SERENDIP6 User Manual-V1.0.5

Wei Liu, Dan Werthimer

February 21, 2023

Contents

1 Overview

The SERENDIP6 SETI spectrometer is a 512M channel spectrum analyzer that analyzes a 500 MHz bandwidth, providing roughly 1Hz spectral resolution. The IF signal from the telescope is digitized by a dual 8 bit ADC board sampling at 1GSps. The 1GHz sampling clock signal is provided by a synthesizer, which is phase locked to the observatory frequency reference. The ADC board (CASPER ADC2x1000-8) and FPGA board (CASPER ROACH2) are open source, as well as the FPGA code, the GPU code and the CPU code.

The ADC board samples are fed to an FPGA board which packetizes the samples into 10Gbit/sec ethernet packets and adds a time stamp into each packet. The packets are sent to a dual 10Gbit network interface PCIe board in the server. The server uses open source data transport software framework called "HASHPIPE" to move the time domain data into a pair of GPU's (one GPU for each polarization). Each GPU computes a real to complex transform (1G real samples input, 512M complex samples output), then calculate the power spectrum, then does box car smoothing to calculate a smoothed average baseline for the spectrum, and then searches for spectral peaks above the average power. Strong signals over threshold are written to a FITs file along with metadata that is sent to the server from the observatory computing system.

The block diagram of the system is shown below.

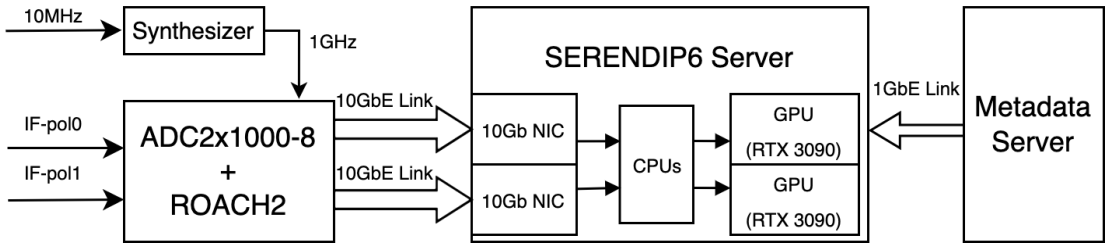

Figure 1: The block diagram of the system.

2 Hardware requirements

The SERENDIP6 receives packets from ROACH2 board, which connects an ADC2x1000-8 board. The ADC board has to be attached to **ZDOK0**.

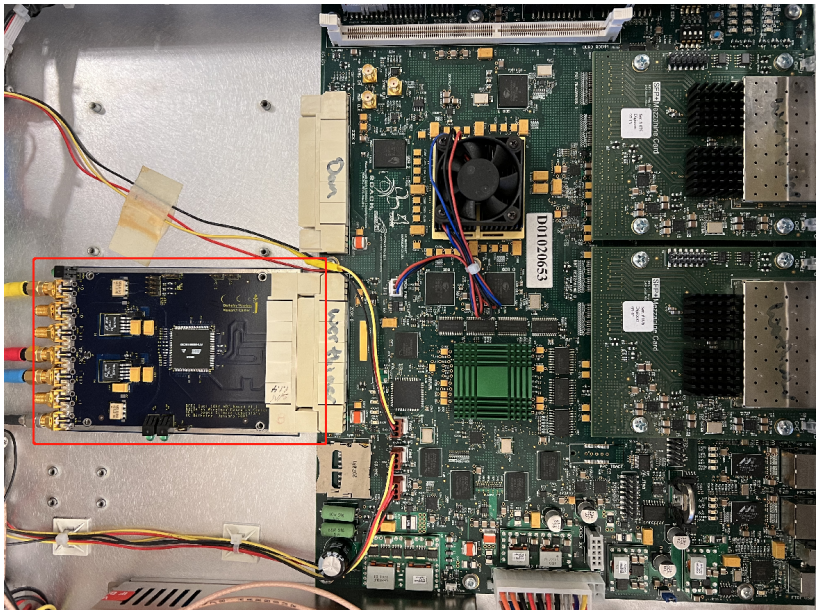

Figure 2: The ADC board connection

The ADC raw data for SERENDIP6 will be sent out from CH2 and CH3 on SLOT0, so please make sure the two fibers are connect to the two 10GbE ports.

