



advanced electrophysiology solutions



LabLynx

User Manual

Portable Digital Acquisition System

© Neuralynx, Inc.
105 Commercial Drive, Bozeman, MT 59715
Phone 406.585.4542 • Fax 866.585.1743

www.Neuralynx.com
support@Neuralynx.com

Revision 1.1
2/18/2021

Table of Contents

1	Document Overview	5
2	LabLynx Overview	5
3	LabLynx Components and Accessories.....	6
3.1	LabLynx Base	6
3.2	Battery Charger	6
3.3	Wireless Access Point	6
3.4	16 or 32 channel Mux Headstage.....	7
3.5	Recommended Additional Products.....	7
3.5.1	Digital Multiplexing (mux) Headstages.....	7
3.5.2	Signal Mouse	7
3.5.3	Minirator	8
3.5.4	HS-36-mux Impedance Plug.....	8
3.5.5	2-Pin to BNC Cable	8
3.5.6	Neuralynx Halo Microdrives	8
4	Setup	9
4.1	Electrostatic Sensitive Equipment.....	9
4.2	Hardware Setup	10
4.3	Set Computer IP Address	11
4.4	Enable Jumbo Packets.....	13
4.5	Using Cheetah with LabLynx	14
4.5.1	Start Cheetah and Turn LabLynx On.....	16
4.5.2	Connect LabLynx to Test Setup	16
4.5.3	Test the LabLynx with a Test Input Signal.....	17
4.5.4	Troubleshooting	20
4.6	Channel Mapping and Channel Masking	21
4.6.1	Channel Mapping.....	21
4.6.2	Channel Masking	21
5	Hardware Overview	22
5.1	Enclosure.....	22
5.1.1	uDB-25 Connector	23
5.1.2	10-pin Connector	25
5.1.3	24-pin Connector	25
5.1.4	Status LED	26
5.1.5	On/Off Push Button	27
5.1.6	Manufacturing Access Panel.....	27
5.1.7	Battery Charger	27
5.1.8	Battery Charge Indicator.....	27
5.1.9	Back Panel Mounting Locations	27
5.2	Amplifier and Analog to Digital Converter	28
5.3	Wireless Communication	29
5.3.1	Wireless Access Point Setup.....	29
5.3.2	Selecting a Good Wi-Fi Channel	30
6	Configuring TTLs	32
7	LabLynx Commands.....	32



7.1	SetDigitalIOPortDirection.....	33
7.1.1	Example .cfg File Text.....	33
7.2	SetDigitalIOBit.....	34
7.2.1	Example .cfg File Text.....	34
7.3	GetDigitalIOPortValue.....	35
7.3.1	Example .cfg File Text.....	35
7.4	SetDigitalIOPortValue	35
7.4.1	Example .cfg File Text.....	35
7.5	GetTransmittedChannels.....	35
7.5.1	Example .cfg File Text.....	35
7.6	SetTransmittedChannels.....	36
7.6.1	Example .cfg File Text.....	38
8	Neuralynx Contact	38

List of Figures and Tables

Figure 3-1 LabLynx Base	6
Figure 3-2 Mini-USB Charger	6
Figure 3-3 Wireless Access Point	6
Figure 3-4 16 and 32 channel Mux Headstages	7
Figure 3-5 HS-64-mux	7
Figure 3-6 HS-64-mux Signal Mouse	7
Figure 3-7 Minirator	8
Figure 3-8 HS-36-mux Impedance Plug	8
Figure 3-9 2-Pin to BNC Cable	8
Figure 3-10 Halo-5 Microdrive	8
Figure 4-1 Steps to Connect Access Point to PC	10
Figure 4-2 Network Interface Properties	11
Figure 4-3 Computer IP Address Settings	12
Figure 4-4 Uncomment LabLynx.cfg	14
Figure 4-5 Setting Channel Configuration	15
Figure 4-6 Bank 1, 2, 3 and 4 Switches Up. Reference Switch Down	17
Figure 4-7 Bank 2, 3 and 4 Switches Up. Bank 1 and Reference Switches Down	17
Figure 4-8 Bank 3 and 4 Switches Up. Bank 1, 2 and Reference Switches Down	18
Figure 4-9 Bank 4 Switches Up. Bank 1, 2, 3 and Reference Switches Down	18
Figure 4-10 All Switches Down (1000 μ V Input Range)	19
Figure 4-11 All Switches Down (25 μ V Input Range)	19
Figure 4-12 All Switches Up (25 μ V Input Range)	20
Figure 5-1 Enclosure Front View	22
Figure 5-2 Enclosure Flipped View	22
Figure 5-3 Connector 1 Pinout	23
Figure 5-4 Connector 2 Pinout	23
Figure 5-5 Connector 3 Pinout	24
Figure 5-6 Connector 4 Pinout	24
Figure 5-7 10-pin Connector Pinout	25
Figure 5-8 24-pin Connector Pinout	25
Figure 5-9 5 LabLynx Set to Channel 106	30
Figure 5-10 Login Page using Microsoft Edge	31
Figure 5-11 Changing WiFi Channel	31



1 Document Overview

This document describes the specifications and features of the LabLynx. It also explains how to set up, test, and use your LabLynx.

2 LabLynx Overview

The LabLynx is a portable, battery powered digital acquisition system. It is capable of recording from up to 256 individual electrodes using Neuralynx digital multiplexing (mux) headstages. The physiological signals are digitized at the Neuralynx mux headstage and transmitted wirelessly from the LabLynx. Wireless communication uses 5 GHz WiFi (802.11 ac).

Features:

- Wirelessly transmit up to 256 channels
- Record up to 256 channels to microSD card
- Record from up to four different digital (mux) headstages
- 8.5 hour wireless run-time
- 7 hour charge time
- Time synchronization with Cheetah to within ± 5 ms
- 30 KHz Sampling Rate
- 16-bit Resolution
- ± 5 mV Input Range
- >80 dB Common Mode Rejection Ratio (CMRR) at 60 Hz
- <2.5 μV_{RMS} Noise (0.1 Hz to 8 kHz)
- Up to eight TTL-compatible digital IO
- Compatible with Neuralynx FLEDs
- Compatible with Neuralynx EIBs
- Compatible with Neuralynx mux headstages

3 LabLynx Components and Accessories

3.1 *LabLynx Base*

- Four uDB-25 connectors for digital headstages
- FLED driver input
- Eight GPIO inputs
- On/off push button
- Mini-USB charging input
- Four 4-40 mounting holes

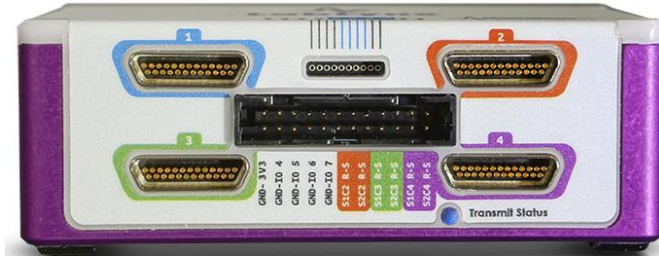


Figure 3-1 LabLynx Base

3.2 *Battery Charger*

- A grounded mini-USB to barrel connector cable adapter necessary to charge the batteries of the LabLynx while device is in use.
- 5 V power supply
- Power cable associated with country of origin.



Figure 3-2 Mini-USB Charger

3.3 *Wireless Access Point*

- Wireless access point
- POE Injector
- Power supply
- Standard Ethernet cables



Figure 3-3 Wireless Access Point

3.4 16 or 32 channel Mux Headstage

- One headstage comes standard with the LabLynx system



Figure 3-4 16 and 32 channel Mux Headstages

3.5 Recommended Additional Products

Neuralynx provides additional products to support the assembly and use of the LabLynx.

3.5.1 Digital Multiplexing (mux) Headstages

- Digitize neural signals at the headstage
- Reduce the wire count, weight, size and cost of the tether
- 30 kHz sampling rate, 16-bit resolution
- A wide variety of mux headstages are available



Figure 3-5 HS-64-mux

3.5.2 Signal Mouse

- Verify your recording setup by inputting a test signal into all headstage channels

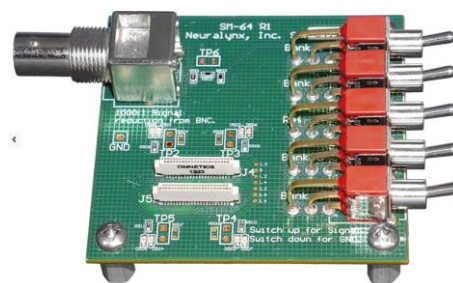


Figure 3-6 HS-64-mux Signal Mouse

3.5.3 Minirator

- A convenient, battery powered, handheld signal generator that outputs Sine/Ramp/Noise waves to simulate spikes
- This device can be used to supply test signal into a Signal Mouse



Figure 3-7 Minirator

3.5.4 HS-36-mux Impedance Plug

- Provides different resistance values in banks of 8 for multiple testing applications on 32/64 channel multiplexing (mux) headstages

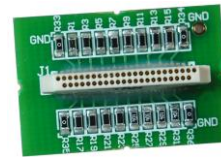


Figure 3-8 HS-36-mux
Impedance Plug

3.5.5 2-Pin to BNC Cable

- Connect signal and ground between a BNC and a standard, male 0.10" pitch connector



Figure 3-9 2-Pin to BNC
Cable

3.5.6 Neuralynx Halo Microdrives

- Light weight, easy to construct, and cost effective microdrives for tetrode freely moving animal research.



Figure 3-10 Halo-5
Microdrive

4 Setup

The LabLynx communicates wirelessly with a PC running Cheetah. A standard LabLynx license for Cheetah is non-transferrable. Your LabLynx license only allows use of Cheetah with a LabLynx. If you purchased your LabLynx with a PC, your system would be pre-configured. If you already own a standard Neuralynx Cheetah license, the LabLynx can be used with Cheetah. The LabLynx requires Cheetah 6.4 or later.

LabLynx supports up to 256 channels. The number of channels that can be viewed and recorded in Cheetah depends on the Cheetah license.

If you need to change the order of acquisition channels or reduce the number of channels recorded, see section **4.6 Channel Mapping and Channel Masking**.

4.1 ***Electrostatic Sensitive Equipment***

All Neuralynx equipment is sensitive to electrostatic discharge (ESD) and should be handled appropriately. Always wear a static wrist strap and use appropriate ESD measures when handling any electronics. If you have questions, please contact Neuralynx for more information.

4.2 Hardware Setup

To connect a LabLynx directly to a PC first check to ensure the PC meets the minimum system requirements. These are outlined on the Neuralynx website in the support section. Next, follow the diagram below to ensure you connect the necessary hardware, and then follow the steps identified to ensure the LabLynx Connects to the PC through the wireless access point. It is recommended that the Ruckus Router wireless access point be mounted on the ceiling for best performance.

