

# **Setup Orange Pi Zero - H2 for CLEES**

Edition: 2019-01-20

by  
Tomba

## Table of Contents

Introduktion.....	3
Om CLEES.....	3
Akronymer.....	3
Steg för steg installation.....	3
1. Förbered SDkortet.....	4
1.1 Ladda ned Mainline.....	4
1.2 Packa upp filen.....	4
1.3 Sätt SD-card i datorn.....	4
1.4 Skriv diskavbildningen till SD-card:et.....	4
2. Starta Opi0 för första gången.....	5
2.1 Sätt i SD-card:et i Opi0.....	5
2.2 Anslut nätverk och matningsspänning.....	6
2.3 Kontakta Opi0.....	6
2.4 Logga in på Opi0 första gången.....	7
2.5 Boota om.....	8
2.6 Filhantering via SFTP.....	8
2.7 Grundinställningar i Armbian.....	9
2.8 Stäng av Opi0 kontrollerat med "halt".....	9
3 Förbereda hårdvaran för CLEES.....	10
3.1 Löd på en stiftlist.....	10
3.2 Aktivera i2c.....	10
3.3 Testa i2c.....	11
3.4 Installera drivrutin för PWM-kortet PCA9685.....	11
3.5 Testa PCA9685 med ett RC-servo.....	12
3.6 Installera smbus.....	12
3.7 Installera OPi.GPIO.....	13
4 Installera CLEES.....	14
4.1 Installera CLEES mjukvara.....	14
4.2 Installera MQTT driver.....	14
4.3 Sätt lämpligt hostname.....	14
4.4 Byt till fast IPadress.....	15
4.5 Märk dina dоторer med etikett.....	15
5. Andra installationer och inställningar.....	16
5.1 Aktivera Wifi och en brygga.....	16
5.2 Installera och Aktivera DHCP.....	17
5.3 Installera och aktiver MQTT.....	18
5.4 Aktivera MQTT Websocket.....	19
5.5 Autostart av CLEES.....	19

## Introduktion

Denna guide förklarar hur du förbereder och kör igång en Orange Pi Zero enkortsdator i allmänhet och hur du sätter upp den för att fungera som styrenhet och dekoder i ditt CLEES system.

## Om CLEES

CLEES är en akronym för "Controll your Layout using Ethernet and Easy Scripts" och är ett öppet system för att styra växlar och signaler på din modelljärnväg.

## Akronymer

Opi0            Orange Pi Zero

## Steg för steg installation

Innan du börjar installationen av mjukvara på Opi0 behöver du installera några program på din dator. Om du redan har dessa program eller motsvarande så kan du hoppa vidare till nästa steg

1. Programmet 7-zip, som används för att packa upp filer du laddar ned från internet. 7-zip hittar du här:  
<http://www.7-zip.org/>
2. Programmet Win32 Disk Imager, som används för att läsa och skiva avbilder av SD-cards. Win32 Disk Imager hittar du här:  
<https://sourceforge.net/projects/win32diskimager/>
3. Programmet SD memory card Formatter, som används för att formatera SD-cards. SD memory card Formatter hittar du här:  
[https://www.sdcard.org/downloads/formatter\\_4/](https://www.sdcard.org/downloads/formatter_4/)
4. Programmet Bitvise, som är ett integrerat program med SSH terminal och SFTP klient för att logga in på och kommunicera med Orange Pi Zero. Bitvise hittar du här:  
<https://www.bitvise.com/ssh-client-download>

Alla program här listade är fria att använda (iallafall när detta skrevs).

## 1. Förbered SDkortet

Opi0 ska laddas med Linuxdistributionen Armbian. Armbian kan köras på många olika enkorts datorer. Du ska välja att ladda ned den som är till "Orange Pi Zero"

<https://www.armbian.com/orange-pi-zero/>

### 1.1 Ladda ned Mainline.<sup>1</sup>

### 1.2 Packa upp filen

Packa upp den nedladdade filen med 7-zip till en temporär katalog.

### 1.3 Sätt SD-card i datorn

Opi0 vill ha ett micro SD-card så du kan behöva en adapter till kortet för att sätta det i datorn.



Du skall välja ett SD-card med speedclass = 10 eller bättre.<sup>2</sup>

### 1.4 Skriv diskavbildningen till SD-card:et

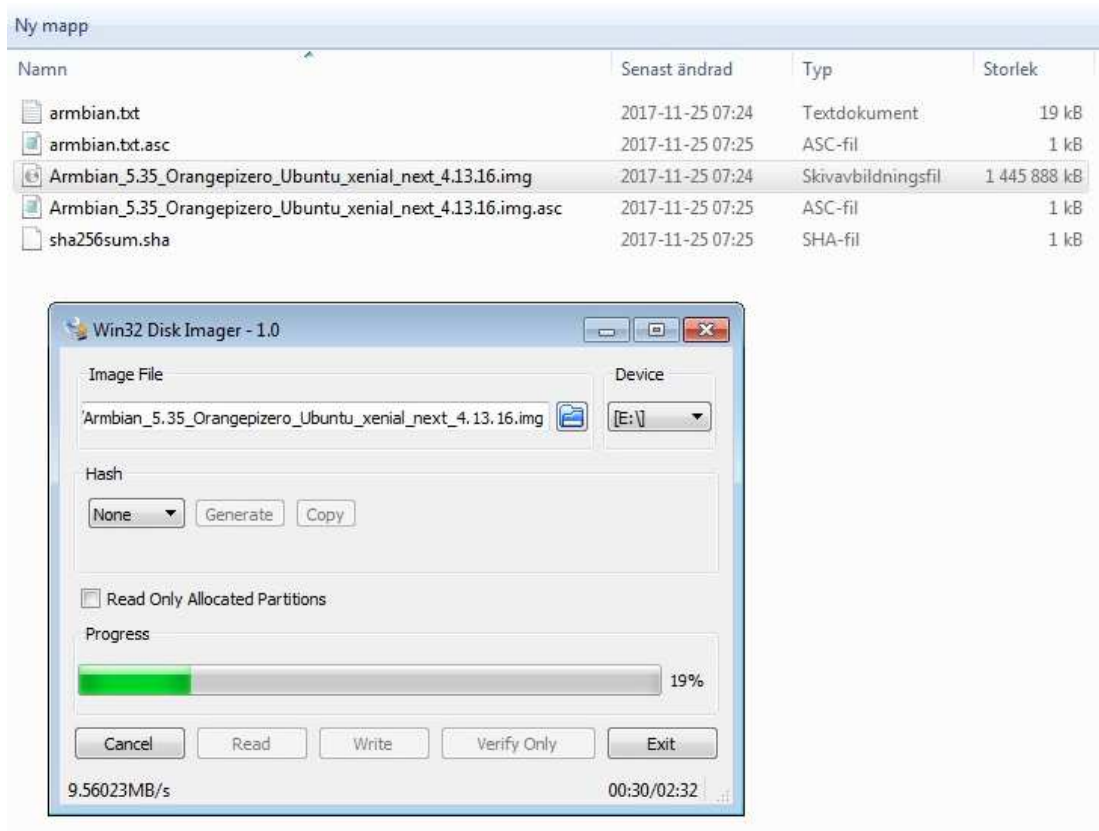
Använd programmet Win32 Disk Imager för att skriva ned Armbianfilen med efternamn .img till SD-card:et. I Win32 Disk Imager programmet väljer du först img fil och sedan klickar du på "Write" knappen så börja programmet skriva Armbianfilen till disken.