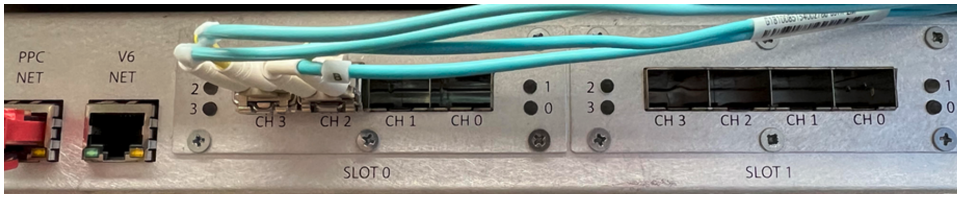

Figure 3: The 10GbE fiber connection

3 Required libraries

(1) Install HASHPIPE

Please follow the instructions here: https://casper.astro.berkeley.edu/wiki/HASHPIPE

(2) Other necessary libraries

```
sudo apt install libhiredis-dev
sudo apt install libcfitsio-dev
sudo apt install libmysqlclient-dev
sudo apt install csh
```

4 Compile and install SERENDIP6

(1) Clone the source code from github

```
git clone -b mro https://github.com/liuweiseu/serendip6.git
```

(2) Compile the code

```
cd serendip6/src
./do_make.csh mro
```

(3) Install SERENDIP6

```
sudo make install-mro
```

5 Configure ROACH2

We have some python scripts for the ROACH2 board configuration.

(1) Clone the python scripts from github

```
git clone -b mro https://github.com/liuweiseu/roach2_control_script.git
```

(2) Set up **Python2** environment

Follow the instructions in README.md to set up the Python2 environment. Python3 will not work here.

(3) Copy bof file to ROACH2

```
cd roach2_control_script
scp bof/italy_seti1_v1.172.bof user@roach2_ip:/boffile
```

Note: The bof file directory on ROACH2 board could be on somewhere else. It depends on what version of rootfs you are using for ROACH2 board.

(4) Upload the bof file to FPGA

```
(roach2_py2)roach2_control_script$ python mb.py --ip 10.0.1.168
Connecting to server 10.0.1.168 on port 7147...
ok
--------------------
Programming FPGA with italy_seti1_v1.172.bof ...
done
Configuring spectrometer "u0" fft_shift, fft_shift=0xFFFF ...
done
Configuring spectrometer "u0" scale coefficients, gain=0x1000100 ...
done
Configuring spectrometer "u0" bit selection, bit_select=0x55 ...
done
......
Initializing xgbe2 fabric mac: 02:02:C0:A8:10:DF,
ip: 192.168.16.223, port: 33333 ...
done
Configuring xgbe2 destination IP and port 239.1.0.4:12345 ...
done
Initializing xgbe3 fabric mac: 02:02:C0:A8:10:E0,
ip: 192.168.16.224, port: 33333 ...
done
```

```
Configuring xgbe3 destination IP and port 239.1.0.3:12345 ...
done
Issue reset signal...
done
```

Note: You cn use **"python mb.py –help"** to get more options.
The default values are:

- IP address of ROACH2: 192.168.100.128
- port: 7147
- bof file: italy_seti1_v1.172.bof

(5) Check the ROACH2 status**(Optional, not necessary)**

Please Use "ssh -XY xx@xxx" to log into the server, or you will not be able to see the GUI shown below.

```
(roach2_py2)roach2_control_script$ python mbc.py
```

You need to select the IP address of the ROACH2 board. If the IP doesn't show up in the IP list, you need to edit mbc.py, and add the correct IP in "roach_list"

```
roach_list = ['192.168.100.128', '10.0.1.168','localhost']
```

If everything goes well, you shoule see the following GUI.