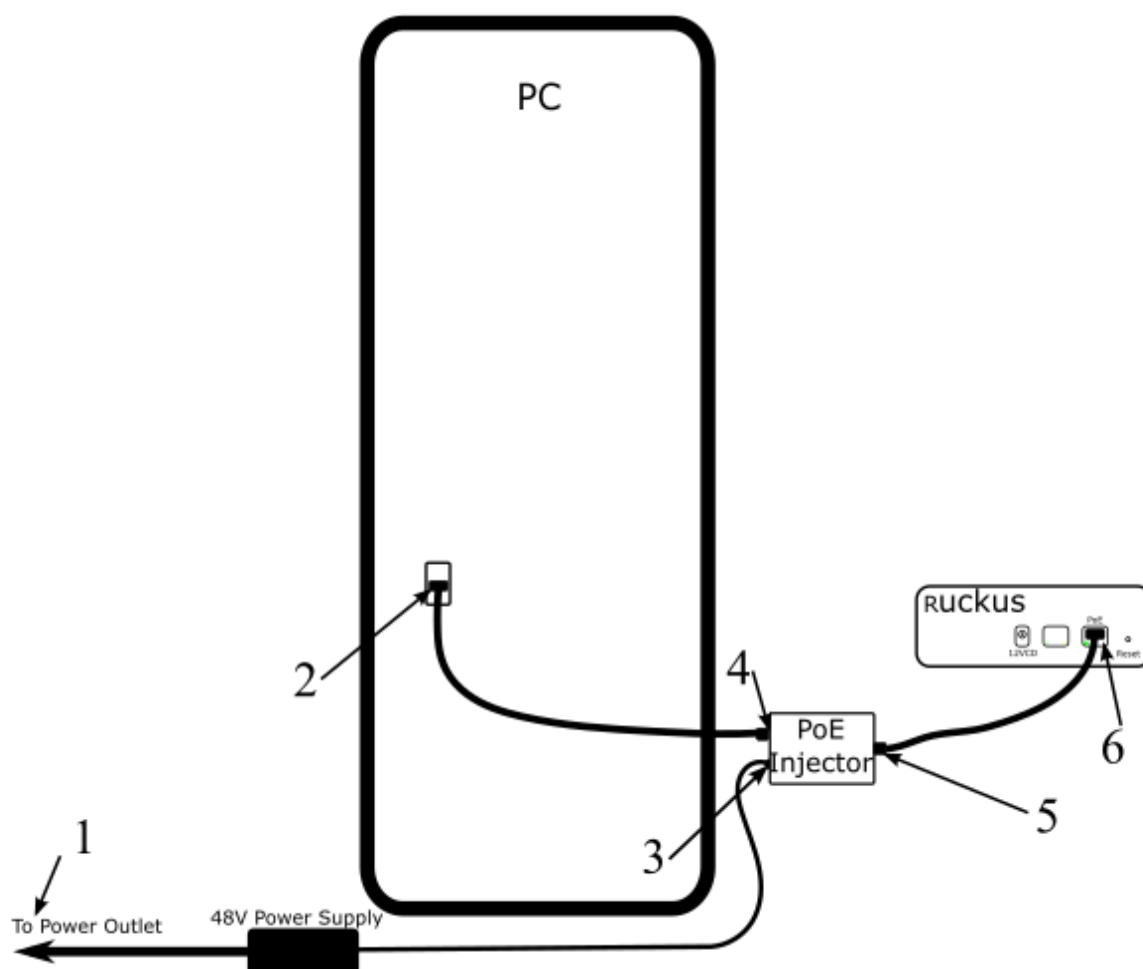


Figure 4-1 Steps to Connect Access Point to PC

Step	Function
1	Connect the 48 V power supply to a standard wall outlet.
2	Connect Ethernet cable A to the Ethernet port on the PC
3	Connect the 48 V power supply to the PoE Injector
4	Connect Ethernet cable A to the LAN IN port of the PoE Injector
5	Connect Ethernet cable B to the POWER+DATA OUT port of the PoE Injector
6	Connect Ethernet cable B to the PoE port on the access point

4.3 Set Computer IP Address

To enable communication between the LabLynx and the acquisition PC, a static IP address must be set for the network interface that is used to connect to the wireless access point. To set the static IP address, go to:

Control Panel > Network and Internet > Network and Sharing Center > Change Adapter Settings

Determine the appropriate network connection and open its properties.

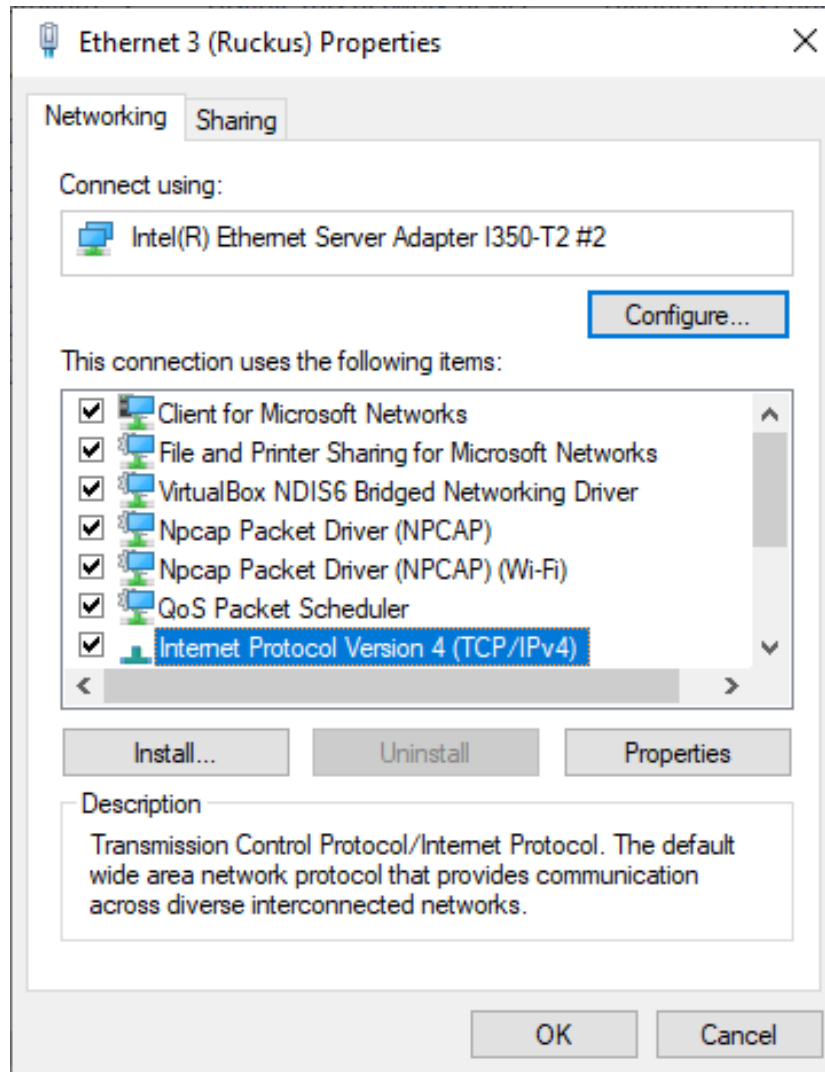
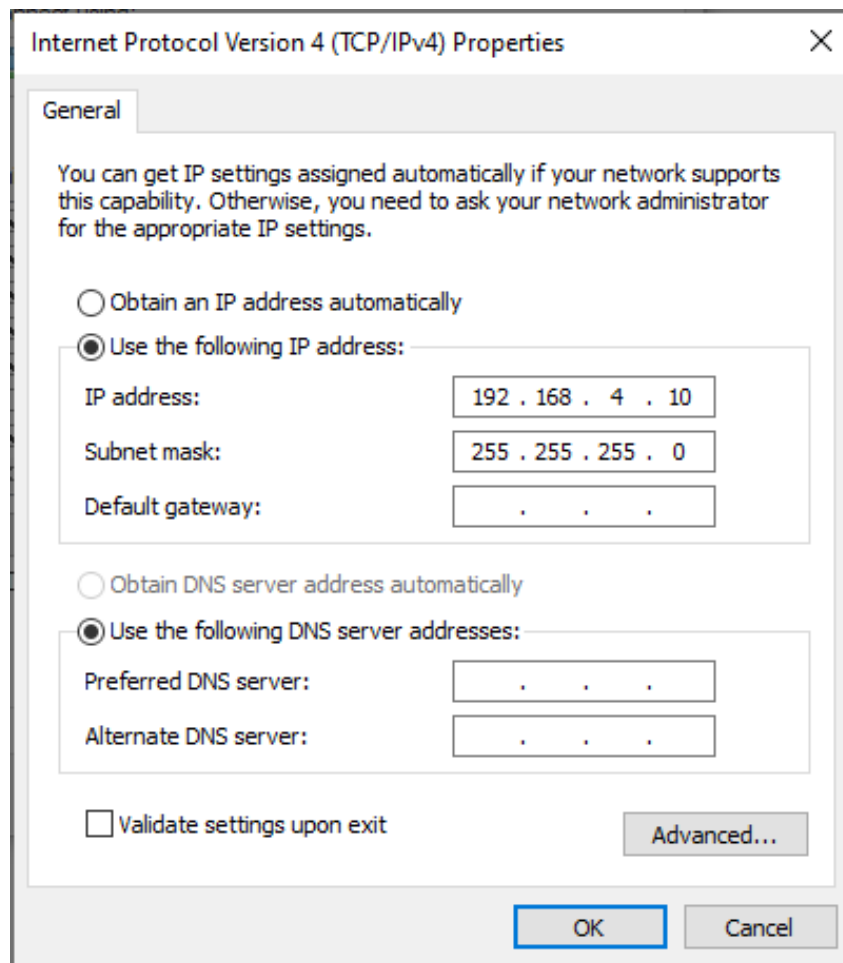


Figure 4-2 Network Interface Properties

Set the IP information as shown below.



The screenshot shows the 'Internet Protocol Version 4 (TCP/IPv4) Properties' dialog box with the 'General' tab selected. The dialog box contains instructions on how to obtain IP settings and two main configuration sections. The first section, 'Use the following IP address', is selected with a radio button and contains three input fields: 'IP address' (192 . 168 . 4 . 10), 'Subnet mask' (255 . 255 . 255 . 0), and 'Default gateway' (three empty fields separated by dots). The second section, 'Use the following DNS server addresses', is also selected with a radio button and contains two input fields: 'Preferred DNS server' and 'Alternate DNS server', both with three empty fields separated by dots. At the bottom, there is a checkbox for 'Validate settings upon exit' and an 'Advanced...' button. The 'OK' and 'Cancel' buttons are at the bottom right.

Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

☐ Obtain an IP address automatically

☒ Use the following IP address:

IP address: 192 . 168 . 4 . 10

Subnet mask: 255 . 255 . 255 . 0

Default gateway: . . .

☐ Obtain DNS server address automatically

☒ Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

☐ Validate settings upon exit

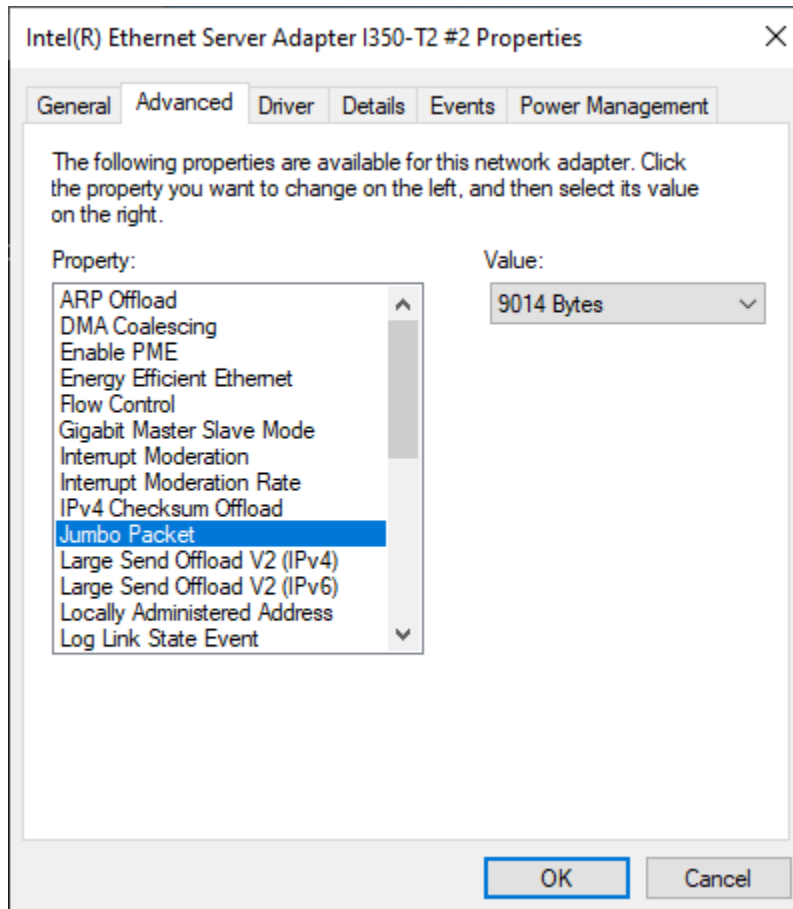
Advanced...

OK Cancel

Figure 4-3 Computer IP Address Settings

4.4 **Enable Jumbo Packets**

In the properties window (see *Figure 4-2 Network Interface Properties* above), click the “Configure...” button and then go to the “Advanced” tab. Select the “Jumbo Packet” property and set the value to 9014 Bytes. Then click OK



4.5 Using Cheetah with LabLynx

Open Cheetah and select the appropriate configuration file for your setup. Open Cheetah.cfg using a text editor like Notepad or Notepad++ and check that the Acquisition System Setup has LabLynx.cfg uncommented as seen below. A line in a .cfg file is uncommented by deleting the # at the start of the line.

