

# **Solarem – energianohjausjärjestelmä**

# Johdanto

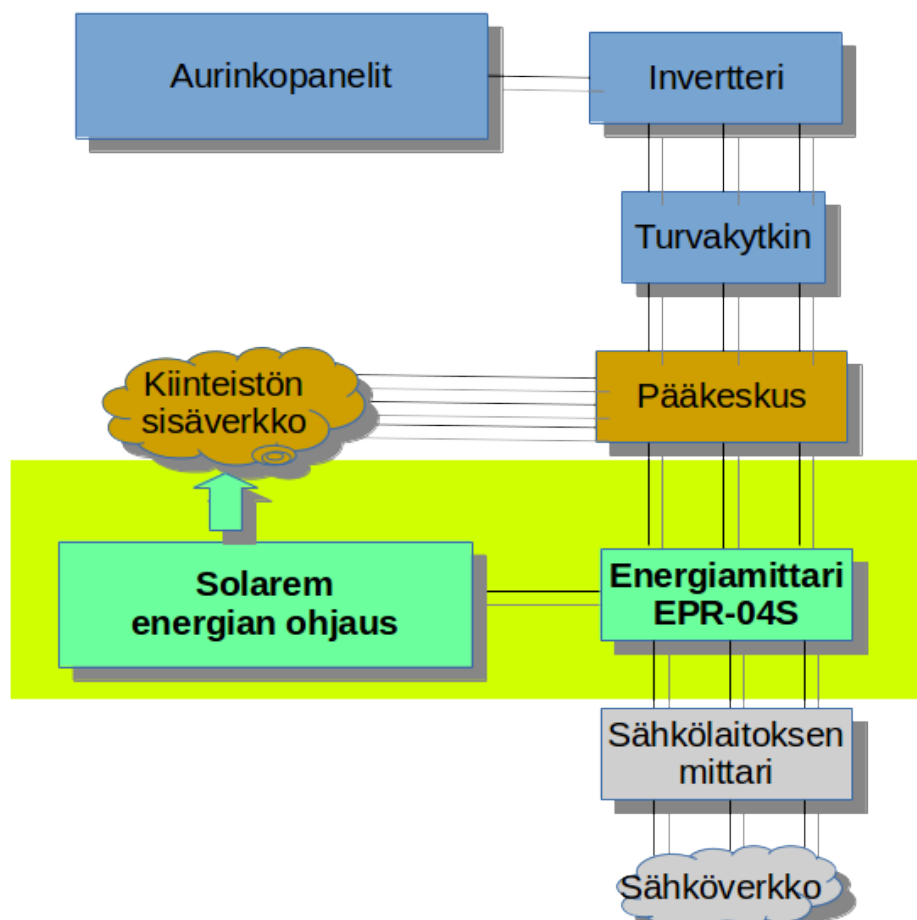
Solarem enegianohjausjärjestelmä on tarkoitettu käytettäväksi pääasiassa kiinteistössä, jossa on aurinkopanelijärjestelmä liitettynä kiinteistön sähköjärjestelmään.

Ennergianohajusjärjestelmän tavoite on kyetä ohjaamaan aurinkopaneleista saatava energia optimaalisesti kiinteistön kulutus- ja varastointikohteisiin, niin että mahdollisimman suuri osa tarjolla olevasta aurinkoenergiasta saadaan kiinteistön omaan käyttöön.

Järjestelmässä käytetään kolmivaihe-energiamittaria, jolla kyetään mittamaan kiinteistön sähköliitännästä tehon määrä ja suunta. Tätä tietoa käytetään ohjaamaan energian kulutusta ja mahdollista varastointia kiinteistössä. Mittarilta saadaan lukemat pätö – ja loistehoista kokonaistehoina tai vaihekohtaisina tehoina. Niitä voidaan käyttää liipaisemaan releitä halutulla tehokriteerillä. Ohjaukseen käytettäviä releitä on kahdeksan.

