

Setup Raspberry Pi - 3B for CLEES

Edition: 2018-01-23

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 NOOBS Lite.....	4
1.2 Packa upp filen.....	4
1.3 Sätt SD-card i datorn.....	4
1.4 Formatera SD-card:et.....	4
Formatera SD-cardet med programmet SD memory card Formatter.....	4
1.5 Kopiera den uppackade NOOBS Lite filen till SD-card:et.....	4
2. Starta Rpi för första gången.....	5
2.1 Sätt i SD-card:et i Rpi.....	5
2.2 Anslut nätverk, HDMI, tangetbord, Mus och matningsspänning.....	5
2.3 Göra de första inställningarna.....	7
2.3 Kontakta Rpi via SSH.....	8
2.4 Logga in med Bitwise på Rpi via SSH första gången.....	8
2.5 Stäng av Rpi kontrollerat med "Shutdown" eller "halt".....	9
3 Förbereda hårdvaran för CLEES.....	10
4.1 Rpi har en stiftlist för I/O anslutning.....	10
4.3 Testa i2c.....	11
4.4 Installera drivrutin för PWM-kortet PCA9685.....	12
4.5 Testa PCA9685 med ett RC-servo.....	13

Introduktion

Denna guide förklarar hur du förbereder och kör igång en Raspberry Pi 3B 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

Rpi Raspberry Pi

Steg för steg installation

Innan du börjar installationen av mjukvara på Rpi 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/
4. Programmet Bitvise, som är ett integrerat program med SSH terminal och SFTP klient för att logga in på och kommunicera med Raspberry Pi. Bitvise hittar du här:
<https://www.bitvise.com/ssh-client-download>

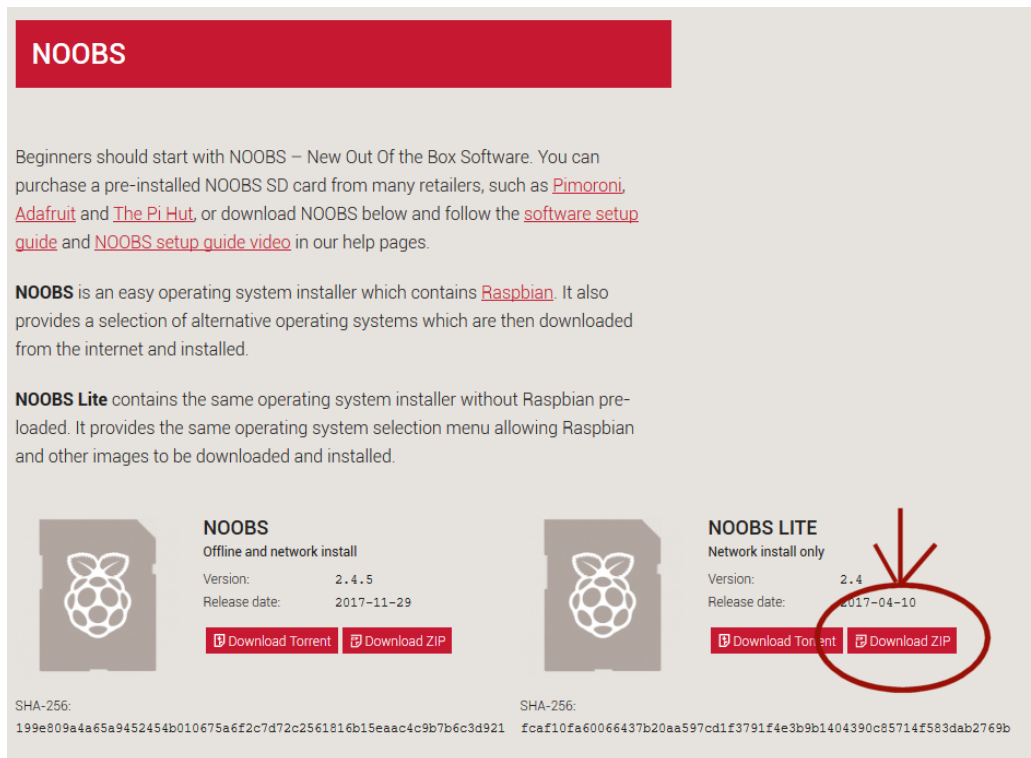
Alla program här listade är fria att använda (iallafall när detta skrevs).

1. Förbered SDkortet

Rpi ska laddas med Linuxdistributionen Raspbian NOOBS. Du ska välja att ladda ned den som heter "NOOBS Lite"

<https://www.raspberrypi.org/downloads/noobs/>

1.1 Ladda ned NOOBS Lite



NOOBS

Beginners should start with NOOBS – New Out Of the Box Software. You can purchase a pre-installed NOOBS SD card from many retailers, such as [Pimoroni](#), [Adafruit](#) and [The Pi Hut](#), or download NOOBS below and follow the [software setup guide](#) and [NOOBS setup guide video](#) in our help pages.

