
RASDR User Manual

1.7.5

Supporting
Software Defined Receiver for Radio
Astronomy
and
RASDRviewer 1.2.2, RASDRproc 0.2.2

RASDR User's Manual 1.7.5

supporting

Software Defined Receiver for Radio Astronomy and RASDRviewer 1.2.2, RASDRproc 0.2.2

Table of Contents

List of Figures	5
List of Tables	8
Foreword	9
License	9
Abstract	9
1) README First	10
2) RASDR goals.....	10
a) A Software-Defined Receiver for Amateur Radio Astronomers	10
b) The RASDR Community: Developer's group and User's group.....	11
c) Operation of RASDR in Various Radio Astronomy Bands.....	11
3) The RASDR Software Defined Receiver	13
a) Specifications.....	14
b) Component Board Considerations	14
c) Power and Data Transfer Considerations	15
4) Startup Guide for a RASDR Observatory.....	16
a) Installation of Signing Certificate	17
b) Installation of Device Driver.....	21
5) RASDR Hardware Basics	25
a) MyriadRF hardware	26

Layout	26
Input/output	27
Configuration [Advanced users]	28
b) DigiRED hardware	29
Input/output	29
USB Control	29
LED indicators	30
Configuration	32
Loading new firmware	32
6) RASDRViewer and RASDR Software Basics	33
a) Introduction	33
b) Installing RASDRviewer	34
c) Software Architecture	34
d) Graphical User Interface	35
e) FFT Output to Disk	37
f) Balancing DC offset for I and Q	39
7) Basic Radio Astronomy with RASDR (discussion and examples)	40
a) Hydrogen H1 Spectroscopy. L-band (1420 MHz)	40
b) System Gain Considerations	42
c) Practical System tests for H1 detection	43
Test System 1 Hydrogen H1 Spectroscopy with degraded amplifier	43
Test System 2 Hydrogen H1 Spectroscopy using a small horn antenna	45
8) Post processing to extract information from recorded data	47
9) Basic Radio Astronomy below RASDR design frequency	48
a) Radio Astronomy in the VHF band (80-190 MHz)	48
b) Band-Extension Spectroscopy Testing and application to wide-band VHF monitoring	51
c) Using the VHF converter to access the NRAO 40' dish	51
d) RASDR Spectroscopy in HF band (2-80 MHz)	51
10) Operating RASDR beyond the limits: SETI and weak signals	54
11) Appendices	56
a) RASDR Drivers	56

b)	RASDR Firmware	56
c)	Firmware Command Set	59
d)	Controlling GPIO	66
e)	RASDR Receiver Data Format and Endpoint	69
f)	Troubleshooting	71
g)	General and Frequently-Asked Questions (FAQs)	71
h)	RASDR & Myriad RF connectors details	74
	X1 Connector - DigiRed to Myriad	74
	X2 Connector - FX3 JTAG interface	77
	X3 Connector - PPS and GPIO Connector	78
i)	X7 Connector - Input Reference Frequency Provision	79
12)	References	80
13)	Index	84

List of Figures

Figure 1 RASDR functional overview.....	13
Figure 2 RASDR mounted in case.....	14
Figure 3 Typical RASDR observatory suitable for H1 observations	16
Figure 4 - Certificate Installation Step #1	19
Figure 5 - Certificate Installation Step #2.....	19
Figure 6 - Certificate Installation Step #3.....	20
Figure 7 - RASDR-authority.cer details after successful Certificate installation.....	20
Figure 8 - First Time Installation with properly installed Certificate.....	22
Figure 9 - RASDR Driver Details.....	22
Figure 10 – The user may receive an error message if RASDR inserted prior to driver update..	23
Figure 11 - Choose Device Manager and Update Driver Software	23
Figure 12 - RASDR Driver successful installed.....	24
Figure 13 - Message Displayed when RASDR is connected to a port with insufficient power delivery	24
Figure 14 - CyControl Software Details after successful installation.....	25
Figure 15 The core hardware RASDR package consists of two circuit boards.....	25
Figure 16. The Myriad-RF board was manufactured in several versions. SARA boards produced in 2014 are matched to the DigiRED board. Earlier versions may require addition of a simple dongle with extra resistors.	26
Figure 17 The MyriadRF board is the RF section of RASDR, and interfaces directly to DigiRED via the X3 connector. The SMA connectors are for transmit and receive sections. Only receive operations are currently supported by the RASDR team.....	27
Figure 18 DigiRED connection descriptions.....	29
Figure 19 Some Konig© (L) and Inateck© PCIE cards provide USB2/USB3 connectivity to PC desktop computers.	30
Figure 20 LED Indicators on digiRED board.....	31
Figure 21 (L) shows Rx and Tx LED indicators and (R) shows location of LED D9.....	31
Figure 22 GUI Start window of the RASDRViewer application	35
Figure 23 Lime Microsystems chip control settings.....	36
Figure 24 ADC Power input	37
Figure 25 FFT Recording configuration	38
Figure 26 - I and Q out of balance	39
Figure 27 - I and Q adjusted	39

Figure 28 Initial choices of RASDR2 control parameters for comparison with typical SpectraCyber results, specified minimum BW for this narrow spectral region. See text for choice of values shown in red.	41
Figure 29 The importance of spectral averaging is shown with processing of 99 frames (left) and 998 frames (right). Control settings were as shown in Figure 19 for a dark section of sky. IBM laptop running WinXP, and using a USB2 interface was used for this particular test.	41
Figure 30 This figure is the difference file using the 99 file average and the 998 file average shown in Figure 20. The 2dB noise component from Figure 20 (left) is all that remains. The common feature is removed, and the same method can be used to remove system noise or select for 'genuine' signals.	42
Figure 31 Example system configuration for H1 detection. RASDR hardware is outlined in blue.....	42
Figure 32(L) shows the "raw" averaged RASDR spectrum (center frequency 1420.407 MHz from RA19, DE-7 and (R) the "background" spectrum from RA20, DE-7, a rather thin part of the Milky Way. The "background" region was chosen using published maps and the small H1 features (see text) were not noted until after the spectral data from RA19, DE-7 on the left was processed.. The background plot has a slightly expanded scale to show the small background features. The spur appears at maximum RASDR gain settings, but disappears with background subtraction.....	44
Figure 33. Subtraction of background from raw data and averaging shows the H1 spectrum from RA19, DE-7. This figure is misleading in that the 'dark sky' background region actually contained minor H1 regions, and contributions from these regions have been unfortunately subtracted in this plot. The center frequency is 1420.405 MHz.	45
Figure 34 Test System 2 is based on a horn antenna, shown here in . Two versions were constructed. The version on the right had added aluminum foil, and lower noise.	46
Figure 35. One quarter of a 12-minute Milky Way (Ra3h, De -8 deg) data scan, with background subtraction and smoothing.	46
Figure 36. Block functions of the VHF converter show conversion of 80-190MHz VHF signals into the RASDR2 input band acceptance range.....	49
Figure 37 Completed VHF converter used with RASDR2. Input is via the coax cable while the output is via the SMA connector shown on the upper right.	49
Figure 38 Commercial FM band. The antenna was a random wire about 2m long, connected to the VHF up converter, then to RASDR2. The red marker was added on the RASDR2 display to denote a local FM station broadcasting at 102.113 MHz	50
Figure 39 RASDR2 internal narrow-banding was chosen to select a region of interest. In practice, narrow banding is useful to reject unwanted outlying signals (noise).	50
Figure 40 The complete RASDR2 video output with VHF input shows the input signal stream as digitized (I and Q values); the I vs. Q plot (a circle if there is a single coherent source), the FFT spectral display with markers, if selected, digitization and internal gain settings, and a power vs.	

time plot. The ratty power vs. time plot on the lower right shows the result of moving the antenna.	51
Figure 41 Functionality of the Nooelec up converter is as shown. The RF input is nominally specified 0-65 MHz and the measured performance covers the range 2-66 MHz. Filter functions are as described with reference to Figure 4.	52
Figure 42 Two up conversion stages are shown here, which provide for conversion of the 2-66MHz portion of the HF band to the 402-466 MHz band prior to processing with RASDR2. The Nooelec converter is on the lower left while the VHF converter described earlier appears at the upper right.	53
Figure 43 Noisy 5.5 MHz section of the HF band centered on 15 MHz. This measurement was made with an electrically short antenna and no preamplifier. It shows various sources of noise including some from the Nooelec up converter. This experiment suggests the importance of a decent antenna, more pre-amplification and background subtraction.	53
Figure 44 The Kepler mission search zone considers a tiny fraction of our galaxy.	54
Figure 45 - CyControl program (Firmware update step 1)	57
Figure 46 - CyControl program (Firmware update step 2)	57
Figure 47 - Select programming file (Firmware update step 5).....	58
Figure 48 - Firmware Update Success	58
Figure 49 - Reinsert USB after Firmware update	58
Figure 50 - GPIO Read Action (part #1)	67
Figure 51 - GPIO Read Action (part #2)	67
Figure 52 - GPIO Write Action (part #1).....	68
Figure 53 - GPIO Write Action (part #2).....	68
Figure 54 - GPIO Action with invalid pin number	69
Figure 55 - RASDR Data with PPS signal ON (FFTsize 2048)	70
Figure 56 - RASDR Data with PPS signal OFF (FFTsize 2048)	70
Figure 57 - Unrecoverable stall using USB3	73
Figure 58 JTAG conector.....	78
Figure 59 LED D9 is illuminated green to denote frequency lock.	79

List of Tables

Table 1. FCC band plan from 3kHz to 30GHz.	11
Table 2 Candidate radio astronomy research bands. Bands shown in and above HF are designated. The band shown as VLF has been proposed for radio astronomy under special conditions [18]	12
Table 3 Basic RASDR specifications are focused on receive operations from app. 0.3-3.8 GHz.	14
Table 4 Typical Limits on Sample Rate per System Specifications	15
Table 5 Example startup procedure for new RASDR observatory	17
Table 6 MyriadRF specifications.....	27
Table 7. Myriad-RF Board Connector Assignments	28
Table 8 DigiRED board connectors and switches.	29
Table 9 DigiRED LED status indicators.....	32
Table 10. Examples of system gain from components for monitoring H1 signals. The buffer amplifier suggested here permits lower gain settings of RASDR, while still giving good H1 spectra (see next section).	43
Table 11 - RASDR receiver USB IDs	56
Table 12 X3 connector pin description	76
Table 13 - GPIO Assignments	78

Foreword

Thanks to the Society of Amateur Radio Astronomers (SARA) for supporting RASDR as the project has migrated through three different hardware approaches and both design and production challenges. This project is a community open-source effort for the benefit of SARA members. We expect that the fruits of the volunteer labor will be useful to the amateur radio astronomy, educational, and scientific communities. The RASDR hardware will be available at cost to SARA members and affiliates.

RASDR software is in a continual process of upgrading, with the intent of adding features. Please communicate with the RASDR group at RASDRapplications@radio-astronomy.org.

Sincere thanks to all on the RASDR team who contributed to this project.

License

Unless otherwise specified, all software source files are licensed using the GNU General Public License Version 2. To read details of the license please visit <http://www.gnu.org/licenses>. All hardware related designs, source artifacts and documentation are licensed under the “Creative Commons Attribution 3.0 Unported”. To read the details of the license please visit <http://creativecommons.org/licenses/by/3.0/legalcode>.

Abstract

RASDR is a software-defined receiver (SDR) for radio astronomy [1]. The device consists of two high-density circuit boards, separated by analog and digital operations. The analog board contains a wide-band radio frequency (RF) to baseband transceiver used for femtocell telecommunications [2] [3] applications. The digital board is a control and data interface to a computer via universal serial bus (USB). The RASDRviewer/RASDRproc software runs in a Windows environment and performs receiver control, fast-Fourier transform (FFT) analysis and spectrum averaging, power monitoring and data collection. The user is responsible for supplying an antenna, preamplifier, band pass filter, optional up conversion or system control devices, and external frequency/time reference signals.

1) README First

Please read [Chapter 2](#) through [Chapter 4](#) before starting to connect and use your RASDR. Please join SARA [1] and the User's group. The SARA RASDR pages [4] [1] provide news and new information. The RASDR User's Group [5] and the RASDR /Discourse [6] are good resources to obtain community support and help. The development sites [4] [7] are available for people who wish to help contribute or to use the design for their own purposes, including RASDR software development.

NOTE: where 'RASDRviewer' is mentioned in the text, and unless otherwise noted, this refers to both 'RASDRviewer' (a proprietary program maintained by Paul Oxley) and 'RASDRproc' (the open-source program maintained on the github [7]).

Users having general questions should post them to one of the User sites. Specific questions may be sent to members of the RASDR team at the following addresses:

Software issues:

Contact RASDRviewer@radio-astronomy.org

Hardware issues:

Contact RASDRhardware@radio-astronomy.org or discuss the issue with the RASDR /Discourse [6] on the MyriadRF forum, <https://discourse.myriadrf.org/c/projects/rasdr>

Application issues:

Contact RASDRapplications@radio-astronomy.org or discuss the issue with the User's Group on Yahoo, <https://groups.yahoo.com/neo/groups/RASDR/info>

2) RASDR goals

This section highlights some of the goals of the RASDR project.

a) **A Software-Defined Receiver for Amateur Radio Astronomers**

RASDR is SDR that is optimized for radio astronomy, suitable for a wide variety of SARA projects, and its functionality extends from a front-end receiver through a software-driven desktop computer back end [4] [8]. The receiver uses a computer chip containing the entire RF digital receiver chain designated LMS6002D [9] [10]. The back-end computer controls the front end and enables the user to control receiver functions, display signals and perform analysis functions (averaging, computation of spectrograms, determination of power time-spectrum, and generating output files).

Signal processing permits dealing with low signal-to-noise ratio (S/N) information, and presents data as spectral plots and data files. RASDRviewer software [11] [12] for the Windows OS,

performs parameter optimizations, user control, spectra output, power characterization and output data formatting via a Graphical User Interface (GUI).

This design evolution is based on the need to have widest possible data pipeline speed for radio astronomy applications, and to make this available to SARA members at cost to produce [13].

b) **The RASDR Community: Developer's group and User's group**

It is recommended that the user begin by becoming a member of SARA [14]. On the home page, there is a tab for RASDR at the top and a graphic for the RASDR open source history on the right side of that page.

There is a RASDR User's Group [5] and a MyriadRF-sponsored RASDR /Discourse [6] forum. There is no cost to join these groups, and you may get questions answered by the community of like-minded folks.

c) **Operation of RASDR in Various Radio Astronomy Bands**

The FCC spectral designations from 3kHz to 30 GHz [15] are shown in Table 1:

Band Number	Symbol	Frequency Range	Wavelength Range
4	<u>VLF</u>	3 to 30 kHz	10 to 100 km
5	<u>LF</u>	30 to 300 kHz	1 to 10 km
6	<u>MF</u>	300 to 3000 kHz	100 to 1000 m
7	<u>HF</u>	3 to 30 MHz	10 to 100 m
8	<u>VHF</u>	30 to 300 MHz	1 to 10 m
9	<u>UHF</u>	300 to 3000 MHz	10 cm to 1 m
10	<u>SHF</u>	3 to 30 GHz	1 to 10 cm

Table 1. FCC band plan from 3kHz to 30GHz.

Development of these bands for radio astronomy research is shown in Table 2. It is taken from various sources and contains comments by the authors. The Jovian kilometric band is not often considered, but it has been observed by Voyager and other space probes, and may be a valid radio astronomy band from earth under certain circumstances [16] [17]. RASDR, along with ancillary equipment discussed in this guide, has been used from 2MHz-3.8GHz.

RASDR2 operates in bands 9-10 (up to 4 GHz), but with additional components the coverage may be considerably extended, as will be discussed beginning on page 40.

Ionospheric Monitoring		ULF-VLF	0.00030 to 0.003
Jovian VLF [10,11]	Jovian kilometric	VLF	.015 to .025
Jovian and Solar	RASDR upconverter F<400MHz	HF	10 to 35
Solar / Solar Wind		VHF	70
Pulsar	1 second for pulse shape	VHF	70
	0.05 Seconds bin width	VHF	150
	Clock Discipline 0.1 Second	UHF	322
		UHF	406
Continuum, VLBI, Solar	Consider Dicke Switching	UHF	608
21 cm Hydrogen	Galactic doppler +/- 1.4 MHz	L-Band	1420
	Extra Galactic +/- 5 MHz		
Solar Continuum		L-Band	1420
OH spectral	galactic doppler +/- 1.6 MHz	L-Band	1665/1667
OH spectral, SETI		L-Band	1702
Continuum, Solar		S-Band	2655
		S-Band	2690
	Calibration to NOAA obs	S-Band	2800
Continuum, Solar		S-Band	3260
		S-Band	3332
		S-Band	3345
	LNB required for RASDR	C-Band	4800
	operation above 3800 MHz	C-Band	4990
		C-Band	5000
		C-Band	6650
		X-Band	10600
Methanol / Doppler	Doppler +/- 12 MHz LNB Req	C-Band	6670
		KU-Band	12178
Methanol Maser Activity	Narrow Band LNB Required	KU-Band	14500

Table 2 Candidate radio astronomy research bands. Bands shown in and above HF are designated. The band shown as VLF has been proposed for radio astronomy under special conditions [18]

RASDR consists of a Myriad RF board (RF Transceiver) and a DigiRED board (clock distribution and USB interface). The RASDR functional overview is shown in Figure 1. Note that RASDR does not include an antenna, LNA, secondary amplification, external frequency reference, computer, etc. Figure 2 shows the hardware packaged into a prototype metal case. Most RASDR cases use inexpensive decals rather than the ink jet process shown here.

RASDR is a versatile instrument, but the user must tailor it to specific applications by including the necessary supporting components to build the desired observing system.

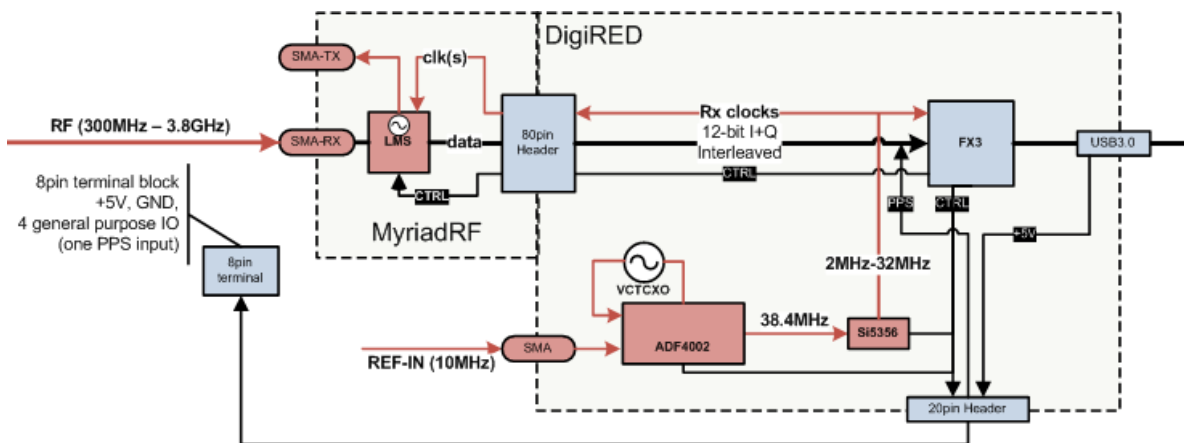


Figure 1 RASDR functional overview.

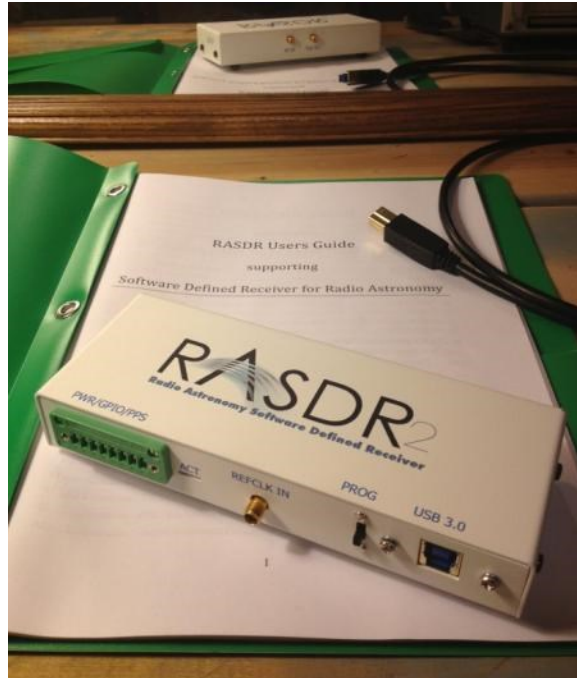


Figure 2 RASDR mounted in case

a) Specifications

Basic RASDR specifications are presented in Table 3:

Transceiver chipset	Lime Microsystems LMS6002D
RF Bandwidth (BW)	300 MHz to 3800 MHz
ADC Resolution	12 bits recorded in 16bit samples
Sampling Rate	32 MHz I & Q (equivalent to 64MHz Nyquist)
Band pass filter BW	1.5MHz to 28 MHz in 16 discrete steps
External Reference Clock Input	10 MHz sine (see p.79)
1 PPS Input	Yes through GPIO pin
RF/REF connectors	SMA female, 50ohm
PC connectors	USB 2.0 or 3.0
Input Voltage	5V Delivered through USB connector (beta units)
Dimensions	70mm x 70mm x 32mm (2.75" x 2.75" x 1.25") for bare PCBs
Maximum internal RF gain	61 dB

Table 3 Basic RASDR specifications are focused on receive operations from app. 0.3-3.8 GHz.

b) Component Board Considerations

RASDR USB connectors on the DigiRED board are delicate; mechanical strain when connecting and removing must be minimized. It is recommended to install the RASDR boards in a suitable case and use a panel mount USB. Switching power supplies can contribute electrical noise and

should be only used with caution. If RASDR is installed in a box, heat sinking or cooling may be advised. The thermal transfer characteristics of any enclosure must be such to radiate the heat into the environment outside the box.

Lime Microsystems, Ltd. has produced two versions of the MyriadRF board, and one of these requires a few additional components to match the DigiRED board. Components obtained from SARA will have compatibility between SARA MyriadRF and SARA DigiRED boards.

c) **Power and Data Transfer Considerations**

Pre-production RASDR2 units operate from a single USB2 or USB3 power source (computer or powered hub). The second, USB “charging” port on the bare DigiRED board should not be used, since this may introduce reverse current flow in other USB-connected components. RASDR2 devices which are packaged in the case have only one USB port exposed. Data are routed through the ‘receive’ USB3 connector, located close to the end of the board with the two SMA connectors.

