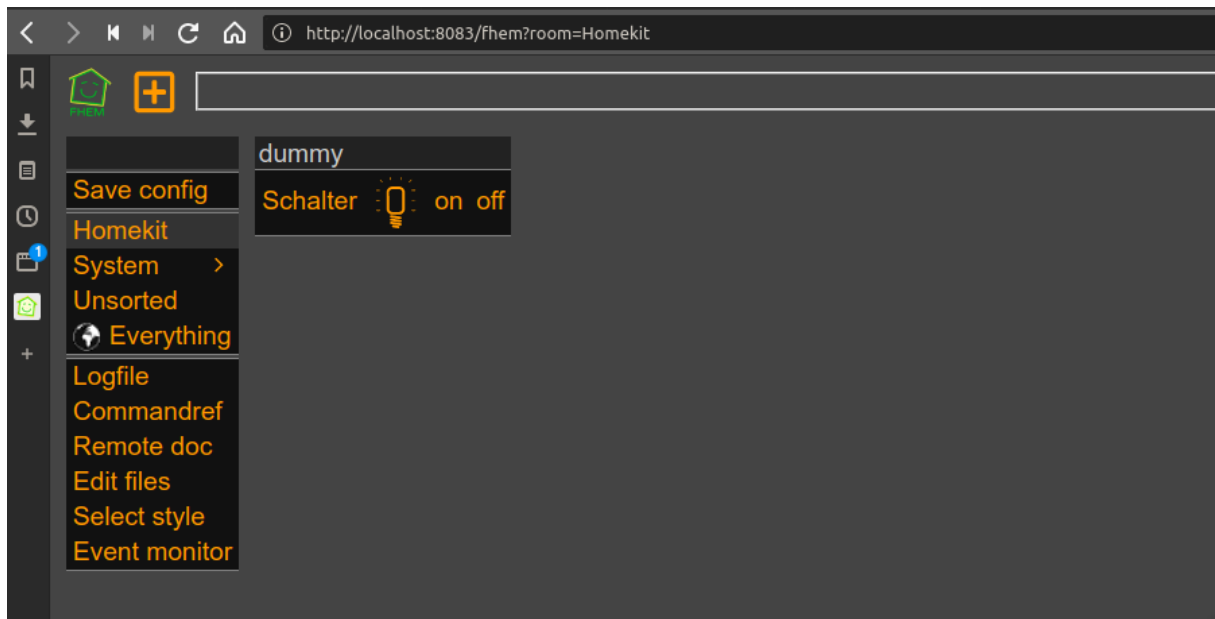

Home Automation Stack



The stack contains everything to run FHEM on a Docker host. Mosquitto is used as message broker. SIRI functions are realized with the help of a homebridge container. The complete stack runs on x86 as well as arm architectures. It is very easy to clone its complete productive environment and has a simple way to build a test system.

Todo

- DBLog Integration
- Boot config Raspberrypi for Homematic modul

define myHmUART HMUARTLGGW /dev/ttyAMA0

Requirements

- docker
- docker-compose

Installation raspberrypi

Raspian Download

Download the image of your choice: Raspian Download Unzip the image and install it with:

```
1  sudo dd bs=4M if=2019-09-26-raspbian-buster-full.img of=/dev/mmcblk0
    conv=fsync
2  sync
```

Eject the card and insert it again to mount the filesystems boot & rootfs. Touch a blank file ssh to enable

```
1  sudo touch /media/boot/ssh
2  sync
3  umount /media/boot
4  umount /media/rootfs
```

Eject the card and insert into your raspberry. After that power on the rpi and login with the known

```
1  ssh pi@raspberrypi4
```

```
1  pi@raspberrypi:~ $ passwd
2  Changing password for pi.
3  Current password:
4  New password:
5  Retype new password:
6  passwd: password updated successfully
7  pi@raspberrypi:~ $
```

System Update

```
1  sudo apt-get update
2  sudo apt-get dist-upgrade
```

Set timezone

```
1  sudo dpkg-reconfigure tzdata
```

Raspberry Config

- 1) Expand the root filesystem (A1 / Advanced Options)
 - 2) Update raspi-config
- ```
sudo raspi-config sudo reboot
```

### Intall additional packages

---

```
1 sudo apt-get install wget git apt-transport-https vim telnet zsh zsh-
autosuggestions zsh-syntax-highlighting
```

