Setting the RDBEGs at the 13.2 m Yebes VGOS antenna

P. de Vicente, L. Barbas, J. González

Informe Técnico IT-CDT 2016-21

Los desarrollos descritos en este informe técnico han sido cofinanciados por el Programa Operativo Fondo Tecnológico FEDER de la UE bajo el convenio IGN-MINECO denominado “Radiotelescopio de VLBI geodésico y astrométrico para su integración en la red VGOS (VGOSYEBES)”

UNIÓN EUROPEA
FONDO
EUROPEO DE
DESARROLLO
REGIONAL
"Una manera de hacer Europa"

Revision history

Version	Date	Author	Updates
1.0	19-10-2016	P. de Vicente	First draft
1.1	15-11-2016		Final version
1.1	15-12-2016	P. de Vicente	Updated sections 2.2 and 2.3

Contents

1 Introduction

RDBE stands for Roach Digital Backend. It is a backend mainly used in geodetic VLBI observations designed by Haystack and manufactured by DIGICOM. There have been several versions of which the Observatorio de Yebes has purchased two RDBE-G units, one in 2015 and one in 2016, to test and compare it with the DBBC3. The RDBE-G is the latest version comercially available at the time of this report and it uses a multipurpose ROACH board and mounts a Virtex 5 FPGA and PPC CPU on it. It can sample two signals 512 MHz bandwidth at 2 Gbps each. The maximum output rate is 4 Gbps. Although the device has connections for 2 more IFs no electronics are mounted inside to manage them because the ROACH board can't handle them. Each unit is 2U in height and can be easily mounted in a rack fastening it with four screws at the front.

VGOS stations using RDBEs usually require four units, since each of them would handle two polarizations from a single frequency band. These bands are tagged as a, b, c and d and these characters may be used in the FS and in the configuration files as we will see through the report. In a first stage in Yebes we use a hybrid system composed by one DBBC and 2 RDBEs, but this will change by the end of 2016 in which 4 RDBEGs will be used. The RDBE works with different firmware configurations: Polyphase Filter Bank (PFB) and Digital Down Conversion (DDC). The usual one for VGOS observations is PFB.

RDBE-G units are not fully VGOS compatible since they can't handle two bands 1 GHz bandwidth as required. A new version based in ROACH 2 is being developed at the time of the report (see McWirther 2016).

2 Setting up the RDBE

2.1 Cabling and hardware configuration

Figs. 1 and 2 show photographs of the back and of the front of the RDBE-G respectively and Table 2.1 summarizes the input and output connections from the RDBE-G. They are mainly located on the rear side, although there also some of them on the front of the device. The front has an LCD display with 4 lines:

- The first line shows the date (year, day of year and time) followed by DBE_DOT delta. For example: **2016 144 07:54:53 9**
- The second line shows VDIF time, which is composed of two fields: semester since 00.00 UTC on first of January 2000 and number of seconds from the reference epoch and information on the channel data characteristics: data rate in Gbps (2), if individual selection of channels is enabled (on) and if PSN (Packet Serial Number) is enabled in the VDIF header. Example of a line: **933:12383693 2:on:ps**
- The third line shows the total power for both IFs: signal RMS, if the bstate is ok and if transmission is on or off. RMS correct values range between 12 and 28. Finally the IP address of the device is shown. This information gets partially displayed and is complete

after refreshing the screen four times (4 seconds). For example one snpashot would show: **IF0: 2 !:! on 172**. The character “!” stands for OK.

- The fourth line shows the the delay of the station PPS with respect to the DOT clock (negative if the internal DOT clock goes after the station PPS) and the GPS pulse minus the DOT clock (positive if the GPS goes after the internal PPS). For example: **-1.56e-08 2.27e-05**.

Figure 1: *Front of the RDBE-G. When in normal operation the three red LEDS at the left (as we see the unit) blink simultaneously every second. The screen shows useful information about the status of the device and of the data*

Figure 2: *Photograph of the back of the RDBE-G. The signals of each connector are described in the text.*

The RDBE-G requires a reference signal which can be either 5 or 10 MHz. 5 MHz is the default signal, but at the Yebes backends room 10 MHz is the standard reference signal. To set it up the RDBE was opened and a jumper was placed on the synthesizer board, connection J1. See Fig. 3.

Before powering up the device the following signals should be connected:

- 5/10 MHz. A 10 MHz signal from a distributor at the Yebes backends room was plugged in.

