

Command References

rspctl (Expert mode)

```
-bash-4.2$ rspctl -X
rspctl usage:

--- RCU control -----
rspctl --rcu                [--select=<set>] # show current rcu control setting
rspctl --rcu=0x00000000    [--select=<set>] # set the rcu control registers
mask      value

0x00000007F INPUT_DELAY  Sample delay for the data from the RCU.
0x000000080 INPUT_ENABLE Enable RCU input.

0x000000100 LBL-EN       supply LBL antenna on (1) or off (0)
0x000000200 LBH-EN       sypply LBH antenna on (1) or off (0)
0x000000400 HB-EN        supply HB on (1) or off (0)
0x000000800 BANDSEL      low band (1) or high band (0)
0x00001000 HB-SEL-0      HBA filter selection
0x00002000 HB-SEL-1      HBA filter selection

Options : HBA-SEL-0 HBA-SEL-1 Function
      0      0      210-270 MHz
      0      1      170-230 MHz
      1      0      110-190 MHz
      1      1      all off
0x00004000 VL-EN         low band supply on (1) or off (0)
0x00008000 VH-EN         high band supply on (1) or off (0)

0x00010000 VDIG-EN       ADC supply on (1) or off (0)
0x00020000 LBL-LBH-SEL   LB input selection 0=LBL, 1=LBH
0x00040000 LB-FILTER      LB filter selection
      0      10-90 MHz
      1      30-80 MHz

0x00080000 ATT-CNT-4     on (1) is 1dB attenuation
0x00100000 ATT-CNT-3     on (1) is 2dB attenuation
0x00200000 ATT-CNT-2     on (1) is 4dB attenuation
0x00300000 ATT-CNT-1     on (1) is 8dB attenuation
```

0x00800000 ATT-CNT-0 on (1) is 16dB attenuation

0x01000000 PRSG pseudo random sequence generator on (1), off (0)

0x02000000 RESET on (1) hold board in reset

0x04000000 SPEC_INV Enable spectral inversion (1) if needed. see --specinv

0x08000000 TBD reserved

0xF0000000 RCU VERSION RCU version, read-only

```
rspctl [ --rcumode      |
        --rcuprsg      |
        --rcureset     |
        --rcuattenuation |
        --rcudelay     |
        --rcuenable     |
        ]+ [ --select=<set> ] # control RCU by combining one or more of these options with RCU selection
```

--rcumode [0..7] # set the RCU in a specific mode

Possible values: 0 = OFF

1 = LBL 10MHz HPF 0x00017900

2 = LBL 30MHz HPF 0x00057900

3 = LBH 10MHz HPF 0x00037A00

4 = LBH 30MHz HPF 0x00077A00

5 = HB 110-190MHz 0x0007A400

6 = HB 170-230MHz 0x00079400

7 = HB 210-270MHz 0x00078400

--rcuprsg[=0] # turn psrg on (or off)

--rcureset[=0] # hold rcu in reset (or take out of reset)

--rcuattenuation=[0..31] # set the RCU attenuation (steps of 0.25dB)

--rcudelay=[0..127] # set the delay for rcu's (steps of 5ns or 6.25ns)

--rcuenable[=0] # enable (or disable) input from RCU's

```
rspctl --specinv[=0] [ --select=<set> ] # enable (or disable) spectral inversion
```

```
rspctl --mode=[0..7] [ --select=<set> ] # set rcumode in a specific mode
```

 # enable(or disable) input from RCU's

 # enable(or disable) spectral inversion

 # set the hbadelays to 253

--- Signalprocessing -----

```
rspctl --weights                [ --select=<set> ] # get weights as complex values
```

Example --weights --select=1,2,4:7 or --select=1:3,5:7

```
rspctl --weights=value.re[,value.im][--select=<set>][--beamlets=<set>] # set weights as complex value
OR   --weights="(value.re,value.im)(value.re,value.im)" [--select=<set>][--beamlets=<set>] # set multiple
weights
```

as complex value for the same amount of selected beamlets

```
rspctl --aweights          [--select=<set>] # get weights as power and angle (in degrees)
rspctl --aweights=amplitude[,angle] [--select=<set>] # set weights as amplitude and angle (in degrees)
rspctl --subbands          [--select=<set>] # get subband selection
rspctl --subbands=<set>     [--select=<set>] # set subband selection
```

Example --subbands sets: --subbands=0:39 or --select=0:19,40:59

```
rspctl --xcsubband          # get the subband selection for cross correlation
rspctl --xcsubband=<int>    # set the subband to cross correlate
rspctl --wg                 [--select=<set>] # get waveform generator settings
rspctl --wg=freq [--phase=..] [--amplitude=..] [--select=<set>] # set waveform generator settings
```

