NeuraLynx

advanced electrophysiology solutions


# FreeLynx

## User Manual

### Wireless Recording for up to 256 Channels


105 Commercial Drive, Bozeman, MT 59715
Phone 406.585.4542 • Fax 866.585.1743
Neuralynx.com
support@Neuralynx.com

Revision 1.5
2/18/2021

# Table of Contents

# List of Figures and Tables

# 1 Document Overview

This document describes the specifications and features of the FreeLynx (formerly Cube2). It also explains how to set up, test and use your FreeLynx. There is a glossary at the end of the document.

# 2 FreeLynx Product Overview

The FreeLynx is a wireless headstage that records from up to 256 individual electrodes. The physiological signals are digitized at the headstage and either transmitted wirelessly to Cheetah or recorded to a microSD memory card in the FreeLynx.

The FreeLynx can be used without a Digital Lynx SX (Connect-to-Cheetah) or with a Digital Lynx SX (Connect-to-SX).

**Features**

- Wirelessly transmit up to 256 channels with one FreeLynx in the Connect-to-Cheetah configuration
- Wirelessly transmit up to 128 channels per FreeLynx in the Connect-to-SX configuration. Connect-to-SX supports two FreeLynxes at once.
- Record up to 256 channels to microSD card with either Connect-to-Cheetah or Connect-to-SX
- The Connect-to-SX configuration allows simultaneous use of a FreeLynx with SX Combo Boards, Mux headstages and other SX features

- Time synchronization with Cheetah to within ±5 ms
- 30 kHz Sampling Rate
- ±5 mV Input Range
- >80 dB Common Mode Rejection Ratio (CMRR) at 60 Hz
- <2.5 $\mu V_{RMS}$ Noise (0.1 Hz to 8 kHz)

- Compatible with Neuralynx EIBs
- Digital IO
- Built-in Inertial Measurement Unit (IMU)
- Support for 32-channel (RHD2132) or 64-channel (RHD2164) electrophysiology amplifier chips

# 3 FreeLynx Components and Accessories

## 3.1 FreeLynx Base

- Included with every FreeLynx
- Interfaces with a variety of AFEs allowing the FreeLynx to be customized to the application
- Transmits data to Cheetah or records data to microSD memory card
- ~21 grams with smallest battery

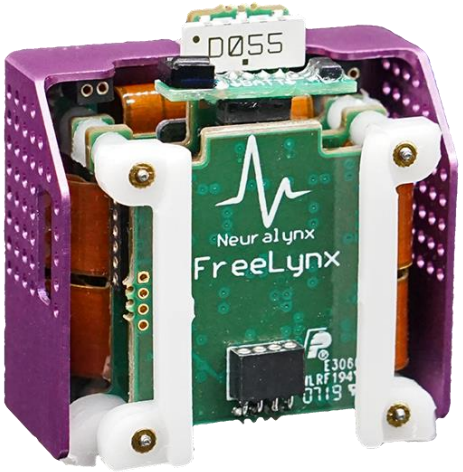



**Figure 3-1 FreeLynx Base**

## 3.2 Analog Front End (AFE)

- 32 to 256 channels of analog to digital acquisition
- Optional 9-axis Inertial Measurement Unit (IMU) with accelerometer, gyroscope, and compass
- Every FreeLynx uses an AFE or LVDS Bridge Board

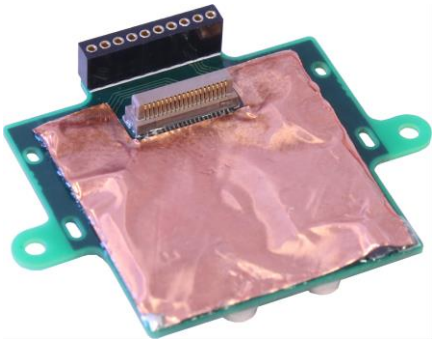


**Figure 3-2 FreeLynx 64 Channel AFE**

## 3.3 256 Channel LVDS Bridge Board

- Adapts FreeLynx to most major neural probes
- Works with Neuralynx mux headstages
- Used in place of an AFE

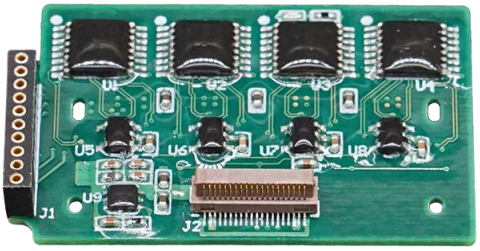



**Figure 3-3 LVDS Bridge Board**

## 3.4 FreeLynx EIB-72-QC-Large

- 64 Channels, 8 References, 2 Stim
- 1 Static Reference for all 32 channels
- For use with Large Gold Pins
- 0-80 Threaded Screw Holes
- 2.0 grams

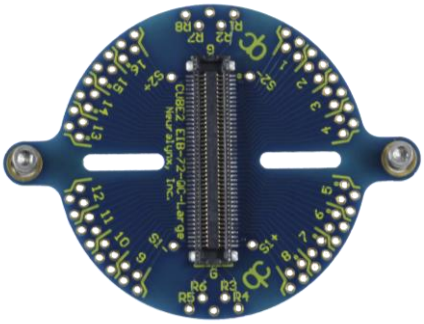


**Figure 3-4 FreeLynx EIB-72-QC Large**

## 3.5 FreeLynx EIB-72-QC-Small

- 64 Channels, 8 References, 2 Stim
- 1 Static Reference for all 32 channels
- For use with Small Gold Pins
- 0-80 Threaded Screw Holes
- 2.0 grams

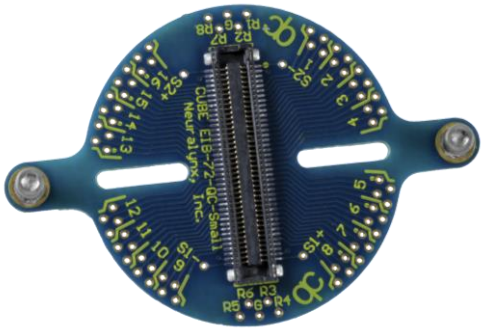


**Figure 3-5 FreeLynx EIB-72-QC Small**

## 3.6 FreeLynx Power Tether

- FreeLynx can be powered with a USB cable
- Standard 2 meter cable

**Figure 3-6 FreeLynx Power Tether**

## 3.7 FreeLynx Battery Charger

- Standard 2 meter USB cable for powering charger
- Green sticker indicates charger with 200 mA charging current
- Blue sticker indicates charger with 800 mA charging current


**Figure 3-7 FreeLynx Battery Charger**

## 3.8 FreeLynx Batteries

- Small: 240 mAh, 6.8 grams
- Medium: 650 mAh, 15 grams
- Large: 1300 mAh, 30 grams
- Contact Neuralynx for more battery options

**Figure 3-8 FreeLynx Battery**

## 3.9 Power Stick

- FreeLynx can be turned on or off by waving the magnet at the end of the power stick near the status LED

**Figure 3-9 Power Stick**

## 3.10 IR Transmitter

- Used with IR Receiver to turn the FreeLynx on and off

**Figure 3-10 IR Transmitter**

## 3.11 IR Receiver

- Used with IR Transmitter to turn the FreeLynx on and off


**Figure 3-11 IR Receiver**

## 3.12 SM-72-QC

- Interface for driving test signals into the FreeLynx
- Switches control Bank 1, Bank 2, Bank 3, Bank 4, and the Reference

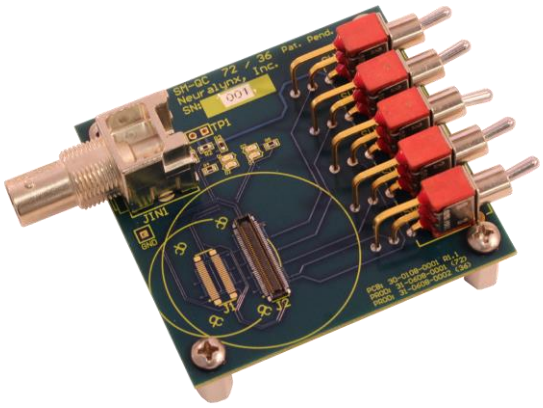



**Figure 3-12 SM-72-QC**

## 3.13 FreeLynx Training Weight

- Approximates the shape, weight, and balance of a FreeLynx

**Figure 3-13 FreeLynx Training Weight**

## 3.14 Video Tracking LEDs

- Available in red, blue, and green

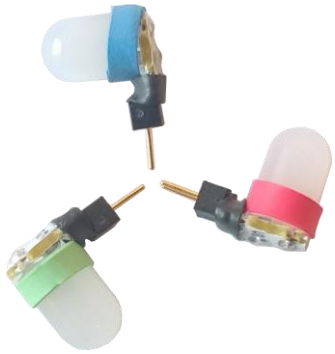


**Figure 3-14 Video Tracking LEDs**

## 3.15 Ultrasonic Microphone

- Low noise, omnidirectional MEMS microphone
- Sampled at 210 kHz
- Connects to FreeLynx 6-pin digital IO connector
- Contact Neuralynx for more information

# 4 Setup

## 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 Two Ways of using a FreeLynx

A FreeLynx can be used in two different setups; Connect-to-Cheetah and Connect-to-SX.

1. **Connect-to-Cheetah**
   A single FreeLynx can acquire up to 256 channels and wirelessly transmit the data to Cheetah or record it to an onboard microSD card. In this setup, an Ethernet cable connects the Cheetah PC to the FreeLynx wireless access point.

2. **Connect-to-SX**
   Up to two FreeLynxes can connect to a Digital Lynx SX. Each FreeLynx can wirelessly transmit up to 128 channels or record up to 256 channels to an onboard microSD card. In this setup, an Ethernet cable connects the SX to the FreeLynx wireless access point.

## 4.3 Connect-to-Cheetah

### 4.3.1 FreeLynx and Cheetah Version Compatibility

Use FreeLynx firmware version 2.4.0 or higher and Cheetah version 6.4.1 or higher. If you have an older version of FreeLynx firmware or Cheetah software that you prefer to continue using, contact Neuralynx for information about compatibility.

The Connect-to-Cheetah configuration supports +/- 5 ms time synchronization with Cheetah. Connect-to-SX does not.

For Neuralynx firmware and software, visit www.neuralynx.com or contact Neuralynx support.

### 4.3.2 Hardware Setup

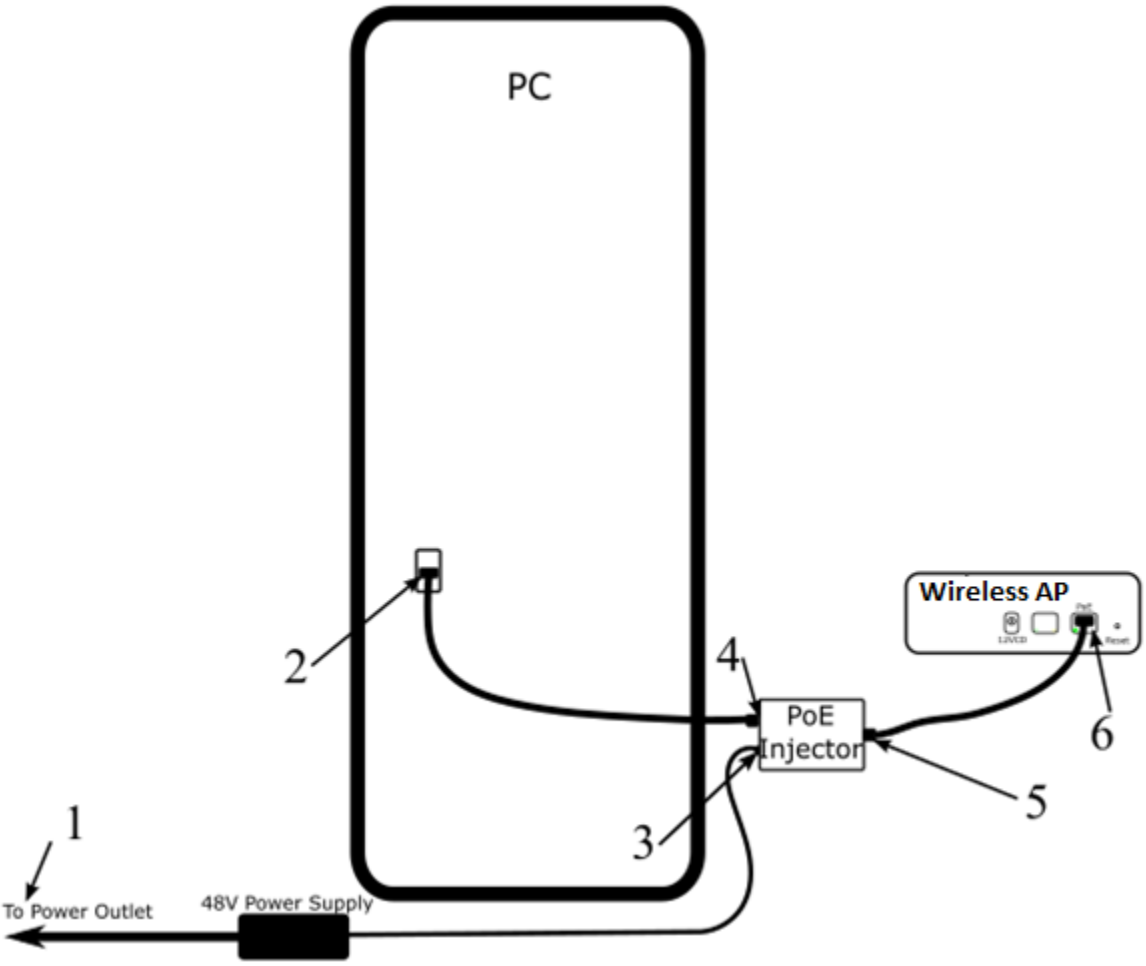



**Figure 4-1 Connect-to-Cheetah Hardware Setup**

1. Connect the 48 V power supply to a standard wall outlet.
2. Connect the first Ethernet cable to the leftmost port on the dedicated network card installed on the PC. This may be an Intel Server Adapter I350 or similar. Contact Neuralynx for questions about network card compatibility.
3. Connect the DC output of the 48 V power supply to the Power over Ethernet (PoE) injector.
4. Connect the other end of the first Ethernet cable to the LAN In Port on the PoE Injector.
5. Connect the second Ethernet cable to the Power+Data Out Port on the PoE Injector.
6. Connect the other end of the second Ethernet cable to the PoE Port on the wireless access point.

### 4.3.3 FreeLynx and Test Input Setup

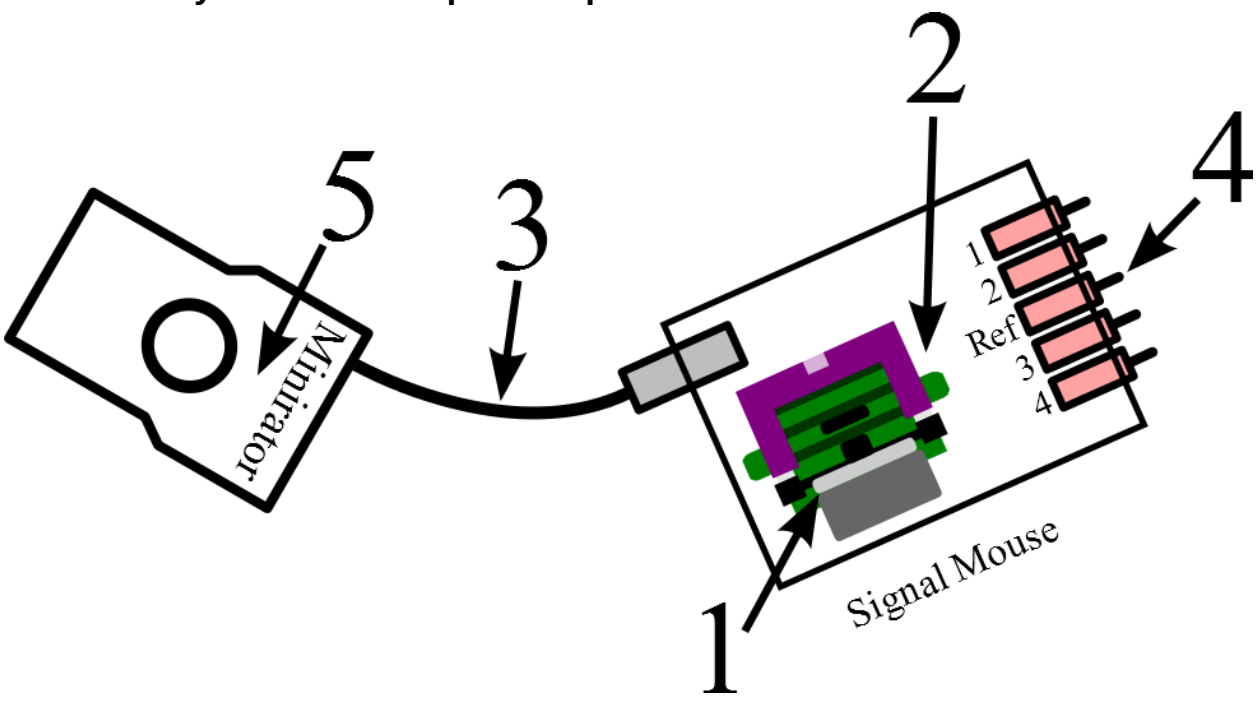



**Figure 4-2 FreeLynx and Test Input Setup**

1. Slide the FreeLynx Battery down the guide rails on the FreeLynx until the battery snaps into place.
2. Connect the FreeLynx to the Signal Mouse.
3. Connect the Minirator, or other signal source, to the Signal Mouse using a BNC Cable.
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. 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 roughly to 1 $mV_{PP}$.

### 4.3.4 Configure Cheetah

Open the Cheetah Configuration folder and modify your preferred configuration file. To use the Connect-to-Cheetah configuration, your config file should include the black text below. Green text indicates a line that is commented out.

######### ACQUISITION SYSTEM SETUP
# Do the basic setup for your acquisition hardware - uncomment one only!

#-ProcessConfigurationFile DigitalLynxSX.cfg
-ProcessConfigurationFile FreeLynx.cfg
#-ProcessConfigurationFile LabLynx.cfg
#-ProcessConfigurationFile RawDataFilePlayback.cfg

Start Cheetah and select your configuration file. The example below uses 64 CSCs. The default 64 CSC configuration file from Neuralynx can be used by including the black text below:

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

Wait until the Clock Status indicator is green before starting acquisition. The FreeLynx will not transmit data until it has synchronized its clock with Cheetah. Time synchronization with Cheetah is only supported in the Connect-to-Cheetah setup.

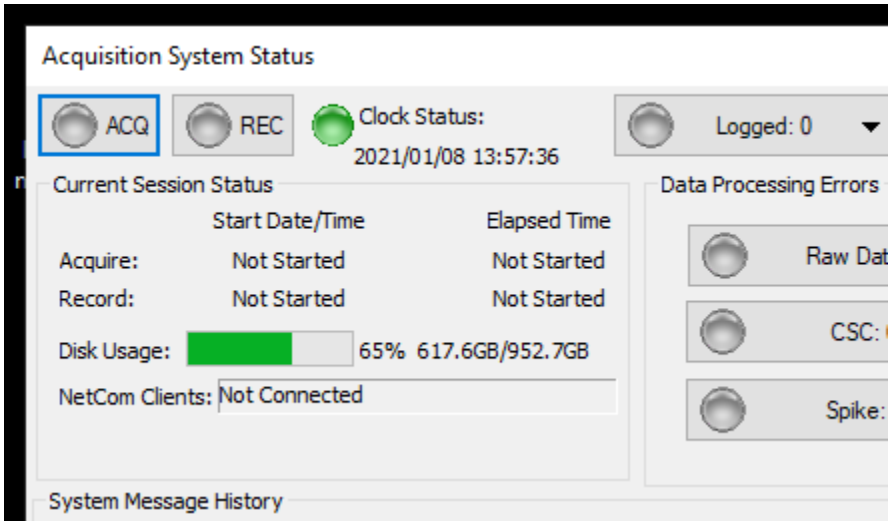



**Figure 4-3 Clock Status Indicator**

To test the FreeLynx, go to ***4.5 Test the FreeLynx with a Test Input Signal***.

### 4.3.5 FreeLynx GUI in Cheetah

When using the FreeLynx in the Connect-to-Cheetah configuration, there is a FreeLynx GUI available in Cheetah under View->Hardware Properties. It allows the use of a small subset of available commands and provides some status information such as temperature, battery level and remaining microSD card memory. See ***9 FreeLynx Commands*** for the full set of commands.

