TECH LIBRARY

VR2 OEM IP Control

Operating an IP controlled relay board.

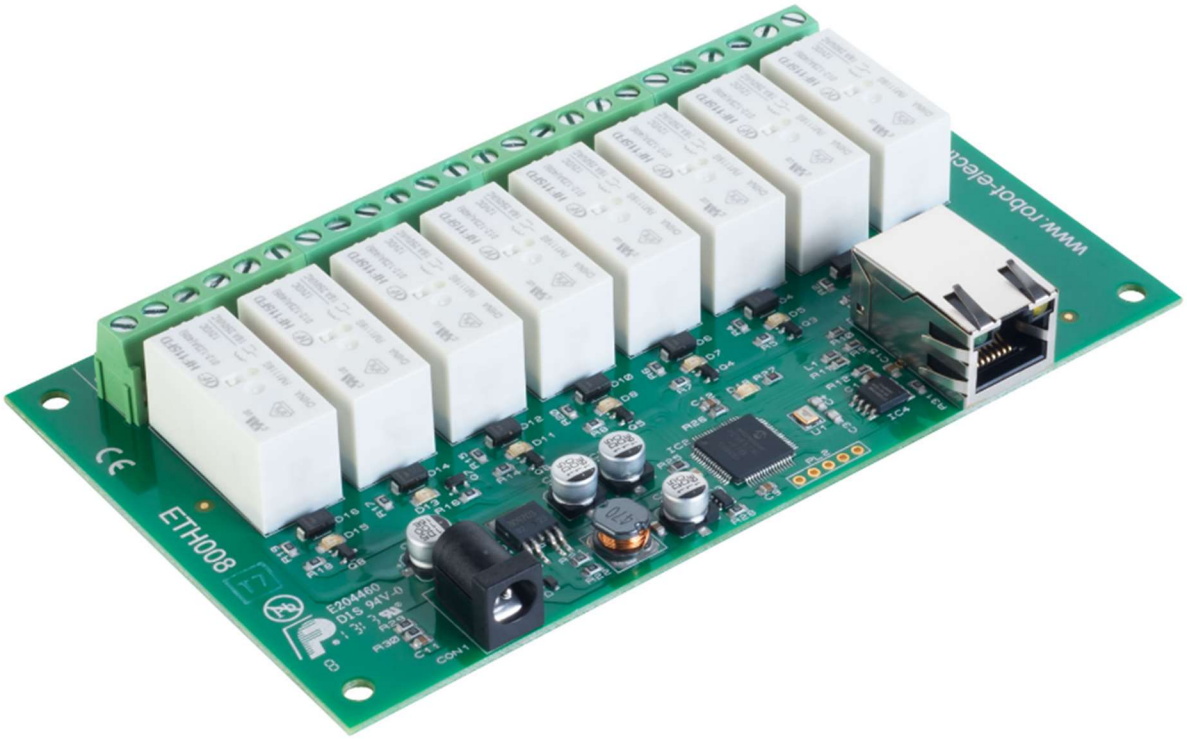
Introduction

This document assumes the user is familiar with our control application, D-Net.

The VR2 PoE Touch Remote can be used to transmit IP based control strings to operate OEM equipment as well as all NST Audio products.

As well as direct control of AV equipment such as projectors and displays, indirect control of peripheral equipment such as motorised projection screens, room dividers/screens and curtains/drapes is possible through the use of a simple relay control board which has IP control.

We will explore the control of a typical control board and how to set this up on a VR2.



IP Control Implementation on a VR2 – Start Simple

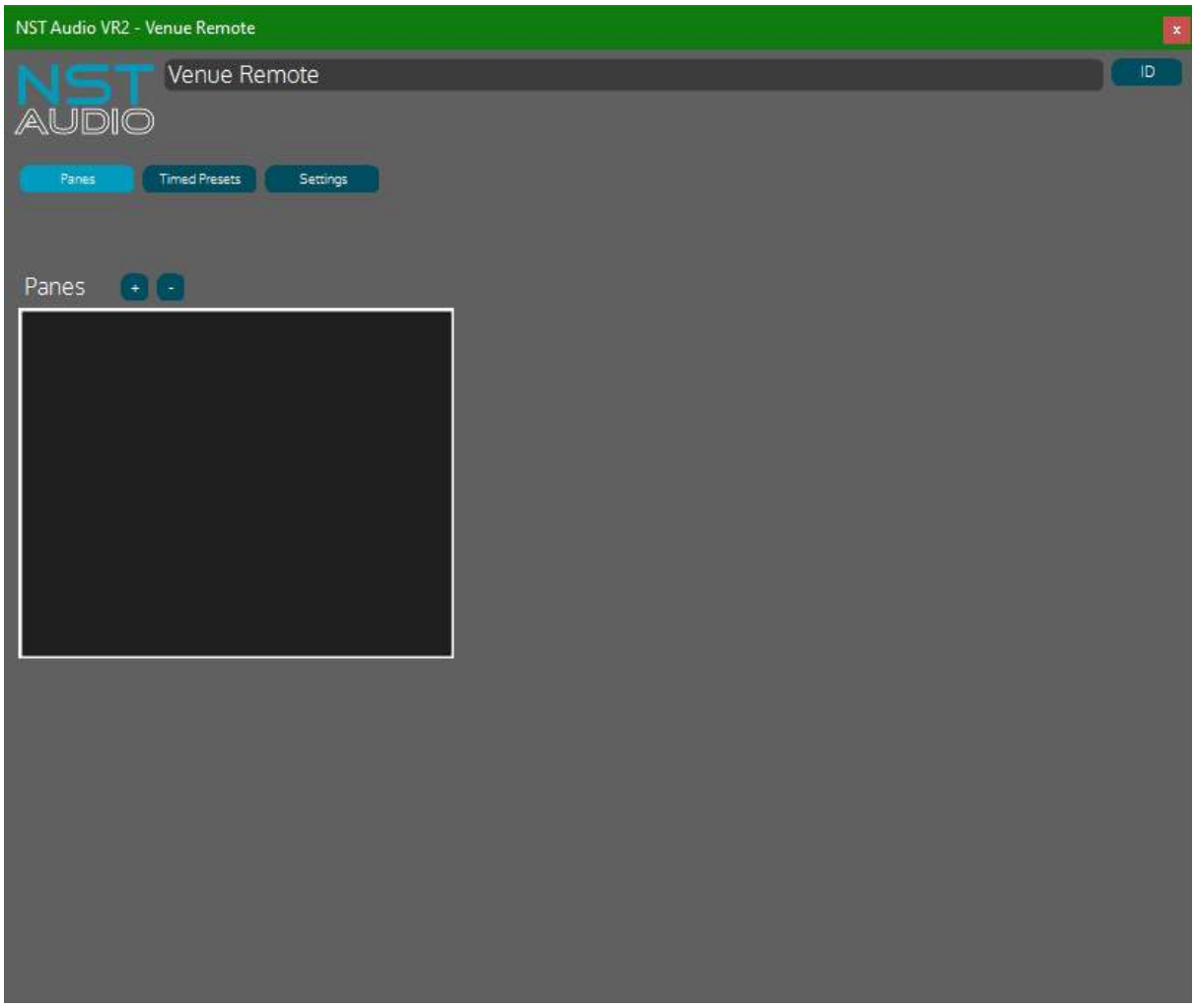
To establish the method of setting up an IP Control Pane, let's start with a simple command for an NST Audio product.