NOOBS is an easy operating system installer which contains [Raspbian](#). It also provides a selection of alternative operating systems which are then downloaded from the internet and installed.

NOOBS Lite contains the same operating system installer without Raspbian pre-loaded. It provides the same operating system selection menu allowing Raspbian and other images to be downloaded and installed.

NOOBS	NOOBS LITE
Offline and network install	Network install only
Version: 2.4.5	Version: 2.4
Release date: 2017-11-29	Release date: 2017-04-10
Download Torrent Download ZIP	Download Torrent Download ZIP

SHA-256: 199e809a4a65a9452454b010675a6f2c7d72c2561816b15eaa4c9b7b6c3d921 fcaf10fa60066437b20aa597cd1f3791f4e3b9b1404390c85714f583dab2769b

1.2 Packa upp filen

Packa upp den nedladdade filen till en temporär katalog.

1.3 Sätt SD-card i datorn

Rpi 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.¹

1.4 Formatera SD-card:et

Formatera SD-cardet med programmet SD memory card Formatter.

1.5 Kopiera den uppackade NOOBS Lite filen till SD-card:et

Hela strukturen med filer och kataloger ska kopieras till SD-card:et.

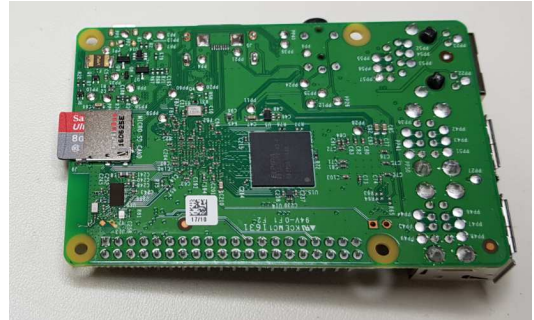


¹ Ett tips är att använda ett SDcard med så lite minne som möjligt. Gärna 8GB om du kan hitta. Det underlättar när man ska göra backupper av hela SD-disken om kortet är litet. Stora kort kan ta timmar att avbilda.

2. Starta Rpi för första gången

2.1 Sätt i SD-card:et i Rpi

Sätt i SD-card:et enligt bilden intill.



2.2 Anslut nätverk, HDMI, tangentbord, Mus och matningsspänning

Koppla nu in ett tangentbord, en mus och en HDMI-monitor. Har du inte någon HDMI monitor går det bra med en datamonitor med DVI och en HDMI->DVI adapter.

