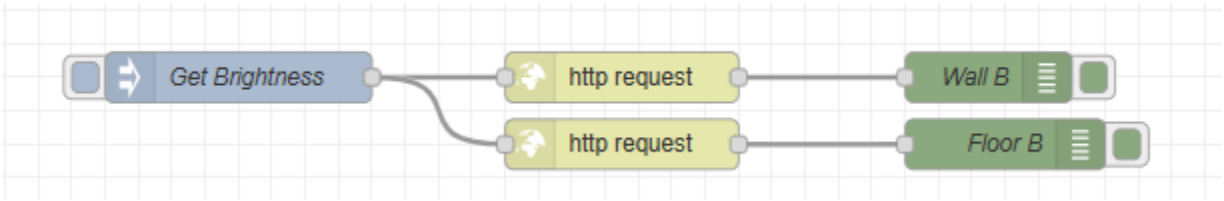


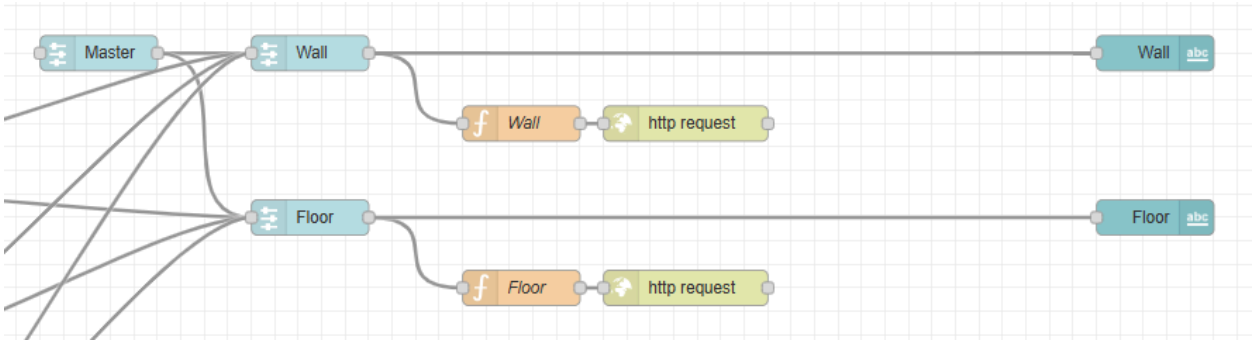

Figure 1.1: Current Brightness (Debug) Path




Current Brightness (Debug) Path

- Inject node for testing sends http request to megapixel
- Prints current brightness as debug message

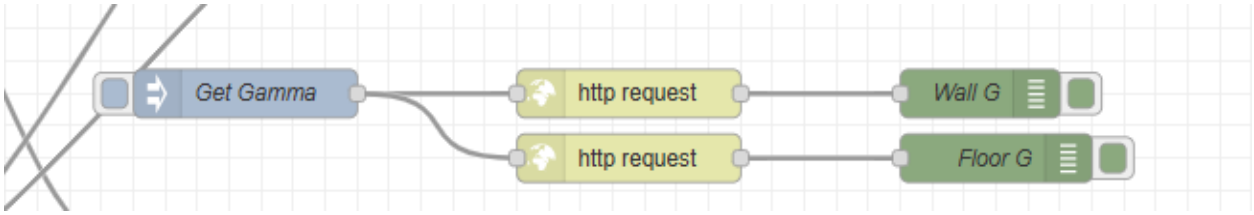
Figure 1.2: Update Brightness Path




Update Brightness Path

- A Master slider drives two separately controllable sliders corresponding to the Wall and the Floor
- The sliders' values are converted into http requests to set the brightness of the Wall and Floor in megapixel
- Text nodes are updated to display the current brightness values for the Wall and Floor

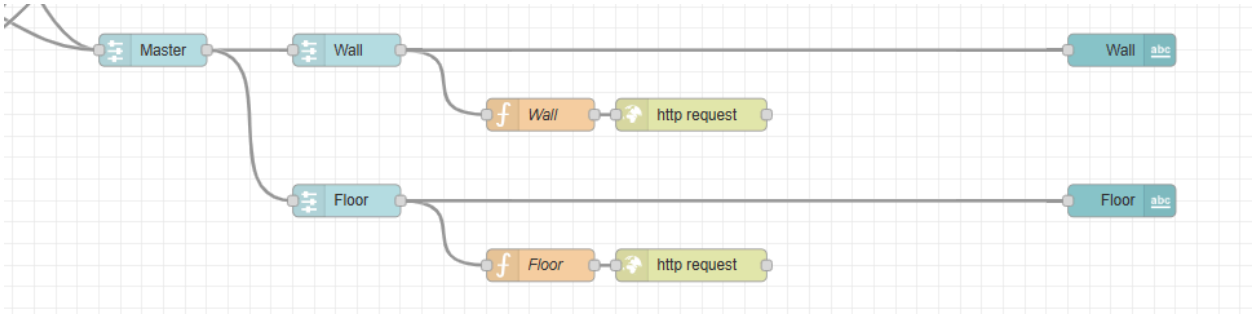
Figure 1.3: Current Gamma (Debug) Path




Current Gamma (Debug) Path

- Inject node for testing sends http request to megapixel
- Prints current gamma as debug message