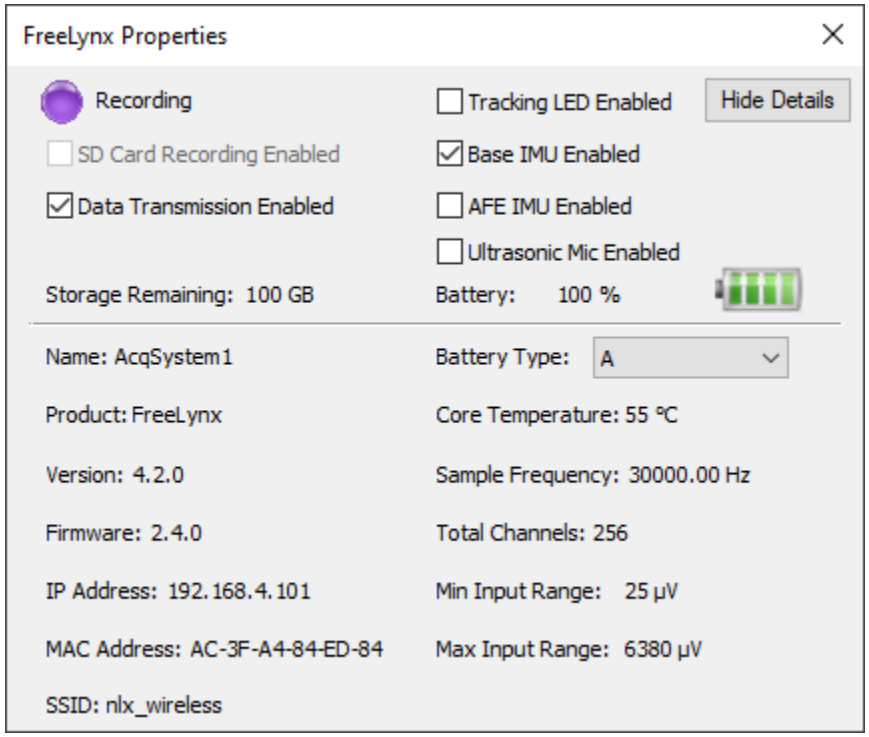



**Figure 4-4 FreeLynx GUI**

## *4.4 Connect-to-SX*

The FreeLynx communicates with the Digital Lynx SX in a different way than a standard Neuralynx analog headstage or multiplexing headstage. Instead of connecting to the SX with a cable, the FreeLynx communicates with the Digital Lynx SX via a wireless access point.

Some older models of Digital Lynx SX do not have the Ethernet connection needed to use a FreeLynx in the Connect-to-SX configuration. If your Digital Lynx SX requires a physical upgrade for compatibility with the FreeLynx, please contact Neuralynx.

### 4.4.1 FreeLynx, Digital Lynx SX and Cheetah Version Compatibility

Use FreeLynx firmware version 2.4.0 or higher, SX firmware 2.3.0 or higher and Cheetah version 6.4.1 or higher. If you have an older version of firmware or Cheetah software that you prefer to continue using, contact Neuralynx for information about compatibility.

Visit www.neuralynx.com to download Cheetah software. After installing Cheetah, the most recent versions of FreeLynx and SX firmware can be found in the *Resources* subfolder of the Cheetah installation. If you need assistance with software or firmware installation, contact Neuralynx support.

### 4.4.2 Hardware Setup

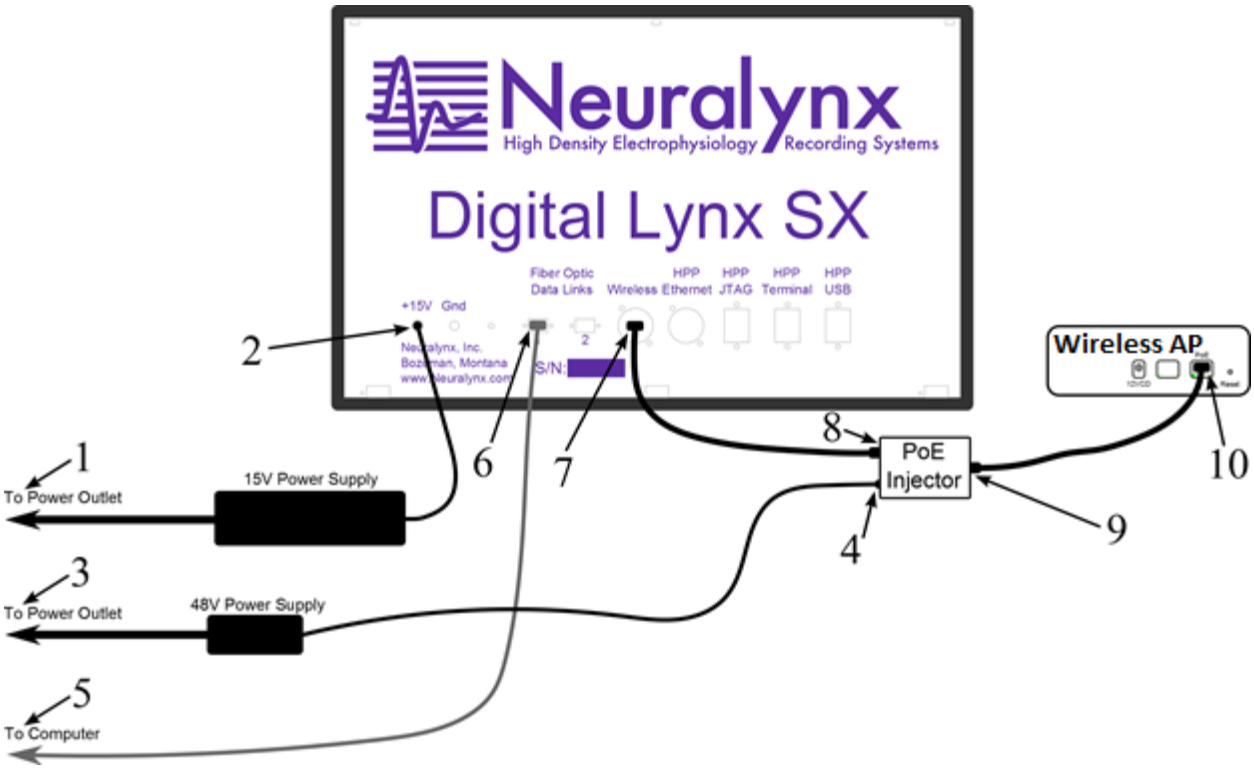



**Figure 4-5 Connect-to-SX Hardware Setup**

1. Connect the 15 V power supply to a standard wall outlet.
2. Connect the DC output of the 15 V power supply to the Digital Lynx SX.
3. Connect the 48 V power supply to a standard wall outlet.
4. Connect the DC output of the 48 V power supply to the Power over Ethernet (PoE) injector.
5. Connect the fiber optic cable to the leftmost fiber optic port on the back of the computer.
6. Connect the other end of the fiber optic cable to Fiber Optic Data Link 1 on the back of the Digital Lynx SX.
7. Connect the first Ethernet cable to the Wireless port on the back of the Digital Lynx SX.
8. Connect the other end of the first Ethernet cable to the LAN In Port on the PoE Injector
9. Connect the second Ethernet cable to the Power+Data Out Port on the PoE Injector.
10. Connect the other end of the second Ethernet cable to the PoE Port on the wireless access point.

### 4.4.3 FreeLynx and Test Input Setup

These connections are the same as in the Connect-to-Cheetah configuration described in ***4.3.3 FreeLynx and Test Input Setup***.

### 4.4.4 Configure Cheetah

In Digital Lynx SX systems that also contain Input Boards (or Combo Boards) and mux headstages, the AD channels associated with the FreeLynx begin after the last mux headstage AD channel. This is illustrated in the table below. This table assumes one HS-64-mux and one 64-channel FreeLynx is being used. The maximum channel count for a Digital Lynx SX is 512 channels.

| Number of 32-ch Input Boards (or Combo Boards) | Number of 64-ch mux Headstages | 64-ch FreeLynx Starting AD Channel | Total Channel Count |
|---|---|---|---|
| 0 | 0 | **0** | 0 |
| 1 | 0 | **32** | 96 |
| 2 | 0 | **64** | 128 |
| 3 | 0 | **96** | 160 |
| 0 | 1 | **64** | 128 |
| 1 | 1 | **96** | 160 |
| 2 | 1 | **128** | 192 |
| 3 | 1 | **160** | 224 |

**Figure 4-6 FreeLynx Starting AD Channel**

Turn on the Digital Lynx SX and wait for the status LED to remain green. Open the Cheetah Configuration folder and modify your preferred configuration file to include your FreeLynx AD Channels. To use the Connect-to-SX configuration, your config file should include the black text below. Green text indicates a line that is commented out.

```
######### ACQUISITION SYSTEM SETUP
# Do the basic setup for your acquisition hardware - uncomment one only!

    -ProcessConfigurationFile DigitalLynxSX.cfg
    #-ProcessConfigurationFile FreeLynx.cfg
    #-ProcessConfigurationFile LabLynx.cfg
    #-ProcessConfigurationFile RawDataFilePlayback.cfg
```

Start Cheetah and select your configuration file. The example below uses 64 CSCs. The default 64 CSC configuration file from Neuralynx can be used by including the black text below:

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

### 4.4.5 Pair the FreeLynx with the Digital Lynx SX

If you received your FreeLynx and Digital Lynx SX together, the FreeLynx is already paired with the SX. If you ordered the FreeLynx and Digital Lynx SX at different times, they need to be paired before continuing. The pairing process is described in ***Section 7 Pairing a FreeLynx with a Digital Lynx SX***.

## 4.5 Test the FreeLynx with a Test Input Signal

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

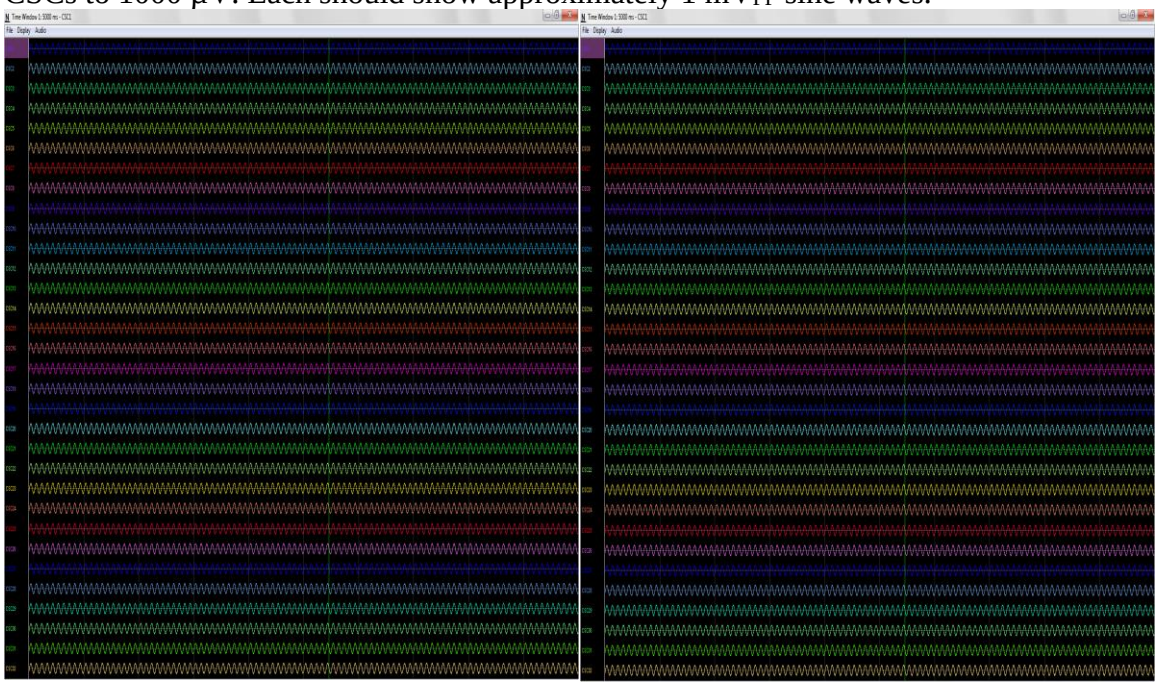


**Figure 4-7 Bank 1, 2, 3 and 4 Switches Up. Reference Switch Down**

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.

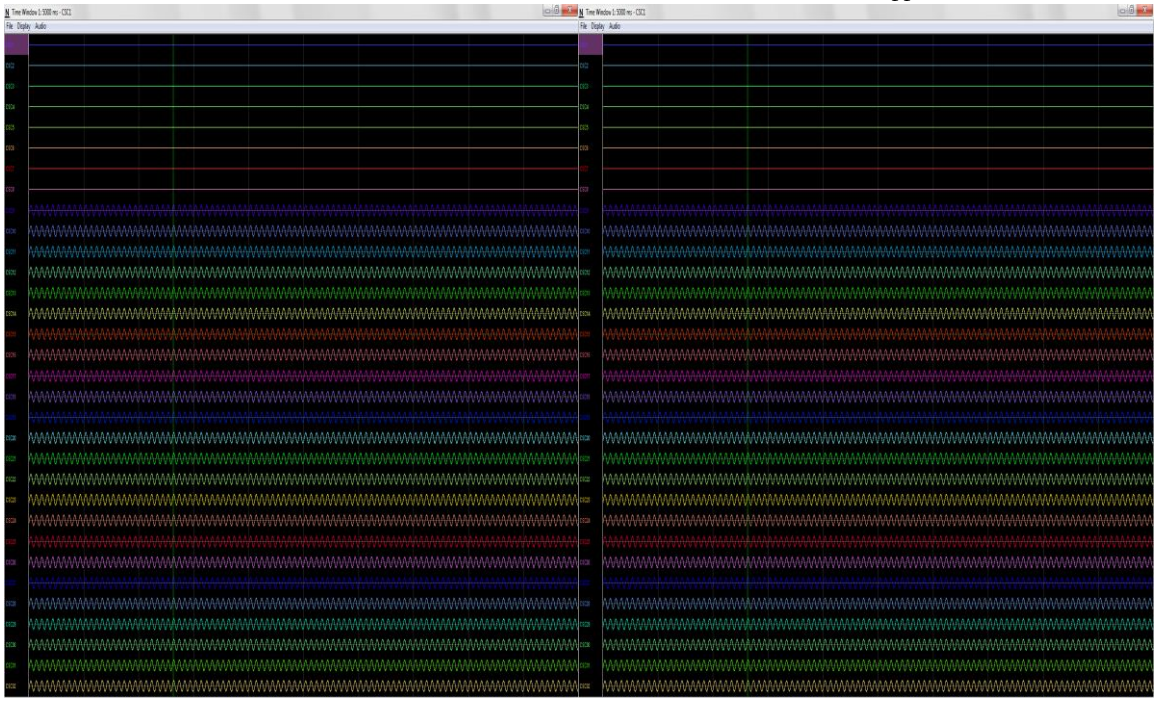


**Figure 4-8 Bank 2, 3 and 4 Switches Up. Bank 1 and Reference Switches Down**

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.

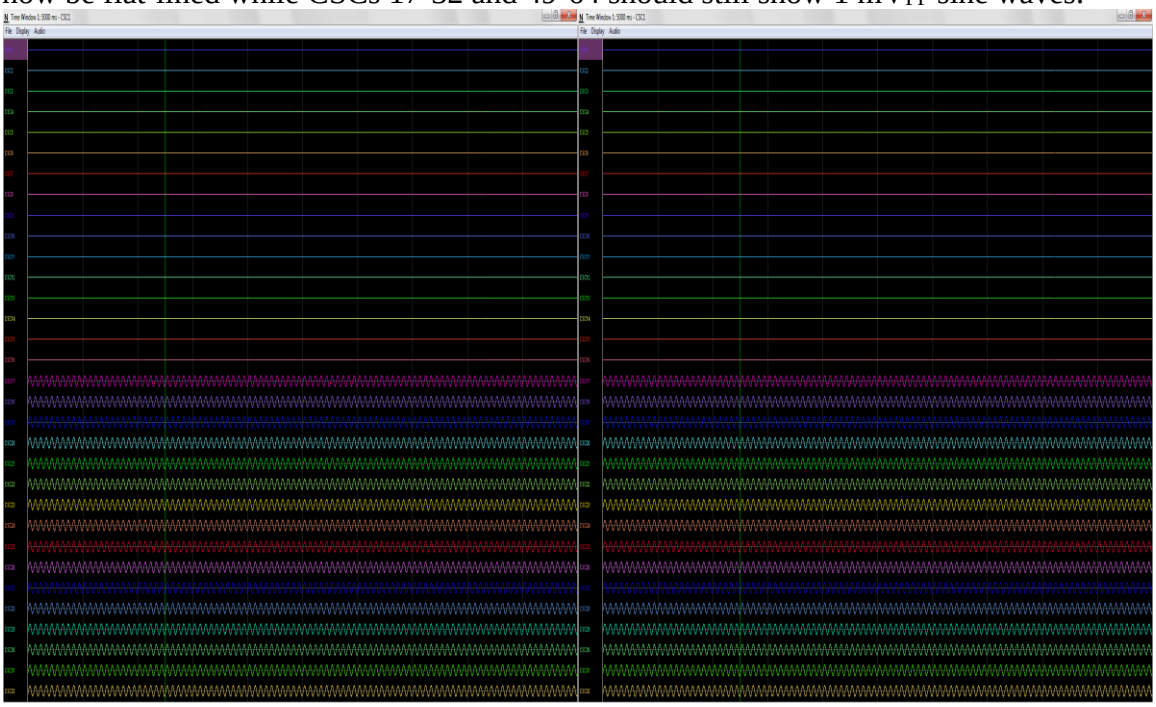


**Figure 4-9 Bank 3 and 4 Switches Up. Bank 1, 2 and Reference Switches Down**

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.

