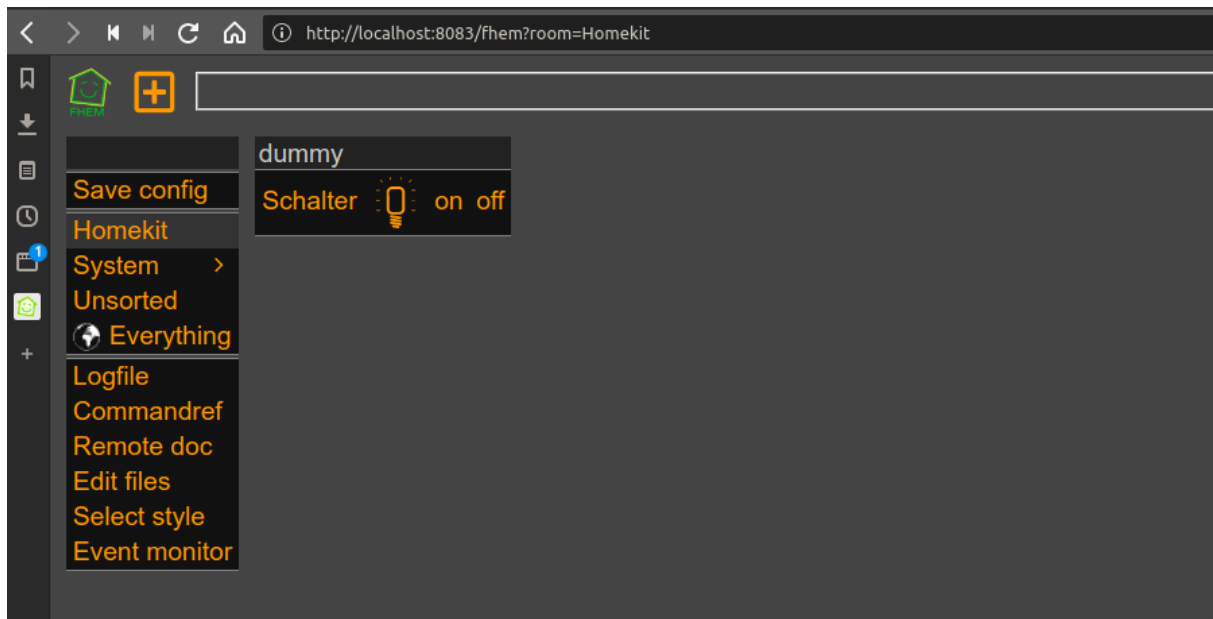

Home Automation Stack



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

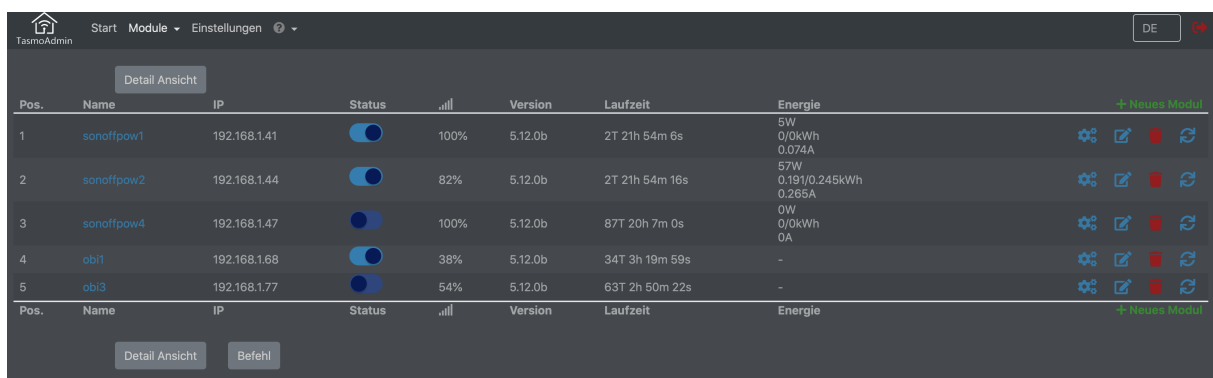
```
1 cd
2 git clone https://github.com/stormmurdoc/fhemdocker.git
3 cd fhemdocker
4 docker-compose up
```

