RCB Intan GUI File Manipulation

The purpose of this document is to provide information regarding the reading of RHD file format data files used by the DSPW version of the Intan Technologies GUI v1.5.

The scope of this document describes the methods used to read RHD format recorded data files using MATLAB®. The MATLAB .m files and RHD format data file described below are available on the dspw.io website.

We will also show an example of how to modify recorded files that have data loss due to lost UDP packet messages.

DSPW UDP network packet structure is not the same as Intan GUI packet structure. It is easy to confuse the two. In this document packet refers to the DSPW UDP data packet that is sent over Wi-Fi from the RCB module to the PC hosting the GUI.

Save a data file

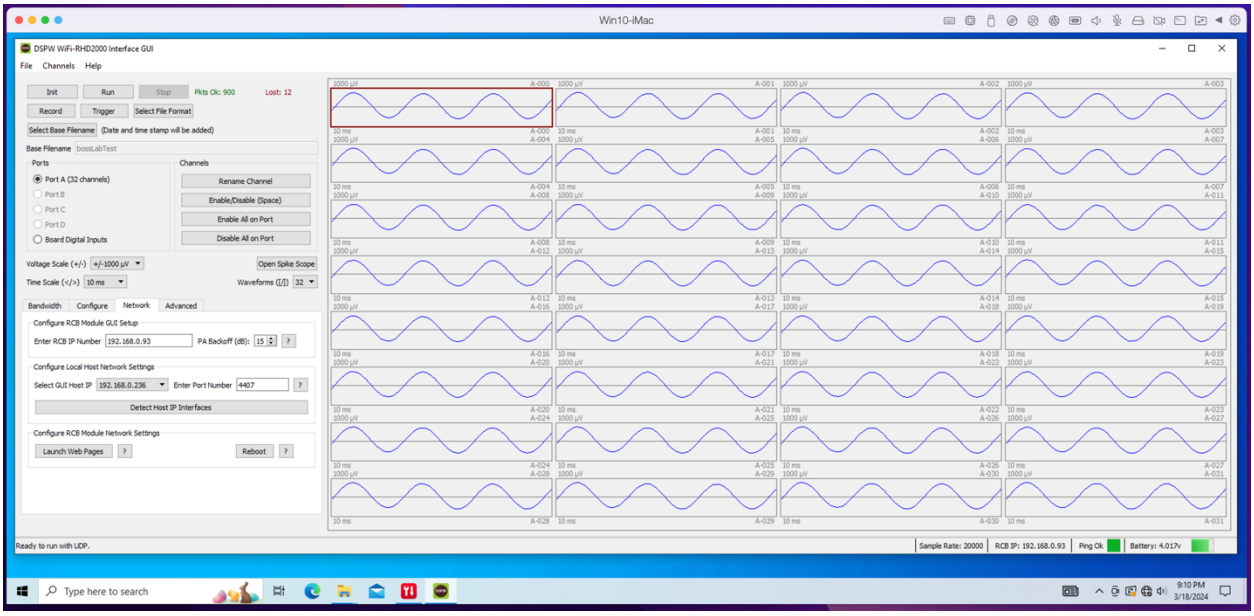
Using the DSPW Intan GUI v1.5, we recorded a data file using standard Intan GUI methods.

Intan GUI v1.5 operation is described in this Intan doc -
https://intantech.com/files/Intan_RHD2000_eval_system.pdf