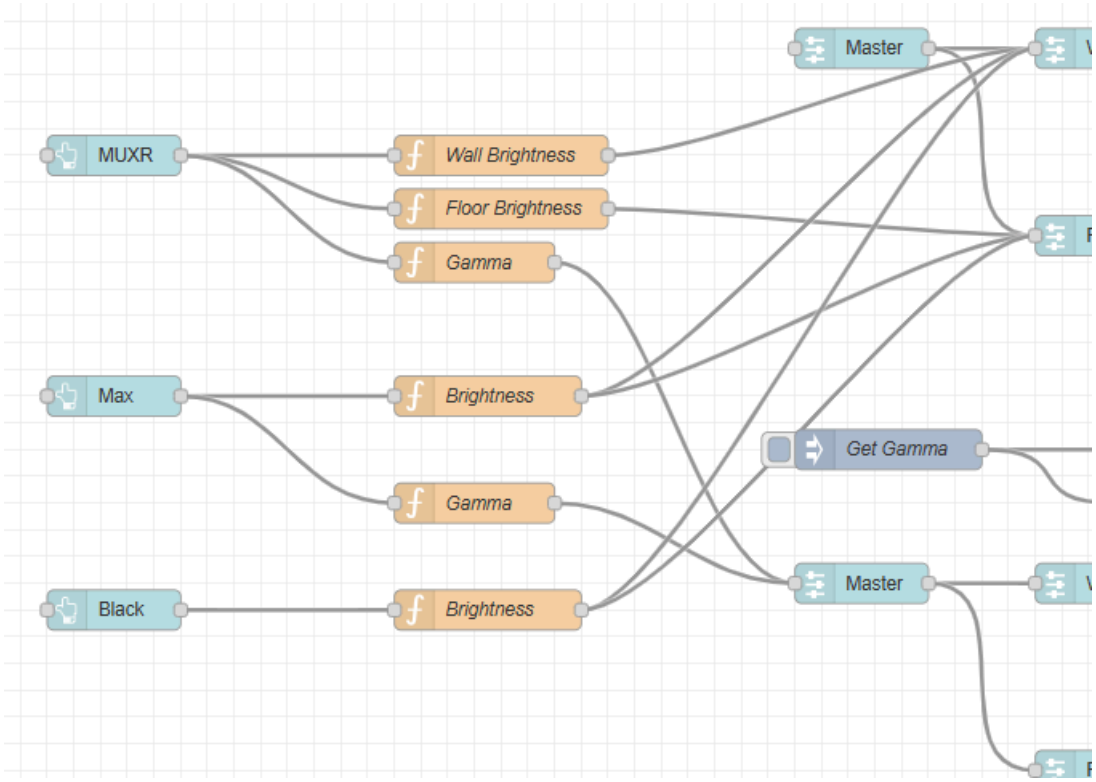
Figure 1.4: Update Gama Path




Update Gama Path

- A Master slider drives two separately controllable sliders corresponding to the Wall and the Floor
- The sliders' values are converted into http requests to set the brightness of the Wall and Floor in megapixel
- Text nodes are updated to display the current brightness values for the Wall and Floor

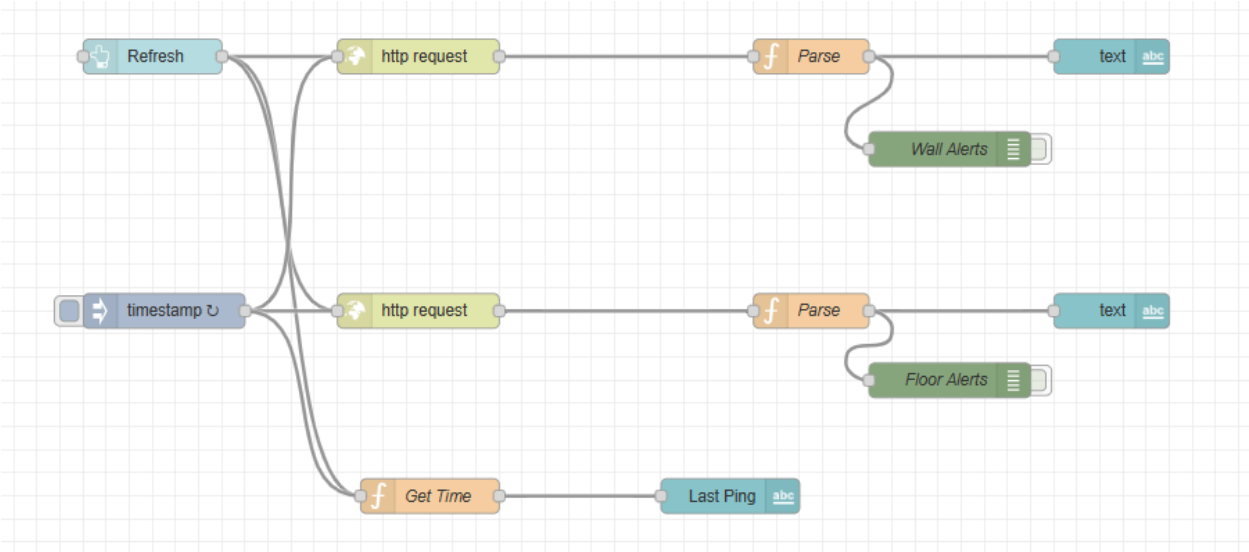
Figure 1.5: Presets Paths




Presets Paths

- Button elements trigger each preset
- These correspond to functions that set the value of the sliders in the Update Brightness Path and the Update Gamma Path
- It is simple to add in new presets by duplicating an existing preset and adjusting it's parameters
- The current presets are as follows:
    - MUXR
        - 70% wall brightness, 59% floor brightness, and 2.4 gamma
    - Max
        - 100% brightness and 4.0 gamma
    - Black
        - 0% brightness

Figure 2: Alerts Paths




Alerts Paths

- Can be manually triggered through the Refresh button or from the timestamp inject node, which automatically fires every minute
- Sends a http request to Megapixel, which returns a parsed JSON object
- This is parsed to highlight relevant data, which is printed both as a debug message and to UI as a text element

        - Prints out the number of alerts, each unique alert's brief, and how long ago each alert occurred
        - The text element includes a link to the alerts page of the screen's Helios online dashboard
- The Last Ping text element gets updated to show when the alerts were most recently refreshed

# Disguise

**Setup**

- Note the IP Address of your Disguise Director machine.
- Ensure Disguise's d3 Designer program is running on the Director machine.
- Ensure your current Disguise project is set up with an OSC Device and an OSC Transport, and that the OSC Transport is engaged.
- Ensure your OSC Recorders are set to "Live" mode, rather than "Play" or "Record" mode.
- On the Control Dashboard, input the Director machine's IP Address into the "Director IP" field, and the "receive port" of the OSC Device into the "OSC Port" field.
- Once these fields are filled, flick on the "Enable Disguise" switch. If it was already turned on but the controls were not running, flick it off and back on again. The Disguise dashboard elements will not update if this switch is not enabled.