Access the application

```
1 http://localhost:80
```

Container

Tasmota Admin



The screenshot shows the Tasmota Admin web interface. At the top, there is a navigation bar with a home icon, the text 'TasmotaAdmin', and a menu with 'Start', 'Module', 'Einstellungen', and a user icon. On the right of the navigation bar are buttons for 'DE' and a red star icon. Below the navigation bar is a 'Detail Ansicht' button. The main content area displays a table of modules. The table has columns for 'Pos.', 'Name', 'IP', 'Status', a signal strength icon, 'Version', 'Laufzeit', and 'Energie'. There are five rows of modules, each with a set of control icons on the right. Below the table are two buttons: 'Detail Ansicht' and 'Befehl'.

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

Abbildung 1: "tasmotaadmin"

Tasmota Compiler

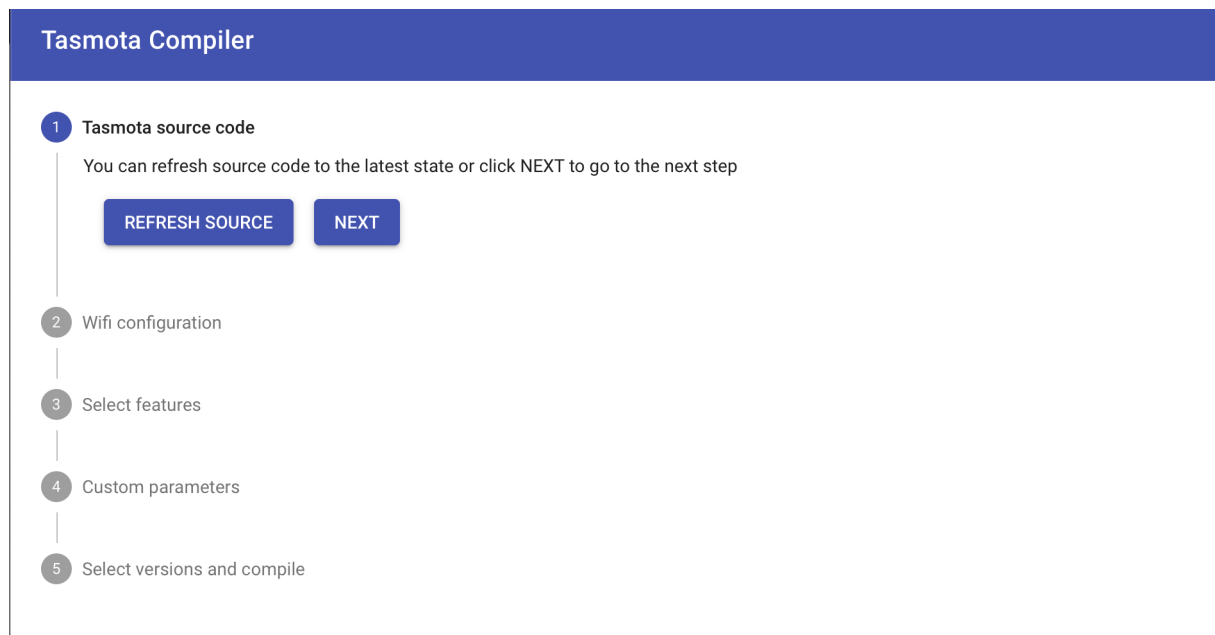


Abbildung 2: "tasmotacompiler"

Homebridge

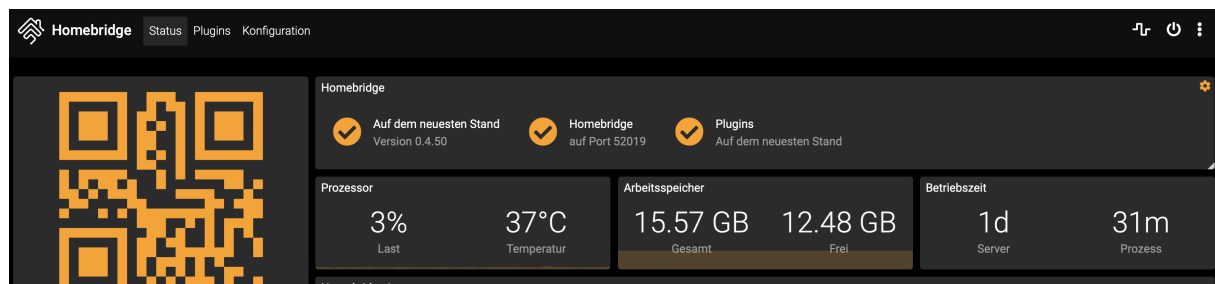


Abbildung 3: "homebridge"