### Install oh-my-zsh

```
1 sh -c "$(curl -fsSL https://raw.githubusercontent.com/ohmyzsh/ohmyzsh/master/
tools/install.sh)"
```

### Install log2ram (/var/log 2 ram)

```
1 echo "deb http://packages.azlux.fr/debian/ buster main" | sudo tee /
etc/apt/sources.list.d/azlux.list
2 wget -qO - https://azlux.fr/repo.gpg.key | sudo apt-key add -
3 apt update
4 apt install log2ram
```

### Setup ssh key for user

```
1 ssh-keygen -t rsa -b 8192
```

### Install .ssh/config file to ignore strictHostKeyChecking

```
1 vi ~/.ssh/config
2
3 Host fhemlocalhost
4 Hostname localhost
5 Port 222
6 User fhem
7 StrictHostKeyChecking no
```

### Install docker & docker-compose

```
1 #curl -sSL https://get.docker.com | sh
2 #sudo systemctl enable docker
3 #sudo systemctl start docker
4 sudo apt-get install docker docker-compose
5 sudo usermod -aG docker pi
6 sudo reboot
```

## git repository export and start all container

```
1 cd
2 git clone https://github.com/stormmurdock/fhemdocker.git
3 cd fhemdocker
4 docker-compose up
```

## Access the application

### FHEM

#### FHEM tmux session inside the container

```
fhem
fhema708a9d03d01--$ fhema
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
inform on
update
Executing the update the background.
2020.02.07 20:18:06.686 1 : Downloading https://fhema.de/fhemupdate/controls_fhem.txt
2020.02.07 20:18:06.917 1 : RMDIR: ./restoreDir/update/2020-02-04
2020.02.07 20:18:07.151 1 : UPD FHEM/30_HUEBridge.pm
2020.02.07 20:18:07.212 1 : UPD FHEM/98_structure.pm
2020.02.07 20:18:07.341 1 : saving fhema.cfg
2020.02.07 20:18:07.344 1 : saving ./log/fhem.save
2020.02.07 20:18:07.351 1 : Calling /usr/bin/perl ./contrib/commandref_modular.pl, this may take
a while
2020.02.07 20:18:07.713 1 :
2020.02.07 20:18:07.714 1 : update finished, "shutdown restart" is needed to activate the change
s.
2020.02.07 20:18:07.714 1 : fhema Statistics data sent to server. See Logfile (level 4) for d
etails.
Global global UPDATE
2020.02.07 20:18:17.680 3 : myHmUART: Unknown code A0FD686102E6B7A0000000A88AA0B0900::84:myHmUART
RT, help me!
HmUARTLGW myHmUART UNKNOWNCODE A0FD686102E6B7A0000000A88AA0B0900::84:myHmUART

1 [|||||] 20.3% Tasks: 38, 0 thr: 3 running
2 [|||||] 17.2% Load average: 0.61 0.72 0.69
3 [|||||] 30.1% Uptime: 1 day, 01:17:18
4 [|||||] 16.3%
Mem[|||||] 391M/974M
Swp[|||||] 76.3M/100.0M

PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
18546 fhema 20 0 12364 7668 4072 R 8.7 0.8 0:00.13 /usr/bin/python3 /usr/bin/power