**Usage**

Track

- Displays a dropdown menu of the Track on each currently running Transport.
- Selecting an option from this dropdown menu will populate the Notes, Cues, and Sections with the annotations from the corresponding track.
- The Dashboard will not automatically check if the Tracks have changed. Use the "Refresh" button if tracks need to be refreshed.

Current Project

- Displays the name and version number of the currently running Project.
- Project versions are denoted as (major version).(minor version).(hotfix version)
- The Dashboard will check to see if the Project has changed once every second.

Cues

- The dropdown menus labelled Notes, Cues, and Sections will display all of that type of Annotation in the currently selected Track. If nothing is showing up, make sure you have a Track selected.

- While a dropdown menu has an Annotation selected, pressing the corresponding “Jump To” button will move the playhead to the point of the selected Annotation.
- These jumps are accomplished via the API, not OSC, so disengaging the OSC transport will not prevent these jumps.
- The lists of Notes, Cues, and Sections are updated when the Disguise Enabled switch is flicked On, and whenever the Track is changed.

Health

- Displays the name of the Director machine, the number of Dropped and Missed Frames, and the Average FPS.
- Health is updated once every second.

Show Control

- Each button in the Show Control section sends the corresponding command over OSC.

Renderstream

- The Renderstream dropdown menu displays a list of all the Renderstream layers on the current Track.
- While the dropdown menu has a layer selected, the layer’s name, workload name, the RX node or nodes it’s running on, and any asset errors will be displayed. This information is updated once every second. Only one asset error is displayed at a time.
- While a layer is selected, its workload can be started and stopped with the corresponding buttons. (TODO: I should see if I can display the stopping/starting/running status of the layer too)
- The list of Renderstream layers is updated when the Disguise Enabled switch is flicked On. It can be manually refreshed using the Refresh button.

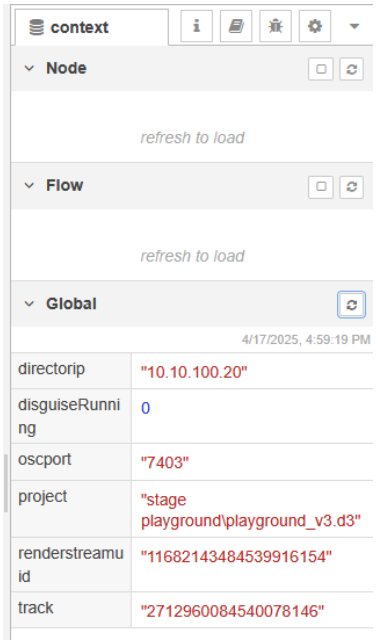
# Scene Control

- The Scene Control section is set up to control the Miami University ESPN Super Bowl set, by default. However, other Unreal Engine scenes can be set up to use these controls, or the dashboard can be modified to accommodate them.

- To set up Scene Control, input the IP Address of the RX Node that the RenderStream project is running on, as well as the port number that the Disguise OSC Server is set up to receive on (for the ESPN set, this number is **4567**). Click “Enable Scene Control” to show the controls themselves.
- The three emissive texture colors correspond to the lights on the left side, right side, and top of the scene. They can be controlled by RGB Sliders and by Brightness sliders, and input is sent over when the user modifies any of these values.
    - To use these controls with another Unreal Engine scene, set up its OSC server to look for the addresses “Mesh1R”, “Mesh1B”, “Mesh1G”, and “Mesh1L”, as well as corresponding addresses for Mesh2 and Mesh 3. 1 is Left, 2 is Right, 3 is Center.

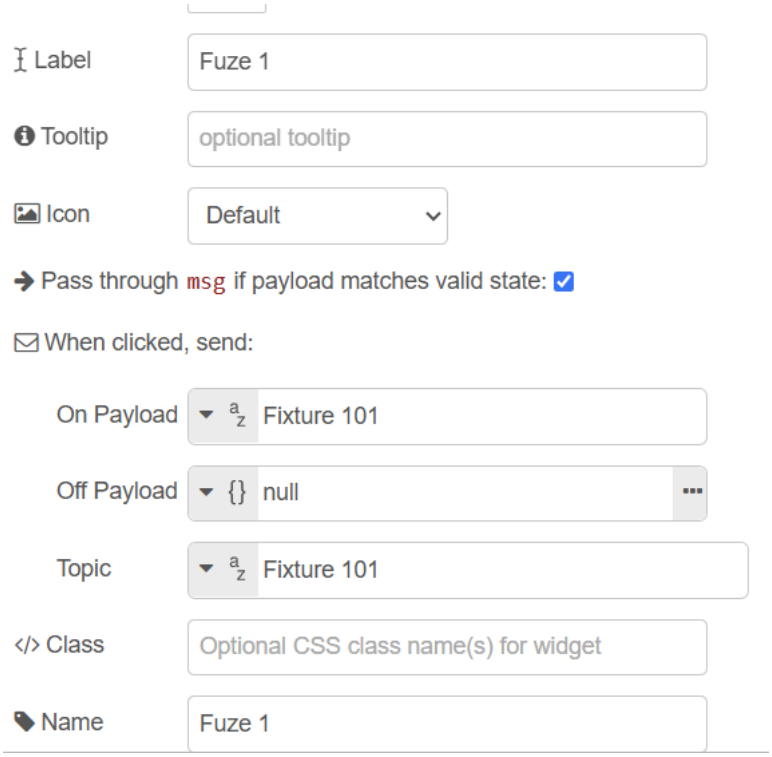
**Troubleshooting**

- All code for the Disguise Dashboard elements can be found in the “Casey - Disguise” flow. Each part of the dashboard is labelled with a comment that corresponds to its label.
- The Dashboard stores many of its elements in Global Variables. These can be seen in the Context Data tab at any time. Click the Reload button to refresh these values.

- 
context
Node
refresh to load
Flow
refresh to load
Global
4/17/2025, 4:59:19 PM
directorip
"10.10.100.20"
disguiseRunning
0
oscport
"7403"
project
"stage playground\playground_v3.d3"
renderstreamuid
"11682143484539916154"
track
"2712960084540078146"

# GrandMa3

A few things to remember about setting up this file for your fixture setup: Each Fixture in your stage/setup should be added as a labeled fixture within Node Red as a UI switch. This makes it so your fixture can be selected, and can be used within the UI. Within each fixture UI node, you will need to add the fixture or group you want to toggle. An example of a fixture is shown below.