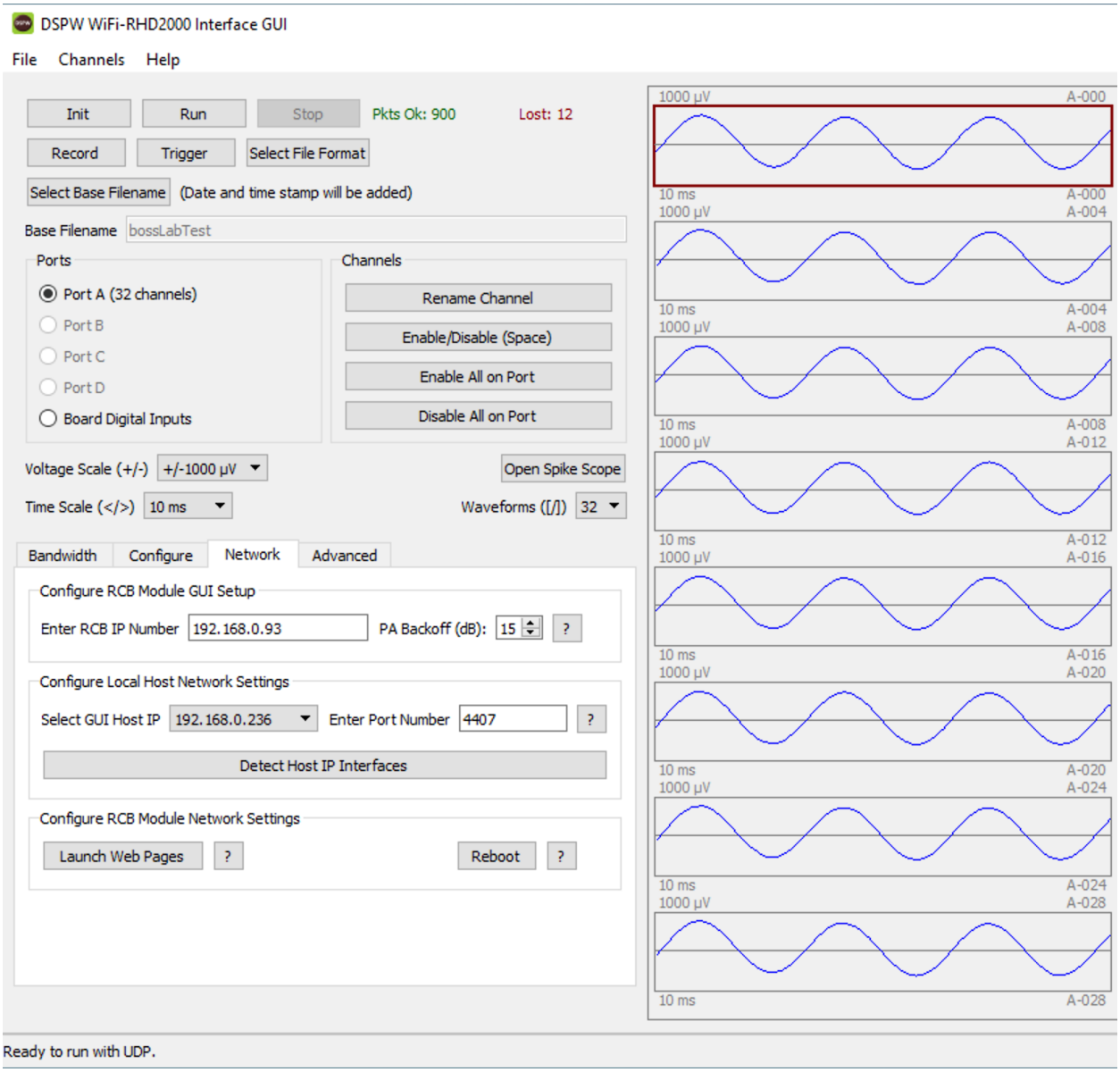
DSPW Intan GUI modifications are described on page 13 of this DSPW doc -
https://www.dspwi.com/uploads/2/4/6/7/24677982/user_guide_04.pdf

In our test recording -
32 channels each sampled at 20ksps (nominal), 20.64ksps (actual).
At 32 channels, the RCB UDP network packet contains 21 data samples for each channel.
Input signal is a 315Hz sine wave.

GUI is running on Win10 in a Parallels VM on x86 iMac.



Above image cropped so can more easily see Packets OK vs Packets Lost.
Packets Lost = 12.



Read the RHD format file

Use MATLAB to read the RHD format file using the Intan read_intan_RHD2000_file.m script.

Notice that there are 12 gaps reported in the timestamp. This indicates that 12 RCB UDP data packets have been lost and that these data samples have been lost.

Original Intan .m script

Run -> Read_intan_RHD2000_file.m
>> read_Intan_RHD2000_file -> RCB-Test_240318_202800.rhd

Reading Intan Technologies RHD2000 Data File, Version 1.5

Found 32 amplifier channels.
Found 3 auxiliary input channels.
Found 1 supply voltage channel.
Found 0 board ADC channels.
Found 0 board digital input channels.
Found 0 board digital output channels.
Found 0 temperature sensor channels.