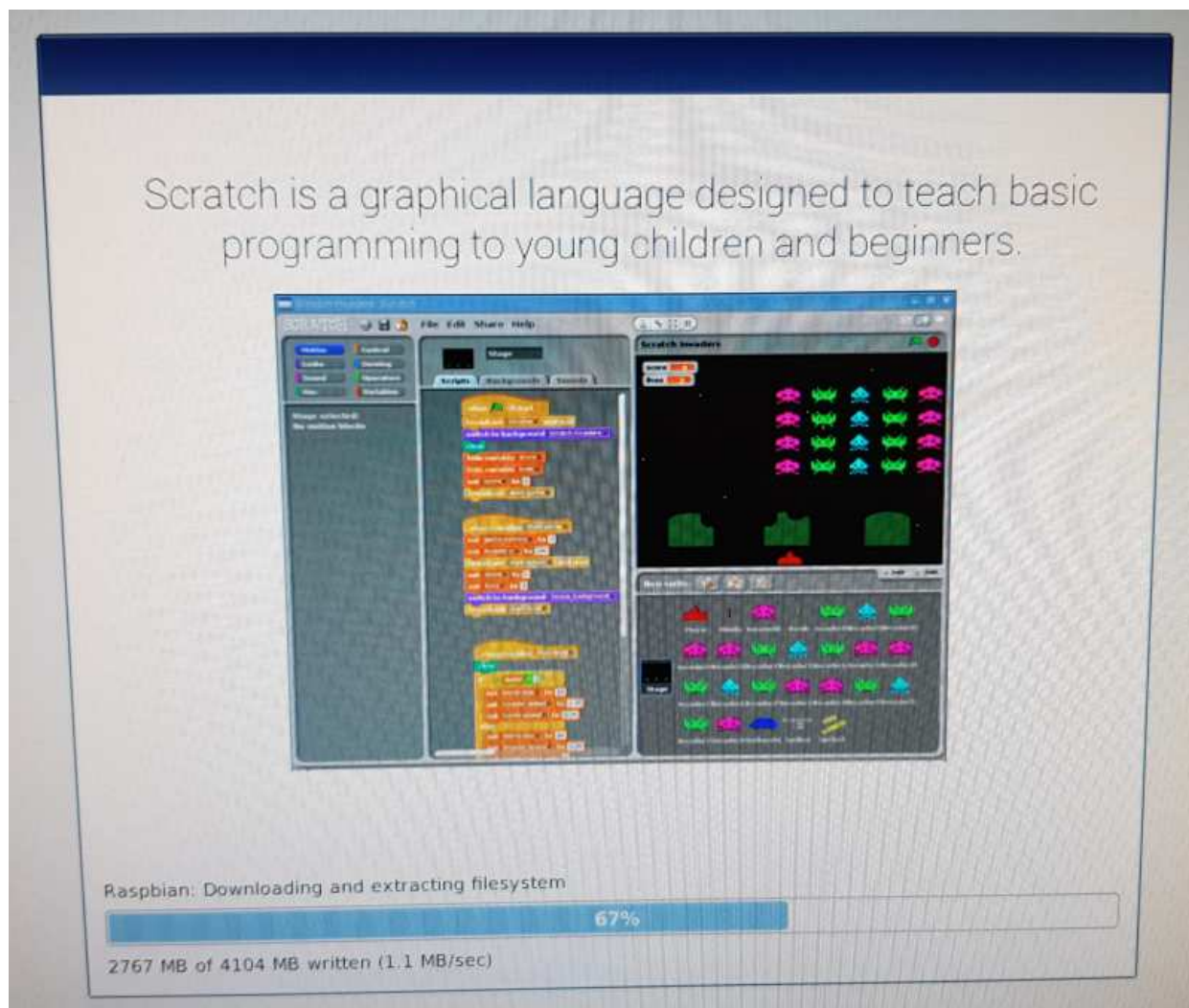


Koppla in Rpi till ditt nätverk. Nätverket måste ha en DHCP server/router som kan dela ut IP-adresser dynamiskt. Alla vanliga nätverk kan detta och har detta aktiverat. Nätverket skall också ha tillgång till Internet.

Så snart strömmen kopplas in så startar Rpi. Skärmen kommer få ett färgmönster och efter 10-20 sekunder kommer en dialog upp som vill att du skall välja operativsystem att installera.

Välj då Rasbian Full installation och starta installationen.

Nu kommer Rpi att ladda ned nödvändiga filer från internet direkt ner i Rpi och sätta upp SD-card:et. Detta kan ta lite tid. 30min eller mer är inte ovanligt.



När installationen är klar ska du starta om Rpi.

2.3 Göra de första inställningarna

De vanligaste inställningarna görs i Raspberry Pi Configuration dialog. Lär dig detta verktyg för du kommer behöva göra fler inställningar här senare.

Börja med att välja fliken "Localisation" och ställa in:

Locale:

Language: sv (Swedish)
Country: SE (Sweden)
Character Set: UTF-8 (default)

Timezone:

Area: Europe
Location: Stockholm

Keyboard Layout:

Sweden -> Swedish

WiFi Country Code:

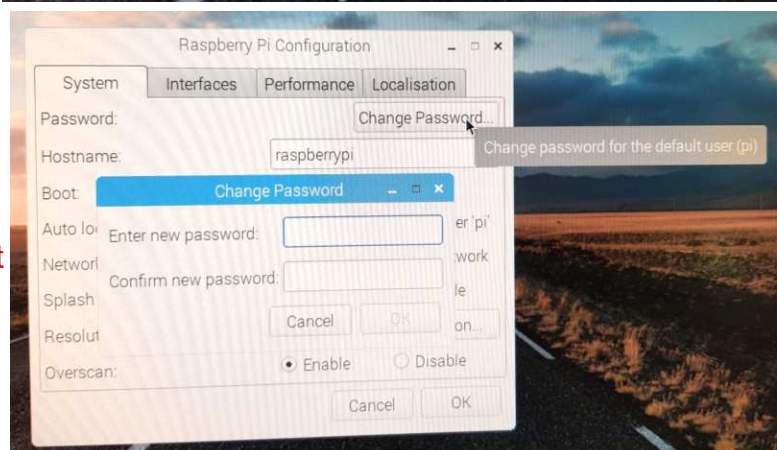
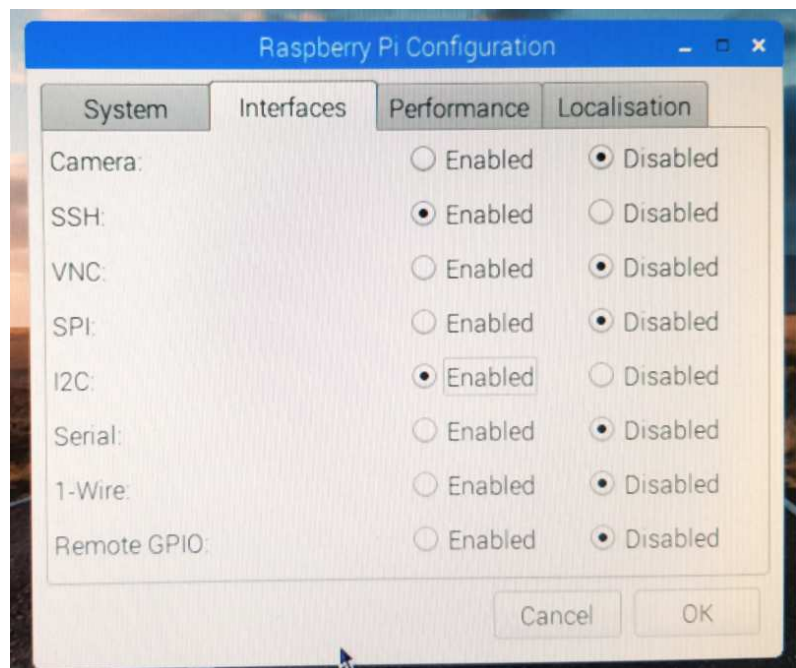
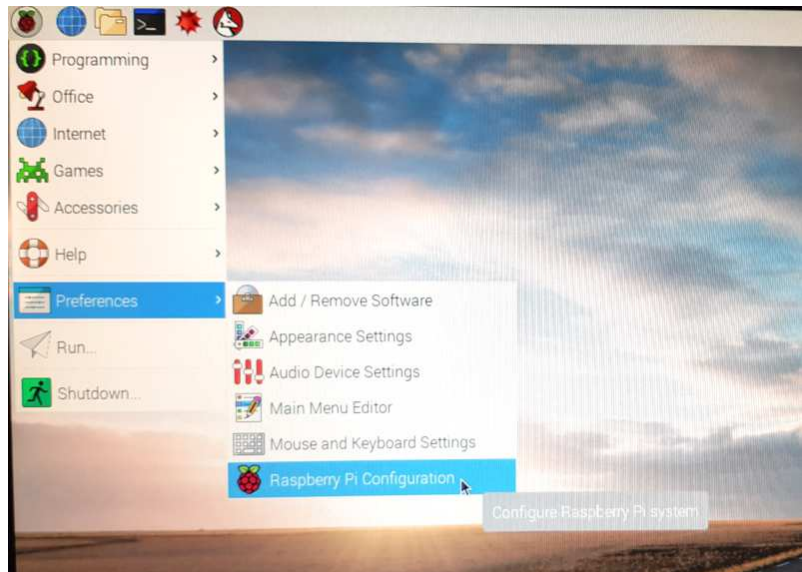
Country: SE Sweden