Allt som finns på SD-card:et kommer skrivas över. Du behöver normalt inte först formatera SD-card:et om det är ett nytt kort. Men om det skulle behövas så kan du använda programmet SD memory card Formatter.

1 Även om det är vanligt att välja den stabila versionen så är det så i Armbians fall att alla beskrivningar på nätet hänvisar till Mainline och flera inställningar i CLEES beskrivningar går inte att göra i den stabila versionen. Båda versionerna anses stabila men supporten varierar.

2 Ett tips är att använda ett SDcard med så lite minne som möjligt. Gärna 8GB om du kan hitta. Det underlättar när man ska göra backupper av hela SD-disken om kortet är litet. Stora kort kan ta timmar att avbilda.

När filen skrivs till disken kan du se skrivhastigheten nedtill i programmet. Den ska vara 10MB/s eller högre. Då vet du att ditt SD-card är class 10 eller bättre



## 2. Starta Opi0 för första gången

### 2.1 Sätt i SD-card:et i Opi0



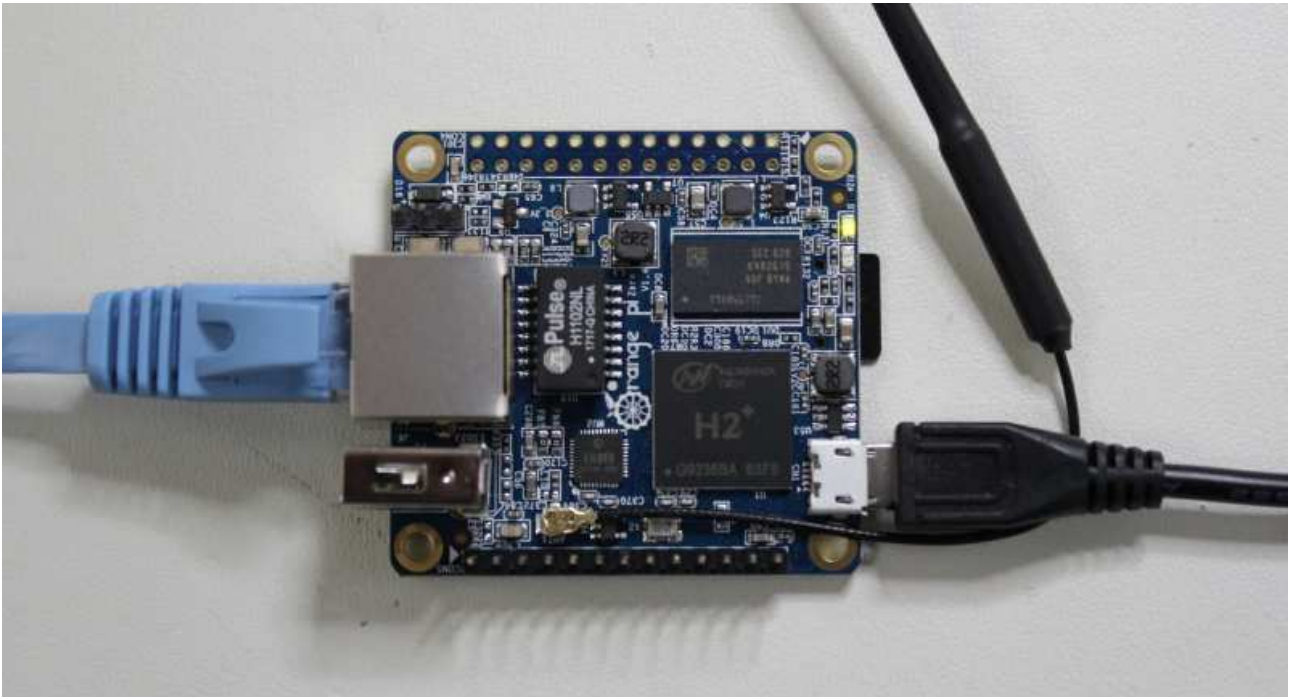
## 2.2 Anslut nätverk och matningsspänning

Koppla in Opi0 till ditt nätverk. Nätverket måste ha en DHCP server/router som kan dela ut IP-adresser dynamiskt. Alla vanliga nätverk kan detta och har detta aktiverat.

Så snart du kopplar ström till Opi0 kommer den automatiskt att börja installationen som tar ca 35 sekunder. STÄNG INTE AV Opi0 UNDER TIDEN DEN STARTAR UPP!