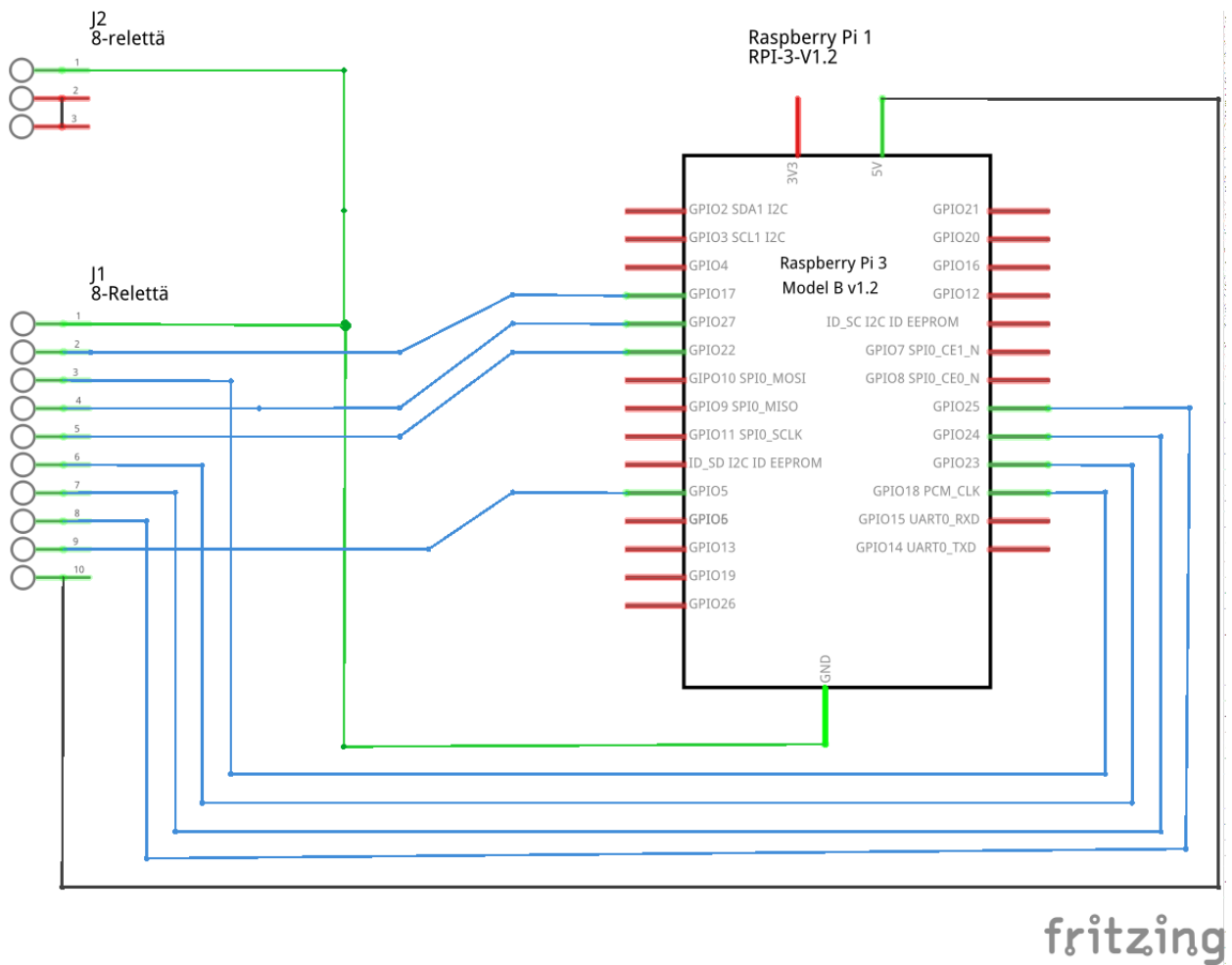
Järjestelmän ytimenä on [Node-Red](#) ohjelmisto, jolla on mahdollista lisätä energian ohjaukseen muitakin ehtoja, kuin mittarilta saatavat tehot. Esimerksi kellonaika, lämpötilat jne.

Oheinen kuva esittää järjestelmän rakennetta:



# Energianohjausjärjestelmän rakenne

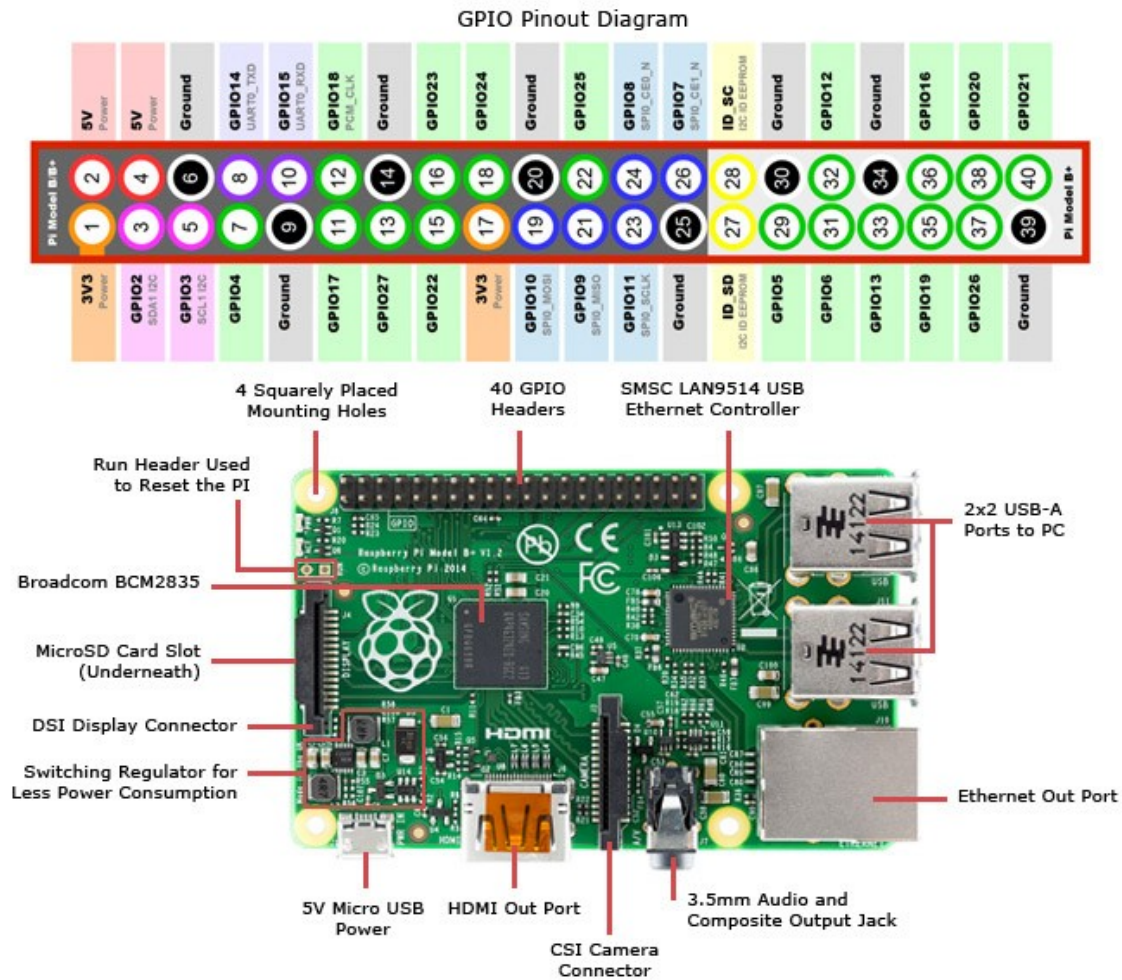
## KytKentäkaavio



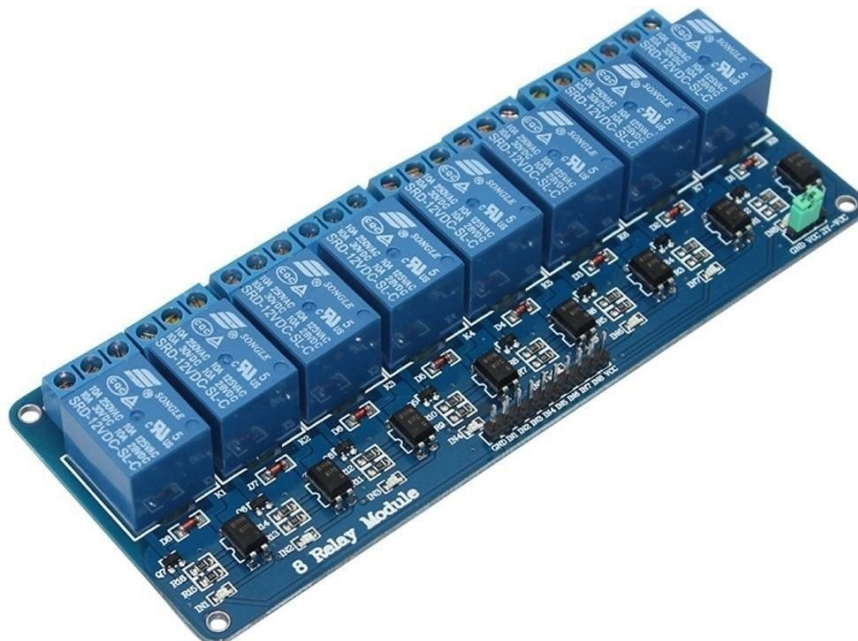
## Releet

Rele	GPIO	Raspberry pin
1	17	11
2	18	12
3	27	13
4	22	15
5	23	16
6	24	18
7	25	22
8	5	29