File contains 0.977 seconds of data. Amplifiers were sampled at 20.64 kS/s.

Allocating memory for data...
Reading data from file...
10% done...
20% done...
30% done...
40% done...
50% done...
60% done...
70% done...
80% done...
90% done...
100% done...
Parsing data...
Warning: 12 gaps in timestamp data found. Time scale will not be uniform!
Done! Elapsed time: 0.0 seconds
Extracted data are now available in the MATLAB workspace.
Type 'whos' to see variables.

>> whos

Name	Size	Bytes	Class
amplifier_channels	1x32	40192	struct
amplifier_data	32x20160	516096	double
ans	21x1	168	double
aux_input_channels	1x3	4418	struct
aux_input_data	3x5040	120960	double
filename	1x26	52	chars
frequency_parameters	1x1	2640	struct
notes	1x1	504	struct
path	1x33	66	char
spike_triggers	1x32	14592	struct
supply_voltage_channels	1x1	1942	struct
supply_voltage_data	1x336	2688	double
t_amp	1x20160	161280	double
t_amplifier	1x20160	161280	double
t_aux_input	1x5040	40320	double
t_supply_voltage	1x336	2688	double

Note that t_amplifier in the recorded file, before MATLAB processing, is a sequence starting at 0 and counting by 1 for each sample.

After running the MATLAB script t_amplifier will be modified -

```
% Scale time steps (units = seconds).
    t_amplifier = t_amplifier / sample_rate;
```

For this app note we add variable t_amp so it is clear how many packets are lost.
>> t_amp = t_amplifier * frequency_parameters.amplifier_sample_rate;