Välj sedan Tab "Interfaces" och aktivera **SSH** och **I2C**

Starta om Rpi för att ändringarna ska träda i kraft

Nu kommer förmodligen Rpi klaga på att du inte bytt lösenordet. Då för vi göra det nu. Öppna Raspberry Pi Configuration dialogen och ändra lösenord för användare "pi" ("pi" är vår rootanvändare)

OBS! Skriv upp lösenordet i en fil eller på en lapp. Glömmer du bort det kommer du behöva installera om allt från början.



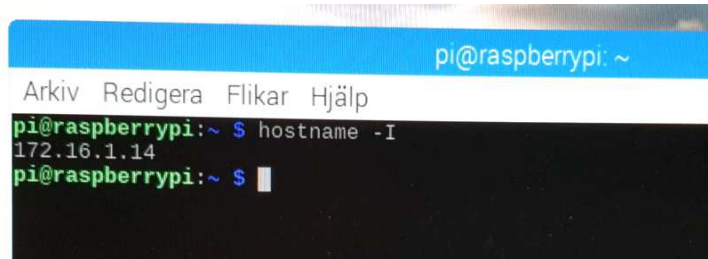
2.3 Kontakta Rpi via SSH

Rpi har fått en IP-adress från din DHCP server. Du behöver Rpi's IP-adress för att kunna logga in på RPi via SSH. För att få reda på vilken adress RPi har kan du använda den inbyggda terminalen.