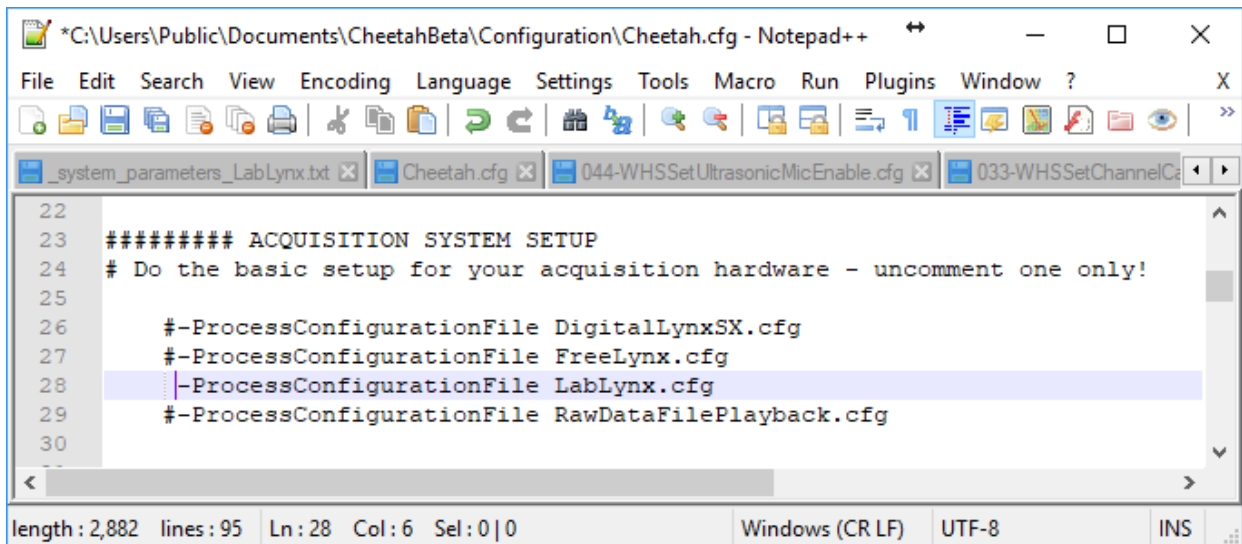
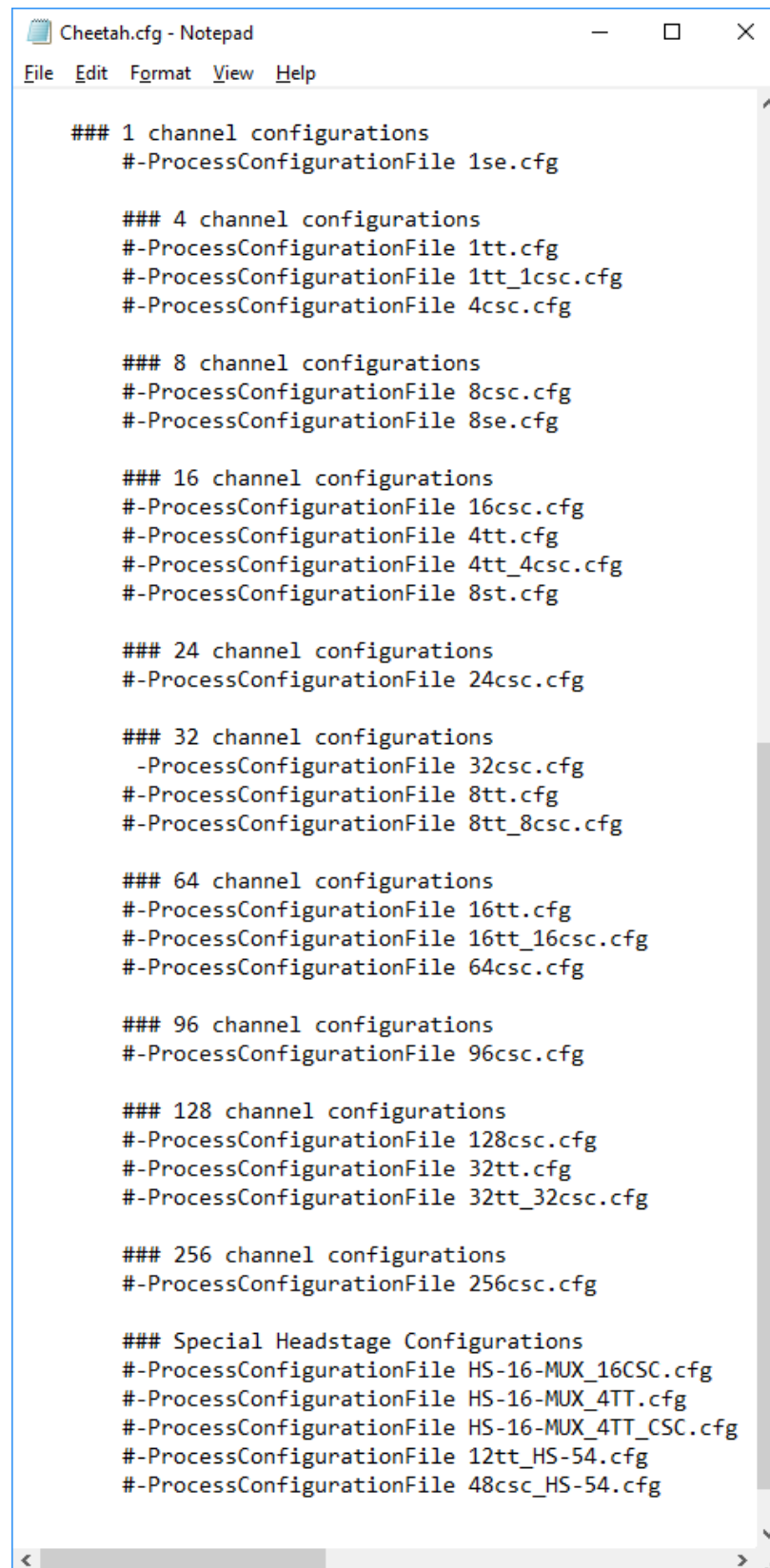


Figure 4-4 Uncomment LabLynx.cfg

Select a channel configuration compatible with your setup by uncommenting the appropriate line. In the example below, the system is setup for 32 channels. A CSC (Continuously Sampled Channel) setup will plot 32 channels on the display.



```
### 1 channel configurations
#-ProcessConfigurationFile 1se.cfg

### 4 channel configurations
#-ProcessConfigurationFile 1tt.cfg
#-ProcessConfigurationFile 1tt_1csc.cfg
#-ProcessConfigurationFile 4csc.cfg

### 8 channel configurations
#-ProcessConfigurationFile 8csc.cfg
#-ProcessConfigurationFile 8se.cfg

### 16 channel configurations
#-ProcessConfigurationFile 16csc.cfg
#-ProcessConfigurationFile 4tt.cfg
#-ProcessConfigurationFile 4tt_4csc.cfg
#-ProcessConfigurationFile 8st.cfg

### 24 channel configurations
#-ProcessConfigurationFile 24csc.cfg

### 32 channel configurations
#-ProcessConfigurationFile 32csc.cfg
#-ProcessConfigurationFile 8tt.cfg
#-ProcessConfigurationFile 8tt_8csc.cfg

### 64 channel configurations
#-ProcessConfigurationFile 16tt.cfg
#-ProcessConfigurationFile 16tt_16csc.cfg
#-ProcessConfigurationFile 64csc.cfg

### 96 channel configurations
#-ProcessConfigurationFile 96csc.cfg

### 128 channel configurations
#-ProcessConfigurationFile 128csc.cfg
#-ProcessConfigurationFile 32tt.cfg
#-ProcessConfigurationFile 32tt_32csc.cfg

### 256 channel configurations
#-ProcessConfigurationFile 256csc.cfg

### Special Headstage Configurations
#-ProcessConfigurationFile HS-16-MUX_16CSC.cfg
#-ProcessConfigurationFile HS-16-MUX_4TT.cfg
#-ProcessConfigurationFile HS-16-MUX_4TT_CSC.cfg
#-ProcessConfigurationFile 12tt_HS-54.cfg
#-ProcessConfigurationFile 48csc_HS-54.cfg
```

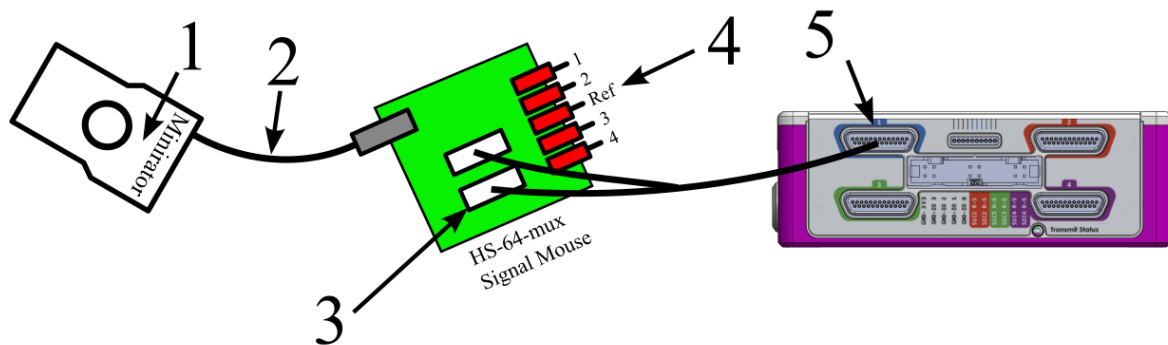
Figure 4-5 Setting Channel Configuration

4.5.1 Start Cheetah and Turn LabLynx On

Open Cheetah using the configuration file you edited above, Cheetah.cfg. While Cheetah is loading, turn on the LabLynx using the push button on the side. When the LabLynx is turned on, the status LED will turn blue. After the LabLynx has booted and connected to Cheetah, the status LED turns green. LED codes are described in **5.1.4 Status LED**.

- Cheetah will attempt to connect with the LabLynx for 60 seconds before it times out.
- The LabLynx detects connected headstages during bootup. Therefore, headstages must be connected to the LabLynx before turning the LabLynx on. Headstages connected after bootup will not function.

4.5.2 Connect LabLynx to Test Setup



Step	Description
1	Set the Minirator, or other signal source, to output a 1 V _{PP} sine wave at 100 Hz. The Signal Mouse will reduce this signal to approximately 1 mV _{PP} .
2	Connect the Minirator, or other signal source, to the Signal Mouse using a BNC cable.
3	Connect HS-64-mux to the two Omnetics connectors located on the HS-64-mux Signal Mouse
4	Turn the Bank 1, 2, 3, and 4 switches on the Signal Mouse to the Signal position (Up). Turn the Reference switch on the Signal Mouse to the Ground position (Down).
5	Connect the uDB-25 connector from the HS-64-mux into connector 1 of the LabLynx.



4.5.3 Test the LabLynx with a Test Input Signal

In Cheetah, select the *ACQ* button to start acquisition. Set the Input Range for all 64 CSCs to 1000 μ V. Each should show approximately 1 mV_{PP} sine waves.

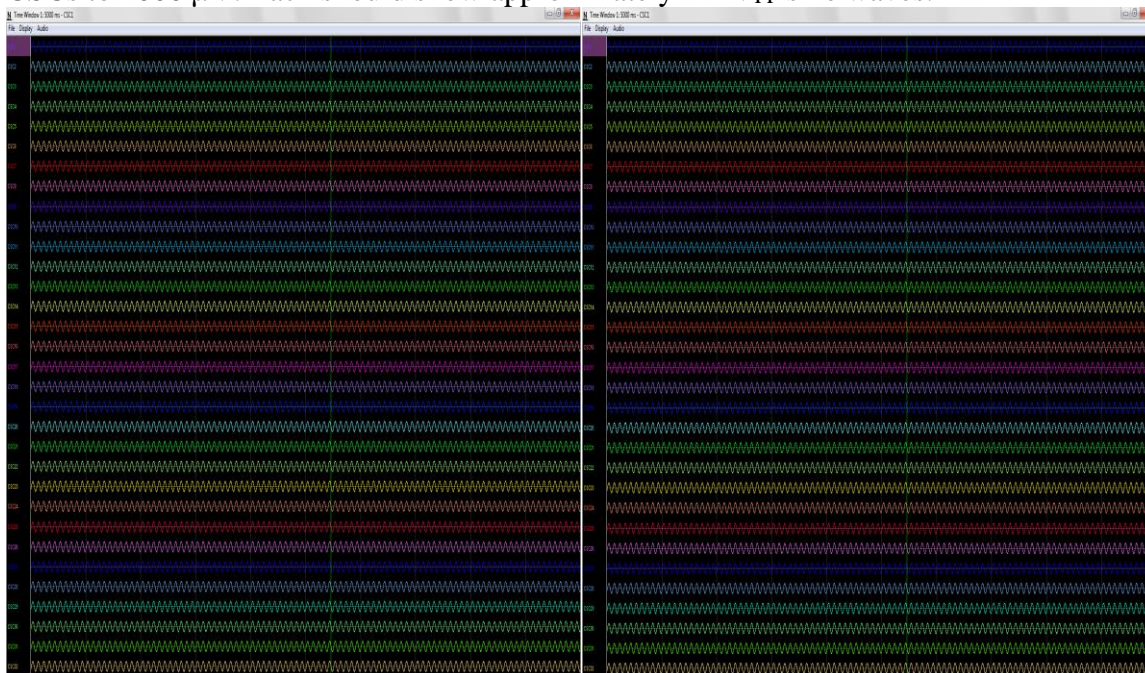


Figure 4-6 Bank 1, 2, 3 and 4 Switches Up. Reference Switch Down

Switch the Bank 1 switch to the Ground position (down). CSCs 1-8 and 33-40 should now be flat-lined while CSCs 9-32 and 41-64 should still show 1 mV_{PP} sine waves.

