
MongoRoc Documentation

Release 0

L.Mirabito

Jan 07, 2020

CONTENTS

1 Indices and tables	7
Python Module Index	9
Index	11

Contents:

class `MongoRoc.MongoRoc` (*host, port, dbname, username, pwd*)

Main class to access the Mongo DB

HR2_ChangeCTest (*channel, ctest, idif=0, iasic=0*)

Change the CTEST value of one channel of specified asics, modified asics are tagged for upload

Parameters

- **channel** – Pad tested
- **ctest** – CTEST value
- **idif** – DIF_ID, if 0 all DIFs are changed
- **iasic** – asic number, if 0 all Asics are changed

HR2_ChangeGain (*idif, iasic, ipad, scale*)

Scale the gain of one pad of specified asics, modified asics are tagged for upload

Parameters

- **idif** – DIF_ID, if 0 all DIFs are changed
- **iasic** – asic number, if 0 all Asics are changed
- **ipad** – Channel number
- **scale** – ratio Gain_new/Gain

HR2_ChangeMask (*M0, M1, M2, idif=0, iasic=0*)

Set the 3 masks for specified asics, modified asics are tagged for upload

Parameters

- **M0** – Hexadecimal string of threshold 0 mask
- **M1** – Hexadecimal string of threshold 1 mask
- **M2** – Hexadecimal string of threshold 2 mask
- **idif** – DIF_ID, if 0 all DIFs are changed
- **iasic** – asic number, if 0 all Asics are changed

HR2_ChangeThreshold (*B0, B1, B2, idif=0, iasic=0*)

Set the 3 thresholds of specified asics, modified asics are tagged for upload

Parameters

- **B0** – First Threshold
- **B1** – Second Threshold
- **B2** – Third Threshold
- **idif** – DIF_ID, if 0 all DIFs are changed
- **iasic** – asic number, if 0 all Asics are changed

HR2_RescaleGain (*gain0, gain1, idif=0, iasic=0*)

Rescale the gain of all pads with scale=gain1/gain0 of specified asics, modified asics are tagged for upload

Parameters

- **gain0** – initial gain
- **gain1** – new gain
- **idif** – DIF_ID, if 0 all DIFs are changed
- **iasic** – asic number, if 0 all Asics are changed

HR2_SetAsicGain (*idif, iasic, vnew*)

Set the gain of all pads of specified asics, modified asics are tagged for upload

Parameters

- **idif** – DIF_ID, if 0 all DIFs are changed
- **iasic** – asic number, if 0 all Asics are changed
- **vnew** – new gain

HR2_SetGain (*idif, iasic, ipad, vnew*)

Set the gain of one pad of specified asics, modified asics are tagged for upload

Parameters

- **idif** – DIF_ID, if 0 all DIFs are changed
- **iasic** – asic number, if 0 all Asics are changed
- **ipad** – Channel number
- **vnew** – new gain

HR2_SetMask (*list, idif=0, iasic=0*)

Mask the channels specified in the list for specified asics, modified asics are tagged for upload

Parameters

- **list** – List of channels to be masked
- **idif** – DIF_ID, if 0 all DIFs are changed
- **iasic** – asic number, if 0 all Asics are changed

HR2_setEnable (*enable, idif=0, iasic=0*)

Set the ENABLE tag for specified asics, modified asics are tagged for upload

Parameters

- **enable** – Enable value (1/0)
- **idif** – DIF_ID, if 0 all DIFs are changed
- **iasic** – asic number, if 0 all Asics are changed

HR2_setPowerPulsing ()

set Power pulsing on all ASICs

HR2_slowShaper ()

Slow down the shaper , Set SW100 F and K to 1

HR2_unsetPowerPulsing ()

Unset Power pulsing on all ASICs

PR2_Change6BDac (*idif, iasic, ich, dac*)

Change the 6BDAC valu of specified asics, modified asics are tagged for upload

Parameters

- **idif** – DIF_ID (IP>>16), if 0 all FEBs are changed
- **iasic** – asic number, if 0 all Asics are changed
- **ich** – The channel number
- **dac** – The DAC value