# Raspberry PI 3



## Relekortti



## Järjestelmään kytkeytyminen

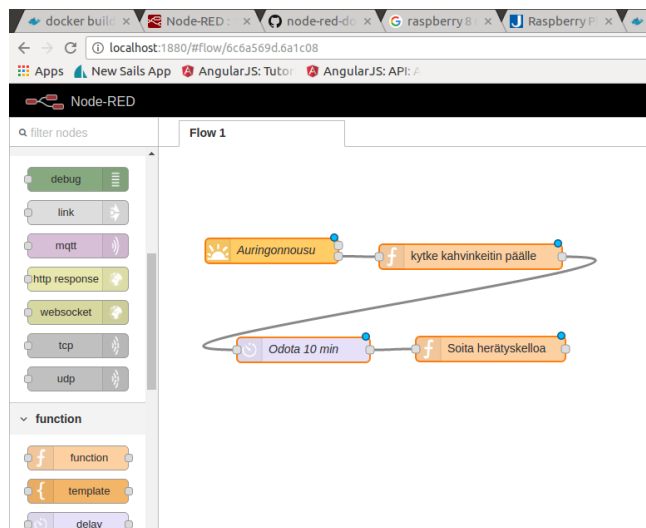
Kun Raspberry PI on käynnistynyt ja liittynyt paikallisverkkoon siihen voidaan kytkeytyä selaimella koneelta, joka on samassa verkossa. Raspberry voi liittyä verkkoon joko ethernet kaapelin tai langattoman wlan verkon kautta: Wlan verkkoon kytkeytyminen vaatii Raspberry PI:n konfigurointia (verkkonimi, verkon salasana), Ethernet kaapelin kautta kytkeytyminen ei vaadi toimenpiteitä. Kytkeytyminen tapahtuu menemällä selaimella osoitteeseen `http://<raspberryn IP osoite>:1880`. Järjestelmä kysyy käyttäjätunnusta ja salasanaa (jos konfiguroitu)

Raspberry PI:n IP osoitteen saa selville esimerkiksi niin, että kytkee Raspberry PI:n HDMI liitäntään näytön. Raspberry näyttää siinä käynnistyessään osoitteensa. Raspberryyyn voi kytkeä myös näppäimistön ja hiiren USB liitäntöjen kautta, mutta ne eivät ole tarpeen järjestelmän käyttämiseksi normaalisti

Ylläolevasta osoitteesta avautuu sivu, jolla järjestelmää voidaan ohjelmoida ja konfiguroida. Sivulta pääsee myös kojetauluun, jossa on näkyvillä mittaukset.

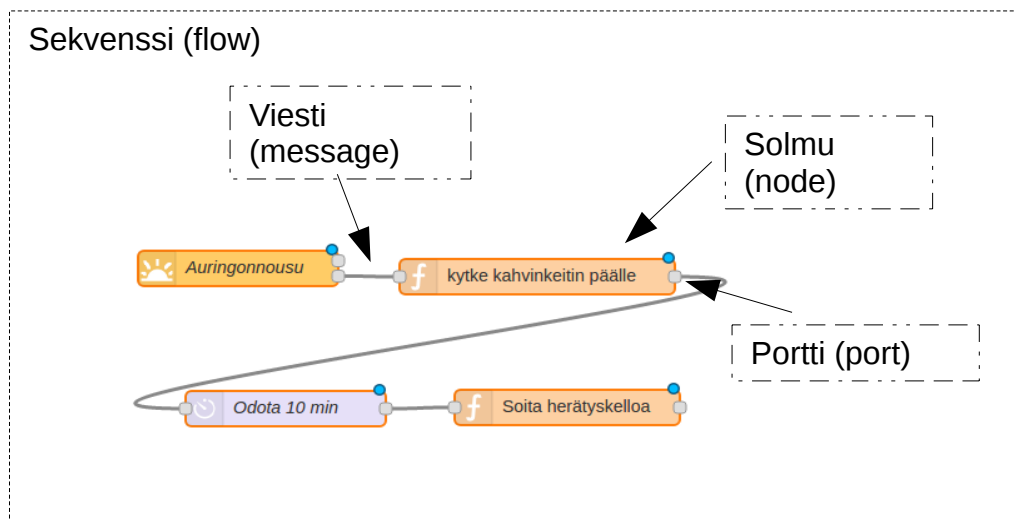
## Node-Red ja konfigurointi

Node-red ohjelmistolla luodaan sekvenssejä (flow), jotka koostuvat solmuista (node), jotka tekevät jonkun toiminnon ja viesteistä (message), joita solmut lähettävät toisilleen porttiansa (port) kautta. Sekvenssit rakennetaan graafisella editorilla vetämällä editorin sivuvalikosta solmuja piirtoalueelle ja yhdistämällä ne sanomia kuvaavilla viivoilla.