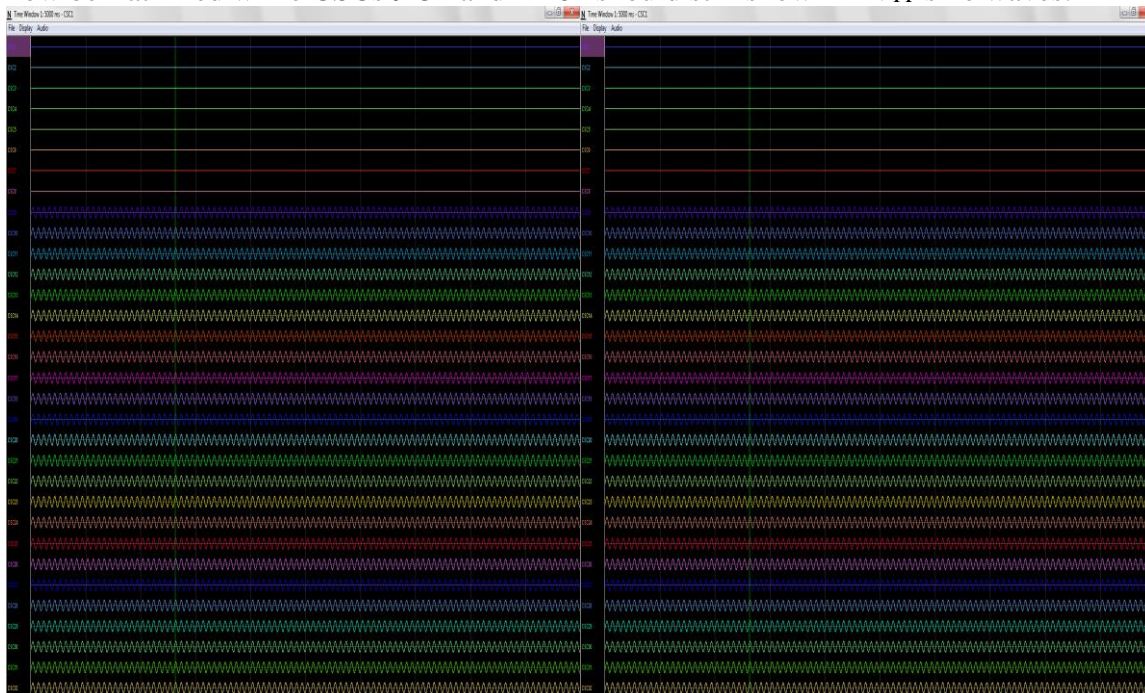


Figure 4-7 Bank 2, 3 and 4 Switches Up. Bank 1 and Reference Switches Down



Switch the Bank 2 switch to the Ground position (down). CSCs 1-16 and 33-48 should now be flat-lined while CSCs 17-32 and 49-64 should still show 1 mV_{PP} sine waves.

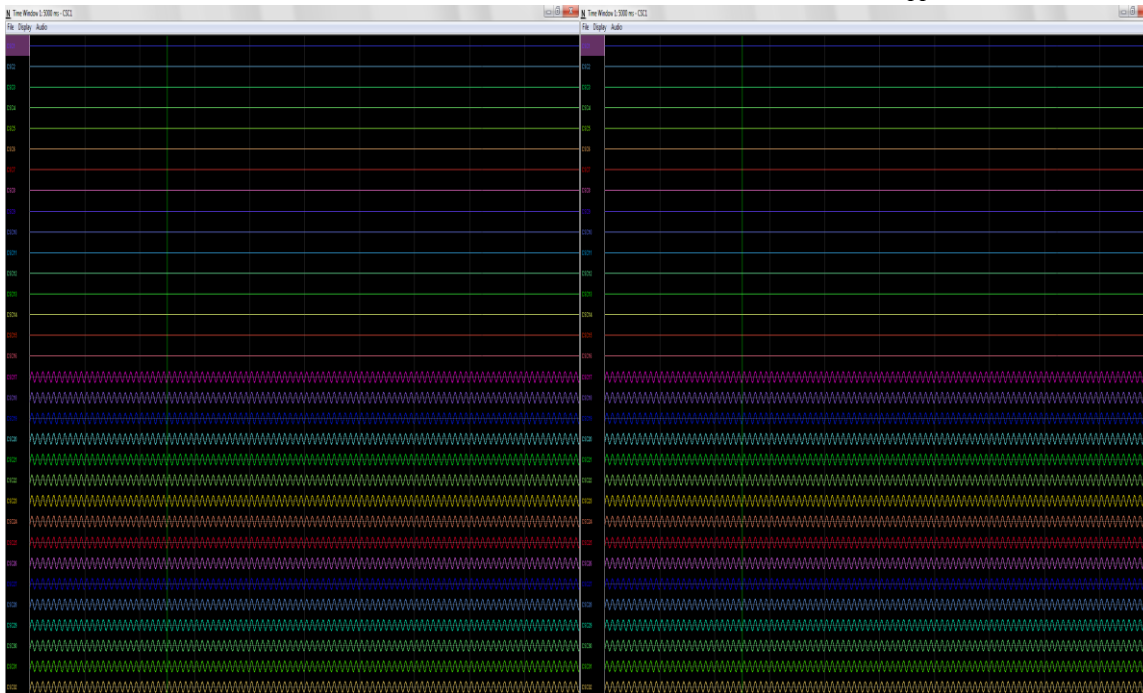


Figure 4-8 Bank 3 and 4 Switches Up. Bank 1, 2 and Reference Switches Down

Switch the Bank 3 switch to the Ground position (down). CSCs 1-24 and 33-56 should now be flat-lined while CSCs 25-32 and 57-64 should still show 1 mV_{PP} sine waves.

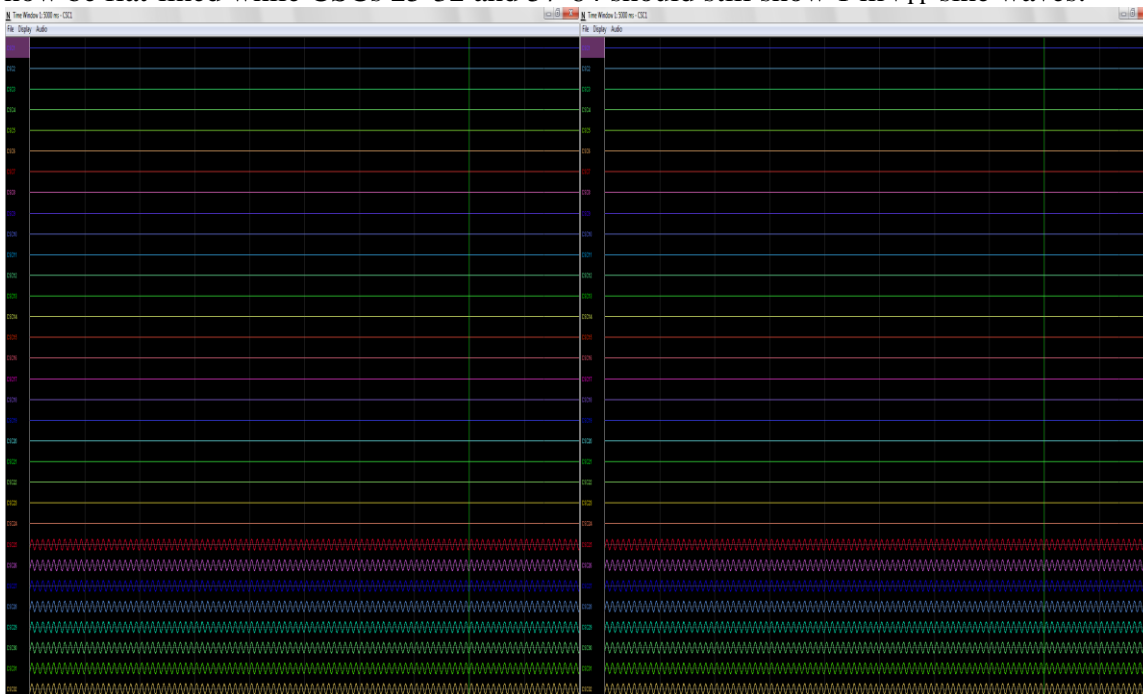


Figure 4-9 Bank 4 Switches Up. Bank 1, 2, 3 and Reference Switches Down



Switch the Bank 4 switch to the Ground position (down). CSCs 1-64 should now be flat-lined.

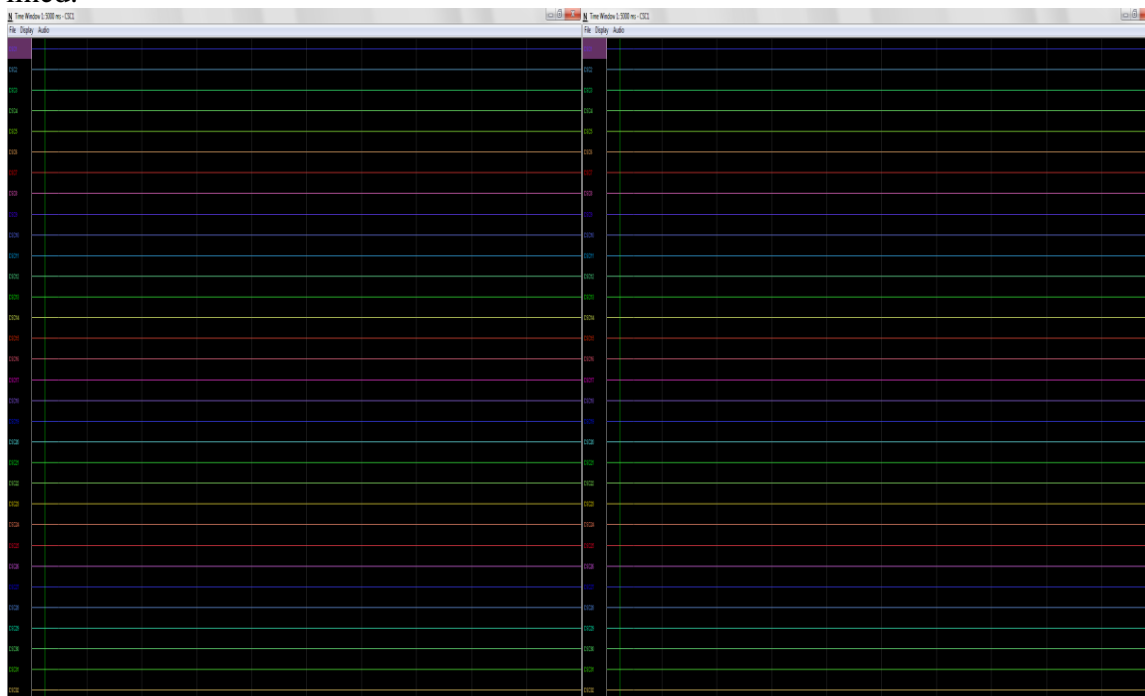


Figure 4-10 All Switches Down (1000 μ V Input Range)

Set the Input Range for all 64 CSCs to 25 μ V. CSCs 1-64 now show the baseline noise. Each should be less than 25 μ Vpp and without repetitive signals such as 60 Hz noise.