We will create a pane that will send a "Global Mute" command to a device. It doesn't matter what NST product this is, as they all respond to a "Global Mute" command, and it's possible to see this working on any product (all Mute LEDs on D48, all Clip LEDS on other products), so it's the ideal starting point.

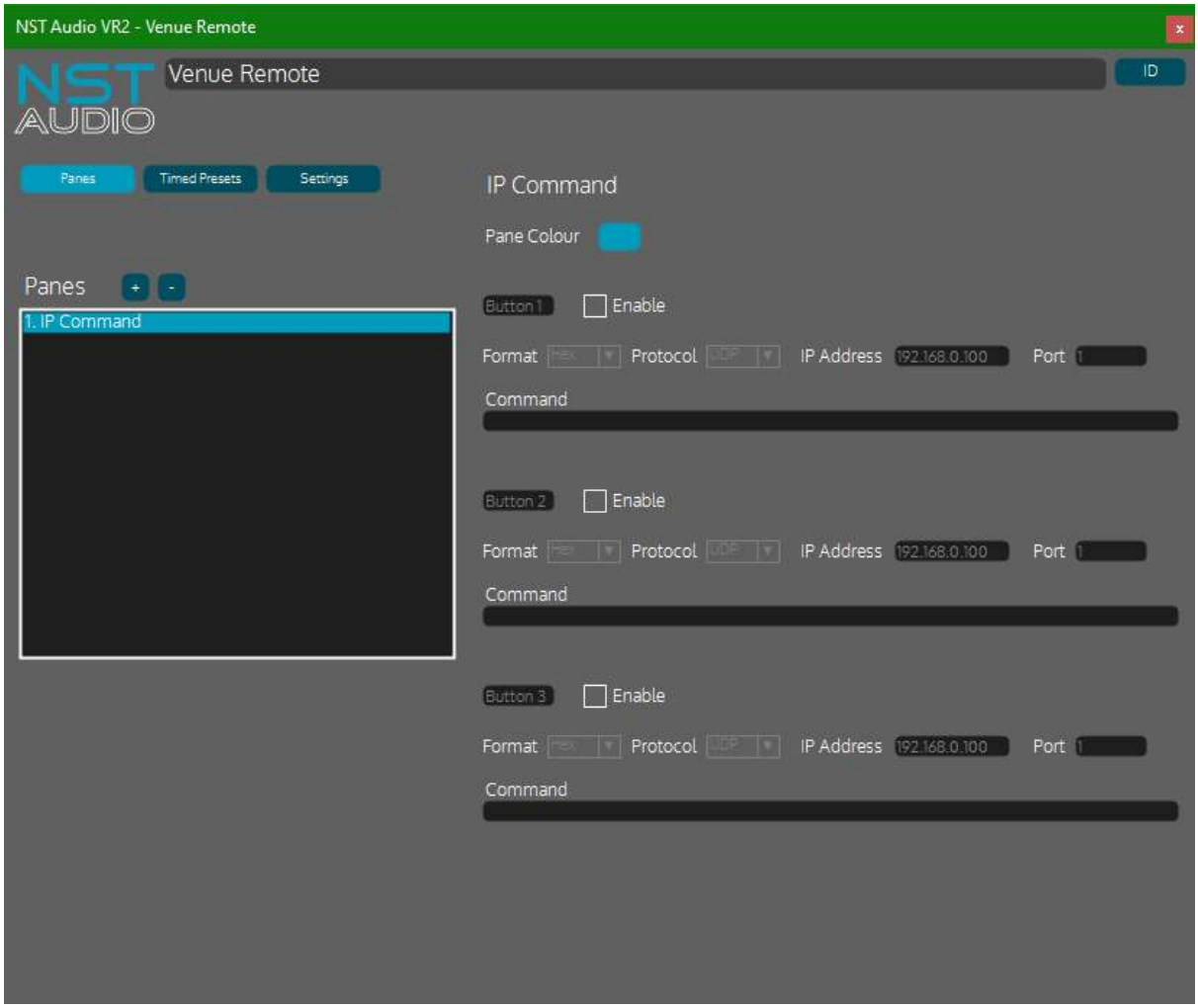
If you are familiar with the process of configuring an IP pane, you may jump straight to page 7 for the relay board control example.

All NST Products also support a simple remote protocol layer, and we will use this to test out the system. Documentation about this is available upon request.

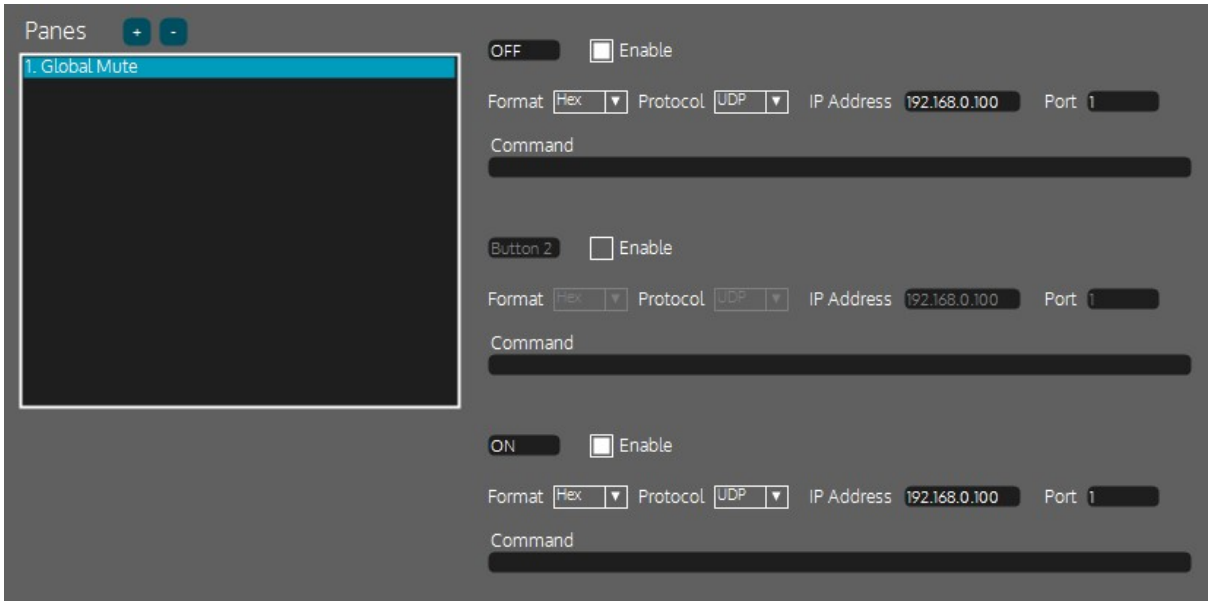
Setting up a control pane for IP control follows the same procedure as for other operations.
Open D-Net and go online, to discover the VR2 on the network.