För att få veta RPi's IPnummer, ge kommandot:

```
hostname -I
```

(-I = stora i)



```

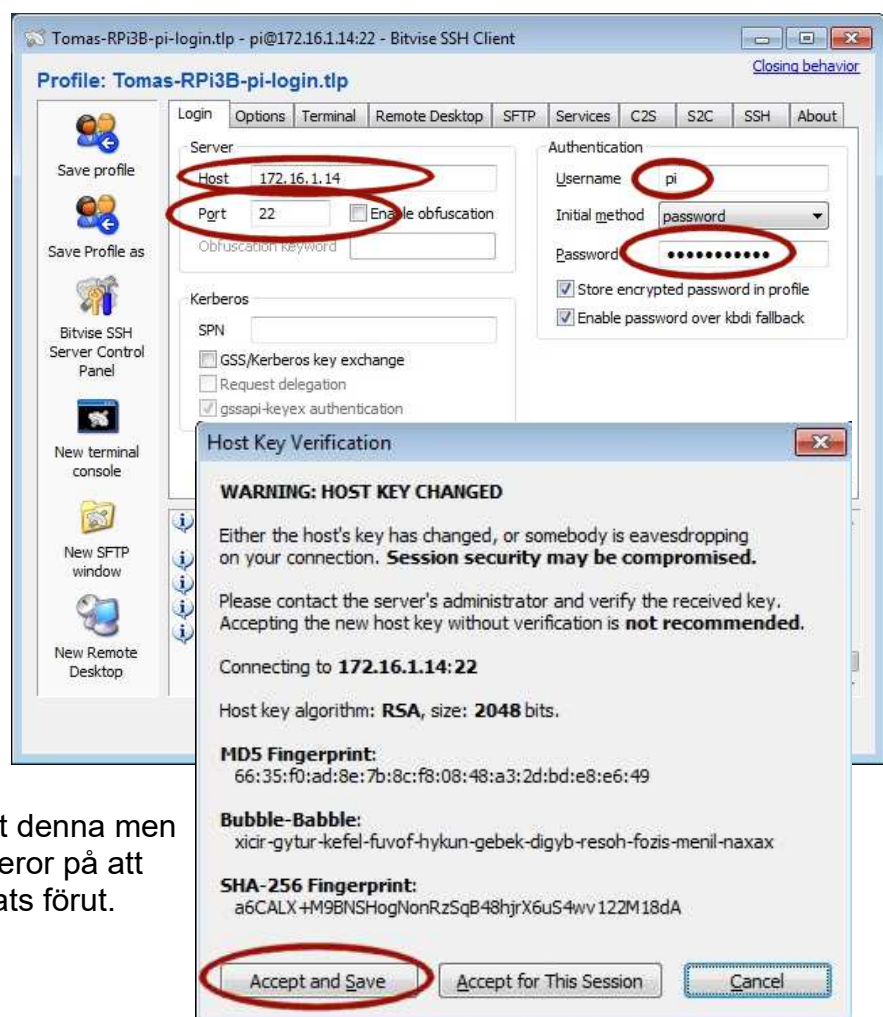
pi@raspberrypi: ~
Arkiv Redigera Flikar Hjälp
pi@raspberrypi:~ $ hostname -I
172.16.1.14
pi@raspberrypi:~ $

```

För att logga in på Rpi 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 Rpi beskrivas med programmet Bitvise.

2.4 Logga in med Bitvise på Rpi via SSH första gången

Starta Bitvise och ange IP-adress, Port=22, username = pi och Password = <det password du angav tidigare>, allt enligt bilden nedan.²



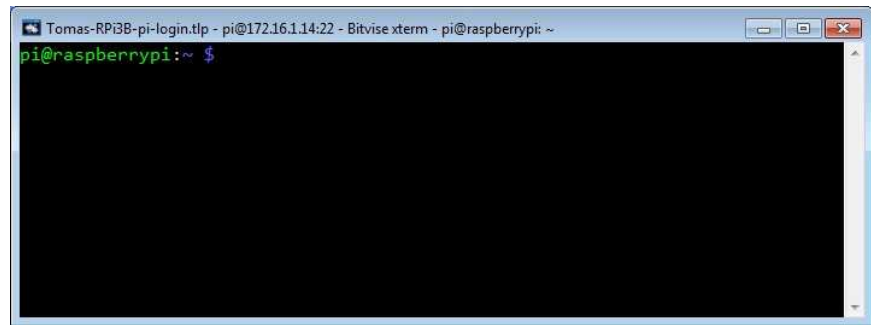
Du kommer få en varning likt denna men det är inte så allvarligt Det beror på att Bitvise och Rpi inte har träffats förut.

Acceptera och spara

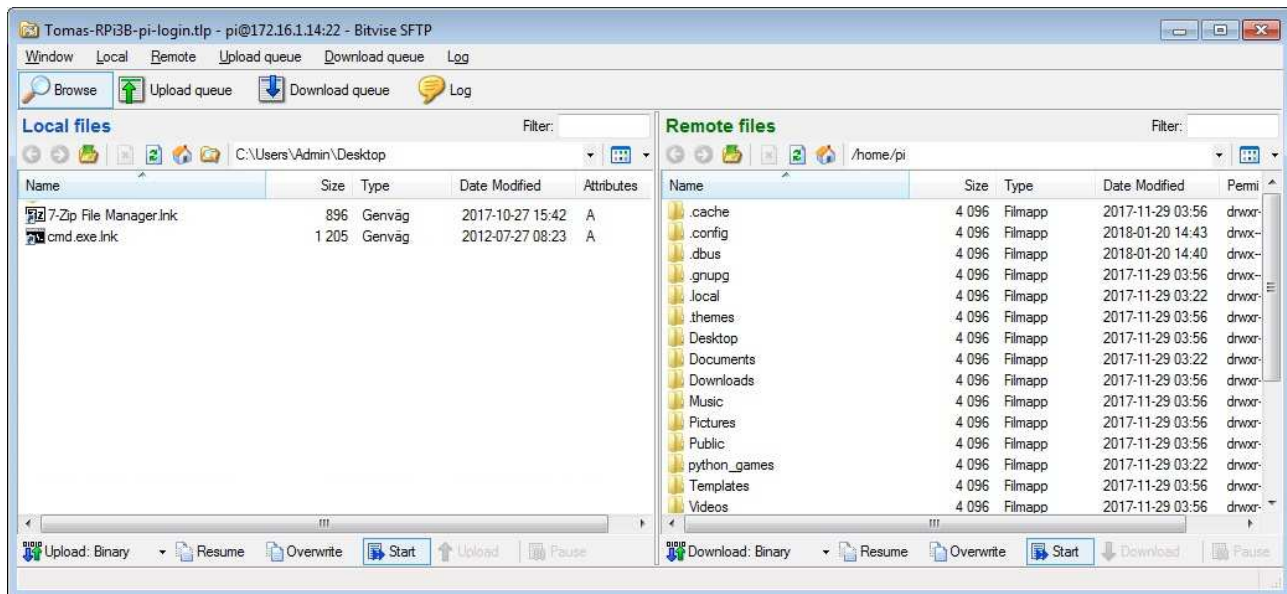
² Du kan spara dina konstellningar i Bitvise genom att spara en profil.