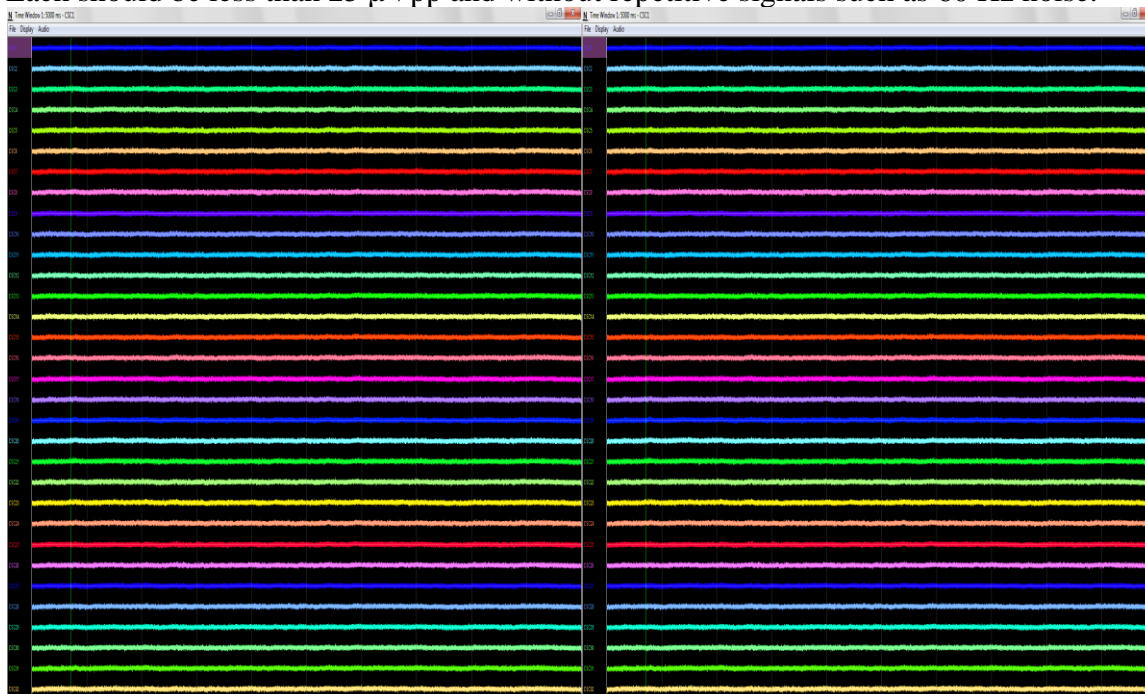


Figure 4-11 All Switches Down (25 μ V Input Range)



Switch all the switches to the Signal position (up). Each should be less than $25\ \mu\text{Vpp}$ and without repetitive signals such as 60 Hz noise.

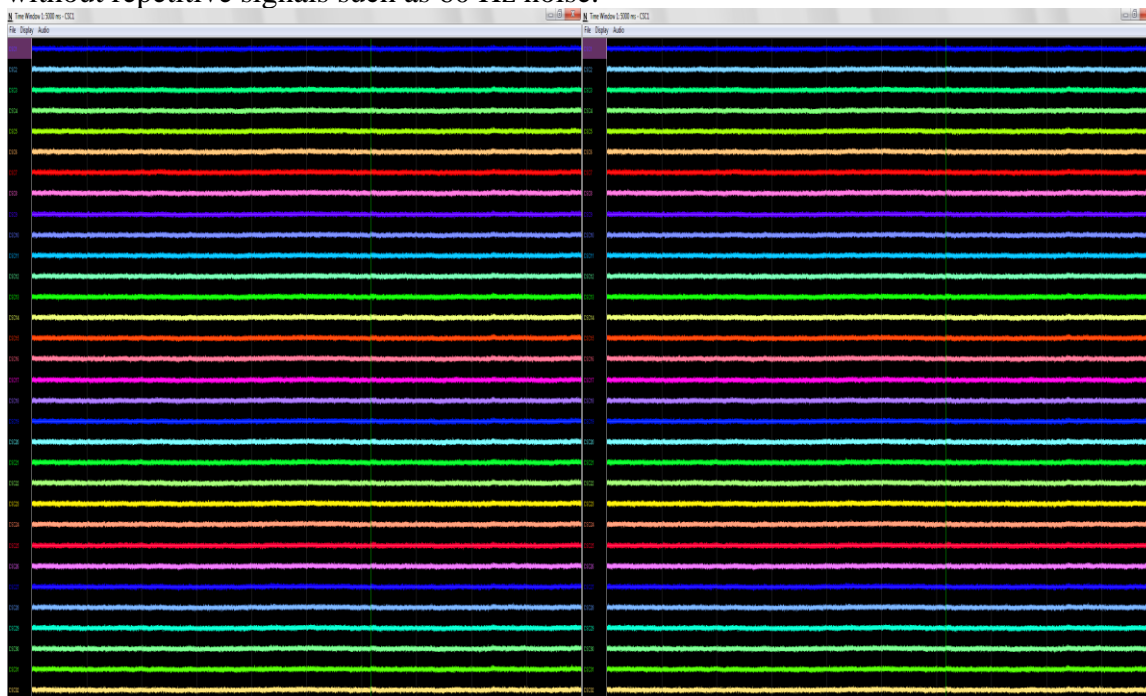


Figure 4-12 All Switches Up ($25\ \mu\text{V}$ Input Range)

4.5.4 Troubleshooting

If Cheetah does not receive data after starting acquisition, your firewall might be preventing Cheetah from receiving LabLynx data.

4.6 **Channel Mapping and Channel Masking**

The LabLynx can transmit up to 256 channels. A 256-channel setup would require four 64-channel mux headstages. Most LabLynx configurations will use fewer than the available 256 channels. To transmit the correct number of channels in the correct order, channel mapping and channel masking may be needed. In most cases, channel mapping and channel masking will be pre-configured by Neuralynx when the LabLynx is shipped.

Channel Mapping – Using a Cheetah configuration file to change the order in which channels are displayed.

Channel Masking – Using a LabLynx command to reduce the number of channels that are transmitted over Wi-Fi.

4.6.1 **Channel Mapping**

Most Neuralynx multiplexing headstages do not require channel mapping. However, some headstages, particularly custom headstages, may require channel mapping to properly correlate the headstage input channel numbering with Cheetah channel numbering. Mapping is done within the same .cfg file that is used to setup your channel layout in Cheetah, such as the 32csc.cfg file used in the test example above.

The headstage input channel numbering can be found in the user manual for your headstage. Contact Neuralynx support for assistance with channel mapping.

4.6.2 **Channel Masking**

If using less than the max available 256 channels of a LabLynx, you may want to reduce the total amount of data generated by reducing the number of channels acquired and transmitted by LabLynx. This is done using channel masking.

Channel masking is done with the `SetTransmittedChannels` command. See **7.6 *SetTransmittedChannels*** for details. Contact Neuralynx support for assistance with channel masking.

5 Hardware Overview

5.1 Enclosure

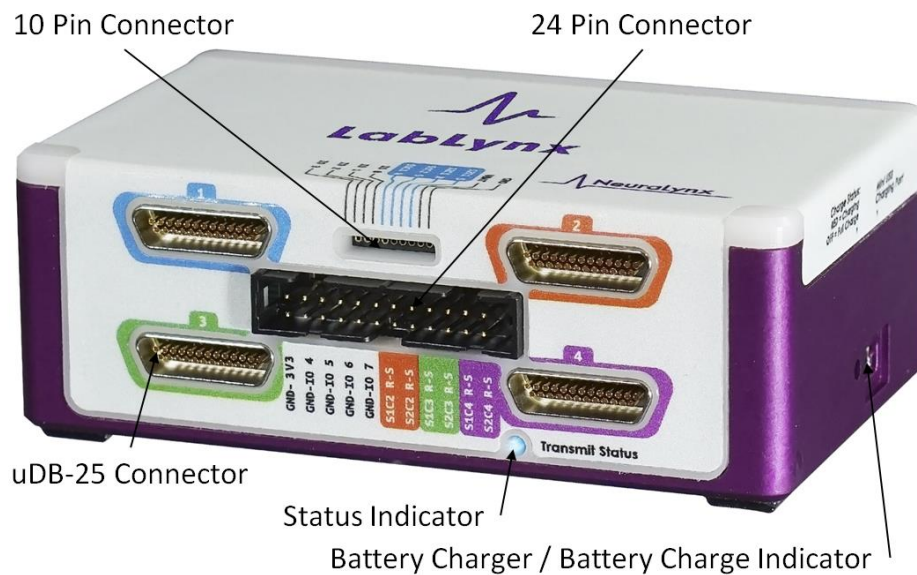


Figure 5-1 Enclosure Front View

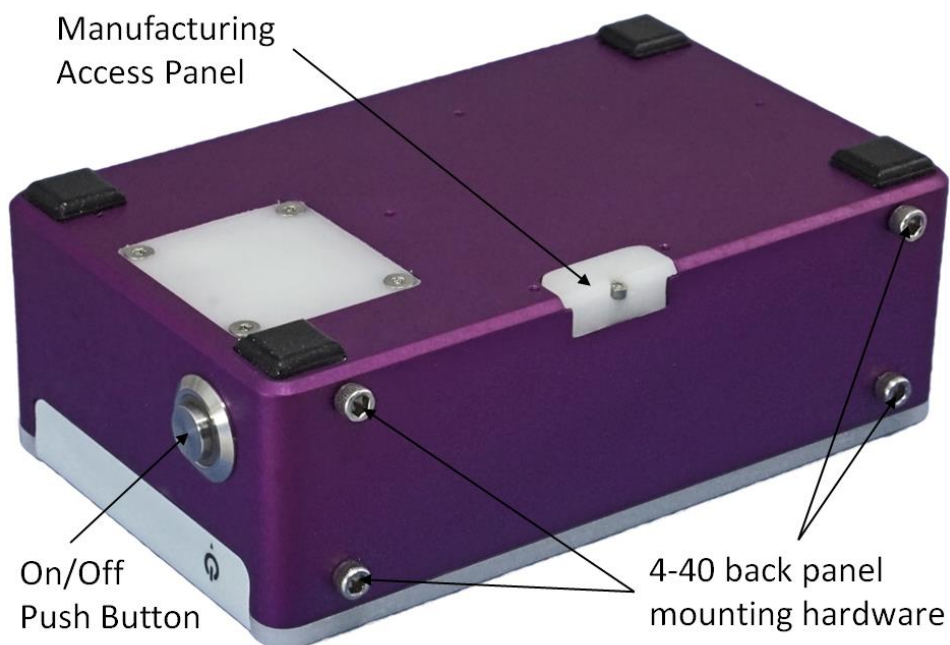


Figure 5-2 Enclosure Flipped View

5.1.1 uDB-25 Connector

The uDB-25 connector is used for mux headstages. Up to four different mux headstages can be connected to a LabLynx. Along with the digital communication lines, each uDB-25 connector has two Stim pairs which can be used to either drive a stimulus current or a FLED. uDB-25 connector 1 stimulus lines are routed to the 10-pin connector so that it can be driven with the Modular FLED driver. See **5.1.2 10-pin Connector**. Stim channels on connectors 2, 3, and 4 are routed to the 24-pin connector, **5.1.3 24-pin Connector**. Routing of stim channels is easy to follow with color coding on the LabLynx label and the connector images below.