Click on the "+" button above and choose "Add IP Command Pane"



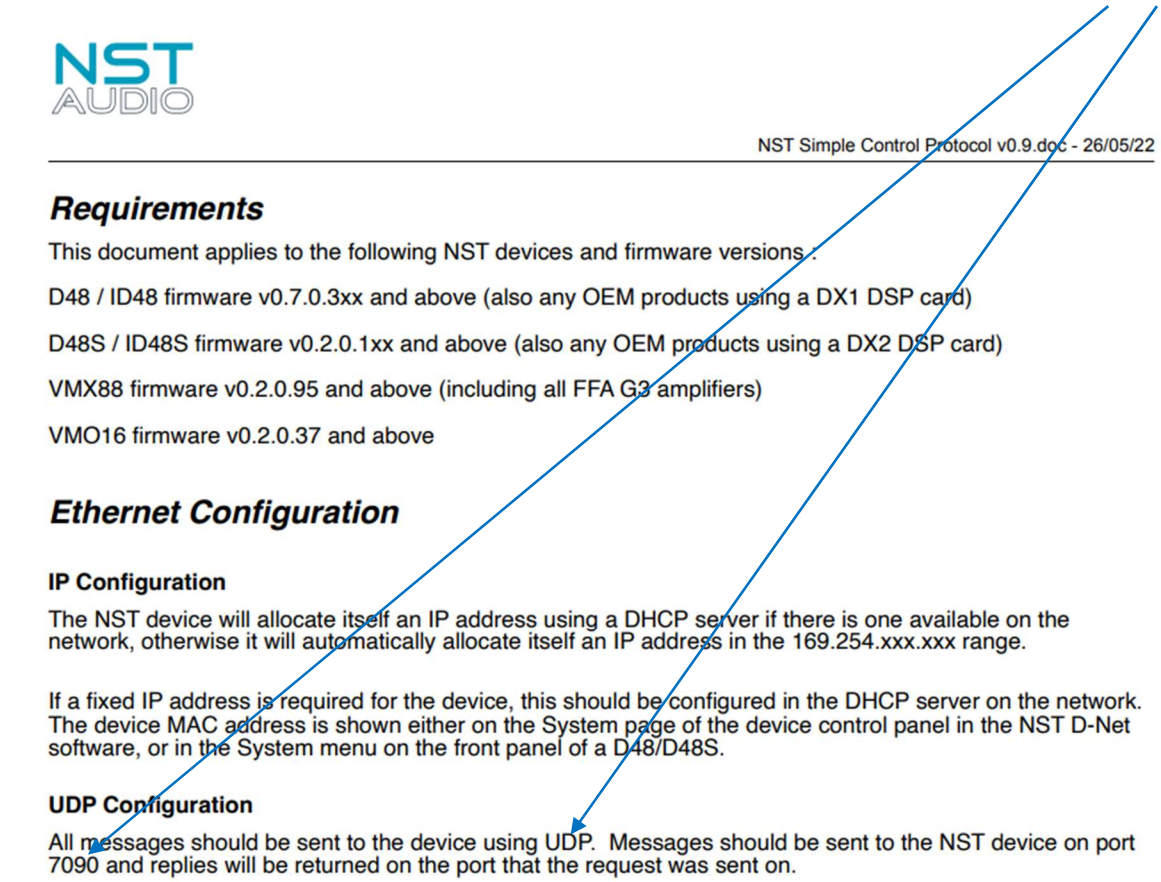
Enable two buttons on this pane, label them “OFF” and “ON”, and name the pane “Global Mute”.



The pane and the labels will update in real time on the VR2.

At this point we are ready to set up the control aspects of the pane. We need to know the protocol (UDP or TCP), the transmit port and of course the IP address of the device to be controlled.

Consulting the "Simple Remote Protocol" specification for NST Products we can answer the first two of these:



NST AUDIO

NST Simple Control Protocol v0.9.doc - 26/05/22

Requirements

This document applies to the following NST devices and firmware versions :

D48 / ID48 firmware v0.7.0.3xx and above (also any OEM products using a DX1 DSP card)

D48S / ID48S firmware v0.2.0.1xx and above (also any OEM products using a DX2 DSP card)

VMX88 firmware v0.2.0.95 and above (including all FFA G3 amplifiers)

VMO16 firmware v0.2.0.37 and above

Ethernet Configuration

IP Configuration

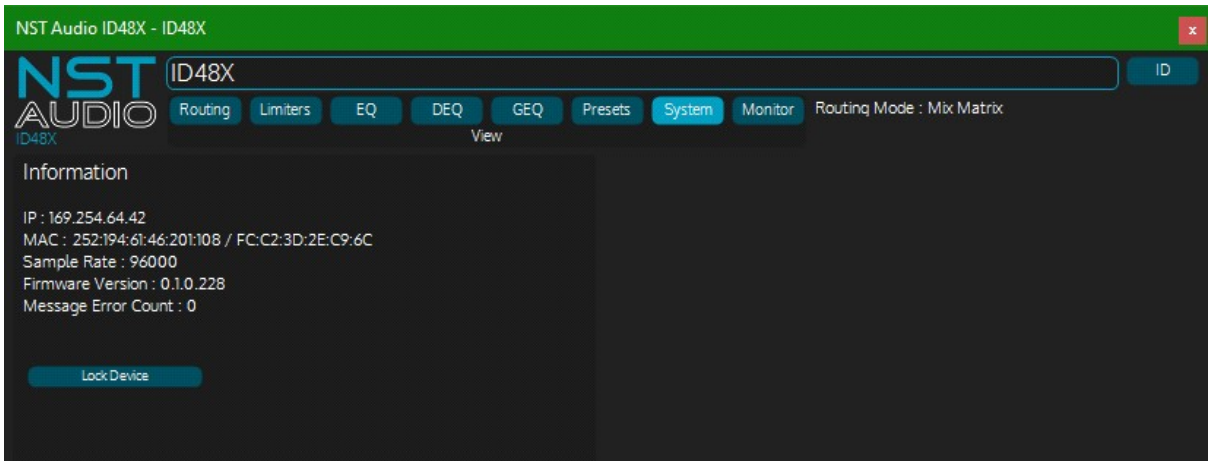
The NST device will allocate itself an IP address using a DHCP server if there is one available on the network, otherwise it will automatically allocate itself an IP address in the 169.254.xxx.xxx range.