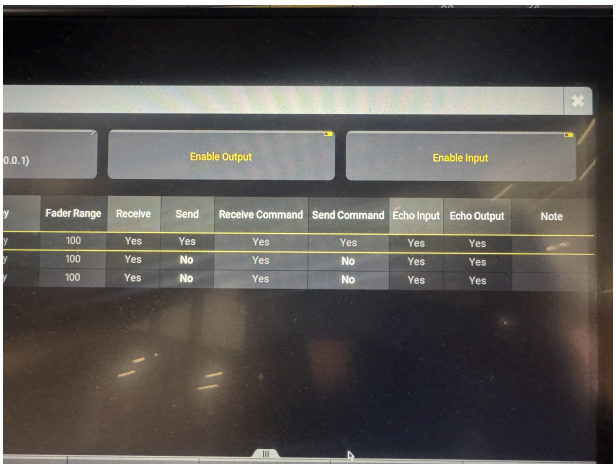



To select a fixture, add the word “Fixture”, followed by the FID. You can also refer to a group of fixtures instead by inserting “Group 1”, or something similar in the Topic and Payload instead.

In terms of controllable attributes, this Node Red setup is based off of the MUXR MA3 fixtures, which are elation fuzes and KL Panels. Therefore, the controllable parameters match these fixture types. If you need to add different parameters, you’ll need to look into the MA3 documentation for specific OSC command lines needed for that type of command. Information on how to set up an OSC port is also in this documentation provided by grandMA’s website.

Another useful part of the MA3 controls is the Sequence selector column. Once you enter the number of a sequence you want to call in your showfile, you can hit the Go+ or Go- buttons to go forward and backwards in the sequence selected. Something to keep in mind with this section of the MA3 controls is this sequence selector only functions when the fader on the console is faded up. This is more of a safety precaution, so make sure a node-red operator is working in tandem with the lighting designer.

To receive OSC from Node Red, the UDP out node needs to be set to the correct port number and IP address. Assuming the MA console is set to loopback (1.0.0.127) in the In & Out tab, has echo, receive input and receive set to yes, and is on the same network as the Node Red machine, the IP address should be 10.10.100.249. Enable input and enable output should also both be checked on in the console (when they are clicked on, the text turns yellow).


You will also need to set the receiving address in the MA console itself– for this setup, the MA console was set to receive from any device on the network by using the address 127.255.255.255. This allows for any device that comes into the space to have the capability of using the Node Red MA3 dashboard.

Another note to keep in mind when using this dashboard- node red is not set up to understand a lighting console's color wheel. Therefore, there are two separate sliders for color temperature and center (center is only applicable to the fuzes). If a color is not updating as expected, remember to test the color temperature and center sliders as well. Further, the gobo wheel sliders are meant to emulate the encoders for gobo control on the console. Therefore, scrolling through the slider will scroll through the

gobo wheels installed within the fixtures. Again, while this is only applicable to the elation fuzes, this may be applicable if other types of movers are used in a different fixture setup.