PR2_ChangeAllEnabled (*idif=0, iasic=0*)

Change all the ENable signals of PETIROC asic

Parameters

- **idif** – DIF_ID (IP>>16), if 0 all FEBs are changed
- **iasic** – asic number, if 0 all Asics are changed

PR2_ChangeDacDelay (*delay, idif=0, iasic=0*)

Change the DAC delay of specified asics, modified asics are tagged for upload

Parameters

- **delay** – Dac delay
- **idif** – DIF_ID (IP>>16), if 0 all FEBs are changed
- **iasic** – asic number, if 0 all Asics are changed

PR2_ChangeLatch (*Latch, idif=0, iasic=0*)

Change the Latch mode of specified asics, modified asics are tagged for upload

Parameters

- **Latch** – Latch value (1/0)
- **idif** – DIF_ID (IP>>16), if 0 all FEBs are changed
- **iasic** – asic number, if 0 all Asics are changed

PR2_ChangeMask (*idif, iasic, ich, mask*)

Change PETIROC2 MASKDISCRITIME parameter for one channel

Parameters

- **idif** – DIF_ID (IP>>16), if 0 all FEBs are changed
- **iasic** – asic number, if 0 all Asics are changed
- **ich** – The channel number
- **mask** – the channel mask

Warning 1 = inactive, 0=active

PR2_ChangeVthTime (*VthTime, idif=0, iasic=0*)

Change the VTHTIME threshold of specified asics, modified asics are tagged for upload

Parameters

- **VthTime** – Threshold of time discriminators
- **idif** – DIF_ID (IP>>16), if 0 all FEBs are changed
- **iasic** – asic number, if 0 all Asics are changed

PR2_Correct6BDac (*idif, iasic, cor*)

Correct the 6BDAC value of specified asics, modified asics are tagged for upload

Parameters

- **idif** – DIF_ID (IP>>16), if 0 all FEBs are changed
- **iasic** – asic number, if 0 all Asics are changed
- **cor** – A 32 channels array of corrections to be applied on the 6BDAC values of all channels

addAsic (*dif_num, header, version='PR2'*)

Add a new PETIROC2 to the asic list

Parameters

- **dif_num** – DIF ID (ipaddr in integer >>16)
- **header** – ASIC number
- **version** – PR2 for 2A , PR2B for 2B

addDIF (*difid, nasic, address='USB'*)

Add a DIF with asics

Parameters

- **address** – IP address of the GRIC or USB for SDHCAL ones
- **nasic** – Number of HARDROC2 asics connected
- **difid** – unused for GRIC, DIF id for SDHCAL

addFEB (*ipname, nasic, asictype='PR2'*)

Add a FEBV1 with asics

Parameters

- **ipname** – IP address of the FEBV1
- **nasic** – Number of PETIROC asics connected
- **asictype** – “PR2” for PETIROC2A , “PR2B” for PETIROC2B

configurations ()

List all jobcontrol configurations in the db

Obsolete Use MongoJob instead

createNewState (*name*)

Create a new state , version is set to 1

Parameters **name** – Name of the state

download (*statename, version, toFileOnly=False*)

Download a state configuration to /dev/shm/mgroc/ directory and load it in the MongoRoc object

Parameters

- **statename** – State name
- **version** – State version
- **toFileOnly** – if True and /dev/shm/mgroc/statename_version.json already exists, it exists

downloadConfig (*cname, version*)

Download a jobcontrol configuration to /dev/shm/mgjob/ directory

Parameters

- **cname** – Configuration name

- **version** – Configuration version

Obsolete Use MongoJob instead

initHR2 (*num, gain=128*)

HARDROC 2 initialisation, it creates a default dictionary representation of a HARDROC

Parameters

- **num** – Asic number
- **gain** – Channel gain

Returns the dictionary

initPR2 (*num, version='PR2'*)

PETIROC 2 initialisation, it creates a default dictionary representation of a PETIROC2