Förloppet under denna tid kan varieras men följande brukar hända:

- 0 sek – Du ansluter strömmen till Opi0
- 3 sek – Gul LED i Etherneuttagen blinkar något
- 5 sek – Både gul och grön LED i ethernetuttaget blinkar till.
- 10 sek – Den lilla gröna LEDen på kortet tänds
- 15-20 sek - Både gul och grön LED i ethernetuttaget blinkar.
- 30 sek – Opi0 gör en DHCP förfrågan på nätverket.
- 35 sek – Nu kan du ansluta med SSH till Opi0



## 2.3 Kontakta Opi0

Opi0 har fått en IP-adres från din DHCP server. Du behöver Opi0's IP-adress för att kunna logga in på Opi0. För att få reda på vilken adress Opi0 har kan du logga in på din router och kolla i listan över IP-adresser som är utdelade.<sup>3</sup>

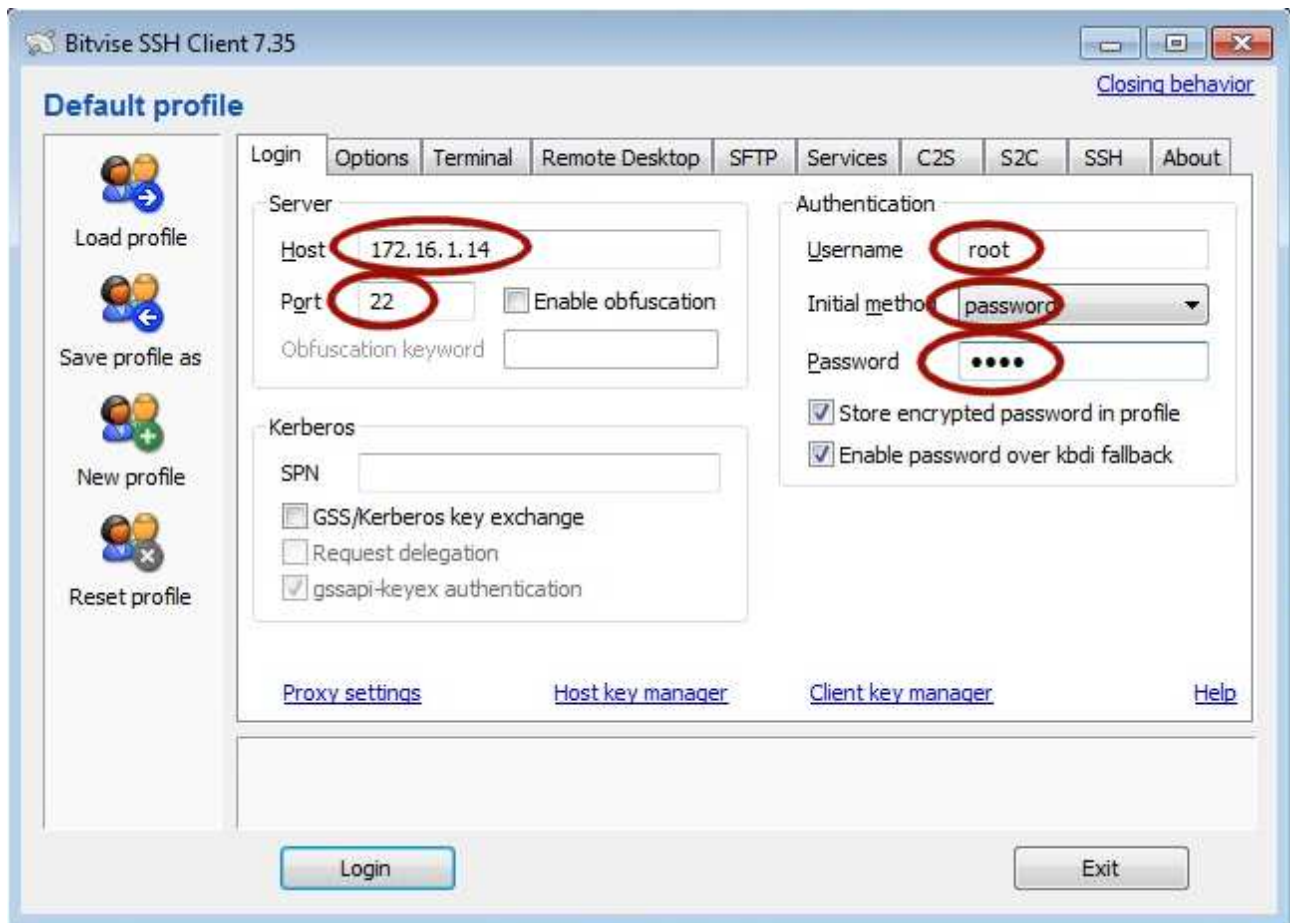
För att logga in på Opi0 används bäst programmet Bitvise. Vill du inte använda Bitvise så fungerar andra också bra, tex SSH klient typ Putty och SFTP via WinSCP.

Fortsättningsvis kommer anslutning till Opi0 beskrivas med programmet Bitvise.

<sup>3</sup> Du kan också använda ett program som LANeye som övervakar nätverket och upptäcker när någon ansluter till nätverket. Då syns direkt att Opi0 har anslutit och vilken IP-adress den har.

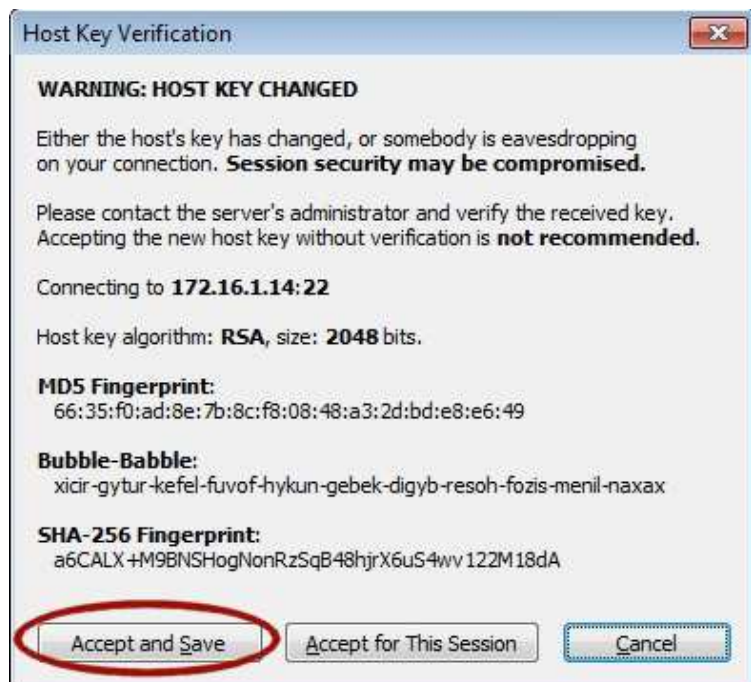
## 2.4 Logga in på Opi0 första gången

Starta Bitvise och ange IP-adress, Port=22, username = root och Password = 1234, allt enligt bilden nedan