Please use RASDRviewer for control. If RASDRviewer is not used then there is a chance that transmit functions may not be suppressed and it will be most important that the transmit SMA connector is fitted with a 50 ohm terminator to minimize RF noise.

Data transfer rates are faster when a USB3 superspeed connection is used, but require more CPU and system capability. A USB2 connection will operate reliably on almost all systems, but with lower bandwidth, up to 10 Million samples/sec (Msps).

System Base Specification	USB2.0	USB3.0
Dual Core, 2GB RAM	2Msps	Not Recommended
Dual Core, 4GB RAM	4Msps	25Msps, minimized ¹
Quad Core, 8GB RAM	10Msps	32Msps

Table 4 Typical Limits on Sample Rate per System Specifications

Some USB3 adapter chipsets function better than others, so if you are adding a USB3 adapter to your system, be prepared to experiment or request suggestions from the User’s Group. See Table 4 above on typical system specifications needed. A minimum system specification for a RASDR should be a dual core machine with 2GB of RAM in order to use minimum bandwidth over USB2.0. For USB3.0, a quad-core machine and 8GB of RAM is necessary.

¹ CPU utilization under RASDRviewer depends on several factors. USB3 data collection uses less CPU, but the higher data rate requires more CPU processing. Rendering the plots can also take time and minimizing the window is shown to reduce the CPU load.

4) Startup Guide for a RASDR Observatory

Each radio observatory is different. Most amateur radio astronomers² begin simply, monitoring the sun or Jupiter with a long wire antenna, or a dual-dipole antenna. RASDR may be used with block RF converters, depending on the spectral region of interest, and because of its versatility, has utility in many observing projects. Some options are covered in this guide, while others may be found in the SARA archives (Journal and Proceedings).

A typical RASDR observatory for operation in the 300-3800 MHz spectral region, including hydrogen electron spin-flip (H1) signals is shown in Figure 3.

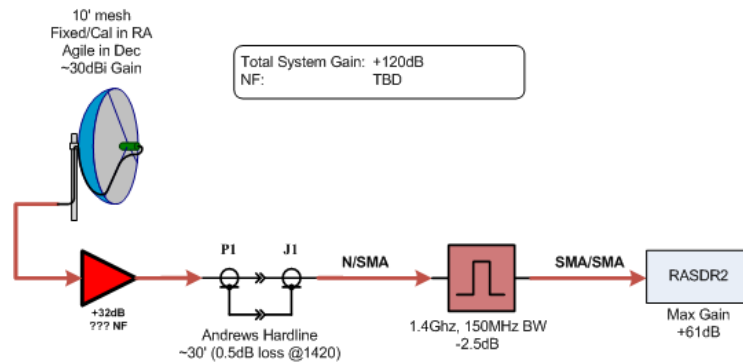


Figure 3 Typical RASDR observatory suitable for H1 observations

System startup has two goals – protection of equipment and validation of performance. These important considerations are developed in Table 5:

- Mentoring is important. SARA members are often willing to mentor new astronomers – contact SARA [14] if you have questions. Prepare a sheltered, protected area and install system components.
- Provide safe, grounded electrical power.
- Test and approve power supplies for proper DC levels and minimal AC or line noise.
- Connect and validate operation of RASDR/computer with no antenna connection.
- Verify that there is no DC on the RASDR input (especially if a bias-T is used to provide power to the preamplifier).

² Astronomy? Impossible to understand and madness to investigate. Sophocles, c. 420 BCE

- Connect a 50ohm terminator to the RASDR RX input and tune the system to the frequency and sample rate desired. Calibrate the DC offset (see section X); you may need to do this at the beginning of each observation.
- Connect antenna/preamp output to RASDR and perform function tests with an unpowered preamp. RASDR should detect interference signals from local 400-425MHz, TV, WiFi, etc.
- Provide power to preamp and observe increase in signal, measured by RASDR.
- Test with noise source, or with antenna pointed at dark sky, vs. at building/trees/hand etc.
- Observe available sources (sun, hydrogen clouds in our galaxy, artificial sources, etc.).

Table 5 Example startup procedure for new RASDR observatory

The above reference to the antenna suggests its importance. Antennas have different forms and must fit the frequency being monitored, the experimenter's budget, and the available space. The intricacies of antenna selection³ lie beyond the scope of this text, but the SARA literature (Journals and Proceedings) can be very helpful.

Antennas can be built at low cost. Especially for frequencies below 1 GHz, the Radio Amateur's Handbook [19] can be useful. For Jovian monitoring near 20 MHz, a dual-dipole antenna is attractive [20], but requires a suitable RF block upconverter for use with RASDR. See chapter 9)d), below for a project in this area.

If interference from RF signals causes spurs and noise floor elevation, an RF filter may be used either ahead of the LNA, or between the LNA and RASDR. Usually the latter is a better starting point.

a) **Installation of Signing Certificate**

To begin, insert the CD that came with your RASDR device or download the RASDR software distribution from the rasdr.org website. Unpack the files and take note of the location, you will need it when you connect the RASDR receiver to your computer. The remainder of this section assumes that you are using the CD and that it has been mapped to drive letter 'D:'.

³ "I remember attaching a wire clothing hanger to the antenna of my radio in my bedroom, so I could get the frequency and get that station and listen to the top 10 every night." ~Nelly Furtado

The RASDR drivers are signed with a certificate that must be provided to the operating system prior to connecting the RASDR receiver to your computer. This certificate is only provided on the RASDR CD. On it you will find a 'RASDR-authority.cer' file. Before you plug-in your RASDR board, please access this file using the Windows explorer. Follow the instructions below to install the certificate to the 'Trusted Root Certification Authorities' store.

1. Double-click the 'RASDR-authority.cer' file, and press 'Install Certificate'.
 - a. If you are asked to choose 'Current User' or 'Local Machine', choose 'Local Machine' (Figure 4). You may be asked to provide an administrator password, please do so.
2. After you do that, it will ask where to put the certificate, please choose 'Trusted Root Certification Authorities' by browsing (Figure 5).
3. You will be presented with a dialog similar to Figure 6.
4. When this process completes, you should be able to re-open the 'RASDR-authority.cer' file and it should show that the certificate is trusted as shown in Figure 7.

The above procedure was tested with Windows 8.1. It should be similar or identical on Windows 7 and Windows XP. The only consideration is to ensure that the procedure is run as an Administrator. If you are using an account that does not have administration privileges (Least User Privilege), then you must switch user to a full administrator account in order to load the 'RASDR-authority.cer' certificate file using the above steps. Once that is done, you can log out and resume loading the driver under your user account.

If you have done the above steps correctly, you should be able to double-click the 'RASDR-authority.cer' file under your user account and see a screen similar to Figure 7.

After this is done, please connect your RASDR to the USB, and point the new device manager to the location on the CD containing the driver files: D:\driver\



Figure 4 - Certificate Installation Step #1

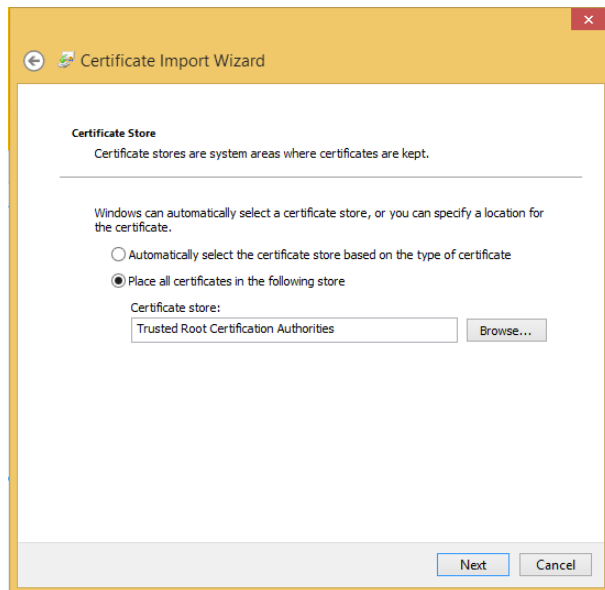


Figure 5 - Certificate Installation Step #2



Figure 6 - Certificate Installation Step #3

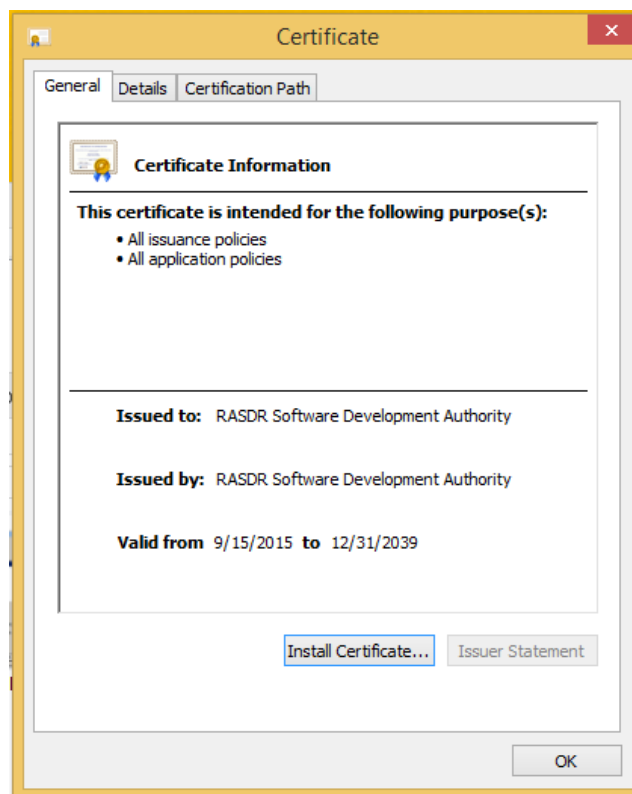


Figure 7 - RASDR-authority.cer details after successful Certificate installation

b) **Installation of Device Driver**

Ensure that your RASDR receiver is set to 'RUN' or that it has the jumper fitted onto the J2 connector of the DigiRED board. Connect the USB cable between the RASDR receiver and the desired USB port on your computer. You may use either a USB2.0 or USB3.0 port, although recommended operation of RASDR for a USB3.0 connection requires a modern multi-core computer and greater than 4GB of system memory.

You will then go through a series of steps to associate the device driver with the RASDR receiver, as shown in the series of screenshots, Figure 10 through Figure 12. The following procedure should be followed:

1. Connect the RASDR receiver USB port to the desired USB port on your computer.
 - a. For USB3.0, selecting the right port may be a challenge as some laptops provide only limited power output. In this case, you will see a message such as Figure 13.
2. Ideally, you will see a message such as Figure 8. This means that the Certificate installation above was successful and the driver package can be installed. Please click 'Install'.
3. Following a successful installation, you can open the device manager, and select the 'DigiRed_RX' device. You should see details such as described in Figure 9. If you see this, then your RASDR is installed and you can stop. Otherwise, please continue.
4. You may see a message such as Figure 10. That is OK. You will update the device driver in the next step. Please note that under Windows 8 or 8.1 the only way to install the device driver is to successfully install the Certificate, so the rest of these instructions will only apply to Windows 7, Vista or XP.
5. Start the Windows Device Manager. You may need to provide an administrator password. Select the 'DigiRed_RX' device, and choose 'Update Driver Software...' as shown in Figure 11.
6. Choose 'Browse my computer...' and navigate to the folder on the CD under 'D:\driver\' based on your operating system:
 - a. 'win10' if you are using Windows 10
 - b. 'win8' or 'win8.1' if you are using Windows 8 or 8.1, respectively
 - c. 'vista' if you are using Windows Vista
 - d. 'wxp' if you are running on Windows XP
7. Ensure that 'Include subfolders' is checked, and press 'Next'
8. Observe that the 'RASDR Radio Astronomy SDR RX Interface' is successfully installed, as shown in Figure 12.

At this point, you may use the CyControl software on your CD to check the details of the RASDR device. See Figure 14.

At this point, your system is ready to run 'RASDRviewer_W.exe' or 'RASDRproc_d.exe'



Figure 8 - First Time Installation with properly installed Certificate

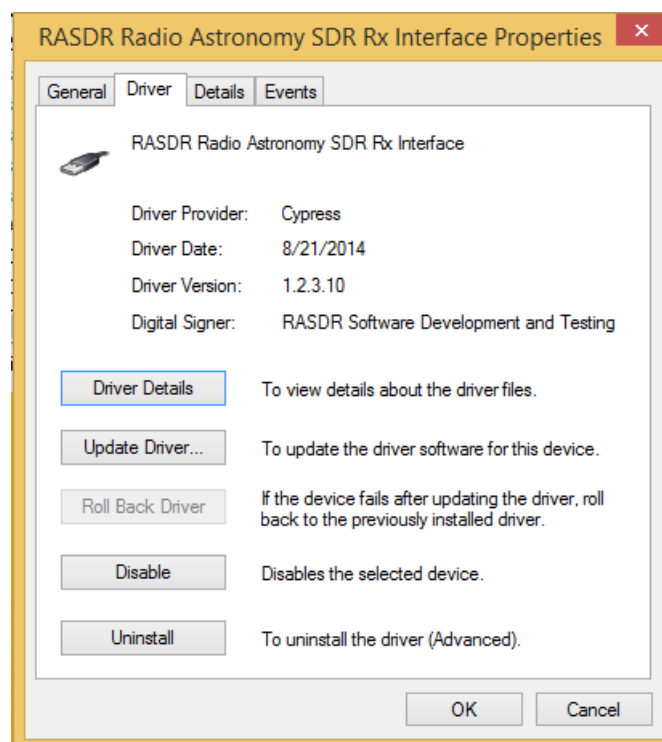


Figure 9 - RASDR Driver Details

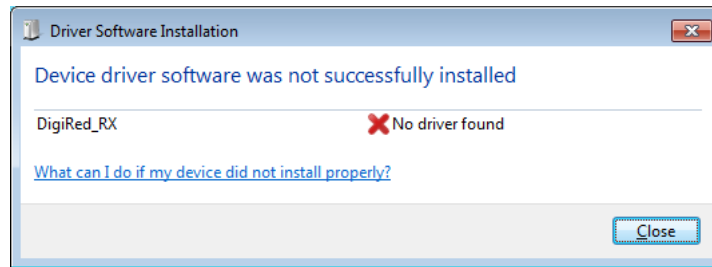


Figure 10 – The user may receive an error message if RASDR inserted prior to driver update.

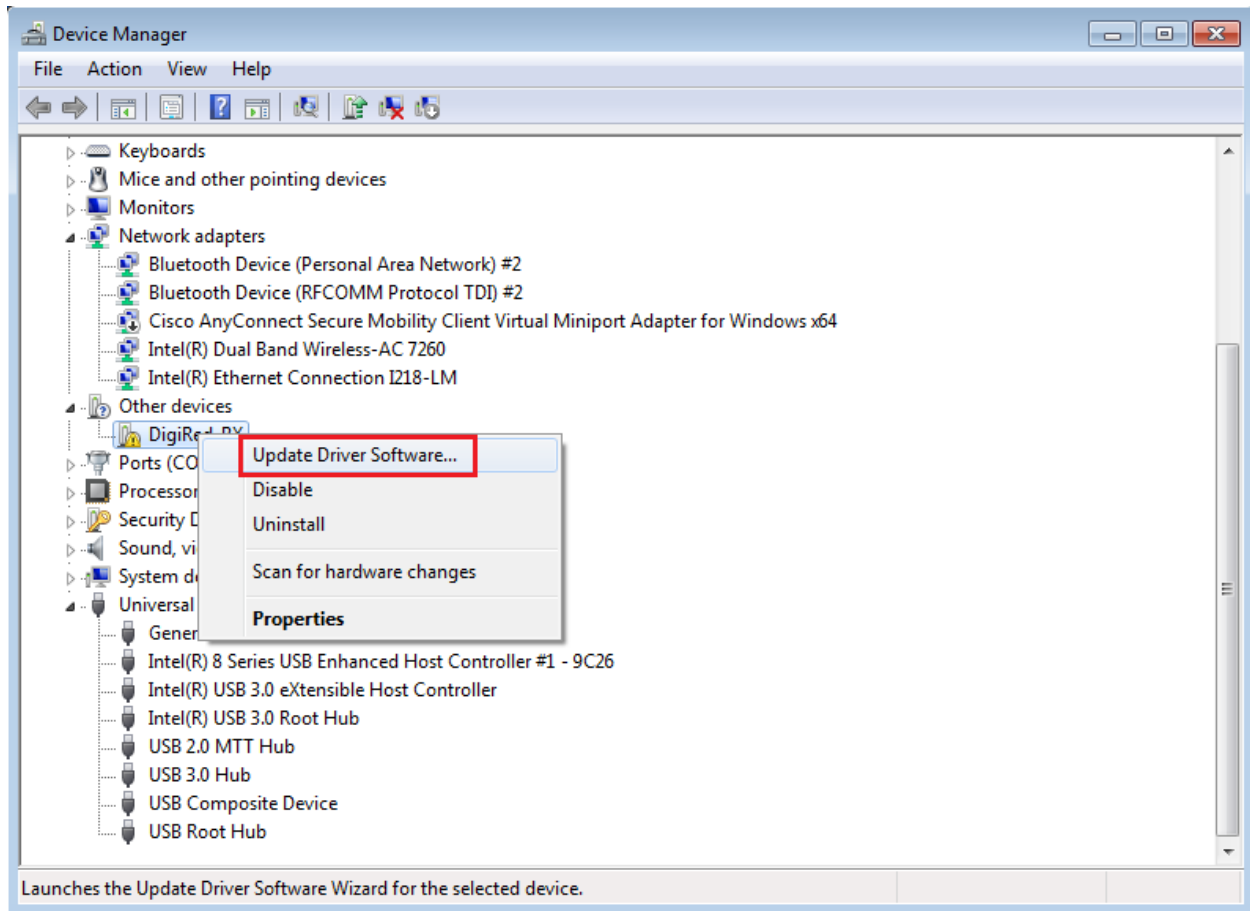


Figure 11 - Choose Device Manager and Update Driver Software

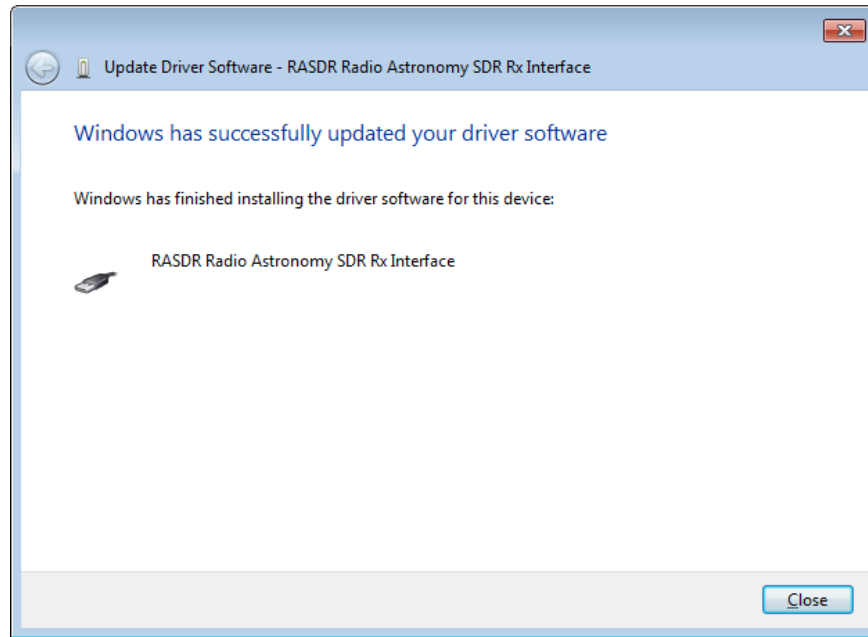


Figure 12 - RASDR Driver successful installed

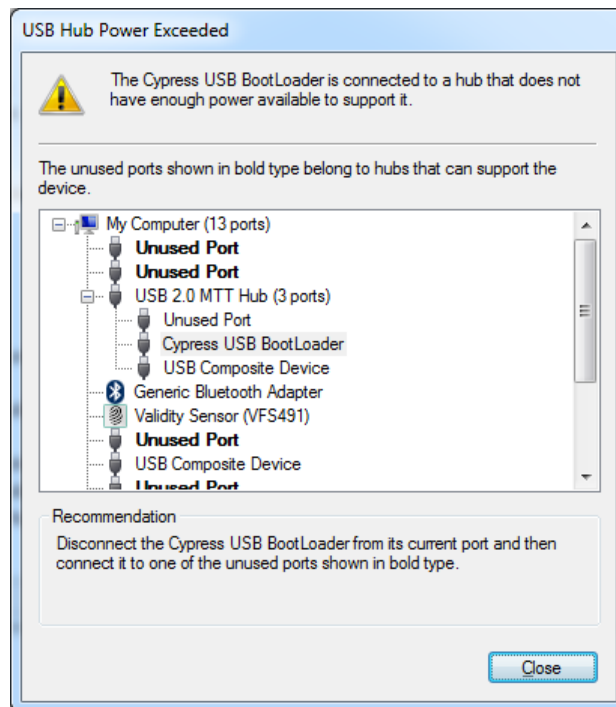


Figure 13 - Message Displayed when RASDR is connected to a port with insufficient power delivery

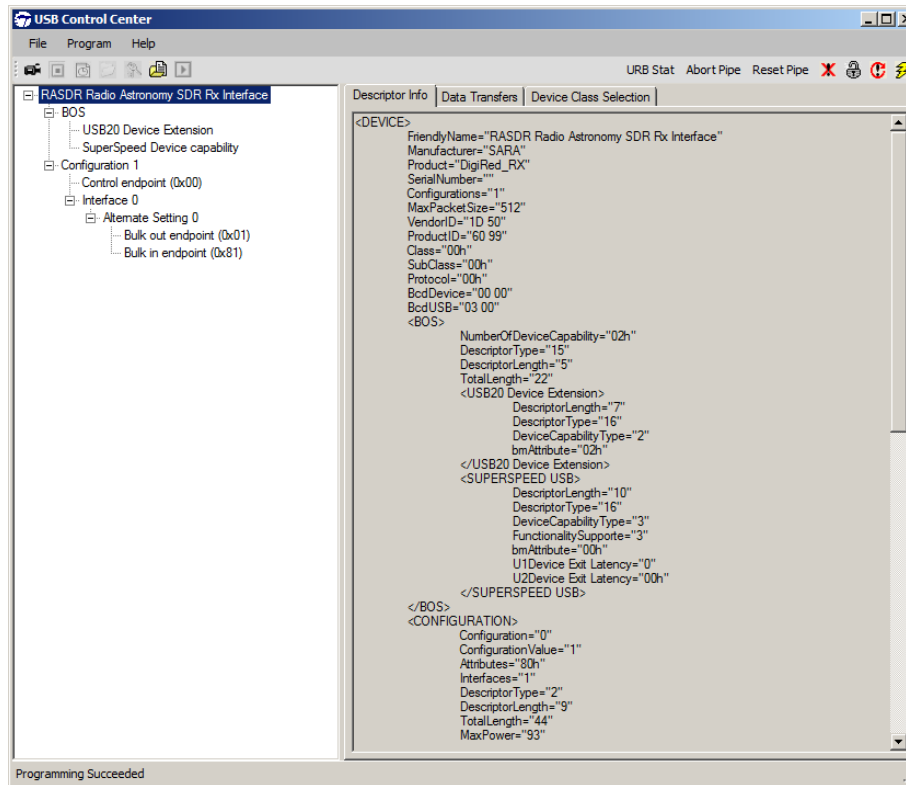


Figure 14 - CyControl Software Details after successful installation

5) RASDR Hardware Basics

As can be seen from the figure below, the core hardware RASDR package consists of two main boards, the Myriad RF and DigiRed boards.