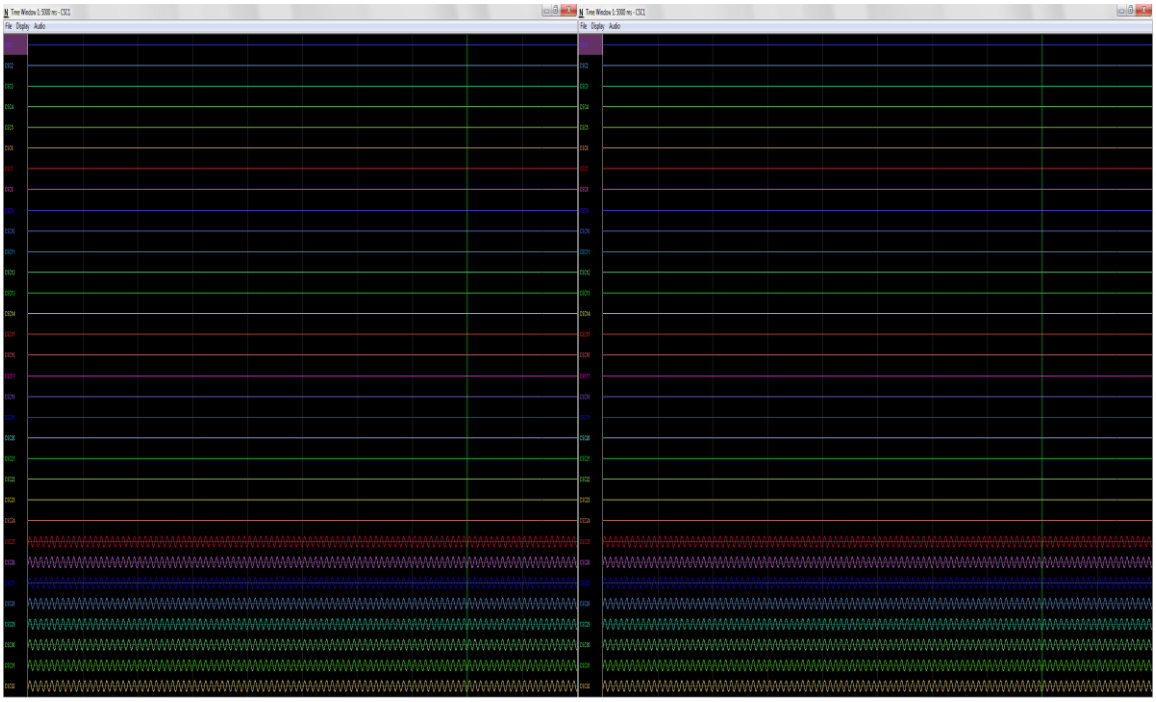


**Figure 4-10 Bank 4 Switches Up. Bank 1, 2, 3 and Reference Switches Down**

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

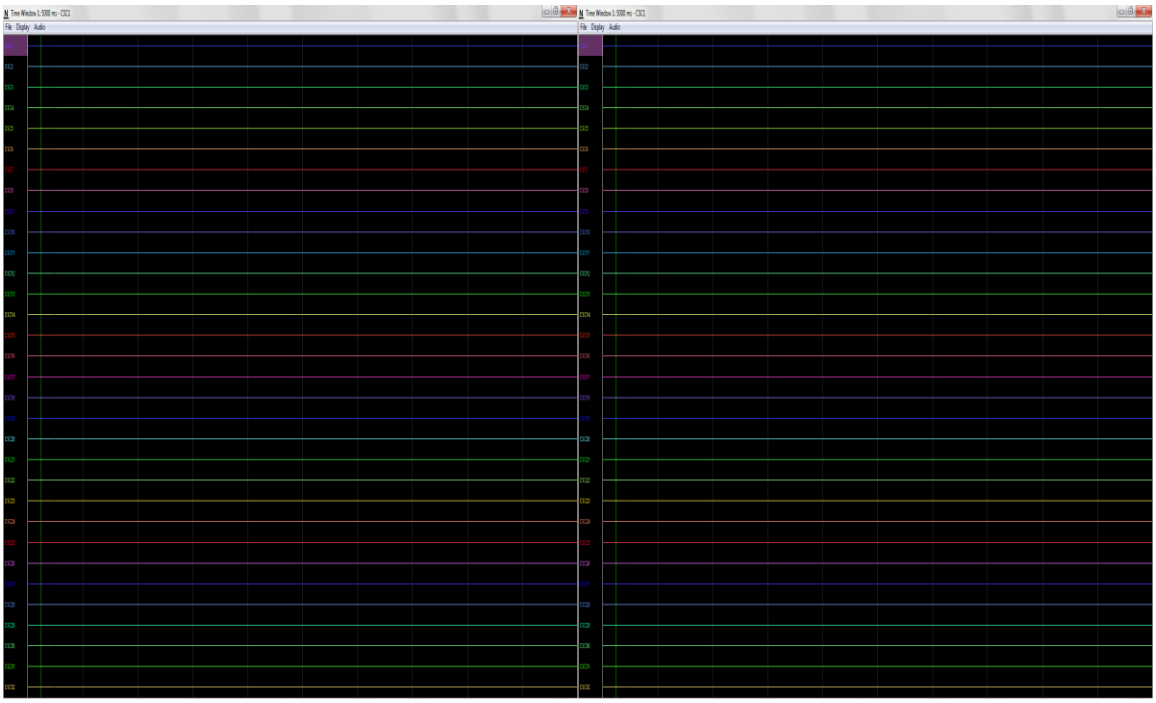


**Figure 4-11 All Switches Down (1000 µV Input Range)**

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

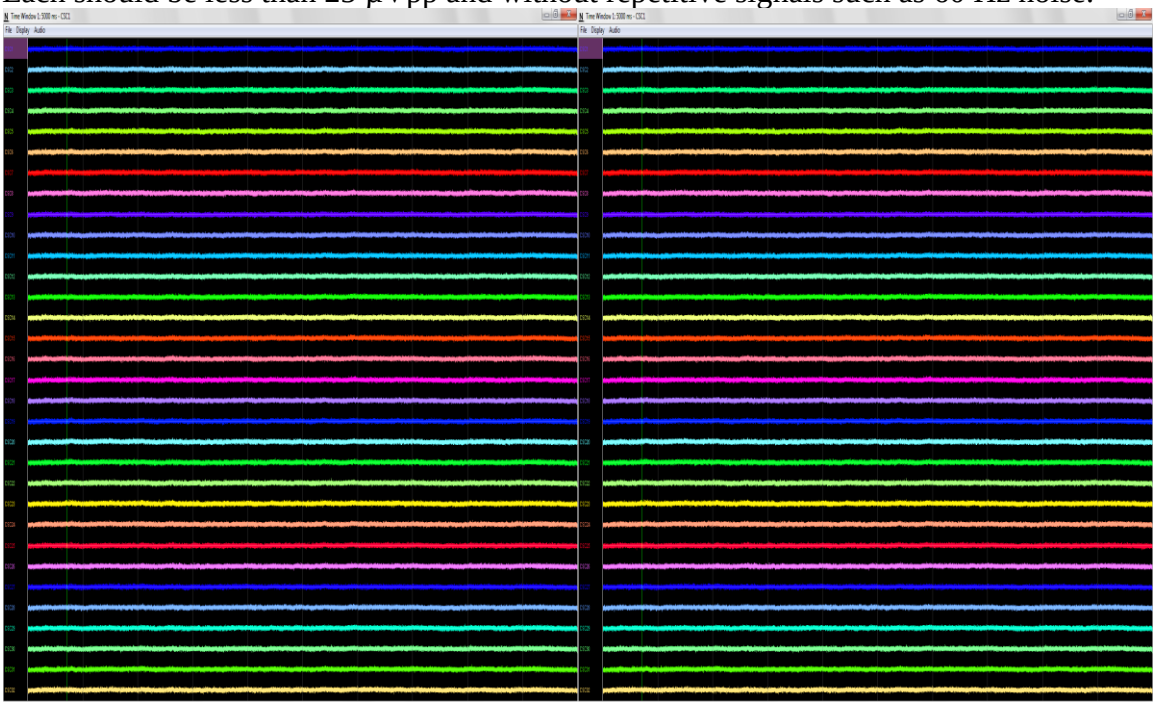


**Figure 4-12 All Switches Down (25 µV Input Range)**

Switch all the switches to the Signal position (up). Each should be less than 25 µVpp and without repetitive signals such as 60 Hz noise.

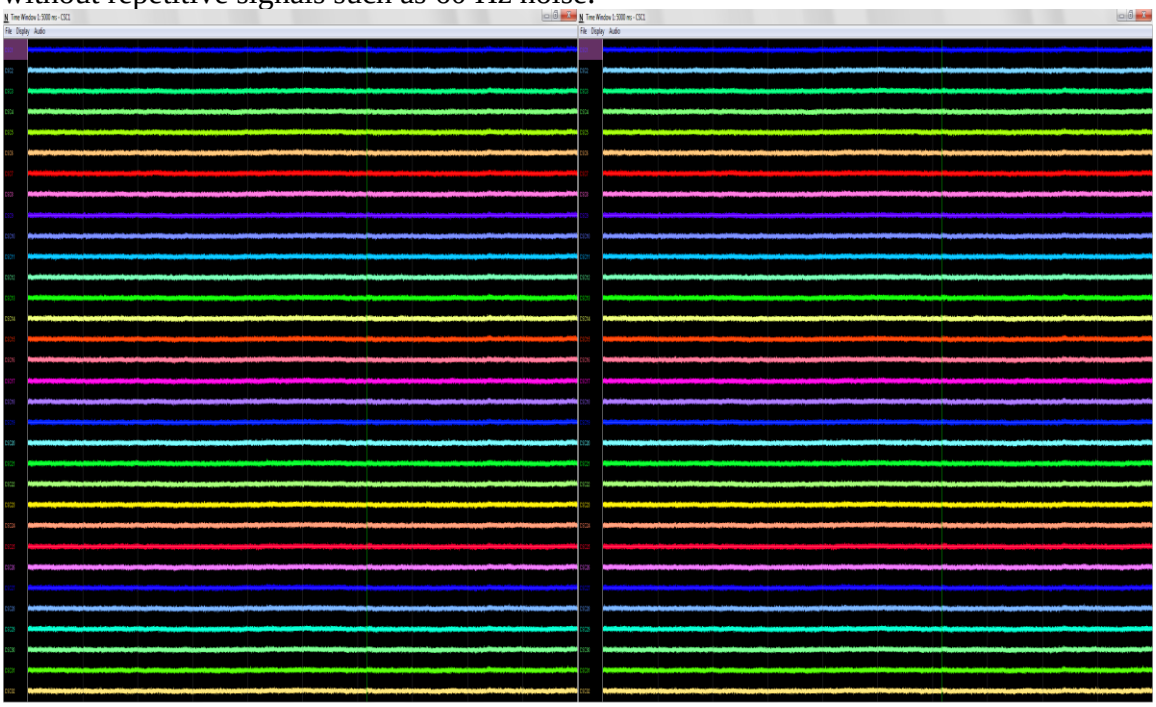


**Figure 4-13 All Switches Up (25 µV Input Range)**

## *4.6 Try a FreeLynx Command by Controlling Tracking LEDs*

The FreeLynx supports two video tracking LEDs. By default, these LEDs are off when the FreeLynx is booted. The video tracking LEDs are controlled by sending commands to the FreeLynx. When the video tracking LEDs are turned on, the FreeLynx status LED is turned off so it will not interfere with video tracking.

The way commands are sent to a FreeLynx depends on whether you are using the Connect-to-Cheetah or Connect-to-SX setup.

### 4.6.1 Connect-to-Cheetah Command

Control of video tracking LEDs is available in the Cheetah GUI under View->Hardware Properties. See ***4.3.5 FreeLynx GUI in Cheetah***. The video tracking command can also be sent using a Cheetah .cfg file or Netcom.

### 4.6.2 Connect-to-SX Command

Digital Lynx SX which forwards the commands to the FreeLynx. FreeLynx commands can be sent through a NetCom interface or through a Cheetah configuration file. All FreeLynx commands and their syntax are discussed in ***9 FreeLynx Commands***. To begin, connect a set of Video Tracking LEDs to the FreeLynx.

1. Connect a set of Video Tracking LEDs to the FreeLynx.

In this example we will use a Cheetah Configuration File to enable the Video Tracking LEDs Ears.

2. Enable the Video Tracking LEDs on FreeLynx #1 using a .cfg command.

   Using the command information in ***9.5 SetTrackingLED****, the following configuration file can be created.

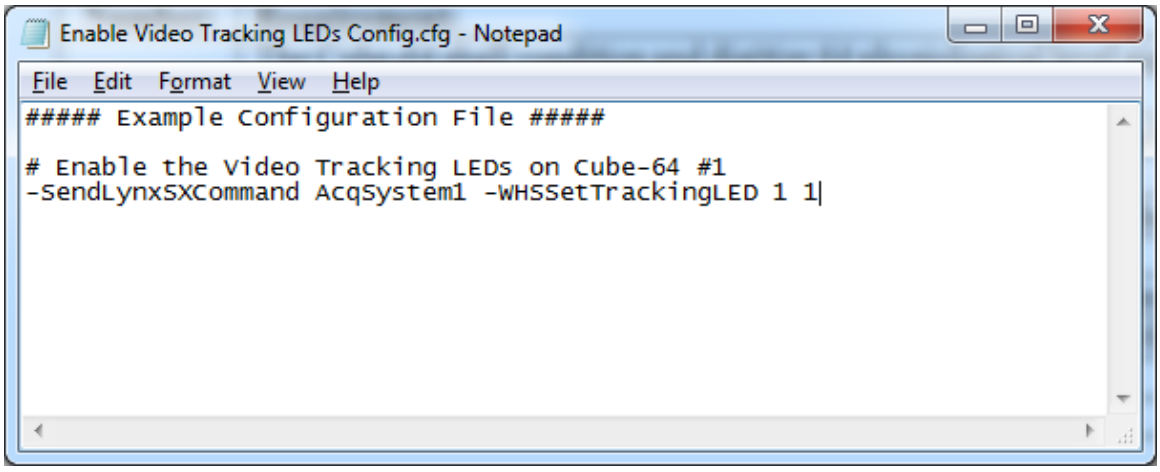



**Figure 4-14 Example Configuration File to Enable Tracking LEDs**

Once the configuration file has been created and saved it can be run in Cheetah by selecting *File -> Open Configuration File*. Browse to the configuration file and select *Open*. The configuration file will immediately execute, enabling the Video Tracking LED Ears. Refer to the figure below. Enabling the tracking LEDs will turn off the FreeLynx's status LED. Disabling the tracking LEDs will turn the status LED back on.

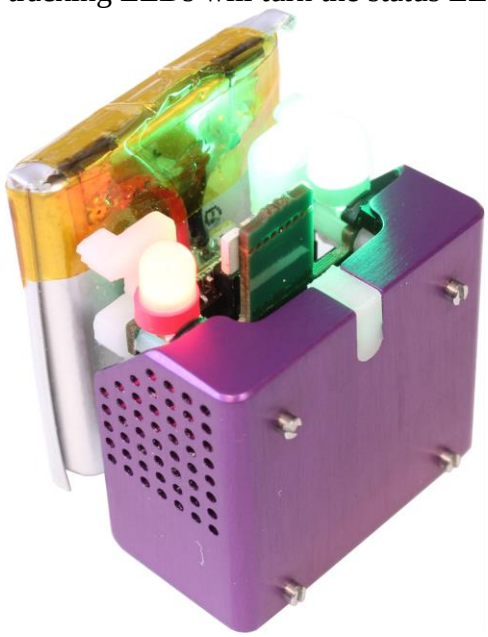


**Figure 4-15 Enabled Video Tracking LEDs**

When the Video Tracking LED Ears are not needed, they should be disabled. Once again this can be done with a Cheetah Configuration File.

3. Disable the Video Tracking LEDs on FreeLynx #1 using a .cfg command.

Using the command information in ***9.5 SetTrackingLED****, the following configuration file can be created.

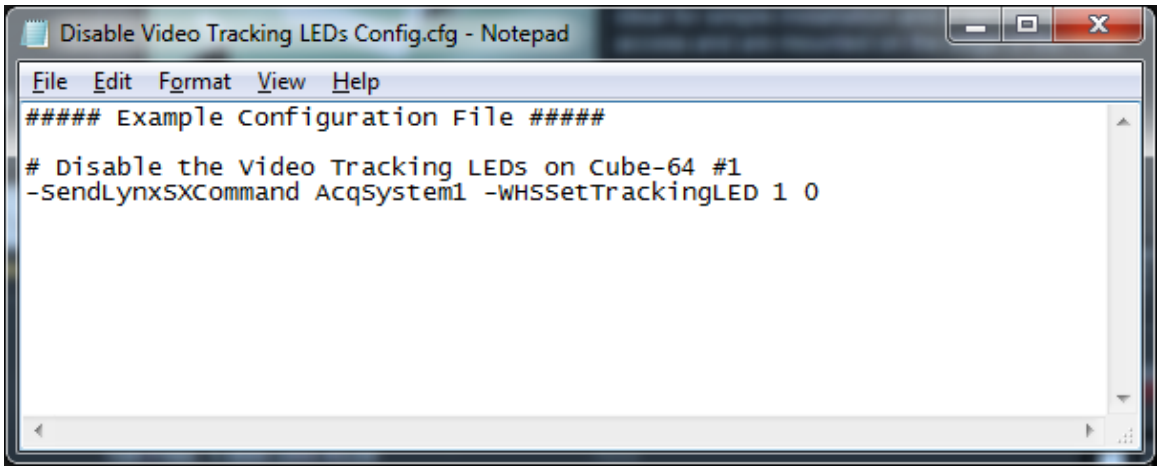



**Figure 4-16 Example Configuration File to Disable Tracking LEDs**

# 5 Recording Data to microSD Card

The FreeLynx can record up to 256 channels of data to microSD card in both the Connect-to-SX and Connect-to-Cheetah setups.

The FreeLynx does not support simultaneous Wi-Fi data recording and microSD recording but can be switched between the two modes at any time using commands. When recording data to microSD card, Cheetah does not receive the data, but it does keep a wireless connection to the FreeLynx for commands, control and status information.

Connect-to-SX is limited to 128 channels in the Wi-Fi recording mode. If you plan to use Connect-to-SX to record 256 channels to microSD but will also sometimes switch to Wi-Fi recording, please review ***5.6 Switching between microSD and Wi-Fi Recording in a Connect-to-SX Setup with more than 128 Channels***.

## *5.1 Commands for microSD Recording*

- ***9.57 GetMicroSDrecordEnable****
- ***9.58 SetMicroSDrecordEnable****
- ***9.59 SetMicroSDdataFileName***
- ***9.60 GetMicroSDMemAvailable***
- ***9.61 GetMicroSDRecordingStatus***

## *5.2 Recorded Data Files*

The filename used for microSD data recording is set using the ***SetMicroSDdataFileName*** command. If no name has been given, the system will use default_nnn.crd, where nnn starts at 001 and is incremented with each new file.

New files are created at these times:

1. When acquisition starts
2. When a data file reaches 2GB
3. If microSD recording is enabled while acquisition is enabled

Files close at these times:

1. When acquisition is stopped
2. When a file reaches 2 GB
3. If the microSD card is full
4. If microSD recording is disabled while acquisition is enabled
5. If the FreeLynx turns off after receiving the ***PowerOffSystem*** command
6. If the FreeLynx turns off due to low battery or high temperature

If microSD recording is enabled at the time a battery is depleted, file writing will resume with a new file after the battery is replaced and the device is turned back on. Data files are written to the NlxData partition of the microSD card. This partition is formatted as

Linux ext4. To access this data in Windows, special software is needed. Here are two programs that can be used to access recorded data files from the microSD card.

- Linux Reader by DiskInternals – This is freeware and is available at: https://www.diskinternals.com/linux-reader/

- extFS for Windows by Paragon Software – This software requires a license. Contact Neuralynx for assistance.

## 5.3 Raw Data File Converter

Once you can access the recorded .crd files on your microSD card, you can use the Neuralynx Wireless Raw Data File Converter to convert your .crd files into the same formats output by Cheetah.

The converter can be found on the Neuralynx website or on your PC in the Cheetah installation folder.

## 5.4 Important Notes to Prevent Data Corruption

- When the battery is low, FreeLynx will automatically close and store files and then power off. *Allowing a FreeLynx to power off due to low battery is ok for all hardware versions.*

- Do not remove the battery while the FreeLynx is on.

- Always turn off the FreeLynx using the Power Stick, IR remote, the ***9.8 PowerOffSystem*** command or by allowing the FreeLynx to power itself off due to low battery. *Do not turn off the FreeLynx by removing the battery.*

- With FreeLynx hardware version 4.1.0 or older, always stop acquisition in Cheetah prior to turning off your FreeLynx. This allows FreeLynx to properly close and store the data file. Powering off suddenly may corrupt the data and require reformatting the microSD card. *Allowing a FreeLynx to power off due to low battery is ok even if acquisition is still running.*

- FreeLynx hardware 4.2.0 or newer safely closes and stores the data file if turned off during acquisition as long as the FreeLynx is turned off using the Power Stick, IR remote, the ***9.8 PowerOffSystem*** command or by allowing the FreeLynx to power itself off due to low battery. *Do not turn off the FreeLynx by removing the battery.*

## 5.5 microSD Card Requirements

1. Recording acquired data to the microSD card requires a high-performance microSD card. The card supplied by Neuralynx has been thoroughly tested with the FreeLynx. If you need an additional microSD card or a card with a different memory capacity, please contact Neuralynx for recommendations.

2. Neuralynx currently offers 64 GB and 256 GB capacity microSD cards.

3. For optimal microSD card recording performance, the ext4 NlxData partition should be formatted using 4K clusters prior to a recording session. Contact Neuralynx for details.

## 5.6 Switching between microSD and Wi-Fi Recording in a Connect-to-SX Setup with more than 128 Channels

Because a FreeLynx in the Connect-to-SX setup can record more channels to microSD than it can transmit wirelessly, there are some special considerations.

- If a FreeLynx is powered on with Channel Capacity set to 256 and Network Stream Data Acq Enable set to '1', the FreeLynx will automatically set the channel mask to AD channels 0-127.

- If switching from microSD recording to wireless transmission during operation, the channel mask will automatically be limited to 128 channels.
    - If the mask was set to 129 channels or more at the time of the switch, the FreeLynx will mask in AD channels 0-127.
    - If the mask was set to 128 or less channels at the time of the switch, the FreeLynx will include all masked in channels selected by the user but they will be mapped to AD channel 0 and increment from there. For example, if the user had masked in AD channels 192 to 255 at the time of the switch, these channels would appear in Cheetah on AD channels 0-63.

- For a 256 channel FreeLynx with wireless transmission enabled, any mask that includes 128 channels or less will be mapped the same way as above.
    - All masked in channels selected by the user will be included and will be mapped to AD channel 0 and increment from there. For example, if the user had masked in AD channels 192 to 255, these channels would appear in Cheetah on AD channels 0-63.

- If switching from wireless transmission to microSD recording during operation, all 256 channels are automatically included in the channel mask.

- Use commands ***GetTransmittedChannels*** and ***SetTransmittedChannels*** to manually check or change the channel mask.

# 6 Hardware Overview

## 6.1 Amplifier and Analog to Digital Converter

Analog to digital conversion (ADC) and amplification is done with either 32-channel RHD2132 chips or 64-channel RHD2164 chips. A different version of FreeLynx firmware is required depending on which type of chip you are using. FreeLynx does not support the simultaneous use of both chip types. Contact Neuralynx support for more information.

Each AD channel is digitized on the FreeLynx using a fixed reference. The channels are AC coupled and the gain is fixed at 192 [V/V]. This concept is illustrated in the figure below.

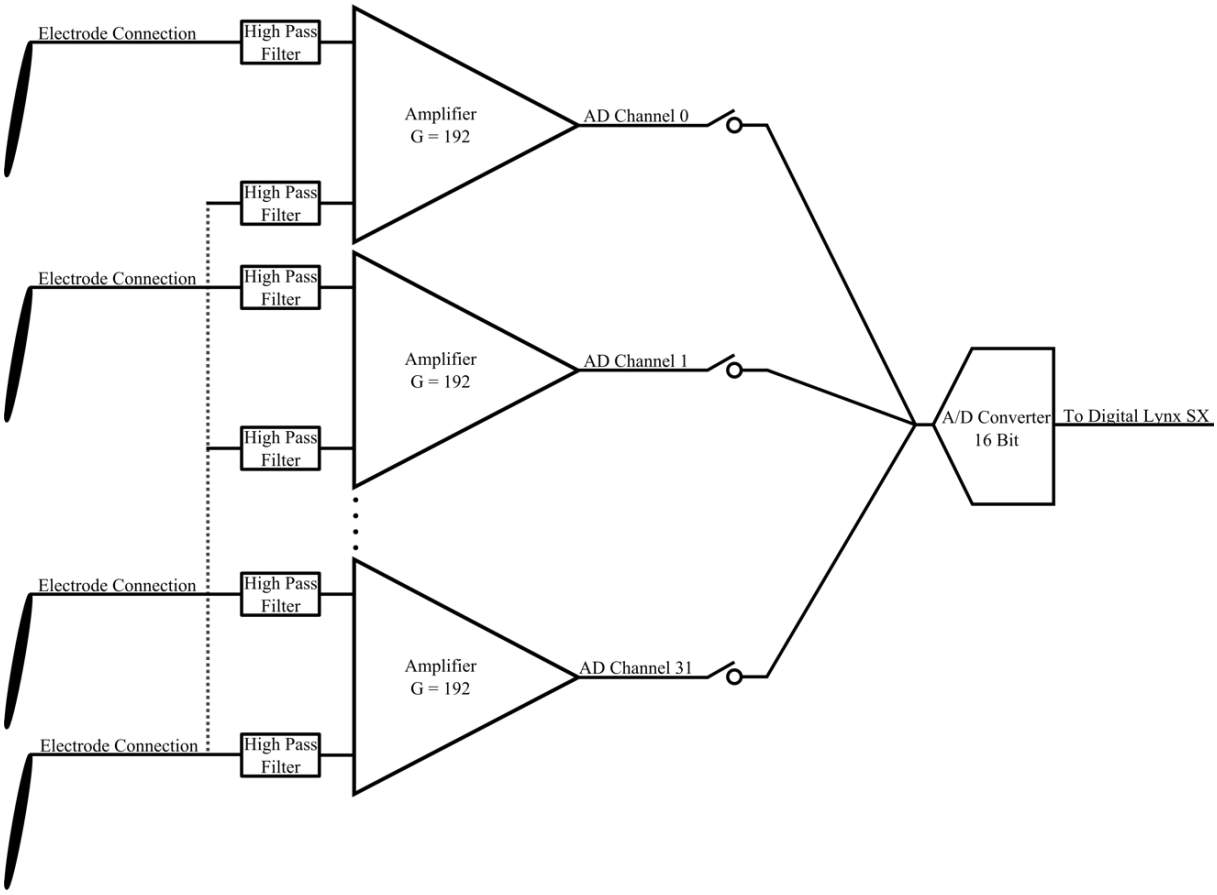



**Figure 6-1 Multiplexing Amplifier and ADC**

## 6.2 Referencing

The most common AFE used with the FreeLynx is the 64 channel AFE. This interface includes two RHD2132 chips, each with its own reference. By default, these references are tied together with a 0 Ω resistor. If you prefer to keep the references separate, contact Neuralynx to have the resistor removed.