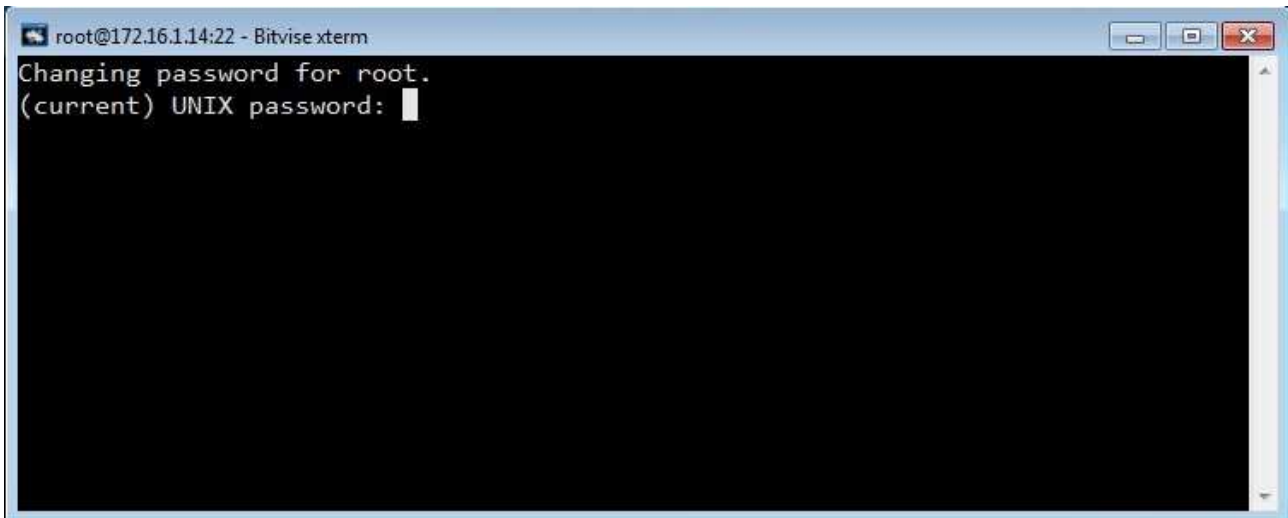
Du kommer få en varning som denna men det är inte så allvarligt Det beror på att Bitvise och Opi0:an inte har träffats förut.

Acceptera och spara





Första gången du ansluter till Opi0:an så tvingas du byta lösenord på root kontot.



Skriv upp lösenordet så du inte glömmer bort det. Tappar du bort det så kan du behöva installera om alltihop från början igen.

Du kommer uppmanas göra ett användarkonto. Gör det genom att följa instruktionerna på skärmen.

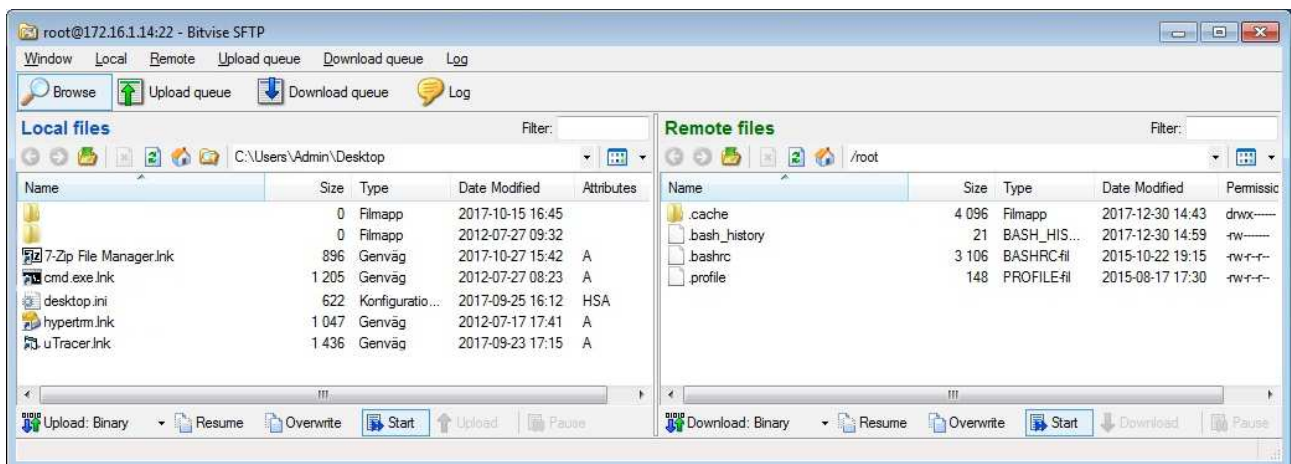
## 2.5 Boota om

Avsluta din första session med att skriva ordet "reboot"

Nu kopplar Opi0:an ned förbindelsen, blinkar med sina LEDar och återkommer efter ca 15-20 sekunder. Nu kommer Bitvise fråga efter lösenordet igen för det har ju ändrats. Skriv in ditt nya rootlösenord och du kommer in.<sup>4</sup>

## 2.6 Filhantering via SFTP

Bitvise öppnar ett FTP-fönster. Där kan du kopiera och flytta/skapa filer på Opi0:ans filsystem.