If a fixed IP address is required for the device, this should be configured in the DHCP server on the network. The device MAC address is shown either on the System page of the device control panel in the NST D-Net software, or in the System menu on the front panel of a D48/D48S.

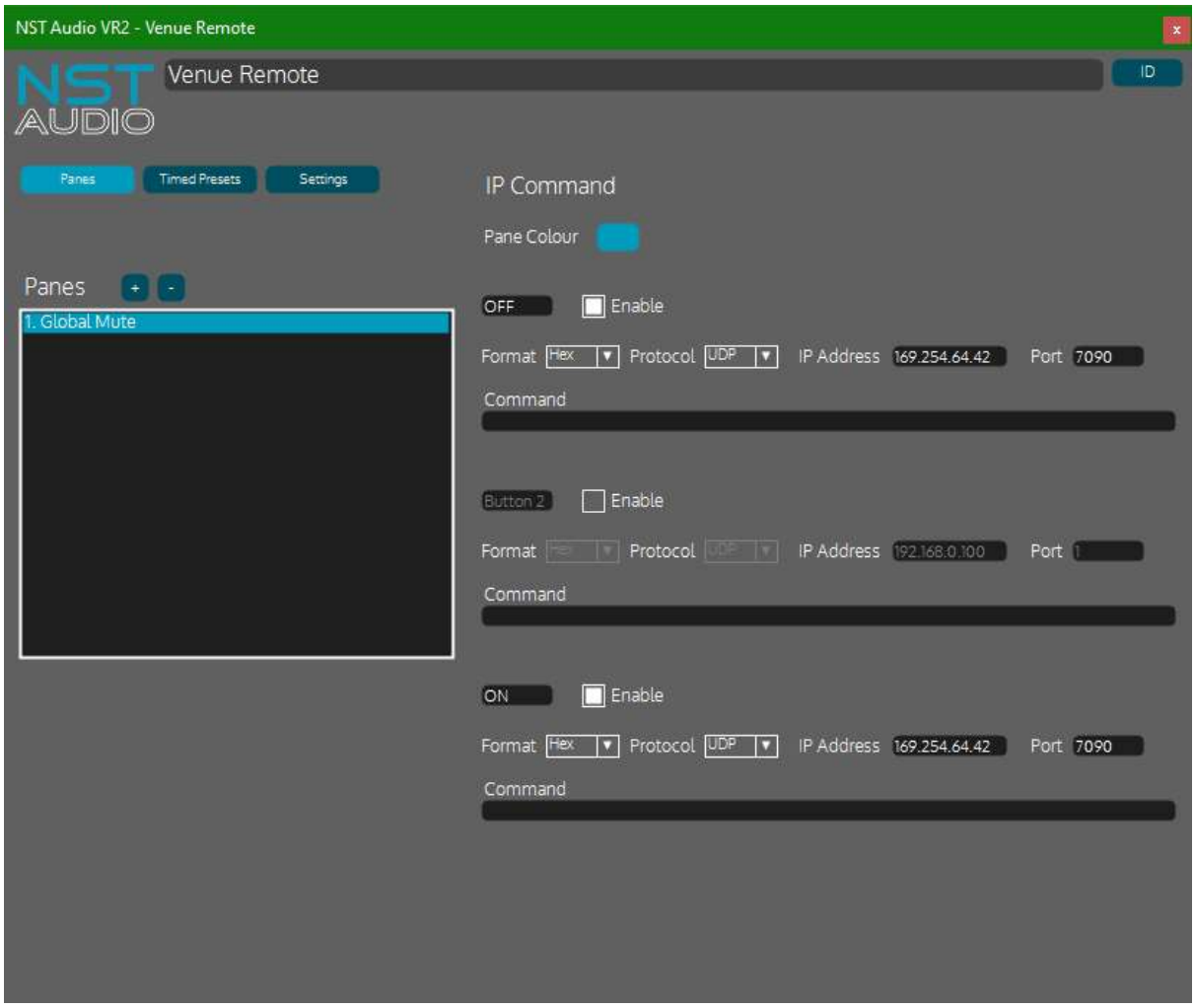
UDP Configuration

All messages should be sent to the device using UDP. Messages should be sent to the NST device on port 7090 and replies will be returned on the port that the request was sent on.

The IP address of the unit is found in D-Net on the unit's "System" page. We will address an ID48X in our example.



So, our destination IP address is 169.254.64.42. Note that this is a self-generated IP address in the absence of a DHCP server. For reliable operation we recommend using reserved IP addresses configured through a router's management console for control of NST equipment, or a static IP address accessed through the device itself. This second method will be used later in this document when we set up the projector control.



Finally, we add the command strings to be sent on pressing each of the buttons.
Referring to the protocol document again and looking up the commands for Global Mute we find:

NST Simple Control Protocol v0.9.doc - 26/05/22

1006 – Global Mute

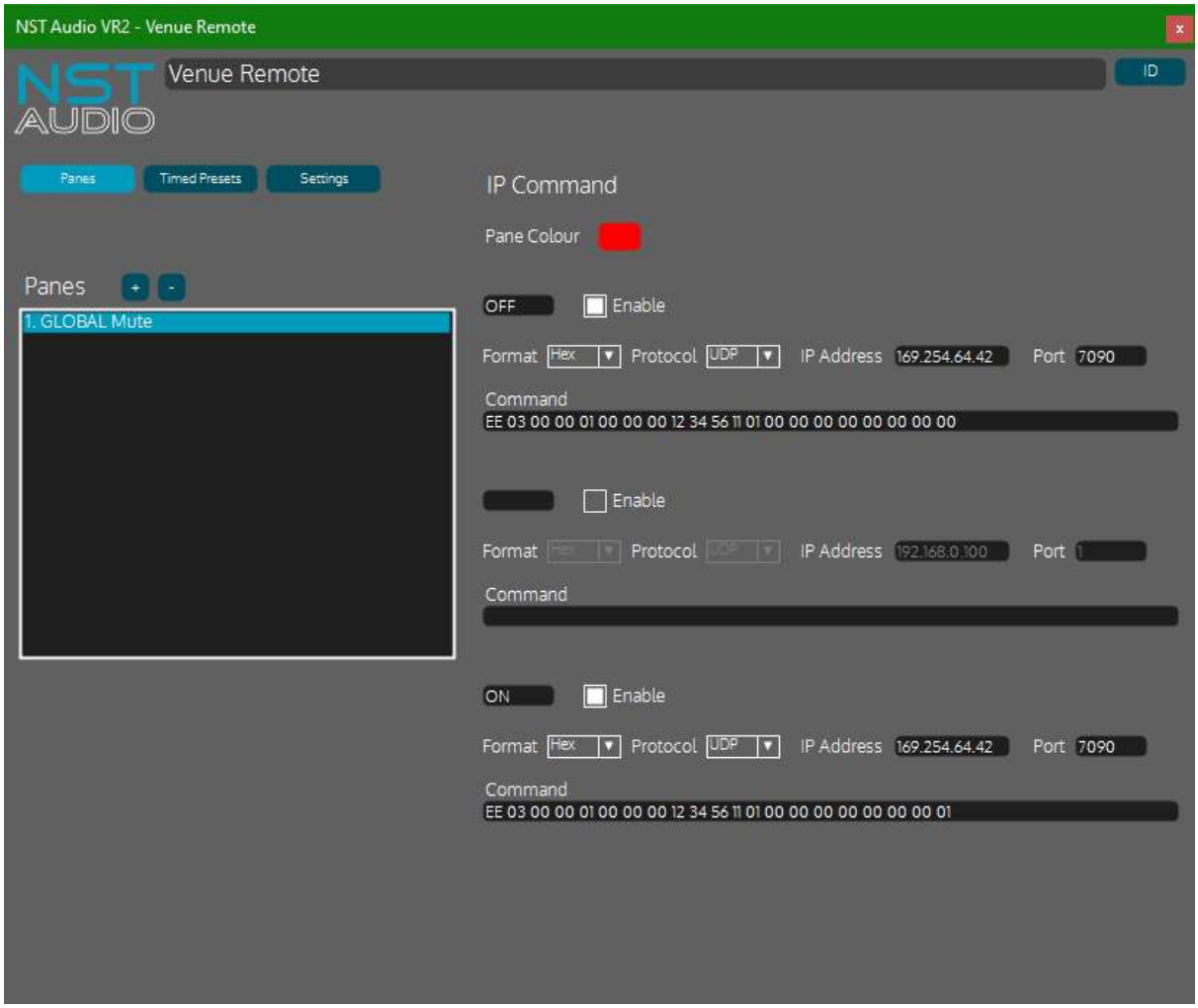
Command Format :
char - Mute status (0x00 = OFF, 0x01 = ON)