These two references map to Neuralynx references R1 and R5.

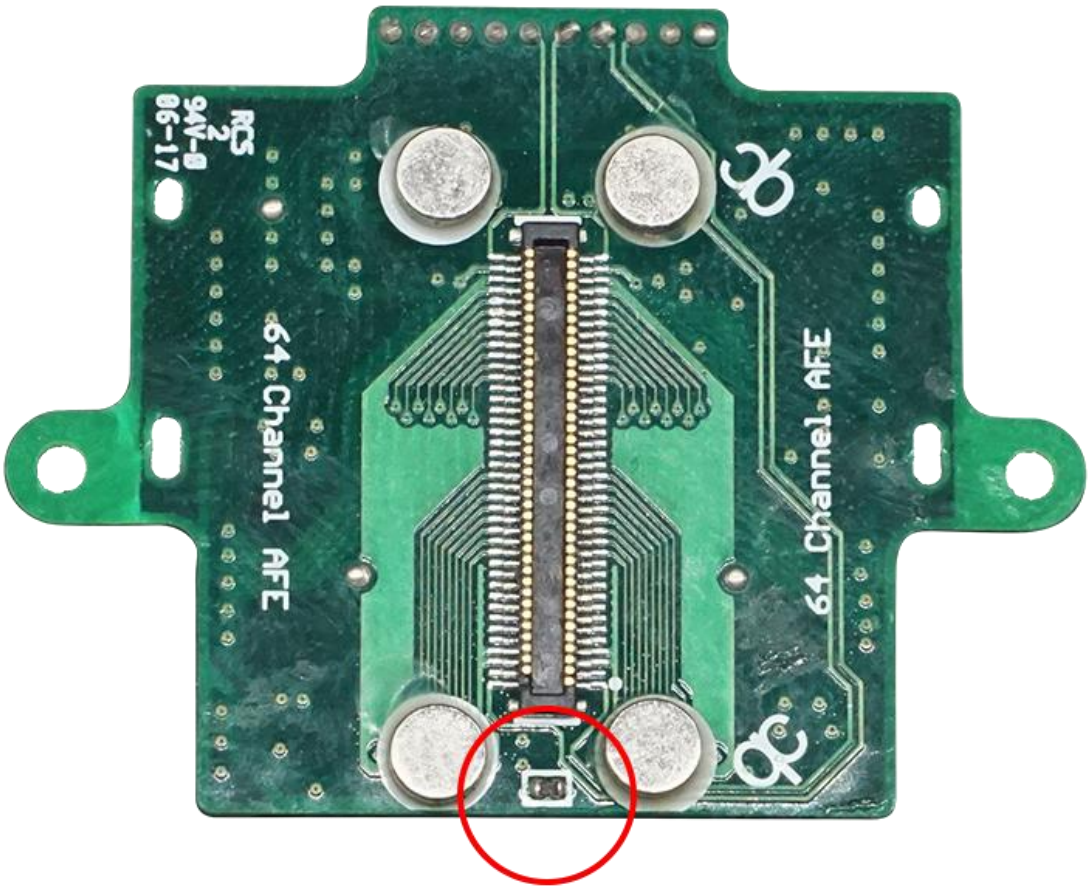



**Figure 6-2 Location of 0 Ω Resistor**

## 6.3 Status LED

The status LED indicates a variety of information about the state of the FreeLynx. If video tracking LEDs are enabled, the status LED remains off to prevent it from interfering with tracking. When video tracking is disabled, the status LED is on.

The status LED is also a good location to wave the power stick magnet for turning the FreeLynx on or off. See ***6.9 Turning the FreeLynx On or Off with the Power Stick***.

**Figure 6-3 Status LED**

### 6.3.1 Normal LED Statuses

#### *6.3.1.1 Power Up*

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

#### *6.3.1.2 Successful Connection*

The LED stays green.

#### *6.3.1.3 System Parameter Update*

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

### 6.3.2 Error Codes

#### *6.3.2.1 microSD Error*

The LED stays red if any of the following microSD errors occur:

- card full
- data write failed
- buffer error

#### *6.3.2.2 Invalid Network Name (SSID)*

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

#### *6.3.2.3 Unable to Contact Access Point*

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

#### *6.3.2.4 Jumbo Packet Test Failed*

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

#### *6.3.2.5 Invalid Channel Capacity*

The LED changes between red and green if the channel capacity set on the FreeLynx does not match the channel capacity set on the SX. This error requires the FreeLynx to be rebooted. This error can only occur when using the FreeLynx in the Connect-to-SX configuration.

## 6.4 Wireless Communication

The FreeLynx communicates wirelessly using a Wi-Fi access point. FreeLynx hardware 4.2.0 and newer uses 802.11ac Wi-Fi and can transmit up to 256 channels of data (Connect-to-Cheetah only). FreeLynx hardware 4.1.0 and older uses 802.11n Wi-Fi and can transmit up to 128 channels of data.

Access points come preconfigured. If needed, they can be configured to use any available 5 GHz Wi-Fi channel. Signal strength decreases as the FreeLynx is moved further away from the access point. For optimal wireless communication, the access point should be ceiling-mounted in the area where the FreeLynx is used. See ***8 Wireless Access Point Setup***.

## 6.5 AFE Interface Connector

The FreeLynx contains an AFE Interface Connector, which allows AFEs to connect to the FreeLynx. AFEs range from 32 to 256 channels and can connect directly to the FreeLynx or through cabling. Contact Neuralynx regarding your specific AFE needs.

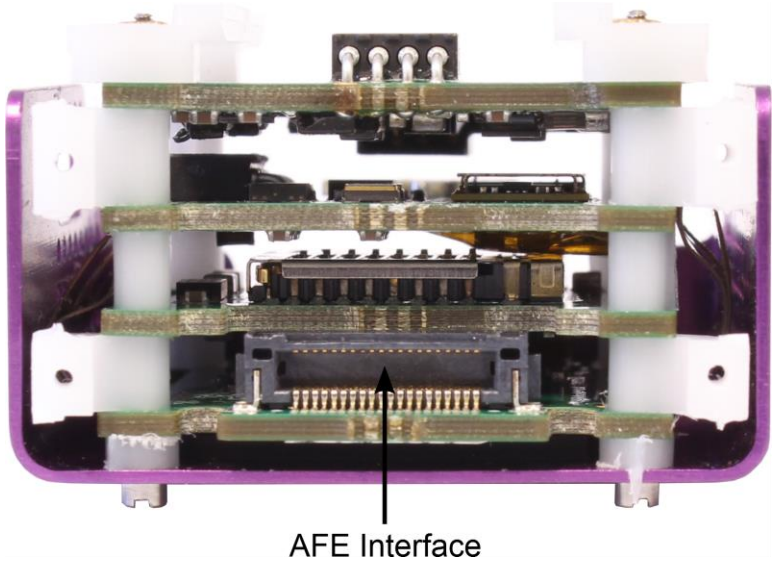



**Figure 6-4 FreeLynx AFE Interface Connector**

## 6.6 Digital I/O Connector

The pinout for the FreeLynx Digital I/O Connector is shown in the figure below.

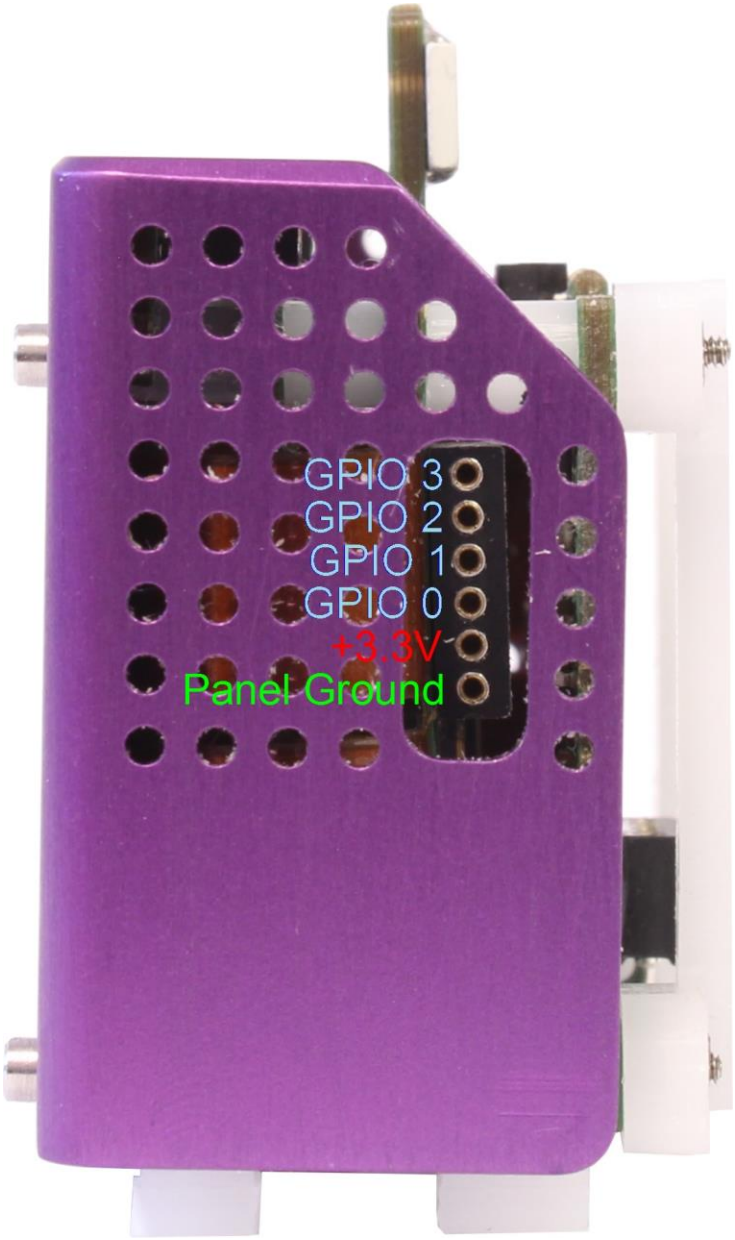



**Figure 6-5 FreeLynx Digital I/O Connector Pinout**

## 6.7 Tracking LEDs

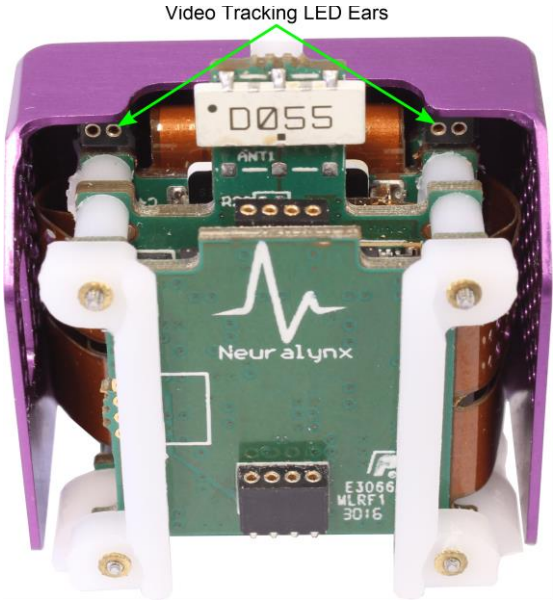



**Figure 6-6 FreeLynx Video Tracking LED Ears**

The FreeLynx contains two Video Tracking LED Ears for video tracking. Please contact Neuralynx for assistance finding a solution for your specific Video Tracking LED needs.

## 6.8 Connecting a Battery to the FreeLynx

The batteries connect to the FreeLynx using a rail guide system. Secure the battery by sliding it down the rails until it clicks.

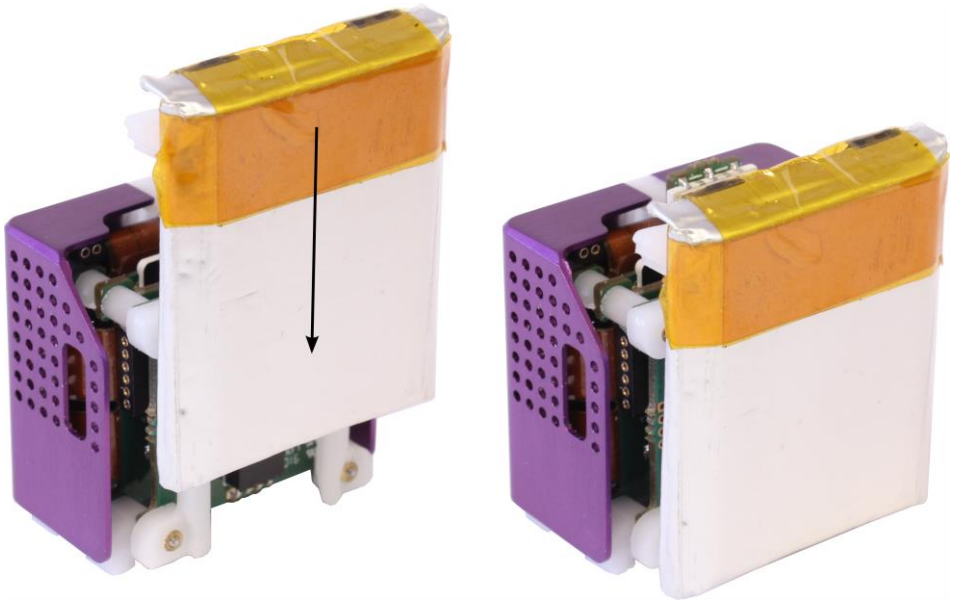


**Figure 6-7 Connecting a Battery**

At this point the FreeLynx can be turned on.

## 6.9 Turning the FreeLynx On or Off with the Power Stick

The FreeLynx comes with a Power Stick, a wand with a magnet at the end of it. The Power Stick is used to turn the FreeLynx On and Off. Simply bring the end of the Power stick within 1 inch of the top of the FreeLynx and remove. The FreeLynx status LED should illuminate Blue when the Power Stick is removed. After the FreeLynx has finished booting, the status LED will illuminate Green.

**Figure 6-8 Power Stick within 1 inch of FreeLynx**

**Figure 6-9 FreeLynx Status LED Illuminated Blue - Booting**

**Figure 6-10 FreeLynx Status LED Illuminated Green - Booted**

If the FreeLynx status LED does not illuminate, try to bring the power stick closer. If it still does not illuminate, replace the battery.

## 6.10 Turning the FreeLynx On or Off with the IR Remote

The IR Transmitter and IR Receiver are FreeLynx accessories that can be used to turn the FreeLynx On and Off. To add this to the FreeLynx, connect the IR Receiver to the top connection on the FreeLynx. This is shown in the figure below. After connecting the battery, point the IR Transmitter at the IR receiver and press the button. The IR Transmitter should be within 2 feet of the IR receiver. The FreeLynx status LED should illuminate Blue after the button is pressed. After the FreeLynx has finished booting, the status LED will illuminate Green.

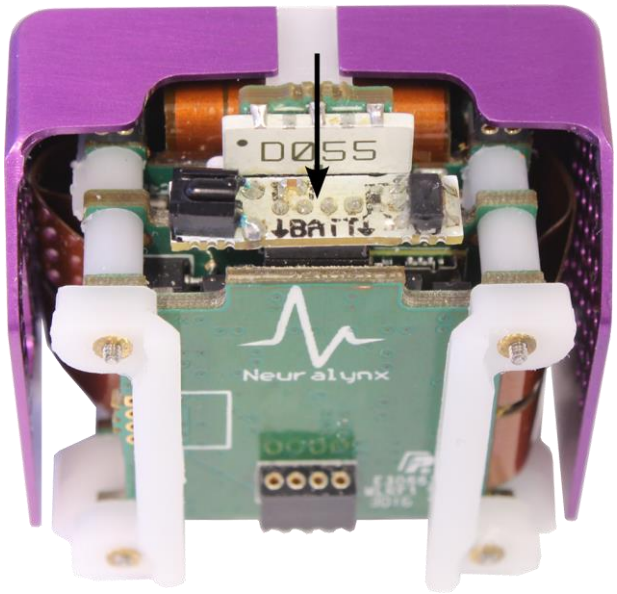



**Figure 6-11 IR Receiver connected to FreeLynx**

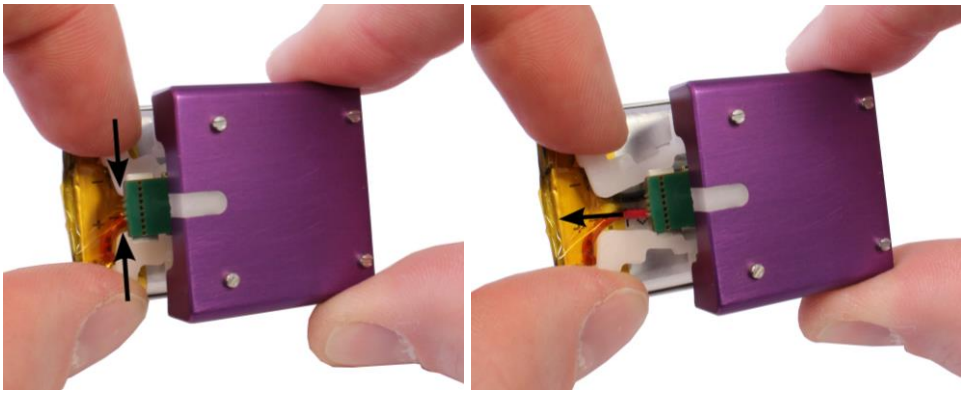
**Figure 6-12 FreeLynx Status LED Illuminated Blue - Booting**

**Figure 6-13 FreeLynx Status LED Illuminated Green – Booted**

If the FreeLynx status LED does not illuminate, try to bring the remote closer. If it still does not illuminate, replace the battery.

## 6.11 Disconnecting a Battery from the FreeLynx

The battery should be removed from the FreeLynx when it is not in use or for charging. Gently squeeze the clip on the battery and pull straight out along the guide rails.



## 6.12 Charging the Battery

The batteries should be charged after each use. There are two versions of the FreeLynx Charger: the 200 mA version and the 800 mA version. The 200 mA version is marked with a green sticker and the 800 mA version is marked with a blue sticker. Each battery contains a sticker that designates which version of the charger it is compatible with. To prevent damaging the batteries, it is important that batteries with one color sticker not be connected to chargers with a different color sticker.

The FreeLynx Charger comes with a Mini-USB cable. Connect the USB Cable to a USB Outlet and the FreeLynx Charger. Once connected, the FreeLynx Charger should illuminate Green. NOTE: For the 800 mA version to operate at the full charge rate, it must be connected to a USB port with a 1 A output.


**Figure 6-14 FreeLynx Charger**

The battery connects to the charger the same way it connects to the FreeLynx, using a rail guide system. Secure the battery by sliding it down the rails until it clicks.

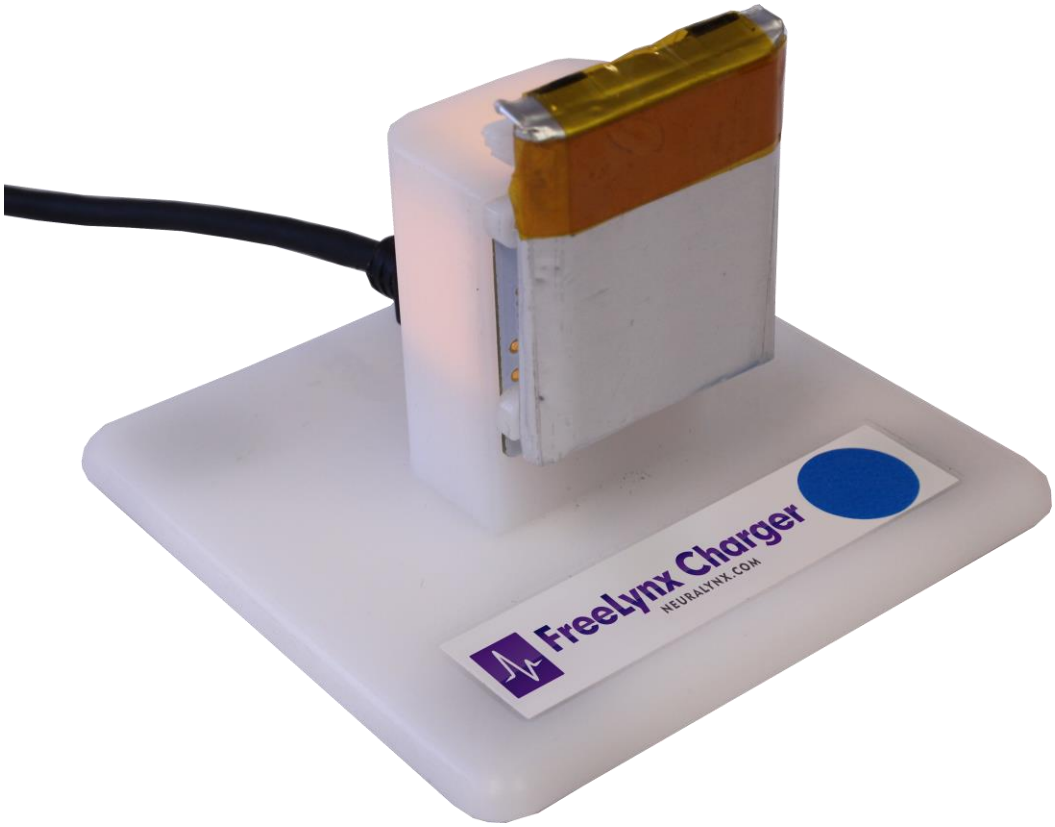



**Figure 6-15 Connecting a Battery**

Once the battery has been slid into place, the FreeLynx Charger should illuminate Orange, signifying it is charging. When the FreeLynx Charger turns back to Green, the charge cycle is complete.

