

# 8-Antenna 8-bit / 4-bit GPU based CORRELATOR TESTING PROCEDURE

Version 3 , Released Date : 01/11/2013.

Written By : Sanjay S. Kudale / S. Harshavardhan Reddy/ I. M. Halagali.

## 1. HARDWARE SETUP

## 2. TERMINALS SETUP

## 3. PROCEDURE

### a. EDIT THE FILES

### b. CONFIGURE THE GPU

### c. ONLINE

## 4. ANALYSING THE RESULT USING TAX PROGRAM

## 5. POWER ON-OFF PROCEDURE

### 1. HARDWARE SETUP :

[CTRL\_PC] : 192.168.4.68 (hostname : ctrlpoco)

[USER\_CTRL\_PC] : gmrt

[CTRL\_PC\_PASS] : gmrttifr

### CLUSTER II

[GPU\_CTRL\_PC] : 192.168.11.15 (hostname : node51)

# This is one of the PC in "cluster II. This m/c can be accessed from gsbm3"

[USER\_GPU\_CTRL\_PC] : gpuuser

[GPU\_CTRL\_PC\_PASS] : gmrt123

[ROACH0\_IP] : 192.168.100.19

[ROACH1\_IP] : 192.168.100.71

### NOTE :

**USING BROADBAND SIGNAL FOR GPU : BEFORE STARTING H/W SETUP, MAKE SURE RF POWER AT ADC INPUT BETWEEN -14dbm to -17dbm, Or equalize to 100-150 counts using GWB power equalization.**

In this setup, two ROACH boards each with two iADC cards are used. Each iADC card has a clock, sync and two inputs for signal . While the clock input is a must for all iADC cards, sync inputs are needed only for iADC 0s of the two ROACH boards. Each GPU PC (a total of 4 GPU PCs) is equipped with two 10 GbE cards apart from the GPU card. One 10 GbE card is directly connected to one of the two ROACH boards while the other 10GbE card is connected to neighbouring GPU PCs via a 10 GBE switch.

The following connections needs to be done for antenna test.

1. Connect the ROACH boards to the controlling PC using 100MBPS ethernet.
2. Feed 400MHz, 0dbm clock signal to the clk\_i inputs of the iADC cards from the signal generator. Also feed the PPS signal to the Sync input of the iADC 0 cards of both ROACH boards.
3. Connect the antenna signals to the iADCs and make note of the sequence as it is needed for “fstc” and “fringe” corrections. The sequence is as follows:

ROACH 0 iADC 0 ( <u>I+ &amp; Q+</u> )	-----	Antennas 0 and 1
ROACH 0 iADC 1 ( <u>I+ &amp; Q+</u> )	-----	Antennas 2 and 3
ROACH 1 iADC 0 ( <u>I+ &amp; Q+</u> )	-----	Antennas 4 and 5
ROACH 1 iADC 1 ( <u>I+ &amp; Q+</u> )	-----	Antennas 6 and 7

The integer delay correction is different from fringe and fstc corrections. For fstc and fringe corrections, every GPU node corrects corresponding values of all antennas for different time slices. While for integer delay corrections, every node corrects corresponding values of only two antennas. The mapping for integer delay correction is as follows.

Node 0	-----	Antennas 0 and 1
Node 1	-----	Antennas 2 and 3
Node 2	-----	Antennas 4 and 5
Node 3	-----	Antennas 6 and 7

4. Connect the ROACH boards to the GPU PCs using 10GbE cables. There are four 10 GbE connectors on each ROACH board. The design has been made to allot one 10 GbE connector per iADC board, which means only two 10 GbE connectors are used. Also, each 10 GbE connector sends out data at the rate of 800 MiB/s.

The mapping of 10 GbE connections is as follows:

10 GbE port on ROACH	GPU PC
ROACH 0 PORT 0	----- Node 0
ROACH 0 PORT 1	----- Node 1
ROACH 1 PORT 0	----- Node 2
ROACH 1 PORT 1	----- Node 3

5. Turn ON the ROACH boards.

## **2. TERMINALS SETUP :**

**Open the terminals and set into the directories :**

-----  
**CONTROL TERMINAL (192.168.4.68) :**

```
ssh -X gmrt@ctrlpoco  
cd harsha
```

=====

**GPU TERMINAL (192.168.11.15) :**

```
ssh -X gpuuser@node51  
cd ~/delay_cal/
```

=====

**SOCKCMD TERMINAL (192.168.11.15):** (GPU CORR- sockcmd, connection to ONLINE)

```
ssh -X gpuuser@node51  
cd ~/bin/released/
```

=====

**COLLECT TERMINAL (192.168.11.15) :** (collect, acq-record connection)

```
ssh -X gpuuser@node51  
cd ~/bin/released/
```

=====

**DASSRV TERMINAL :** (dassrv, communication between ONLINE-GPU/GSB CORR)

```
ssh -X observer@lenyadri/shivneri  
( lenyadri or shivneri depend on whether GPU corr  
is running from standby ONLINE or default ONLINE along with GSB)  
cd /odisk/online1/gsbe/dassrv-gpu/
```

=====

**RECORD TERMINAL (192.168.11.15) :**

```
ssh -X gpuuser@node51  
cd /data1/gpuuser/
```

=====

**ONLINE USER TERMINALS :**

```
ssh -X observer@lenyadri/shivneri  
It is assumed, ONLINE is running, master, user0, user4  
are started and running in terminals MASTER/USER0/USER4 respectively.
```

=====

### **3. PROCEDURE :**

#### **a. EDIT THE FILES :**

=====

**GPU TERMINAL (ssh -X gpuuser@node51) :**

ONLINE m/c:  
(lenyadri or shivneri depend on whether GPU corr  
is running from standby ONLINE or default ONLINE)

```
shivneri 192.168.1.12  
lenyadri 192.168.1.13
```

For change ONLINE m/c to other, edit host file /home/gpuuser/bin/released/SYS\_FILES/hosts.dat,  
comment/uncomment relevant m/c.