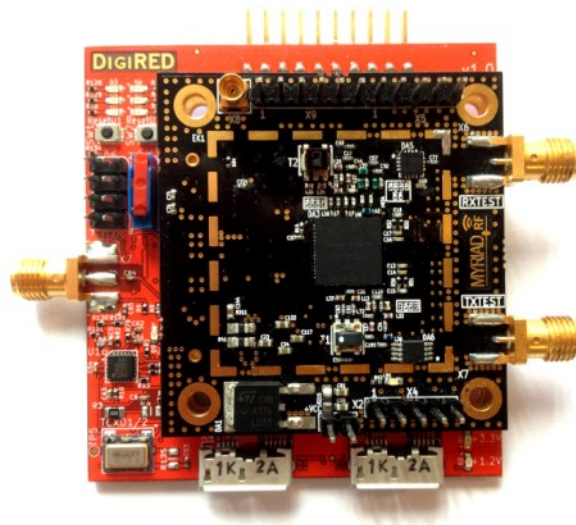


Figure 15 The core hardware RASDR package consists of two circuit boards.

a) MyriadRF hardware

Layout

RASDR uses the MyriadRF board [21] for RF digitations and front-end processing.

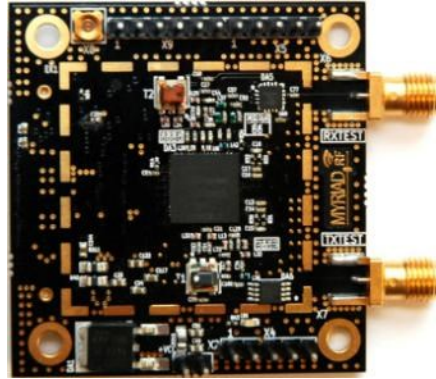


Figure 16. The Myriad-RF board was manufactured in several versions. SARA boards produced in 2014 are matched to the DigiRED board. Earlier versions may require addition of a simple dongle with extra resistors.

Shown above in Figure 16 the Myriad–RF 1 board is a multi-band, multi-standard RF module, based on the state-of-the-art LMS6002D transceiver IC by Lime Microsystems. It has one RF broadband output, one RF broadband input with digital baseband interface, established via a high-density connector FX10A-80P. The board also provides the user with pin headers for power supply, reference clock, analog I/Q input/output and SPI interface connections. It contains everything needed for it to be connected to baseband (BB) chipsets, FPGAs or to connect to a host interface chip.

The Transceiver specifications are as shown in Table 6:

Item	Specification
Transceiver	Lime Microsystems, Ltd. LMS6002D
RF Bandwidth (BW)	300 MHz to 3800 MHz
Baseband BW (per I/Q channel)	Programmable (16 selections); 0.75 – 14 MHz, Bypass mode
RF Module Control	Via SPI interface via DigiRed
Reference Clock frequency	23 – 41 MHz (10 MHz only for external Reference unless specific frequencies are programmed.)
Input Voltage	5 V (recommended). The Myriad–RF1 board is powered through the DigiRED board via the USB cable.

Dimensions	5.5x5.5 cm
------------	------------

Table 6 MyriadRF specifications.

Input/output

Figure 17 and Table 7 below provide key Input/Output configurations, connector assignments and descriptions. The analog differential IQ interface is available on Myriad-RF board and provided via X3 and X4 connectors. X6 and X7 are the RF connectors for receive input and transmitter output, clearly marked in the picture. The front end switches are configurable for selecting Receiver inputs and Transmitter outputs via GPIO's which in turn can be controlled via the X3 connector.

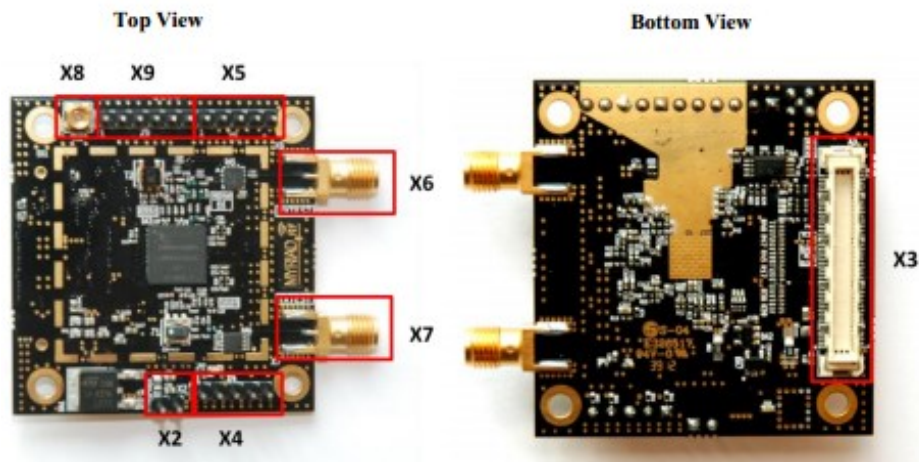


Figure 17 The MyriadRF board is the RF section of RASDR, and interfaces directly to DigiRED via the X3 connector. The SMA connectors are for transmit and receive sections. Only receive operations are currently supported by the RASDR team.

Connector	Name	Description
X2	+5 V supply	External +5 V supply. Supplied via DigiRed
X3	Digital I/O	The FX10A-80P is a connector used to interface the RF board directly to interface board or any other baseband board. Used by DigiRed. For details see Appendix E.
X4	TX Analog I/Q	Connector used to supply Transmit analog I/Q signals. Not usable on RASDR2 without hardware change.
X5	RX Analog I/Q	Connector used to measure Receive analog I/Q signals. Not usable on RASDR2 without hardware change.
X6	RXTEST	SMA connector provides connection to low band or high band RX input.
X7	TXTEST	SMA connector that provides connection to low band or high band TX output. Not supported by the RASDR2.
X8	Ext – CLK	Connector used to supply PLL clock externally. Use external clock on DigiRed connector.

X9	Ext – SPI	Connector used to control LMS6002DFN SPI registers externally. SPI register are controlled via X3 connector through DigiRed board.
----	-----------	--

Table 7. Myriad-RF Board Connector Assignments

Configuration [Advanced users]

The MyriadRF board can be configured to a high level of detail through the RASDRViewer software configuration utility via the menu. This contains the Lime Microsystems Ltd. configuration utility that comes with the Myriad RF board [22]. The main GUI takes care of all the settings needed to operate the receiver for most use situations.

b) DigiRED hardware

Input/output

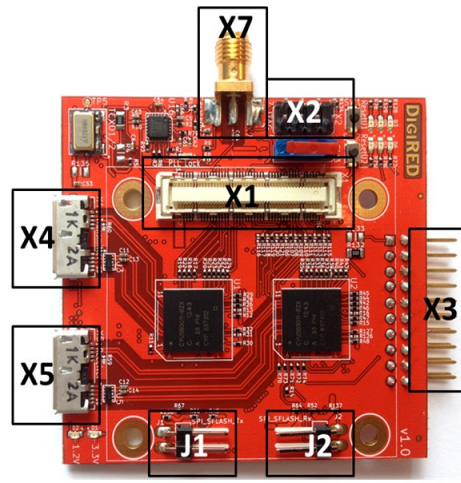


Figure 18 DigiRED connection descriptions.

The following table describes the DigiRED board connectors shown in the figure above.

Connector	Name	Description
X1	Myriad RF board connector	The FX10A-80P is a connector used to interface the Myriad-RF board directly to a base band board. See appendix e for more details.
X2	JTAG connector	Five-pin JTAG interface to connect to a JTAG debugger in FX3 USB microcontroller. See appendix e for more details.
X3	GPIO connector	Connections to FX3 USB microcontroller GPIOs, for more details see the next sections. See appendix e for more details.
X4	TX USB	USB2/3 interface connection for TX path.
X5	RX USB	USB2/3 interface connection for RX path.
X7	SMA female	10 MHz Reference clock input for ADF4002 to lock the external clock from test equipment with DigiRED board clock.
J1	TX Flash	This connector enables TX uC to load the firmware at startup. If firmware is already loaded, these pins should be shorted (jumper in place).
J2	RX Flash	This connector enables RX uC to load the firmware at startup. If firmware is already loaded, these pins should be shorted (jumper in place).

Table 8 DigiRED board connectors and switches.

USB Control

Rx control and data streaming is performed through USB3.0 RX connector which is also backwards compatible with USB2.0.

A computer with USB2 connectivity can accept RASDR data using a USB2 cable; however, the bandwidth (BW) will be limited, up to 10 Msps. A USB2/USB3 adaptor can be added to permit USB3 input, but not all such adaptors are equivalent, since different chipsets are used by different vendors.

For desktop PCs, good results have been obtained with the Konig Electronic CMP-PCIE2USB3 PCI Express card with the NEC/Renesas UPD720202 chipset, available from Amazon for about \$36 (2015 price).



Figure 19 Some Konig© (L) and Inateck© PCIE cards provide USB2/USB3 connectivity to PC desktop computers.

Good results were also obtained with an Inateck KT4007 which uses the NEC D720201 chipset. The Konig and Inateck boards are shown in Figure 19.

Please share your observations and experiences with the RASDR User group.

LED indicators

RASDR status may be monitored by LEDs on the DigiRED board.

DigiRED board is supplied from the computer via USB connection. Once the USB cable is connected, the green power LED's (labeled +3.3V and +1.2V) should go on, see Figure 20.

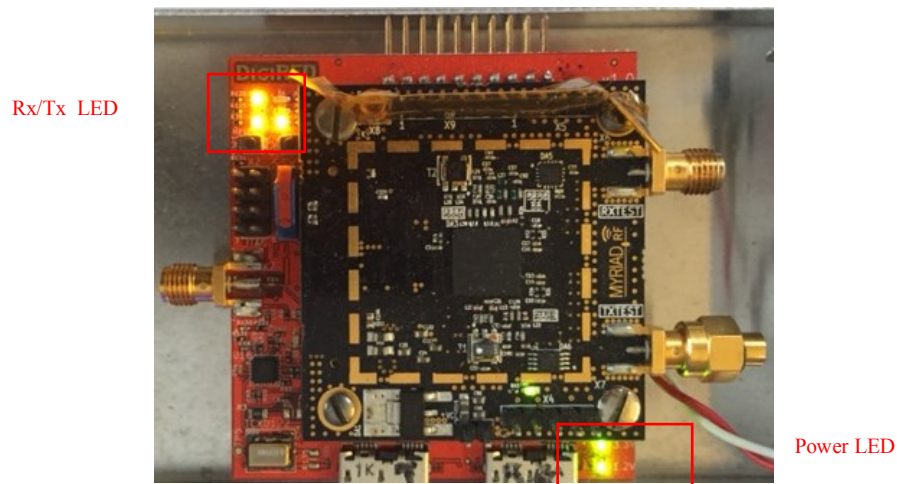


Figure 20 LED Indicators on digiRED board.

Six yellow LEDs show the Rx and Tx digital signal status. See Figure 20 and Figure 21 for board position and Table 9 for the interpretation of the status LEDs. External frequency input (verified by LED D9 is discussed on page 79.

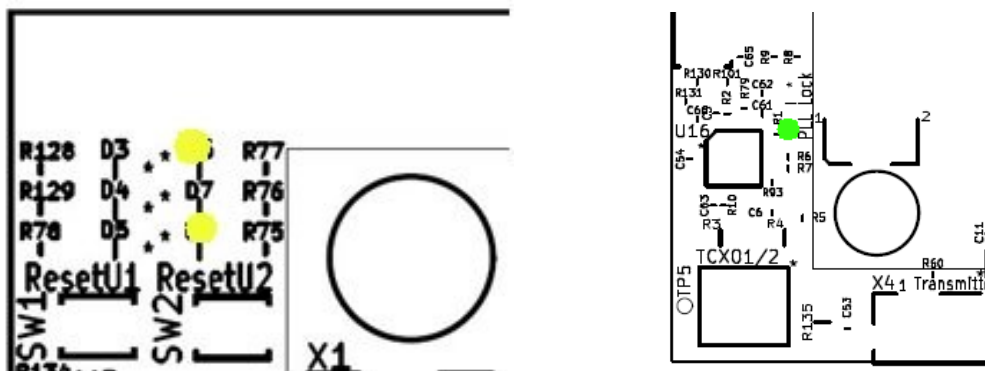


Figure 21 (L) shows Rx and Tx LED indicators and (R) shows location of LED D9.

LED ID	LED Color	Interpretation	Flash status
D1	Green	3.3v regulator power status	On when powered
D2	Green	1.2v regulator power status	On when powered
Transmitter data status (from Chip U1B, CYUSB3011)			
D3	Yellow/orange	Not used in production RASDR	
D4	Yellow/orange	Not Used in production RASDR	
D5	Yellow/orange	Not Used in production RASDR	
Receiver data status (from Chip U2B, CYUSB3011)			
D6	Yellow/orange	Rx command indicator	Flashes while DigiRED is receiving commands. On in bootloader mode
D7	Yellow/orange	Tx command indicator	Flashes while DigiRED is sending data. On when data are streamed.
D8	Yellow/orange	Firmware indicator	Flashes when firmware is ok. On in bootloader mode
Frequency lock status			
D9	Green	Frequency lock indicator	Green when frequency lock is achieved

Table 9 DigiRED LED status indicators.

Configuration

The DigiRED board is controlled by RASDRviewer application, see next chapter.

Loading new firmware

From factory, the RASDR receiver will be loaded with the latest firmware; however, updates are available via rasdr.org. Please refer to Chapter 11)b) for details on updating the firmware in the field.

6) RASDRViewer and RASDR Software Basics

a) Introduction

For many users, the ‘meat’⁴ of the Software Defined Receiver is the software. Evolving software provides the promise for developing algorithms that process the detected radio frequency signals to extract details on the underlying physics and astronomy processes that generated the signals. There are various approaches to data processing. The foci of RASDRviewer is to program MyriadRF, demonstrate via its GUI (Graphical User Interface) the existence and form of the signal, and route the digitized data to the background computer for processing, display, and program control. RASDRviewer is also a data capture utility that selects and saves data for subsequent processing.

RASDRviewer [23] [24] is the primary software that controls RASDR2 and presents captured data to the user. It uses a Windows based GUI that is designed for portability to both the Linux and MAC platforms. This portability is mainly based on the use of the wxWidgets development tool that is available as open source freeware. wxWidgets abstracts most of the common graphical window objects to a common language that is applicable across all of the platforms. Thus the look and feel of the user experience is the same regardless of the platform being used. Loading and using RASDRviewer on a Windows operating system is straightforward. The Linux and MAC options have not been developed as compiled code on these machines.

RASDRviewer is a ‘viewer’ and shows FFT output. It is in fact an extension of the Lime Microsystems FFTviewer that adds radio astronomy functionality. The original FFTviewer presented three charts: I & Q samples vs time, I vs Q for system verification and an output display showing results of a large Fast Fourier Transform (FFT) that operates in a best-effort manner. The FFT is capable of delivering up to 16,384 frequency bins multiple times per second.

For RASDRviewer, modifications have been made to customize the software for Radio Astronomy use. This includes optimization of control functions for Radio Astronomy use, addition of a power vs time plot, file outputs and inclusion of a simplified selection of the user options. Work is continuing to add additional features aimed at Radio Astronomy applications. This includes the Pulsar feature recently published in the SARA journal.

The system uses a device driver that is repackaged from the driver provided by Cypress Semiconductor, the manufacturer of the USB3 chip on the RASDR2 DigiRED board. The GUI provides the user interface in a format that is familiar to the user of the chosen platform. However, compilation chain can prove difficult to establish because of its complexity. Details of setting up the compilation chain can be found on the developer site [7].

⁴ I myself eschew all stimulants. I also practically abstain from meat. ~Nikola Tesla

RASDRviewer currently operates on most Windows platforms that are in current use. This includes Windows XP, Vista, 7 and 8. It is developed on Windows 7 and 8. Development continues to port the system to Linux and MAC platforms.

b) **Installing RASDRviewer**

RASDRviewer runs under Windows XP, Win 7 and 8. It is available for download from rasdr.org as described in one of the “new member” messages for the RASDR User’s Group or in the README file following [25]. It should be noted that the README text file will change with each subsequent version of RASDRviewer.

c) **Software Architecture**

The software is organized in classes, modules and files. Threading, call-back and event timers are used to process the data collected. High speed multiple core platforms will provide the maximum performance, which includes data collection at 32 Msp/s and the full 28 MHz bandwidth of the MyriadRF board. The rendering system for the plots makes use of OpenGL capabilities of your system. This means you will need a reasonably strong graphics processing unit (GPU) and optimized drivers. NVIDIA graphics cards made after 2009 are generally OK.

The system uses custom firmware that is loaded onto the FX3 chip in the DigiRED board. This includes a unique RASDR Vendor and Product ID (VID/PID) pair⁵. This allows the user to connect to RASDR2 without concern of not finding the correct USB3 driver. The software uses this unique VID/PID together with a version register on the Lime Microsystems, Ltd. chip to verify that system connections are correct. The user must use the driver provide on the CD or from rasdr.org that matches the VID/PID of the RASDR device.

The firmware provides not only the high speed data interface, but also a capability of establishing a Serial Peripheral Interface (SPI) bus to perform the control of the ADF4002 clock generator and the LMS6002D chips on RASDR2 board set. In addition, the firmware can provision the use of the General Purpose Input Output (GPIO) pins on RASDR2. A separate FPGA has been avoided on RASDR2. This has reduced the cost of RASDR2.

⁵ Graciously provided by OpenMoko

d) Graphical User Interface

When the user executes RASDRviewer, a screen that provides charts of the collected data is displayed, as seen in the figure below:

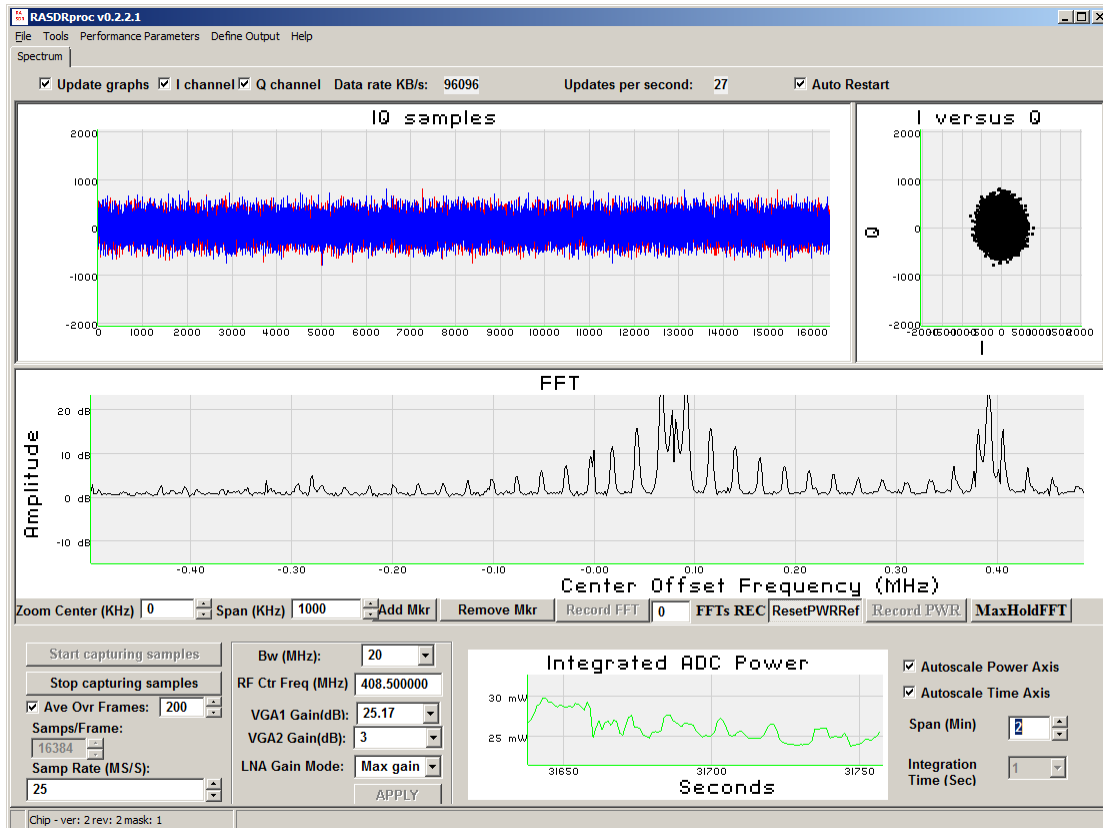


Figure 22 GUI Start window of the RASDRviewer application

Starting in the upper left side, the In Phase (I) and Quadrature (Q) samples are displayed against time.

To the right, there is a chart that displays the phase relationship between the I and Q samples. Since a significant spectral energy is present, the circle is present. The circle is generated by sinusoidal content in the sampled data. This chart is used mainly to verify the system. However, it is a useful educational tool for students that are learning the phase relationship between I & Q signals and flags the existence of any possible imbalance between I and Q channels.

In the middle of the screen there is a display of the FFT output. The vertical axis is in decibels (dB). The horizontal axis is the frequency of the FFT bin at baseband. Zero frequency corresponds with the center RF frequency. The frequency bin bandwidth can be changed by using different sample rates and samples per frame, which are set on the bottom left of the screen.

Figure 23 below, shows an enlarged portion of the screen where operating parameters can be set. The user can change the experiment parameters by using the controls in the bottom left section of the screen. This is where one would set the frequency, bandwidth, sample rate, frame size and gain. The changes are not put into effect until the user pushes the apply button. When the apply button is clicked, the system forwards the necessary commands via the SPI bus described above to the Lime chip on RASDR2.