## 6.13 FreeLynx Power Tether

The FreeLynx can also be powered with a FreeLynx Power Tether. This can be useful to ensure power is not lost during a system parameters update or pairing.

The FreeLynx Power Tether comes with a thin Mini USB cable. Connect the USB cable to a USB outlet. For the FreeLynx Power Tether to operate properly, it must be connected to a USB outlet with a 1 A output.

The FreeLynx Power Tether connects to the FreeLynx the same way the battery connects to the FreeLynx, using a rail guide system. Secure the FreeLynx Power Tether by sliding it down the rails until it clicks. Refer to the figure below.

**Figure 6-16 Connecting the FreeLynx Power Tether**

At this point the FreeLynx can be turned on. Power on the FreeLynx using the Power Stick as described in ***Section 6.9 Turning the FreeLynx On or Off with the Power Stick***. Ensure the status LED illuminates. Using the Power Tether, the FreeLynx can be run indefinitely.

## 6.14 MicroSD Card

The FreeLynx contains a microSD memory card that has two partitions. The first, labeled NlxOS, contains the device firmware and system parameters file. The second, labeled NlxData, contains recorded acquisition data files. When newer versions of firmware are released, they can be loaded onto the microSD card NlxOS partition. To remove the card, gently press it in to unlatch it and then pull the card out. ESD-safe tweezers can also be used to remove the card.

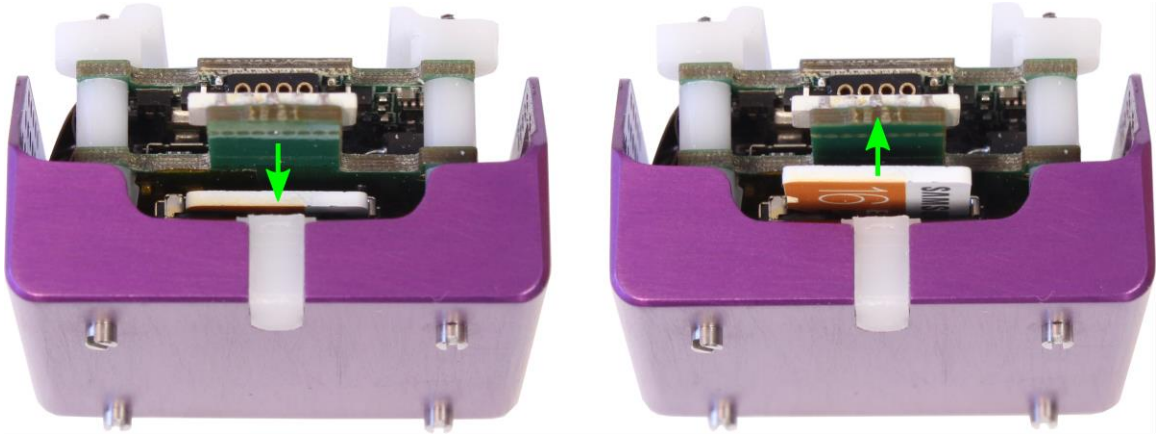


**Figure 6-17 Removing the microSD Card**

Please contact Neuralynx before making any changes to the microSD card.

# 7 Pairing a FreeLynx with a Digital Lynx SX

To use a FreeLynx in the Connect-to-SX configuration, the FreeLynx and SX must go through the pairing process. Pairing is not required when using a FreeLynx in the Connect-to-Cheetah configuration.

If you ordered FreeLynx and Digital Lynx SX together, the FreeLynx will already be paired with the SX. If you ordered the FreeLynx as an addition to your Digital Lynx SX System, they must be paired using the following procedure.

## *7.1 Manual Pairing*

### 7.1.1 Obtain the FreeLynx IP Address

Obtain the IP Address for the FreeLynx you wish to Pair. The default FreeLynx IP Address is 192.168.4.101. If you don't know the FreeLynx's IP Address, continue to ***7.2 Automatic Pairing***.

### 7.1.2 Start Cheetah

Start Cheetah with a generic configuration. Keep in mind the FreeLynx is not paired yet, meaning you cannot use a configuration that calls for AD Channels associated with the FreeLynx. The purpose of starting Cheetah here is to send FreeLynx pairing commands to the Digital Lynx SX.

### 7.1.3 Remove All Paired FreeLynx

Create and open the configuration file shown in the figure below. Once the configuration file has been executed, no IP Addresses will be Paired with the Digital Lynx SX.

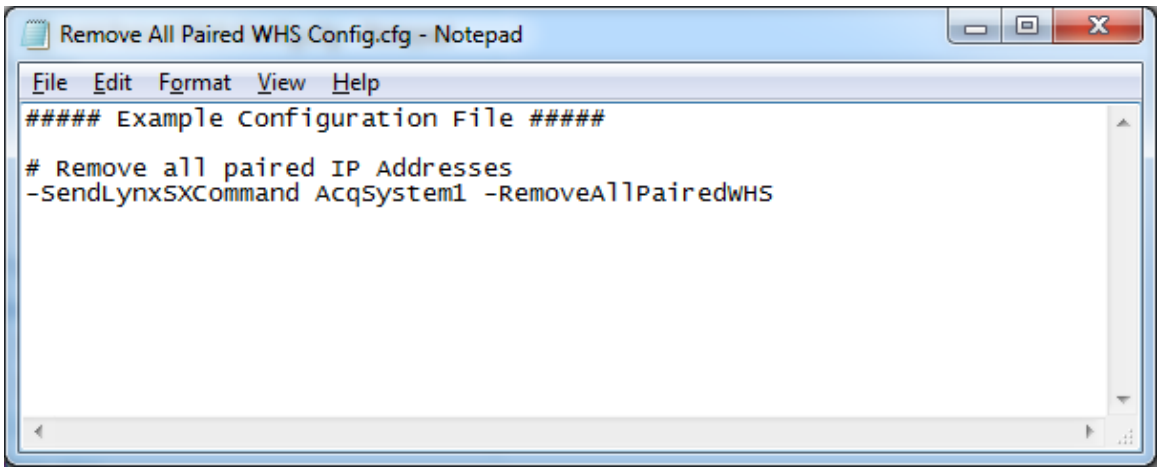



**Figure 7-1 Remove All Paired IP Address**

### 7.1.4 Pair a New FreeLynx

Create and open the configuration file shown in the figure below. Substitute your FreeLynx's IP Address for what is listed below. Once the configuration file has been executed, your FreeLynx will be paired with the Digital Lynx SX.

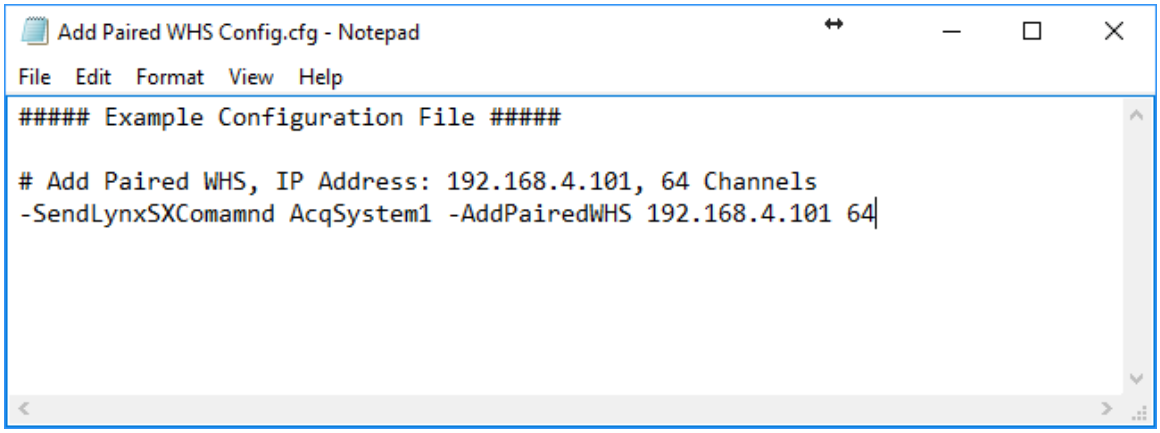



**Figure 7-2 Pair Wireless Headstage**

### 7.1.5 Close Cheetah

Close the currently running instance of Cheetah.

### 7.1.6 Turn the FreeLynx On

Verify that the Digital Lynx SX is turned on and booted. Then use the Power Stick to turn on the FreeLynx.

### 7.1.7 Restart Cheetah

Restart Cheetah with the appropriate configuration file. Start acquisition and observe the AD channels. Verify they show the expected signals.

## *7.2 Automatic Pairing*

### 7.2.1 Start Cheetah

Start Cheetah with a generic configuration. Keep in mind the FreeLynx is not paired yet, meaning you cannot use a configuration that calls for AD Channels associated with the FreeLynx. The purpose of starting Cheetah here is to send FreeLynx pairing commands to the Digital Lynx SX.

### 7.2.2 Remove All Paired IP Addresses

Create and Open the configuration file shown in the figure below. Once the configuration file has been executed, no IP Addresses will be Paired with the Digital Lynx SX.

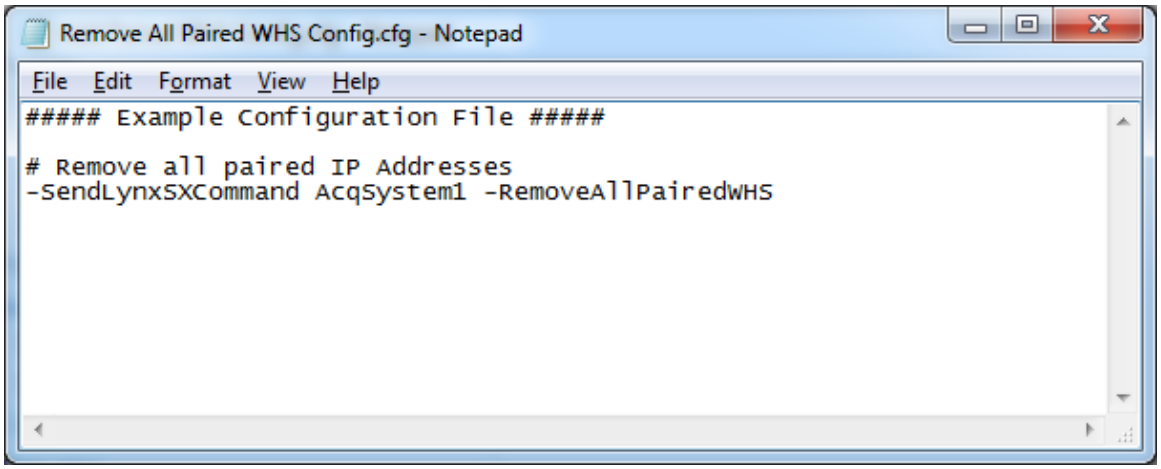



**Figure 7-3 Remove All Paired IP Address**

### 7.2.3 Enter Automatic Pairing Mode

Create and Open the configuration file shown in the figure below. Once the configuration file has been executed, you have 60 seconds to complete the following step. The pairing period can be from 5 seconds to 255 seconds.

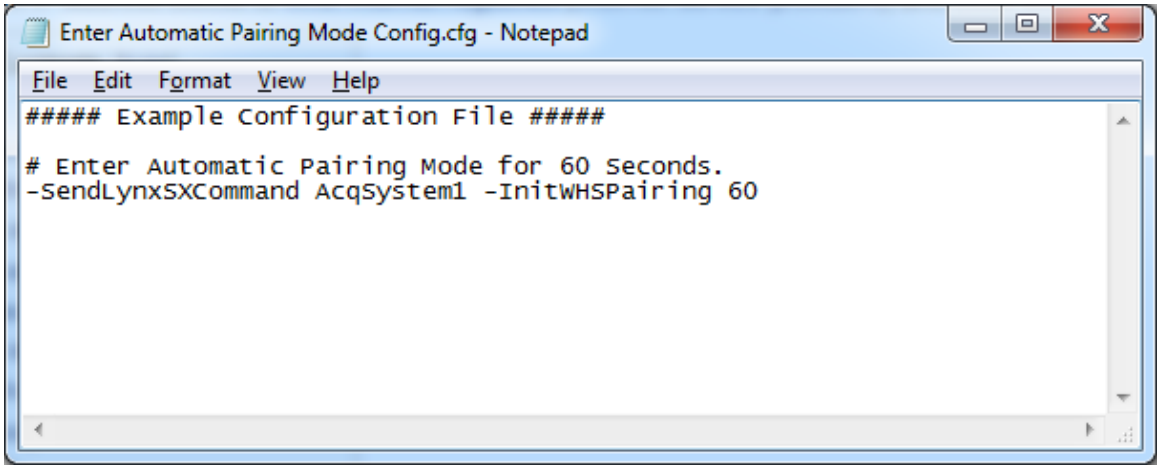



**Figure 7-4 Enter Automatic Pairing Mode**

### 7.2.4 Turn the FreeLynx On

Using the Power Stick, power On the FreeLynx. It takes approximately 30 seconds for the FreeLynx and Digital Lynx SX Pair.

### 7.2.5 Turn the FreeLynx Off

Using the Power Stick, power off the FreeLynx.

### 7.2.6 Close Cheetah

Close the currently running instance of Cheetah.

### 7.2.7 Turn the FreeLynx On

Using the Power Stick, power on the FreeLynx.

### 7.2.8 Restart Cheetah

Restart Cheetah with the appropriate configuration file. Start acquisition and observe the AD channels. Verify they show the expected signals.

# 8 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 FreeLynx and access point.
- Minimize distance between the FreeLynx and access point.
- Ceiling-mount the access point over the area where the FreeLynx 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.

## 8.1 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 FreeLynx wireless network with SSID nlx_wireless_freelynx, and three other networks indicated by colored bumps in the top third of the image.

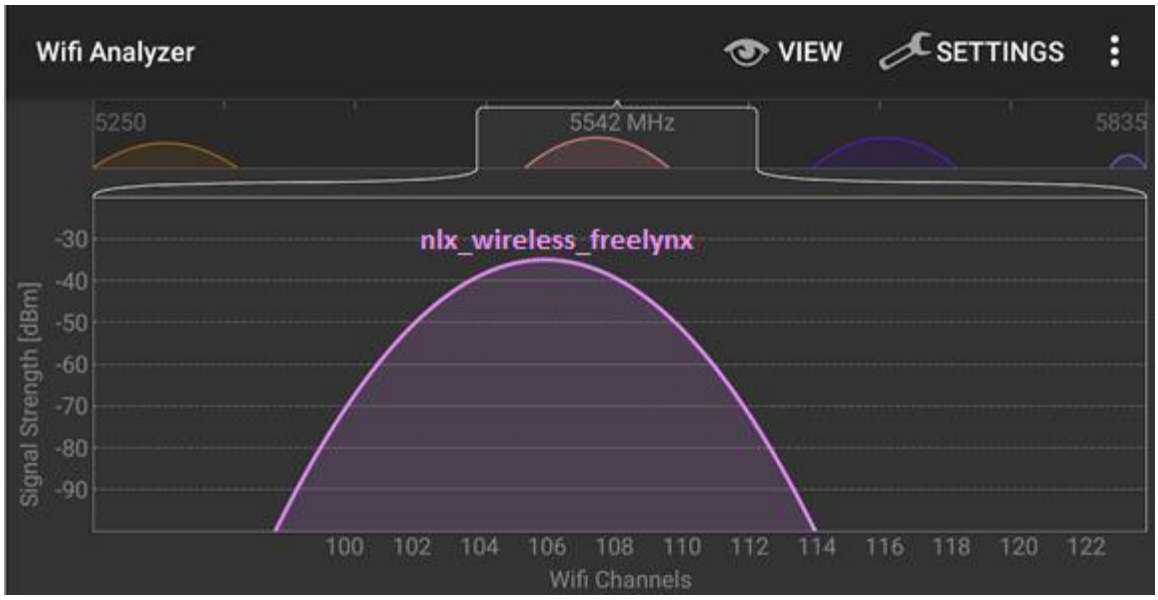



**Figure 8-1 5 FreeLynx Set to Channel 106**

If your network scan shows your FreeLynx 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

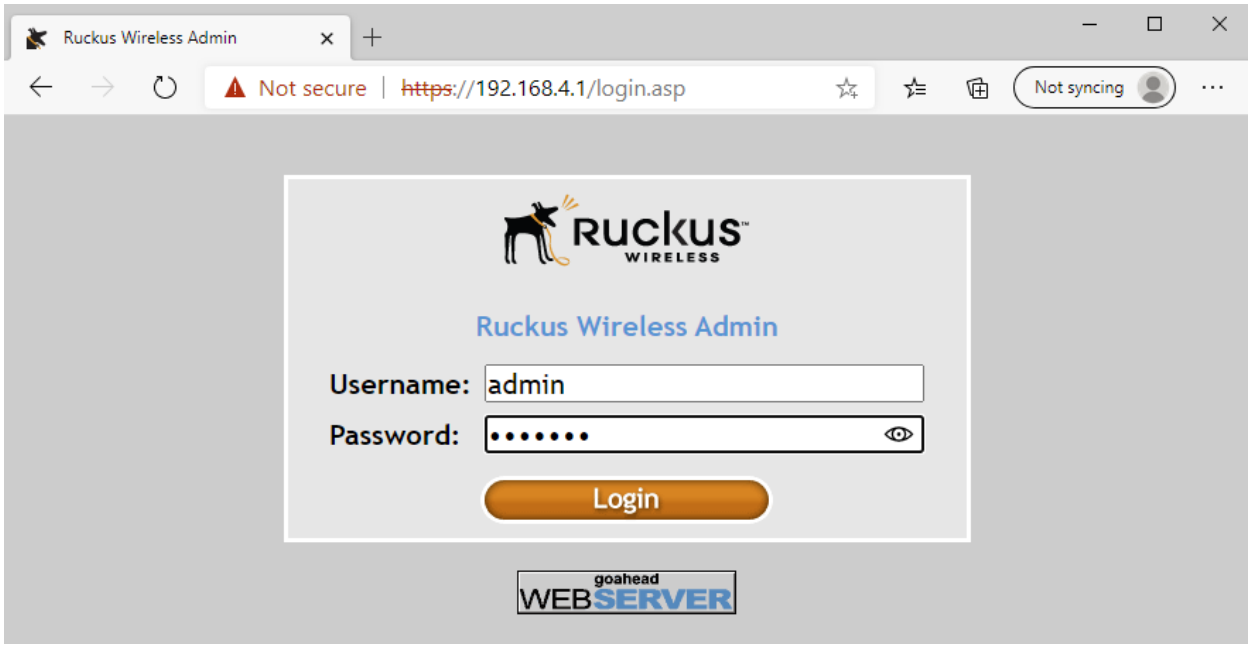



**Figure 8-2 Login Page using Microsoft Edge**

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

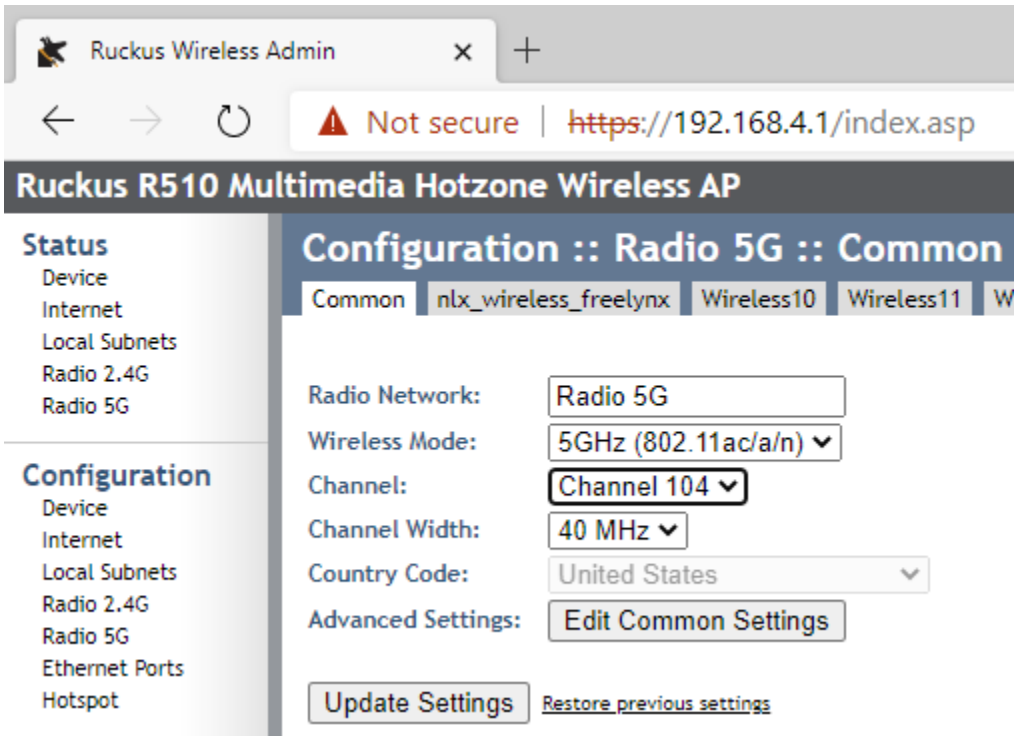



**Figure 8-3 Changing Wi-Fi Channel**

# 9 FreeLynx Commands

Commands can be sent to the FreeLynx through Cheetah or NetCom to configure and query the device. Command format depends on whether you are using the Connect-to-Cheetah or Connect-to-SX setup.