Side	**Tag**	**Signal**
Front	GPS	1 PPS GPS output copy
Front	GPS	1 PPS station ouput copy
Front	GPS	1 PPS DOT output
Front	DIODE	Diode trigger output copy
Back	GPS	1 PPS GPS input
Back	PPS	1 PPS station input
Back	DIODE	Diode trigger output
Back	IF0	Horizontal (X) polarization IF input
Back	IF1	Vertical (Y) polarization IF input
Back	IF2	Not connected internally
Back	IF3	Not connected internally
Back	Ethernet	Ethernet connection
Back	USB	USB connection used to plug an USB stick with a Linux PPC architecture OS
Back	10GBE_0	CX4 VDIF output connector
Back	10GBE_1	CX4 VDIF output connector
Back	10GBE_2	CX4 VDIF output connector
Back	10GBE_3	CX4 VDIF output connector

Table 1: *Front and back input and output connections on the RDBE-G*

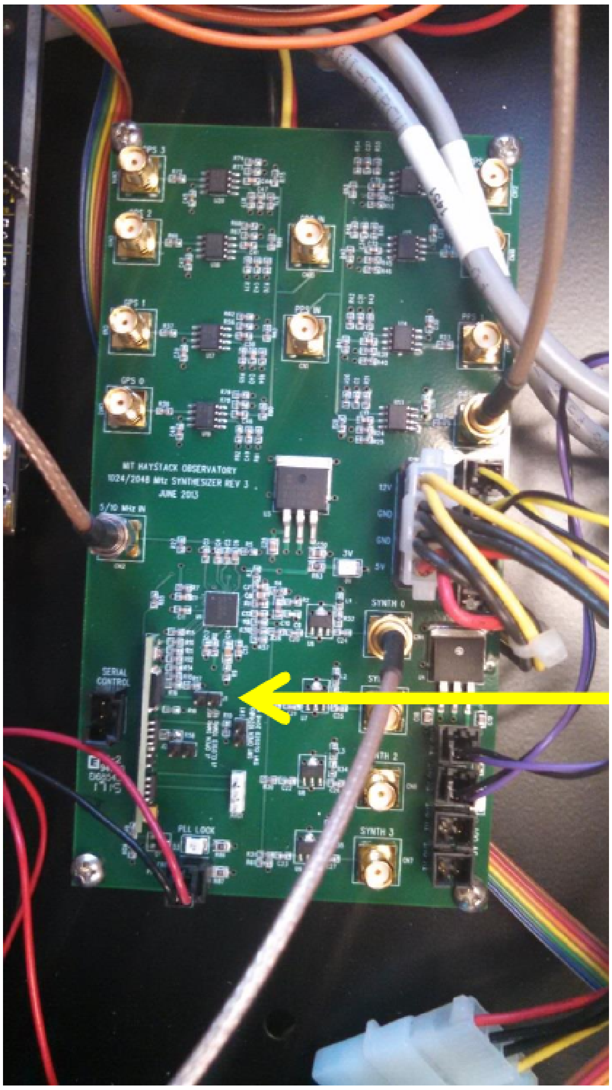

Figure 3: *Jumper to be placed in J1 (synthesizer board) to force the RDBE using a 10 MHz reference signal. If no jumper is placed 5 MHz is the default assumed reference signal*

- IF0. This should be a filtered signal between 500 and 1000 MHz from the Horizontal (X) polarization of the IF. Since the UDCs at Yebes deliver a signal between 0 and 1 GHz a 500-1000 MHz filter was plugged at the output of the UDC. Optimum level should be around -31 dBm (see section 3).
- IF1. This should be a filtered signal between 500 and 1000 MHz from the Vertical (Y) polarization of the IF. As in the previous case a 500-1000 MHz filter was plugged in the output of the UDC.
- GPS. One PPS signal from the GPS receiver should be injected in this connector. At the backends room of the 13m/40m radiotelescope there is a PPS home made distributor for the GPS signals.
- PPS. One PPS signal from the station should be injected in this connector. This signal is coming from a Timetech PPS distributor feeded by a timing box which generates the pulses from the 5 MHz maser reference. This distributor is used for all backends: DBBC2s, FFTs, continuum detector and counters.
- DIODE. This is an output to drive a switching diode with a square wave signal whose frequency can be set up between 10 and 100 Hz. The signal has a level of 2.25 volts (50 Ohm impedance).
- Ethernet. An RJ45 ethernet cable from the private VLAN was connected.
- USB. The OS of the RDBE-G is in a mini USB stick which is connected to the only available USB connector, between the XPORT and the Ethernet connectors. The stick tagged as "RDBE OS USB" together with another USB stick tagged as "RDBE DOCS" should have been received with the equipment. Usually they are in a plastic bag fastened on the upper side of the device.
- 10GBE_0. A CX4 cable should be connected to the Mark6 recorder.
- 10GBE_1. A CX4 cable should be connected to the Mark6 recorder.
- 10GBE_2. Not connected.
- 10GBE_3. Not connected.

Since the Mark6 recorder at Yebes uses optical fiber ports (SFP+ transceivers) and the RDBE-G uses CX4 connectors a converter is required. We are using two "Glappers" from HAT-Lab, which have a CX4 port and an optical port with one XFP transceiver. This device has LEDs on the front side which indicate the status of the device. Fig. 4 shows a photograph of the Glapper as seen from the back. To avoid a long CX4 cable the glapper was placed very close to the RDBE-G unit. Off the shelf commercial units, like D-LINK DMC-805X 10Gbps CX4 to 10Gbps SFP+ media converter, are also available.