The top left portion of the controls section on Figure 22 contains buttons that start and stop the capture of the data. When the start button is clicked, the charts will show the results of the samples. The start button will also activate the “Apply” button, which ensures that changes in parameters are included in the results. Just below the “Start” and “Stop” buttons is where the user can establish averaging of the FFT output. This is done by clicking the check box and setting the number of FFTs to average. Averaging is useful to enhance spectral lines by reducing the variance of the baseline noise. Since the baseline noise is random, its average value is reduced to the mean of the noise. Averaging may help visualize a Spectral line that is close to the noise baseline.

The FFT chart has the ability to set markers to help identify the specific frequency and peak value of a spectral line. The marker controls are below the FFT chart enlarged in Figure 22. To add a marker, click the “Add Marker” button. Then move the cursor to the screen and click where the marker should be placed. The marker can be moved left or right by dragging it with the mouse. With a little bit of practice, the user can find the peak value. The user can also set the span and center frequency of the display.

Figure 23 Lime Microsystems chip control settings

In the bottom right corner is a display of power at the ADC input. Figure 24 ADC Power input, below, shows this portion enlarged. The power is calculated over a frame of complex samples creating a display of the RMS power in the frame. The power chart is useful during drift scan experiments. The power will peak when the desired object is centered in the antenna beam width. The display may be renormalized by clicking on Set Power Reference (SetPwrRef).

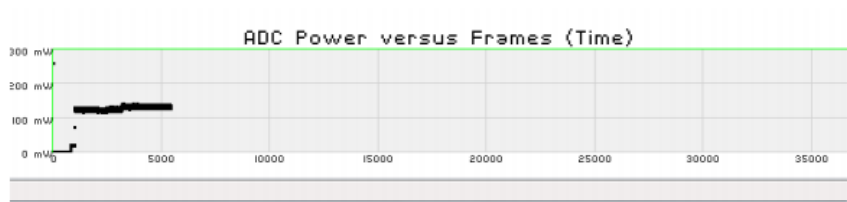


Figure 24 ADC Power input

All of the charts allow the user to change the range of the chart by dragging a box around the section of the chart to enlarge. Clicking on the left side of the chart restores the default ranges.

The charts are updated multiple times per second. The time between updates varies with different setup parameters. The frequency (updates / Second) is shown on the top of the screen. In RASDRproc, the ADC Power Input plot has controls to autoscale the vertical (power axis) or the horizontal (time axis) independently, as well as to set the visible span of the time axis (in minutes).

e) FFT Output to Disk

Recording data to a disk file or files can be accomplished by using the Define Output menu item on the second line of the screen. Clicking the button will bring up a sub-menu of the types of outputs available (only FFT output at present). Clicking the FFT output menu item will bring up an overlay panel for the setup of recording parameters. Figure 25 is a screen shot of this window. On the left of the top line the user can select the File type. Two options are available. The first is for use in MS-Excel where the number of frequency bins is reduced to a maximum of 128 to allow Excel to be able to process a 3D chart in a reasonable period of time. The user can select the General Purpose output for use in other post processing programs. The output in this case uses whichever sample rate, samples per FFT and bandwidth is present in the controls on the bottom left of the main screen. The general purpose output produces an ASCII-formatted text file containing some details about the data capture, a separator string, a comma-separated list of frequency for a column followed by rows of records containing a timestamp, and power spectral density values per frequency column. The extension used is provided by the user, but using .csv can allow the data to be loaded in MS-Excel easily.

The top center of the setup window is used to determine the behavior in the event multiple record sessions are desired. There are two options: one to append the data to the previous file and one to create a new file with a suffix number in the file name. *NOTE: If you are planning to use post-processing, it is important to select the multiple file option, as the append option is not supported by the post processing tools.*

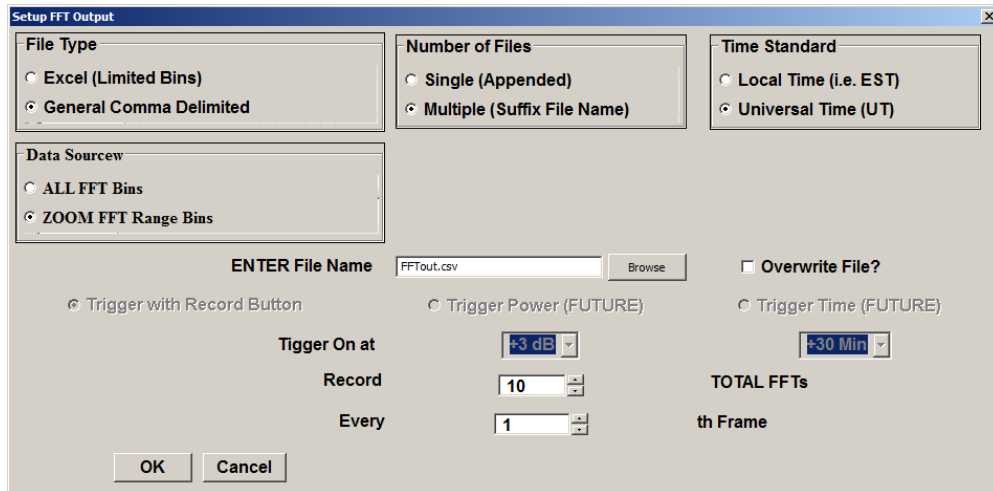


Figure 25 FFT Recording configuration

The top right section of the setup screen is used to define the type of time stamp to be used. There are two options Universal Time and Local Time. Local Time is taken from the settings used by your computer. The second section of the Setup Window is used to define the file name. A browse button is included to allow the user to find the directory and file name for the data. If the file already exists, it is necessary to either select a different file name, or check the “Overwrite” box. On the next line, in the future, the user will be able to define the condition that will trigger recording to the disk. At present, only the manual trigger is operational. Near the bottom of the screen, the user can establish how many FFTs to record and how frequently to record them.

Keep in mind that the ‘Nth Frame’ represents the FFT after all the averaging that was performed. So the interval between frames is a function of the Average * (Samples per Frame) / Sample Rate.

The bottom section contains the “OK” and “Cancel” buttons. To abandon the setup, click the cancel button. To establish the selected parameters, click “OK.”

Back on the main GUI, to begin a recording, the user first starts to collect data, then clicks the “Record FFT” button, which is located on the line below the FFT chart. See Figure 22. This button is only active when data is being captured by use of the “Start Capturing Samples” button on the left of the screen and an output is defined. The defined number of FFTs will be recorded on the disk. The “Stop FFT Rec” button can be used to stop recording data before the defined number of FFTs is recorded. Once the recording is stopped, the “Record FFT” button can be clicked for additional data as defined (either appended or new file suffix). In the ver. 1.2.2 of RASDR viewer, the next file is kept open until the user exits RASDRviewer. If the file is examined before closure, then the file will be shown to contain zero bytes.

Post processing of the files produced by RASDRviewer is described in Section 8).

f) **Balancing DC offset for I and Q**

With RASDRviewer, the DC offset correction is continuously applied. In RASDRproc, the DC offset correction is a calibration parameter that can be set, but does not change unless adjusted manually. See Figure 26 for an example showing I/Q out-of-balance.

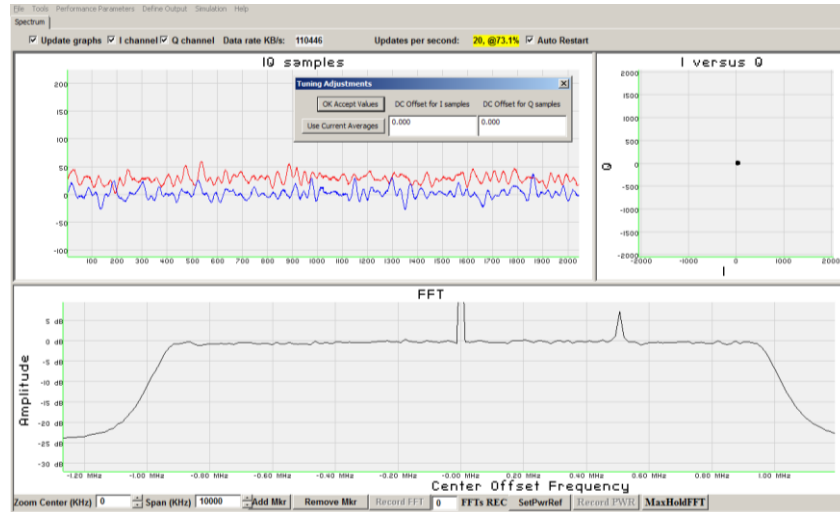


Figure 26 - I and Q out of balance

To correct it, use the pop-up from the menu tab ‘Performance Parameters->Tuning Parameters’ while collecting data to adjust the I and Q offsets.

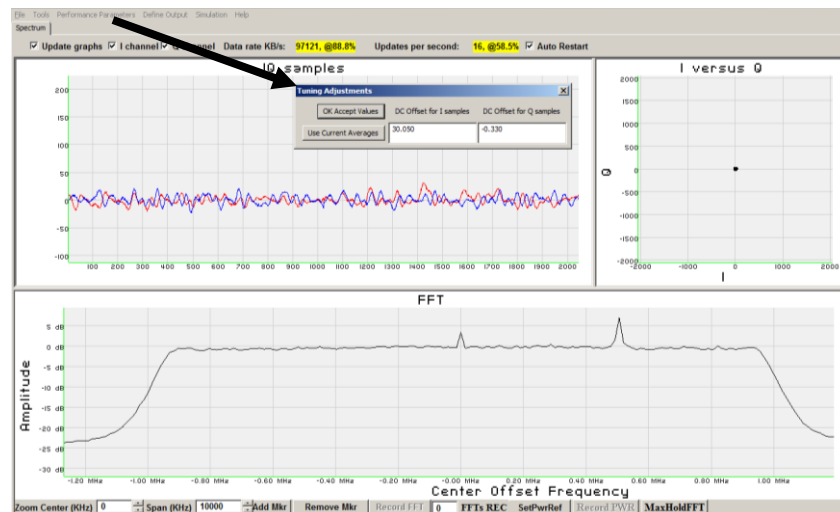


Figure 27 - I and Q adjusted

Press the ‘Use Current Averages’ to automatically populate the values with the average values collected according to the integration parameters. This will “approximate” the correct DC offset quickly. Press ‘OK Accept Values’ to close the pop-up and apply the offsets.

7) Basic Radio Astronomy with RASDR (discussion and examples)

RASDR has been applied in several modes [26] to solve some common experimental challenges encountered by members of the community of amateur radio astronomers.

A few examples will be presented in Chapters 7) and 9) but RASDR is a new tool and 2015 will be the first year of availability. Results of operation in several RF spectral bands will be shown and discussed. These results include monitoring HI emissions from distant clouds, interfacing to the NRAO 40' radio telescope to record spectral data; interfacing to the NRAO 20m radio telescope [27] to record and extract information from data taken at different observing sessions; monitoring a 10MHz section of the commercial FM band with a wire antenna; and monitoring a section of the crowded HF short wave band with a wire antenna.

a) Hydrogen HI Spectroscopy. L-band (1420 MHz)

Using RASDR2 for observing an HI hydrogen cloud signal requires a suitable antenna, a high-gain preamp (LNA), perhaps a supplementary amplifier and signal averaging to extract the HI signal from the RF noise.

A common question asked is how much sensitivity is needed? A popular special-purpose spectrometer that works well for observing HI emissions from distant hydrogen clouds is SpectraCyber II [28], which was invented and constructed by RASDR team member Carl Lyster.

As a point of reference, the SpectraCyber bin width is about 5 KHz. If we desire to cover 10 MHz of HI spectrum then we probably require about 2000 frequency bins or about 2048 samples/FFT. One may choose a higher number of samples/FFT observe more detail, although more averaging would be required. One of the significant advantages of RASDR2 is that it has a wide bandwidth.

RASDR2 control data consist of only a few values and they are discussed in Chapter 6). Based on the preceding discussion, an initial data screen for H1 spectroscopy is shown in Figure 28.

NOTE: The internal tuning oscillator inside the LMS6002D produces a small amount of spectral energy which leaks into the ADC front end. Thus, it is necessary to tune slightly off the spectral region of interest to avoid desensitizing.

Zoom Center (KHz)	0	Span (KHz)	2000	Add Mkr	Remove Mkr
Start capturing samples		Bw (MHz):		1.5	
Stop capturing samples		RF Ctr Freq (MHz)		1420.405000	
<input checked="" type="checkbox"/> Ave Ovr Frames:	128	VGA1 Gain(dB):		25.17	
Samps/Frame:	1024	VGA2 Gain(dB):		30	
Samp Rate (MS/S):	4	LNA Gain Mode:		Max gain	
		APPLY			

Figure 28 Initial choices of RASDR2 control parameters for comparison with typical SpectraCyber results, specified minimum BW for this narrow spectral region. See text for choice of values shown in red.

The first tests with these settings provided a noisy graphical output. The LNA gain used in this series of measurements was only 15dB and skies were cloudy. In general, one expects to average over many frames. Initial runs with post-processing averaging at 99 and 998 frames (each frame consisting of an average over 64 FFTs) produced the output shown in Figure 29. Typically, more gain should be used (see discussion of Figure 31). Note that the noise spurs (that appeared only after averaging) are about 9 dB above the floor and they are present in both tests.

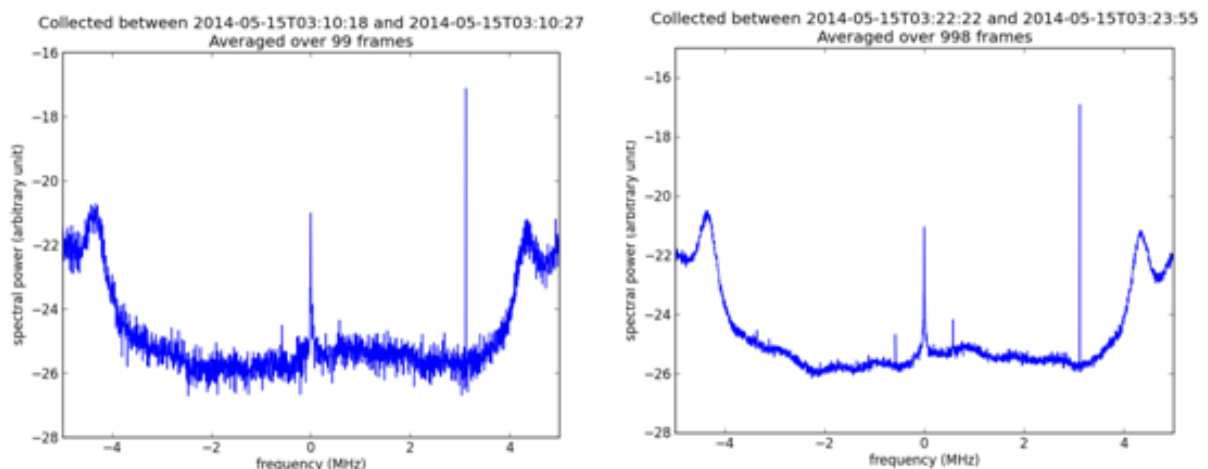


Figure 29 The importance of spectral averaging is shown with processing of 99 frames (left) and 998 frames (right). Control settings were as shown in Figure 28 for a dark section of sky. IBM laptop running WinXP, and using a USB2 interface was used for this particular test.

A second necessity is to remove the background noise and normalize for system response. Figure 30 shows the result of subtracting the two files shown in Figure 29 to remove the common baseline variation and the 9 dB spurs. This component is not present in the 998 FFT average.

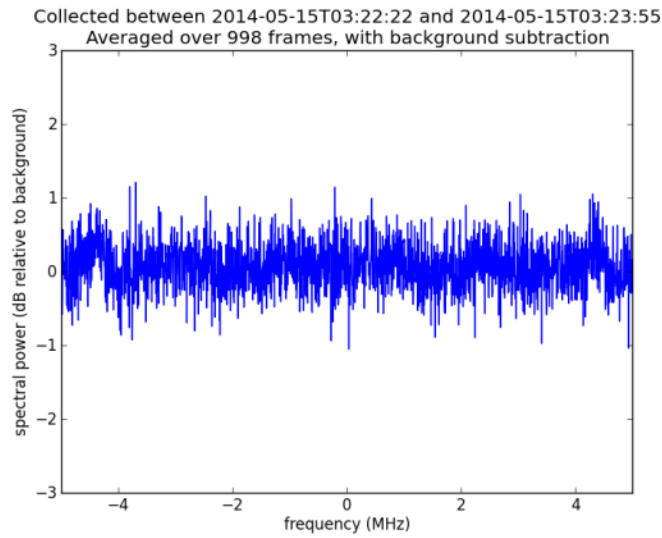


Figure 30 This figure is the difference file using the 99 file average and the 998 file average shown in Figure 29. The 2dB noise component from Figure 29 (left) is all that remains. The common feature is removed, and the same method can be used to remove system noise or select for 'genuine' signals.

b) System Gain Considerations

The question arises as to how much amplifier gain and how much post-processing is necessary to detect H1 signals. A user's system might be configured as shown in Figure 31.

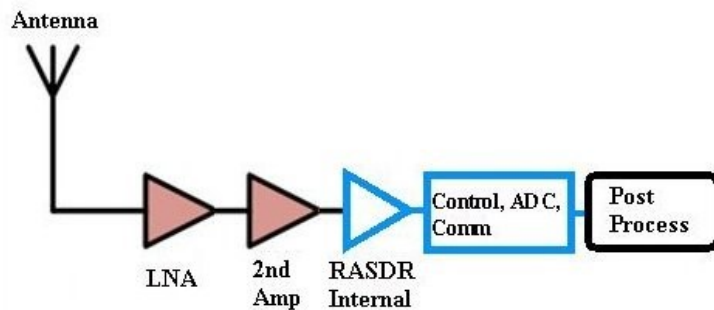


Figure 31 Example system configuration for H1 detection. RASDR hardware is outlined in blue.

Table 10 presents examples of rough calculation of system gain for possible monitoring of H1 signals:

Component	Low (dB)	Nominal gain (dB)	High (dB)
Antenna	20	30	40
LNA	18	25	30
Buffer Amp	20	30	40
RASDR (internal gain)	61	61	61
Total System Gain	109	136	171

Table 10. Examples of system gain from components for monitoring H1 signals. The buffer amplifier suggested here permits lower gain settings of RASDR, while still giving good H1 spectra (see next section).

The nominal gain for this example is 136dB. RASDR gain is adjustable, so for this example, the internal amplifiers have been set to a maximum internal gain of 61dB. To bring a modest H1 signal to a detectable level, the desired System Gain might be, depending on the object in the antenna beam, a range from 125-140dB. Some systems will not need a buffer amplifier or post-processing for weak signals while others may require it. The user may boost the amplifier gain or rely on numerical averaging, either in RASDRviewer or in post-processing with Excel or Python in order to observe the signal.

For post-processing to be useful, the input signal to RASDR must have remained above the internal noise floor during the measurement. Depending on the noise statistics, a rule of thumb is that the power level of the signal+noise from the external amplifier chain is above the RASDR noise floor by greater than 5%. This will ensure that the signal is being digitized and that subsequent processing can improve the weak signal.

c) Practical System tests for H1 detection

The observing systems discussed in Table 10. Examples of system gain from components for monitoring H1 signals. The buffer amplifier suggested here permits lower gain settings of RASDR, while still giving good H1 spectra (see next section). Thus, it is not absolutely necessary to use a buffer amplifier. It was attempted to demonstrate H1 spectrum capture without using a buffer amplifier [29].

Test System 1 Hydrogen H1 Spectroscopy with degraded amplifier

The initial tests included data collection at 19h right ascension (RA) and -7 deg declination (DE). Background data were collected in a nearby weak signal area (maintaining the elevation setting for convenience at 20h RA and -7 deg DE.

Data were saved for 5 minutes and (as expected from a system operated 8 dB below a ‘typical’ amateur system) were noisy. Both signals were averaged using the plotcsv [30] program and showed (Figure 32) a small artifact on the left and the data from RA29, DE-7 showed two or three hydrogen regions with different Doppler shifts.

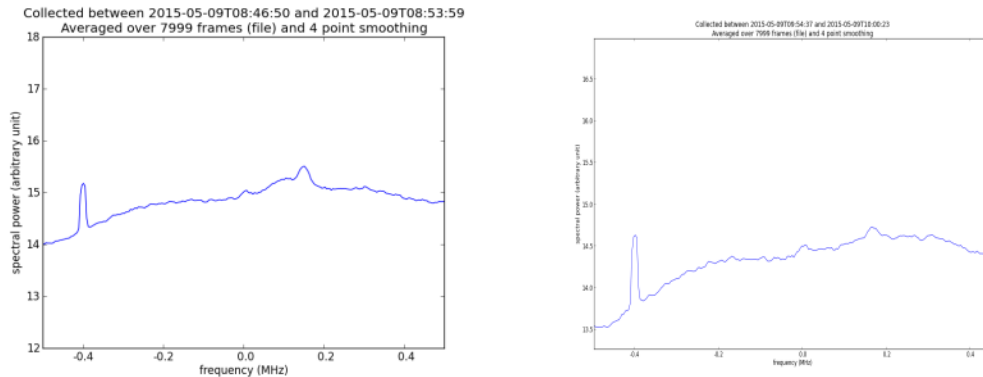


Figure 32(L) shows the “raw” averaged RASDR spectrum (center frequency 1420.407 MHz from RA19, DE-7 and (R) the “background” spectrum from RA20, DE-7, a rather thin part of the Milky Way. The “background” region was chosen using published maps and the small H1 features (see text) were not noted until after the spectral data from RA19, DE-7 on the left was processed.. The background plot has a slightly expanded scale to show the small background features. The spur appears at maximum RASDR gain settings, but disappears with background subtraction.

Reexamination of the background file from RA20, DE -7 showed the presence of small H1 signals, apparent in Figure 32(R) at the rest frequency and about +0.18 MHz from 1420.407 Mhz, plus a broad region at higher frequencies..

Figure 33 shows the post-processed H1 spectrum from RA19, DE-7, and this time with the center offset frequency specified for the plot axis. The small rest-frequency peak (no Doppler shift) has disappeared! Thus the background had been badly chosen. The AL rotator motor failed the next day, so a repeat measurement with darker background was not made. This shows the importance of choosing a dark sky reference region for background, and for carefully inspecting the background data before using it.



Figure 33. Subtraction of background from raw data and averaging shows the H1 spectrum from RA19, DE-7. This figure is misleading in that the 'dark sky' background region actually contained minor H1 regions, and contributions from these regions have been unfortunately subtracted in this plot. The center frequency is 1420.405 MHz.

Test System 2 Hydrogen H1 Spectroscopy using a small horn antenna

SETI research has been a spur to amateur radio astronomy [31]. Associated antennal designs include variations on the horn antenna. The antenna of test system 2 was a modification of the SETI Horn of Plenty [32] [33] [34] adapted to available local hardware store materials and other readily available parts. The calculated gain is 21.6dB [35].

