DBBC3 PYTHON PACKAGE

Helge Rottmann

MPIfR

DBBC3 PYTHON PACKAGE

Purpose: DBBC3 monitoring and control from python

- Available on github:
  - https://github.com/mpifr-vlbi/dbbc3
- Documentation is (partially) available:
  - https://mpifr-vlbi.github.io/dbbc3/index.html
- Will be made available also via PIP

Max-Planck-Institut für Radioastronomie

OVERVIEW

- Package contents
  - „Low-level“ implementation of most DBBC3 commands
  - „High-level“ validation routines (e.g. check sampler offsets/gains)
  - Utility scripts to interact with the DBBC3 (e.g. dbbc3client, dbbc3ctl)

SIMPLE EXAMPLE

```
dbbc3 = DBBC3(host=134.104.30.223)
print (dbbc3.dbbcif(0))
print (dbbc3.dbbcif('B‘))
dbbc3.disconnect()
```

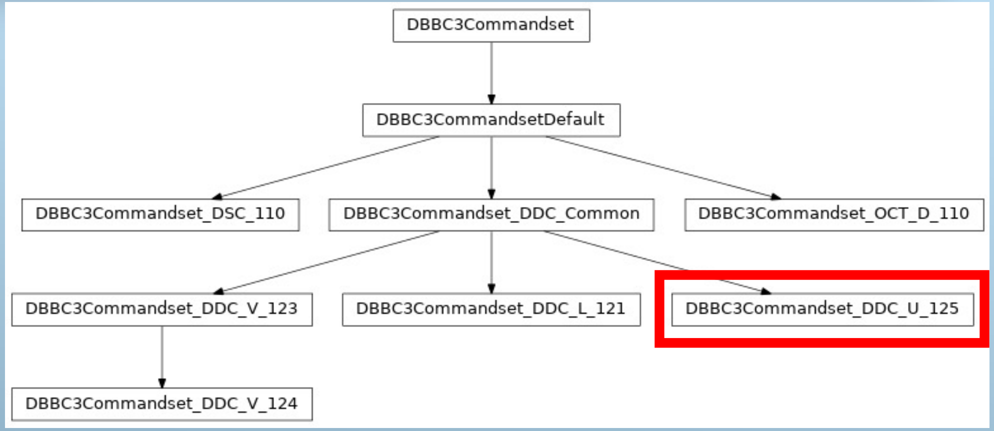
Output:

```
{'inputType': 2, 'attenuation': 24, 'mode': 'agc', 'count': 32095,
'target': 32000}
{'inputType': 2, 'attenuation‘: 27, 'mode‘: 'agc', 'count‘: 31083,
'target': 32000}
```

Max-Planck-Institut für Radioastronomie

DBBC3 COMMANDSET

On initialization the mode is determined and the corresponding set of commands is attached


DBBC3Commandset
DBBC3CommandsetDefault
DBBC3Commandset_DSC_110
DBBC3Commandset_DDC_Common
DBBC3Commandset_OCT_D_110
DBBC3Commandset_DDC_V_123
DBBC3Commandset_DDC_L_121
DBBC3Commandset_DDC_U_125
DBBC3Commandset_DDC_V_124

Command set logic:

Python methods name = prefix_dbbc3 command name

Prefixes: adb3l, core3h or no prefix for "general" dbbc3 commands

Examples:

```
dbbc3.dbbcifa()
dbbc3.core3h_core3_bstat(board, sampler)
dbbc3.adb3l_reseth()
```

Command set documentation:
https://mpifr-vlbi.github.io/dbbc3/source/dbbc3commandset.html#module-dbbc3.DBBC3Commandset

VALIDATION METHODS

`DBBC3Validation` class implements higher-level checks

e.g. `validateSamplersOffsets` checks that sampler offsets have been calibrated correctly

- set IF input power to optimal level
- determine and evaluate asymmetry of bit statistics
- reset IF input power to initial level

All validation methods provide a *validation report* with feedback on state, checks performed, problem resolution suggestion etc., e.g.:

```
...
[OK] ===Checking synthesizer lock state of board A - Locked
[OK] ===Checking GCoMo synthesizer frequency of board A - Freq=8048 MHz
[FAIL]/[ERROR] === Checking IF power level on core board A - IF power not on target value. Should be close to 32000 is 2842
[RESOLUTION] Check and adjust IF input power levels (should be @ -11dBm)
...
```

Max-Planck-Institut für Radioastronomie

UTILITY SCRIPTS

`dbbc3client.py`	simple client to send commands to the DBBC3
`dbbc3ctl.py`	script for performing higher-level checks and tasks
`setupDBBC3_DDC_U.py`	validation script for DDC_U mode
`setupDBBC3_DDC_V.py`	validation script for DDC_V mode
`setupDBBC3_OCT_D.py`	validation script for OCT_D mode

MULTICAST

Some DBBC3 modes send multicast of current state / settings:

- Tsys
- TP On/off
- BBC settings: frequency bandwidth
- IF settings: counts, attenuation etc.
- Synthesizer state /freq
- Etc.

`DBBC3Multicast` class supports parsing of multicast messages. Contents are returned as a dictionary.

FURTHER DEVELOPMENTS

- Write DBBC3 monitoring client (text-based and GUI)
- `dbbc3ctl.py`
  - Add additional verification tasks
  - Allow execution of command batch files
  - Add looping for long-term stability tests