

Bug report 2017-12-4

Mochad and Home Assistant

I have a situation with a running mochad installation on a server with a local ip on port 1099

(as defaults suggest)

I am migrating from OpenHab to Home Assistant (and before that openRemote) because I am a fan of MQTT and python.

OpenHab and mochad work fine so Home Assistant should be able to do the job too.

I have multiple iot devices I like to integrate but when I am not able to control the lights. I have 5 very old Marmitec (x10) devices and 3 less old Marmitek devices. Marmitek does not sell these x10 devices any more but I have them installed in my walls.

My super user (My Wife) will not approve my new solution. So here my motivation to solve this issue with you.

I found out that with the HA is using a asynchronous way of starting up devices.

I added some nasty debug logging to the sources to find out what goes wrong.

```
2017-12-04 20:54:59 DEBUG (Thread-5) [pymochad.device] get status for c12
2017-12-04 20:54:59 DEBUG (Thread-2) [pymochad.device] get status for c1
2017-12-04 20:54:59 DEBUG (Thread-8) [pymochad.device] get status for c2
2017-12-04 20:54:59 DEBUG (Thread-5) [pymochad.controller] Send command from
controller: getstatus c12

2017-12-04 20:54:59 DEBUG (Thread-2) [pymochad.controller] Send command from
controller: getstatus c1

2017-12-04 20:54:59 DEBUG (Thread-8) [pymochad.controller] Send command from
controller: getstatus c2

2017-12-04 20:54:59 DEBUG (Thread-5) [pymochad.controller] Start read data
from controller
2017-12-04 20:54:59 DEBUG (Thread-2) [pymochad.controller] Start read data
from controller
2017-12-04 20:54:59 DEBUG (Thread-2) [pymochad.controller] data read: off
on
on

2017-12-04 20:54:59 DEBUG (Thread-8) [pymochad.controller] Start read data
from controller
2017-12-04 20:54:59 DEBUG (Thread-2) [homeassistant.components.light.mochad]
Got device status: off
on
on for c1
```

just a snapshot of the logging but you can see what goes wrong. Status for lights c12, c1 and c2 are queried. So the commands are sent off to pymochad. And stuff is read.

But thread-2 (light c1) got all results off, on and on.

So I tried some locking to make stuff atomic.

```
def _get_device_status(self):
    """Get the status of the light from mochad."""
    from threading import Lock
    lock = Lock()
    lock.acquire()
    _LOGGER.debug("Start getting light device status for %s",
self._address)
    status = self.device.get_status().rstrip()
    _LOGGER.debug("Got device status: %s for %s", status, self._address)
    lock.release()
    return status == 'on'
```

So I got a logging like this:

```
2017-12-04 21:00:43 DEBUG (Thread-7)
[homeassistant.components.switch.mochad] Start getting switch device status
for c2
2017-12-04 21:00:43 DEBUG (Thread-7) [pymochad.device] get status for c2
2017-12-04 21:00:43 DEBUG (Thread-2)
[homeassistant.components.switch.mochad] Start getting switch device status
for c12
2017-12-04 21:00:43 DEBUG (Thread-5) [homeassistant.components.light.mochad]
Start getting light device status for c1
2017-12-04 21:00:43 DEBUG (Thread-7) [pymochad.controller] Send command from
controller: getstatus c2
```

So this is definitely not the solution.

My idea towards a solution is that pymochad (the external pipy module) should not accept new commands (requests) from HA before answering a former request.

Maybe someone can shine some lights on this? What is the best way to approach?