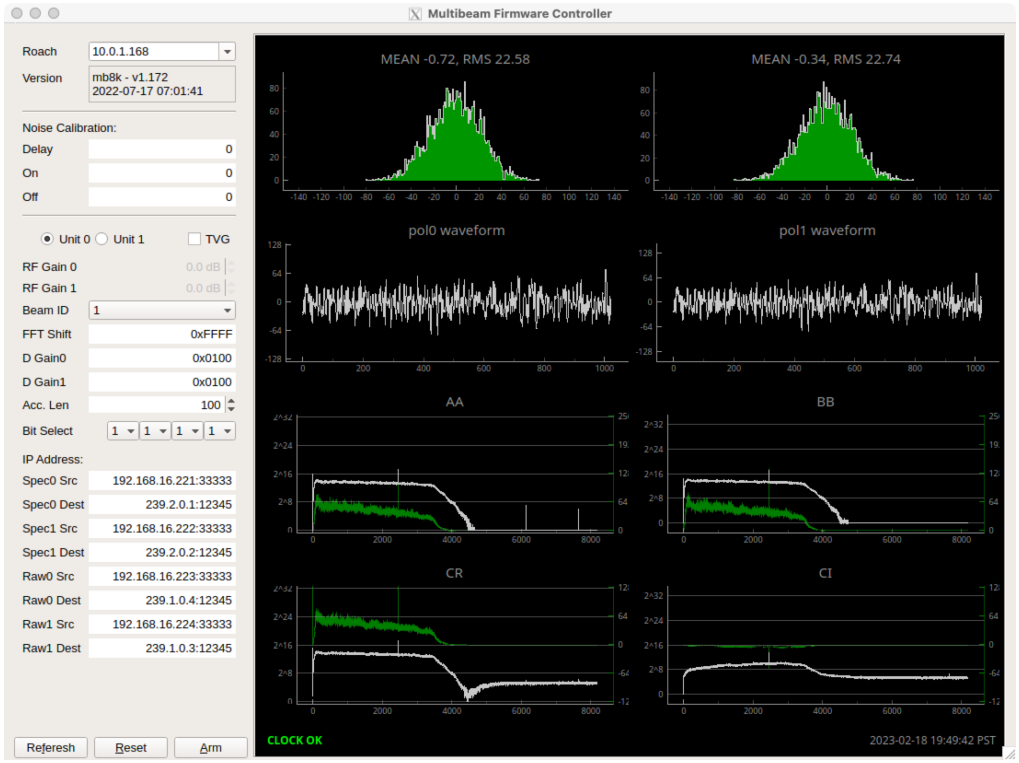

Figure 4: The GUI software for status check

6 Start/Stop SERENDIP6

(1) Set environment variables

You need to set three environment variables for SERENDIP6:

- S6_DATA_DIR: data directory for SERENDIP6
- POL0_ETH: the 10GbE port for pol0
- POL1_ETH: the 10GbE port for pol1

```
~$ export S6_DATA_DIR=/data01/serendip6_data
~$ export POL0_ETH=enp216s0f0
~$ export POL1_ETH=enp216s0f1
```

Note: "/data01/serendip6_data", "enp216s0f0" and "enp216s0f1" are is the default values. If you're happy with them, you don't need to set the environment variables.

(2) Set MTU for ethernet ports

The default mtu for ethernet port is 1500. You have to change the value to 9000 for both of the ethernet ports, or SERENDIP6 will not receive any packets.

```
~$ sudo ifconfig enp216s0f0 mtu 9000
~$ sudo ifconfig enp216s0f1 mtu 9000
```

Note: We suggest you have a script to do this automatically when the server boots up.

(3) Start SERENDIP6

Just run "s6_restart_mro.sh" to start SERENDIP6.

```
~$ s6_restart_mro.sh
Data Path: /data01/serendip6_data
removing old semaphore, if any
rm: cannot remove '/dev/shm/sem.serendip6_gpu_sem_device_*':
No such file or directory
Starting instance s6c/1
binding s6_pktsock_thread to enp216s0f1
numactl --physcpubind=16,17,18 --membind=0,1 hashpipe -p serendip6 -I
1 -o VERS6SW=0.8.0 -o VERS6GW=0.1.0 -o RUNALWYS=1 -o MAXHITS=2048 -o
Instance s6c/1 pid 646568
Starting instance s6c/2
binding s6_pktsock_thread to enp216s0f0
numactl --physcpubind=24,25,26 --membind=0,1 hashpipe -p serendip6 -I
2 -o VERS6SW=0.8.0 -o VERS6GW=0.1.0 -o RUNALWYS=1 -o MAXHITS=2048 -o
Instance s6c/2 pid 646569
Sleeping to let instances come up...
Resetting MISSEDPK count for s6c/1
Resetting NETDRPTL count for s6c/1
Resetting NETPKTTL count for s6c/1
Resetting MISSEDPK count for s6c/2
Resetting NETDRPTL count for s6c/2
Resetting NETPKTTL count for s6c/2
Turning on TESTMODE for /1
Turning on TESTMODE for /2
Turning on RUNALWYS for /1
Turning on RUNALWYS for /2
```

If SERENDIP6 starts up successfully, you will see two hashpipe instances:

```
~$ ps -x
240550 pts/15 Sl+ 0:14 hashpipe -p serendip6 -I 0 -o VERS6SW=0.8.0 ...
240551 pts/15 .l+ 0:14 hashpipe -p serendip6 -I 1 -o VERS6SW=0.8.0 ...
```

You will also see data files created in the data directory

(4) Stop SERENDIP6 Just run "s6_stop_mro.sh" to stop SERENDIP6.

```
~$ s6_stop_mro.sh
```

7 Check data file

Here is a simple python script for check SERENDIP6 data file. You can get it here:

```
git clone https://github.com/liuweiseu/seti_analysis.git
```

You need to change the file name in the .ipynb file:

```
filename = '../data/serendip6_panoseti_sxp_1_1_20230217_154019.working'
```