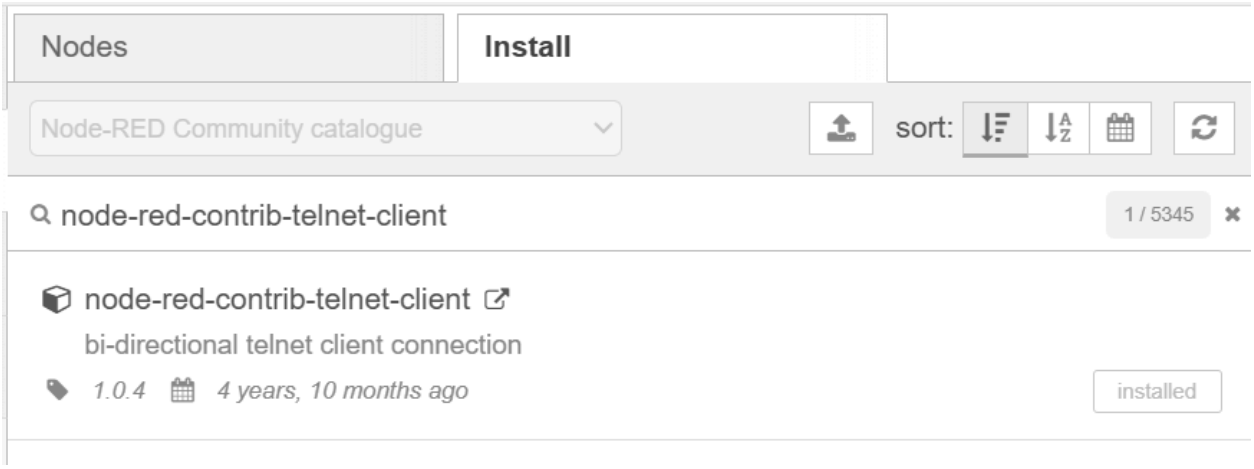
# Hyperdeck

### Hardware Check

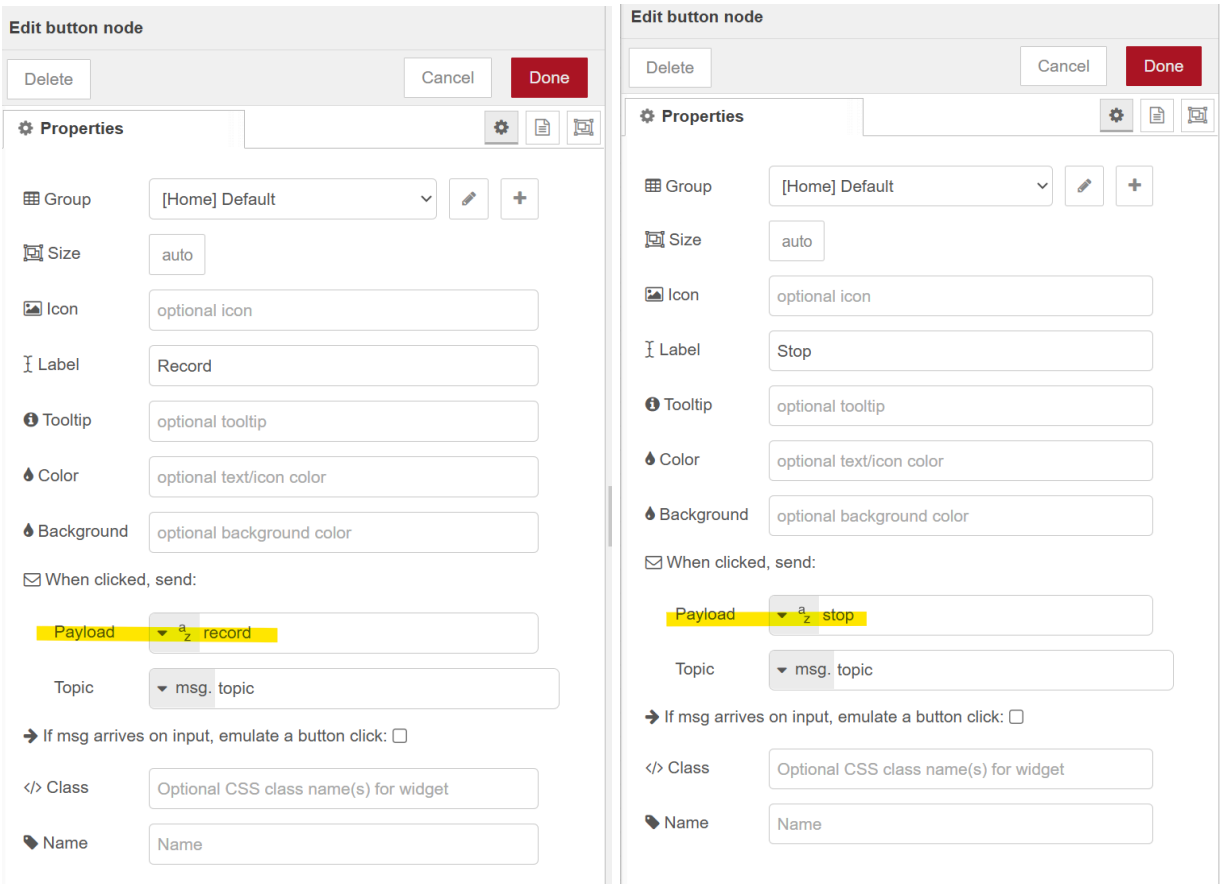
Verify that your BlackMagic Hyperdeck is connected using an ethernet cable. From the Hyperdeck manual: “The Blackmagic HyperDeck Ethernet Protocol is a text based protocol accessed by connecting to TCP port 9993 on HyperDeck models that have a built in Ethernet connection.”

### Node Red Setup

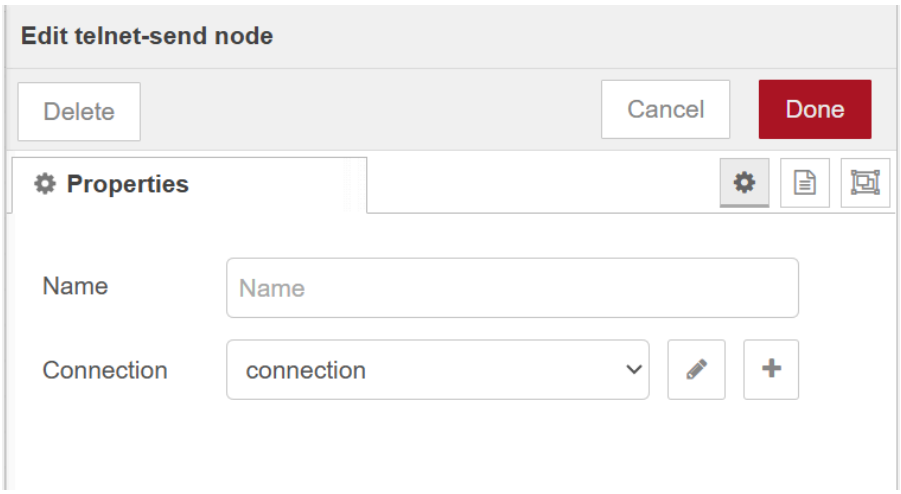
We are utilizing the palette made by *cabrouwers* to use Telnet communication with the Hyperdecks: [https://flows.nodered.org/node/node-red-contrib-telnet-client](https://flows.nodered.org/node/node-red-contrib-telnet-client)



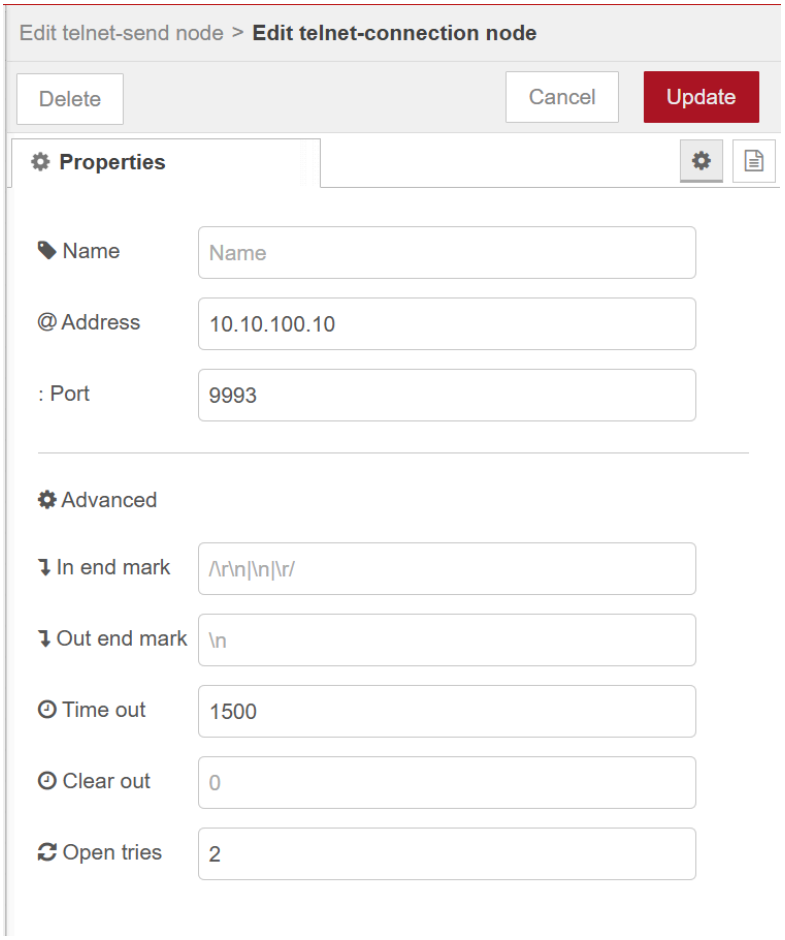

We are sending 2 different Ethernet Protocol commands from Node Red through Telnet. These commands are set in the payload of the UI buttons, “record” to start a recording, and “stop” to stop a recording.