Each setup allows commands to be sent by opening .cfg files containing commands or by using the Neuralynx Netcom API. When a FreeLynx is used in the Connect-to-Cheetah configuration, a small subset of the commands is available in the Cheetah GUI. See ***Figure 4-4 FreeLynx GUI***.

In all examples, AcqSystem1 is used as the Hardware Sub System Name.

**Notes on Connect-to-Cheetah Commands**

- Commands with an * next to them are available in the Cheetah GUI.

- Cheetah does not currently provide a way to display the return results of any commands except those already available in the Cheetah GUI.

- Each Connect-to-Cheetah 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 Connect-to-Cheetah 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.

## 9.1 SetDigitalIOPortDirection

### 9.1.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetDigitalIOPortDirection <Headstage> <Value>** | |
|---|---|
| Sets the direction for each bit of the Digital IO Port on the FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetDigitalIOPortDirection 1 11001100 | |
| **Default:** The default direction for all digital IO is set to Input (Value = "11111111") | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Value | 0 or 1 for each bit, 0 for output, 1 for input.<br>**00000000 - 11111111** |

### 9.1.2 Connect-to-Cheetah

```
# 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"
```

## *9.2 SetDigitalIOBit*

### 9.2.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetDigitalIOBit <Headstage> <Value>** | |
|---|---|
| Sets the logic value of a bit of the Digital IO Port on the FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetDigitalIOBit 1 0 1 | |
| **Default:** The default value of all Digital IO Port bits are logic low. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Bit number | **0**: Bit 0<br>**1**: Bit 1<br>**2**: Bit 2<br>**3**: Bit 3<br>**4**: Bit 4<br>**5**: Bit 5<br>**6**: Bit 6<br>**7**: Bit 7 |
| Value | **0**: Logic Low<br>**1**: Logic High |

### 9.2.2 Connect-to-Cheetah

```
# 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 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

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

-SendBinaryCommand AcqSystem1 "010003060100"        # AFE GPIO 1 = 0
-SendBinaryCommand AcqSystem1 "010003060200"        # AFE GPIO 2 = 0
-SendBinaryCommand AcqSystem1 "010003060300"        # AFE GPIO 3 = 0
-SendBinaryCommand AcqSystem1 "010003060400"        # base GPIO 0 = 0
-SendBinaryCommand AcqSystem1 "010003060500"        # base GPIO 1 = 0
-SendBinaryCommand AcqSystem1 "010003060600"        # base GPIO 2 = 0
-SendBinaryCommand AcqSystem1 "010003060700"        # base GPIO 3 = 0

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

## *9.3 GetDigitalIOPortValue*

### 9.3.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetDigitalIOPortValue <Headstage>** | |
|---|---|
| Gets the digital IO port value of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetDigitalIOPortValue 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Digital IO Port Value | Decimal value of the two 4-bit ports<br>**0-255:** Value in decimal |

### 9.3.2 Connect-to-Cheetah

```
# 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"
```

## *9.4 SetDigitalIOPortValue*

### 9.4.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetDigitalIOPortValue <Headstage> <Value>** | |
|---|---|
| Sets the logic value of the whole port of the Digital IO Port on the FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetDigitalIOPortValue 1 2 | |
| **Default:** The default value of all Digital IO Port bits are logic low. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Value | Decimal value of the two 4-bit ports<br>**0-255:** Value in decimal |

### 9.4.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x07
# parameter 1:  8-bit value to simultaneously set all 4 base GPIO and
#  all 4 AFE GPIO bits (1 byte)

# 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

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

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

## *9.5 SetTrackingLED**

### 9.5.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetTrackingLED <Headstage> <Value>** | |
|---|---|
| Sets the tracking LED enable | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetTrackingLED 1 0 | |
| **Default:** The default value of the tracking LED enable is logic low. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Bit number | **0**: Tracking LED disabled (enables status LED)<br>**1**: Tracking LED enabled (disables status LED) |

### 9.5.2 Connect-to-Cheetah

This command is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0002
# command ID:    x08
# parameter1:    x00 (1 byte)


# Examples

# list of command values:      01 0002 08 00
-SendBinaryCommand AcqSystem1 "0100020800"        # off

# -SendBinaryCommand AcqSystem1 "0100020801"        # on
```

## *9.6 GetRegister*

### 9.6.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetRegister <Headstage> <Bank> <Register Number>** | |
|---|---|
| Gets a register from one of the two Intan RHD2132 chips | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetRegister 1 1 0 | |
| **Default:** The default value of the registers are documented in the RHD2132 data sheet | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Bank | This value can be one of the following keywords:<br>**1**: Bank 1 (channels 1-32)<br>**2**: Bank 2 (channels 33-64)<br>**3**: Bank 3 (channels 65-96)<br>**4**: Bank 4 (channels 97-128)<br>**5**: Bank 5 (channels 129-160)<br>**6**: Bank 6 (channels 161-192)<br>**7**: Bank 7 (channels 193-224)<br>**8**: Bank 8 (channels 225-256) |
| Register Number | **0-17**: Register number in decimal |
| **Return Value:** | |
| Register Data | **0-255**: 8-bit register value in decimal |

### 9.6.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0003
# command ID:    x0A
# parameter 1:   bank select (1 byte) - a bank represents one 32-ch
#  Intan chip
# parameter 2:   register select (1 byte)
#
# return:        8-bit value (1 byte)


# Examples

# get register 0 on bank 1
# list of command values:      01 0003 0A 01 00
-SendBinaryCommand AcqSystem1 "0100030A0100"
```

## 9.7 SetRegister

### 9.7.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetRegister <Headstage> <Bank> <Register Number> <RegisterData>** | |
|---|---|
| Sets a register in one of the two Intan RHD2132 chips | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetRegister 1 1 0 222 | |
| **Default:** The default value of the registers are documented in the RHD2132 data sheet. | |
| **Usage:** Can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Bank | This value can be one of the following keywords:<br>**1**: Bank 1 (channels 1-32)<br>**2**: Bank 2 (channels 33-64)<br>**3**: Bank 3 (channels 65-96)<br>**4**: Bank 4 (channels 97-128)<br>**5**: Bank 5 (channels 129-160)<br>**6**: Bank 6 (channels 161-192)<br>**7**: Bank 7 (channels 193-224)<br>**8**: Bank 8 (channels 225-256) |
| Register Number | **0-17**: Register number in decimal |
| Register Data | **0-255**: 8-bit register value in decimal |

### 9.7.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0004
# command ID:    x09
# parameter 1:   bank select 0-7 (1 byte) - a bank represents one 32-ch
#  Intan chip
# parameter 2:   register select (1 byte)
# parameter 3:   register value (1 byte)


# Examples

# on bank 1, set register 8 to x14
# list of command values:      01 0004 09 01 08 14
-SendBinaryCommand AcqSystem1 "01000409010814"
```

## *9.8 PowerOffSystem*

### 9.8.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSPowerOffSystem <Headstage>** | |
|---|---|
| Powers down the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSPowerOffSystem 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |

### 9.8.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x0B


# Examples

# list of command values:      01 0001 0B
-SendBinaryCommand AcqSystem1 "0100010B"
```

## *9.9 GetSystemSerialNumber*

### 9.9.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetSystemSerialNumber <Headstage>** | |
|---|---|
| Gets the serial number of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetSystemSerialNumber 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Serial Number | **0-65535**: 16-bit value in decimal |

### 9.9.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x0C
#
# return:        16-bit value (2 bytes)


# Examples

# list of command values:      01 0001 0C
-SendBinaryCommand AcqSystem1 "0100010C"
```

## *9.10 GetProductName**

### 9.10.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetProductName <Headstage>** | |
|---|---|
| Gets the product name of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetProductName 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Product Name | Null-terminated variable-length ASCII string up to 33 bytes |

### 9.10.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x0D
#
# return:        2-31 ASCII character, null terminated string
#                (3-32 bytes)


# Examples

# list of command values:      01 0001 0D
-SendBinaryCommand AcqSystem1 "0100010D"
```

## *9.11 GetProductRevision**

### 9.11.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetProductRevision <Headstage>** | |
|---|---|
| Gets the product revision of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetProductRevision 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Product Revision | Null-terminated variable-length ASCII string up to 11 bytes |

### 9.11.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x0E
#
# return:        2-11 ASCII character, null terminated string
#                (3-12 bytes)


# Examples

# list of command values:      01 0001 0E
-SendBinaryCommand AcqSystem1 "0100010E"
```

## *9.12 GetFirmwareVersion**

### 9.12.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetFirmwareVersion <Headstage>** | |
|---|---|
| Gets the firmware version of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetFirmwareVersion 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Firmware Version | Null-terminated variable-length ASCII string up to 11 bytes |

### 9.12.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x0F
#
# return:        2-11 ASCII character, null terminated string
#                (3-12 bytes)


# Examples

# list of command values:      01 0001 0F
-SendBinaryCommand AcqSystem1 "0100010F"
```

## *9.13 GetChannelCapacity**

### 9.13.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetChannelCapacity <Headstage>** | |
|---|---|
| Gets the channel capacity of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetChannelCapacity 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Channel Capacity | Null-terminated variable-length ASCII string up to 4 bytes |

### 9.13.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x10
#
# return:        16-bit value (2 bytes)


# Examples

# list of command values:      01 0001 10
-SendBinaryCommand AcqSystem1 "01000110"
```

## 9.14 SetChannelCapacity

### 9.14.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetChannelCapacity <Headstage> < Channel Capacity>** | |
|---|---|
| Sets the channel capacity of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetChannelCapacity 1 128 | |
| **Default:** Channel capacity is device specific and factory set by Neuralynx. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Channel Capacity | This value can be one of the following keywords:<br>**64**: 64-channel capacity<br>**128**: 128-channel capacity<br>**256**: 256-channel capacity (microSD recording only, wireless transmission works but is limited to 128 channels) |

### 9.14.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0003
# command ID:    x21
# parameter 1:   16-bit channel capacity


# Examples

# Set channel capacity to 64
# list of command values:      01 0003 21 0040
-SendBinaryCommand AcqSystem1 "010003210040"

# Set channel capacity to 128
-SendBinaryCommand AcqSystem1 "010003210080"

# Set channel capacity to 256
-SendBinaryCommand AcqSystem1 "010003210100"
```

## *9.15 GetMACAddress**

### 9.15.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetMACAddress <Headstage>** | |
|---|---|
| Gets the MAC address of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetMACAddress 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** The MAC is returned as 6 separate octets (Octet1 Octet2 Octet3 Octet4 Octet5 Octet6) | |
| Octet1 | Octet1 (1 Byte): Range 0..255 |
| Octet2 | Octet2 (1 Byte): Range 0..255 |
| Octet3 | Octet3 (1 Byte): Range 0..255 |
| Octet4 | Octet4 (1 Byte): Range 0..255 |
| Octet5 | Octet5 (1 Byte): Range 0..255 |
| Octet6 | Octet6 (1 Byte): Range 0..255 |

### 9.15.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x11
#
# return:        6 octets (6 bytes)


# Examples

# list of command values:      01 0001 11
-SendBinaryCommand AcqSystem1 "01000111"
```

## *9.16 GetIPAddress**

### 9.16.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetIPAddress <Headstage>** | |
|---|---|
| Gets the IP address of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetIPAddress 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| IP Address | Null-terminated variable-length ASCII string up to 16 bytes |

### 9.16.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x12
#
# return:        4 octets (4 bytes)


# Examples

# list of command values:      01 0001 12
-SendBinaryCommand AcqSystem1 "01000112"
```

## 9.17 SetIPAddress

### 9.17.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetIPAddress <Headstage>** | |
|---|---|
| Sets the IP address of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetIPAddress 1 | |
| **Default:** IP Addpress is device specific and factory set by Neuralynx. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| IP Address | Null-terminated variable-length ASCII string up to 16 bytes |

### 9.17.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0005
# command ID:    x13
# parameter1:    Octet 1 (1 byte)
# parameter2:    Octet 2 (1 byte)
# parameter3:    Octet 3 (1 byte)
# parameter4:    Octet 4 (1 byte)


# Examples

# Set IP address to 192.168.4.101
# Octet1: dec 192   = hex xC0
# Octet2: dec 168   = hex xA8
# Octet3: dec 4     = hex x04
# Octet4: dec 101   = hex x65
# list of command values:      01 0005 13 C0 A8 04 65
-SendBinaryCommand AcqSystem1 "01000513C0A80465"
```

## 9.18 GetBatteryVoltage

### 9.18.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetBatteryVoltage <Headstage>** | |
|---|---|
| Gets the battery voltage (in mV) of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetBatteryVoltage 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Battery Voltage | Null-terminated variable-length ASCII string up to 6 bytes |

### 9.18.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x14
#
# return:        voltage in mV (2 bytes)


# Examples

# list of command values:      01 0001 14
-SendBinaryCommand AcqSystem1 "01000114"
```

## 9.19 GetAverageCurrent

### 9.19.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetAverageCurrent <Headstage>** | |
|---|---|
| Gets the battery voltage (in mA) of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetAverageCurrent 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Average Current | Null-terminated variable-length ASCII string up to 6 bytes |

### 9.19.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x15
#
# return:        current in mA (2 bytes)


# Examples

# list of command values:      01 0001 15
-SendBinaryCommand AcqSystem1 "01000115"
```

## 9.20 GetMaxCurrent

### 9.20.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetMaxCurrent <Headstage>** | |
|---|---|
| Gets the battery voltage (in mA) of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetMaxCurrent 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Max Current | Null-terminated variable-length ASCII string up to 6 bytes |

### 9.20.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x16
#
# return:        current in mA (2 bytes)


# Examples

# list of command values:      01 0001 16
-SendBinaryCommand AcqSystem1 "01000116"
```

## *9.21 GetStateOfCharge*

### 9.21.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetStateOfCharge <Headstage>** | |
|---|---|
| Gets the battery voltage (in %) of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetStateOfCharge 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| State of Charge | Null-terminated variable-length ASCII string up to 4 bytes |

### 9.21.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x17
#
# return:        % value 0-100 (1 byte)


# Examples

# list of command values:      01 0001 17
-SendBinaryCommand AcqSystem1 "01000117"
```

## *9.22 GetCoreTemp**

### 9.22.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetCoreTemp <Headstage>** | |
|---|---|
| Gets the core temperature (in degrees C) of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetCoreTemp 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Core Temperature | Null-terminated variable-length ASCII string up to 6 bytes |

### 9.22.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x18
#
# return:        Celsius (1 byte)


# Examples

# list of command values:      01 0001 18
-SendBinaryCommand AcqSystem1 "01000118"
```

## *9.23 GetShutdownVoltage*

### 9.23.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetShutdownVoltage <Headstage>** | |
|---|---|
| Gets the battery voltage (in mV) of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetShutdownVoltage 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Shutdown Voltage | Null-terminated variable-length ASCII string up to 6 bytes |

### 9.23.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x19
#
# return:        voltage in mV (2 bytes)


# Examples

# list of command values:      01 0001 19
-SendBinaryCommand AcqSystem1 "01000119"
```

## 9.24 GetTransmittedChannels

### 9.24.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetTransmittedChannels <Headstage>** | |
|---|---|
| Gets the transmitted channel mask of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetTransmittedChannels 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Channel Mask<br>with Channel Capacity = 64 | 64-bit hex value<br>Range:<br>0x0000000000000000 to 0xFFFFFFFFFFFFFFFF<br>Each bit represents a channel to be transmitted |
| Channel Mask<br>with Channel Capacity = 128 | 128-bit hex value<br>Range:<br>0x00000000000000000000000000000000 to 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF |
| Channel Mask<br>with Channel Capacity = 256 | 256-bit hex value<br>Range:<br>0x0000000000000000000000000000000000000000000000000000000000000000 to<br>0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF |

### 9.24.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x1B
#
# return:        Each hexadecimal character represents 4 bits
#                64-bit mask for 64-ch (8 bytes)
#                128-bit mask for 128-ch (16 bytes)
#                256-bit mask for 256-ch (32 bytes)


# Examples

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

## 9.25 SetTransmittedChannels

- The behavior of this command depends on the FreeLynx channel capacity. See ***5.6 Switching between microSD and Wi-Fi Recording in a Connect-to-SX Setup with more than 128 Channels***.

### 9.25.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetTransmittedChannels <Headstage> <Channel Mask>** | |
|---|---|
| Sets the transmitted channel mask of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetTransmittedChannels 1 0xFFFFFFFFFFFFFFF0<br>In this example, out of 64 total channels, AD channels 0-3 would appear zeroed out in Cheetah. If recording to microSD card, AD channels 0-3 would not be stored on the card. | |
| **Default:** All channels are enabled by default. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Channel Mask<br>with Channel Capacity = 64 | 64-bit hex value<br>Range:<br>0x0000000000000000 to 0xFFFFFFFFFFFFFFFF<br>Each bit represents a channel to be transmitted |
| Channel Mask<br>with Channel Capacity = 128 | 128-bit hex value<br>Range:<br>0x00000000000000000000000000000000 to 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF |
| Channel Mask<br>with Channel Capacity = 256 | 256-bit hex value<br>Range:<br>0x0000000000000000000000000000000000000000000000000000000000000000 to<br>0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF |

### 9.25.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0009 for 64-ch (9 bytes)
#                x0011 for 128-ch (17 bytes)
#                x0021 for 256-ch (33 bytes)
# command ID:    x1A
# parameter1:    Each hexadecimal character represents 4 bits
#                64-bit mask for 64-ch (16 characters or 8 bytes)
#                128-bit mask for 128-ch (32 characters or 16 bytes)
#                256-bit mask for 256-ch (64 characters or 32 bytes)


# Examples

# 64-ch
# all channels enabled
# list of command values:      01 0009 1A FFFFFFFF FFFFFFFF
-SendBinaryCommand AcqSystem1 "0100091AFFFFFFFFFFFFFFFF"

# only the first channel enabled
# list of command values:      01 0009 1A 00000000 00000001
-SendBinaryCommand AcqSystem1 "0100091A0000000000000001"

# 128-ch
# all channels enabled
# list of command values:      01 0011 1A FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
-SendBinaryCommand AcqSystem1 "0100111AFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF"

# only the first channel enabled
# list of command values:      01 0011 1A 00000000 000000000 0000000 00000001
-SendBinaryCommand AcqSystem1 "0100111A00000000000000000000000000000001"

# 256-ch
# all channels enabled
# list of command values:
# 01 0021 1A FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
-SendBinaryCommand AcqSystem1
"0100211AFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF"
# 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
```

## *9.26 GetBatteryType**

### 9.26.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetBatteryType <Headstage>** | |
|---|---|
| Gets the battery type of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetBatteryType 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Battery Type | 1 Byte ASCII string |

### 9.26.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x1C
#
# return:        1-2 ASCII character, null-terminated string (2-3 bytes)


# Examples

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

## *9.27 SetBatteryType**

### 9.27.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetBatteryType <Headstage>** | |
|---|---|
| Sets the battery type of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetBatteryType 1 A | |
| **Default:** Battery type is device specific and factory set by Neuralynx. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Battery Type | ASCII Battery type, “A”, contact Neuralynx for type definition |

### 9.27.2 Connect-to-Cheetah

This command is available in the Cheetah GUI.