Nu kommer Bitvise öppna ett terminalfönster och ett SFTP fönster.

I terminalfönstret kan du skicka skriftliga kommando till Rpi.



I SFTPfönstret kan du filhantera. Där kan du kopiera och flytta/skapa filer på Rpi's filsystem.



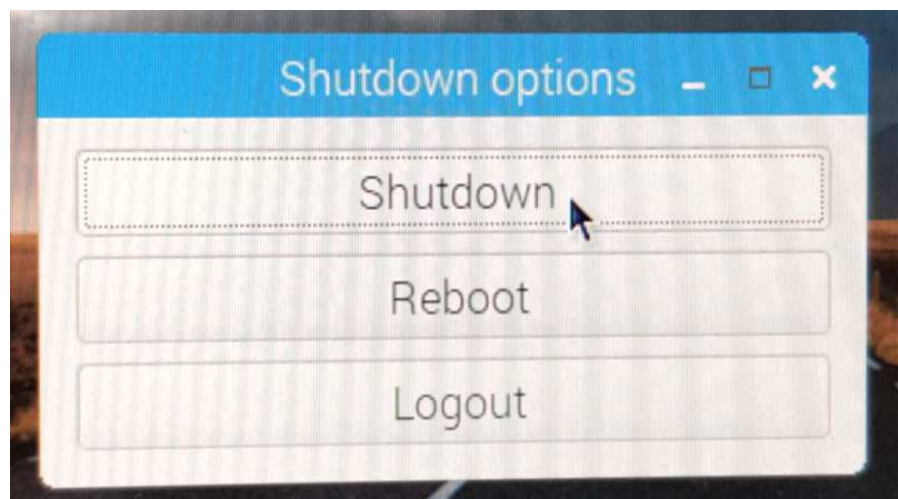
2.5 Stäng av Rpi kontrollerat med "Shutdown" eller "halt"

En linuxdator är alltid lite känslig för abrupt avstängning.

Du ska alltid innan du tar bort strömmen först göra Shutdown.

Om du glömmer göra shutdown finns risk för att filsystemet på SD-card:et blir skadat.

Du kan också ge kommandot "halt" från terminalen. Är du inte admin kan du behöva skriva "sudo halt".