Then you will see the metadata like this:

```
XTENSION= 'BINTABLE'           / FITS binary table
BITPIX  =                    8 / Binary data
NAXIS   =                    2 / 2-dimensional binary table
NAXIS1  =                    0 / width of table in bytes
NAXIS2  =                    0 / Number of rows in table
PCOUNT  =                    0 / size of special data area
GCOUNT  =                    1 / one data group (required keyword)
TFIELDS =                    0 / Number of fields per row
EXTNAME = 'MROSTATUS'          / name of this binary table extension
NMATRIX =                    1 / 1 dataset per row
COARCHID=                    0 / coarse channel ID
TIME    = '*         '         / unix time, decimal seconds
SOURCE  = '0406-127'
ONSOURCE=                    1
SITE    = 'Mc        '
RX_CODE = 'sxp       '
YEAR_DOY= '#2023-026/174535'
YEAR    =                 2023
DOY_UTC =                   26
UTC     =               174535
LO_FREQ =               8080.
TSYS    =            40.900002
XC      =              -331.09
YC      =              -306.92
Z1C     =                -4.51
Z2C     =               -53.14
Z3C     =                66.82
XA      =              -331.09
YA      =              -306.92
Z1A     =                -7.7
Z2A     =               -48.77
Z3A     =                51.04
SUBMODE =                    0
RX_SUB  = 'sxp       '
SCU_STAT=                   63
EXTVER  =                    1 / auto assigned by template parser
```

You will also see a figure like this:

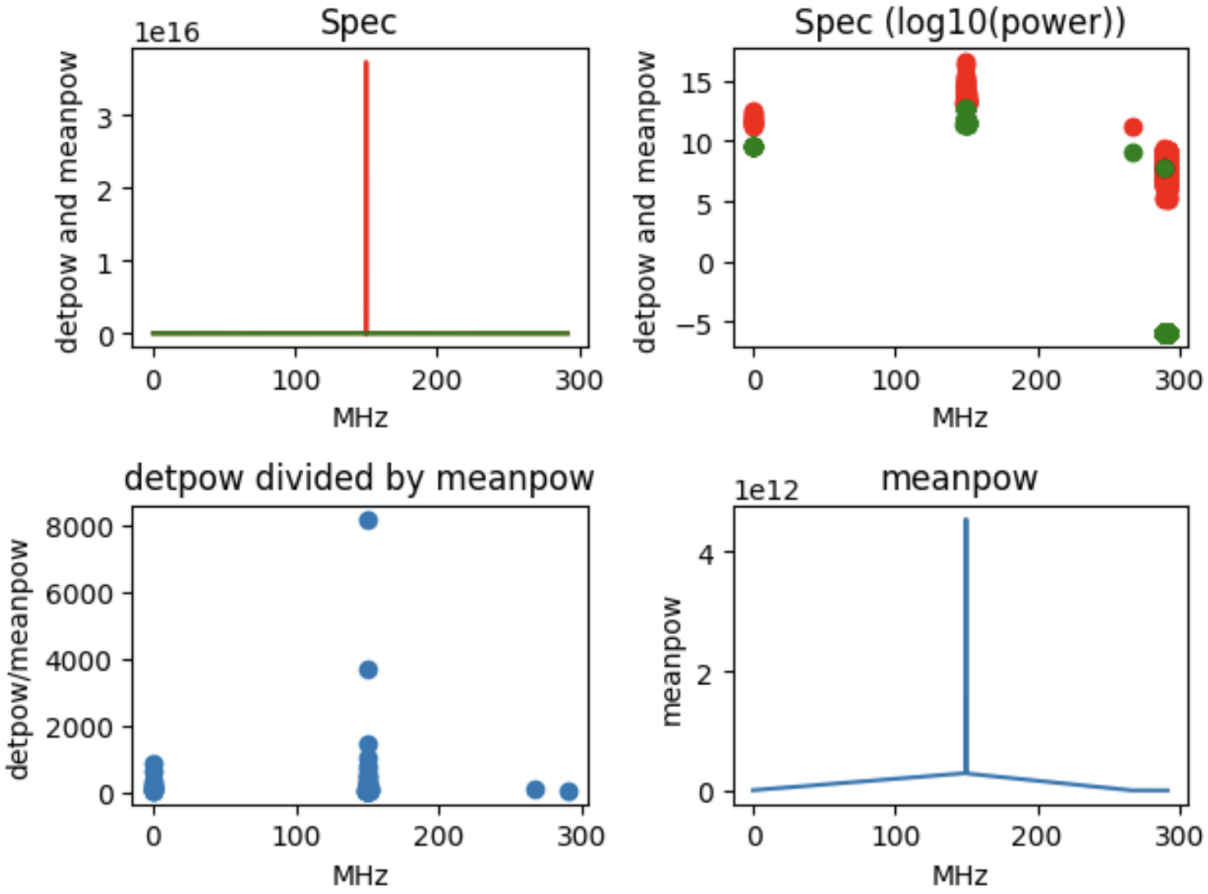

Figure 5: SERENDIP6 data result