Parameters

- **num** – Asic number
- **version** – Asic type (PR2 or PR2B)

Returns the dictionary

reset ()

Reset connection to download another state

states ()

List all states in the DB

uploadChanges (*statename, comment*)

Upload a new version of the state it finds the last version of the state and upload a new one with incremented version number

Parameters

- **statename** – Name of the state
- **comment** – A comment on the changes

uploadConfig (*name, fname, comment, version=1*)

jobcontrol configuration upload

Obsolete Use MongoJob instead

uploadFromFile (*fname*)

Upload a state in DB from a JSON file

Parameters **fname** – File name

uploadFromOracle (*asiclist, statename, version, comment='NEW'*)

Migration method to update an ASIC list created with OracleAccess class to the DB

Parameters

- **asiclist** – List of asics created with OracleAccess
- **statename** – Name of the state
- **version** – version of the state
- **comment** – A comment on the state

uploadNewState (*comment='NEW'*)

Create a new state in the DB with data stored in object memory

Parameters `comment` – A comment on the state

MongoRoc.**IP2Int** (*ip*)

convert IP address string to int

Parameters `IP` – the IP address

Returns the encoded integer

MongoRoc.**instance** ()

Create a MongoRoc Object

Returns The MongoRoc Object

INDICES AND TABLES

- genindex
- modindex
- search

m

MongoRoc, 1

A

addAsic() (MongoRoc.MongoRoc method), 4
 addDIF() (MongoRoc.MongoRoc method), 4
 addFEB() (MongoRoc.MongoRoc method), 4

C

configurations() (MongoRoc.MongoRoc method), 4
 createNewState() (MongoRoc.MongoRoc method), 4

D

download() (MongoRoc.MongoRoc method), 4
 downloadConfig() (MongoRoc.MongoRoc method), 4

H

HR2_ChangeCTest() (MongoRoc.MongoRoc method), 1
 HR2_ChangeGain() (MongoRoc.MongoRoc method), 1
 HR2_ChangeMask() (MongoRoc.MongoRoc method), 1
 HR2_ChangeThreshold() (MongoRoc.MongoRoc method), 1
 HR2_RescaleGain() (MongoRoc.MongoRoc method), 1
 HR2_SetAsicGain() (MongoRoc.MongoRoc method), 2
 HR2_setEnable() (MongoRoc.MongoRoc method), 2
 HR2_SetGain() (MongoRoc.MongoRoc method), 2
 HR2_SetMask() (MongoRoc.MongoRoc method), 2
 HR2_setPowerPulsing() (MongoRoc.MongoRoc method), 2
 HR2_slowShaper() (MongoRoc.MongoRoc method), 2
 HR2_unsetPowerPulsing() (MongoRoc.MongoRoc method), 2

I

initHR2() (MongoRoc.MongoRoc method), 5
 initPR2() (MongoRoc.MongoRoc method), 5
 instance() (in module MongoRoc), 6
 IP2Int() (in module MongoRoc), 6

M

MongoRoc (class in MongoRoc), 1
 MongoRoc (module), 1

P

PR2_Change6BDac() (MongoRoc.MongoRoc method), 2

PR2_ChangeAllEnabled() (MongoRoc.MongoRoc method), 3

PR2_ChangeDacDelay() (MongoRoc.MongoRoc method), 3

PR2_ChangeLatch() (MongoRoc.MongoRoc method), 3

PR2_ChangeMask() (MongoRoc.MongoRoc method), 3

PR2_ChangeVthTime() (MongoRoc.MongoRoc method), 3

PR2_Correct6BDac() (MongoRoc.MongoRoc method), 3

R

reset() (MongoRoc.MongoRoc method), 5

S

states() (MongoRoc.MongoRoc method), 5

U

uploadChanges() (MongoRoc.MongoRoc method), 5

uploadConfig() (MongoRoc.MongoRoc method), 5

uploadFromFile() (MongoRoc.MongoRoc method), 5

uploadFromOracle() (MongoRoc.MongoRoc method), 5

uploadNewState() (MongoRoc.MongoRoc method), 5