Reply Format :
Header only, no data. MessageDirection value indicates success or failure.

Example Hex Bytes:

EE	03	00	00	01	00	00	00	12	34	56	11	01	00	00	00	00	00	00	00	01	--	--	--
MessageType				MessageSize				MessageCounter				Dir	Reserved							Global Mute State			
Message Header																				Message Data			

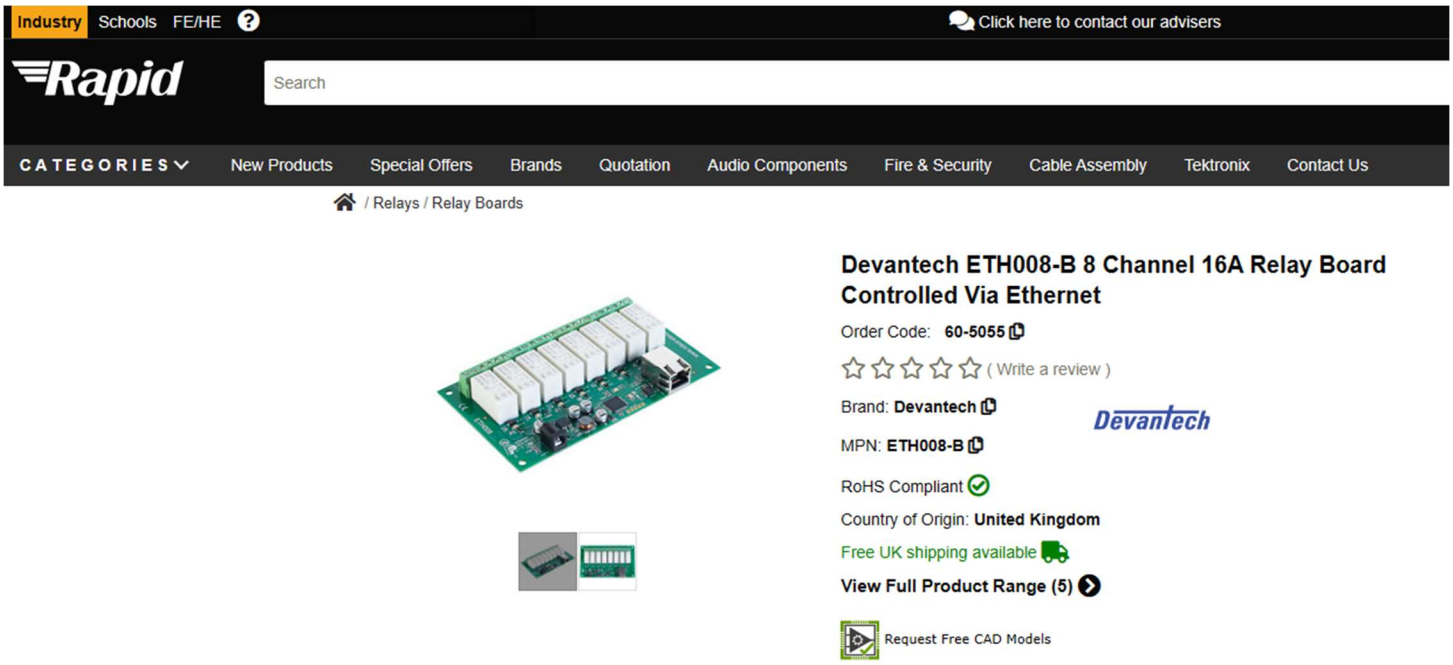
Adding the string (in hex) to each button, we only change the final byte in the command to set Global Mute On (01) and Global Mute Off (00):



And that's it – pressing the buttons on the VR2 will now mute/unmute all the inputs and outputs on the ID48X.

Relay Board Control

The relay board we are using in this example is a Devantech ETH008-B. There are other variations of this board available including a smaller two channel version.