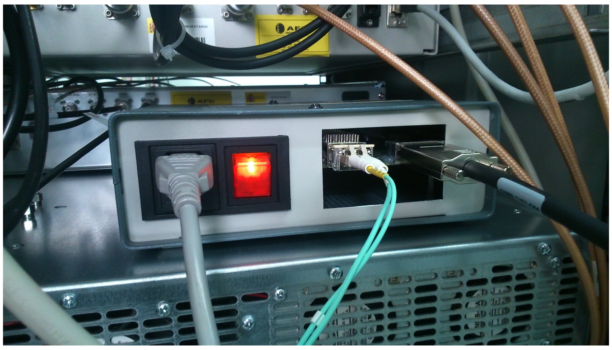

Figure 4: *Glapper stands for Glass to Copper and it is a unit designed and built by HAT-Lab. Currently there are commercial products that convert from CX4 to optical fiber.*

2.2 Operating System configuration

The RDBE boots directly from the OS in the USB stick and loads a Debian 4.0 version. Since no input/output ports for a keyboard, mouse or monitor are available the conenction should be done remotely using an ssh session. Previous to booting, the USB stick can be mounted in another PC and file `/etc/network/interfaces` modified to provide a fixed IP address, like for example:

```
auto eth0
iface eth0 inet static
address 172.16.20.30
netmask 255.240.0.0
network 172.16.0.0
broadcast 172.16.255.255
gateway 172.16.0.1
```

Once the RDBE boots any modification done in the system is permanent and gets saved in the USB stick. To log in, the root account should be used.

The applied modifications consisted in setting up the correct date and time using Linux command `date` since `hwclock` does not work properly, installing NTP and configuring the network:

```
date --utc --set "10/13/2016 14:13"
apt-get install ntp
```

The pool server IP address in the `/etc/ntp.conf` is set to:

```
peer 193.146.252.14 minpoll 4 maxpoll 8
```

We tested and fine tuned the time using `ntpdate` on a local NTP stratum at Yebes Observatory:

```
ntpdate -q 193.146.252.14
```

Two more operations should be performed, setting the name of the host which requires modifying `/etc/hostname`, for example:

```
rdbe-g1
```

and filling file `/etc/resolv.conf` with the name server:

```
domain oan.es
search oan.es
nameserver 193.146.252.16
```

The latter allows to connect to any host using its name and domain name.

2.3 Firmware and server configuration

The RDBE loads the firmware into the FPGA and starts a server that allows communication from the outside. The command set for the server is summarized in a Haystack Report by C. Ruszczyk and R. McWhirter. The curent version is 3.0.

The firmware, the RDBE server and the configuration are all stored in `/home/roach` directory. The PFB firmware is located at: `/home/roach/personalities/PFBG_3_0.bin` which is linked to:

```
PFBG_3_0.bin -> /home/roach/personalities/PFBG_3_0_3706.bin
```

The RDBE server is located at `/home/roach/bin/rdbe_server`. The version delivered with the unit by DIGICOM should be replaced by a new version, which at the time of the report is 5220. To manage different versions it is advisable to keep all of them and link to the latest one:

```
rdbe_server -> /home/roach/bin/rdbe_server_5220
```

The RDBE server should be started by hand:

```
rbin
nohup ./rdbe_server 5000 6 &
```

`rbin` is an alias for a command that changes to directory `/home/roach/bin`. `nohup` keeps the execution even if we log out of the session.

The RDBE server can be tested with a small test script installed at the FS or at the same RDBE unit:

```
./RDBE_test.py
server : 193.146.252.20  srv port:  5000
FPGA personality load test with PFBG_3_0.bin     - start
FPGA load personality                                   - PASSED

Synthesizer 1pps test                            - start
Synthesizer 1pps time incrementing properly      - PASSED
```

Both the latest stable firmware and server are available at the same FTP page at Haystack: `ftp://gemini.haystack.mit.edu/pub/car/RDBE/`.

The configuration is located in directory `/home/roach/personalities/` and the configuration parameters used by the server in file `conf/PFBG_3_0.conf`. This file is linked to `pfbg_3_0_c.conf` in that same directory. `c` character stands for RDBE-G at the third band, c. The content for RDBE-G at the third band (c) is as follows:

```
ifconfig=up:9000:4:192.168.1.102:1
mac=00.00.11.22.33.42
chsel_en=2: chsel_enable : psn_enable
#chsel=0:1:2:4:6:9:13:14:15
#chsel=1:1:2:4:6:9:13:14:15
arp=192.168.1.4:00.60.dd.43.8e.8e
data_connect=192.168.1.4:9002:0xBdB:0
pps_mon=disable
pps_mon=enable:239.0.2.30:20023
pcal=1.4e6
diode=80
data_send=on
option=time_long
```

