ixbps

Library scope:	global
Named arguments:	supported

Introduction

provides functions for Ixia Breaking Point

Breaking Point testers setting is set by `tester` section in `local.yaml`. Users need to specify the physical ports used by the test by its card and port number.

Setting example:

```
ixbps01:
    device: ixbps01
    config: test.bpt
    real-port:
      - card:  1
        port:  0
      - card:  1
        port:  1
```

Notes: Ignore the *self* parameters when using those keywords. The module requires Breaking Point Python library installedd properly to work.

Shortcuts

Cleanup Tests · **C**lose · **G**et Card Config · **G**et Card Mode · **G**et Test Report · **L**oad Config · **S**et Card Config · **S**tart Test · **S**top Test · **W**ait Until Finish

Keywords

Keyword	Arguments	Documentation
Cleanup Tests	*self*	Cleans up running test and release their ports
Close	*self*	Closes the connection to the BP box
Get Card Config	*self*, *slot_num*	Get card configuration for *slot_num* Parameter: ▪ *slot_num* is *all* or an integer start from 1 Result is a json formatted string contains the informaition for specific slot or *all* slots.
Get Card Mode	*self*, *slot_num*	Gets the *mode* of a specific slot
Get Test Report	*self*, *report_name=result*, *format=csv*	Gets and saves the test report to local disk The report will be in PDF format. If `export_csv` is `True` then test results are also exported by CSV format.
Load Config	*self*, *config_name=*, *force=True*	Loads test configuration config_name is defined in `local.yaml` or specific by user in the main scenario.
Set Card Config	*self*, *slot*, *action=mode*, *param=ixload*	Changes the configuration of BPS card Parameters: ▪ slot: slot number ▪ action: `mode` or `perfacc` ▪ param: depending on *action* Values of *param*: ▪ if *action* is `mode` then *param* should be `ixload`, `bp` or `bpl23` (BreakingPoint L2/3) ▪ if *action* is `perfacc` then *param* should be `${TRUE}` or `${FALSE}`
Start Test	*self*, *test_name=None*, *force=True*	Starts a test by its name The system automatically reserve the ports defined in `local.yaml`. The reserved ports are released when the test is stopped `test_name` is the name of the testmodel saved in the configuration. If `test_name` is *None*, the 1st testmodel will be used. If `force` is *True* then all running tests and their reserved ports will be released.
Stop Test	*self*, *wait=5s*	Stops a running test
Wait Until Finish	*self*, *interval=30s*, *timeout=30m*, *verbose=False*	Waits until the test finished or timeout **Notes**: This is a blocking keyword