

SOP for 15mtr Backend Receiver Tests

Irappa, Navnath, Ajith : 11/01/2012

I : Power ON and instrument settings

- (a) Power on wall socket and AC distrib in the RACK (*verify : PC and Analog bin will be ON*)
- (b) Now Switch ON the instruments and do the following settings -
- (c) Signal generator settings as CLOCK to iBOB : Freq = 200MHz , Power = 0dbm, Set “Mod to OFF and RF to ON”
as LO Ref : connect from 10MHz REF OUT at instrument rear side (*verify : will be usually connected*)
- (d) Wavetek Generator settings as 1 pps trigger for ADC : SQUARE standard waveform, FREQ 1 Hz , Amplitude 5 Vp (*default on power on*)
- (e) Power on the iBOB by switching on the power supply switch at the rear side. (*verify : current is below 3 Amps*)

II : Control PC Login and Settings

- (a) LOGIN in to Linux as user : sandeep / gmrt passwd : sandeep123 / gmrt123
- (b) Open three Xterm (*one for LO Freq setting, one for Backend control, one for Offline plotting*)
- (c) Login as root in all terminals [sandeep@sdippc ~]\$ su passwd : Root123
- (d) LO Freq setting terminal [root@sdippc ~]# cd /home/sandeep/15mtr_backend/LO_set/
- (e) Backend control terminal [root@sdippc ~]# source /usr/Xilinx/settings.sh
[root@sdippc ~]# cd /home/sandeep/15mtr_backend/PY_SCRIPTS/
- (f) Offline plotting terminal [root@sdippc ~]# cd /home/sandeep/15mtr_backend/TAX/
Dump data in the directory [root@sdippc ~]# cd /home/sandeep/15mtr_backend/DATA/

III : LO Frequency Settings

- (a) Open LO freq setting terminal
 - (f) Set LO freq [root@LO_set]# ./lo3fsw -f 277 (*set LO to 277 Mhz*)
- (keep this xterm open for further changes in LO setting)

IV. Programming the iBOB

(Skip this step for regular tests, since released version of program is available in PROM and will be automatically loaded on POWER-ON. Move to Step V)

V. Initialization, plotting and data dumping for designs not using PPS to iBOB

(Presently PPS is connected, skip this step, go to Step VI)

(a) Open Backend control terminal

(b) Set initialisation [root@sdippc PY_SCRIPTS]# ./init_wb_poco.py example_wb_poco.conf

(c) Online plotting [root@sdippc PY_SCRIPTS]# ./plot_wb_cross_dly_update.py example_wb_poco.conf -l

(c) Data dumping for offline plotting with tax

[root@sdippc PY_SCRIPTS]# ./plot_wb_cross_dly_update_dump.py example_wb_poco.conf -f <../DATA/file_name>

(keep this xterm open for control of backend)

VI. Initialization, plotting and data dumping for designs using PPS to iBOB

(a) Open Backend control terminal

(b) Set initialisation [root@sdippc PY_SCRIPTS]# ./init_wb_poco_pps.py example_wb_poco_pps.conf

(verify : “All Done” message on screen, if message not seen, power reset to iBOB and start with Step VI(b))

(c) Online plotting [root@sdippc PY_SCRIPTS]# ./plot_wb_cross_dly_update_pps_ack_ts.py example_wb_poco_pps.conf -l

(verify : A separate window with “live” self plots will appear within a few minutes)

(d) Data dumping for offline plotting with tax

[root@sdippc PY_SCRIPTS]# ./plot_wb_cross_dly_update_dump_pps_ack_ts.py example_wb_poco_pps.conf -f <../DATA/file_name>

(verify : Output data file with <filename> in cd /home/sandeep/Sandeep/iBOB/digital_backend_15m/wb_200mhz_min1sec_acc/)

(keep this xterm open for control of backend)

VII. Offline Analysis using tax

(a) Open Offline Plotting terminal

(b) Plot self spectrum for Ch 1 – 1000

[root@sdippc TAX]# ./xtrgsb32 -v ../DATA/<file_name> -c 1,1000 -t 1,10000

(c) Plot cross spectrum for Ch 10 – 1000

[root@sdippc TAX]# ./xtrgsb32 -v ../DATA/<file_name> -c 10,1000,1 -t 1,5,1,5 -n 1 -r C00

(d) Plot cross Ch – 300 over time

[root@sdippc TAX]# ./xtrgsb32 -v ../DATA/<file_name> -c 300 -t 1,100000000 -n 1 -r C00

(e) Plot required records

[root@sdippc TAX]# ./xtrgsb32 -v ../DATA/<file_name> -c 10,1000,1 -t 1,57,1,57 -n 1 -r C00

(keep this xterm open for Offline plotting)