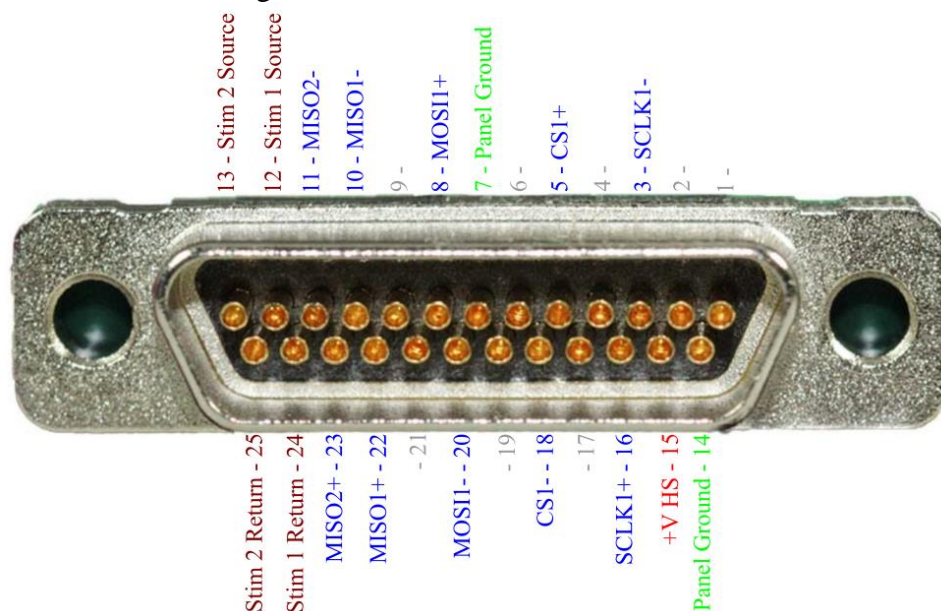


Figure 5-3 Connector 1 Pinout

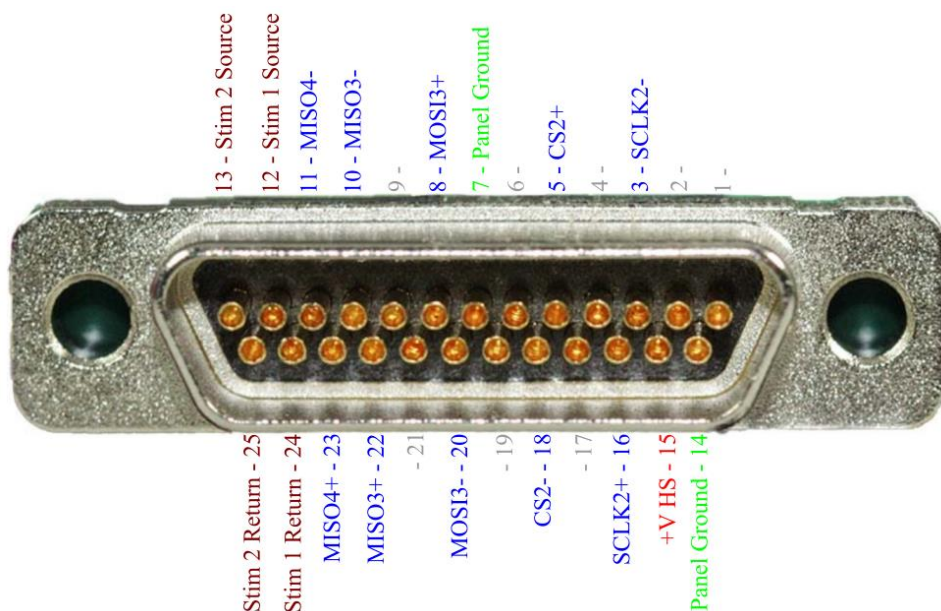


Figure 5-4 Connector 2 Pinout

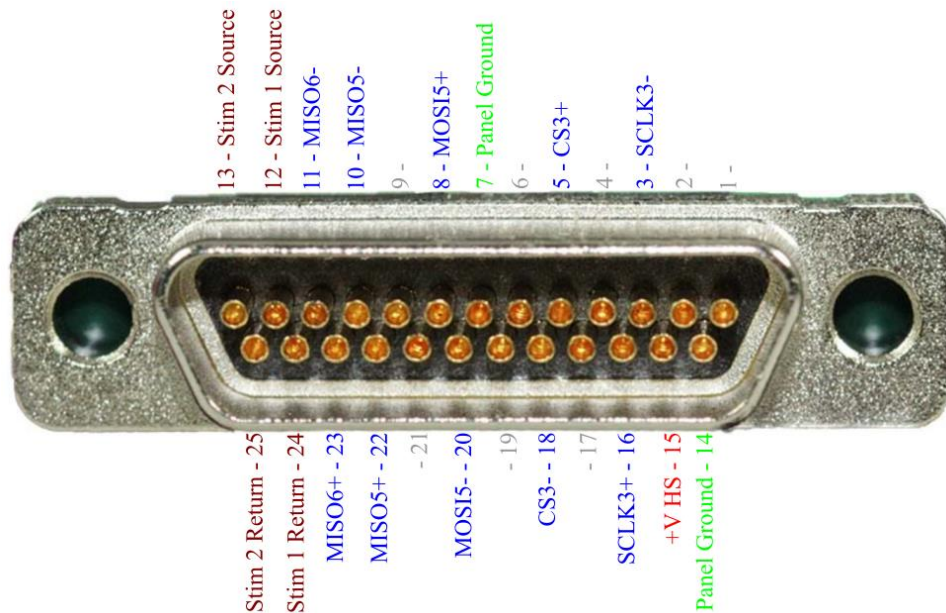


Figure 5-5 Connector 3 Pinout

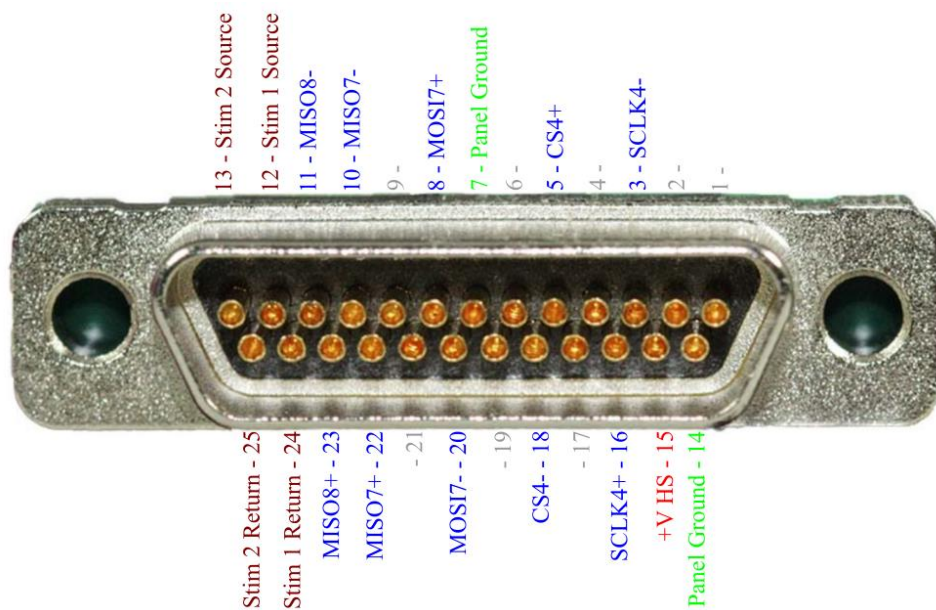


Figure 5-6 Connector 4 Pinout

5.1.2 10-pin Connector

The 10-pin connector utilizes a 0.05" pitch Mill-Max connector. It is the interface for the Modular FLED Driver and Ultrasonic Microphone. An additional 4 GPIO signals can be connected through the 10-pin connector as well as the stimulus lines of the connector 1, if the FLED Driver or Ultrasonic microphone are not in use.

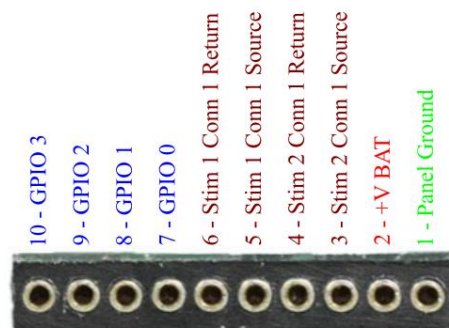


Figure 5-7 10-pin Connector Pinout

5.1.3 24-pin Connector

The 24-pin connector utilizes a 0.10" pitch 3M connector. It is the primary location for GPIO inputs. Stimulation channels to uDB-25 connectors 2, 3, and 4 are also located on the connector. On the LabLynx, stimulation channels are labeled CiSj where 'i' is the connector number and 'j' is the stim channel number. For example, C1S2 stands for connector 1, stim channel 2.

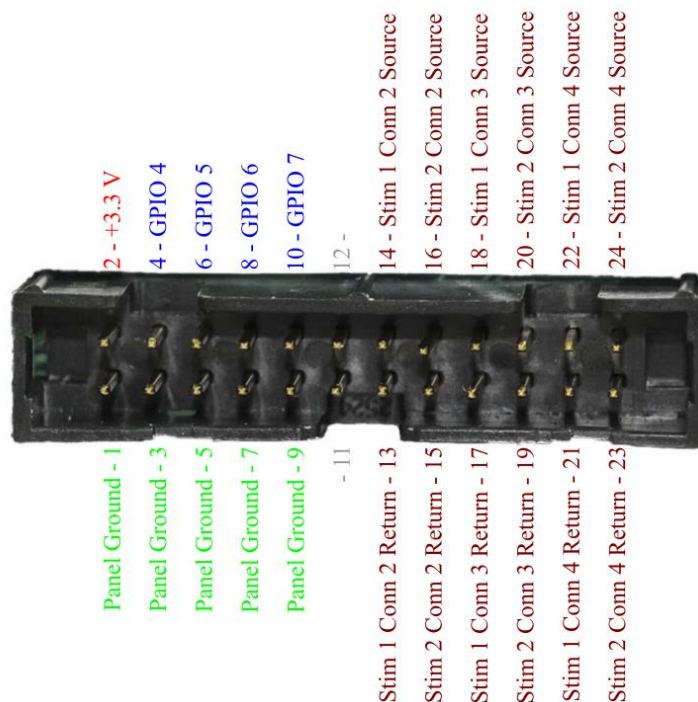


Figure 5-8 24-pin Connector Pinout



5.1.4 Status LED

5.1.4.1 Normal LED Statuses

5.1.4.1.1 Power Up

The LED is solid blue at power up and remains blue while trying to connect, unless an error occurs.

5.1.4.1.2 Successful Connection

The LED stays green.

5.1.4.1.3 System Parameter Update

The LED blinks blue when the LabLynx is updating its parameters using `system_parameters.txt` on the microSD card.

5.1.4.2 Error Codes

5.1.4.2.1 Invalid Network Name (SSID)

The LED changes between red and blue at a consistent rate.

5.1.4.2.2 Unable to Contact Access Point

The LED changes between red and off. This error requires the LabLynx to be rebooted.

5.1.4.2.3 Jumbo Packet Test Failed

The LED changes between red and blue with a long wait on blue.

5.1.5 On/Off Push Button

Press and release the button to turn LabLynx on/off. The status LED will light blue when turning on and change to green after connecting to Cheetah. The status LED will turn off when the LabLynx is turned off.

5.1.6 Manufacturing Access Panel

Please contact Neuralynx support prior to removing hatch or updating LabLynx firmware.

LabLynx firmware is stored on a microSD card that is inserted at this location. To remove the microSD card access cover, unscrew the 0-80 socket head screw and carefully remove the cover.

5.1.7 Battery Charger

A mini-USB connector is used as the battery charging interface. To achieve optimal charge rate, use the provided charging cable. The LabLynx can be operated while the batteries are being charged. The provided charging cable creates a grounded mini-USB interface. If the provided charging cable is not used to charge batteries while device is in operation, AC power will induce noise.

LabLynx can charge from a PC USB interface, but at a much slower rate. When charging from a PC USB interface, LabLynx will likely deplete the battery faster than the battery can charge.