The overall front end gain was **52 dB with NF 0.35 dB**, less than typical for amateur-scale observatories. Thus testing with test systems 1 and 2 demonstrate that radio astronomers with a small horn antenna or a 2m dish reflector antenna should be able to see hydrogen (H1) signals with a bit of averaging.

Two versions of the horn antenna are shown in Figure 34.



Figure 34 Test System 2 is based on a horn antenna, shown here in . Two versions were constructed. The version on the right had added aluminum foil, and lower noise.

A thin part of the Milky Way was available for testing and a 3-minute data scan using Test System 2 with subtraction of a 3-minute background provided the H1 spectrum shown in Figure 35.

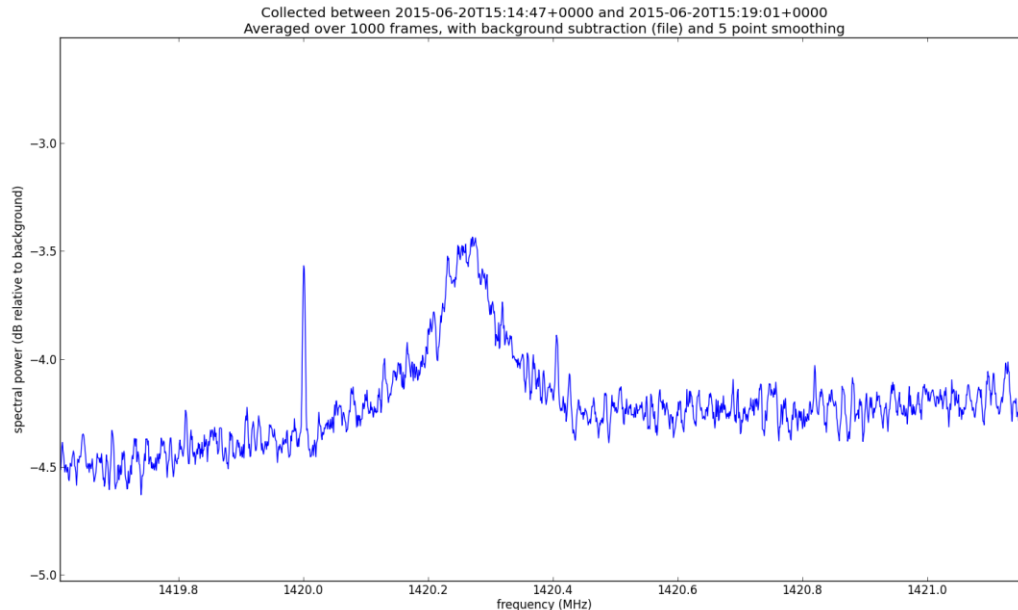


Figure 35. One quarter of a 12-minute Milky Way (Ra3h, De -8 deg) data scan, with background subtraction and smoothing.

8) Post processing to extract information from recorded data

RASDRviewer can output CSV (comma-separated-value) files in long-duration format (large files permitted), or in a shortened format that complies with Excel limitations. These data files are useful for analysis via Excel or other programs such as Python routines. The RASDR release provides a pre-compiled program called 'plotcsv' to post-process the output files that RASDRviewer makes.

```
Usage: plotcsv.py <filename.csv> [options]

Options:
--version                show program's version number and exit
-h, --help              show this help message and exit
-a N, --average=N       Specify the number of spectra to average for each
                        plot; default=1
-b PATH, --background=PATH
                        Specify how to perform background subtraction; if the
                        word automatic is used, then the background will be
                        taken from the average of all lines in the file.
                        Otherwise, it is taken as a file to process. The file
                        must have the same frequency plan as the foreground
                        file.
-c, --cancel-dc         Cancel out component at frequency bin for 0Hz
-d CHAR, --delimiter=CHAR
                        Specify the delimiter character to use; default=","
-e, --excel, --localtime
                        Indicate that .csv file has timestamps in
                        RASDRviewer's "LocalTime" format
-k CONST, --calibration=CONST
                        Specify the calibration constant for the system;
                        0.0=uncal, default=0.0
-l, --line              Perform line-by-line processing instead of loading
                        entire file(s); NOTE: much slower but tolerates low
                        memory better.
-i, --info              Produce information about a file only; do not generate
                        any plots
--statistics=PATH       Dump statistical information to a file in comma-
                        separated-values format, default=none
-v, --verbose           Verbose
-g, --gui              Create interactive PLOTS
-s N, --smooth=N        Smooth final plot using a sliding window of N points
--fcenter=FC            Define the offset for the center frequency in Hz;
                        default=0.0
--hold                 Perform a maximum value HOLD during averaging and plot
                        it as a second line
--bplot                If using background file, choose whether to plot the
                        background reference in a different color;
                        default=False
--ptype=TYPE            Control plot vertical scale (linear or log);
                        default=log
--atype=TYPE            Control averaging method (linear or log); default=log
--format=X.Y.Z          Specify the RASDRviewer .csv output format to
                        interpret; default=1.2.2
```

The tool is command line driven, and can be used with a batch file to set parameters. Please experiment with the tool, the example data files and settings that produce the type of processing and plotting that is desired. The program is distributed as both an .exe and as .py to allow inspection of the code and modification if necessary. A file "go.bat" can be edited and used.

We have setup an area in the User's Group [36] to share scripts and other items developed by the community at: <https://groups.yahoo.com/neo/groups/RASDR/files/PostprocessingSoftware/>

9) Basic Radio Astronomy below RASDR design frequency

This chapter describes the application of up converters to extend RASDR to frequencies from 20-400MHz; this is **below** the nominal limits of the bare RASDR unit. Software control was via RASDRviewer.

a) Radio Astronomy in the VHF band (80-190 MHz)

It is highly desirable to extend RASDR2 coverage to lower-frequency RF bands. Such an extension makes it far more useful to SARA members.

This section describes extension of operation to the Very High Frequency (VHF) band, specifically 80-190 MHz. This band is useful for solar monitoring and for meteor detection and it also contains an AM aircraft band, amateur radio activity, commercial FM band that are interesting for performance verification, and the output Intermediate Frequency (IF) band for the 40' Green Bank Radio Telescope [37]. To add functionality below the intrinsic RASDR2 band, an up converter was designed and constructed.

The basic LM6002d covers a wide frequency range from 300-3800 MHz. An amateur-built VHF up converter extends reception to an input frequency band between 80-190 MHz. Filters were included to reject incoming signals that upon mixing would have been up converted to below 300MHz, to avoid the third harmonic of the local oscillator (LO) that was used for mixing, and to remove the LO fundamental frequency component.

Our VHF up converter uses the third harmonic of a 100MHz oscillator [38] [39]. The 3rd harmonic (300MHz) is mixed with the input signal. The following basic components are also required, plus about 25 additional inexpensive components:

Crystal oscillator	Raltron CO19025-100.000MHz [39]
Low Pass Filter	LFCN-225 [40]
Mixer,	IAM81000 [41]
Output Bandpass filter combination	PHP-400 [42]
	LFCN-225 [40]

Figure 36 shows the functional diagram of the VHF up converter constructed for RASDR2 tests.

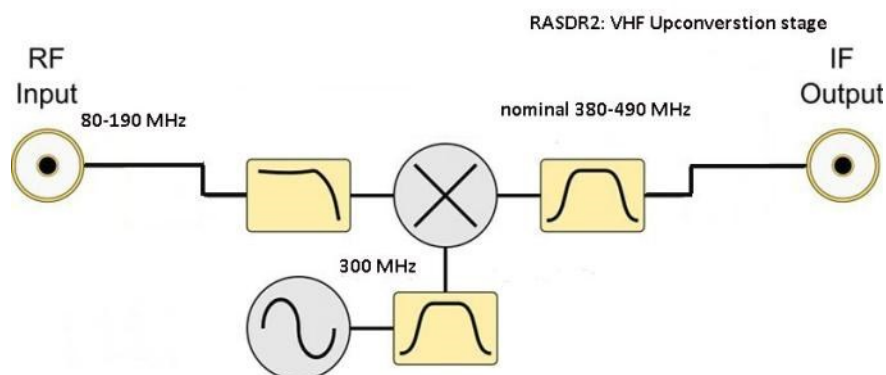


Figure 36. Block functions of the VHF converter show conversion of 80-190MHz VHF signals into the RASDR2 input band acceptance range.

Several filters are shown, the input low pass filter to avoid VHF components in the input from reaching the mixer; the 300 MHz bandpass filter to clean up the third harmonic of the oscillator, and the output bandpass filter (two filters are used in our unit) to select just the frequency components desired from the mixer stage.



Figure 37 Completed VHF converter used with RASDR2. Input is via the coax cable while the output is via the SMA connector shown on the upper right.

The circuit shown has a mixer with about 10dB gain resulting in no overall signal loss for the unit. The noise floor is about -120 dBm.

The unit is intended for several applications, including meteor detection, spectroscopy with the NRAO 40' antenna, and solar monitoring. A convenient test signal is the commercial FM band, using a wire antenna about one wavelength long. A length of hookup wire was connected to the input of the up converter.

Figure 38 shows a 10 MHz section of the commercial FM band taken directly from the RASDR2 GUI video output, using a 2m piece of hookup wire for an antenna, and the up converter of Figure 37.

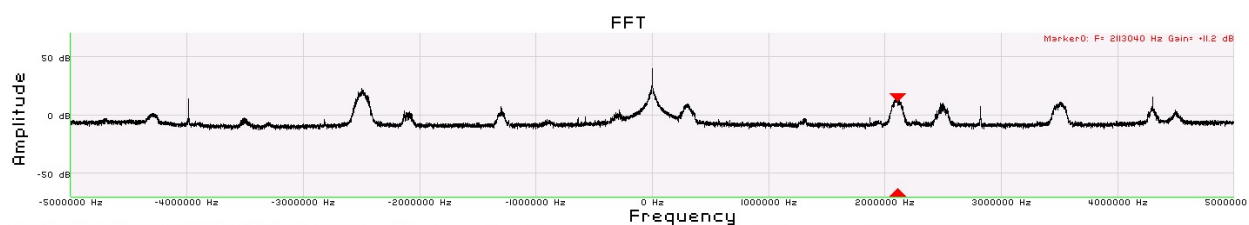


Figure 38 Commercial FM band. The antenna was a random wire about 2m long, connected to the VHF up converter, then to RASDR2. The red marker was added on the RASDR2 display to denote a local FM station broadcasting at 102.113 MHz

Selecting RASDR2 internal filters to narrow the band to 2.5 MHz width, demonstrates a practice that is useful to reject interfering signals outside the region of interest and avoiding any intermodulation. The result of setting the bandwidth to 2.5 MHz is shown in Figure 39.

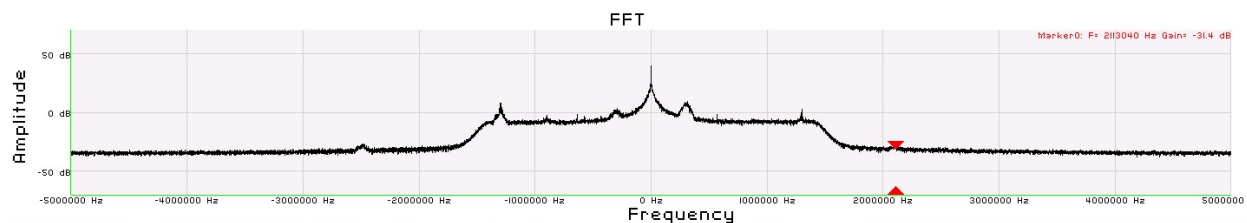


Figure 39 RASDR2 internal narrow-banding was chosen to select a region of interest. In practice, narrow banding is useful to reject unwanted outlying signals (noise).

RASDR2 output is more information-rich than suggested here. Figure 40 shows the complete primary-window display of a 10MHz-wide band centered at 90 MHz, with the RASDR2 marker function used to denote a couple of interesting local FM stations. Additional windows provide “Tools” and “Define Output”. As data were taken, the writer was listening to 91.9 MHz (red triangle marker) on the stereo.

b) Band-Extension Spectroscopy Testing and application to wide-band VHF monitoring



Figure 40 The complete RASDR2 video output with VHF input shows the input signal stream as digitized (I and Q values); the I vs. Q plot (a circle if there is a single coherent source), the FFT spectral display with markers, if selected, digitization and internal gain settings, and a power vs. time plot. The ratty power vs. time plot on the lower right shows the result of moving the antenna.

c) Using the VHF converter to access the NRAO 40' dish

Radio access to signals collected by the NRAO 40' telescope are restricted to using the local IF feed. This VHF up converter was designed with this challenge in mind. Study of HI using RASDR2 at NRAO, has been done by dialing in the 40' telescope local oscillator to 1315 MHz and measuring the signal at the output IF feed (1420 -1315 MHz), which would be at 105 MHz. This can be up converted to 405MHz to be examined with RASDR2.

d) RASDR Spectroscopy in HF band (2-80 MHz)

RASDR2 operation has been described as applied from 400MHz to 4GHz, and with the VHF up converter, to the input range from 80-190 MHz. To demonstrate functionality in the HF band from about 2-66 MHz, we used an up converter obtained from Nooelec [43] that up converts this HF input band to a 102-166 MHz output band. Thus the output is compatible with input of the VHF converter described earlier.

The Nooelec upconverter functionality is shown in Figure 41. The device shown has a measured internal loss of about -10 dB of signal. The noise floor is about -80 dBm. This device was used with a 100 MHz crystal oscillator, but one can also obtain it with a 125 MHz oscillator. If this component selection is made, then the output will be shifted by 125 MHz, and avoid the commercial and aircraft bands.

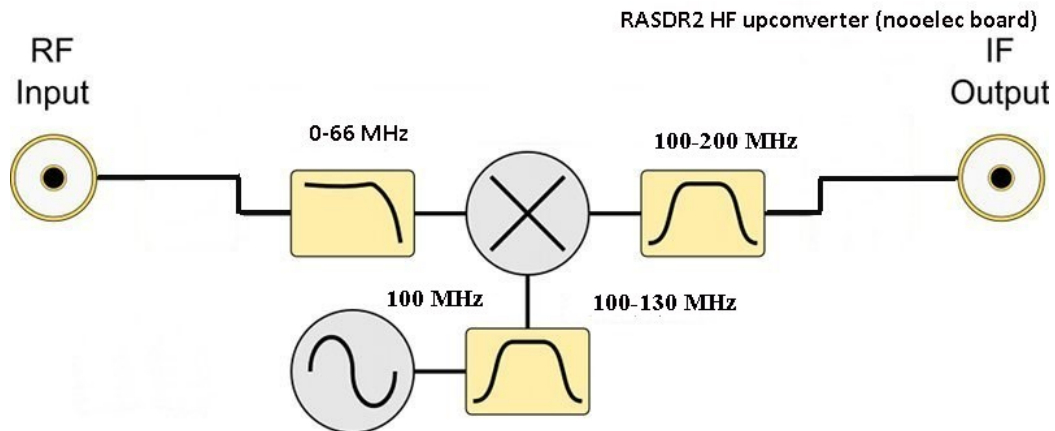


Figure 41 Functionality of the Nooelec up converter is as shown. The RF input is nominally specified 0-65 MHz and the measured performance covers the range 2-66 MHz. Filter functions are as described with reference to Figure 4.

For testing purposes, the HF up converter was mounted as shown in Figure 42, where it is the PC board on the lower left. The VHF converter discussed earlier is shown on the upper right.

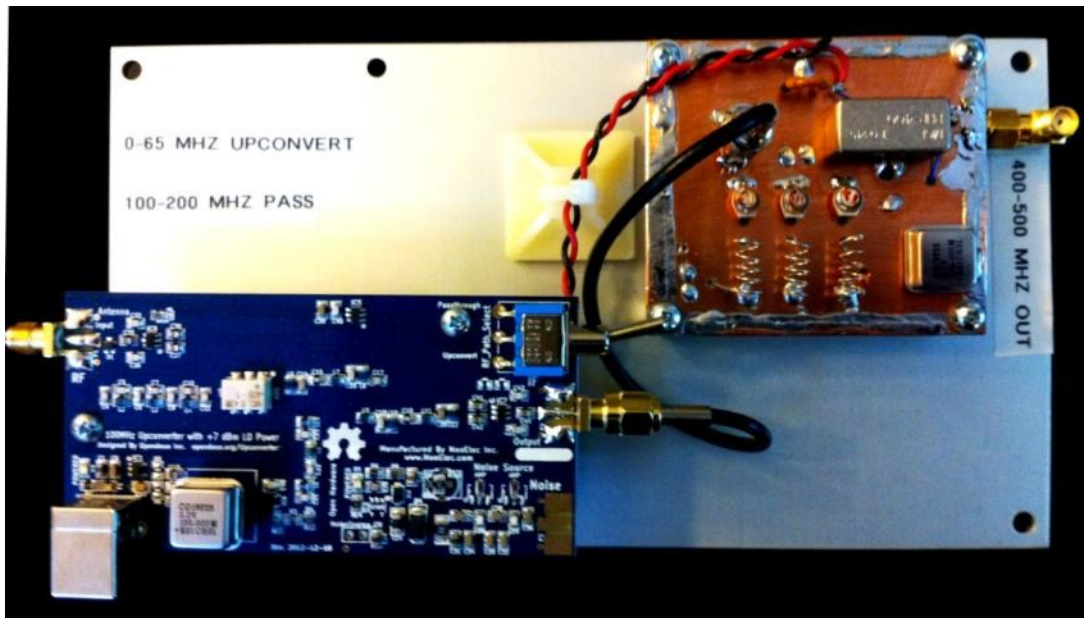


Figure 42 Two up conversion stages are shown here, which provide for conversion of the 2-66MHz portion of the HF band to the 402-466 MHz band prior to processing with RASDR2. The Nooelec converter is on the lower left while the VHF converter described earlier appears at the upper right.

Results were as might be expected using an antenna that is very short (2m) compared to the frequency. Figure 43 shows a 5.5 MHz section of the HF band centered on 15 MHz. The plot contains signals, spurs, and noise. The lesson from this figure is that one must use a respectable antenna and preamplifier, possibly a preselection filter, and background subtraction for a noisy band.

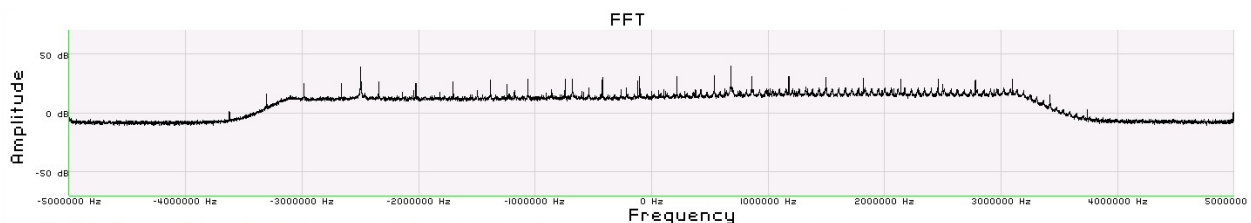


Figure 43 Noisy 5.5 MHz section of the HF band centered on 15 MHz. This measurement was made with an electrically short antenna and no preamplifier. It shows various sources of noise including some from the Nooelec up converter. This experiment suggests the importance of a decent antenna, more pre-amplification and background subtraction.

10) Operating RASDR beyond the limits: SETI and weak signals

Extraterrestrial civilizations⁶ may use unknown modes for transmission of information. Nevertheless, they are probably constrained by physics (of which we know a part) to the electromagnetic spectrum for long-range transmissions. RASDR covers a useful section of this spectrum for communication with civilizations of the same level of technological development as our own.

Various techniques exist for pulling weak signals out of the noise. They include using larger antennas, employing lower-noise preamplifiers, signal averaging, and correlating received signals (mostly noise) with known reference values. This is a useful technique for pulsar discovery and investigation, and might also be useful for SETI (Search for Extraterrestrial Intelligence).

The physics of natural (non-life based) systems can generate periodic and structured signals. Examples include periodic solar emissions, Jovian Io-related radio emission ‘storms,’ pulsar signals, structured signals (Faraday bands) arising from ionospheric effects – and SETI. The Kepler mission searches a very tiny section of the sky, and thus a tiny nearby section of the Milky Way [44] The number of probable extra-solar planets discovered by the Kepler mission alone exceeds 4500 in number and a significant fraction of these lie in the ‘Goldilocks zone’ where life as we know it might prosper. There is an interesting reference to this SETI application of the product, available via YouTube. [31]

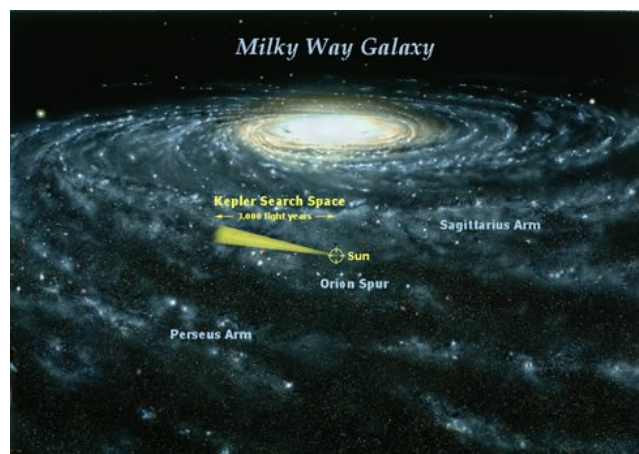


Figure 44 The Kepler mission search zone considers a tiny fraction of our galaxy.

⁶ We, all of us, are what happens when a primordial mixture of hydrogen and helium evolves for so long that it begins to ask where it came from. ~Jill Tarter

Furthermore, life may indeed inhabit artificial worlds [45] [46]. Thus, as Seth Shostak said, "...the broader point is that we now know two things that we didn't know 20 years ago. First those planets, including ones that might be like Earth, are incredibly plentiful in the visible universe. There could be a billion trillion cousins of our world. Second, life got started on Earth very early." Of course, coincident transmission and reception of radio signals from different solar systems requires degrees of proximity in both space and cultural level.⁷ Meanwhile, the rich physics of the universe provides many opportunities to detect and study weak signals.

The applicability of RASDR to SETI activities has been noted [47], and successful detection of signals from intelligent non-human life will probably one day be accomplished.

⁷ I think it quite likely that we are the only civilization within several hundred light years; otherwise we would have heard radio waves. ~*Stephen Hawking*

11) Appendices

a) RASDR Drivers

Microsoft Certified drivers are available for the RASDR device at rasdr.org.