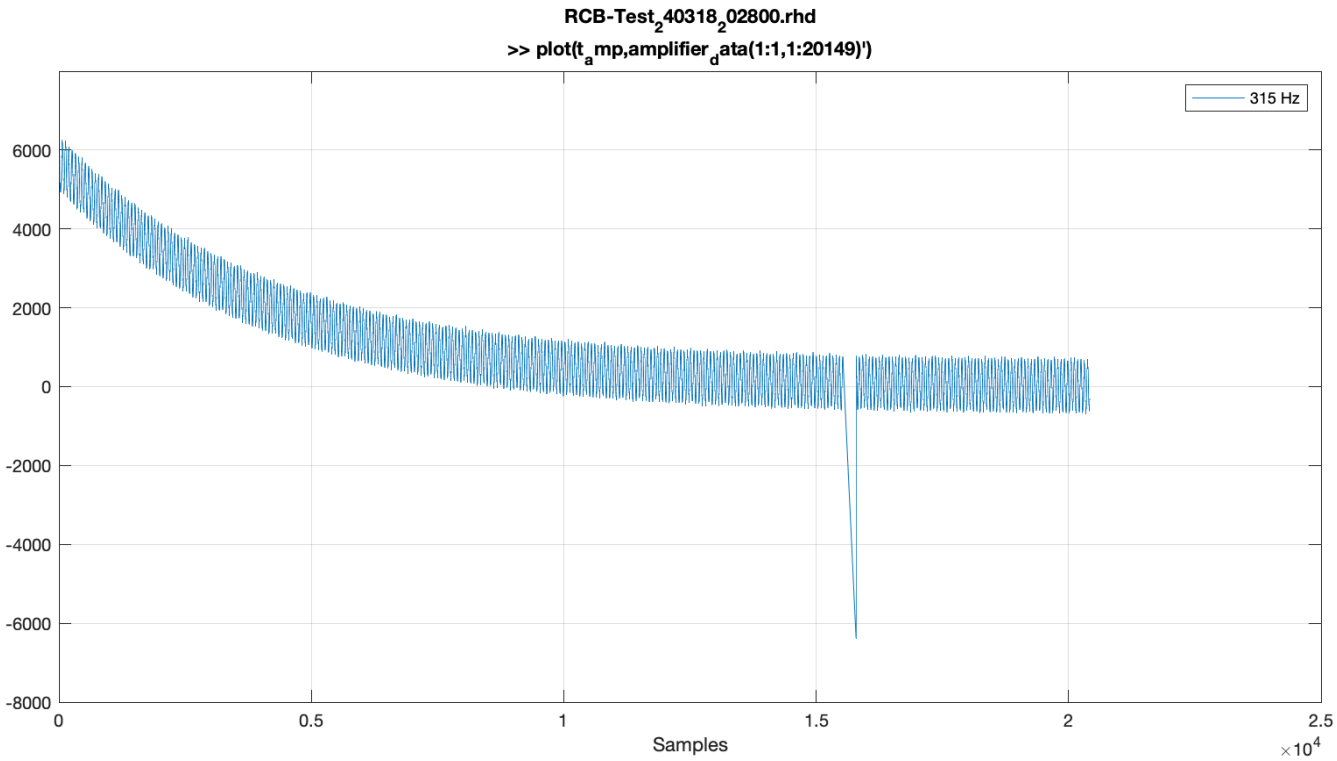
Plot RHD Data

Below is a plot of the complete RHD format data file. File contains a small section of UDP dropouts. These are samples that have been lost and cannot be recovered.

UDP packet dropouts can be seen as full scale negative data samples. We leave it up to the user how to best modify these samples for their specific post processing needs.

There are multiple factors, reasons, that you may see UDP packet loss. Please see the DSPW "How to avoid packet loss" Application Note that is available for download at dspw.io.

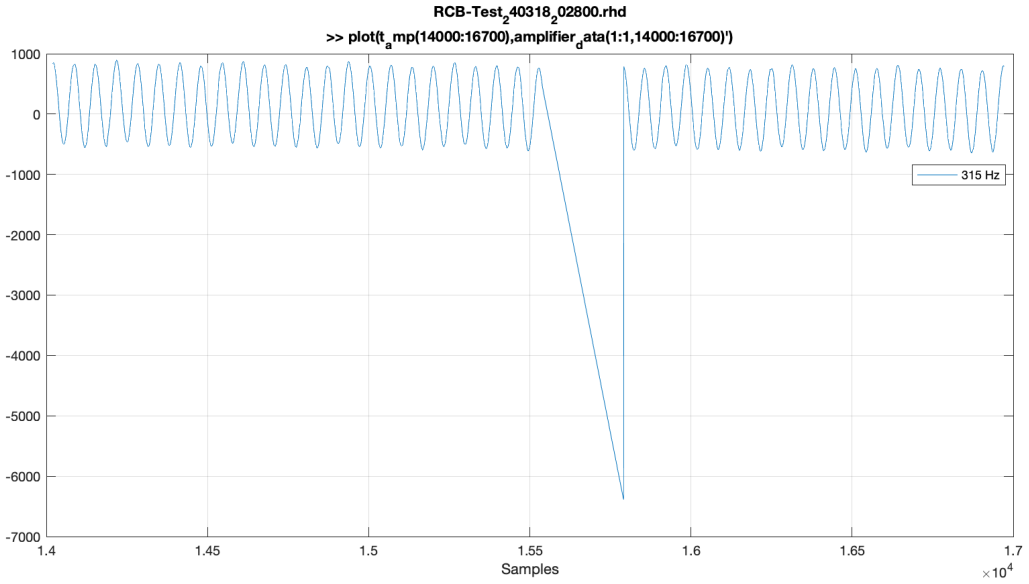
>> plot(t_amp,amplifier_data(1:1,1:20149)')



Recording shows data samples from start of a session. The initial values show data before the RHD on-chip dc removal filter settles out. When channels = 32, there will be 21 samples for each channel in each UDP packet. Since the first UDP data packet is always discarded t_amp index starts at 21.

Zoom view plot of the packet dropout area.

>> plot(t_amp(14000:16700),amplifier_data(1:1,14000:16700)')



Looking at t_amp we see 12 repeat timestamps. These correspond to the 12 lost UDP data packets. When channels = 32, there will be 21 samples per channel in each UDP packet.

Gap is 15791-15539 = 252. 252 / 21 samples per UDP packet = 12 Lost packets.

```
>> (t_amp(15515:15535)')
ans = 1.0e+04 *
   1.553500000000000
   1.553600000000000
   1.553700000000000
   1.553800000000000
   1.553900000000000
   1.579100000000000
   1.579100000000000
   1.579100000000000
   1.579100000000000
   1.579100000000000
   1.579100000000000
   1.579100000000000
   1.579100000000000
   1.579100000000000
   1.579100000000000
   1.579100000000000
   1.579200000000000
   1.579300000000000
   1.579400000000000
   1.579500000000000
```

DSPW modified Intan .m script

DSPW provides a modified Intan read_Intan_RHD2000_file.m script that will remove the redundant timestamps and data.

