

## Image auf SD

```
diskutil list
```

```
sudo diskutil unmount /dev/disk2s1
```

```
sudo dd bs=1m if=~/.Backups/Raspberry-Pi/2016-03-18-raspbian-jessie.img of=/dev/disk2
```

**oder besser Lite Version verwenden, wenn kein Monitor Anschluss am Pi benötigt wird.**

```
sudo dd bs=1m if=~/.Backups/Raspberry-Pi/2016-03-18-raspbian-jessie-lite.img of=/dev/disk2
```

```
sudo diskutil eject /dev/rdisk2
```

## SD Karte im Raspi booten

```
sudo raspi-config      ausführen
```

---

## Static IP

Entweder IP im Router über MAC Adresse setzen.

oder

```
sudo nano /etc/dhcpd.conf
```

und am Ende der Datei mit euren IP's einfügen:

```
interface eth0
static ip_address=192.168.xxx.xxx/24
static routers=192.168.xxx.x
static domain_name_servers=192.168.xxx.x
```

---

---

## Weiter mit dem Terminal vom Imac

```
sudo nano /boot/config.txt
```

```
dtoverlay=w1-gpio          am Ende einfügen
```

```
sudo nano /etc/modules
```

```
w1-gpio pullup=1  
w1-therm
```

## root Password ändern optional

```
sudo passwd root
```

## eventuell sftp für root freigeben optional

```
sudo nano /etc/ssh/sshd_config
```

nach

```
PermitRootLogin
```

suchen und auf yes setzen

## Installation

```
~ $ sudo apt-get update
```

```
~ $ sudo apt-get upgrade -y
```

```
~ $ sudo apt-get install git          > im Falle von Jessie  
light
```

```
~ $ sudo apt-get install php5-common php5-cgi php5-cli  
php-apc
```

```
~ $ sudo apt-get install apache2 libapache2-mod-php5    ≥  
kann entfallen bei einer SHC Slave Installation
```

Anschließend die Datei öffnen und in Zeile 12 das html

nach /var/www/ löschen.

```
sudo nano /etc/apache2/sites-enabled/000-default.conf
```

Wenn Jessie light dann Java installieren Wird nur auf dem Master benötigt!

```
sudo apt-get install openjdk-8-jre
```

### Neustart

Für die Installation eines Slave's also nur als Schaltserver und Sensorsender entfällt der nächste Schritt.

### Installation Reds und PHPRedis

```
~ $ sudo apt-get install redis-server php5-redis
```

---

Den nächsten Schritt besser erst ausführen nach kompletter Installation !!!

Bei einer Slave Installation entfällt dieser Schritt komplett!

Folgende Zeile in die /etc/fstab Datei eingetragen:

Achtung: Will man den shc Folder einmal mit: `sudo rm -R shc` löschen, muss die Zeile vorher in der fstab auskommentiert werden! Sonst ist kein vollständiges löschen möglich!

### Hinweis:

Wird dieser Schritt vor der eigentlichen Installation des SHC ausgeführt, kam es vor das unter Jessie Dateien später im shc Ordner fehlten. So das ich dazu übergegangen bin, diesen Schritt erst nach der SHC Installation auszuführen. Der Zusammenhang ist mir nicht bekannt.

Ferner muss auch der Eintrag mit # auszementiert werden wenn:

Der shc Ordner gelöscht werden soll, oder ein restore aus

einem Backup erfolgen soll. Danach immer Neustart, damit die Änderungen übernommen werden.

```
sudo nano /etc/fstab
```

```
tmpfs          /var/www/shc/rwf/data/cache    tmpfs
nosuid,size=5M      0          0
```

Letzte Zeile Leerzeile  
Neustart !!!

---

Neustart!!

**A) Installation als Master. Als reiner Slave überspringen, weiter mit (B).**

```
cd /var/www
/var/www $ sudo git clone https://github.com/agent4788/SHC_Framework
/var/www $ sudo mv SHC_Framework/ shc/
/var/www $ sudo chmod 0777 -R shc/
/var/www $ cd shc/
/var/www/shc $ php install.php Hier können alle Fragen mit Return quittiert werden.
/var/www/shc $ sudo chmod 0777 -R ./
```

**B) Installation als reiner Slave ohne Apache.**

```
cd /var
sudo mkdir www
```

```
cd /var/www
/var/www $ sudo git clone https://github.com/agent4788/SHC_Framework
/var/www $ sudo mv SHC_Framework/ shc/
/var/www $ sudo chmod 0777 -R shc/
/var/www $ cd shc/
/var/www/shc $ sudo chmod 0777 -R ./
```

**Nur am Master mit Apache.**

Nach diesen Schritten ist die Weboberfläche einsatzbereit und sollte unter <http://localhost/shc/index.php?app=shc>

erreichbar sein. Doppelpunkt Port nicht vergessen, wenn er geändert wurde!