You will need to create a new Connection within the Telnet Send node.

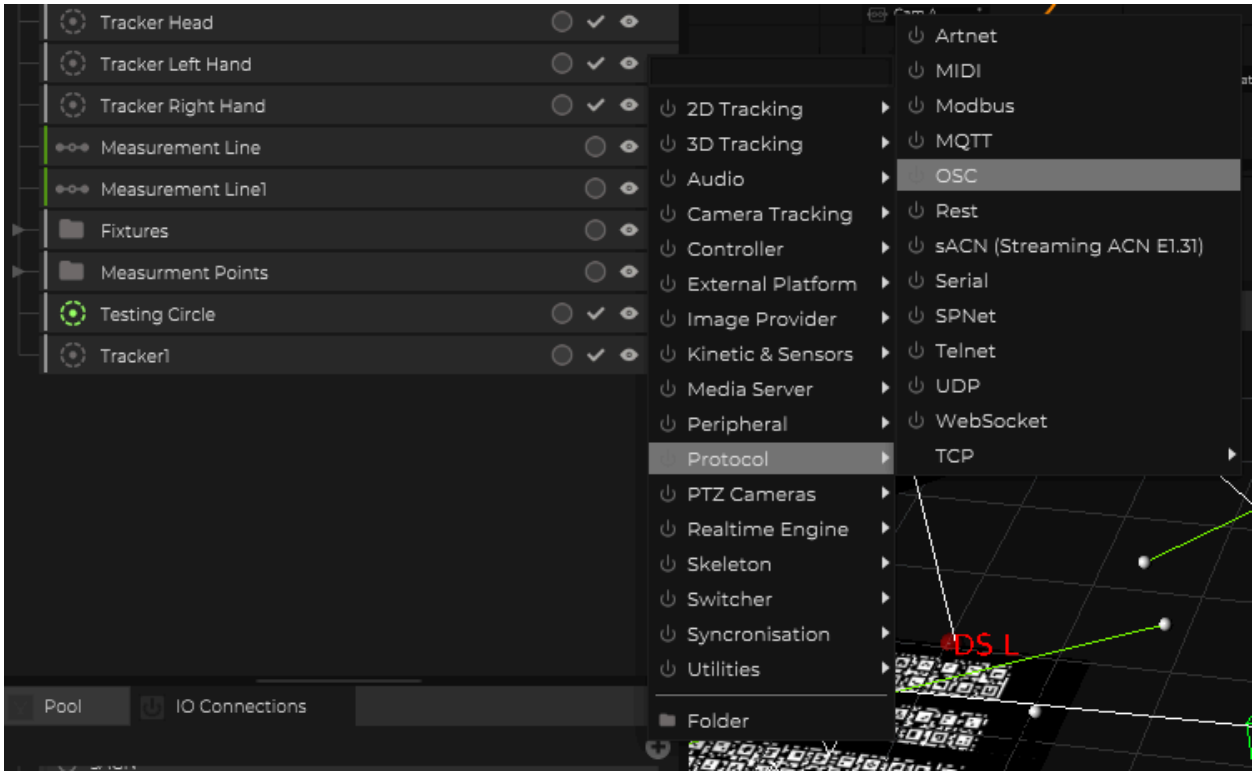


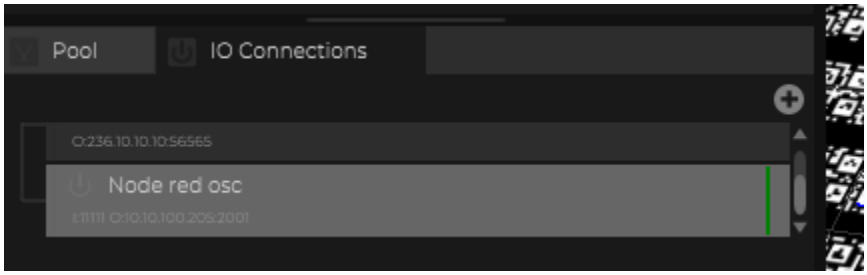

Within that Connection, you may set the IP of your Hyperdeck and you **must** have the port set to 9993 to work on any Hyperdeck machine.

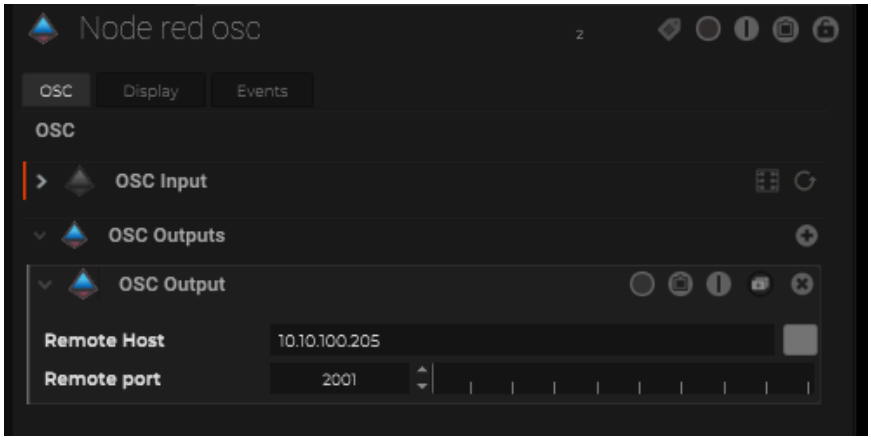

Edit telnet-send node > Edit telnet-connection node
Delete
Cancel
Update
Properties
Name
Name
Address
10.10.100.10
: Port
9993
Advanced
In end mark
/\r\n|\n|\r/
Out end mark
\n
Time out
1500
Clear out
0
Open tries
2