Vendor ID ⁸	0x1D50
Product ID - RX	0x6099
Product ID - TX	0x609a
Product ID - Bootloader	0x609b

Table 11 - RASDR receiver USB IDs

Table 11 above details the USB IDs that are used by the RASDR receiver. There are some cases when an un-programmed board will identify itself as ‘Westbridge’ or ‘Cypress USB Bootloader’. If this is the case, please refer to section b) below on updating the firmware.

b) RASDR Firmware

To operate properly, the RASDR receiver must contain the VID/PID described in the above table. To support firmware updates, the receiver can be put into bootloader mode so that the firmware update can take place. This is done by disconnecting the RASDR receiver from the USB port on the computer and setting the switch to ‘PROG’ or removing the jumper on the J2 connector of the DigiRED board.

When the RASDR receiver is plugged into the USB again, it will identify itself as the bootloader above, or ‘Westbridge’⁹. Either way, the procedure for firmware update is the same:

1. Open the CyControl.exe program that is provided in the software distribution folder or ‘CyControl.zip’ from the rasdr.org site.
2. Select the ‘Cypress USB BootLoader’ in the window
3. Select Program->FX3->SPI FLASH
 - a. You may see a 2nd device driver association. If that occurs, simply wait until it is finished and then select the ‘Cypress USB BootProgrammer’
4. IMPORTANT: change the switch from ‘PROG’ to ‘RUN’ or fit the jumper on J2 connector of the DigiRED board. If you do not do this before the next step, the firmware will fail to upload.
5. Choose ‘usb_rx.img’ from the software distribution folder ‘firmware/’ or ‘D:\firmware\’ on the RASDR CD.
6. Wait until the ‘Programming of SPI Flash Succeeded’ message is shown. See Figure 48.
7. Disconnect the RASDR from the USB, wait a few seconds and then re-connect it. Ensure the switch is set to the ‘RUN’ position that was setup in step #4 above.
8. Observe that the ‘RASDR Radio Astronomy RX Interface’ is loaded.

⁸ Generously provided by OpenMoko

⁹ This can occur if the EEPROM ID has not been set, and may be seen on the pre-production RASDR receivers.

- a. You may need to re-associate the device driver, see Chapter 4)b) above.

Figure 45 thru Figure 49 detail the steps you are likely to encounter using the CyControl program.

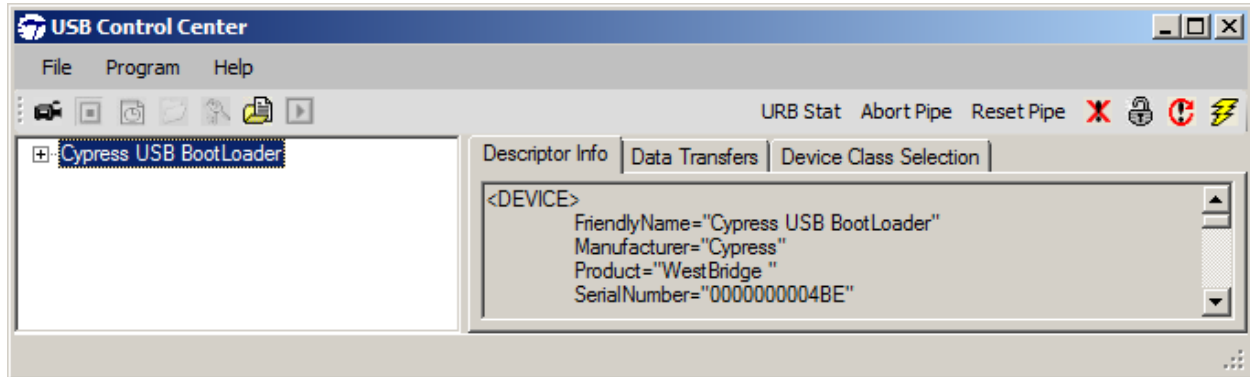


Figure 45 - CyControl program (Firmware update step 1)

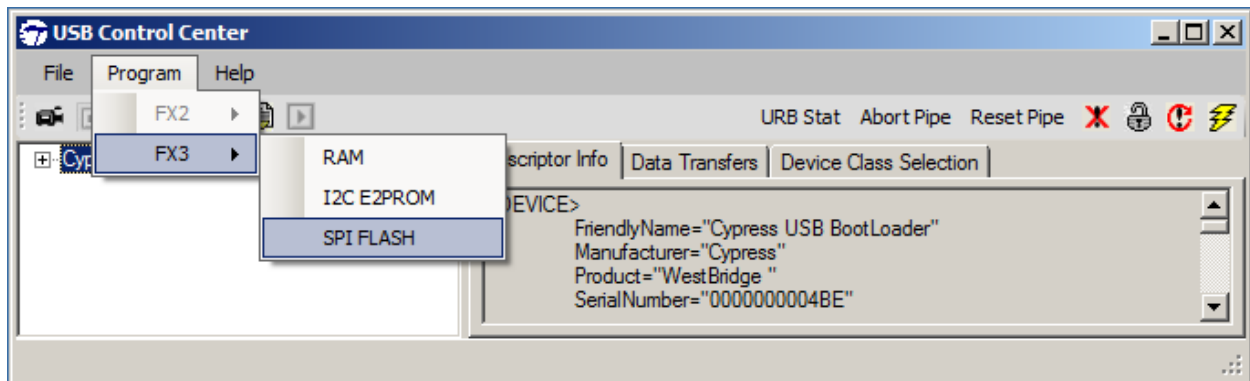


Figure 46 - CyControl program (Firmware update step 2)

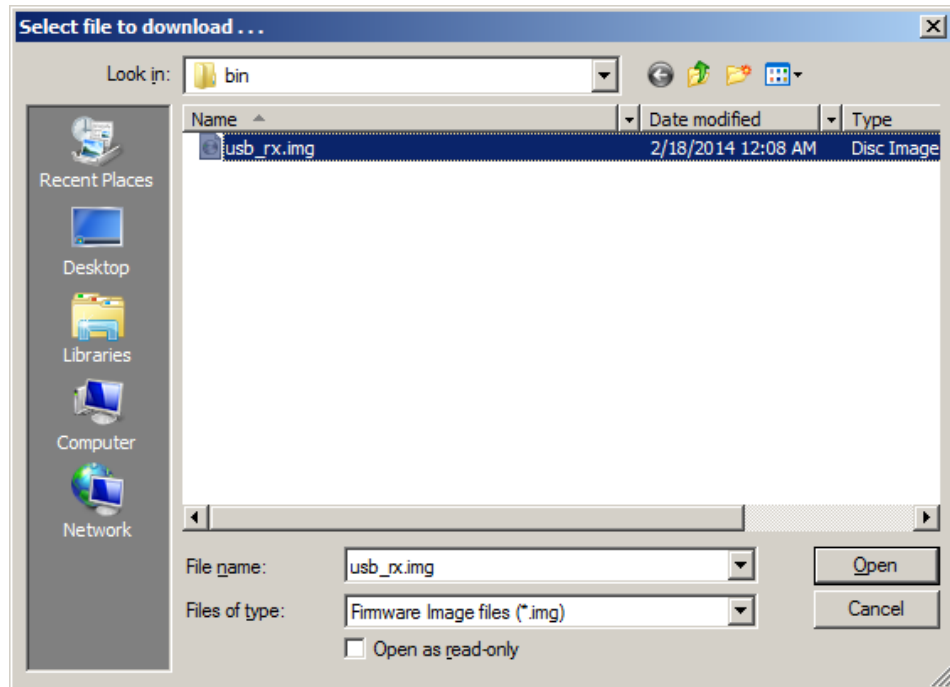


Figure 47 - Select programming file (Firmware update step 5)

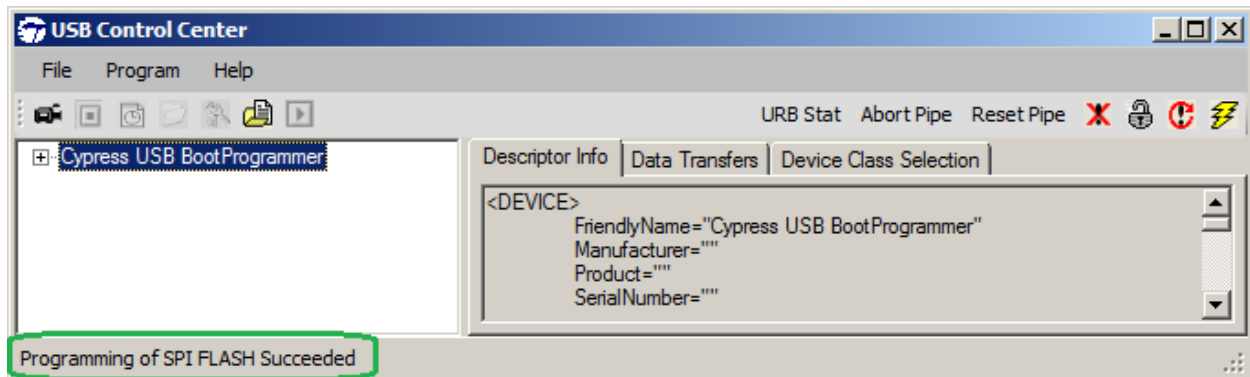


Figure 48 - Firmware Update Success

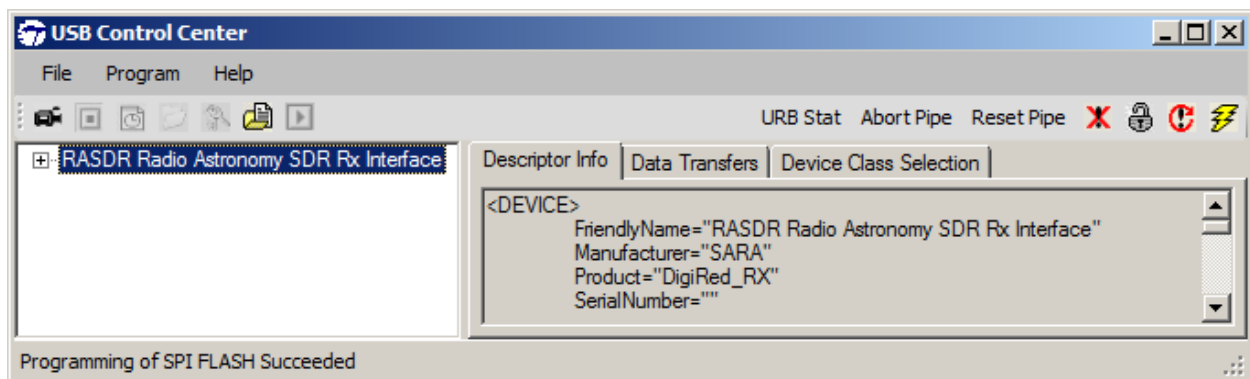


Figure 49 - Reinsert USB after Firmware update

c) **Firmware Command Set**

This section describes the command set implemented by the RASDR DigiRED firmware. To control various aspects of the DigiRED board, the following VENDOR command codes are defined using the following protocol:

- 1) the payload for all packets are fixed at 64 bytes in length
- 2) every request payload contains a 4 byte header followed by a data block starting at byte 4 (64 total bytes)
- 3) every response payload contains a 64 byte data block
- 4) the first byte of the header is a command code to the DigiRED control program on the FX3, the remaining three bytes are defined according to the command code or ignored as appropriate
- 5) A sequence of REQUEST OUT (VENDOR 0xC1) must always be followed by a RESPONSE IN (VENDOR 0xC0)
- 6) A RESPONSE IN shall never be sent before a REQUEST OUT

These are the commands that are supported by the firmware:

CMD_FX2 (0x40) - no operation

CMD_CFG_I2C_RD (0x15) - read I2C bus

```
//Command format:  
// src[0] : 0x15  
// src[1] : I2C address  
// src[2] : Data blocks in buffer  
// src[3] : reserved  
//Data block:  
// src[4] : register address  
// ...
```

Performs an I2C transaction with the base address given by src[1]. src[2] describes the number of transactions (n) to perform, while src[4]...src[4+n-1] are the register to read. Upon completion, the command will fill the transmit buffer with n values read. A subsequent VENDOR 0xC0 command will collect the transmit buffer.

See the [FFTviewer source](#) for specific details on the use of this command.

CMD_CFG_I2C_WR (0x14) - write I2C bus

```
//Command format:  
// src[0] : 0x14  
// src[1] : I2C address  
// src[2] : Data blocks in buffer  
// src[3] : reserved  
//Data block:  
// src[4] : register address  
// src[5] : data to write  
// ...
```

Performs an I2C transaction with the base address given by src[1]. src[2] describes the number of transactions (n) to perform, while src[4],src[5]...src[2+2n],src[3+2n] are the (register,value) pairs to write. The command does not modify the transmit buffer, but the protocol demands that a dummy VENDOR 0xC0 command must be issued.

See the [FFTviewer source](#) for specific details on the use of this command.

CMD_SET_LNA (0x30) - select input

```
//Command format:  
// src[0] : 0x30  
// src[1] : reserved  
// src[2] : 1  
// src[3] : reserved  
//Data block:  
// src[4] : new LNA value  
// ...
```

Sets signals on the MyriadRF board to select which LNA input on the LMS6002D is activated and connected to the RX SMA connector. The value provided in src[4] encodes the bits as follows:

Bit 0 - 0=LNA0 OFF, 1=LNA0 ON

Bit 1 - 0=LNA1 OFF, 1=LNA1 ON

These bits control the Peregrine Semiconductor PE42440 RF Switch under the following table:

LNA0	LNA1	Code	Function
OFF	OFF	0x0	Broadband, LNA3, 0.3-3.0GHz, 10dB NF
ON	OFF	0x1	Band XI, LNA2, 1.5-3.8GHz, 5.5dB NF
OFF	ON	0x2	50ohm termination
ON	ON	0x3	Band V, LNA1, 0.3-2.8Ghz, 3.5dB NF

The command does not modify the transmit buffer, but the protocol demands that a dummy VENDOR 0xC0 command must be issued.

See the [FFTviewer source](#) for specific details on the use of this command.

CMD_SET_PA (0x31) - enable power amplifier (enable TX)

```
//Command format:
// src[0] : 0x31
// src[1] : reserved
// src[2] : 1
// src[3] : reserved
//Data block:
// src[4] : new PA value
// ...
```

Sets signals on the MyriadRF board to select which LNA input on the LMS6002D is activated and connected to the RX SMA connector. The value provided in src[4] encodes the bits as follows:

Bit 0 - 0=PA0 OFF, 1=PA0 ON

These bits control the Peregrine Semiconductor PE42422 RF Switch under the following table:

PA0	Code	Function
OFF	0x0	Band I, TXOUT1
ON	0x1	Broadband, TXOUT2

The command does not modify the transmit buffer, but the protocol demands that a dummy VENDOR 0xC0 command must be issued.

See the [FFTviewer source](#) for specific details on the use of this command.

CMD_LMS_RESET (0x10) - reset transceiver

```
//Command format:
// src[0] : 0x10
// src[1] : reserved - FPGA Configuration I2C address
// src[2] : Data blocks in buffer
// src[3] : reserved
//Data block:
// src[4] : new reset pin level: 0 - low, 1 - high
// ...
```

Drives the RESET pin on the MyriadRF board to reset the LMS6002D. Multiple bytes can be used extend the reset pulse length. The command does not modify the transmit buffer, but the protocol demands that a dummy VENDOR 0xC0 command must be issued.

See the [FFTviewer source](#) for specific details on the use of this command.

CMD_LMS_RD (0x17) - read transceiver register

```
//Command format:  
// src[0] : Op code i.e. read from the LMS6002D Configuration via I2C/SPI  
// src[1] : reserved - FPGA Configuration I2C address  
// src[2] : Data blocks in buffer  
// src[3] : reserved  
//Data block:  
// src[4] : SPI Instruction (W/R + module addr + register address )  
// ...
```

Reads LMS6002D registers. src[2] describes the number of transactions (n) to perform, while src[4] ... src[4+n-1] are the the registers to read. Upon completion, the command will fill the transmit buffer with n values read. A subsequent VENDOR 0xC0 command will collect the transmit buffer.

See the [FFTviewer source](#) for specific details on the use of this command.

CMD_LMS_WR (0x16) - write transceiver register

```
//Command format:  
// src[0] : Op code i.e. read from the LMS6002D Configuration via I2C/SPI  
// src[1] : reserved - FPGA Configuration I2C address  
// src[2] : Data blocks in buffer  
// src[3] : reserved  
//Data block:  
// src[4] : SPI Instruction (W/R + module addr + register address )  
// src[5] : SPI Data  
// ...
```

Writes LMS6002D registers. src[2] describes the number of transactions (n) to perform, while src[4],src[5] ... src[2+2n],src[3+2n] are the (register,value) pairs to write. The command does not modify the transmit buffer, but the protocol demands that a dummy VENDOR 0xC0 command must be issued.

See the [FFTviewer source](#) for specific details on the use of this command.

CMD_GET_INFO (0x50) - query information about the FX3 controller firmware

```
//Command format:  
// src[0] : 0x50  
// src[1] : reserved  
// src[2] : reserved  
// src[3] : reserved  
//Data block:  
// src[4] : not used  
// ...  
// src[63]: not used
```

Obtains information on the firmware. Upon execution, fills the transmit buffer with 10 bytes of data:

Byte 0	'0x0' if the device is the Receiver and '0x1' if the device is the Transmitter (not used in RASDR)
Byte 1	contains the value returned by CyU3PGpifGetSMState(), see the Cypress documentation for details.
Byte 2-9	contain the ASCII code for the board serial number in 'nnn-nnnn' format.

A subsequent VENDOR 0xC0 command will collect the transmit buffer.

See the [FFTviewer source](#) for specific details on the use of this command.

CMD_ADF_WR (0xAD) - change frequency synthesizer parameters of ADF4002

```
//Command format:  
// src[0] : 0xAD  
// src[1] : reserved  
// src[2] : Data blocks in buffer  
// src[3] : reserved  
//Data block:  
// src[4] : data_h  
// src[5] : data  
// src[6] : data_l  
// ...
```

src[2] defines the number of bytes to send to the ADF, which must be formatted according to the protocol defined by the ADF4002 for register changes. Does not modify the contents of the return buffer, but the 0xC0 (RESPONSE IN) must be made, and the contents are to be ignored.

See the [FFTviewer source](#) for specific details on the use of this command.

CMD_GPIO_RD (0x19) - change GPIO pins to input and read state

```
//Command format:  
// src[0] : 0x19  
// src[1] : GPIO pin number to read  
// src[2] : reserved  
// src[3] : reserved  
//Data block:  
// src[4] : not used  
// ...  
// src[63]: not used
```

Turns the GPIO pin on the DigiRED expansion header into an input and reads its value. src[1] represents the GPIO pin to use which are described by the following table:

GPIO pin/Case PIN	DigiRED Connector.Pin
1	X3.3
2	X3.5
3	X3.7
4	X3.9

NOTE: the GPIO number and the Prototype Case connector pin are the same.

One word is written to the transmit buffer, representing the status of the command, as follows:

0x0	The pin is LOW
0x1	The pin is HIGH
0xFF	The gpio pin number was invalid, no operation performed

The 0xC0 (RESPONSE IN) VENDOR command must be issued to collect the transmit buffer.

CMD_GPIO_WR (0x20) - change GPIO pins to output and set value

```
//Command format:  
// src[0] : 0x20  
// src[1] : GPIO pin number to write  
// src[2] : reserved  
// src[3] : reserved  
//Data block:  
// src[4] : value to set. 0=off, 1=on  
// src[5] : not used  
// ...  
// src[63]: not used
```

Turns the GPIO pin on the DigiRED expansion header into an output and sets its value. src[1] represents the GPIO pin to use which are described by the following table:

GPIO pin/Case PIN	DigiRED Connector.Pin
1	X3.3
2	X3.5
3	X3.7
4	X3.9

NOTE: the GPIO number and the Prototype Case connector pin are the same.

One word is written to the transmit buffer, representing the status of the command, as follows:

0x0	The pin was set LOW
0x1	The pin was set HIGH
0xFF	The gpio pin number was invalid, no operation performed

The 0xC0 (RESPONSE IN) VENDOR command must be issued to collect the transmit buffer.

d) **Controlling GPIO**

All of the firmware functions can be accessed without software by using the CyControl application. It may be possible to control the GPIO via this interface if support is not provided in RASDRviewer or another program.

To read GPIO pins, the following steps are performed:

- 1) Setup an OUT transfer of type VENDOR with code 0xC1.
- 2) Fill 5 bytes of data with 0x19 N 0 0 0, where 'N' is the GPIO number (1-4) you wish to read. The 0x19 is the CMD_GPIO_RD code.
- 3) Ensure the number of bytes to send is 64 (it must always be 64).
- 4) Press 'Transfer Data'.
- 5) See Figure 50.
- 6) Change the code from 0xC0 and the direction to IN.
- 7) Press 'Transfer Data'.
- 8) See Figure 51.
- 9) The first byte of the return data represents the value of the GPIO read.

To write GPIO pins, the following steps are performed:

- 1) Setup an OUT transfer of type VENDOR with code 0xC1.
- 2) Fill 5 bytes of data with 0x20 N 0 0 V, where 'N' is the GPIO number (1-4) you wish to read, and 'V' is the value (0 or 1) you wish to set the GPIO to OFF or ON respectively. The 0x20 is the CMD_GPIO_WR code.
- 3) Ensure the number of bytes to send is 64 (it must always be 64).
- 4) Press 'Transfer Data'.
- 5) See Figure 52.
- 6) Change the code from 0xC0 and the direction to IN.
- 7) Press 'Transfer Data'.
- 8) See Figure 53.
- 9) The first byte of the return data should match the value of 'V' you chose. A '0xFF' value will indicate that the GPIO was not changed. For example, Figure 54.

Software may be written in Python or C++ or any convenient language that can "talk" to a USB device based on the firmware vendor command protocol. The following figures illustrate the above steps.