Mit dem Benutzernamen "admin" und dem Passwort "admin" kann man sich am SHC anmelden.

```
/var/www/shc $ php index.php app=shc -sh -c > Sheduler konfigurieren nur beim Master
```

```
/var/www/shc $ php index.php app=shc -ss -c > Bei reinem Master ohne GPIO Nutzung wird dieser nicht benötigt.
```

### Dienste anzeigen

```
php index.php app=shc -ds
```

```
sudo service shcd stop
```

```
sudo service shcd start
```

Info: Der Schaltserver wird solange als deaktiviert angezeigt, bis er auch in der Weboberfläche konfiguriert wurde. Nicht vergessen!

### wiringPi Installieren

```
~ $  
~ $ git clone git://git.drogon.net/wiringPi  
~ $ cd wiringPi  
~ $ git pull origin  
~ $ ./build
```

### rcswitch-Pi installieren

```
~ $ cd /opt  
/opt $ sudo git clone https://github.com/r10r/rcswitch-pi.git  
/opt $ cd rcswitch-pi  
/opt/rcswitch-pi $ sudo make
```

### Test

Beispiel: Hauscode 10001 Lampe 1 An

```
$ sudo /opt/rcswitch-pi/send 10001 1 1
```

---

## Pilight Installieren wenn benötigt optional!

```
sudo su
```

```
sudo nano /etc/apt/sources.list
```

### Einfügen

```
deb http://apt.pilight.org/ stable main
```

```
root@pilight:~# wget -O - http://apt.pilight.org/  
pilight.key | apt-key add -  
root@pilight:~# apt-get update  
root@pilight:~# apt-get install pilight  
root@pilight:~# service pilight start  
root@pilight:~# service pilight stop
```

```
root@pilight:~# su pi
```

### Pilight Webserver disable

Zusätzlich kann in der /etc/pilight/config.json Datei die Weboberfläche von Pilight deaktiviert werden, dieser wird vom SHC geliefert.

```
"webserver-enable": 0
```

---

### sudo visudo

Nur die benutzten Anwendungen eintragen!

```
www-data ALL = NOPASSWD: /opt/rcswitch-pi/send  
www-data ALL = NOPASSWD: /sbin/reboot, /sbin/halt  
www-data ALL = NOPASSWD: /var/www/shc/shc/lib/external/  
python/dht.py
```

```
www-data ALL = NOPASSWD: /var/www/shc/shc/lib/external/  
python/bmp.py  
www-data ALL = NOPASSWD: /usr/local/bin/pilight-send
```

### Test

```
~ $ cd /var/www/shc  
/var/www/shc $ php index.php app=shc -ss
```

### Autostart der Dienste einrichten

```
/var/www/shc $ sudo mv shcd.sh /etc/init.d/shcd  
/var/www/shc $ sudo update-rc.d shcd defaults
```

### DS18x20

Auslesen siehe SHC Wiki Troubleshooting

### BMP085

In der raspi-config I2C Bus einschalten!!!!

```
git clone https://github.com/adafruit/Adafruit_Python_BMP  
cd Adafruit_Python_BMP  
$ sudo python setup.py install
```

```
sudo nano /etc/modules
```

```
i2c-bcm2708
```

```
i2c-dev
```

In /etc/modprobe.d/raspi-blacklist.conf sind die *blacklist* Einträge zu entfernen oder auszukommentieren, falls vorhanden:

```
#blacklist spi-bcm2708
```

```
#blacklist i2c-bcm2708
```

```
sudo apt-get install python-smbus i2c-tools
```

```
sudo adduser pi i2c
```

```
i2cdetect -y 1
```

### Address 77 sollte jetzt angezeigt werden

### Test BMP auslesen

```
sudo python /var/www/shc/shc/lib/external/python/bmp.py
```

### BMP im Sensortransmitter aktivieren

```
/var/www/shc $ php index.php app=shc -st -enableBMP  
77
```

### DHT22 Installieren

```
~ $ sudo apt-get install build-essential python-dev  
~ $ git clone https://github.com/adafruit/  
Adafruit_Python_DHT  
~ $ cd Adafruit_Python_DHT  
~/Adafruit_Python_DHT $ sudo python setup.py install
```

```
/var/www/shc $ php index.php app=shc -st -c
```