```
# packet ID:    x01
# size:         x0003 for 1 char or x0004 for 2 chars
# command ID:   x1D
# parameter1:   1-2 ASCII character, null-terminated string (2-3 bytes)


# Examples

# set battery type to A, ASCII A is x41, null terminator is x00
# list of command values:      01 0003 1D 41 00
-SendBinaryCommand AcqSystem1 "0100031D4100"

# set battery type to AB, ASCII A is x41, B is x42, null terminator is
#  x00
# list of command values:      01 0004 1D 41 42 00
-SendBinaryCommand AcqSystem1 "0100041D414200"
```

## *9.28 GetSSID**

### 9.28.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetSSID <Headstage>** | |
|---|---|
| Gets the SSID for network connection of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetSSID 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| SSID | Null-terminated variable-length ASCII string up to 64 bytes |

### 9.28.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x1E
#
# return:        2-63 ASCII character, null terminated string
#                (3-64 bytes)


# Examples

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

## 9.29 SetSSID

### 9.29.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetSSID <Headstage>** | |
|---|---|
| Sets the SSID for network connection of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetSSID 1 nlx_wireless_1 | |
| **Default:** SSID is device specific and factory set by Neuralynx. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| SSID | ASCII SSID string, “nlx_wireless_1”, should match the Access Point SSID |

### 9.29.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0004 for 2 char up to x0041 for 63 chars
# command ID:    x1F
# parameter 1:   2-63 ASCII character, null terminated string
#                (3-64 bytes)


# Examples

# set to nlx_wireless
# list of command values: 01 000E 1F 6E 6C 78 5F 77 69 72 65 6C 65 73 73 00
-SendBinaryCommand AcqSystem1 "01000E1F6E6C785F776972656C65737300"
```

## *9.30 GetShutdownTemp*

### 9.30.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetShutdownTemp <Headstage>** | |
|---|---|
| Gets the shutdown temp (in degrees C) of the specified FreeLynx device | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSGetShutdownTemp 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created and the FreeLynx is connected. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Shutdown Temperature | Null-terminated variable-length ASCII string up to 4 bytes |

### 9.30.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x20
#
# return:        Celsius (1 byte)


# Examples

# list of command values:      01 0001 20
-SendBinaryCommand AcqSystem1 "01000120"
```

## 9.31 Convert

### 9.31.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSConvert <Headstage> <Bank> <Channel>** | |
|---|---|
| Performs a single data conversion on the specified channel. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSConvert 1 1 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This command can be used any time after acquisition has been started. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system which will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Bank | This value can be one of the following keywords:<br>**1**: Selects Bank 1 (first 32 channels) as the command target.<br>**2**: Selects Bank 2 (second 32 channels) as the command target. |
| Channel | **1-32**: Selects Channel |
| **Return Value:** | |
| Sampled Value | Null-terminated variable-length ASCII string up to 7 bytes<br>Range is 0x0000 to 0xFFFF, 2's compliment format |

### 9.31.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0003
# command ID:    x60
# parameter1:    bank (1 byte) - a bank represents one 32-ch Intan chip
# parameter2:    channel 1-32 (1 byte)
#
# return:        16-bit value (2 bytes)


# Examples

# Get channel 1 (channel 1 on bank 0)
# list of command values:      01 0003 60 00 01
-SendBinaryCommand AcqSystem1 "010003600001"        # channel 1

# Get channel 33 (channel 1 on bank 1)
# list of command values:      01 0003 60 01 01
-SendBinaryCommand AcqSystem1 "010003600101"        # channel 33

# Get channel 88 (channel 24 on bank 2)
# list of command values:      01 0003 60 02 18
-SendBinaryCommand AcqSystem1 "010003600218"        # channel 88
```

## 9.32 Calibrate

### 9.32.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSCalibrate <Headstage> <Bank>** | |
|---|---|
| Performs a single data conversion on the specified channel. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSCalibrate 1 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This command can be used any time after acquisition has been started. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system which will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Bank | This value can be one of the following keywords:<br>**1**: Bank 1 (channels 1-32)<br>**2**: Bank 2 (channels 33-64)<br>**3**: Bank 3 (channels 65-96)<br>**4**: Bank 4 (channels 97-128)<br>**5**: Bank 5 (channels 129-160)<br>**6**: Bank 6 (channels 161-192)<br>**7**: Bank 7 (channels 193-224)<br>**8**: Bank 8 (channels 225-256) |

### 9.32.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x61
# parameter1:    bank (1 byte) - a bank represents one 32-ch Intan chip


# Examples

# list of command values:      01 0002 61 00
-SendBinaryCommand AcqSystem1 "0100026100"        # bank 0

# -SendBinaryCommand AcqSystem1 "0100026101"        # bank 1
# -SendBinaryCommand AcqSystem1 "0100026102"        # bank 2
# -SendBinaryCommand AcqSystem1 "0100026103"        # bank 3
# -SendBinaryCommand AcqSystem1 "0100026104"        # bank 4
# -SendBinaryCommand AcqSystem1 "0100026105"        # bank 5
# -SendBinaryCommand AcqSystem1 "0100026106"        # bank 6
# -SendBinaryCommand AcqSystem1 "0100026107"        # bank 7
```

## 9.33 ClearCalibration

### 9.33.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSClearCalibration <Headstage> <Bank>** | |
|---|---|
| Performs a single data conversion on the specified channel. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSClearCalibration 1 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This command can be used any time after acquisition has been started. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system which will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Bank | This value can be one of the following keywords:<br>**1**: Bank 1 (channels 1-32)<br>**2**: Bank 2 (channels 33-64)<br>**3**: Bank 3 (channels 65-96)<br>**4**: Bank 4 (channels 97-128)<br>**5**: Bank 5 (channels 129-160)<br>**6**: Bank 6 (channels 161-192)<br>**7**: Bank 7 (channels 193-224)<br>**8**: Bank 8 (channels 225-256) |

### 9.33.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x62
# parameter1:    bank (1 byte) - a bank represents one 32-ch Intan chip


# Examples

# list of command values:      01 0002 62 00
-SendBinaryCommand AcqSystem1 "0100026200"        # bank 0

# -SendBinaryCommand AcqSystem1 "0100026201"        # bank 1
# -SendBinaryCommand AcqSystem1 "0100026202"        # bank 2
# -SendBinaryCommand AcqSystem1 "0100026203"        # bank 3
# -SendBinaryCommand AcqSystem1 "0100026204"        # bank 4
# -SendBinaryCommand AcqSystem1 "0100026205"        # bank 5
# -SendBinaryCommand AcqSystem1 "0100026206"        # bank 6
# -SendBinaryCommand AcqSystem1 "0100026207"        # bank 7
```

## 9.34 SetOffsetRemovalEnabled

### 9.34.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetOffsetRemovalEnabled <Headstage> <Bank> <Value>** | |
|---|---|
| Enables or disables the DSP Offset removal in the Intan chip. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetOffsetRemovalEnabled 1 1 1 | |
| **Default:** The internal offset removal filter is by default enabled. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Bank | This value can be one of the following keywords:<br>**1**: Bank 1 (channels 1-32)<br>**2**: Bank 2 (channels 33-64)<br>**3**: Bank 3 (channels 65-96)<br>**4**: Bank 4 (channels 97-128)<br>**5**: Bank 5 (channels 129-160)<br>**6**: Bank 6 (channels 161-192)<br>**7**: Bank 7 (channels 193-224)<br>**8**: Bank 8 (channels 225-256) |
| Value | This value can be one of the following keywords:<br>**1**: Enables the internal Electrode Impedance Measurement Circuitry.<br>**0**: Disables the internal Electrode Impedance Measurement Circuitry. |

### 9.34.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0003
# command ID:    x64
# parameter1:    bank (1 byte) - a bank represents one 32-ch Intan chip
# parameter2:    enable (1 byte)

# default is enabled = 1


# Examples

# list of command values:      01 0003 64 00 00
-SendBinaryCommand AcqSystem1 "010003640000"    # disable on bank 0

# list of command values:      01 0003 64 02 01
-SendBinaryCommand AcqSystem1 "010003640201"    # enable on bank 2
```

## 9.35 SetOffsetRemovalMultiplier

### 9.35.1 Connect-to-SX

<table>
<tr><td colspan="2">-SendLynxSXCommand <Hardware Sub System Name> -WHSSetOffsetRemovalMultiplier <Headstage> <Bank> <Frequency></td></tr>
<tr><td colspan="2">Sets the DSP Offset removal frequency in the Intan chip.</td></tr>
<tr><td colspan="2">Example: -SendLynxSXCommand AcqSystem1 –WHSSetOffsetRemovalMultiplier 1 1 0</td></tr>
<tr><td colspan="2">Default: The internal offset removal filter multiplier is by default set to 16.</td></tr>
<tr><td colspan="2">Usage: This Command can be used after the Digital Lynx SX Hardware Sub System has been created.</td></tr>
<tr><td colspan="2">Arguments:</td></tr>
<tr><td>Hardware Subsystem Name</td><td>Name of sub system that will be controlled.</td></tr>
<tr><td>Headstage</td><td>This value can be one of the following keywords:<br>1: Selects FreeLynx #1 as the command target.<br>2: Selects FreeLynx #2 as the command target.</td></tr>
<tr><td>Bank</td><td>This value can be one of the following keywords:<br>1: Bank 1 (channels 1-32)<br>2: Bank 2 (channels 33-64)<br>3: Bank 3 (channels 65-96)<br>4: Bank 4 (channels 97-128)<br>5: Bank 5 (channels 129-160)<br>6: Bank 6 (channels 161-192)<br>7: Bank 7 (channels 193-224)<br>8: Bank 8 (channels 225-256)</td></tr>
<tr><td>Frequency</td><td>Use the following table to set the integer selection for the desired frequency.<br><br>
<table>
<tr><th>Integer 1-16</th><th>Offset Removal Cutoff Frequency (fc)</th></tr>
<tr><td>1</td><td>differentiator mode</td></tr>
<tr><td>2</td><td>0.1103 * fsample</td></tr>
<tr><td>3</td><td>0.04579 * fsample</td></tr>
<tr><td>4</td><td>0.02125 * fsample</td></tr>
<tr><td>5</td><td>0.01027 * fsample</td></tr>
<tr><td>6</td><td>0.005053 * fsample</td></tr>
<tr><td>7</td><td>0.002506 * fsample</td></tr>
<tr><td>8</td><td>0.001248 * fsample</td></tr>
<tr><td>9</td><td>0.0006229 * fsample</td></tr>
<tr><td>10</td><td>0.0003112 * fsample</td></tr>
<tr><td>11</td><td>0.0001555 * fsample</td></tr>
<tr><td>12</td><td>0.00007773 * fsample</td></tr>
<tr><td>13</td><td>0.00003886 * fsample</td></tr>
<tr><td>14</td><td>0.00001943 * fsample</td></tr>
<tr><td>15</td><td>0.000009714 * fsample</td></tr>
<tr><td>16</td><td>0.4857 e</td></tr>
</table>
</td></tr>
</table>

### 9.35.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0003
# command ID:    x65
# parameter1:    bank (1 byte) - a bank represents one 32-ch Intan chip
# parameter2:    select (1 byte)
#                range 1-16 (1 is more attenuation, 16 is less)
#                default = 16


# Examples

# set bank 0 to default 16 = x10
# list of command values:      01 0003 65 00 10
-SendBinaryCommand AcqSystem1 "010003650010"

# set bank 1 to 11 = x0B
# list of command values:      01 0003 65 01 0B
-SendBinaryCommand AcqSystem1 "01000365010B"
```

## 9.36 SetHighCutFilterFrequency

### 9.36.1 Connect-to-SX

<table>
<tr><td colspan="2">-SendLynxSXCommand <Hardware Sub System Name> -WHSSetHighCutFilterFrequency <Headstage> <Bank> <Frequency></td></tr>
<tr><td colspan="2">Sets the high cut filter frequency in the Intan chip.</td></tr>
<tr><td colspan="2">Example: -SendLynxSXCommand AcqSystem1 –WHSSetHighCutFilterFrequency 1 1 9000</td></tr>
<tr><td colspan="2">Default: The internal high cut filter is by default set to 8500.</td></tr>
<tr><td colspan="2">Usage: This Command can be used after the Digital Lynx SX Hardware Sub System has been created.</td></tr>
<tr><td colspan="2">Arguments:</td></tr>
<tr><td>Hardware Subsystem Name</td><td>Name of sub system that will be controlled.</td></tr>
<tr><td>Headstage</td><td>This value can be one of the following keywords:<br>1: Selects FreeLynx #1 as the command target.<br>2: Selects FreeLynx #2 as the command target.</td></tr>
<tr><td>Bank</td><td>This value can be one of the following keywords:<br>1: Bank 1 (channels 1-32)<br>2: Bank 2 (channels 33-64)<br>3: Bank 3 (channels 65-96)<br>4: Bank 4 (channels 97-128)<br>5: Bank 5 (channels 129-160)<br>6: Bank 6 (channels 161-192)<br>7: Bank 7 (channels 193-224)<br>8: Bank 8 (channels 225-256)</td></tr>
<tr><td>Frequency</td><td>Use the following table to set the desired frequency.<br><table><tr><th>High Cutoff Frequencies (Hz)</th></tr><tr><td>100</td></tr><tr><td>150</td></tr><tr><td>200</td></tr><tr><td>250</td></tr><tr><td>300</td></tr><tr><td>500</td></tr><tr><td>750</td></tr><tr><td>1000</td></tr><tr><td>1500</td></tr><tr><td>2000</td></tr><tr><td>2500</td></tr><tr><td>3000</td></tr><tr><td>5000</td></tr><tr><td>6000</td></tr><tr><td>7500</td></tr><tr><td>8500</td></tr><tr><td>9000</td></tr><tr><td>10000</td></tr><tr><td>15000</td></tr><tr><td>20000</td></tr></table></td></tr>
</table>

### 9.36.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0006 to x0008
# command ID:    x66
# parameter1:    bank (1 byte) - a bank represents one 32-ch Intan chip
# parameter2:    3-5 ASCII character, null terminated string (4-6 bytes)
#                The available high cut frequencies are limited to the
#                following (see Intan data sheet):
#                    100 to ASCII    = x31 x30 x30 x00
#                    150 to ASCII    = x31 x35 x30 x00
#                    200 to ASCII    = x32 x30 x30 x00
#                    250 to ASCII    = x32 x35 x30 x00
#                    300 to ASCII    = x33 x30 x30 x00
#                    500 to ASCII    = x35 x30 x30 x00
#                    750 to ASCII    = x37 x35 x30 x00
#                    1000 to ASCII   = x31 x30 x30 x30 x00
#                    1500 to ASCII   = x31 x35 x30 x30 x00
#                    2000 to ASCII   = x32 x30 x30 x30 x00
#                    2500 to ASCII   = x32 x35 x30 x30 x00
#                    3000 to ASCII   = x33 x30 x30 x30 x00
#                    5000 to ASCII   = x35 x30 x30 x30 x00
#                    6000 to ASCII   = x36 x30 x30 x30 x00
#                    7500 to ASCII   = x37 x35 x30 x30 x00
#                    8500 to ASCII   = x38 x35 x30 x30 x00 (default)
#                    9000 to ASCII   = x39 x30 x30 x30 x00
#                    10000 to ASCII  = x31 x30 x30 x30 x30 x00
#                    15000 to ASCII  = x31 x35 x30 x30 x30 x00
#                    20000 to ASCII  = x32 x30 x30 x30 x30 x00


# Examples

# set bank 0 to default 8500 Hz
# 8500 to ASCII = x38 x35 x30 x30 x00
# list of command values:      01 0007 66 00 38 35 30 30 00
-SendBinaryCommand AcqSystem1 "01000766003835303000"

# set bank 2 to 750 Hz
# 750 to ASCII = x37 x35 x30 x00
# list of command values:      01 0006 66 02 37 35 30 00
-SendBinaryCommand AcqSystem1 "010006660237353000"
```

## 9.37 SetLowCutFilterFrequency

### 9.37.1 Connect-to-SX

<table>
<tr><td colspan="2">-SendLynxSXCommand <Hardware Sub System Name> -WHSSetLowCutFilterFrequency <Headstage> <Bank> <Frequency></td></tr>
<tr><td colspan="2">Sets the high cut filter frequency in the Intan chip.</td></tr>
<tr><td colspan="2">Example: -SendLynxSXCommand AcqSystem1 –WHSSetLowCutFilterFrequency 1 1 200</td></tr>
<tr><td colspan="2">Default: The internal low cut filter is by default set to 0.1.</td></tr>
<tr><td colspan="2">Usage: This Command can be used after the Digital Lynx SX Hardware Sub System has been created.</td></tr>
<tr><td colspan="2">Arguments:</td></tr>
<tr><td>Hardware Subsystem Name</td><td>Name of sub system that will be controlled.</td></tr>
<tr><td>Headstage</td><td>This value can be one of the following keywords:<br>1: Selects FreeLynx #1 as the command target.<br>2: Selects FreeLynx #2 as the command target.</td></tr>
<tr><td>Bank</td><td>This value can be one of the following keywords:<br>1: Bank 1 (channels 1-32)<br>2: Bank 2 (channels 33-64)<br>3: Bank 3 (channels 65-96)<br>4: Bank 4 (channels 97-128)<br>5: Bank 5 (channels 129-160)<br>6: Bank 6 (channels 161-192)<br>7: Bank 7 (channels 193-224)<br>8: Bank 8 (channels 225-256)</td></tr>
<tr><td>Frequency</td><td>Use the following table to set the desired frequency.
<table>
<tr><th>Low Cutoff Frequencies (Hz)</th></tr>
<tr><td>0.1</td></tr>
<tr><td>0.25</td></tr>
<tr><td>0.3</td></tr>
<tr><td>0.5</td></tr>
<tr><td>0.75</td></tr>
<tr><td>1</td></tr>
<tr><td>1.5</td></tr>
<tr><td>2</td></tr>
<tr><td>2.5</td></tr>
<tr><td>3</td></tr>
<tr><td>5</td></tr>
<tr><td>7.5</td></tr>
<tr><td>10</td></tr>
<tr><td>15</td></tr>
<tr><td>20</td></tr>
<tr><td>25</td></tr>
<tr><td>30</td></tr>
<tr><td>50</td></tr>
<tr><td>75</td></tr>
<tr><td>100</td></tr>
<tr><td>150</td></tr>
<tr><td>200</td></tr>
<tr><td>250</td></tr>
<tr><td>300</td></tr>
<tr><td>500</td></tr>
<tr><td>600</td></tr>
</table></td></tr>
</table>

### 9.37.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0005 to x0007
# command ID:    x67
# parameter1:    bank (1 byte) - a bank represents one 32-ch Intan chip
# parameter2:    2-4 ASCII character, null terminated string (3-5 bytes)
#                The available low cut frequencies are limited to the
#                following (see Intan data sheet):
#                    0.1 to ASCII     = 30 2E 31 00
#                    0.25 to ASCII    = 30 2E 32 35 00
#                    0.3 to ASCII     = 30 2E 33 00
#                    0.5 to ASCII     = 30 2E 35 00
#                    0.75 to ASCII    = 30 2E 37 35 00
#                    1 to ASCII       = 31 00
#                    1.5 to ASCII     = 31 2E 35 00
#                    2 to ASCII       = 32 00
#                    2.5 to ASCII     = 32 2E 35 00
#                    3 to ASCII       = 33 00
#                    5 to ASCII       = 35 00
#                    7.5 to ASCII     = 37 2E 35 00
#                    10 to ASCII      = 31 30 00
#                    15 to ASCII      = 31 35 00
#                    20 to ASCII      = 32 30 00
#                    25 to ASCII      = 32 35 00 (default)
#                    30 to ASCII      = 33 30 00
#                    50 to ASCII      = 35 30 00
#                    75 to ASCII      = 37 35 00
#                    100 to ASCII     = 31 30 30 00
#                    150 to ASCII     = 31 35 30 00
#                    200 to ASCII     = 32 30 30 00
#                    250 to ASCII     = 32 35 30 00
#                    300 to ASCII     = 33 30 30 00
#                    500 to ASCII     = 35 30 30 00
#                    600 to ASCII     = 36 30 30 00


# Examples

# set bank 0 to default 25 Hz
# 25 to ASCII = x32 x35 x00
# list of command values:      01 0005 67 00 32 35 00
-SendBinaryCommand AcqSystem1 "0100056700323500"

# set bank 1 to 0.5 Hz
# 0.5 to ASCII = x30 x2E x35 x00
# list of command values:      01 0006 67 01 30 2E 35 00
-SendBinaryCommand AcqSystem1 "0100066701302E3500"
```

## 9.38 SetImpedanceMeasureEnabled

### 9.38.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetImpedanceMeasureEnabled <Headstage> <Value>** | |
|---|---|
| Enables or disables the internal Electrode Impedance Measurement Circuitry on the specified FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetImpedanceMeasureEnabled 2 1 | |
| **Default:** The internal Electrode Impedance Measurement Circuitry is by default disabled. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Value | This value can be one of the following keywords:<br>**1**: Enables the internal Electrode Impedance Measurement Circuitry.<br>**0**: Disables the internal Electrode Impedance Measurement Circuitry. |

### 9.38.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x68
# parameter1:    enable (1 byte)
#                default disabled = 0


# Examples

# list of command values:      01 0002 68 00
-SendBinaryCommand AcqSystem1 "0100026800"    # disable (default)

# list of command values:      01 0002 68 01
-SendBinaryCommand AcqSystem1 "0100026801"    # enable
```

## 9.39 SetImpedanceMeasureCurrent

### 9.39.1 Connect-to-SX

<table>
<tr><td colspan="2">-SendLynxSXCommand <Hardware Sub System Name> -WHSSetImpedanceMeasureCurrent <Headstage> <Value></td></tr>
<tr><td colspan="2">Sets the current that the multiplexing headstage will use for its internal Electrode Impedance Measurement Circuitry.</td></tr>
<tr><td colspan="2">Example: -SendLynxSXCommand AcqSystem1 –WHSSetImpedanceMeasureCurrent 2 3</td></tr>
<tr><td colspan="2">Default: The Impedance Measurement Current Value is by default set to 1, but the internal Electrode Impedance Measurement Circuitry is by default disabled.</td></tr>
<tr><td colspan="2">Usage: This Command can be used after the Digital Lynx SX Hardware Sub System has been created.</td></tr>
<tr><td colspan="2">Arguments:</td></tr>
<tr><td>Hardware Subsystem Name</td><td>Name of sub system that will be controlled.</td></tr>
<tr><td>Headstage</td><td>This value can be one of the following keywords:<br>1: Selects FreeLynx #1 as the command target.<br>2: Selects FreeLynx #2 as the command target.</td></tr>
<tr><td>Value</td><td>This value can be one of the following keywords:
<table>
<tr><th>Value</th><th>Current</th><th></th></tr>
<tr><td>1</td><td>0.128</td><td>nA</td></tr>
<tr><td>2</td><td>0.257</td><td>nA</td></tr>
<tr><td>3</td><td>0.385</td><td>nA</td></tr>
<tr><td>4</td><td>1.28</td><td>nA</td></tr>
<tr><td>5</td><td>2.57</td><td>nA</td></tr>
<tr><td>6</td><td>3.85</td><td>nA</td></tr>
<tr><td>7</td><td>12.8</td><td>nA</td></tr>
<tr><td>8</td><td>25.7</td><td>nA</td></tr>
<tr><td>9</td><td>38.5</td><td>nA</td></tr>
</table></td></tr>
</table>

### 9.39.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x69
# parameter1:    current select (1 byte)
#                range 1-9
#                select        current
#                   1            0.128 nA
#                   2            0.257 nA (default)
#                   3            0.385 nA
#                   4            1.280 nA
#                   5            2.570 nA
#                   6            3.850 nA
#                   7            12.80 nA
#                   8            25.70 nA
#                   9            38.50 nA


# Examples

# Set to default 0.257 nA
# list of command values:      01 0002 69 02
-SendBinaryCommand AcqSystem1 "0100026902"
```