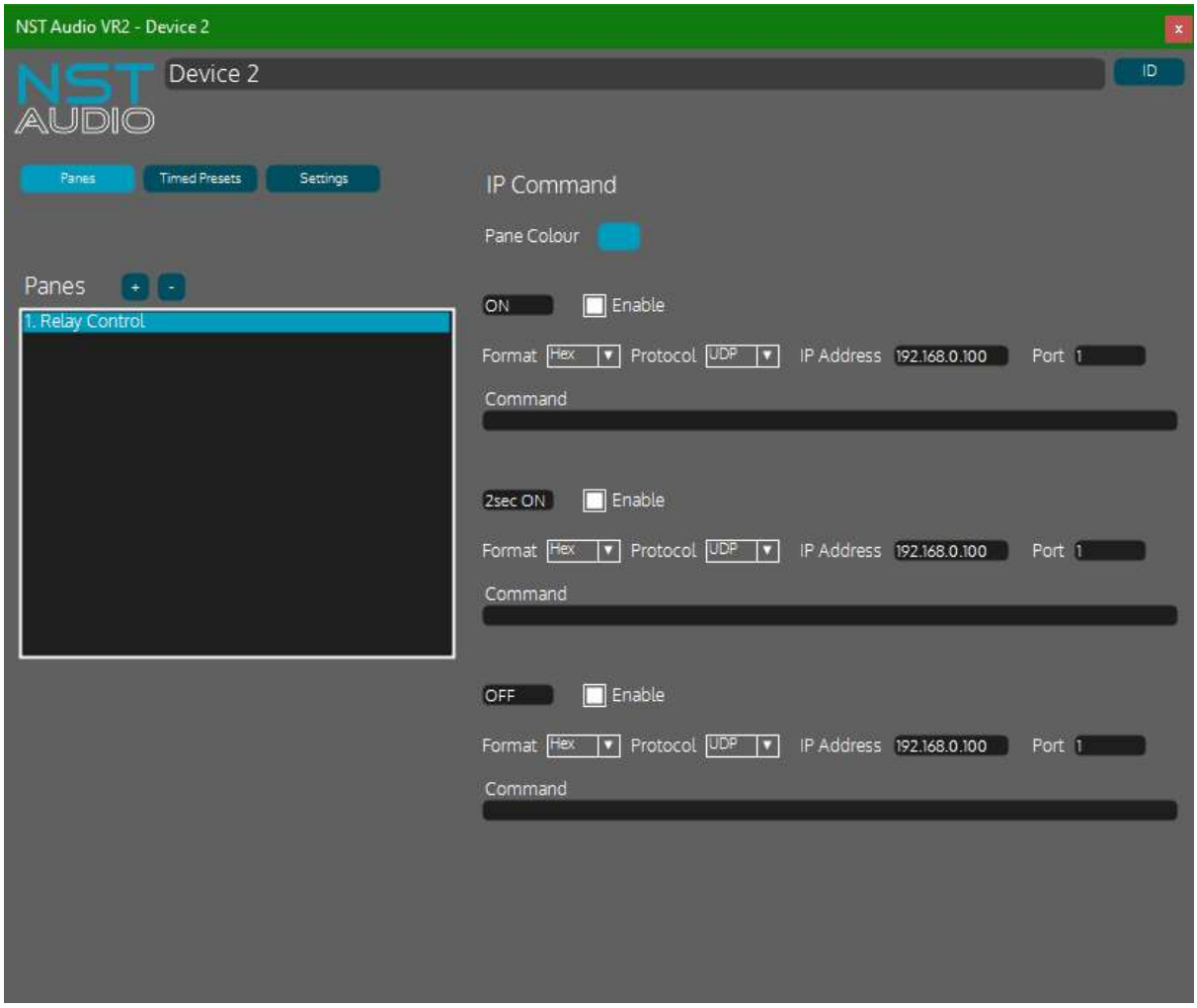


Relay Operation

The relays on the board can be set to latch on/off or stay on for a programmed period (up to 25,5 seconds) and then revert to off. This allows for operation of, for example, mains powered equipment such as lighting or fans, and also for momentary activation in applications such as motorised projection screens or GPI activation of other equipment.

Initial Set-up

Let's configure a relay on the board with three buttons to turn relay #1 off, on and a timed 2 second on>off action. We will start by setting up a pane on the VR2 with the three buttons we require, labelled as appropriate.



Example Configuration

Consulting the documentation for the board, we can find out what the default IP address of the board is, in the absence of a DHCP server.

Network connection and Http access

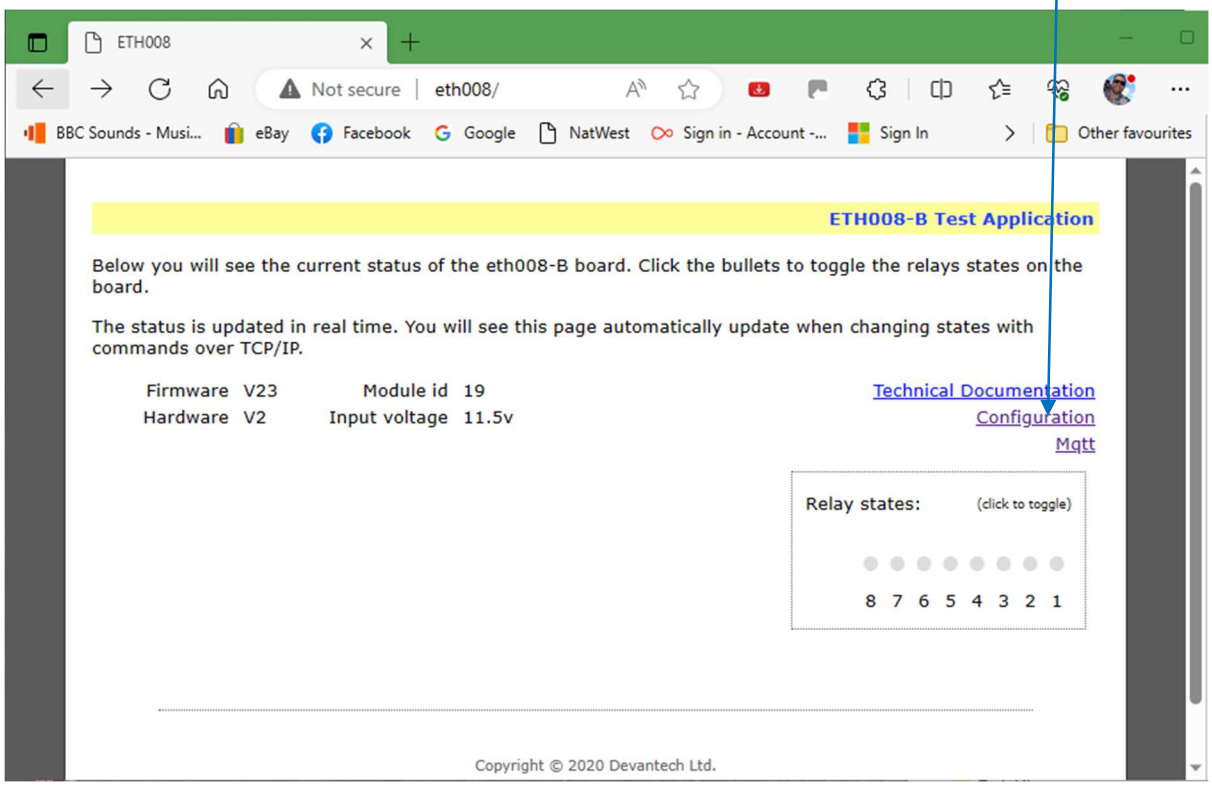
IP Addresses & DHCP Servers

The easiest way to use the ETH008-B is to connect it to a network with a DHCP server. In this case the ETH008-B will have its IP address assigned automatically by the DHCP server.