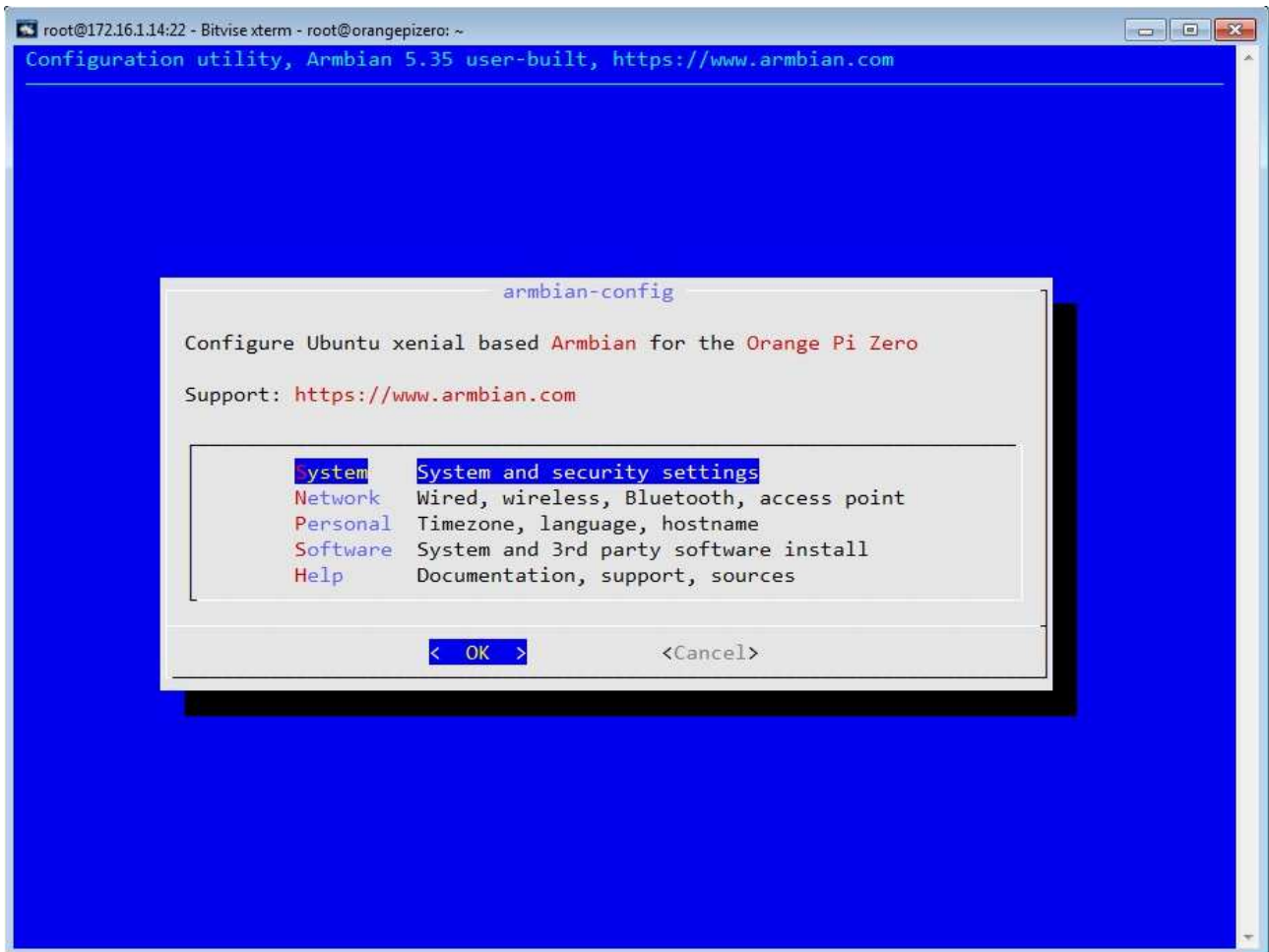


<sup>4</sup> Du kan spara dina konstellningar i Bitvise genom att spara en profil. Gör en profil för root och en annan profil för ditt konto du skapade nyss.



## 2.7 Grundinställningar i Armbian

Följande grundinställningar i Armbian bör man göra. För det använder du det inbyggda config-stödet. Logga in som root och skriv "armbian-config" och configverktygen startar.



Gör dig familjär med detta verktyg för du kommer behöva göra flera inställningar längre fram.

Nu ska du ändra

Tidzon till Europa och Stockholm (eller annan världsdel/stad där du befinner dig).

## 2.8 Stäng av Opi0 kontrollerat med "halt"

En linuxdator är alltid lite känslig för abrupt avstängning. Du ska alltid innan du tar bort strömmen först ge kommandot "halt". Om du inte är inloggad som root kan du behöva skriva "sudo halt".

Om du glömmer ge kommandot "halt" finns risk för att filsystemet på SD-card:et blir skadat.

### 3 Förbereda hårdvaran för CLEES

CLEES använder i/o-pinnar för att kommunicera med annan hårdvara.

#### 3.1 Löd på en stiftlist

Opi0 kommer med en 26polig anslutning för att kommunicera med annan hårdvara via i2c eller GPIO. Men Opi0 kortet saknar stiftlist så en sådan behöver du löda dit.

När du lött dit den ska det se ut som på bilden intill.



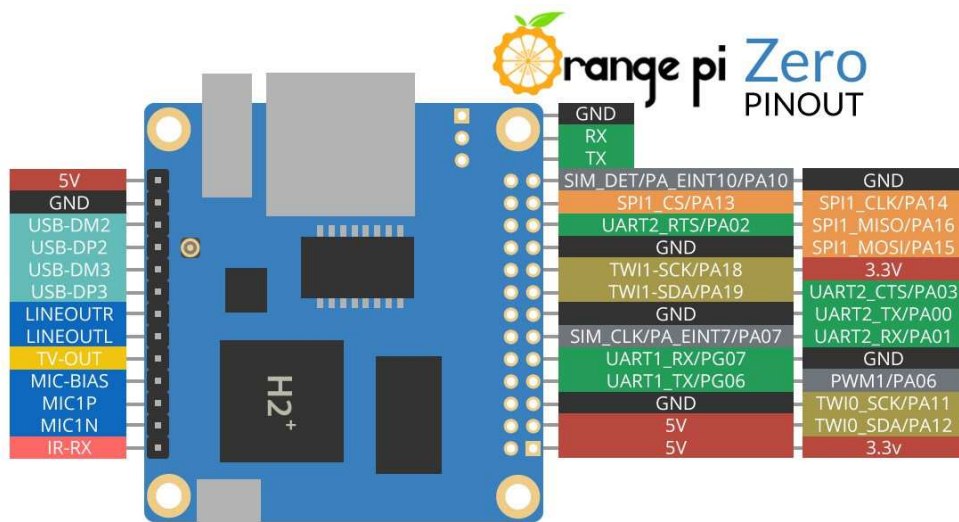
#### 3.2 Aktivera i2c

Vi behöver aktivera en i2c-kanal som skall användas för att kommunicera med andra kort. Aktiveringen görs i armbian-config

Starta armbian-config. Gå in under "System" -> "Hardware" och aktivera i2c0 och spara. (Du använder "spacebar" för att "kryssa" i rutan intill i2c0)

Reboot för att inställningen ska träda i kraft

Nu har vi en I2c-buss på pin 3+5 i den 26poliga kontakten.