## *9.40 SetImpedanceMeasureChannel*

### 9.40.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetImpedanceMeasureChannel <Headstage> <Value>** | |
|---|---|
| Sets the channel that the multiplexing headstage will connect to its internal Electrode Impedance Measurement Circuitry. | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSSetImpedanceMeasureChannel 2 38 | |
| **Default:** The Impedance Measurement Channel is by default set to 1, but the internal Electrode Impedance Measurement Circuitry is by default disabled. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Value | This value can be one of the following keywords:<br>**1-256** |

### 9.40.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x6A
# parameter1:    channel (1 byte)
#                range 0-255
#
# * note * With custom AFEs, channel mapping may vary


# Examples

# list of command values:      01 0002 6A 00
-SendBinaryCommand AcqSystem1 "0100026A00"    # channel 0

# list of command values:      01 0002 6A 30
-SendBinaryCommand AcqSystem1 "0100026A30"    # channel 48
```

## 9.41 ResetToDefaults

### 9.41.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSResetToDefaults <Headstage> <Bank>** | |
|---|---|
| Resets registers to defaults in the Intan chip. See RHD2132 data sheet for default register values | |
| **Example:** -SendLynxSXCommand AcqSystem1 –WHSResetToDefaults 1 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Bank | This value can be one of the following keywords:<br>**1**: Bank 1 (channels 1-32)<br>**2**: Bank 2 (channels 33-64)<br>**3**: Bank 3 (channels 65-96)<br>**4**: Bank 4 (channels 97-128)<br>**5**: Bank 5 (channels 129-160)<br>**6**: Bank 6 (channels 161-192)<br>**7**: Bank 7 (channels 193-224)<br>**8**: Bank 8 (channels 225-256) |

### 9.41.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x6B
# parameter1:    bank (1 byte) - a bank represents one 32-ch Intan chip


# Examples

# list of command values:      01 0002 6B 00
-SendBinaryCommand AcqSystem1 "0100026B00"    # bank 0

# list of command values:      01 0002 6B 07
-SendBinaryCommand AcqSystem1 "0100026B07"    # bank 7
```

## *9.42 GetBaseIMUEnable**

### 9.42.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetBaseIMUEnable <Headstage>** | |
|---|---|
| Gets the Base IMU enabled status of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetBaseIMUEnable 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Value | This value can be one of the following keywords:<br>**1**: Enabled<br>**0**: Disabled |

### 9.42.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x25
#
# return:        enable (1 byte)


# Examples

# list of command values:      01 0001 25
-SendBinaryCommand AcqSystem1 "01000125"
```

## *9.43 SetBaseIMUEnable**

### 9.43.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetBaseIMUEnable <Headstage> <Value>** | |
|---|---|
| Enables or disables the base IMU on the specified FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSSetBaseIMUEnable 1 1 | |
| **Default:** The base IMU is by default disabled. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Value | This value can be one of the following keywords:<br>**1**: Enable<br>**0**: Disable |

### 9.43.2 Connect-to-Cheetah

This command is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0002
# command ID:    x26
# parameter1:    enable (1 byte)


# Examples

# disable base IMU
# list of command values:      01 0002 26 00
-SendBinaryCommand AcqSystem1 "0100022600"

# enable base IMU
# list of command values:      01 0002 26 01
-SendBinaryCommand AcqSystem1 "0100022601"
```

## *9.44 GetAFE_IMUEnable**

### 9.44.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetAFE_IMUEnable <Headstage>** | |
|---|---|
| Gets the AFE IMU enabled status of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetAFE_IMUEnable 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Value | This value can be one of the following keywords:<br>**1**: Enabled<br>**0**: Disabled |

### 9.44.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x27
#
# return:        enable (1 byte)


# Examples

# list of command values:      01 0001 27
-SendBinaryCommand AcqSystem1 "01000127"
```

## *9.45 SetAFE_IMUEnable**

### 9.45.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetAFE_IMUEnable <Headstage> <Value>** | |
|---|---|
| Enables or disables the AFE IMU on the specified FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSSetAFE_IMUEnable 1 1 | |
| **Default:** The AFE IMU is by default disabled. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Value | This value can be one of the following keywords:<br>**1**: Enable<br>**0**: Disable |

### 9.45.2 Connect-to-Cheetah

This command is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0002
# command ID:    x28
# parameter1:    enable (1 byte)


# Examples

# disable AFE IMU
# list of command values:      01 0002 28 00
-SendBinaryCommand AcqSystem1 "0100022800"

# enable AFE IMU
# list of command values:      01 0002 28 01
-SendBinaryCommand AcqSystem1 "0100022801"
```

## *9.46 GetIMUAccelScale*

### 9.46.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetIMUAccelScale <Headstage> <IMU>** | |
|---|---|
| Gets the IMU Accelerometer scale of the specified IMU of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetIMUAccelScale 1 0 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| IMU | This value can be one of the following keywords:<br>**0**: Base<br>**1**: AFE |
| **Return Value:** | |
| Scale | This value can be one of the following keywords:<br>**0**: ±2g<br>**1**: ±4g<br>**2**: ±8g<br>**3**: ±16g |

### 9.46.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x32
# parameter1:    IMU select (1 byte)
#                0: base IMU
#                1: AFE IMU
#
# return:        Scale (1 byte)
#                0: +/- 2g (default)
#                1: +/- 4g
#                2: +/- 8g
#                3: +/- 16g


# Examples

# Get base IMU accelerometer scale
# list of command values:      01 0002 32 00
-SendBinaryCommand AcqSystem1 "0100023200"

# Get AFE IMU accelerometer scale
# list of command values:      01 0002 32 01
-SendBinaryCommand AcqSystem1 "0100023201"
```

## 9.47 SetIMUAccelScale

### 9.47.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetIMUAccelScale <Headstage> <IMU> <Scale>** | |
|---|---|
| Sets the IMU accelerometer scale on the specified FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSSetIMUAccelScale 1 0 1 | |
| **Default:** The IMU accelerometer scale is by default set to 0. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| IMU | This value can be one of the following keywords:<br>**0**: Base<br>**1**: AFE |
| Scale | This value can be one of the following keywords:<br>**0**: ±2g<br>**1**: ±4g<br>**2**: ±8g<br>**3**: ±16g |

### 9.47.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0003
# command ID:    x33
# parameter1:    IMU select (1 byte)
#                0: base IMU
#                1: AFE IMU
# parameter2:    Scale (1 byte)
#                0: +/- 2g (default)
#                1: +/- 4g
#                2: +/- 8g
#                3: +/- 16g


# Examples

# Set base IMU to default +/- 2g
# list of command values:      01 0003 33 00 00
-SendBinaryCommand AcqSystem1 "010003330000"

# Set AFE IMU to +/- 8g
# list of command values:      01 0003 33 01 02
-SendBinaryCommand AcqSystem1 "010003330102"
```

## *9.48 GetIMUGyroScale*

### 9.48.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetIMUGyroScale <Headstage> <IMU>** | |
|---|---|
| Gets the IMU Gyroscope scale of the specified IMU of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetIMUGyroScale 1 0 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| IMU | This value can be one of the following keywords:<br>**0**: Base<br>**1**: AFE |
| **Return Value:** | |
| Scale | This value can be one of the following keywords:<br>**0**: ±250degree/s<br>**1**: ±500degree/s<br>**2**: ±1000degree/s<br>**3**: ±2000degree/s |

### 9.48.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x34
# parameter1:    IMU select (1 byte)
#                0: base IMU
#                1: AFE IMU
#
#return:         Scale in degrees per second (1 byte)
#                0: +/- 250 dps (default)
#                1: +/- 500 dps
#                2: +/- 1000 dps
#                3: +/- 2000 dps


# Examples

# Get base IMU gyro scale
# list of command values:      01 0002 34 00
-SendBinaryCommand AcqSystem1 "0100023400"

# Get AFE IMU gyro scale
# list of command values:      01 0002 34 01
-SendBinaryCommand AcqSystem1 "0100023401"
```

## 9.49 SetIMUGyroScale

### 9.49.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetIMUGyroScale <Headstage> <IMU> <Scale>** | |
|---|---|
| Sets the IMU gyroscope scale on the specified FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSSetIMUGyroScale 1 0 1 | |
| **Default:** The IMU gyroscope scale is by default set to 0. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| IMU | This value can be one of the following keywords:<br>**0**: Base<br>**1**: AFE |
| Scale | This value can be one of the following keywords:<br>**0**: ±250degree/s<br>**1**: ±500degree/s<br>**2**: ±1000degree/s<br>**3**: ±2000degree/s |

### 9.49.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0003
# command ID:    x35
# parameter1:    IMU select (1 byte)
#                0: base IMU
#                1: AFE IMU
# parameter2:    Scale in degrees per second (1 byte)
#                0: +/- 250 dps (default)
#                1: +/- 500 dps
#                2: +/- 1000 dps
#                3: +/- 2000 dps


# Examples

# Set base IMU to default +/- 250 dps
# list of command values:      01 0003 35 00 00
-SendBinaryCommand AcqSystem1 "010003350000"

# Set AFE IMU to +/- 2000 dps
# list of command values:      01 0003 35 01 03
-SendBinaryCommand AcqSystem1 "010003350103"
```

## 9.50 CalibrateIMU

### 9.50.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSCalibrateIMU <Headstage> <IMU>** | |
|---|---|
| Calibrates the IMU on the specified FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSCalibrateIMU 1 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| IMU | This value can be one of the following keywords:<br>**0**: Base<br>**1**: AFE |

### 9.50.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0003
# command ID:    x36
# parameter1:    unused (1 byte)
# parameter2:    IMU select (1 byte)
#                0: base IMU
#                1: AFE IMU


# Examples

# Calibrate base IMU
# list of command values:      01 0003 36 00 00
-SendBinaryCommand AcqSystem1 "010003360000"

# Calibrate AFE IMU
# list of command values:      01 0003 36 00 01
-SendBinaryCommand AcqSystem1 "010003360001"
```

## *9.51 GetUltrasonicMicEnable**

### 9.51.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetUltrasonicMicEnable <Headstage>** | |
|---|---|
| Gets the Ultrasonic Microphone enabled status of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetUltrasonicMicEnable 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Value | This value can be one of the following keywords:<br>**1**: Enabled<br>**0**: Disabled |

### 9.51.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x2B
#
# return:        enable (1 byte)


# Examples

# list of command values:      01 0001 2B
-SendBinaryCommand AcqSystem1 "0100012B"
```

## *9.52 SetUltrasonicMicEnable**

### 9.52.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetUltrasonicMicEnable <Headstage> <Value>** | |
|---|---|
| Enables or disables the ultrasonic microphone on the specified FreeLynx. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSSetUltrasonicMicEnable 1 0 | |
| **Default:** The ultrasonic microphone is by default disabled. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Value | This value can be one of the following keywords:<br>**1**: Enable<br>**0**: Disable |

### 9.52.2 Connect-to-Cheetah

This command is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0002
# command ID:    x2C
# parameter1:    enable (1 byte)


# Examples

# Disable Ultrasonic mic
# list of command values:      01 0002 2C 00
-SendBinaryCommand AcqSystem1 "0100022C00"

# Enable Ultrasonic mic
# list of command values:      01 0002 2C 01
-SendBinaryCommand AcqSystem1 "0100022C01"
```

## 9.53 GetUltrasonicDigitalIOPort

### 9.53.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetUltrasonicDigitalIOPort <Headstage>** | |
|---|---|
| Gets the port the Ultrasonic Microphone is enabled on of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetUltrasonicDigitalIOPort 1 0 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Port | This value can be one of the following keywords:<br>**1**: Base GPIO Port<br>**0**: AFE GPIO Port |

### 9.53.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x39
#
# return:        Mic port select (1 byte)
#                0: mic on base digital IO
#                1: mic on AFE digital IO


# Examples

# list of command values:      01 0001 39
-SendBinaryCommand AcqSystem1 "01000139"
```

## *9.54 SetUltrasonicDigitalIOPort*

### 9.54.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetUltrasonicDigitalIOPort <Headstage> <Port>** | |
|---|---|
| Selects which digital IO port the ultrasonic microphone circuit is connected to. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSSetUltrasonicDigitalIOPort 1 0 | |
| **Default:** The base IMU is by default disabled. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Port | This value can be one of the following keywords:<br>**0**: Base GPIO Port<br>**1**: AFE GPIO Port |

### 9.54.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0002
# command ID:    x3A
# parameter1:    Mic port select (1 byte)
#                0: mic on base digital IO
#                1: mic on AFE digital IO


# Examples

# Set mic to use base digital IO
# list of command values:      01 0002 3A 00
-SendBinaryCommand AcqSystem1 "0100023A00"

# Set mic to use AFE digital IO
# list of command values:      01 0002 3A 01
-SendBinaryCommand AcqSystem1 "0100023A01"
```

## *9.55 GetWiFiStreamEnable**

### 9.55.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetWiFiStreamEnable <Headstage>** | |
|---|---|
| Gets the radio data enabled status of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetWiFiStreamEnable 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Value | This value can be one of the following keywords:<br>**1**: Enabled<br>**0**: Disabled |

### 9.55.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x23
#
# return:        enable (1 byte)


# Examples

# list of command values:      01 0001 23
-SendBinaryCommand AcqSystem1 "01000123"
```

## *9.56 SetWiFiStreamEnable**

### 9.56.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetWiFiStreamEnable <Headstage> <Value>** | |
|---|---|
| Enables or disables the wireless data stream on the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSSetWiFiStreamEnable 1 0 | |
| **Default:** The radio data is by default enabled. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Value | This value can be one of the following keywords:<br>**1**: Enable<br>**0**: Disable |

### 9.56.2 Connect-to-Cheetah

This command is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0002
# command ID:    x24
# parameter1:    enable (1 byte)


# Examples

# disable WiFi streaming
# list of command values:      01 0002 24 00
-SendBinaryCommand AcqSystem1 "0100022400"

# enable WiFi streaming
# list of command values:      01 0002 24 01
-SendBinaryCommand AcqSystem1 "0100022401"
```

## *9.57 GetMicroSDrecordEnable**

### 9.57.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetMicroSDrecordEnable <Headstage>** | |
|---|---|
| Gets the data file enabled status of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetMicroSDrecordEnable 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Value | This value can be one of the following keywords:<br>**1**: Enabled<br>**0**: Disabled |

### 9.57.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x2D
#
# return:        enable (1 byte)


# Examples

# list of command values:      01 0001 2D
-SendBinaryCommand AcqSystem1 "0100012D"
```

## *9.58 SetMicroSDrecordEnable**

### 9.58.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetMicroSDrecordEnable <Headstage> <Value>** | |
|---|---|
| Enables or disables the microSD file data stream on the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSSetMicroSDrecordEnable 1 0 | |
| **Default:** This value is stored in non-volatile device memory and restored at boot time. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Value | This value can be one of the following keywords:<br>**1**: Enable<br>**0**: Disable |

### 9.58.2 Connect-to-Cheetah

This command is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0002
# command ID:    x2E
# parameter1:    enable (1 byte)


# Examples

# disable microSD card recording
# list of command values:      01 0002 2E 00
-SendBinaryCommand AcqSystem1 "0100022E00"

# enable microSD card recording
# list of command values:      01 0002 2E 01
-SendBinaryCommand AcqSystem1 "0100022E01"
```

## 9.59 SetMicroSDdataFileName

### 9.59.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSSetMicroSDdataFileName <Headstage> <Filename>** | |
|---|---|
| Sets the data file base filename of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSSetMicroSDdataFileName 1 my_recording_filename | |
| **Default:** Default filename is “default”. This value is stored in non-volatile device memory and restored at boot. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| Filename | Null-terminated variable-length ASCII string up to 256 bytes |

### 9.59.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0004 for 2 char up to x0101 for 255 chars
# command ID:    x2F
# parameter 1:   2-255 ASCII character, null terminated string
#                (3-256 bytes)
#                Nlx appends _nnn.crd to the filename. nnn is a 3-digit
#                number starting at 000 that increments when new files
#                are created.


# Examples

# Set file name to default
# default to ASCII = x64 x65 x66 x61 x75 x6C x74 x00
# list of command values:      01 0009 2F 64 65 66 61 75 6C 74 00
-SendBinaryCommand AcqSystem1 "0100092F64656661756C7400"
```

## *9.60 GetMicroSDMemAvailable*

### 9.60.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetMicroSDMemAvailable <Headstage>** | |
|---|---|
| Gets the remaining storage capacity (in bytes) of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetMicroSDMemAvailable 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Capacity | The number of bytes available is returned as a 40-bit value (5 bytes) |

### 9.60.2 Connect-to-Cheetah

This data is available in the Cheetah GUI.

```
# packet ID:     x01
# size:          x0001
# command ID:    x30
#
# return:        40-bit mem available in bytes (5 bytes)


# Examples

# list of command values:      01 0001 30
-SendBinaryCommand AcqSystem1 "01000130"
```

## *9.61 GetMicroSDRecordingStatus*

### 9.61.1 Connect-to-SX

| **-SendLynxSXCommand <Hardware Sub System Name> -WHSGetMicroSDRecordingStatus <Headstage>** | |
|---|---|
| Gets the recording status of the specified FreeLynx device. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -WHSGetMicroSDRecordingStatus 1 | |
| **Default:** This command is an action, there is no default value. | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Headstage | This value can be one of the following keywords:<br>**1**: Selects FreeLynx #1 as the command target.<br>**2**: Selects FreeLynx #2 as the command target. |
| **Return Value:** | |
| Value | This value can be one of the following keywords:<br>**1**: Recording<br>**0**: Not Recording |

### 9.61.2 Connect-to-Cheetah

```
# packet ID:     x01
# size:          x0001
# command ID:    x31
#
# return:        recording status (1 byte)
#                0: not recording to microSD
#                1: recoring to microSD

# Examples

# list of command values:      01 0001 31
-SendBinaryCommand AcqSystem1 "01000131"
```

# 10 SX Pairing Commands (Connect-to-SX only)

## 10.1 InitWHSPairing

| **-SendLynxSXCommand <Hardware Sub System Name> -InitWHSPairing <Time>** | |
|---|---|
| Puts the Digital Lynx SX in automatic pairing mode for <Time> seconds | |
| **Example:** -SendLynxSXCommand AcqSystem1 -InitWHSPairing 60 | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Time (seconds) | Range of value: 5-255 |

## 10.2 GetWHSPairedIPAddress

| **-SendLynxSXCommand <Hardware Sub System Name> -GetWHSPairedIPAddress <Index>** | |
|---|---|
| The SX can pair up to two FreeLynxs. This command returns the IP address of the FreeLynx paired at Index 0 or Index 1. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -GetWHSPairedIPAddress 0 | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Index | This value can be:<br>**0**: to get IP address of 1st paired FreeLynx<br>**1**: to get IP address of 2nd paired FreeLynx |
| **Return Value:** | |
| IP Address | Null-terminated variable-length ASCII string up to 16 bytes |

## 10.3 GetWHSPairingStatus

| **-SendLynxSXCommand <Hardware Sub System Name> -GetWHSPairingStatus <Index>** | |
|---|---|
| This command returns a 1 if a FreeLynx is paired at the specified Index and returns a 0 if not. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -GetWHSPairing Status 0 | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| Index | This value can be:<br>**0**: Returns whether FreeLynx is paired at Index 0<br>**1**: Returns whether FreeLynx is paired at Index 1 |
| **Return Value:** | |
| Pairing Status | 0: FreeLynx not paired at specified Index<br>1: FreeLynx is paired at specified Index |

## 10.4 AddPairedWHS

| **-SendLynxSXCommand <Hardware Sub System Name> -AddPairedWHS <IP Address> <Channel Capacity>** | |
|---|---|
| This command is used for manually pairing a FreeLynx | |
| **Example:** -SendLynxSXCommand AcqSystem1 -AddPairedWHS 192.168.4.101 64 | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| IP Address | This value should be on the 192.168.4.X network<br>Neuralynx recommends values for X starting at 101 |

## 10.5 RemovePairedWHS

| **-SendLynxSXCommand <Hardware Sub System Name> -RemovePairedWHS <IP Address>** | |
|---|---|
| Manually unpair a FreeLynx.<br>• If the specified address is not paired, the SX returns an error<br>• If the IP address is found at Index 0<br>a. If a FreeLynx is paired at Index 1, the FreeLynx at Index 0 is removed and the FreeLynx at Index 1 is moved to Index 0<br>b. If a FreeLynx is not paired at Index 1, the FreeLynx at Index 0 is removed<br>• If the IP address is found at Index 1, the FreeLynx is removed. The FreeLynx at Index 0 remains at Index 0. | |
| **Example:** -SendLynxSXCommand AcqSystem1 -RemovePairedWHS 192.168.4.101 | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |
| IP Address | Use standard IP address format. All paired FreeLynxs should be on the 192.168.4.X network. |

## 10.6 RemoveAllPairedWHS

| **-SendLynxSXCommand <Hardware Sub System Name> -RemoveAllPairedWHS** | |
|---|---|
| Manually unpair all FreeLynxs from the Digital Lynx SX | |
| **Example:** -SendLynxSXCommand AcqSystem1 -RemoveAllPairedWHS | |
| **Usage:** This Command can be used after the Digital Lynx SX Hardware Sub System has been created. | |
| **Arguments:** | |
| Hardware Subsystem Name | Name of sub system that will be controlled. |