DSPW read_rcb_lvds_file.m file modification is shown below. This is the only DSPW modification made to the original Intan .m file.

```
% Check for gaps in timestamps.
    num_gaps = sum(diff(t_amplifier) ~= 1);
    if (num_gaps == 0)
        fprintf(1, 'No missing timestamps in data.\n');
    else
        fprintf(1, 'Warning: %d gaps in timestamp data found.  Time scale
will not be uniform!\n', ...
            num_gaps);

        num_pre = length(t_amplifier);
        [t_amplifier, index_sorted] = unique(t_amplifier);
        amplifier_data = amplifier_data(:, index_sorted);
        num_post = length(t_amplifier);
        fprintf(1, 'Removed %d duplicate timestamps.\n', num_pre-num_post);

        num_gaps = sum(diff(t_amplifier) ~= 1);
        if (num_gaps == 0)
            fprintf(1, 'Corrected: No missing timestamps in data.\n');
        else
            fprintf(1, 'Corrected: Warning: %d gaps in timestamp data found.
Time scale will not be uniform!\n', ...
                num_gaps);
        end
```

The DSPW script removes the 11 duplicate timestamps and associated data samples.

Data samples received from the Intan RHD2000 ADC (headstage) are unsigned binary, 16 bits = 0-65536, where 0 represents negative full scale.

In the Intan .m file the ADC samples are converted to microvolts using –

```
% Scale voltage levels appropriately.
    amplifier_data = 0.195 * (amplifier_data - 32768); % units = microvolts
```

Lost UDP packet samples are shown as negative full scale values so is easy to find these samples in the data file.

This may or may not be the best approach for your post processing of the data file. Ex. Setting the full scale negative sample to zero, remove the sample from the data, …

Run -> read_rcb_lvds_file.m
>> read_Intan_RHD2000_file -> RCB-Test_240318_202800.rhd

Reading Intan Technologies RHD2000 Data File, Version 1.5

Found 32 amplifier channels.
Found 3 auxiliary input channels.
Found 1 supply voltage channel.
Found 0 board ADC channels.
Found 0 board digital input channels.
Found 0 board digital output channels.
Found 0 temperature sensors channels.

File contains 0.977 seconds of data. Amplifiers were sampled at 20.64 kS/s.

Allocating memory for data...
Reading data from file...
10% done...
20% done...
30% done...
40% done...
50% done...
60% done...
70% done...
80% done...
90% done...
100% done...
Parsing data...
Warning: 12 gaps in timestamp data found. Time scale will not be uniform!
Removed 11 duplicate timestamps.
Corrected: Warning: 1 gaps in timestamp data found. Time scale will not be uniform!
Done! Elapsed time: 0.0 seconds
Extracted data are now available in the MATLAB workspace.
Type 'whos' to see variables.

>> whos

Name	Size	Bytes	Class
amplifier_channels	1x32	40192	struct
amplifier_data	32x20149	5158144	double
ans	21x1	168	double
aux_input_channels	1x3	4418	struct
aux_input_data	3x5040	120960	double
filename	1x26	52	chars
frequency_parameters	1x1	2640	struct
notes	1x1	504	struct
path	1x33	66	char
spike_triggers	1x32	14592	struct
supply_voltage_channels	1x1	1942	struct
supply_voltage_data	1x336	2688	double
t_amp	1x20149	161192	double
t_amplifier	1x20149	161192	double
t_aux_input	1x5040	40320	double
t_supply_voltage	1x336	2688	double

Notice that amplifier_data and t_amplifier are now 1 x 20149 doubles vs 1 x 20160 doubles. Redundant timestamps have been removed from the array.

When channels = 32, there will be 21 samples for each channel in each UDP packet.