18545 fhema 20 0 11876 7148 3888 R 8.1 0.7 0:00.12 /usr/bin/python3 /usr/bin/power
11217 fhema 20 0 8152 2644 1872 S 2.0 0.3 0:02.44 tmux new-session ; send-keys jo
1 root 20 0 7620 2492 2232 S 1.3 0.2 1:11.34 /bin/bash /entry.sh start
15170 fhema 20 0 2852 1856 1552 S 0.7 0.2 0:00.60 htop
17561 fhema 20 0 2852 1840 1536 R 0.7 0.2 0:00.17 htop
3967 fhema 20 0 67764 58756 5408 S 0.7 5.9 0:20.85 perl fhema.pl fhema.cfg
17113 fhema 20 0 9908 3096 2356 S 0.7 0.3 0:00.09 sshd: fhema@pts/1
11268 fhema 20 0 2720 1800 1496 S 0.0 0.2 0:01.07 htop
3941 root 20 0 8880 2756 2632 S 0.0 0.3 0:00.00 /usr/sbin/sshd
4060 root 20 0 9908 5076 4336 S 0.0 0.5 0:00.03 sshd: fhema [priv]
4077 fhema 20 0 9908 3288 2548 S 0.0 0.3 0:00.02 sshd: fhema@pts/0
4087 fhema 20 0 2688 2124 1804 S 0.0 0.2 0:00.01 -bash
11252 fhema 20 0 2688 2176 1860 S 0.0 0.2 0:00.00 -bash
11254 fhema 20 0 2688 2168 1852 S 0.0 0.2 0:00.01 -bash
11257 fhema 20 0 2688 2128 1864 S 0.0 0.2 0:00.01 -bash
12770 fhema 20 0 2688 2172 1852 S 0.0 0.2 0:00.01 -bash
12784 fhema 20 0 2084 1476 1312 S 0.0 0.1 0:00.00 rlrwrap telnet localhost 7072
F1help F2Setup F3Search F4Filter F5Free F6Sortby F7Vice F8Vice F9Kill F10Quit

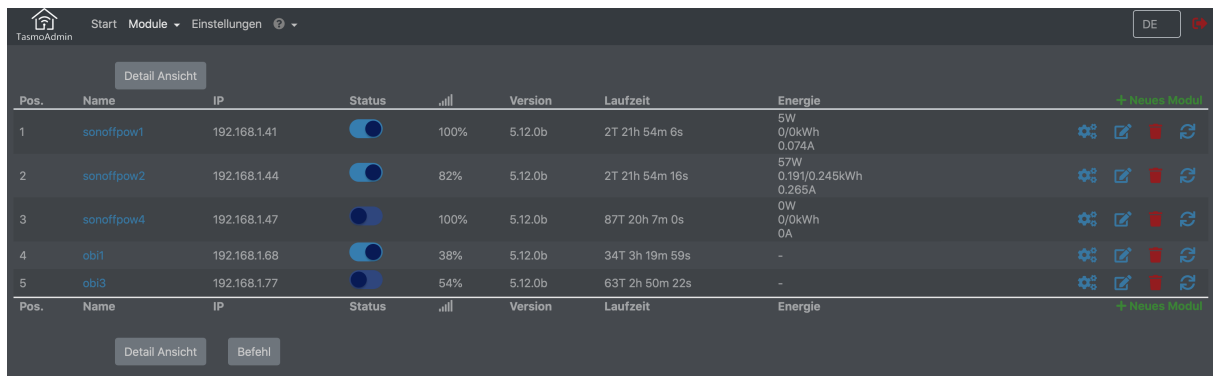
ART, help me!
2020.02.07 20:15:43.152 3 : myHmUART: Unknown code A0C66847030575F00000000D026::83:myHmUART, h
elp me!
2020.02.07 20:16:39.330 3 : myHmUART: Unknown code A0C2086702687CA000000003E64::59:myHmUART, h
elp me!
2020.02.07 20:16:44.320 3 : myHmUART: Unknown code A0CA4865A358B95000000A0C527::84:myHmUART, h
elp me!
2020.02.07 20:17:04.307 3 : myHmUART: Unknown code A0CA48470358B950000000C527::85:myHmUART, h
elp me!
2020.02.07 20:18:06.686 1 : Downloading https://fhema.de/fhemupdate/controls_fhem.txt
2020.02.07 20:18:06.917 1 : RMDIR: ./restoreDir/update/2020-02-04
2020.02.07 20:18:07.151 1 : UPD FHEM/30_HUEBridge.pm
2020.02.07 20:18:07.212 1 : UPD FHEM/98_structure.pm
2020.02.07 20:18:07.341 1 : saving fhema.cfg
2020.02.07 20:18:07.344 1 : saving ./log/fhem.save
2020.02.07 20:18:07.351 1 : Calling /usr/bin/perl ./contrib/commandref_modular.pl, this may tak
e a while
2020.02.07 20:18:07.713 1 :
2020.02.07 20:18:07.714 1 : update finished, "shutdown restart" is needed to activate the chang
es.
2020.02.07 20:18:07.714 1 : fhema Statistics data sent to server. See Logfile (level 4) for
details.
2020.02.07 20:18:17.680 3 : myHmUART: Unknown code A0FD686102E6B7A0000000A88AA0B0900::84:myHmU
ART, help me!
```

Abbildung 1: “fhemt看”