=====

**GPU TERMINAL (ssh -X gpuuser@node51) :**

edit /home/gpuuser/bin/released/SYS\_FILES/sampler.hdr for antenna connections.  
For 4 node cluster, keep no. of antennas = 8  
For 2 node cluster, keep no. of antennas = 4  
Follow the antenna connection seq. in sampler.hdr  
file as below. (4 antennas per ROACH board)

ANTE1-ROACH0  
ANTE2-ROACH0  
ANTE3-ROACH0  
ANTE4-ROACH0  
ANTE1-ROACH1  
ANTE2-ROACH1  
ANTE3-ROACH1  
ANTE4-ROACH1

-----  
**GPU TERMINAL (ssh -X gpuuser@node51) :**

edit /home/gpuuser/bin/released/gpu.hdr for required setup  
GSB\_LTA=1 is equal to 671088.64 micro sec.  
GSB\_LTA=4 is equal to 2684354.56 micro sec= ~2.6 sec  
GSB\_ACQ\_BW is acquisition bandwidth (generally 200 MHz)  
GSB\_CHAN\_MAX= 2048 for 2k channels,  
GSB\_CHAN\_NUM= 0:2047:1 for its range.  
GSB\_STOKES = 1, currently only '1' is possible.  
currently, GSB\_RF, GSB\_SIDE BAND & GSB\_LO-1 are not  
implemented, so ignore.

=====  
**b. CONFIGURE THE GPU :**

=====  
**CONTROL TERMINAL (ssh -X gmrt@ctrlpoco) :**

./dual\_adc\_pps\_8bit\_cluster\_das2.py

=====  
**GPU TERMINAL :**

./gpu\_corr\_released.sh

=====  
**c. ONLINE :**

**Warning : Pl. ensure that sampler.hdr & gpu.hdr files are updated.**

=====  
**SOCKCMD TERMINAL :**

./sockcmd.sh

=====  
**COLLECT TERMINAL :**

./collect

=====  
**DASSRV TERMINAL :**

./dassrv\_released

**USER0 TERMINAL :**

cp 9;allant;cmode 1;tpa(11)=15;initndas '/temp2/data/gsb.hdr' # use “cmode 1” for GPU ONLY.

OR

cp 9;cmode 9;tpa(11)=15;initndas '/temp2/data/gsb.hdr' # use “cmode 9” for BOTH GSB & GPU.

OR

cp 9;cmode 8;tpa(11)=15;initndas '/temp2/data/gsb.hdr' # use “cmode 8” for GSB ONLY.

=====

**CONTROL TERMINAL (ssh -X gmrt@ctrlpoco) :**

./set\_cluster\_das2.py

=====

**USER0 TERMINAL :**

**NOTE : Wait till minute boundry appears in GSB acq.**

ante 8 5 3 9 10 29 30 8 25 #edit the antennas used and makesure sampler.hdr has same set of ant.

cp 0;defs 4;suba 4

cp 9;suba 4;prjtit'GPU';prjobs'GPU';initprj(15,'PRJCODE')

# use “initprj(15,'PRJCODE)” for BOTH GSB & GPU for default online.

OR

cp 9;suba 4;prjtit'GPU';prjobs'GPU';initprj(1,'PRJCODE')

# use “initprj(1,'PRJCODE)” for GPU only for standby or only one backend.

=====

**USER4 TERMINAL :**

>tpa 1299 1299 1350 1350 51 51

>prjfreq

>lnkndasq

>gts'3c48'

>strndas

=====

**RECORD TERMINAL :**

~/bin/released/record PRJCODE filename integration

eg. ~/bin/released/record PRJCODE GPU\_TEST.lta 1

=====

**4. ANALYSING THE RESULT USING TAX PROGRAM :**

The results are analysed using the software tax (built at GMRT). Follow the steps given below :

scp the data to any NIS m/c's /scratch area.

edit fmt to 31.4f

# Offline analysis: (currently in node51 "192.168.11.15" m/c)

~/bin/released/ltahdr

~/bin/released/listscan

~/bin/released/gvfits

## **5. POWER ON-OFF PROCEDURE**

### **a. POWER OFF :**

1. Shutdown the nodes using script “ “ on node51.
2. Switch OFF Roach units in GPU rack : GPU ROACH 1 & 2 by holding down the black switch on the front panel for 10sec.
3. Switch off signal generator at bottom of the GPU rack.

### **a. POWER ON :**

1. Turn On the GPU nodes 51 to 54.
2. Switch ON Roach units in GPU rack : GPU ROACH 1 & 2 by pressing the black switch on the front panel.
3. Switch ON the signal generator in the GPU rack.
4. Settings on the signal generator to feed clock.
  1. Set Frequency :
  
  
  2. Set Amplitude :
  
  
  3. RF ON and MOD off :
  
  
  4. Reference :