--- Status info -----

```
rspctl --version            [--select=<set>] # get version information
rspctl --status             [--select=<set>] # get status of RSP boards
rspctl --tdstatus           [--select=<set>] # get status of TDS boards
rspctl --spustatus          [--select=<set>] # get status of SPU board
rspctl --realdelays[=<list>] [--select=<set>] # get the installed 16 delays of one or more HBA's
rspctl --regstate           # show update status of all registers once every second
rspctl --latency            # show latency of ring and all lanes
```

--- Statistics -----

```
rspctl --statistics[=(subband|beamlet)] # get subband (default) or beamlet statistics
      [--select=<set>]                  #
      [--duration=<seconds>]            #
      [--integration=<seconds>]         #
      [--directory=<directory>]        #
rspctl [--xcangle] --xcstatistics [--select=first,second] # get crosscorrelation statistics (of pair of RSP boards)
      [--duration=<seconds>]            #
      [--integration=<seconds>]         #
      [--directory=<directory>]        #
```

--- Miscellaneous -----

```
rspctl --clock[=<int>]      # get or set the clock frequency of clocks in MHz
rspctl --rspclear           [--select=<set>] # clear FPGA registers on RSPboard
rspctl --hbadelays[=<list>] [--select=<set>] # set or get the 16 delays of one or more HBA's
rspctl --tbbmode[=transient | =subbands,<set>] # set or get TBB mode, 'transient' or 'subbands', if subbands
then specify subband set
```

```

rspctl --datastream[=0|1|2|3]          # set or get the status of data stream to cep
rspctl --swapxy[=0|1] [--select=<set>]  # set or get the status of xy swap, 0=normal, 1=swapped
rspctl --bitmode[=4|8|16]              # set or get the number of bits per sample

--- Raw register control -----
### WARNING: to following commands may crash the RSPboard when used wrong! ###
rspctl --readblock=RSPboard,hexAddress,offset,datalength  # read datalength bytes from given address

rspctl --writeblock=RSPboard,hexAddress,offset,hexData    # write data to given address

In all cases the maximum number of databytes is 1480
Address order: BLPID, RSP, PID, REGID

```

beamctl

- (Undocumented) -j/--remotehost: Use a remote server to host the beamserv?

```

bash-4.2$ beamctl -h
Usage: beamctl <rcuspec> <dataspec> <digpointing> [<digpointing> ...] FOR LBA ANTENNAS
      beamctl <rcuspec> <anapointing> [<anapointing> ...] [<dataspec> <digpointing> [<digpointing> ...]]
FOR HBA ANTENNAS
      beamctl --calinfo
where:
<rcuspec>    = --antennaset --rcus --band (or --antennaset --rcus --rcumode)
<dataspec>   = --subbands --beamlets
<digpointing> = --digdir
<anapointing> = --anadir
with option arguments:
--antennaset=name # name of the antenna (sub)field the RCU's are part of, may not conflict with band
                  # name = LBA_INNER | LBA_OUTER | LBA_SPARSE_EVEN | LBA_SPARSE_ODD |
                  # LBA_X | LBA_Y | HBA_ZERO | HBA_ONE | HBA_DUAL | HBA_JOINED |
                  # HBA_ZERO_INNER | HBA_ONE_INNER | HBA_DUAL_INNER | HBA_JOINED_INNER
--rcus=<set>      # subselection of RCU's
--band=name       # name of band selection, may not conflict with antennaset
                  # name = 10_90 | 30_90 | 110_190 | 170_230 | 210_250
--subbands=<set>  # set of subbands to use for this beam
--beamlets=<list> # list of beamlets on which to allocate the subbands
                  # beamlet range = 0..247 when Serdes splitter is OFF
                  # beamlet range = 0..247 + 1000..1247 when Serdes splitter is ON
--digdir=longitude,latitude,type[,duration]
                  # lon,lat are floating point values specified in radians

```

```
# type is SKYSCAN or almost any other coordinate system
# SKYSCAN will scan the sky with a L x M grid in the (l,m) plane
--anadir=longitude,latitude,type[,duration]
# direction of the analogue HBA beam
--rcumode=0..7 # Old-style RCU mode to use (DEPRECATED; only available for
compatibility with existing scripts. Please use antenna-
set + band selection. The rcumode selected here must not
conflict with the selected antennaset)
--help # print this usage
```

The order of the arguments is trivial.

This utility connects to the CalServer to activate the antennas in set --antennaSet containing the selected RCU's. The CalServer sets those RCU's in the mode specified by --rcumode. Another connection is made to the BeamServer to create a beam on the selected antennafield pointing in the direction specified with --digdir.

Revision #1

Created 8 December 2020 14:03:29 by David

Updated 12 July 2021 19:19:38 by David