<http://localhost:80>

## Container

### Tasmota Admin

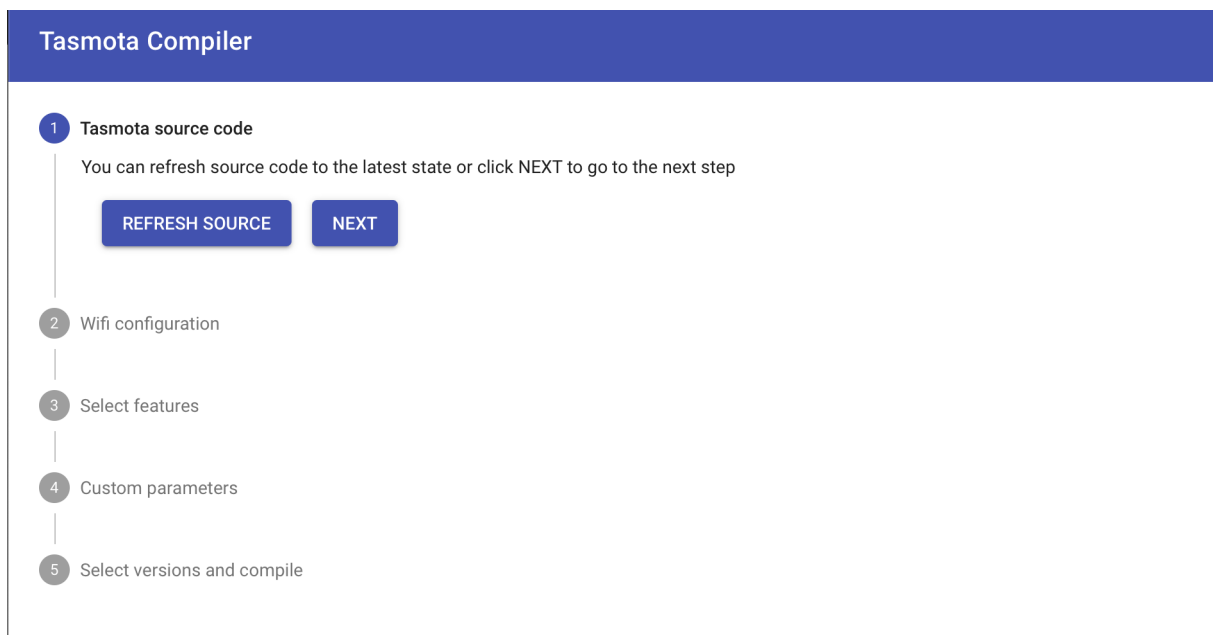


The screenshot shows the Tasmota Admin web interface. At the top, there is a navigation bar with 'Start', 'Module', and 'Einstellungen'. A 'DE' button is in the top right. Below the navigation bar, there is a 'Detail Ansicht' button. The main content is a table with the following columns: Pos., Name, IP, Status, Signal strength, Version, Laufzeit, and Energie. The table lists five modules: sonoffpow1, sonoffpow2, sonoffpow4, obi1, and obi3. Each module row has a status toggle, a signal strength indicator, and a 'Neues Modul' button. Below the table, there are 'Detail Ansicht' and 'Befehl' buttons.

| Pos. | Name       | IP           | Status | Signal | Version | Laufzeit       | Energie                         | Neues Modul |
|------|------------|--------------|--------|--------|---------|----------------|---------------------------------|-------------|
| 1    | sonoffpow1 | 192.168.1.41 | On     | 100%   | 5.12.0b | 2T 21h 54m 6s  | 5W<br>0/0kWh<br>0.074A          | ⚙️ 📄 🛑 ↺    |
| 2    | sonoffpow2 | 192.168.1.44 | On     | 82%    | 5.12.0b | 2T 21h 54m 16s | 57W<br>0.191/0.245kWh<br>0.265A | ⚙️ 📄 🛑 ↺    |
| 3    | sonoffpow4 | 192.168.1.47 | On     | 100%   | 5.12.0b | 87T 20h 7m 0s  | 0W<br>0/0kWh<br>0A              | ⚙️ 📄 🛑 ↺    |
| 4    | obi1       | 192.168.1.68 | On     | 38%    | 5.12.0b | 34T 3h 19m 59s | -                               | ⚙️ 📄 🛑 ↺    |
| 5    | obi3       | 192.168.1.77 | On     | 54%    | 5.12.0b | 63T 2h 50m 22s | -                               | ⚙️ 📄 🛑 ↺    |

Abbildung 2: “tasmotaadmin”

### Tasmota Compiler



The screenshot shows the Tasmota Compiler web interface. It has a blue header with the title 'Tasmota Compiler'. Below the header, there is a vertical list of steps: 1. Tasmota source code, 2. Wifi configuration, 3. Select features, 4. Custom parameters, and 5. Select versions and compile. Step 1 is currently selected. Below step 1, there is a text prompt: 'You can refresh source code to the latest state or click NEXT to go to the next step'. Below this text are two buttons: 'REFRESH SOURCE' and 'NEXT'.