Portainer

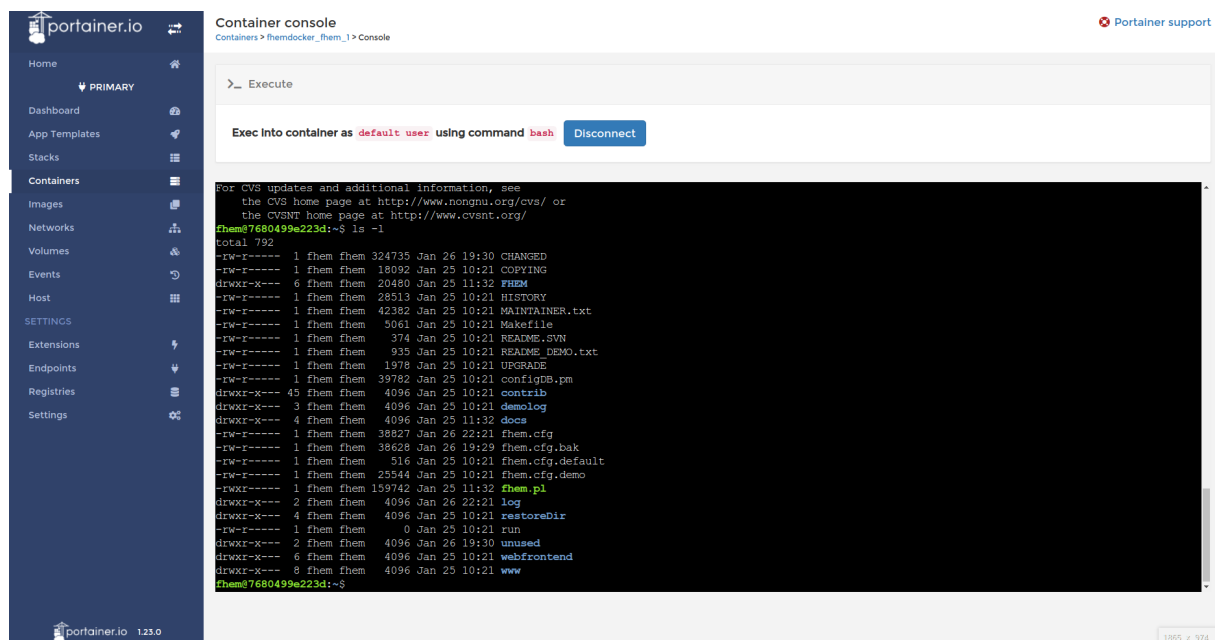


Abbildung 4: “portainer”

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.

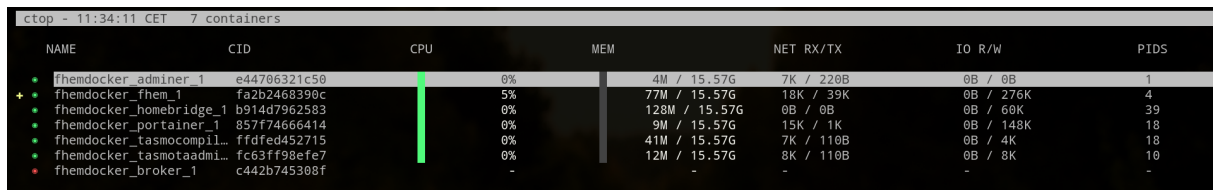
On Raspberry Pi 4 verify that file /boot/config.txt does NOT contain a line “enable_uart=0”. If the line exists remove or comment (#) this line.

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 ctop interface with a title bar indicating 'ctop - 11:34:11 CET 7 containers'. The main display is a table with columns: NAME, CID, CPU, MEM, NET RX/TX, IO R/W, and PIDS. The CPU column features a green progress bar. The data is as follows:

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_tasmotaadmi...	fc63ff98efe7	0%	12M / 15.57G	8K / 110B	0B / 8K	10
fhemdocker_broker_1	c442b745308f	-	-	-	-	-

Abbildung 5: “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!