Here we can see the 12 repeat timestamps are reduced to 1. Gap is 15791-15539 = 252. 252 / 21 samples per UDP packet = 12 Lost UDP packets (as expected).

```
>> (t_amp(15515:15535)')
ans =
   1.0e+04 *

   1.553500000000000
   1.553600000000000
   1.553700000000000
   1.553800000000000
   1.553900000000000
   1.579100000000000
   1.579200000000000
   1.579300000000000
   1.579400000000000
   1.579500000000000
   1.579600000000000
   1.579700000000000
   1.579800000000000
   1.579900000000000
   1.580000000000000
   1.580100000000000
   1.580200000000000
   1.580300000000000
   1.580400000000000
   1.580500000000000
   1.580600000000000
```

Since the timestamp gap is the same in both data sets the plot will look the same for both. However, the new data set does not contain the redundant timestamps.

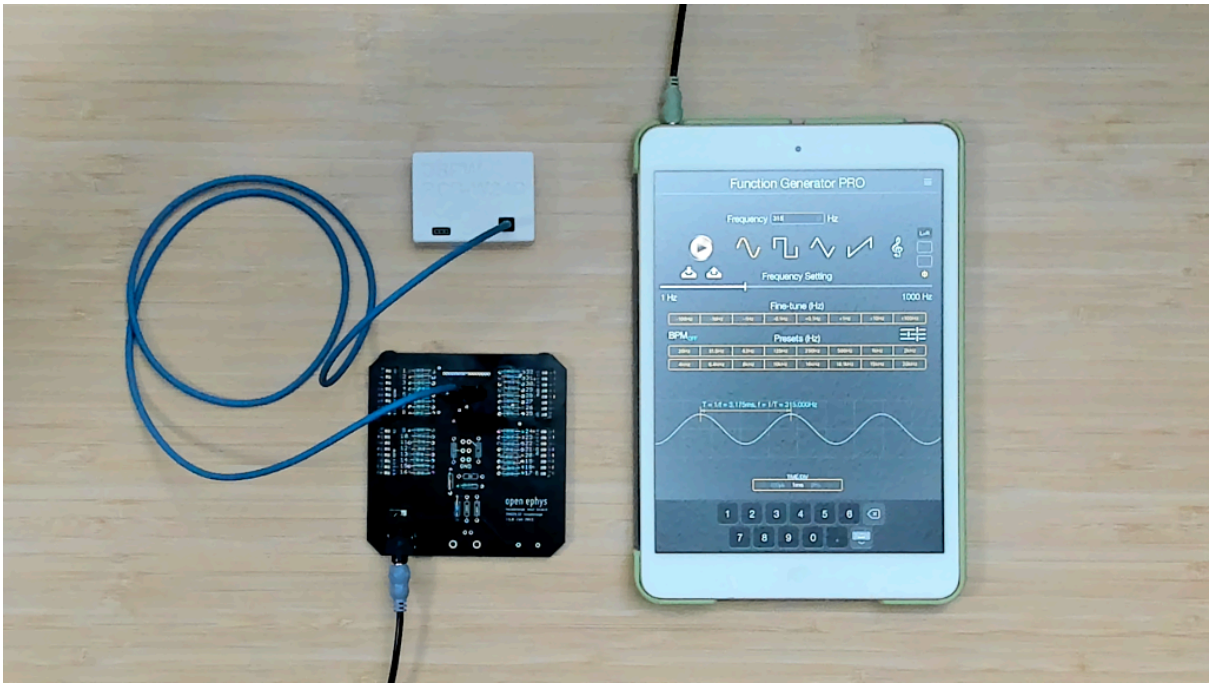
Lost UDP packet samples are shown as negative full scale values so is easy to find these samples in the data file.

Test Setup

Equipment

Photo showing items used to create the test files for this App Note.

DSPW RCB-W24B-LVDS – White box
Intan RHD2132 Headstage – connected to Blue SPI cable
Intan SPI cable – Blue cable
Open Ephys headstage test board – Black PCB with audio jack
Apple iPad Mini – rather old – Function Generator Pro App
Audio cable – Green / Blue connectors , Black cable
Wi-Fi Router – TP-Link AX21 - not shown in photo, located 20ft away from RCB device.



Files

The following files are available on the intantech.com and dspw.io websites:

MATLAB scripts
read_Intan_RHD2000_file.m
https://intantech.com/files/RHD_MATLAB_functions.zip

read_rcb_lvds_file.m
https://www.dspwi.com/uploads/2/4/6/7/24677982/read_rcb_lvds_file.m

RHD format data file
RCB-Test_240318_202800.rhd
https://www.dspwi.com/uploads/2/4/6/7/24677982/rcb-test_240318_202800.rhd