**Tasmota Compiler**

- 1 Tasmota source code**  
You can refresh source code to the latest state or click NEXT to go to the next step  
[REFRESH SOURCE](#) [NEXT](#)
- 2 Wifi configuration
- 3 Select features
- 4 Custom parameters
- 5 Select versions and compile

Abbildung 3: “tasmotacompiler”

## Homebridge

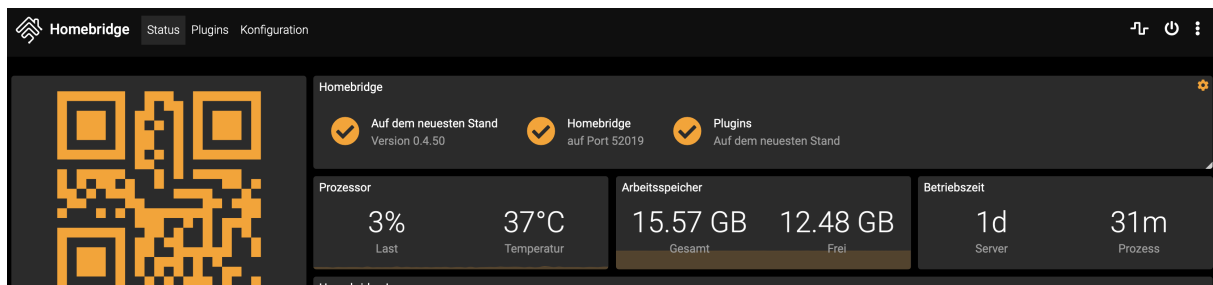


Abbildung 4: “homebridge”

## Portainer

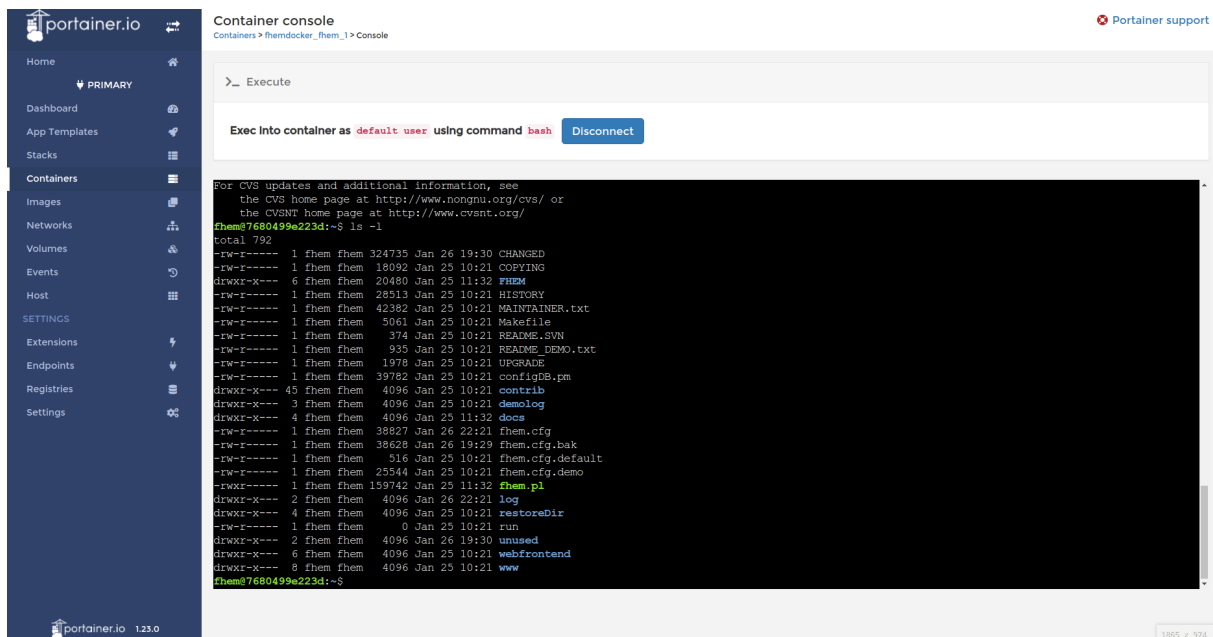


Abbildung 5: “portainer”

## Deconz

### deCONZ Image Container Integration

---

## Configuring Raspbian for RaspBee

Raspbian defaults Bluetooth to /dev/ttyAMA0 and configures a login shell over serial (tty). You must disable the tty login shell and enable the serial port hardware, and swap Bluetooth to /dev/S0, to allow RaspBee to work properly under Docker.