- The first line configures the 10G interface : brings the interface up : sets MTU size : Layer 4 transport : IP address of port, and port ID.
- The second line sets the MAC address of the 10 G port.
- The third line sets up the packet information = 2Gbps : allowed to select channels , instead of all : enables Packet Serial Number (VTP) headers
- The fourth line sets the destination IP (Mark6 10G IP) : and its MAC Address
- The fifth line sets what is used to establish the connection, that is destination IP (Mark6 port IP), port number to use in VTP header : ASCII identifier : Thread ID to use.
- The `pps_mon` lines first disable and later enable the monitor using multicast on IP address 239.0.2.30 and port 20023. There is a convention here to use: 239.0.2.10 for RDBE-1, 239.0.2.20 for RDBE-2, 239.0.2.30 for RDBE-3 and 239.0.2.40 for RDBE-4. Ports also change depending on the RDBE unit: 20021, 20022, 20023 and 20024 respectively. According to the "RDBE Command Set (version 3)", enabling `pps_mon` activates multicast for Tsys, raw data, phase cal and other information. See section 5 for further information on multicast at the RDBE.
- The `pcal` line sets the tone frequency to be analyzed in Hz. It ranges from 0 to 5 MHz. Since the LO is usually set to a multiple of 0.4 MHz, the phase cal tone is also a multiple of 0.4 MHz.
- The `diode` line states the frequency rate at which the continuous cal will work. Allowed values are 0, and 10 to 100 Hz.
- The `data_send` line states that data should be sent through the 10 G interface.
- The `option` line will set long time formats for the return responses.

All the previous lines are associated to RDBE commands which can be issued manually. Being in the configuration file, the commands get executed when the server starts for the first time.

The IP and MAC addresses are private and arbitrary since the RDBE 10G interfaces are directly connected to the Mark6 and there is no switch in between.

Update: Although the assumed configuration file is `conf/PFBG_3_0.conf`, it seems that running command `dbe_runfile` may have a buggy behaviour since it does not use the specified configuration file. We have tested this behaviour using the `rdbe.prc` procedure library at the FS which has several RDBE configuration functions like, `rdbe_init`. `rdbe_init` contains

```
rdbe_inita
rdbe_initb
rdbe_initc
rdbe_initd
```

In turn each of the procedure functions contain commands like, for example (`rdbe_inita`):

```
rdbea=dbe_runfile=/home/roach/personalities/conf/pfbg_3_0_a.conf;
```

If we remove file `conf/pfbg_3_0.conf`, command `rdbe_inita` will fail, whereas if file `conf/pfbg_3_0.conf` is created, the command works correctly. Therefore in order to have a correct configuration we have copied `conf/PFBG_3_0.conf` into `conf/pfbg_3_0.conf` and now we keep three identical configuration files in the configuration directory.

In order to get an automatic load of the firmware and the execution of the server after a powerup, file `/etc/rc.local` contains the following lines:

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

ifdown eth0
sleep 2
ifup eth0

/home/roach/./install_rdbe_dev.sh

sleep 2
#  sync to Yebes GPS
ntpdate -u -p 8  193.146.252.14

sleep 2

/home/roach/./cleanup.sh

/home/roach/bin/./rdbe_server 5000 6

exit 0
```

3 Power detection: linearity range

We have tested the linearity of the RDBE continuum power detection injecting white noise from a laboratory noise source whose power we can control from variable attenuators. The noise was injected into IF0 of one RDBE and had a bandwidth of one 1 GHz. The signal was filtered before entering the IF using a filter 500-1000 MHz. The results are summarized in Table 3. Fig **??** also summarizes the results so that they can be seen at a glance.

According to Figs. 5 and 6 the detection is linear in the range between -45 dBm and -24 dBm in 500 MHz which corresponds to 5 RMS units and 34 RMS units respectively. The recommended range for VLBI observations is, according to Haystack, between 12 and 28 RMS units which correspond to an input power between -36 to -28 dBm in a 500 MHz band, equivalent to a Spectral Power density of -63 to -5r54 dBm/Hz.

Att. [dB]	SPD [dB/MHz]	Power in 500 MHz [dB]	RMS (RDBEG)
13	-46.0	-19.0	73
14	-47.0	-20.0	68
15	-48.0	-21.0	62
16	-49.0	-22.0	54
17	-50.0	-23.0	51
18	-51.0	-24.0	46
19	-52.0	-25.0	41
20	-53.0	-26.0	37
21	-54.0	-27.0	30
22	-55.0	-28.0	29
23	-56.0	-29.0	25
24	-57.0	-30.0	23
25	-58.0	-31.0	21
26	-59.0	-32.0	18
27	-60.0	-33.0	16
28	-61.0	-34.0	14
29	-62.0	-35.0	13
30	-63.0	-36.0	12

Table 2: *Attenuation from the noise source, Input Spectral Power Density (SPD), power in 500 MHz bandwidth and power detected by the RDBE in "RMS units". According to Haystack Observatory the optimum level is 20 and the range between 12 and 28 is acceptable*

4 Using the RDBEs from the FS

RDBE backends are supported by the FS since version 9.12. Yebes uses, at the time of the report, a hybrid system composed of 2 RDBEs and one DBBC2 and this special setup required modifications which are available since version 9.12.8. Several control files need to be set up to

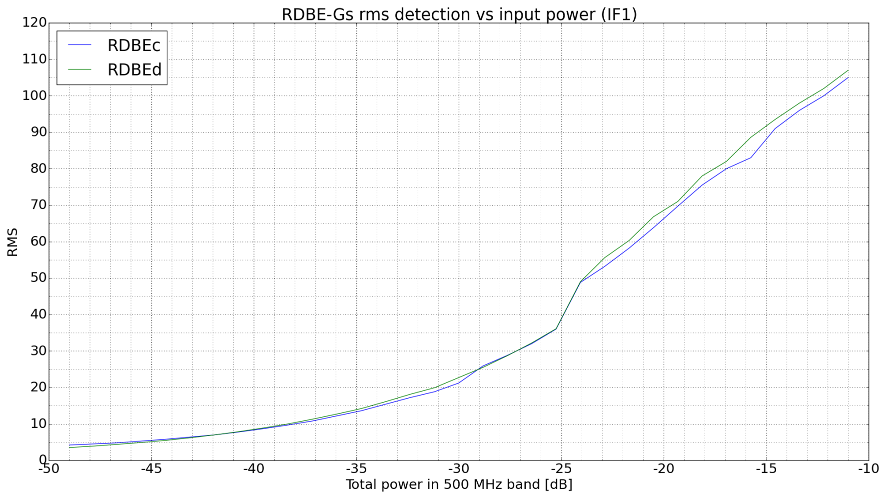

Figure 5: *Detected power at the RDBE (RMS of the signal) versus total input power in 500 MHz in dBm.*

Figure 6: *Logarithm of the detected power at the RDBE (RMS of the signal) versus total input power in 500 MHz in dBm.*

be able to remotely command the RDBEs from the FS. Files `/usr2/control/rdbeX.ctl`, where X is a, b, c or d, contain the IP address, port and time out for the RDBE server. Setting these files should suffice to command the RDBEs. However for a full 4 RDBE system, file `equip.ctl` should also be set to `rdbe` allowing to perform onoff observations.

Commands for the RDBEs should be preceded by "rdbe=", which commands all units at the same time. In case we want to address an individual RDBE commands should start by `rdebX=`, where X can be a, b, c or d. Below we show an example with a list of commands:

```
rdbe=sw_version?
rdbe=dbe_chsel_en=2:chsel_enable:psn_enable;
rdbe=dbe_chsel=0:1:2:4:6:9:13:14:15;
rdbe=dbe_chsel=1:1:2:4:6:9:13:14:15;
rdbe=pcal=1.4e6;
```

The first command will return the following answer from both RDBEs:

```
2016.307.14:57:38.20/rdbed/!dbe_sw_version?0:rdbe_server 3.0 SVN 5202:Linux 2.6.18-128.1.1.el5;
2016.307.14:57:38.20/rdbec/!dbe_sw_version?0:rdbe_server 3.0 SVN 5202:Linux 2.6.18-128.1.1.el5;
```

The FS provides library `rdbe.prc` to perform several operations in single procedures. For example `rdbe_fpga` contains a list of commands for each of the RDBEs which tells the unit to load the firmware from the FPGA. Usually this library is not required for usual operations.

4.1 Measuring the IF power

Command `rdbe_attenX` is used to set the attenuation and monitor the level of the detected signal per RDBE unit. Unlike other commands it allows to set the attenuation to the different units. Typical values for the RMS of the signal are between 12 and 28 and they are set at control file `/usr2/control/rdbe.ctl`, together with the recommended target level which is 20, and a fourth parameter that determines the units of the RMS, wether they are to be raw (10^{-7}), normalized (divided by $2.5\,10^{-5}$) or in correlator units. Setting the attenuation is done, for example for band c, like this:

```
rdbe_attenc=0,auto,0
rdbe_attenc=1,auto,0
```

The target level, when not specified, is read from `/usr2/control/rdbe.ctl`. When specified as third parameter the attenuation changes to achieve the target value. In the current FS version (9.12.8) automatic setting when specifying the third parameter does not work correctly, but it does if not specified. It is also possible to fix the attenuation value to a given value if the second parameter is a value between 0 and 31.5 in steps of 0.5 dB.

Monitoring is done issuing command:

```
rdbe_atten
```

which will output the attenuation and RMS of the raw detected signal for all RDBE units. The RMS of the raw signal can also be polled issuing:

```
rdbe=dbe_raw?0:0
rdeb=dbe_raw?1:1
```

4.2 Doing ONOFFs

In order to perform ONOFF observations with RDBEs the system should be set following these steps:

- Set the rack type to RDBE in `equip.ctl`
- Set the correct `.rxg` files for the required Local Oscillators (LOs).
- Set the LOs with `LO=` commands. The names for the RDBE are of the form loBI, where B is the band (a,b,c,d), and I is the IF (0,1). For example:

  ```
  lo=
  lo=loa0,2472.4,usb,lcp,5
  lo=loa1,2472.4,usb,rcp,5
  lo=lob0,4712.4,usb,lcp,5
  lo=lob1,4712.4,usb,rcp,5
  lo=loc0,5832.4,usb,lcp,5
  lo=loc1,5832.4,usb,rcp,5
  lo=lod0,9672.4,usb,lcp,5
  lo=lod1,9672.4,usb,rcp,5
  ```

  It is possible to only declare those LOs which are installed. In our hybrid system that would be, for example:

  ```
  lo=
  lo=loc0,5832.4,usb,lcp,5
  lo=loc1,5832.4,usb,rcp,5
  lo=lod0,9672.4,usb,lcp,5
  lo=lod1,9672.4,usb,rcp,5
  ```

- The `onoff=` command sets the detectors to be used which are of the form NNBI, where NN is the channel number (00-15) and B and I are as before. An example of usage where we detect channels 9 and 14 for both polarizations:

  ```
  onoff=2,1,75,3,,120,09c0,09c1,09d0,09d1,14c0,14c1,14d0,14d1
  ```

- Command `onoff` without parameters will perform the ONOFF measurement.

5 Multicast

RDBEs use multicast to broadcast status information and data to other computers. It is a very convenient method that does not overload the network and eases spreading the information across different hosts. Multicast is much less network intrusive than broadcast because it only sends the information to those hosts where multicast is enabled at their OS and that have a client listening in the appropiate port. The server knows where to send the information because it first polls the network looking for listeners.

According to the DBE command set, RDBEs can multicast several information when issuing command:

```
dbe_1pps_mon = enable
```

The dot clock comparison is multicasted, together with Tsys, raw data, phase cal amplitude and phase and other information. Application `rdbe30_mon_2994.py` is an example of a multicast client written in Python which depicts the information obtained in real time from an RDBE. `monit6`, another application running on the FS computer, also takes advantage of multicast to show the signal power for both IFs, the Tsys, and the phase cal amplitude and phase of a selected tone.

Multicast on a Linux computer is active if when querying information about the ethernet interface one gets a line with "UP BROADCAST RUNNING MULTICAST":

```
/sbin/ifconfig
eth0      Link encap:Ethernet  HWaddr 78:e7:d1:85:d4:76
          inet addr:172.16.0.34  Bcast:172.31.255.255  Mask:255.240.0.0
          inet6 addr: fe80::7ae7:d1ff:fe85:d476/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
  .....
```

One can also test it by issuing:

```
ip maddr show

1:      lo
        inet  224.0.0.1
        inet6 ff02::1
        inet6 ff01::1
2:      eth1
        link  33:33:00:00:00:01
        inet6 ff02::1
        inet6 ff01::1
3:      eth0
        link  33:33:00:00:00:01
        link  01:00:5e:00:00:01
        link  33:33:ff:be:c6:d0
        link  33:33:00:00:02:02
        link  33:33:00:00:00:fb
        link  01:00:5e:00:00:fb
        link  01:00:5e:00:02:1e
        link  01:00:5e:00:02:28
        inet  239.0.2.40
        inet  239.0.2.30
        inet  224.0.0.251
        inet  224.0.0.1
        inet6 ff02::fb
        inet6 ff02::202
        inet6 ff02::1:ffbe:c6d0
        inet6 ff02::1
        inet6 ff01::1
```

In the previous case only eth0 has a client working in multicast. The IP V4 addresses the host is listening at are listed: 239.0.2.40 (RDBE-G4), 239.0.2.30 (RDBE-G3) and generic onces 224.0.0.251 and 224.0.0.1. Multicast, as already mentioned in a previous section, is enabled within the RDBE server configuration file (for example):

```
pps_mon = enable:239.0.2.30:20023
```

Multicast does not work properly in a host, for example the FS computer, with two interfaces in different networks: public and private. We have tested two different setups with the FS computer having two IP addresses and being the gateway in the public network: if the RDBEs were in the public network the multicast worked fine. If the RDBEs were set in the private network the multicast did not work. To make the latter option work we disabled the public IP address at the FS computer. This is the current setup.

5.1 Monit6

Monit6 is a monitor application from the FS that displays information on the station clock compared to the GPS, the system temperature for each IF and the amplitude and phase for a given tone. Fig. 7 shows a snapshot of the application:

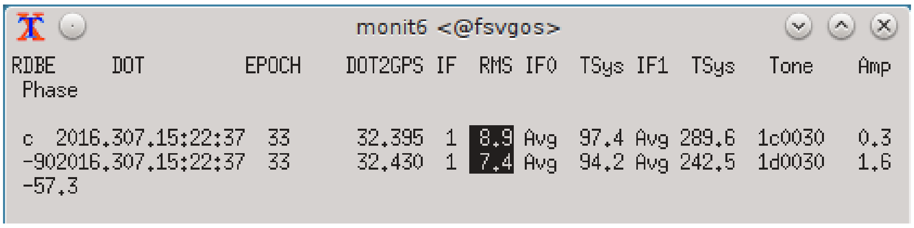

Figure 7: *Snapshot of monit6 with two working RDBEs.*

FS 9.12.8 version had to be fixed to make monit6 work for a hybrid system like the Yebes one. Monit6 can be tuned using configuration file `/usr2/control/monit6.ctl`. This file allows to determine the system temperature of each IF from three options: averaging all channels, summing them, or selecting an individual one. The phase cal tone for each IF is also selected from this file. There is one line per RDBE as we show below:

```
*RDBE-A:
 avg avg 30 30
```

In the example above Tsys for IF0 and IF1 is obtained by averaging the channels selected with command `rdbe=dbe_chsel=`. The tone selected is that one at 30 MHz from the start of the IF band (530 MHz). Tsys is computed using the LO2 values and the `rxg_file` which contains the Tcal as function of frequency:

$$Tsys = \frac{V_{on} + V_{off}}{2(V_{on} - V_{off})} T_{cal} \tag{1}$$

5.2 rdbe30_mon.py

This Python client developed in Haystack by Chester Ruszczyk, displays all information multicasted by one RDBE using several graphical windows. Fig. 8 shows how it looks like. This application opens 4 windows: two of them contain text with information, a third can be used to send commands and the fourth one displays graphs with information.

The graphical window shows, from top to bottom, raw data in the band, spectrum in a band of 512 MHz, histogram, phase cal pulses as a function of time, phase cal spectrum, phase cal phase, and uncalibrated Tsys as a function of time. Uncalibrated Tsys means that the plotted values should be multiplied by Tcal to get meaningful values.

5.3 Using the RDBE as a continuum detector

We (F. Beltran) have developed code to use the RDBE as a continuum detector taking advantage that multicast delivers the uncalibrated Tsys. We are using the sum of the ON and OFF cycles