### 3.3 Testa i2c

Vi ska installera ett litet bra verktyg för i2c som kan detektera vilka adresser på i2c-bussen som det sitter abonnenter på.

Kör kommandot "apt-get install i2c-tools"

När det är klart, skriv följande kommando "i2cdetect -y 0".  
och terminalen kommer visa en tabell över alla i2c adresser på buss 0

```
root@orange-pi-zero:~# i2cdetect -y 0
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Eftersom inget ännu är kopplat till bussen hittas ingen abonnent. Men om vi kopplar in ett av våra CLEES kort, tex ett PWM kort. Så ser det ut såhär

```
root@orange-pi-zero:~# i2cdetect -y 0
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40: 40  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70: 70  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Nu har vi en fungerande i2c buss.

### 3.4 Installera drivrutin för PWM-kortet PCA9685

Det finns en drivrutin för Adafruits PWM-kort som fungerar fint även till andra kort så länge dom är bestyckade med PCA9685 kretsen.

Du hittar den på Github, här

[https://github.com/adafruit/Adafruit\\_Python\\_PCA9685](https://github.com/adafruit/Adafruit_Python_PCA9685)

Du behöver inte ladda ner den. Installera direkt från github genom att ge följande kommandon:

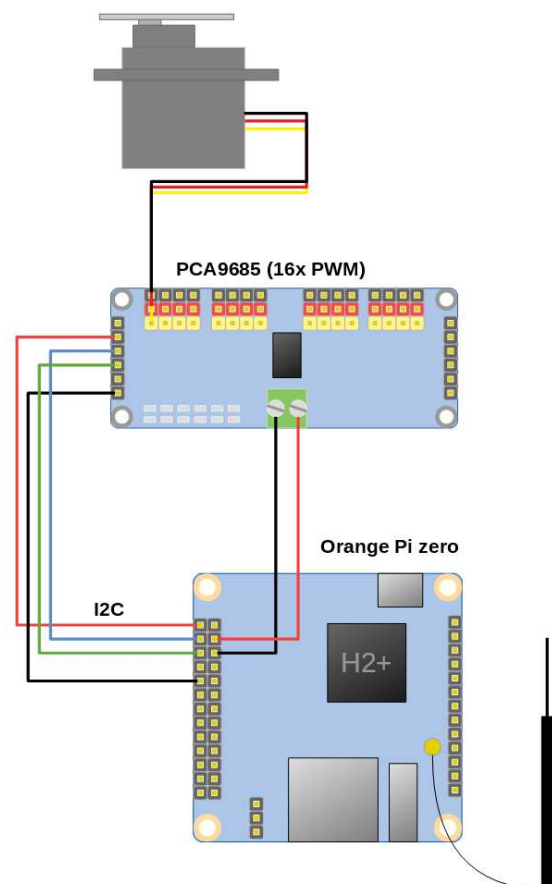
```
sudo apt-get install git build-essential python-dev
```

```
cd ~
```

```
git clone https://github.com/adafruit/Adafruit_Python_PCA9685.git
```

```
cd Adafruit_Python_PCA9685
```

```
sudo python setup.py install
```



### 3.5 Testa PCA9685 med ett RC-servo

Koppla in ett servo till första stiftlisten, se bilden. Anslut också 5VDC till skruvplinten

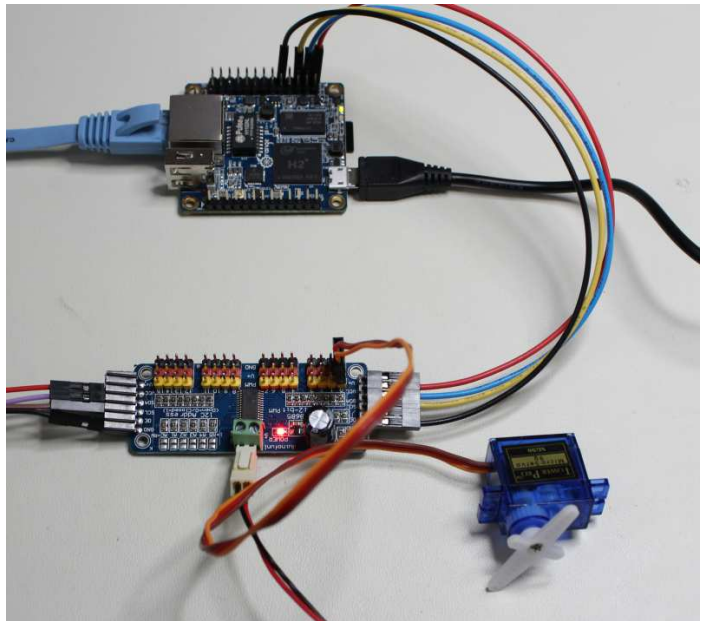
Test görs bäst med ett pythonskript.