To disable the login shell over serial and enable the serial port hardware:

- 1) sudo raspi-config
- 2) Select Interfacing Options
- 3) Select Serial
- 4) “Would you like a login shell to be accessible over serial?” Select No
- 5) “Would you like the serial port hardware to be enabled?” Select Yes
- 6) Exit raspi-config and reboot To swap Bluetooth to /dev/S0 (moving RaspBee to /dev/ttyAMA0), run the following command and then reboot:

```
echo 'dtoverlay=miniuart-bt' | sudo tee -a /boot/config.txt
```

This will exchange the UART and the Mini-UART so the Mini-UART is connected to the bluetooth and the UART to the GPIO pins.

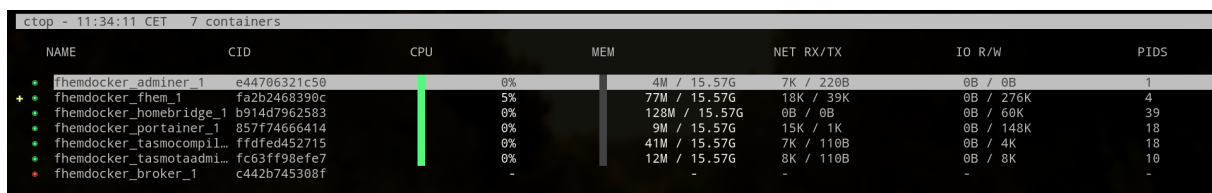
On Raspberry Pi 4 verify that file /boot/config.txt does NOT contain a line “enable\_uart=0”. If the line exists remove or comment (#) this line.

After running the above command and rebooting, RaspBee should be available at /dev/ttyAMA0.

## ctop

### Description

ctop is a commandline monitoring tool for linux containers



The screenshot shows the output of the 'ctop' command, which displays real-time statistics for 7 containers. The table includes columns for NAME, CID, CPU usage, MEM usage, NET RX/TX, IO R/W, and PIDS. The first container, 'fhemdocker\_adminer\_1', is highlighted with a green bar in the CPU column.

| NAME                          | CID          | CPU | MEM           | NET RX/TX | IO R/W    | PIDS |
|-------------------------------|--------------|-----|---------------|-----------|-----------|------|
| * fhemdocker_adminer_1        | e44706321c50 | 0%  | 4M / 15.57G   | 7K / 220B | 0B / 0B   | 1    |
| * fhemdocker_fhem_1           | fa2b2468390c | 5%  | 77M / 15.57G  | 18K / 39K | 0B / 276K | 4    |
| * fhemdocker_homebridge_1     | b914d7962583 | 0%  | 128M / 15.57G | 0B / 0B   | 0B / 60K  | 39   |
| * fhemdocker_portainer_1      | 857f74666414 | 0%  | 9M / 15.57G   | 15K / 1K  | 0B / 148K | 18   |
| * fhemdocker_tasmocompil...   | ffdfed452715 | 0%  | 41M / 15.57G  | 7K / 110B | 0B / 4K   | 18   |
| * fhemdocker_tasmotaadmini... | fc63ff98efe7 | 0%  | 12M / 15.57G  | 8K / 110B | 0B / 8K   | 10   |
| * fhemdocker_broker_1         | c442b745308f | -   | -             | -         | -         | -    |

Abbildung 6: “ctop”

---

## Installation

ctop is available in AUR, so you can install it using AUR helpers, such as YaY, in Arch Linux and its variants such as Antergos and Manjaro Linux.

## Installation Linux

```
1 sudo wget https://github.com/bcicen/ctop/releases/download/v0.7.3/
 ctop-0.7.3-linux-amd64 -O /usr/local/bin/ctop
2 sudo chmod +x /usr/local/bin/ctop
```

```
1 sudo wget https://github.com/bcicen/ctop/releases/download/v0.7.3/
 ctop-0.7.3-linux-arm -O /usr/local/bin/ctop
2 sudo chmod +x /usr/local/bin/ctop
```

## Contributing to fhemdoker

Contributions are encouraged and welcome!