5.1.8 Battery Charge Indicator

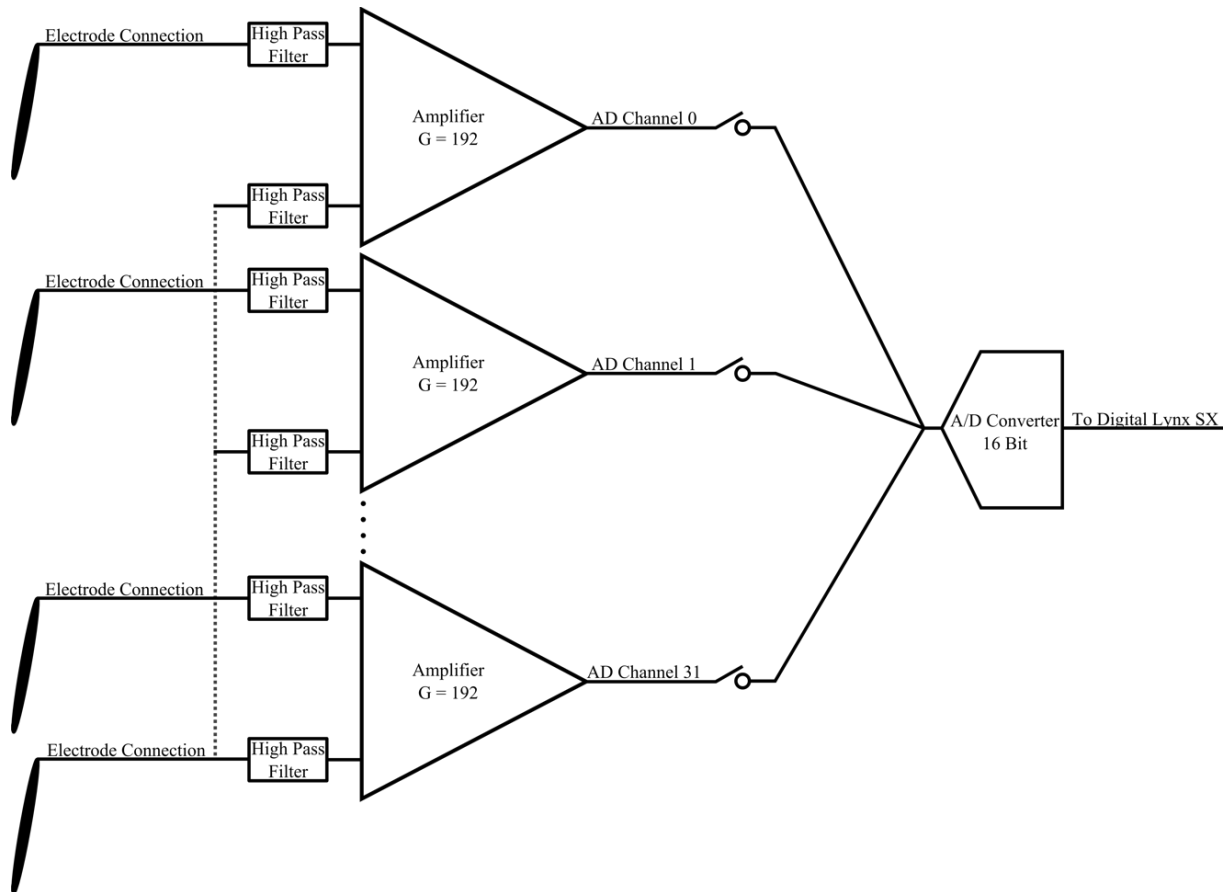
When charging, the battery charge indicator will light red. When charging is complete, the charge indicator will turn off. If no mini-USB charger is connected, the battery charge indicator is off.

5.1.9 Back Panel Mounting Locations

To mount the LabLynx enclosure to a ceiling, cage, or a Gimbal mechanism (to act as a commutator), there are four available 4-40 tapped holes on the back panel of the LabLynx. These holes are centered on the midline of the enclosure and spaced 1" x 3.5" apart in a rectangle pattern.

5.2 Amplifier and Analog to Digital Converter

Each channel is digitized on the mux headstage using a fixed reference. The channels are AC coupled and the gain is fixed at 192 [V/V]. Each analog to digital converter is 16-bit and can digitize 32 channels. This concept is illustrated in the figure below.



5.3 **Wireless Communication**

The LabLynx communicates wirelessly using a Wi-Fi access point. LabLynx uses 802.11ac Wi-Fi and can transmit up to 256 channels of data.

Wireless access points come preconfigured. If needed, they can be configured to use any available 5 GHz Wi-Fi channel. Signal strength decreases as the LabLynx is moved further away from the access point. For optimal wireless communication, the access point should be ceiling-mounted in the area where the LabLynx is used. See below.

5.3.1 **Wireless Access Point Setup**

If the access point needs to be configured, its interface can be accessed via a web browser. The IP address for an access point received from Neuralynx is 192.168.4.1. The PC accessing this IP address should be on the same subnet 192.168.4.xxx. The login to the access point is:

- Username: “admin”
- Password: “cheetah”

Tips for Improving Wireless Communication

- Minimize obstructions between the LabLynx and access point.
- Minimize distance between the LabLynx and access point.
- Ceiling-mount the access point over the area where the LabLynx is used.
- If you are losing wireless data, use a computer with a Wi-Fi network card or use a mobile phone to scan the area for wireless networks. Configure your access point to use a clear 5 GHz channel. The further your channel is from other networks, the better.

Contact Neuralynx if you need assistance with access point setup.

5.3.2 Selecting a Good Wi-Fi Channel

Due to the common use of wireless communication, it may be necessary to change the Wi-Fi channel on which your device communicates. The default channel is 136.

The following devices can be used to scan your area for other wireless networks:

- Mobile phone – WiFi Analyzer is a free Android app
- Laptop with a wireless network card
- PC with a wireless network card

You can likely find a free app for your mobile phone, laptop or PC that will allow you to view wireless networks available in your area. The screenshots below are from WiFi Analyzer on Android.

The picture below shows 5 GHz networks in the area. You can see the LabLynx wireless network with SSID `nlx_wireless_lablynx`, and three other networks indicated by colored bumps in the top third of the image.

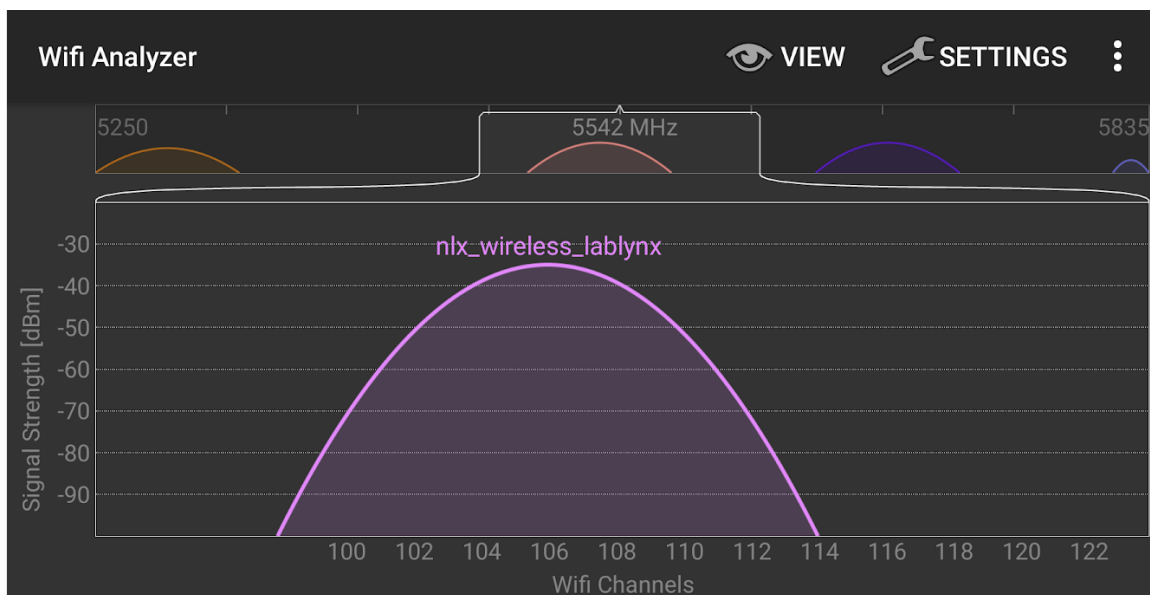


Figure 5-9 5 LabLynx Set to Channel 106

If your network scan shows your LabLynx wireless network is near or overlapping another network, try locating an area of the spectrum that is free of other networks. Then use the access point's web interface to change the channel.

Open a web browser and enter 192.168.4.1 as the URL. You may need to dismiss a security warning to proceed. Log in using:

- Username: admin
- Password: cheetah

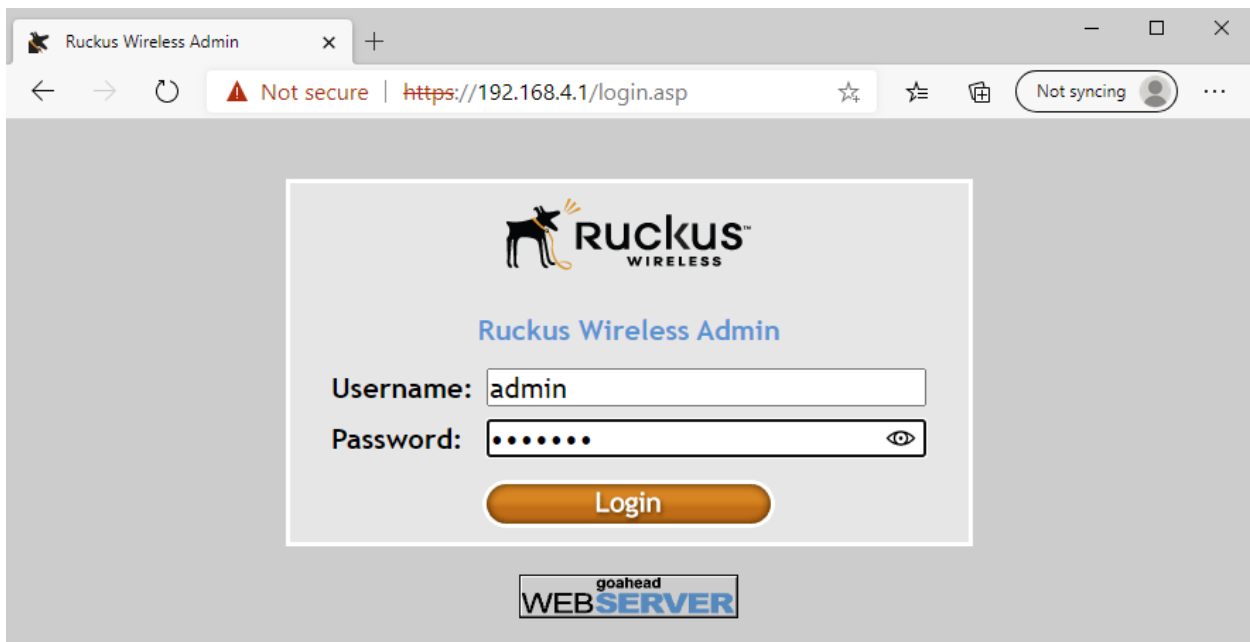


Figure 5-10 Login Page using Microsoft Edge

Under the “Configuration” list, click “Radio 5G”. Select the clearest channel you found using your network analyzer and then click the “Update Settings” button.

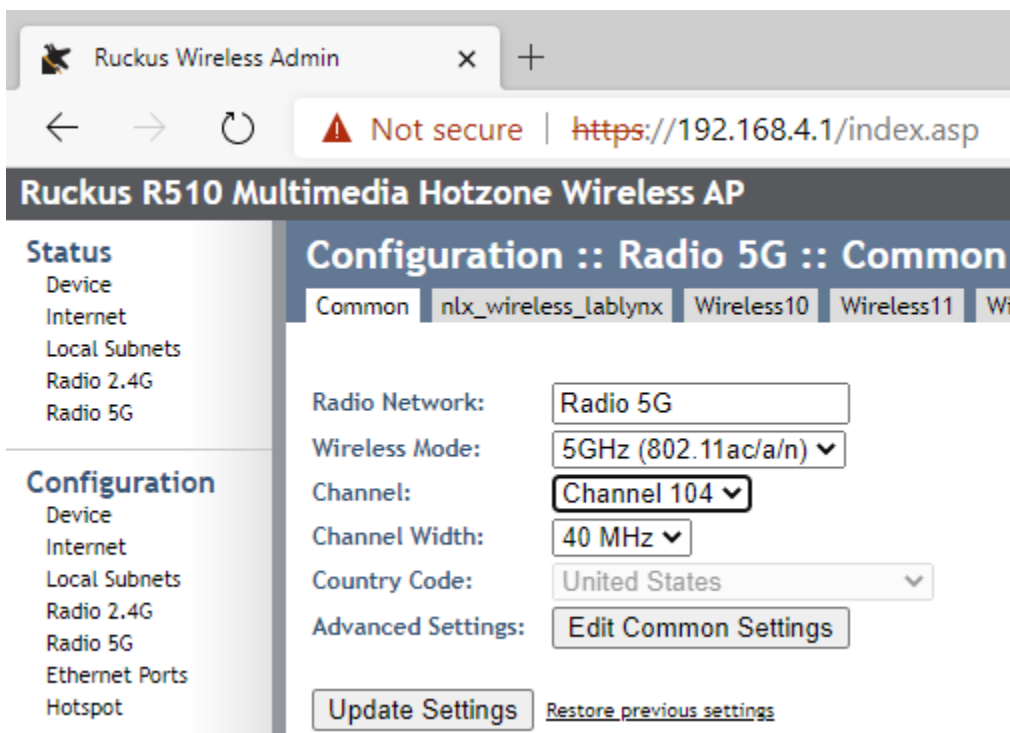


Figure 5-11 Changing WiFi Channel

6 Configuring TTLs

LabLynx can output TTL signals and read TTL inputs. TTL input/output data is recorded along with the neural data at a sample rate of 30 kHz. This TTL functionality is designed to provide basic experiment control functionality with the lowest latency possible.

