
dragon_rest Documentation

Release 0.0.1

Brenden Matthews

Jan 06, 2022

CONTENTS:

1	Quickstart	3
1.1	Install pip package	3
1.2	Write the Python codes	3
2	Reference	5
2.1	API Reference	5
	Python Module Index	9
	Index	11

If you use Halong Mining's DragonMint T1, and would like to interact with it programmatically using Python, then look no further! I have implemented the full HTTP REST API for your enjoyment and pleasure.

QUICKSTART

1.1 Install pip package

```
$ pip install dragon-rest
```

1.2 Write the Python codes

```
from dragon_rest.dragons import DragonAPI

dragon_host = '10.0.0.1'
api = DragonAPI(dragon_host,
                username='admin',
                password='dragonadmin')

r = api.summary() # get summary
print(r)
api.upgradeUpload('t1_20180515_060842.swu') # upgrade firmware
# now you're in the big leagues, boye
```

For more details on the API, take a look at `DragonAPI()`.

REFERENCE

For details on the DragonMint API, see: <https://halongmining.com/api/>

2.1 API Reference

This page contains auto-generated API reference documentation¹.

2.1.1 dragon_rest

Submodules

`dragon_rest.dragons`

DragonAPI is a Python wrapper for the DragonMint T1.

Module Contents

Classes

<i>DragonAPI</i>	DragonMint/Innosilicon API wrapper.
------------------	-------------------------------------

class `dragon_rest.dragons.DragonAPI`(*host*, *username*='admin', *password*='dragonadmin', *timeout*=60, *jwt*=None)

DragonMint/Innosilicon API wrapper.

This wrapper should work with both DragonMint and Innosilicon branded miners. If you have problems with the API, please open GitHub issue accordingly.

The official documentation for the API is available at: <https://halongmining.com/api/>

Example:

```
from dragon_rest.dragons import DragonAPI

dragon_host = '10.0.0.1'
api = DragonAPI(dragon_host,
```

(continues on next page)

¹ Created with sphinx-autoapi

(continued from previous page)

```
        username='admin',
        password='dragonadmin')

r = api.summary()
print(r)
# now you're in the big leagues, boye
```

static is_dragon(host, timeout=1)

Check if host is a dragon.

Check if the specified host is a dragon based on simple heuristic. The code simply checks if particular strings are in the index page. It should work for DragonMint or Innosilicon branded miners.

__post(self, path, data=None)**__post_files**(self, path, files)**__get_stream**(self, path)**auth**(self)

Authenticate with the miner and obtain a JSON web token (JWT).

summary(self)

Fetch DEVS, POOLS and Fan Speed from the cgminer API.

overview(self)

Fetch miner overview.

Fetch miner type, hardware information, network information and versions of the miner

errors(self)

Returns errors

pools(self)

Receive the configured pools of the miner.

updatePools(self, pool1, username1, password1, pool2=None, username2=None, password2=None, pool3=None, username3=None, password3=None)

Change the pools of the miner. This call will restart cgminer.

updatePassword(self, user, currentPassword, newPassword)

Change the password of a user.

network(self)

Get the current network settings.

updateNetwork(self, dhcp='dhcp', ipaddress=None, netmask=None, gateway=None, dns=None)

Change the current network settings.

type(self)

Return the type of the miner.

reboot(self)

Reboot the miner.

poweroff(self)

Power Off the Miner.

restartCgMiner(self)

Restart cgminer.

factoryReset(*self*)

Remove all user settings and reboot the miner.

getAutoTune(*self*)

Return cgminer auto-tune mode.

getAutoTuneStatus(*self*)

Return cgminer Auto-Tune status.

setAutoTune(*self*, *autotune*, *level*=2)

Set cgminer to use or not embedded auto-tune functionality.

upgradeUpload(*self*, *file*)

Upgrade the firmware of the miner.

upgradeDownload(*self*, *url*)

Upgrade the firmware of the miner with a URL of the update file.

getLatestFirmwareVersion(*self*)

Return the latest version available of the miner.

getDebugStats(*self*)

Get debug status of miner.

Return the cgminer stats of each board and each chip of the miner.

streamLogs(*self*)

Return systemd-journald logs in chunked packages.

Package Contents

dragon_rest.name = dragon-rest

- genindex
- modindex
- search

PYTHON MODULE INDEX

d

`dragon_rest`, [5](#)

`dragon_rest.dragons`, [5](#)

Symbols

`__get_stream()` (*dragon_rest.dragons.DragonAPI method*), 6
`__post()` (*dragon_rest.dragons.DragonAPI method*), 6
`__post_files()` (*dragon_rest.dragons.DragonAPI method*), 6

A

`auth()` (*dragon_rest.dragons.DragonAPI method*), 6

D

`dragon_rest`
 module, 5

`dragon_rest.dragons`
 module, 5

`DragonAPI` (*class in dragon_rest.dragons*), 5

E

`errors()` (*dragon_rest.dragons.DragonAPI method*), 6

F

`factoryReset()` (*dragon_rest.dragons.DragonAPI method*), 6

G

`getAutoTune()` (*dragon_rest.dragons.DragonAPI method*), 7

`getAutoTuneStatus()`
 (*dragon_rest.dragons.DragonAPI method*), 7

`getDebugStats()` (*dragon_rest.dragons.DragonAPI method*), 7

`getLatestFirmwareVersion()`
 (*dragon_rest.dragons.DragonAPI method*), 7

I

`is_dragon()` (*dragon_rest.dragons.DragonAPI static method*), 6

M

module

`dragon_rest`, 5

`dragon_rest.dragons`, 5

N

`name` (*in module dragon_rest*), 7

`network()` (*dragon_rest.dragons.DragonAPI method*), 6

O

`overview()` (*dragon_rest.dragons.DragonAPI method*), 6

P

`pools()` (*dragon_rest.dragons.DragonAPI method*), 6

`poweroff()` (*dragon_rest.dragons.DragonAPI method*), 6

R

`reboot()` (*dragon_rest.dragons.DragonAPI method*), 6

`restartCgMiner()` (*dragon_rest.dragons.DragonAPI method*), 6

S

`setAutoTune()` (*dragon_rest.dragons.DragonAPI method*), 7

`streamLogs()` (*dragon_rest.dragons.DragonAPI method*), 7

`summary()` (*dragon_rest.dragons.DragonAPI method*), 6

T

`type()` (*dragon_rest.dragons.DragonAPI method*), 6

U

`updateNetwork()` (*dragon_rest.dragons.DragonAPI method*), 6

`updatePassword()` (*dragon_rest.dragons.DragonAPI method*), 6

`updatePools()` (*dragon_rest.dragons.DragonAPI method*), 6

`upgradeDownload()` (*dragon_rest.dragons.DragonAPI method*), 7

`upgradeUpload()` (*dragon_rest.dragons.DragonAPI method*), 7