# Setting up Stage Precision

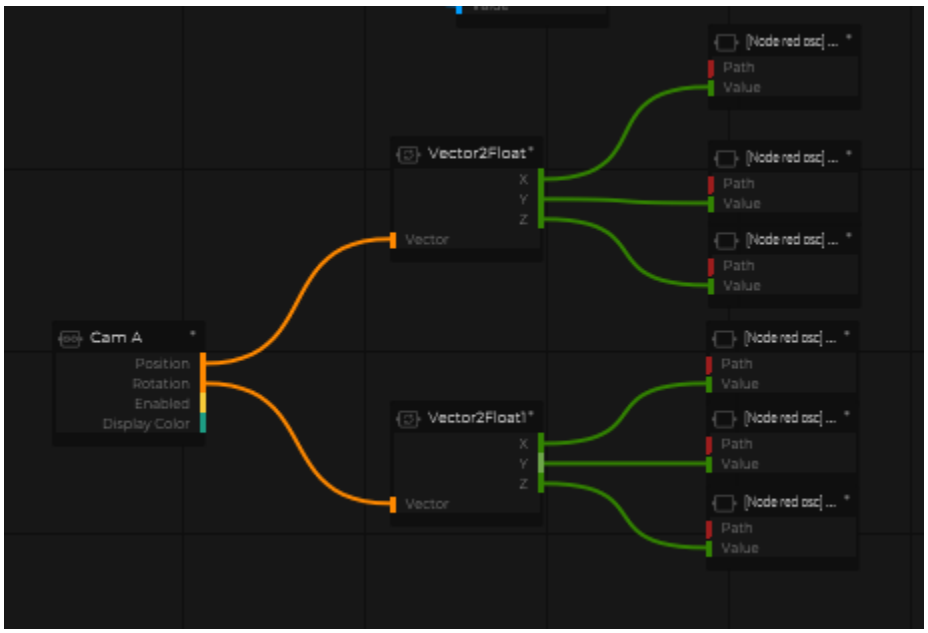
- Create an OSC IO Connection
    - Make your Remote Host the computer that is running Node-Red, or broadcast to your network.
    - Take note of the port number you choose, it should not be in use in any other OSC connections in order to work properly.



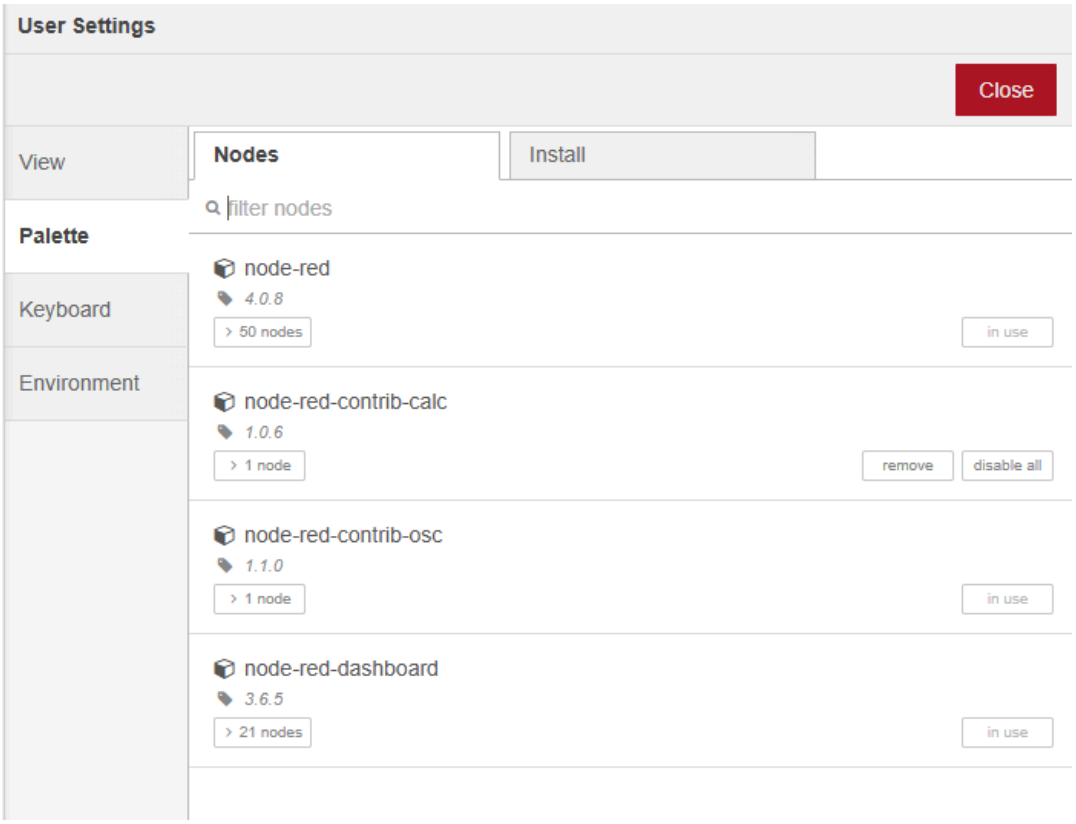


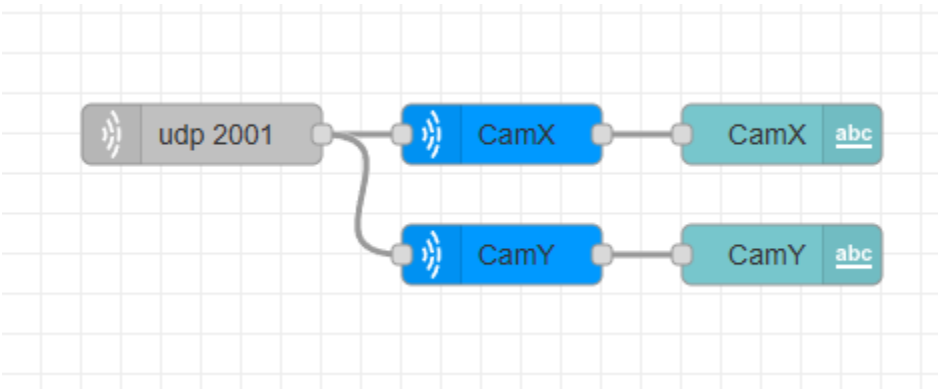

- Drag your camera to the board and take it’s position and rotation into OSC outs