- Skapa ett testbibliotek, lämpligtvis under root. Kalla biblioteket "test". (Använd SFTP klienten i Bitvise)
- Skapa en fil i testbiblioteket ock klistra in servotest.py koden nedan"
- Spara filen (servotest.py)

Via terminalen, kör den nya filen

```
cd test
python servotest.py
```

Servot ska nu röra sig fram och tillbaka.



```
#--- servotest.py ---
import time

# Import the PCA9685 module.
import Adafruit_PCA9685

# Initialise the PCA9685 specify a address and/or bus:
pwm = Adafruit_PCA9685.PCA9685(address=0x40, busnum=0) #busnum0 = i2c0

# Configure min and max servo pulse lengths
servo_min = 250 # Min pulse length out of 4096
servo_max = 400 # Max pulse length out of 4096

# Set pwm frequency to 60hz.
pwm.set_pwm_freq(60)

print('Moving servo on channel 0, press Ctrl-C to quit...')
while True:
    # Move servo on channel 0 between extremes.
    pwm.set_pwm(0, 0, servo_min)
    time.sleep(1)
    pwm.set_pwm(0, 0, servo_max)
    time.sleep(1)
#-----
```

### 3.6 Installera smbus

CLEESmjukvaran kommunicerar med flera olika typer av I2C hårdvara och för vissa så behövs en smbus driver. Installera därför följande

```
apt-get install -y python-smbus
```

### 3.7 Installera OPi.GPIO

Raspberry pi har sin Rpi.GPIO. OrangePi har en clone av denna som heter Opi.GPIO som används av CLEES. Installera därför följande:

```
pip install --upgrade OPi.GPIO
```

Mer om Opi finns på webben, här

<https://pypi.python.org/pypi/OPi.GPIO>

och en PDF

<https://media.readthedocs.org/pdf/opi-gpio/latest/opi-gpio.pdf>

## 4 Installera CLEES

Nu är det dags att lägga på själva CLEESmjukvaran. Mjukvara består av flera filer med pythonskript och inställningsfiler. Filerna hämtar du från github, här

<https://github.com/TomasLan/CLEES>

### 4.1 Installera CLEES mjukvara

Filerna du laddar ned från Github skall läggas i en egen katalog, denna "/home/CLEES/"

Gör såhär:

1. Skapa en katalog under /home som heter CLEES  
`sudo mkdir CLEES`
2. Ändra rättigheterna så att alla kan läsa och skriva (Du kan ändra detta senare)  
`sudo chmod 777 CLEES`
3. Gå till Github och hämta filerna  
<https://www.github.com/TomasLan/CLEES>  
Enklarest är att välja "Download zip"
4. Kopiera filer till den nya katalogen "/home/CLEES". Kopiera filer med efternamn ".py" och ".json"  
Filer med efternamn som ".pdf" eller andra ordbehandlingsfiler behöver inte kopieras

### 4.2 Installera MQTT driver

CLEESdatorer kommunicerar med varandra genom MQTT (Message Queue Telemetry Transport). Ett MQTTnätverk består av en "Broker" (=server) och till den ansluter sig olika "clients" (=som lyssnar på meddelanden) och "publisher" (som sänder meddelanden).

CLEES pythonskript både lyssnar och publicerar MQTT meddelanden. För att göra det krävs en installation av paho-mqtt. Skriv följande kommando

```
pip install paho-mqtt
```

### 4.3 Sätt lämpligt hostname

CLEES förespråkar att du namnger dina CLEESdatorer till din stn-förkortning, ex

Xyz#.

Där # är ett nummer som identifierar vilken av dina datorer på stationen det är (även små stationer har alltid minst två stycken datorer, en i varje ända av stationen).

Du kan använda armbian-config för att ändra hostname eller så kan du editera in ändringen i systemfilerna direkt förhand.

Skriv in nya namnet i filen "etc/hostname"

Öppna filen "etc/hosts" och ändra alla ställen med gamla namnet till det nya.

Gör sedan en omstart med

```
reboot
```



## 4.4 Byt till fast IPadress

En fast IPadress underlättar installation och inställningar på den lokala trafikplatsen.

Du kan använda armbian-config för att sätta fast IP eller så editerar du systemfilerna.

Kopiera filen `etc/network/interfaces` som en säkerhetsåtgärd

```
cp interfaces interfaces.orginal
```

Ersätt innehållet i filen `interfaces` med följande

```
# --- eth0
auto eth0
allow-hotplug eth0
iface eth0 inet static
address 172.16.1.199
netmask 255.255.255.0
gateway 172.16.1.1
dns-nameservers 8.8.8.8 8.8.8.4

# Local loopback
auto lo
iface lo inet loopback
```

Ändra värdena i filen så det passar dina beov.

Address skall vara en lokal IP adress:

10.0.0.0 – 10.255.255.255

172.16.0.0 – 172.31.255.255

192.168.0.0 – 192.168.255.255.

netmask kan du låta vara.

gateway skall vara adressen till din router

dns-nameservers kan du låta vara. Den pekar nu på Google DNS och det är OK att använda den.

Starta om med

```
reboot
```

Tänk på att när datorn startat om kommer den ha sin nya IPadress så du kommer behöva byta adress i Bitvise profilen.

## 4.5 Märk dina dotorer med etikett

När du är klar med allt detta så kanske du vill koppla isär allt.

Då är det bra att ha märkt upp sin dator med det viktiga.

Hostname och IPadress. Ett bra ställe att märka på är ethernetkontakthuset.



## 5. Andra installationer och inställningar

En av datorerna på stationen låter man får lite extra uppgifter, såsom att sköta Wifi och köra servrar av olika slag, tex DHCP, MQTT.

### 5.1 Aktivera Wifi och en brygga

Det är lämpligt att välja att bara aktivera Wifi på en av CLEES datorerna på trafikplatsen och sätta upp en brygga mellan Wifi och Ethernet så dom utgör samma nätverk.

Se till att ha wifiantennen inkopplad.

Öppna armbian-config och gå in under Network -> Hotspot.

Opi0 kommer nu göra ett flertal tester. När det är klart ska du boota om Opi0  
reboot

Gå till katalogen "etc/"

Öppna och editera filen "hostapd.conf "

Ändra SSID till Stationen långa namn + förkortningen med #, ex "Holmfors Hfs1".

Filens rad skall se ut såhär

```
ssid=Holmfors Hfs1
```

Aktivera bridge genom att ta bort #.

Filens rad skall se ut såhär

```
bridge=br0
```

Spara "hostadp.conf"

Gå till katalogen "etc/network"

byt namn på filen "interfaces" till "interfaces.orginal"

kopiera filen "interfaces.hostapd" där den nya kopian heter "interfaces"

Starta om Opi0

```
reboot
```

Efter omstartten kommer Opi0 upp med Wifi som är en brygga till det trådburna nätverket.