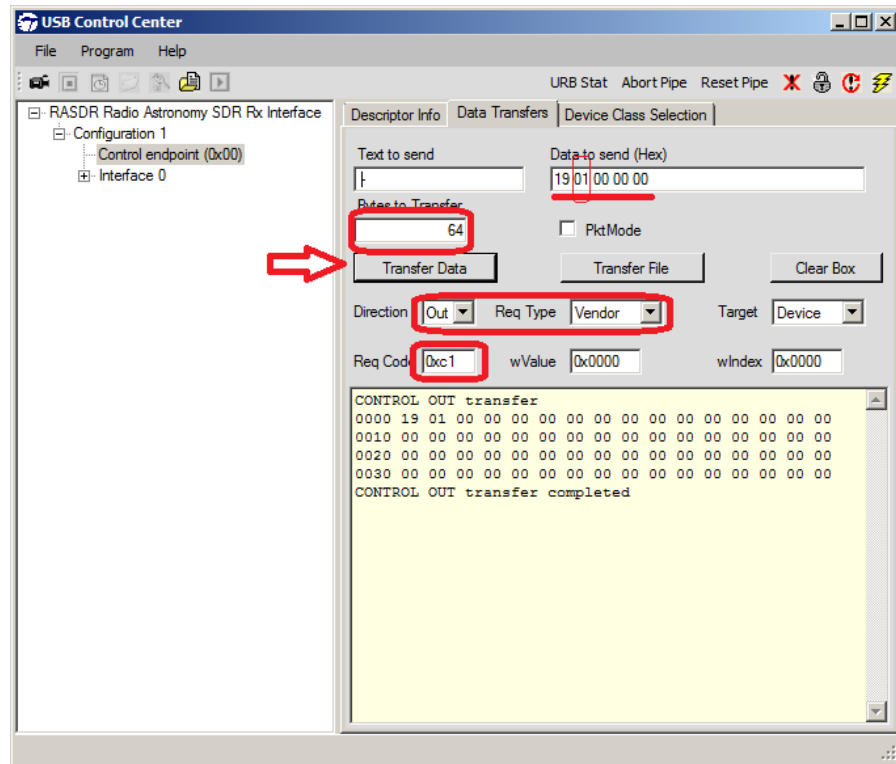


Figure 50 - GPIO Read Action (part #1)

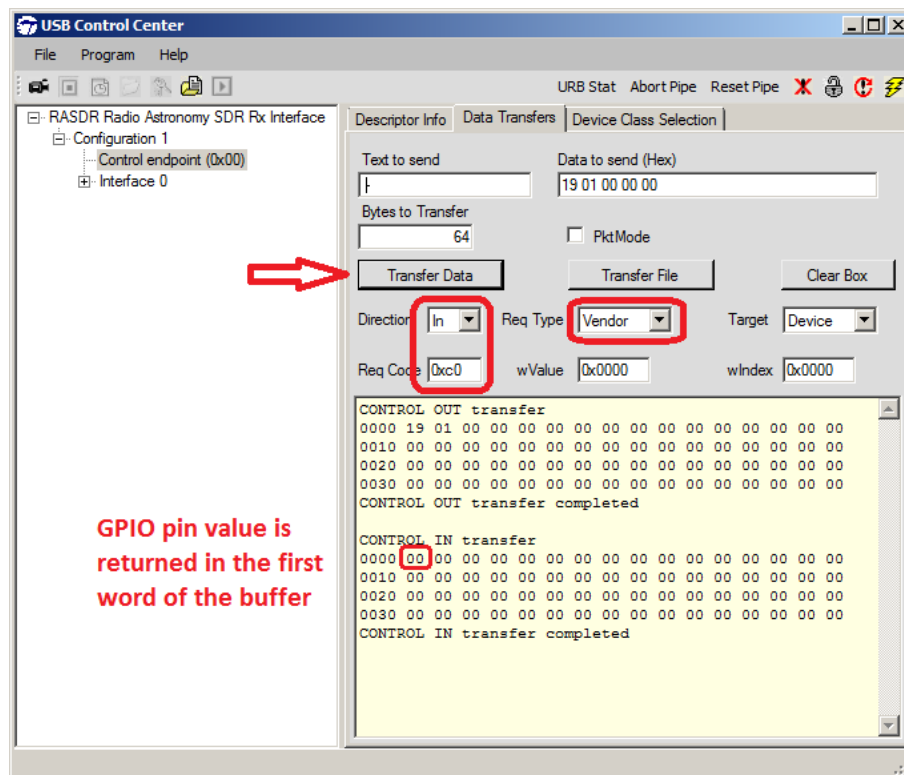


Figure 51 - GPIO Read Action (part #2)

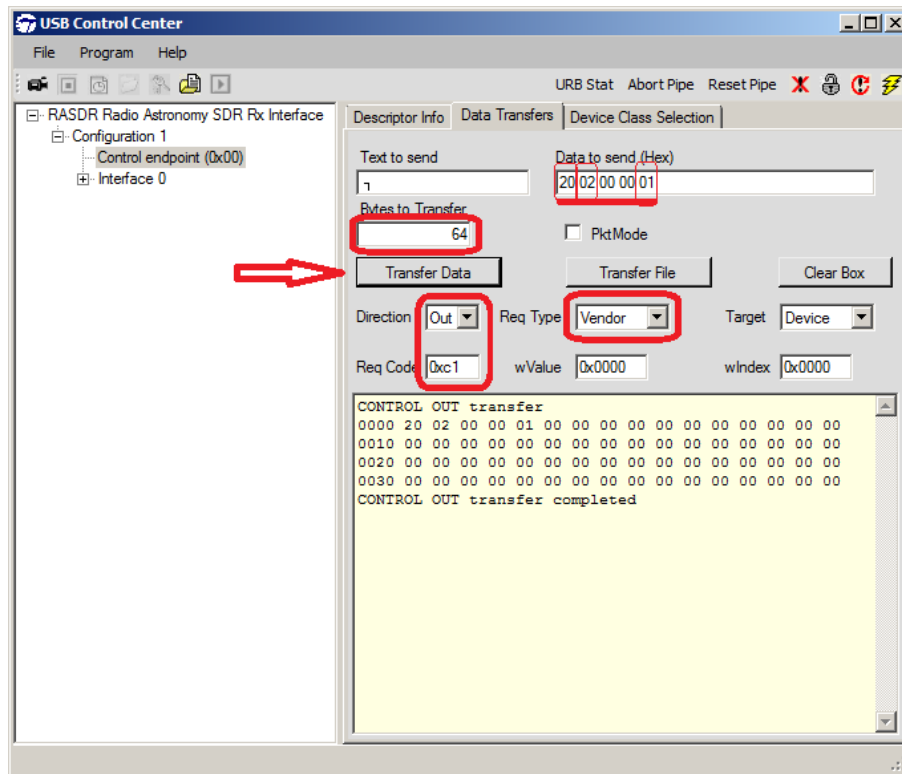


Figure 52 - GPIO Write Action (part #1)

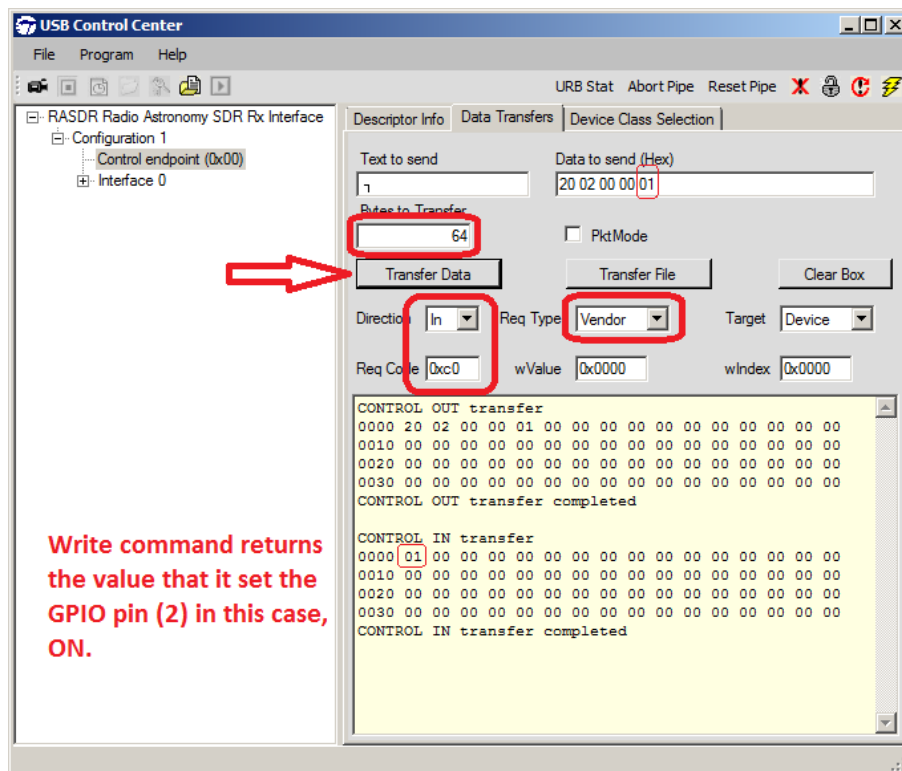


Figure 53 - GPIO Write Action (part #2)

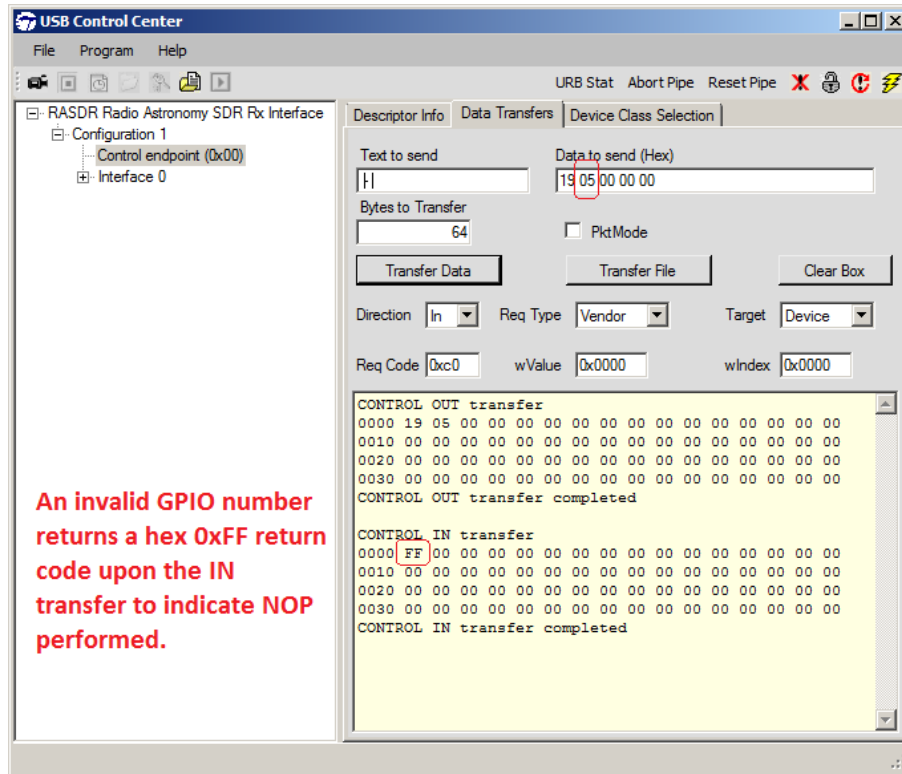


Figure 54 - GPIO Action with invalid pin number

e) RASDR Receiver Data Format and Endpoint

The RASDR firmware is written to transfer data from the LMS6002D device synchronously upon the activation of the sampling clock. As soon as the sampling clock starts, the data begins to fill the buffers in the FX3 USB chip. The buffers are available on Bulk Endpoint 0x81.

Filling these buffers will raise flow control signals on the FX3 chip, but there is no flow control between the LMS6002D and the FX3. Instead, two FX3 fifo flags are brought to the upper input pins of every data word. These flags allow samples to be invalidated if the host is not able to keep up with the data transfer.

The GPIO pin 'PPS' is likewise brought to the most significant bit of every data word. The indication of I or Q is also encoded within each sample so that software can properly interpret the stream.

Data Bit	Description
0-11	RXD[0..11]
12	RXIQSEL
13	FLAGA
14	FLAGB
15	PPS

The bit value of IQSEL should toggle between '1' (I) and '0' (Q) when sampling normally and without gaps. FLAGA and FLAGB should typically always be '1'. The PPS will have a value of '1' if nothing is connected to the pin and '0' when a signal is driving it. For example, Figure 55 and Figure 56 show representative data.

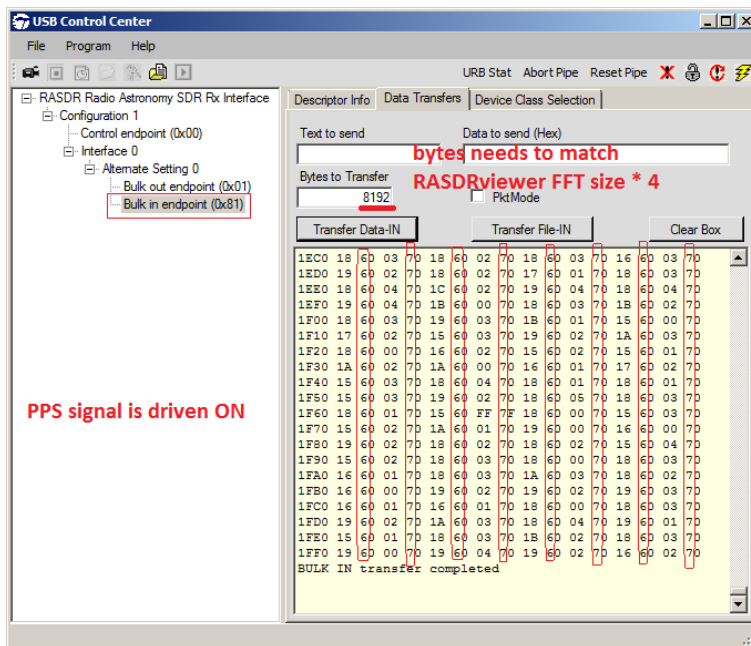


Figure 55 - RASDR Data with PPS signal ON (FFTsize 2048)

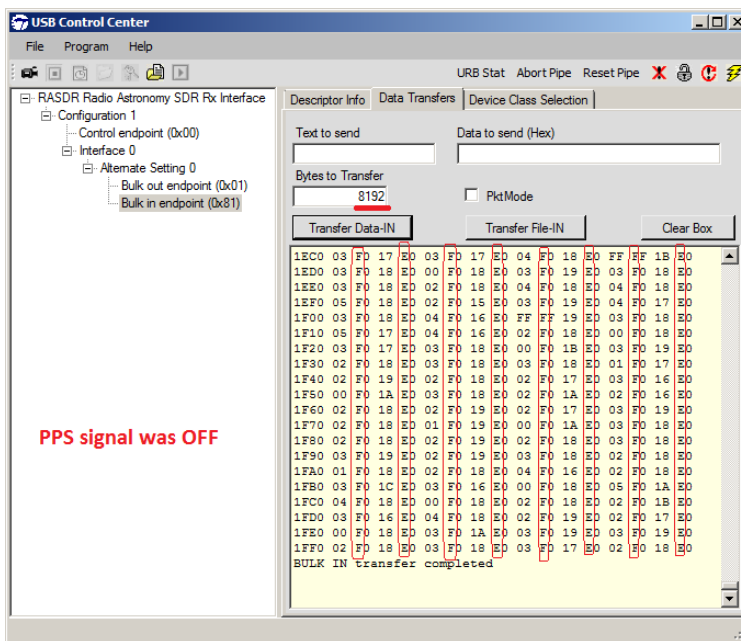


Figure 56 - RASDR Data with PPS signal OFF (FFTsize 2048)

f) **Troubleshooting**

It is recommended that the new user read (at least) the contents and Chapter 1 of this Guide. The FAQs may prove helpful.

A few users may have ill-advisedly skipped the README First (Chapter 1). This contains useful sources of information and specific links to the RASDR User's Group and other RASDR support groups, as well as email addresses for users with specific questions.

g) **General and Frequently-Asked Questions (FAQs)**

Q: Will RASDR work using USB2 cables/connectivity?

A: Yes, but with reduced bandwidth. An adapter can be added to permit USB3 input, but not all such USB2/USB3 adapters are equivalent, since different chipsets are used by different vendors. For desktop PCs, good results have been obtained with the Konig Electronic CMP-PCIE2USB3 pci express card with the NEC/Renesas UPD720202 chipset, available from Amazon for about \$36. An ASUS u3s6 PCIE didn't work well. An Inateck KT4007 board functioned well in tests.

Q: May I use an old computer?

A: Older computers that don't have sufficient graphics OpenGL capability may need to be updated. NVIDIA graphics cards made after 2009 are probably OK.

Q: How do I avoid Excel files becoming large and unmanageable?

A: Set the file "save spectral span" to a subset of the displayed BW. The display FFT span is intended to be a way to zero in on a narrow portion of the bandwidth. It is used to obtain a smaller number of columns on the output FFT file. This is done by selecting the option on the file setup screen. The setting of the span to match the sample rate is intended to prevent displaying meaningless data.

You can also edit the resulting .csv file in a text editor before loading into Excel to reduce the number of rows in the file.

Q: How significant is the Doppler shift for H1 measurements?

A: We see signals from neutral hydrogen atoms (H1 emission) at different frequencies. All those H1 sources are in motion, and a good rule of thumb is to examine the velocity Doppler shift over

a 1.5MHz section of spectrum. One can compute the velocity v for a given frequency from the Doppler shift:

$$\frac{v}{c} = \Delta f / f$$

where v is the velocity in question, c is light speed (3e8)m/s, Δf is the frequency range (1.5e6)Hz, and f is 1.42e9 Hz.

This gives $v = 316$ km/s, so the conversion factor is 3.16km/s per 1.5MHz.

Q: I am thinking about adding an RASDR2 that you describe and another antenna, probably a dish with much larger aperture. Other than the limitation you mention about SPCY (5kHz spectra mode bandwidth), how do you think it will compare with the RASDR2 ?

A: I like both instruments. SPCY is a great special-purpose instrument and if you want only to record H1 spectra, then it's the way to go. But RASDR2 can be reconfigured by software to do a variety of things. I hope that you will work with us on some creative projects.

Q: I realize that is a very open ended question, but I am thinking about stability of the oscillators to do Radio Astronomy as well as ability to do good radio astronomy and overall ease of use for someone who is not good with software and no programming ability?

A: RASDR2 has a very stable TCXO. In addition, it has an SMA connector to connect to an external GPS-stabilized frequency reference. These GPS-stabilized sources are available from eBay. The old ones used Trimble GPS and were about \$300 but the new ones (TCXO or Rb clock versions) appear to be about \$140. I haven't used them, but we stabilized to the NRAO facility clock last year for measurements at the SARA meeting.

Q: Does the RASDRviewer have the math programs to do FFT analysis in it or does it access the math software come from some other site?

A: It contains the math programs. The FFT analysis in RASDRviewer is done by FFTW, developed at MIT.

Q: Why does data collection using RASDR2 stall when using USB3?

A: Some computers, operating systems, USB3 driver sets and environments can introduce latency in servicing data pipelines of USB devices. The current RASDR2 firmware use BULK I/O endpoints that do not have guaranteed bandwidth or timing requirements. This can lead to transient situations in which the buffers in the USB chip fill up.

When this happens, sometimes the device driver context for the application making the requests stalls and will not provide more data to the application until the device is closed and reopened, and the transfers restarted. Prior to the firmware release of June 2016, this happened almost immediately for any USB3 usage on some computers. The problem was significantly reduced by not using the USB3 burst mode (previously set to 4 x 1K, now using only transfers of 1K).

This did not eliminate the stall possibility, so that a software solution is still required. We implemented this for RASDRstreamer and RASDRproc in the 1.2.4 software distribution released in June 2016. For RASDRproc, the function is enabled by ticking the 'Auto Restart' box in the upper right corner of the application.

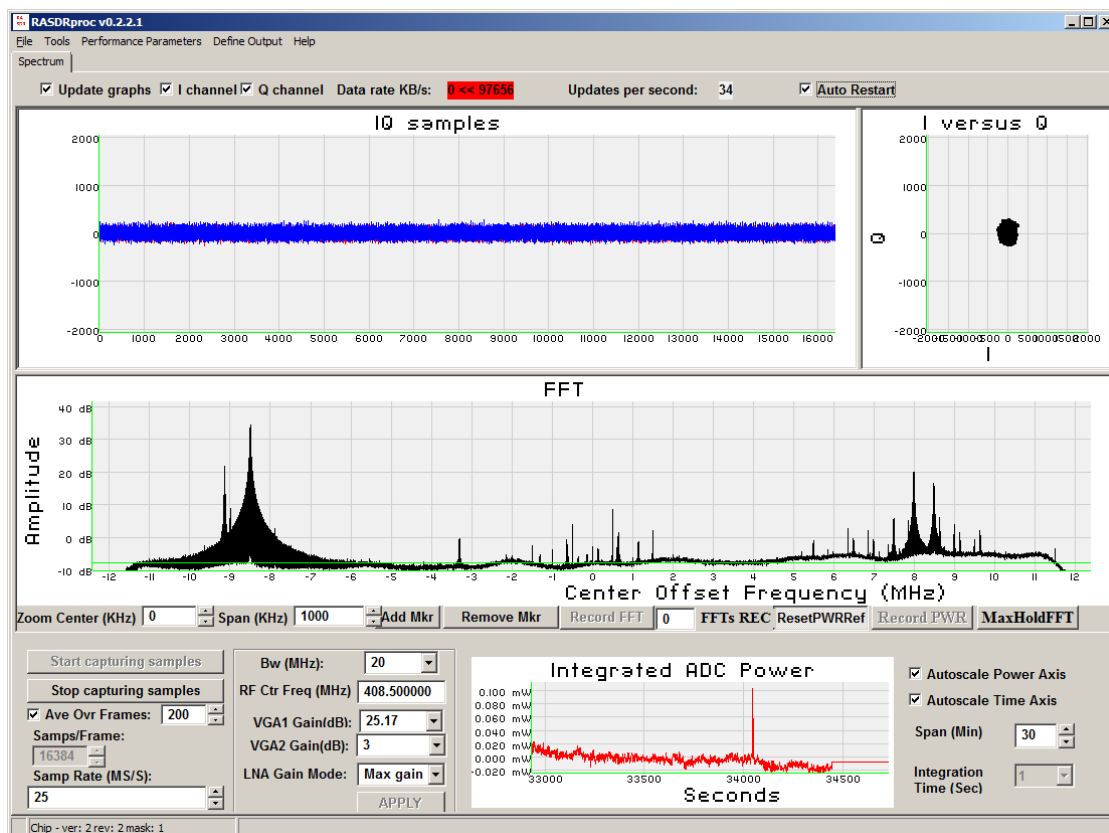


Figure 57 - Unrecoverable stall using USB3

As Figure 57 shows, it is still possible to enter an unrecoverable condition that requires the RASDR2 to be disconnected and then reconnected to the USB in order to resume capture.

h) RASDR & Myriad RF connectors details

X1 Connector - DigiRed to Myriad

The Myriad-RF board X3 connector (type FX10A-80P0) is pin compatible with J1 connector on interface board. It provides the digital and SPI interface for LMS6002DFN together with the supply voltage and GPIO control for RF switches for Myriad-RF board. The pin descriptions on this connector are given in the table below:

Pin No	Pin Name	Type	Description
1	+5 V	in DC	+5 V power supply
2	+5 V	in DC	+5 V power supply
3	+5 V	in DC	+5 V power supply
4	+5 V	in DC	+5 V power supply
5	GND		Ground pin
6	GND		Ground pin
7	+3.3V	in DC	+3.3 V power supply optional
8	+3.3 V	in DC	+3.3 V power supply optional
9	+3.3V	in DC	+3.3 V power supply optional
10	+3.3V	in DC	+3.3 V power supply optional
11	GND		Ground pin
12	GND		Ground pin
13	-		Not used
14	-		Not used
15	-		Not used
16	-		Not used
17	GND		Ground pin
18	GND		Ground pin
19	TXIQSEL	in cmos	TX digital interface IQ flag
20	-		Not used
21	-		Not used
22	-		Not used
23	TXD0	in cmos	DACs digital input, bit 0 (LSB)
24	TXD1	in cmos	DACs digital input, bit 1
25	TXD2	in cmos	DACs digital input, bit 2
26	TXD3	in cmos	DACs digital input, bit 3
27	GND		Ground pin
28	GND		Ground pin

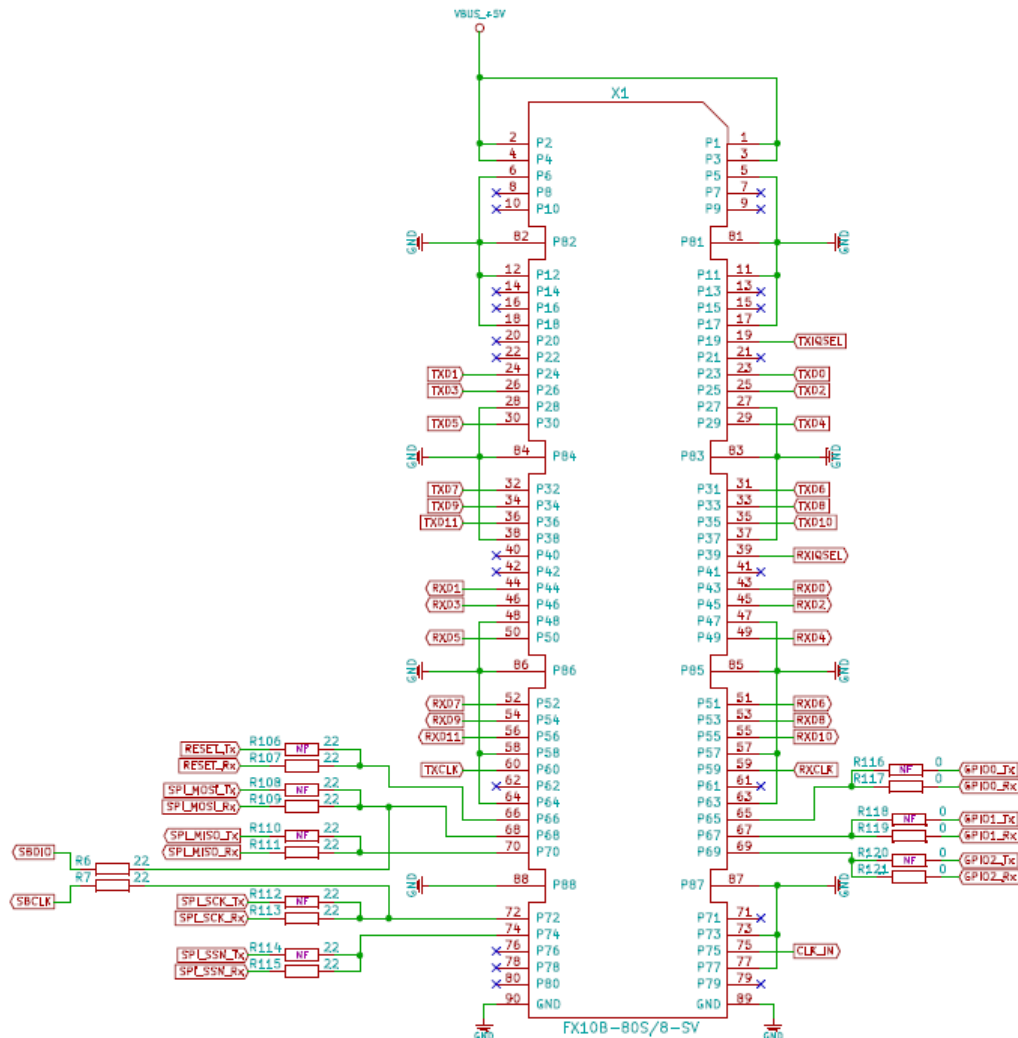
29	TXD4	in cmos	DACs digital input, bit 4
30	TXD5	in cmos	DACs digital input, bit 5
31	TXD6	in cmos	DACs digital input, bit 6
32	TXD7	in cmos	DACs digital input, bit 7
33	TXD8	in cmos	DACs digital input, bit 8
34	TXD9	in cmos	DACs digital input, bit 9
35	TXD10	in cmos	DACs digital input, bit 10
36	TXD11	in cmos	DACs digital input, bit 11 (MSB)
37	GND		Ground pin
38	GND		Ground pin
39	RXIQSEL	out cmos	RX digital interface IQ flag
40	-		Not used
41	-		Not used
42	-		Not used
43	RXD0	out cmos	ADCs digital output, bit 0 (LSB)
44	RXD1	out cmos	ADCs digital output, bit 1
45	RXD2	out cmos	ADCs digital output, bit 2
46	RXD3	out cmos	ADCs digital output, bit 3
47	GND		Ground pin
48	GND		Ground pin
49	RXD4	out cmos	ADCs digital output, bit 4
50	RXD5	out cmos	ADCs digital output, bit 5
51	RXD6	out cmos	ADCs digital output, bit 6
52	RXD7	out cmos	ADCs digital output, bit 7
53	RXD8	out cmos	ADCs digital output, bit 8
54	RXD9	out cmos	ADCs digital output, bit 9
55	RXD10	out cmos	ADCs digital output, bit 10
56	RXD11	out cmos	ADCs digital output, bit 11 (MSB)
57	GND		Ground pin
58	GND		Ground pin
59	RXCLK	in cmos	RX digital interface clock
60	TXCLK	in cmos	TX digital interface clock
61	-		Not used
62	-		Not used
63	GND		Ground pin
64	GND		Ground pin
65	GPIO0		

66	RESET	in cmos	Hardware reset, active low
67	GPIO1		
68	SPI_MOSI	out cmos	Serial port data out
69	GPIO2		
70	SPI_MISO	in/out cmos	Serial port data in/out
71	-		Not used
72	SPI_CLK	in cmos	Serial port clock, positive edge sensitive
73	GND		Ground pin
74	SPI_NCSO	in cmos	Serial port enable, active low
75	CLK_IN	in cmos	PLL reference clock input
76	-		Not used
77	GND		Ground pin
78	-		Not used
79	TXEN	in cmos	Transmitter enable, active high
80	RXEN	in cmos	Receiver enable, active high
81	GND		Ground pin
82	GND		Ground pin
83	GND		Ground pin
84	GND		Ground pin
85	GND		Ground pin
86	GND		Ground pin
87	GND		Ground pin
88	GND		Ground pin

Table 12 X3 connector pin description

A schematic diagram of the connector is provided below:

The Myriad-RF board is directly plugged into the X1 connector. The digital I/Q connector is a digital transmit (TX) and receive (RX) interface to the ADC/DAC of the LMS6002D.



X2 Connector - FX3 JTAG interface

FX3's JTAG interface has a standard five-pin interface to connect to a JTAG debugger in order to debug firmware through the CPU-core's on-chip-debug circuitry. The FX3's is selected via the JTAG switch.

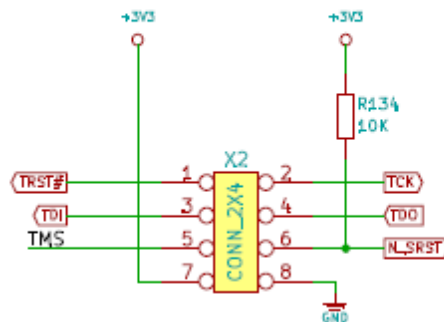
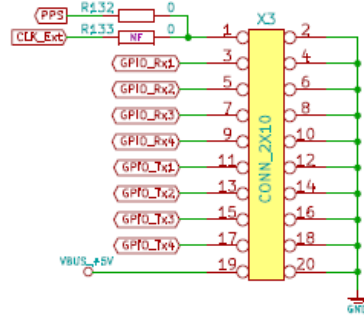


Figure 58 JTAG connector.

NOTE: the JTAG connector will not be populated in production RASDR devices.

X3 Connector - PPS and GPIO Connector



A GPIO connector is available on the DigiRED board for expansion. This connector is brought out in the prototype RASDR as describes in the following table:

GPIO pin/Case PIN	DigiRED Connector.Pin	Function
1	X3.3	GPIO (+3.3V, +5V tolerant)
2	X3.5	GPIO (+3.3V, +5V tolerant)
3	X3.7	GPIO (+3.3V, +5V tolerant)
4	X3.9	GPIO (+3.3V, +5V tolerant)
5	X3.1	PPS (+3.3V, +5V tolerant)
6	GND	GND
7	GND	GND
8	X3.19	+4.3V, 250mA supply

Table 13 - GPIO Assignments

The PPS pin is specially routed to the highest-order bit (bit 15) of the digital I/Q data input. As such it is sampled synchronously at the sampling rate used for the device. Software can use this transition to refine the timestamp. When the PPS signal is provided by a GPS or other synchronized source, any number of RASDRs can be synchronized to each other.

The voltage output pin is provided to allow an up converter or small device to be powered by the RASDR. A diode is fitted between X3.19 and the case pin 8. This prevents current from flowing back through the USB connector if a large voltage is inadvertently applied to the voltage pin.

The GPIO pins all have Zener diodes to clamp the input to +5.1V, in case a large voltage is inadvertently applied.

i) X7 Connector - Input Reference Frequency Provision

The Input Reference Frequency has performed well in tests at NRAO, where a 10MHz facility reference signal was used.. As indicated above, the PPS feature is currently not supported, but is expected to be activated in future firmware, by providing 1Hz 3.3V pulse with 10% duty cycle on the port.

CLK_Ext will provide the clock from the SI5356 to the user if required.

The 10 MHz reference input drives an Analog Devices ADF4002 PLL chip on the DigiRED board. When the reference frequency is present and the PLL is locked to it, the green LED D9 (see Figure 59) will be illuminated.

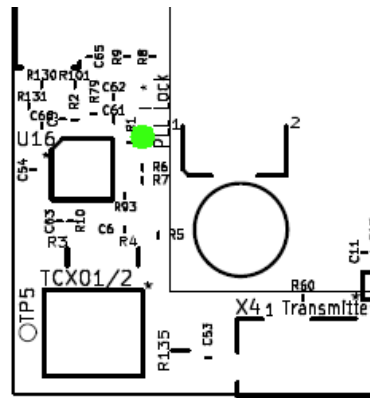


Figure 59 LED D9 is illuminated green to denote frequency lock.

The input signal should be a square wave or clipped sine wave with LOW/HIGH voltage of 0 and 3V (2.7 - 3.3V max value range). The ADF4002 reference input is a CMOS type with a nominal threshold of 1.5V and a dc equivalent input resistance of 100 k Ω . Thus expected current draw from the reference is in the order of 100 μ A.

The RASDR firmware is fixed to use 10 MHz as an input, custom programming of the ADF4002 registers can be performed. The ADF4002 feeds the clock distribution chip SI5356. The clock distribution chip feeds the down conversion Rx PLL as well as the ADC sample clock. The software sets the values of these devices in a transparent manner to the user. For example when the user changes the RF center frequency on the screen appropriate registers are set via the SPI bus to change the frequency. Likewise, registers are set when the user changes the sampling rate.

An example reference clock could be based upon either an OCXO or MASER clock, possibly locked to GPS time to provide the PPS signal.

12) References

- [1] SARA, "RASDR – Radio Astronomy Software Defined Radio," 2015. [Online]. Available: <http://www.radio-astronomy.org/rasdr>. [Accessed 2015].
- [2] Lime, "Lime Microsystems Field Programmable RF (FPRF) Transceiver IC," Lime Microsystems, 2015. [Online]. Available: <https://github.com/myriadrf/LMS6002D-docs>.
- [3] Digikey, "Digi-Key Catalog," Digikey, 2014. [Online]. Available: <http://www.digikey.com/product-search/en?x=0&y=0&lang=en&site=us&keywords=myriadRF>.
- [4] RASDRgroup, "RASDR – Radio Astronomy Software Defined Radio," 2014. [Online]. Available: <http://rasdr.org>.
- [5] Users, ""RASDR User's Group"," 2015. [Online]. Available: <https://groups.yahoo.com/neo/groups/RASDR/info>.
- [6] Lime Microsystems, Ltd., "MyriadRF /Discourse for RASDR," [Online]. Available: <https://discourse.myriadrf.org/c/projects/rasdr>. [Accessed 6 10 2015].
- [7] Github, 2015. [Online]. Available: <https://github.com/myriadrf/RASDR>.
- [8] P. Oxley, B. Vacaliuc, D. Fields, C. Lyster, S. Kurtz and Z. Tamosevicius, "RASDR update," *Radio Astronomy -- Journal of the Society of Amateur Radio Astronomers*, no. January-February, p. 72ff, 2014.
- [9] LimeMicrosystems, "Field Programmable RF (FPRF) Tranceiver IC," Lime Microsystems, 2014. [Online]. Available: <https://github.com/myriadrf/LMS6002D-docs>.
- [10] Myriadrf, "Home page of Myriadrf," [Online]. Available: <http://myriadrf.org/>. [Accessed 20 May 2013].
- [11] Myriad, *Personal Communications to RASDR team*, 2013.
- [12] P. Oxley, D. Fields and S. Kurtz, "RASDR2 Control and Analysis Software," *Proceedings of the Annual Meeting of the Society of Amateur Radio Astronomers. NRAO, Green Bank WV*, 2014.
- [13] B. Vacaliuc, D. Fields, P. Oxley, S. Kurtz, C. Lyster, R. Vadoklis and Z. Tamosevicius, "Production Manufacturing Plan for RASDR2 Appliance," *Proceedings of the Annual*

- Meeting of the Society of Amateur Radio Astronomers. NRAO, Green Bank WV, 2014.*
- [14] SARA, "Society of Amateur Radio Astronomers Home Page," 2015. [Online]. Available: www.radio-astronomy.org.
 - [15] Wikipedia, "Radio_spectrum," 2014. [Online]. Available: http://en.wikipedia.org/wiki/Radio_spectrum. [Accessed 11 May 2014].
 - [16] D. Fields and N. Tesla, "Detection of Jupiter Radio Emissions below the Plasma Cutoff Frequency: Implications for SID Monitoring," *Annual Meeting of SARA at NRAO, Green Bank, WV*, 2011.
 - [17] J. Avellone, "Detection of Jovian VLF Noise," *Radio-Astronomy Journal*, no. March - April, p. 48ff, 2014.
 - [18] D. Fields, R. Kennedy and K. Roy, "Interplanetary Radio Transmission through Serial Ionospheric and Material Barriers," *Genta, Giancarlo, Ed; Proceedings of the Severnth IAA Symposium on Relalistic Near-Term Advanced Scientific Space Missions: Missions to the outer solar system and beyond.*, 2011.
 - [19] A. R. R. L. ARRL, "Radio Amateur's Handbook (purchase)," ARRL, 2015. [Online]. Available: <http://www.arrl.org/shop/ARRL-Handbook-2013-Softcover-Edition>.
 - [20] SARA, "Radio Jove Project," 2015. [Online]. Available: <http://radio-astronomy.org/node/211>.
 - [21] L. M. S. LMS RDK, "myriad-RF reference development kit," 2015. [Online]. Available: <https://myriadrf.org/projects/rdk/>.
 - [22] L. M. S. LMS, "myriadRF projects," 2015. [Online]. Available: <https://myriadrf.org/projects/lms-suite/>.
 - [23] P. Oxley, D. Fields and S. Kurtz, "RASDRWin: Companion Software for RASDR," *Proceedings of the Annual Meeting of the Society of Amateur Radio Astronomers*, no. July 14-17, 2013.
 - [24] P. Oxley and D. K. S. Fields, "RASDRviewer: RASDR2 Control and Analysis Software," *Proceedings of the Annual Meeting of the Society of Amateur Radio Astronomers*, no. June 29-July 2, 2014.
 - [25] RASDRviewer, "RASDRviewer downloads," 2015. [Online]. Available: <https://groups.yahoo.com/neo/groups/RASDR/files/RASDRviewerSoftware/>.

- [26] D. Fields, P. Oxley, B. Vacaliuc, S. Kurtz, C. Lyster, C. S. Z. Tamosevicius and S. Berl, "Radio Astronomy with RASDR2," *Proceedings of the Annual Meeting of the Society of Amateur Radio Astronomers. June 29-July 2, 2014, at NRAO, Green Bank WV*, 2014.
- [27] P. Oxley, D. Fields, B. Vacaliuc and C. L. a. F. Ghigo., "Detection of Narrow Spectral Features using RASDR2 and the NRAO 20m Telescope," vol. p.90, no. July-Aug, 2014.
- [28] J. Bernard, "SpectraCyber I/II™ 1420 MHz Hydrogen Line Spectrometer Installation and Operation Manual No. SC1/2-2006," Carl Lyster through Radio Astronomy Supplies, 2010. [Online]. Available: http://www.ncra.tifr.res.in/rpl/facilities/4m-srt/SCI_IIManual.pdf.
- [29] D. F. a. R. group, ""Radio Astronomy with Small Antennas using RASDR"," 2015. [Online]. Available: <https://groups.yahoo.com/neo/groups/RASDR/files>. [Accessed 2015].
- [30] RASDRteam, "RASDR – Radio Astronomy Software Defined Radio," 2014. [Online]. Available: <http://rasdr.org>.
- [31] D. Fields and P. Shuch, "Frontiers of Radio Astronomy & SETI," Third Tennessee Valley Interstellar Workshop, Nov 10-11, 2014, Oak Ridge, TN, [Online]. Available: <https://www.youtube.com/watch?v=Ci4cwZtub5U>. [Accessed 2015].
- [32] P. Shuch, "SETI Horn of Plenty: An Argus Antenna Alternative," CQ- VHF Ham Radio above 50MHz, Sept. 2003. [Online]. Available: <http://www.setileague.org/articles/ham/horn.pdf> . [Accessed 2015].
- [33] H. Shuch, "SETI Horn of Plenty construction photos," The SETI League, [Online]. Available: <http://www.setileague.org/photos/wghorn.htm> . [Accessed 2015].
- [34] P. Shuch, "SETI Horn of Plenty," SETI League, [Online]. Available: <http://www.setileague.org/articles/horn.htm>. [Accessed 2015].
- [35] R. W. World, "Horn Antenna Calculator," [Online]. Available: <http://www.rfwireless-world.com/calculators/Horn-Antenna-Calculator.html>. [Accessed 2015].
- [36] R. Postprocessing, "Postprocessing with Python," 2015. [Online]. Available: <https://groups.yahoo.com/neo/groups/RASDR/files/PostprocessingSoftware/>.
- [37] NRAO, "NATIONAL RADIO ASTRONOMY OBSERVATORY 40-FOOT RADIO TELESCOPE OPERATOR'S MANUAL," 2014. [Online]. Available: <http://www.gb.nrao.edu/epo/manual.pdf>.
- [38] Ebay, "Raltron CO19025-100.000MHz," 2014. [Online]. Available: www.ebay.com.

- [39] Raltron, "Through hole clock oscillators – 3 state HCMOS compatible," Raltron, 2014. [Online]. Available: http://www.raltron.com/products/pdfsspecs/clock_co15-co19.pdf.
- [40] Miltron, "LFCN-225," Mitron, 2014. [Online]. Available: <http://www.mitron.cn/cxchanpin/Filters-n/LFCN-225.pdf>.
- [41] Hewlett-Packard, "Silicon Bipolar MMIC 5 μ GHz," 2014. [Online]. Available: <http://www.qsl.net/n9zia/omnitracs/IAM81008.pdf>.
- [42] Minicircuits, "Plugin High Pass Filter," [Online]. Available: <http://www.minicircuits.com/pdfs/PHP-100+.pdf>.
- [43] Nooelec, "Ham It Up v1.2 - RF Upconverter For Software Defined Radio," Nooelec, 2014. [Online]. Available: <http://www.noelec.com/store/ham-it-up-v1-0-rf-upconverter-for-software-defined-radio.html>.
- [44] NASA, "Kepler Discoveries," NASA, 2015. [Online]. Available: <http://kepler.nasa.gov/Mission/discoveries/>.
- [45] K. Roy, R. Kennedy and D. Fields, ""Shell Worlds" in Genta, Giancarlo, Ed; Proceedings of the Severnth IAA Symposium on Relativistic Near-Term Advanced Scientific Space Missions: Missions to the outer solar system and beyond. International Academy of Astronautics, Aosta, IT.," 2011.
- [46] D. Fields, R. Kenendy, K. Roy and B. Vacaliuc, "Interplanetary radio transmission through serial ionospheric and material barriers. Acta Astronautica," no. 82, 2, pp 251-256, 2013.
- [47] D. Fields, S. Kurtz, P. Shuch, S. Hogle, B. Vacaliuc, P. Oxley, K. Roy and R. Kennedy, *Algorithmic Communication Structures for Interstellar Travel and SETI*, Huntsville, AL: Second Tennessee Valley Interstellar Workshop, 2013.
- [48] P. Oxley, D. Fields and S. Kurtz, "RASDR2 Control and Analysis Software," in *Proceedings of the 2014 SARA Conference*, Green Bank, WV, 2014.

13) Index

Add Marker, 36
Architecture, 34
artificial worlds, 55
Basic Radio Astronomy, 40
display FFT span, 71
Excel files, 71
External frequency input, 31
FAQs, 71
Faraday bands, 54
FFT output file contains zero bytes, 38
FFT Output to Disk, 37
firmware, 34
Goldilocks zone, 54
Graphical User Interface, 35
graphics GL capability, 71
Installing RASDRviewer, 34
internal noise floor, 43
Kepler, 54
LED D9, 31
LED indicators, 30
Maximum internal RF gain, 14
number of FFTs to average, 36
Nvidia graphics cards, 34
old computer, 71, 72
pci express card, 30
post-processing, 43
Postprocessing, 47
RASDR noise floor, 43
reference clock, 79
save spectral span, 71
Serial Peripheral Interface (SPI) bus, 34
SpectraCyber, 40
spectral span, 71
System Gain Considerations, 42
USB2 cables/connectivity, 71
USB2/USB3 adaptor, 30
USB2/USB3 adaptors, 71
VHF band, 48