3 Förbereda hårdvaran för CLEES

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

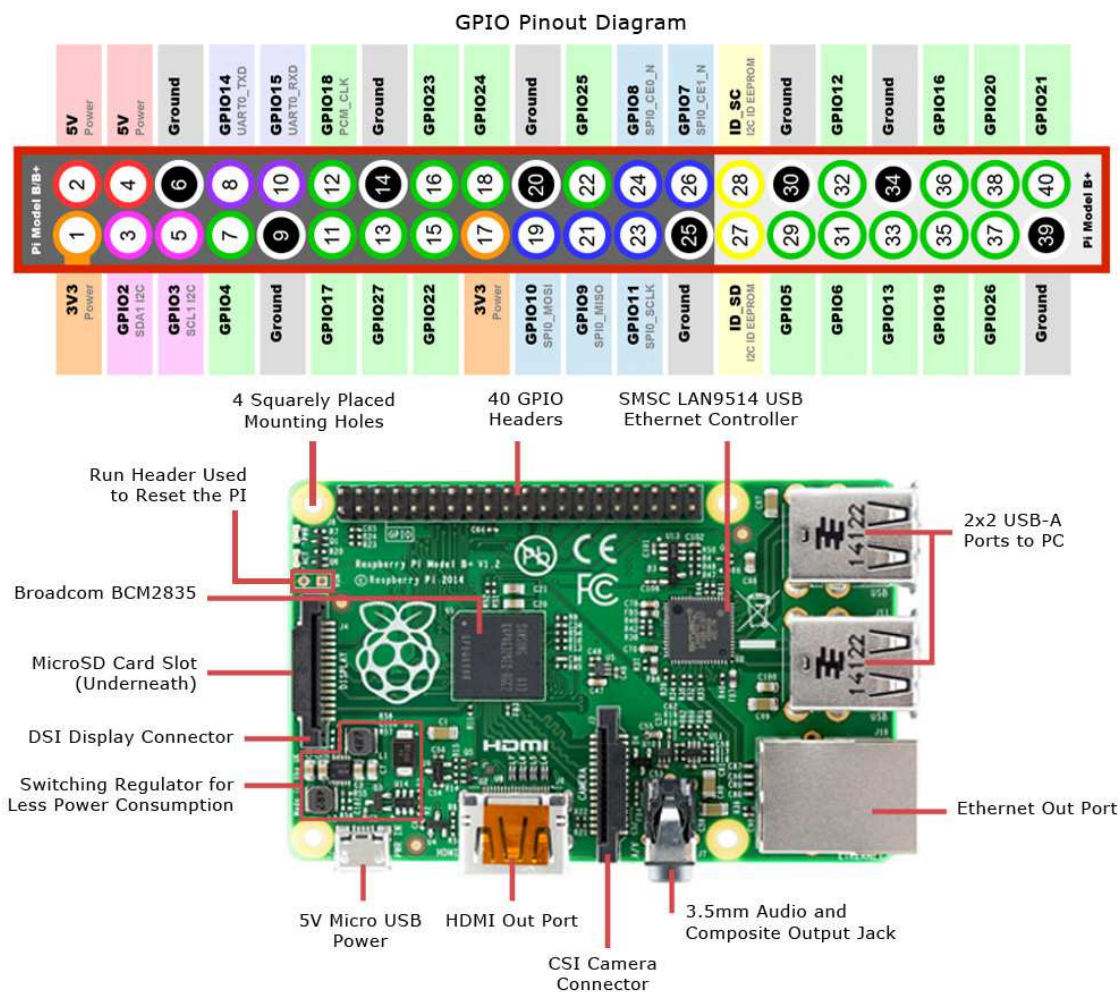
4.1 Rpi har en stiftlist för I/O anslutning

Rpi kommer med en 40polig stiftkontakt för att kommunicera med annan hårdvara tex. via i2c eller GPIO.

Vi har redan aktiverat i2c (se rubrik 2.3) så vi kan direkt börja testa att vi kan styra några av dessa I/O-pinnar via pythonskript.



Vi en I2c-buss på pin 3+5 i den 40poliga kontakten.