## 5.2 Installera och Aktivera DHCP

Kan tyckas motstridigt att först sätta fasta IPadresser på CLEES datorerna och därefter aktivera en DHCP server vars uppgift är att dela ut dynamiska adresser. Men det har en bra förklaring.

DHCP serverns huvuduppgift är att ge anslutande enheter som tex mobiltelefoner eller paddor eller t.o.m datorer en möjlighet att koppla upp sig mot CLEES datorerna och kommunicera med dom.

Har du en Wifi brygga aktiverad, se eget avsnitt, kan enheter som ansluter sig till Wifi-nätverket som CLEES delar ut komma åt att göra inställningar på CLEES datorerna. Och då är det bra att dessa har fast IPadress så länge vi inte har en DNS eller annna typ av namnuppslagningsserver installerad.

Installera DHCP

```
apt-get install isc-dhcp-server
```

Gå till katalogen "etc/dhcp" och gör en backup kopia på dhcpd.conf filen såhär:

```
cd ~  
cd ..  
cd etc/dhcp  
cp dhcpd.conf dhcp.conf.orginal
```

Öppna och editera filen "dhcpd.conf". Byt innehållet i filen till detta

```
# CLEES DHCP /etc/dhcpd.conf  
# 2018-07-30  
  
default-lease-time 600;  
max-lease-time 7200;  
  
option domain-name-servers 8.8.8.8, 8.8.4.4;  
option routers 172.16.1.1;  
  
subnet 172.16.1.0 netmask 255.255.255.0 {  
    range 172.16.1.200 172.16.1.248;  
}
```

Öppna och editera "/etc/default/isc-dhcp-server"

Ange bryggan som interface som dhcpservern skall svara på.

```
INTERFACES="br0"
```

starta sedan om dhcp servern

```
systemctl restart isc-dhcp-server
```

Kolla statusen att DHCP servern starta ok med

```
systemctl status isc-dhcp-server
```

och servern kommer svara något i stil med:

```
isc-dhcp-server.service - ISC DHCP IPv4 server
  Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; ...)
  Active: active (running) since Mon 2018-07-30 10:15:57 CEST; 8min ago
    Docs: man:dhcpd(8)
  Main PID: 3886 (dhcpd)
  ...
```

Nu uppdaterar vi vårt nätverk så att bryggan får en statisk IPadress genom att byta innehållet i filen etc/network/interfaces till följande:

```
auto lo br0
iface lo inet loopback

auto eth0
iface eth0 inet manual

auto wlan0
iface wlan0 inet manual

# -- STATIC
iface br0 inet static
address 172.16.1.2
netmask 255.255.255.0
gateway 172.16.1.1
dns-nameservers 8.8.8.8 8.8.4.4
bridge_ports eth0 wlan0

# -- DHCP
#iface br0 inet dhcp
#bridge_ports eth0 wlan0
```

Nu kommer telefoner, paddor och datorer som vill koppla upp sig via Wifi få en dynamisk adress från DHCP servern. Även datorer som kopplar in sig på Ethernet kommer att få en dynamisk IPadress.

Det kanske bör tilläggas att så länge CLEESnätverket inte har en gateway till internet kommer telefonen/datorn inte kunna surfa på internet samtidigt som den är kopplad till CLEESnätverket.

## 5.3 Installera och aktiver MQTT

MQTT servern, eller brokern som den också kallas installeras med följande kommandon

```
sudo apt-get install mosquitto
sudo apt-get install mosquitto-clients
```

Den andra raden installera två kommandoradsklienter så man enkelt kan testa brokern.

Prova med att prenumerera på ett mqtt topic genom att i ge följande kommando

```
mosquitto_sub -h 172.16.1.2 -t 'clees/test' -v
```

Nu fastnar kommandofönstret och väntar på att någon skall publicera något med topic = "clees/test"

Öppna nu ett nytt kommandofönster och publicera ett meddelande med ovan topic

```
mosquitto_pub -h 172.16.1.2 -t 'clees/test' -m 'hello world'
```

Nu ser du att meddelande "hello world" dyker upp i det första kommandofönstret.

Nu är brokern installerad och testad och kommer starta automatiskt varje gång du startar CLEES datorn.

## 5.4 Aktivera MQTT Websocket

Om du vill använda CLEESweb, en webclient som gör att du kan fjärrkontrollera din station via telefonen, så behöver du aktivera websockets i MQTT servern.

Skapa en textfil med följande rader

```
listener 1883  
protocol mqtt
```

```
listener 1884  
protocol websockets
```

Döp filen till något som slutar på .conf, tex: websocks.conf

Lägg filen i katalogen "etc/mosquitto/conf.d/"

Starta om Opi0

```
reboot
```

När Opi0 startar upp igen kommer websocket communication med brokern vara aktiverat på port 1884. All mqtt trafik kommer speglas till det andra interfacet oavsett på vilket man publicerar.

## 5.5 Autostart av CLEES

Innan du gör följande inställningar ska du se till att du har en fungerande CLEES setup och att alla CLEES .json filer mm är uppsatta som du vill det ska fungera. Hur CLEES inställningar fungerar finns beskrivet i CLEES Systembeskrivning.

För att få CLEES att starta automatiskt i dina CLEES datorer lägger du in ett startkommando i filen "/etc/rc.local"

Var försiktig när du editerar denna fil från en windowsdator. Det får nämligen inte komma in några "carriage return" tecken i filen för då fungerar den inte alls. Så endast "Line feed" för radbrytning.

Innan filen avslutats med sitt exit kommand skall du skriva in start kommandod. Så här ska filen se ut:

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
:
:
:
cd ..
cd home/CLEES
sudo python clees.py &
exit 0
```

Raderna som börjar med kolon ska inte vara med i filen utan betyder att det kanske redan står en massa kommando i filen. Låt befintliga kommandon vara som dom är men se till att filen avslutas med CLEES raderna och exit kommandot.

När du startar CLEES datorn kommer nu CLEES programmet gå igång automatiskt.

Glöm inte att göra detta på samtliga CLEESdatorer på din trafikplats.