### DHT22 anmelden

```
/var/www/shc $ php index.php app=shc -st -addDHT <id  
(0-999)> <typ (11|22|2302)> <pin (BCM Pin Nummer)>
```

### Beispiel 2 DHT22 einer mit Id22 und einer mit Id 23 Pin 22 und Pin 4

```
cd /var/www/shc  
php index.php app=shc -st -addDHT 22 22 22  
php index.php app=shc -st -addDHT 23 22 4
```

```
php index.php app=shc -st -listDHT
```

```
php index.php app=shc -st -removeDHT XX  
ID
```

### DHT22 auslesen



Beispiel wenn als Id 22 und Pin 4 gewählt wurde

Typ Pin

```
sudo python /var/www/shc/shc/lib/external/python/dht.py 22  
4
```

### Sensortransmitter aktivieren

```
/var/www/shc $ php index.php app=shc -st -c Bei einem  
Slave wird die IP des Masters eingetragen.
```

Bei einem oder mehreren Slave's darauf achten das die  
Sensor ID nur einmal im gesamten System vorkommt.

Ansonsten nur mit Return bestätigen.

### SHC Update

```
sudo service shcd stop
```

```
/var/www/shc $ git pull  
/var/www/shc $ sudo chmod 0777 -R ./  
/var/www/shc $ php update.php
```

```
sudo service shcd start
```

### Restore vom SHC Backup

Ramdisk ausschalten. Dazu den tmpfs Eintrag in der etc/  
fstab mit # auskommentieren speichern und Neustart.

Dies wird im Terminal mit der backup\_restore.php gemacht.

```
/var/www/shc $ sudo service shcd stop
```

```
/var/www/shc $ php backup_restore.php
```

```
/var/www $ sudo chmod 0777 -R shc/
```

```
/var/www/shc $ sudo chmod 0777 -R shc/
```

```
/var/www/shc $ sudo service shcd start
```

Ramdisk wieder einschalten. Dazu den tmpfs Eintrag in der etc/fstab das # entfernen speichern und Neustart.

### Backup SD Karte

```
diskutil list          device checken, hier dev disk2
```

```
sudo dd if=/dev/rdisk2 of=~/.Backups/Raspberry-Pi/RaspiTest.img bs=1m
```

```
sudo dd if=/dev/rdisk2 of=~/.Backups/Raspberry-Pi/SHC227_Master.img bs=1m
```

```
sudo dd if=/dev/rdisk2 of=~/.Backups/Raspberry-Pi/SHC224_Slave_Wintergarten.img bs=1m
```

```
sudo dd if=/dev/rdisk2 of=~/.Backups/Raspberry-Pi/SHC227_Slave_Flur.img bs=1m
```

```
sudo diskutil eject /dev/rdisk2
```

### Restore

```
sudo diskutil unmount /dev/disk2s1
```

```
sudo dd bs=1m if=~/.Backups/Raspberry-Pi/RaspiTest.img of=/dev/disk2
```

```
sudo dd bs=1m if=~/.Backups/Raspberry-Pi/SHC224_Slave_Wintergarten.img of=/dev/disk2
```

```
sudo dd bs=1m if=~/.Backups/Raspberry-Pi/SHC227_Slave_Flur.img of=/dev/disk2
```

```
sudo dd bs=1m if=~/.Backups/Raspberry-Pi/
```

```
SHC226_Master_1.img of=/dev/disk2
```

```
sudo diskutil eject /dev/rdisk2
```

### Log Dateien

```
cd /var/www/shc/rwf/data/log
```

```
error.log und exception.log
```

Zum Löschen der Inhalte

```
> datei.log
```

### „TASTER“ Countdown weniger als 30 Sekunden bei Bedarf.

```
sudo nano /var/www/shc/shc/lib/form/forms/elements/  
countdownform.class.php
```

```
/var/www/shc/shc/lib/form/forms/elements $ sudo nano  
countdownform.class.php
```

Zeile 74

```
$min = 2
```

Zeile 90 suchen und mit folgendem Code ersetzen:

```
$interval = new IntegerInputField('interval',  
($countdown instanceof Countdown ? $countdown-  
>getInterval() : 0), array('min' => $min, 'max' =>  
14400, 'step' => 1));
```

Habe mal ein Countdown mit 1 Sekunde erstellt. Sind bei mir aber paar Sekunden länger als eine.