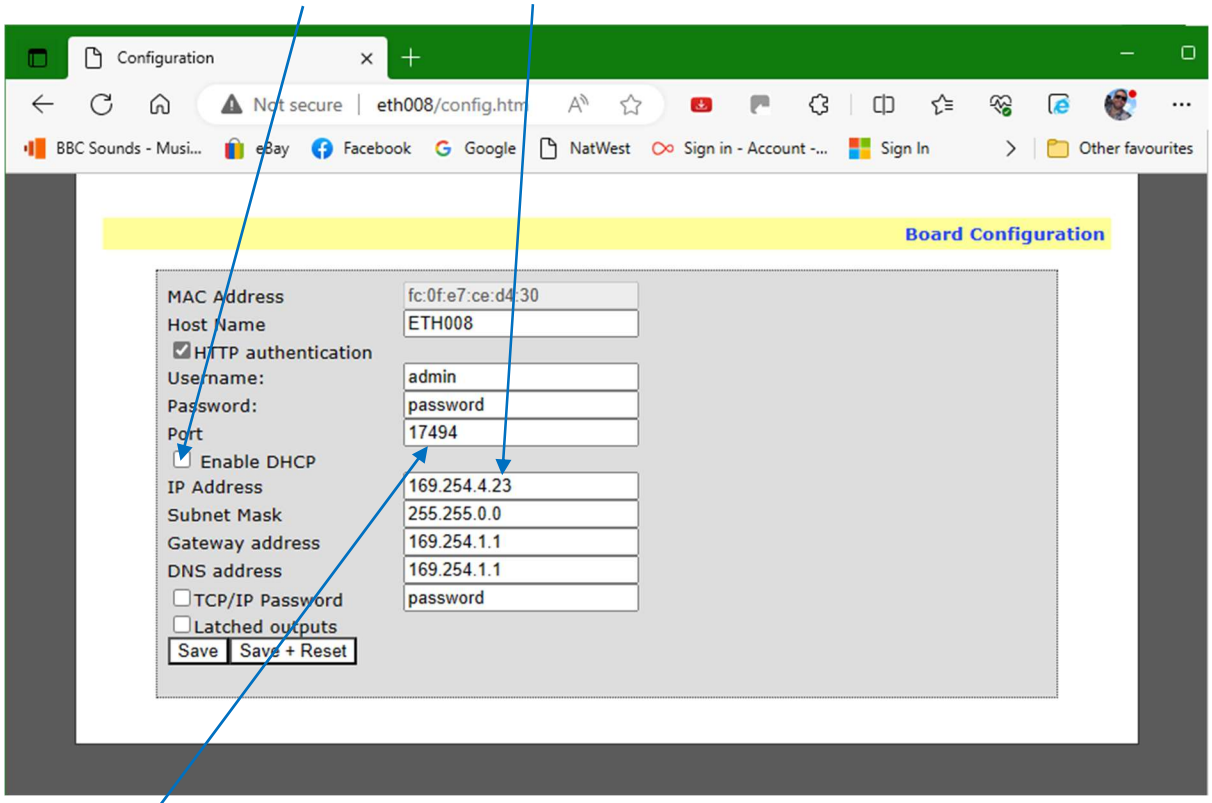
If there is no DHCP server on the network, then a fixed IP address of 192.168.0.200 is used. To control the ETH008-B using this fixed IP address your computer MUST be on the same subnet. The next step is to set your computers IP address to 192.168.0.x where x is in the range of 1 to 255 but not 200 (the ETH008-B is there!) or any other used IP addresses on the network. The subnet mask dictates what IP addresses the PC can communicate with, we set this to 255.255.255.0 so the PC can talk to any module with an IP address of 192.168.0.x

We need to fix the IP address of the board so that it always powers up with the same IP address and port number if the VR2 is going to address it reliably.

This begins with accessing the board webserver through a browser and via the "Configuration" link, disabling DHCP and setting a fixed IP address:

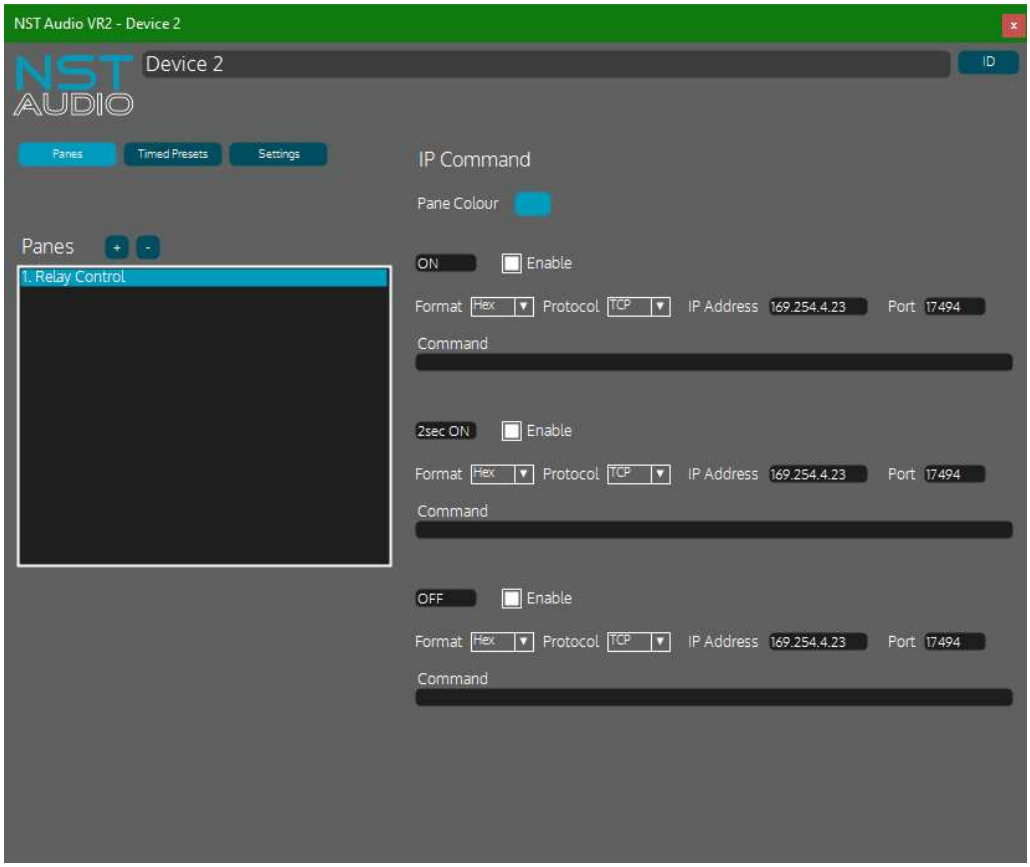


DHCP has now been disabled and the IP address set as below:



The port number, 17494 is also shown here.

We will add these details to the VR2 configuration – remember to do this for all three buttons!