The LabLynx has up to 8 TTL-compatible IO pins which can be configured to receive input of operant chamber triggering devices such as food dispensers, beam breaks, etc. IO pins can also be configured as outputs; however, outputs controlled from PC software will have a latency due to the network communication delay.

7 LabLynx Commands

Commands can be sent to the LabLynx through Cheetah or NetCom to configure and query the device. Commands can be sent by opening .cfg files containing commands or by using the Neuralynx Netcom API.

Each command includes example text for executing the command within a .cfg file.

Notes

- Cheetah does not currently provide a way to display the return results of any commands except those already available in the Cheetah GUI.
- Each command starts with 4-byte header
 - Byte 1 – packet ID, always equal to x01
 - Bytes 2-3 – command size in bytes, equal to the number of bytes in the command that follow the command size
 - Byte 4 – command ID, different for each command
- The values within quotes use hexadecimal notation. Two hexadecimal characters make one byte.
- Each command example includes a commented-out line labeled “list of command values.” This is a space delimited list of the command’s header values and parameters. This list is shown to make it easier to compose the complete string of command bytes.

7.1 ***SetDigitalIOPortDirection***

7.1.1 Example .cfg File Text

```
# packet ID:      x01
# size:           x0002
# command ID:     x05
# parameter 1:    8-bit ASCII mask (1 byte)

# parameter 2:
# 1 for input, 0 for output
# default is all inputs = xFF
# Upper 4 bits are for base GPIO: bits 4-7 for base GPIO 0-3
# Lower 4 bits are for AFE GPIO: bits 0-3 for AFE GPIO 0-3

#Examples

# 00000000, all outputs
# -SendBinaryCommand AcqSystem1 "0100020500"

# 11110000, base GPIO are inputs, AFE GPIO are outputs
# -SendBinaryCommand AcqSystem1 "01000205F0"

# 00001111, base GPIO are outputs, AFE GPIO are inputs
# -SendBinaryCommand AcqSystem1 "010002050F"

# 11111111, all inputs (default)
# list of command values:      01 0002 05 FF
-SendBinaryCommand AcqSystem1 "01000205FF"
```

7.2 SetDigitalIOBit

7.2.1 Example .cfg File Text

```
# packet ID:      x01
# size:           x0003
# command ID:     x06
# parameter 1:    bit number 0-7 (1 byte)
# parameter 2:    bit value 0 or 1 (1 byte)

# Upper 4 bits on 24-pin connector: bits 4-7 for GPIO 0-3 on 24-pin
# Lower 4 bits on 10-pin connector: bits 0-3 for GPIO 0-3 on 10-pin

# Examples

# list of command values:      01 0003 06 00 00
-SendBinaryCommand AcqSystem1 "010003060000"      # 24-pin GPIO 0 = 0

-SendBinaryCommand AcqSystem1 "010003060100"      # 24-pin GPIO 1 = 0
-SendBinaryCommand AcqSystem1 "010003060200"      # 24-pin GPIO 2 = 0
-SendBinaryCommand AcqSystem1 "010003060300"      # 24-pin GPIO 3 = 0
-SendBinaryCommand AcqSystem1 "010003060400"      # 10-pin GPIO 0 = 0
-SendBinaryCommand AcqSystem1 "010003060500"      # 10-pin GPIO 1 = 0
-SendBinaryCommand AcqSystem1 "010003060600"      # 10-pin GPIO 2 = 0
-SendBinaryCommand AcqSystem1 "010003060700"      # 10-pin GPIO 3 = 0

# -SendBinaryCommand AcqSystem1 "010003060001"      # 24-pin GPIO 0 = 1
# -SendBinaryCommand AcqSystem1 "010003060101"      # 24-pin GPIO 1 = 1
# -SendBinaryCommand AcqSystem1 "010003060201"      # 24-pin GPIO 2 = 1
# -SendBinaryCommand AcqSystem1 "010003060301"      # 24-pin GPIO 3 = 1
# -SendBinaryCommand AcqSystem1 "010003060401"      # 10-pin GPIO 0 = 1
# -SendBinaryCommand AcqSystem1 "010003060501"      # 10-pin GPIO 1 = 1
# -SendBinaryCommand AcqSystem1 "010003060601"      # 10-pin GPIO 2 = 1
# -SendBinaryCommand AcqSystem1 "010003060701"      # 10-pin GPIO 3 = 1
```

7.3 *GetDigitalIOPortValue*

7.3.1 Example .cfg File Text

```
# packet ID:      x01
# size:           x0001
# command ID:     x22
#
# return:         8 bits for 8 digital IO (1 byte)
```

Examples

```
# list of command values:      01 0001 22
-SendBinaryCommand AcqSystem1 "01000122"
```

7.4 *SetDigitalIOPortValue*

7.4.1 Example .cfg File Text

```
# packet ID:      x01
# size:           x0002
# command ID:     x07
# parameter 1:    8-bit value to simultaneously set all 4 GPIO on 24-pin
# connector and all 4 GPIO on 10-pin connector (1 byte)
```

```
# Upper 4 bits on 24-pin connector: bits 4-7 for GPIO 0-3 on 24-pin
# Lower 4 bits on 10-pin connector: bits 0-3 for GPIO 0-3 on 10-pin
```

Examples

```
# list of command values:      01 0002 07 00
-SendBinaryCommand AcqSystem1 "0100020700"      # value 0

# -SendBinaryCommand AcqSystem1 "0100020701"      # value 1
# -SendBinaryCommand AcqSystem1 "01000207FF"      # value 255
```

7.5 *GetTransmittedChannels*

7.5.1 Example .cfg File Text

```
# packet ID:      x01
# size:           x0001
# command ID:     x1B
#
# return:         Each hexadecimal character represents 4 bits
#                 256-bit mask for 256-ch (32 bytes)
```

Examples

```
# list of command values:      01 0001 1B
-SendBinaryCommand AcqSystem1 "0100011B"
```

-SendBinaryCommand <Hardware Sub System Name> “0100211A<Mask>”

Default: The default value of a channel mask is all F which means all channels are on.

Channels 1-64 come from Connector 1 and are highlighted in blue below
Channels 65-128 come from Connector 2 and are highlighted in red below
Channels 129-192 come from Connector 3 and are highlighted in green below
Channels 193-256 come from Connector 4 and are highlighted in purple below

Channel 1-32 (connector 1 channels 0-31) are on
Channel 33-64 (connector 1 channels 32-63) are off
Channel 65-96 (connector 2 channels 0-31) are on
Channel 97-256 are off



Arguments:																																									
Hardware Subsystem Name	Name of sub system that will be controlled.																																								
Mask	<p>To create a mask, start with a binary value of 1s and 0s: 1 for channel on 0 for channel off Highlight color corresponds to connector color on the LabLynx.</p> <p>Example 1: Four HS-64-mux headstages</p> <table><tr><td>Connector 1 channels 1-32:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 1 channels 33-64:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 2 channels 65-96:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 2 channels 97-128:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 3 channels 129-160:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 3 channels 161-192:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 4 channels 193-224:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 4 channels 225-256:</td><td>11111111 11111111 11111111 11111111</td></tr></table> <p>In the above example, all channels are on. Converting the above binary values into hexadecimal values gives the mask that will be included in the command:</p> <table><tr><td>FFFFFFFFFFFFFFFF</td><td>FFFFFFFFFFFFFFFF</td><td>FFFFFFFFFFFFFFFF</td><td>FFFFFFFFFFFFFFFF</td></tr></table> <p>Example 2: Four HS-32-mux headstages</p> <table><tr><td>Connector 1 channels 1-32:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 1 channels 33-64:</td><td>00000000 00000000 00000000 00000000</td></tr><tr><td>Connector 2 channels 65-96:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 2 channels 97-128:</td><td>00000000 00000000 00000000 00000000</td></tr><tr><td>Connector 3 channels 129-160:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 3 channels 161-192:</td><td>00000000 00000000 00000000 00000000</td></tr><tr><td>Connector 4 channels 193-224:</td><td>11111111 11111111 11111111 11111111</td></tr><tr><td>Connector 4 channels 225-256:</td><td>00000000 00000000 00000000 00000000</td></tr></table> <p>Converting the above binary values into hexadecimal values gives the mask that will be included in the command:</p> <table><tr><td>00000000FFFFFFFF</td><td>00000000FFFFFFFF</td><td>00000000FFFFFFFF</td><td>00000000FFFFFFFF</td></tr></table>	Connector 1 channels 1-32:	11111111 11111111 11111111 11111111	Connector 1 channels 33-64:	11111111 11111111 11111111 11111111	Connector 2 channels 65-96:	11111111 11111111 11111111 11111111	Connector 2 channels 97-128:	11111111 11111111 11111111 11111111	Connector 3 channels 129-160:	11111111 11111111 11111111 11111111	Connector 3 channels 161-192:	11111111 11111111 11111111 11111111	Connector 4 channels 193-224:	11111111 11111111 11111111 11111111	Connector 4 channels 225-256:	11111111 11111111 11111111 11111111	FFFFFFFFFFFFFFFF	FFFFFFFFFFFFFFFF	FFFFFFFFFFFFFFFF	FFFFFFFFFFFFFFFF	Connector 1 channels 1-32:	11111111 11111111 11111111 11111111	Connector 1 channels 33-64:	00000000 00000000 00000000 00000000	Connector 2 channels 65-96:	11111111 11111111 11111111 11111111	Connector 2 channels 97-128:	00000000 00000000 00000000 00000000	Connector 3 channels 129-160:	11111111 11111111 11111111 11111111	Connector 3 channels 161-192:	00000000 00000000 00000000 00000000	Connector 4 channels 193-224:	11111111 11111111 11111111 11111111	Connector 4 channels 225-256:	00000000 00000000 00000000 00000000	00000000FFFFFFFF	00000000FFFFFFFF	00000000FFFFFFFF	00000000FFFFFFFF
	Connector 1 channels 1-32:	11111111 11111111 11111111 11111111																																							
	Connector 1 channels 33-64:	11111111 11111111 11111111 11111111																																							
	Connector 2 channels 65-96:	11111111 11111111 11111111 11111111																																							
	Connector 2 channels 97-128:	11111111 11111111 11111111 11111111																																							
Connector 3 channels 129-160:	11111111 11111111 11111111 11111111																																								
Connector 3 channels 161-192:	11111111 11111111 11111111 11111111																																								
Connector 4 channels 193-224:	11111111 11111111 11111111 11111111																																								
Connector 4 channels 225-256:	11111111 11111111 11111111 11111111																																								
FFFFFFFFFFFFFFFF	FFFFFFFFFFFFFFFF	FFFFFFFFFFFFFFFF	FFFFFFFFFFFFFFFF																																						
Connector 1 channels 1-32:	11111111 11111111 11111111 11111111																																								
Connector 1 channels 33-64:	00000000 00000000 00000000 00000000																																								
Connector 2 channels 65-96:	11111111 11111111 11111111 11111111																																								
Connector 2 channels 97-128:	00000000 00000000 00000000 00000000																																								
Connector 3 channels 129-160:	11111111 11111111 11111111 11111111																																								
Connector 3 channels 161-192:	00000000 00000000 00000000 00000000																																								
Connector 4 channels 193-224:	11111111 11111111 11111111 11111111																																								
Connector 4 channels 225-256:	00000000 00000000 00000000 00000000																																								
00000000FFFFFFFF	00000000FFFFFFFF	00000000FFFFFFFF	00000000FFFFFFFF																																						

7.6.1 Example .cfg File Text

```
# packet ID:      x01
# size:          x0021 for 256-ch (33 bytes)
# command ID:    x1A
# parameter1:    Each hexadecimal character represents 4 bits
#               256-bit mask for 256-ch (64 characters or 32 bytes)

# Examples

# 256-ch
# all channels enabled
# list of command values:
# 01 0021 1A FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
-SendBinaryCommand AcqSystem1
"0100211AFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF"
# The above command must be on one line

# only the first channel enabled
# list of command values:
# 01 0021 1A 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000001
-SendBinaryCommand AcqSystem1
"0100211A0000000000000000000000000000000000000000000000000000000000000001"
# The above command must be on one line
```

8 Neuralynx Contact

To purchase the LabLynx, contact sales@neuralynx.com. For technical support, contact support@neuralynx.com.