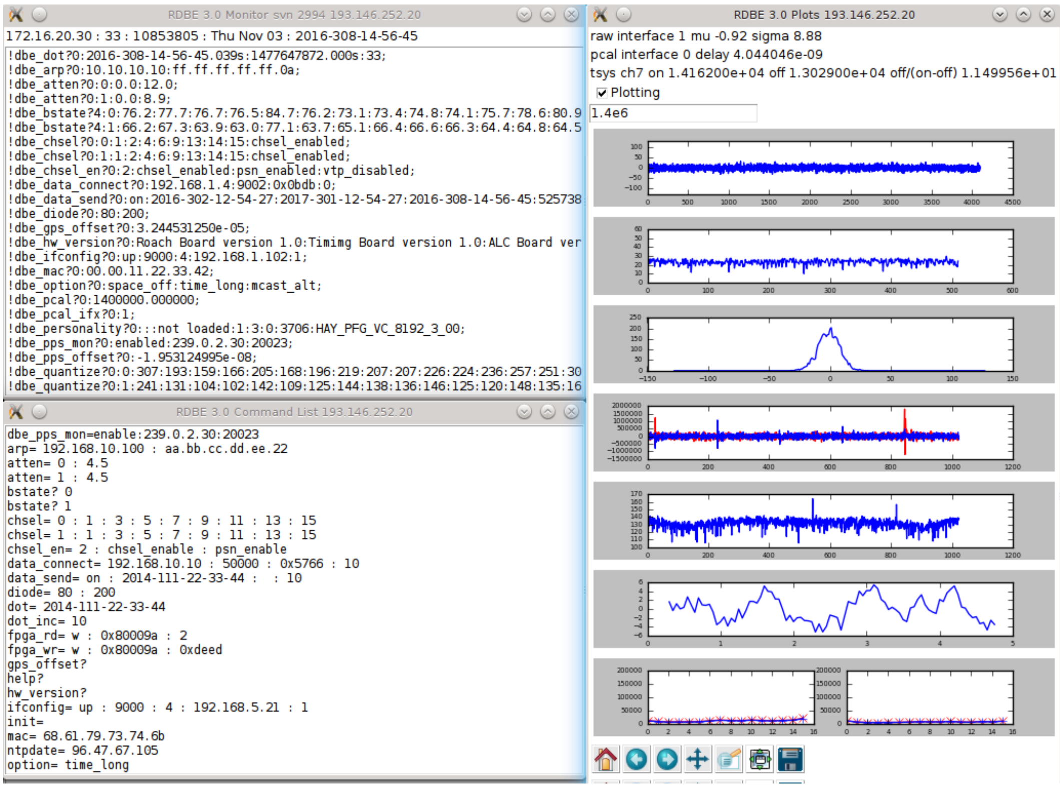

Figure 8: *Snapshot of rdbe30_mon.py*

of the 80 Hz and summing up all defined channels to get an integrated value every 1 second. The code is a python client that listens to the Multicast and delivers the total integrated value in a notification channel. Further information on the development and usage of the RDBE for this purpose will be presented in a separate report.

6 Final setup at Yebes

It is foreseen that before the end of 2016, the 13.2m telescope will be equipped with 4 RDBE units. Fig. 9 depicts the schematics of the connections. Each RDBE is connected to one UDC and receives two polarizations from one band. Since all RDBE outputs use CX4 connections and our Mark6 uses optical SFP+ transceivers a converter is required for each connection. Two of them are provided by "Glappers" built and delivered by HATLab. The other two are commercial units from D-Link, model DMC-805X.

References

[1] R. McWirther. 5th International VLBI Workshop. Hasyatck Observatory 2016. Progress towards commodity signal processing for next-gen DBEs.

[2] C. Ruszczyk. TOW 2015

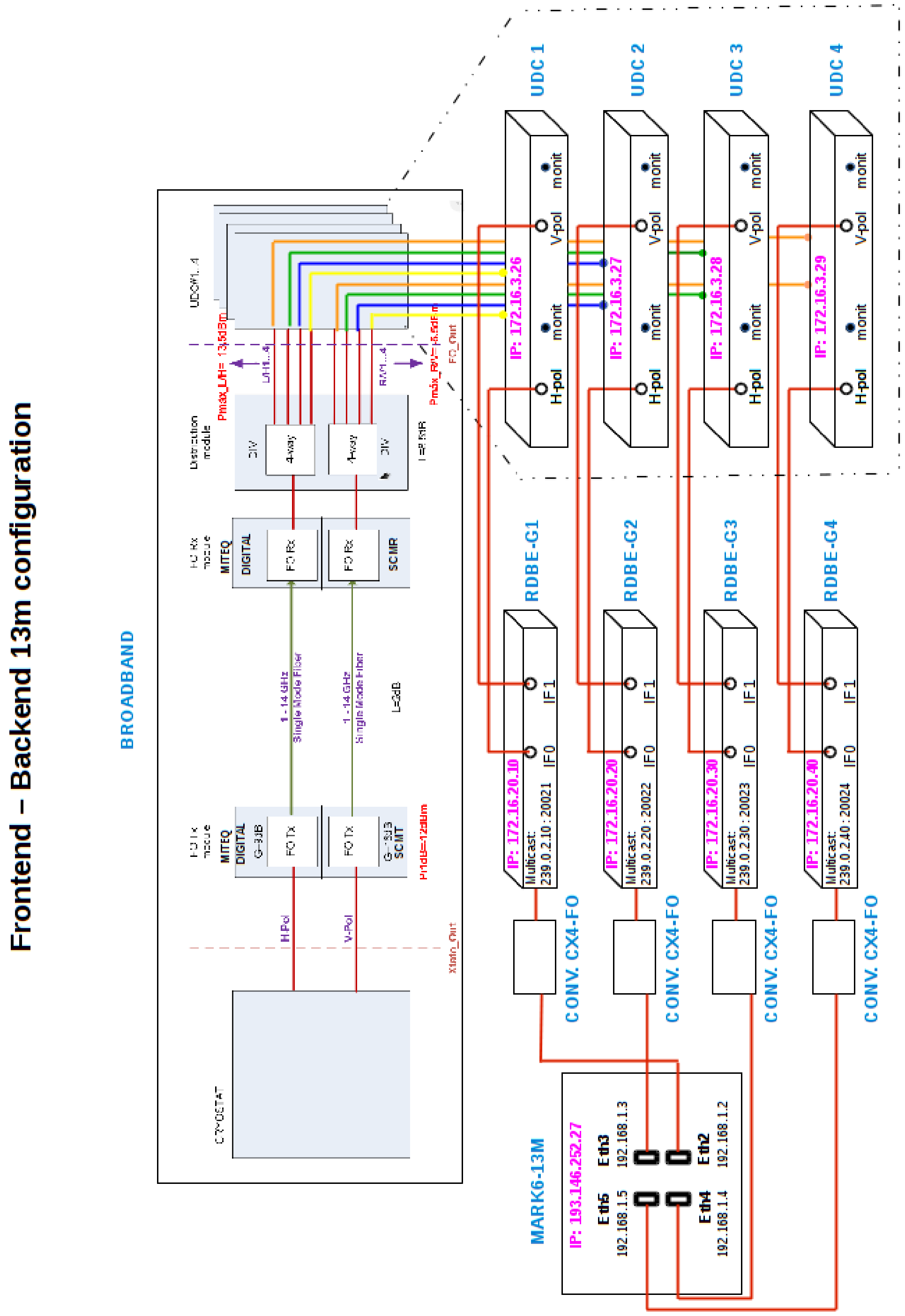

Figure 9: Backend setup with UDCs and RDBEs for the 13.2m

[3] C. Ruszczyk, R. McWhirter. 3 July 2015. DBE Command Set V3.0