4.3 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 kommandona

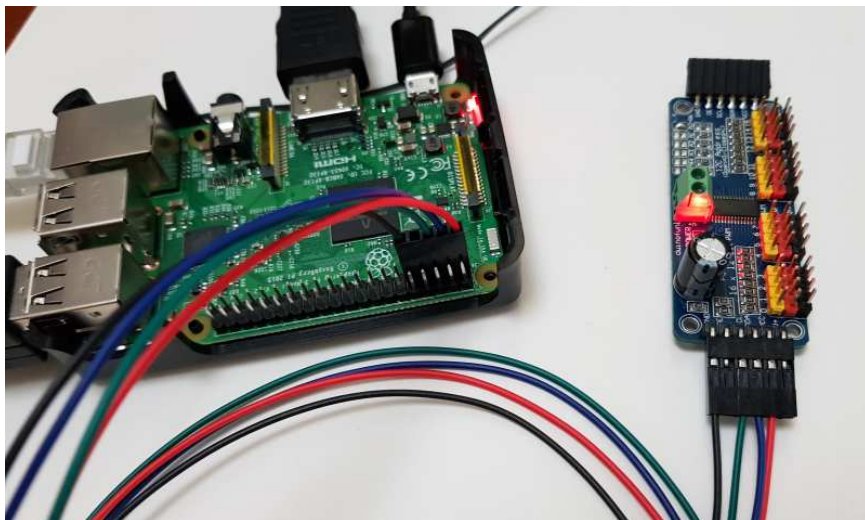
```
sudo apt-get update
```

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

När det är klart, skriv följande kommando "i2cdetect -y 1".
och terminalen kommer visa en tabell över alla i2c adresser på buss 1

```
root@orangepizero:~# i2cdetect -y 1
    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



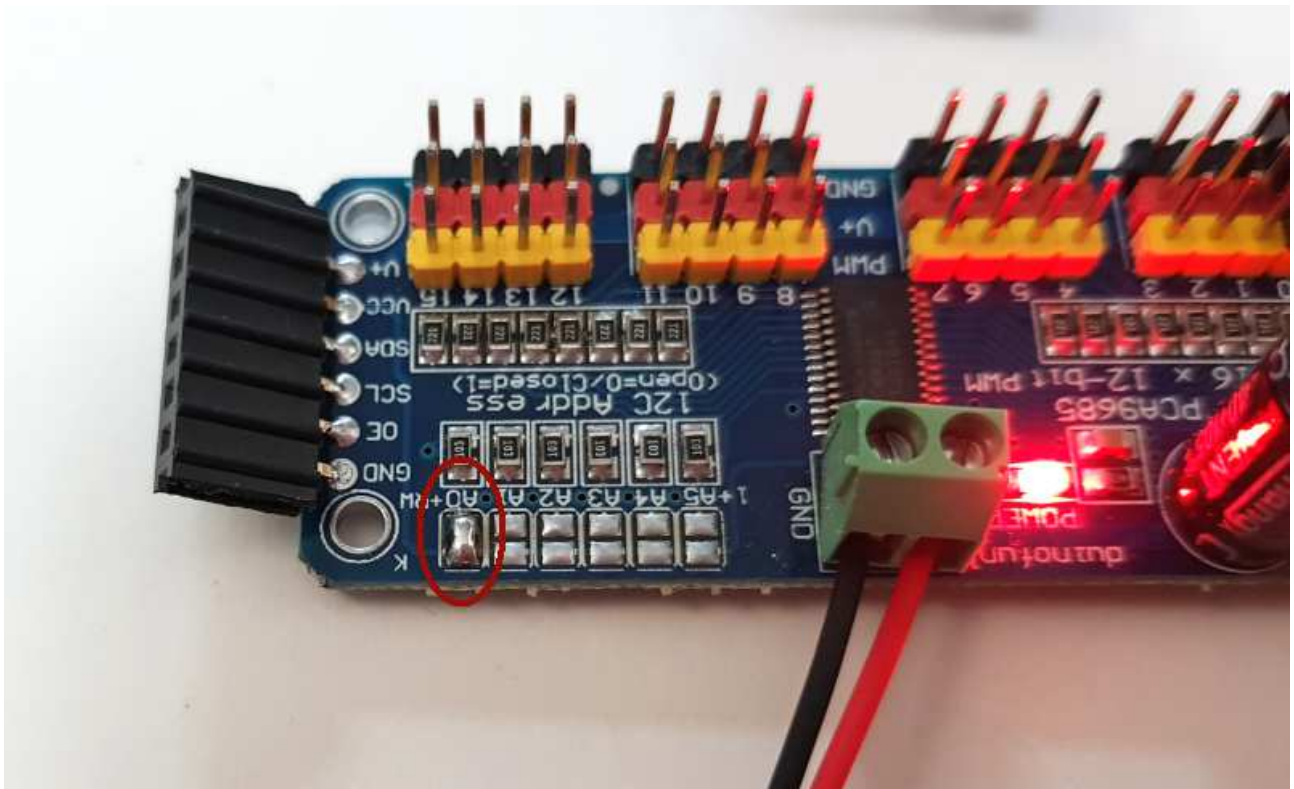
Så ser det ut såhär

```
root@orangepizero:~# i2cdetect -y 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  41  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  70  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Nu har vi en fungerande i2c buss.

Notera att adressen som hittades var 41 och inte 40 som är default på dessa PWM-kort.

Det beror på att just det här kortet har fått en adress programmerad på sig. Det ser man att det finns en lite lödöverbrygging, se bilden nedan



4.4 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

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
```

4.5 Testa PCA9685 med ett RC-servo

Koppla in ett servo till första stiftlisten, se bilden. Anslut också 5VDC till skruvplinten. Det går att ta 5V från Rpi men tänk på att servona kan dra så mycket ström att Rpi bootar om.

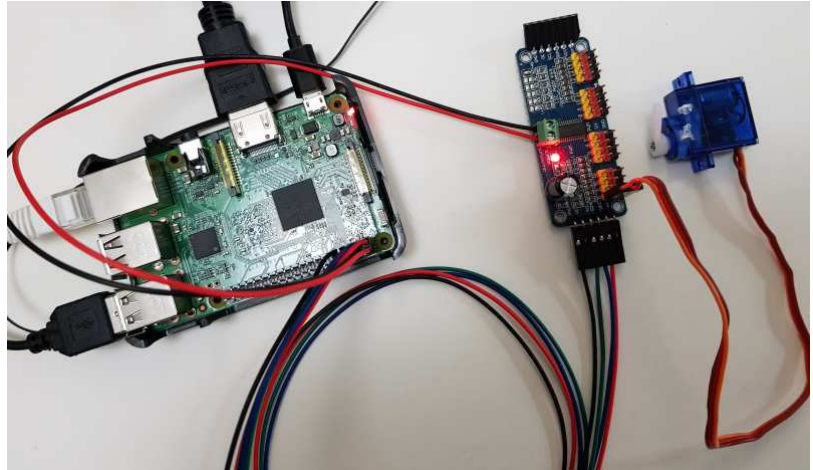
Test görs bäst med ett pythonskript.

- Skapa ett testbibliotek, lämpligtvis under root. Kalla biblioteket "test". (Använd SFTP klienten i Bitwise)
- 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=0x41, busnum=1) #busnum1 = i2c1

# 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)
#-----
```