The final step is to create the correct command strings to perform the actions we require, and add these strings to each button's action. The documentation summarises the avaialble actions below:

TCP access and commands

The command set designed to provide consistent expansion and new features, they are sent over TCP/IP on port 17494 (0x4456). This is the default port, it can be changed in the configuration settings. Five connections are allowed at any one time, these are independently protected but all using the same password as defined in the board configuration.

Command		Action
dec	hex	
16	0x10	Get Module Info - returns 3 bytes. Module Id (19 for ETH008), Hardware version, Firmware version.
32	0x20	Digital Active - follow with 1-8 to set relay on, then a time for pulsed output from 1-255 (100ms resolution) or 0 for permanent. Board will return 0 for success, 1 for failure
33	0x21	Digital Inactive - follow with 1-8 to turn relay off, then a time for pulsed output from 1-255 (100ms resolution) or 0 for permanent. Board will return 0 for success, 1 for failure
35	0x23	Digital Set Outputs - will set all relays states, All on = 255 (11111111) All off = 0 Board will return 0 for success, 1 for failure
36	0x24	Digital Get Outputs - returns 1 bytes, corresponding with relays being powered
58	0x3A	ASCII text commands - allows a text string to switch outputs, see section below
119	0x77	Get Serial Number - Returns the unique 6 byte MAC address of the module.
120	0x78	Get Volts - returns relay supply voltage as byte, 125 being 12.5V DC
121	0x79	Password Entry - see TCP/IP password, board will return 1 for success or 2 for failure
122	0x7A	Get Unlock Time - see section below
123	0x7B	Log Out - immediately re-enables TCP/IP password protection, board will return 0 for success

In particular, we require the "Digital Active" and "Digital Inactive" commands, which are explained in more detail:

Digital Active/Inactive Commands

These are 3 byte commands:
The first byte is the command, 32 (active means on) or 33 (inactive).
Second byte is the output number, 1-8 for the relays.
Third byte is the on time. Set this to zero for non pulsed mode, or 1-255 for a pulse in 100mS intervals (100mS to 25.5 seconds).
For example:

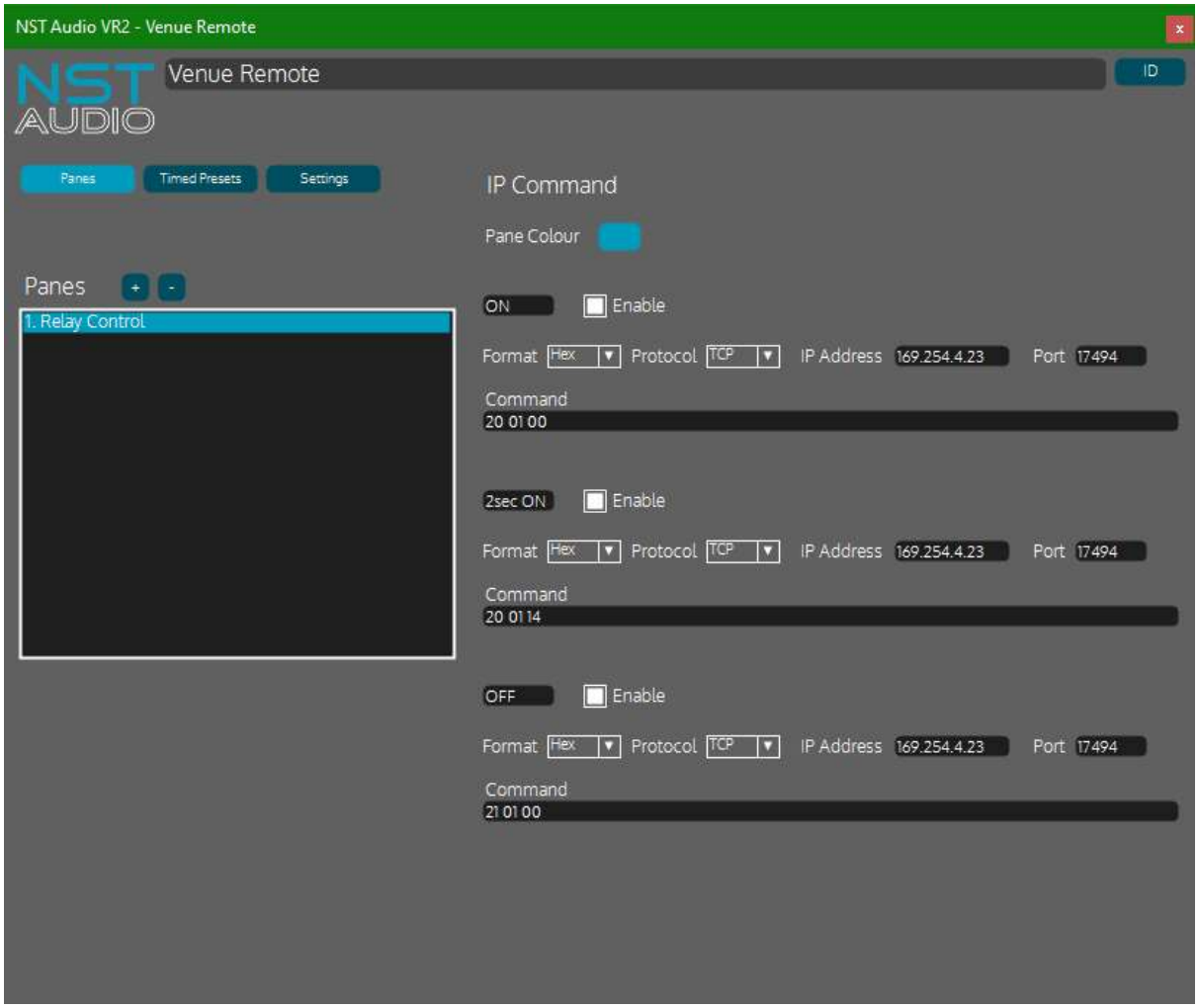
0x20 - turn the relay on command
0x02 - relay 2
0x32 (50) - 5 seconds (50 * 100ms)

Board will return 0 for success, 1 for failure.
Note - All bytes in a command must be sent in one TCP/IP packet .

To turn on relay #1, we would use (in hex) 0x20, 0x01, 0x00, and to turn it off, 0x21, 0x01, 0x00.

For a 2 second on>off pulse the command string will be 0x20, 0x01, 0x14.
0x14 = 20 in decimal, so 20 x 100ms = 2 seconds.

Let's add these three command strings to the VR2 pane configuration (overleaf)...



And we are done! The first relay on the board can now be turned on and off and pulsed for 2 seconds.

To select other relays, the second bye in the string is changed (01-08 corresponding to the physical relay number).

To turn multiple relays on or off together, use the command type 0x23, followed by a bitmapped representation of the relays outputs, for example, turn on relays 1,3,5 and 7, that is 10101010 which is 0x55.

Note that this is not an OR type operation – any relays with a 0 in the data byte above will be set to off.

This command is probably best used just to turn everything on (0x23, 0xff) or reset everything to off (0x23, 0x00).