Setting up Node-Red

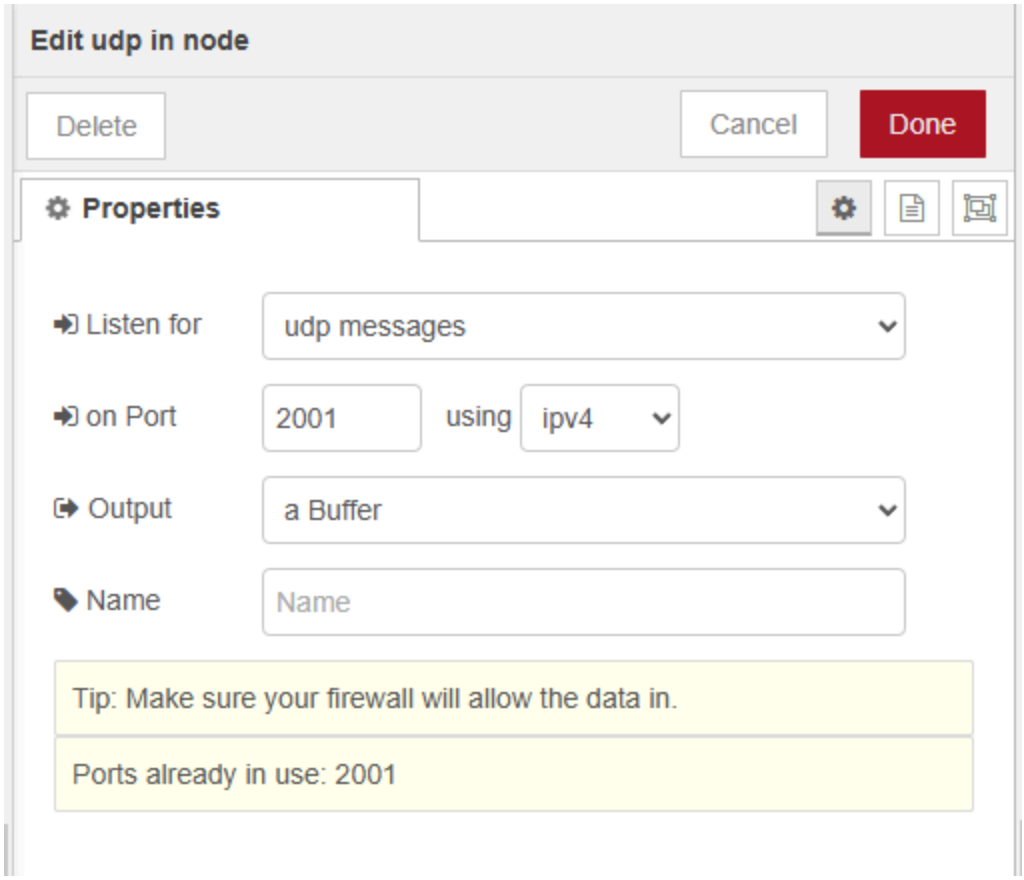
**Palettes to have:**



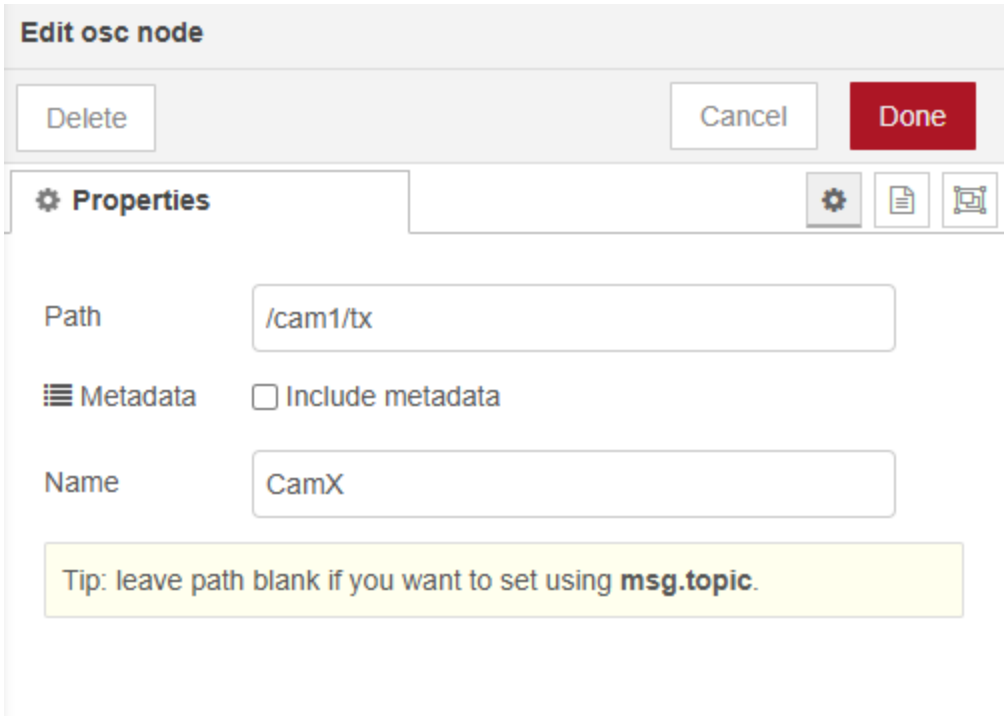

UDP Node:

- Set the port to listen to




OSC Node:

- Set the path to listen to


Edit osc node
Delete
Cancel
Done
Properties
Path
/cam1/tx
Metadata
Include metadata
Name
CamX
Tip: leave path blank if you want to set using msg.topic.