Contents

[Installation of python paramiko package 2](#_Toc499016218)

[Settings of monitors 2](#_Toc499016219)

[File description 2](#_Toc499016220)

[Launch monitor script 2](#_Toc499016221)

[Testing 3](#_Toc499016222)

[Manual control 3](#_Toc499016223)

## Installation of python paramiko package

**sudo apt-get update**

**sudo apt-get install libffi-dev  
sudo pip install paramiko**

## Settings of monitors

In current situation, setting are placed in /static/data/setting.json file.

[  
 {**"id"**:**"1st\_monitor"**, **"ipaddr"**:**"192.168.1.159"**, **"gpio"**:18, **"description"**:**"1st relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"2nd\_monitor"**, **"ipaddr"**:**"192.168.1.160"**, **"gpio"**:17, **"description"**:**"2nd relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"3rd\_monitor"**, **"ipaddr"**:**"192.168.1.161"**, **"gpio"**:27, **"description"**:**"3rd relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"4th\_monitor"**, **"ipaddr"**:**"192.168.1.162"**, **"gpio"**:23, **"description"**:**"4th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"5th\_monitor"**, **"ipaddr"**:**"192.168.1.163"**, **"gpio"**:22, **"description"**:**"5th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"6th\_monitor"**, **"ipaddr"**:**"192.168.1.164"**, **"gpio"**:24, **"description"**:**"6th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"7th\_monitor"**, **"ipaddr"**:**"192.168.1.165"**, **"gpio"**:25, **"description"**:**"7th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"8th\_monitor"**, **"ipaddr"**:**"192.168.1.166"**, **"gpio"**:5, **"description"**:**"8th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
  
 {**"id"**:**"9th\_monitor"**, **"ipaddr"**:**"192.168.1.170"**, **"gpio"**:6, **"description"**:**"9th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"10th\_monitor"**, **"ipaddr"**:**"192.168.1.171"**, **"gpio"**:12, **"description"**:**"10th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"11th\_monitor"**, **"ipaddr"**:**"192.168.1.172"**, **"gpio"**:13, **"description"**:**"11th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"12th\_monitor"**, **"ipaddr"**:**"192.168.1.173"**, **"gpio"**:19, **"description"**:**"12th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"13th\_monitor"**, **"ipaddr"**:**"192.168.1.174"**, **"gpio"**:16, **"description"**:**"13th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"14th\_monitor"**, **"ipaddr"**:**"192.168.1.175"**, **"gpio"**:20, **"description"**:**"14th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"15th\_monitor"**, **"ipaddr"**:**"192.168.1.176"**, **"gpio"**:26, **"description"**:**"15th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**},  
 {**"id"**:**"16th\_monitor"**, **"ipaddr"**:**"192.168.1.177"**, **"gpio"**:21, **"description"**:**"16th relay"**, **"cycle"**:30, **"uid"**:**"ubuntu"**, **"pwd"**:**"0170101"**, **"auto"**:**false**}  
  
]

In every row,

* gpio: BCM pin number
* ipaddr: ip address specified miner
* cycle: ssh checking interval
* auto: false-manual control, true-automatic control of relay

## File description

* rebootcontroller.py: implementation of miner monitor threading class
* monitor.py: test python script using Flask backend

## Launch monitor script

sudo python monitor.py

## Testing

After launch script, flask backend will be working on port 5000.

You can get information and status of monitor threads by visiting http://<rpiip>:5000.

[

{

"SSHCkecking": "failed",

"cycle": 30,

"description": "1st relay",

"gpio": 18,

"id": "1st\_monitor",

"ipaddr": "192.168.1.159",

"status": "checking action"

},

{

"SSHCkecking": "failed",

"cycle": 30,

"description": "2nd relay",

"gpio": 17,

"id": "2nd\_monitor",

"ipaddr": "192.168.1.160",

"status": "checking action"

},

……

]

## Manual control

In manual mode, we can force reboot and power on to specified miner via relay.

1. Reboot action

Ex: we gonna reboot 2nd relay channel, we can send get request to rpi.

http://<rpiip>:5000/reboot?id=2nd\_monitor

if action succeeded, it will return “reboot action is executed”.

And program will enter into power off and power on step.

1. Power on action

Ex: we gonna reboot 1st relay channel, we can send get request to rpi.

http://<rpiip>:5000/poweron?id=1st\_monitor

And program will enter into power on step (relay on for POWERONTIME).