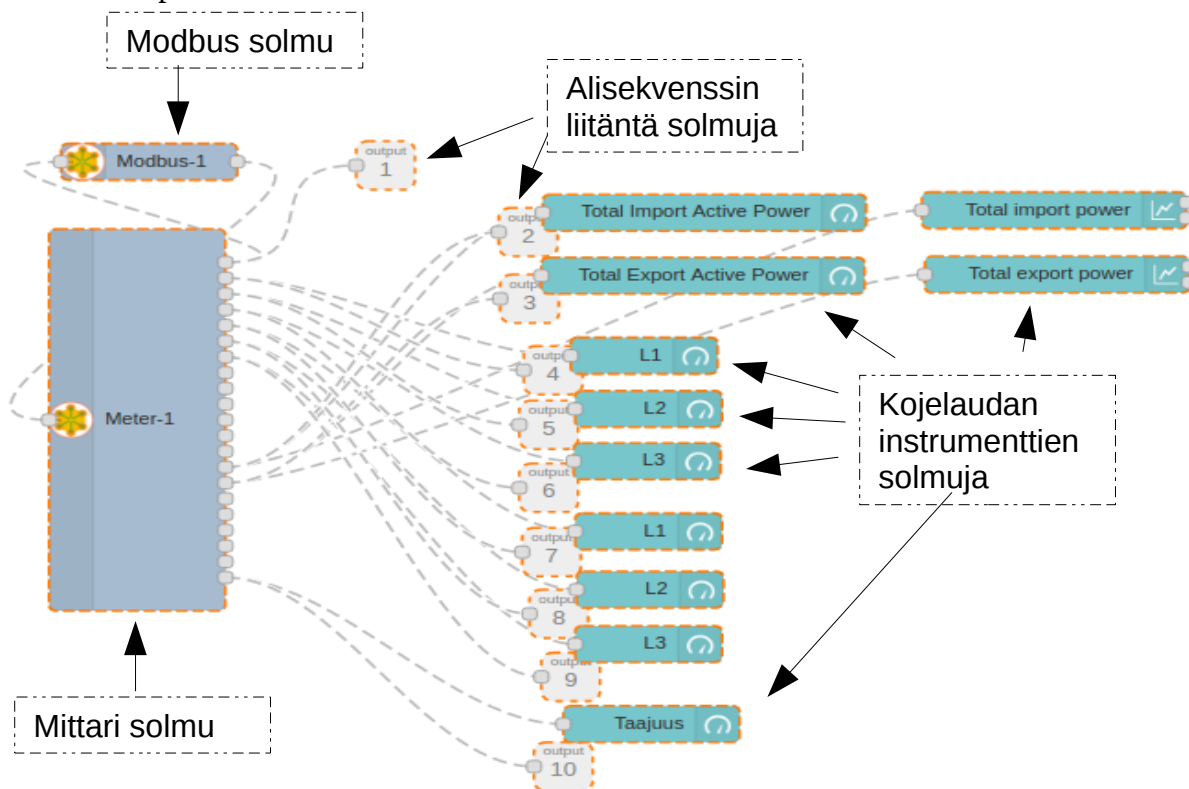
Tarjolla on iso kirjasto solmuja eri tarkoituksiin ja niitä on myös mahdollista luoda lisää itsekin. Esimerkiksi tämän järjestelmän yhteydessä on tehty muutama uusi solmutyyppi (modbus-RTU, Emet mittari ja tehokontrolli). Lähes kaikissa solmuissa on parametreja, joilla niitä voidaan konfiguroida ja esimerkiksi funktiosolmuun voidaan kirjoittaa JavaScript ohjelmointikielellä logiikka, jolla se toimii.

Alla on esimerkki yksinkertaisesta sekvenssistä, joka alkaa auringonnousun ajankohdan tuntevasta solmusta. Se lähettää auringon noustessa viestin funktiosolmulle, joka kytkee kahvinkeittimeen virran (kahvinkeitin on toivottavasti muistettu illalla ladata valmiiksi). Funktiosolmu taas lähettää viestin viivesolmulle, joka odottaa 10 minuuttia ennen kuin lähettää viestin seuraavalle funktiosolmulle, joka soittaa herätyskelloa.

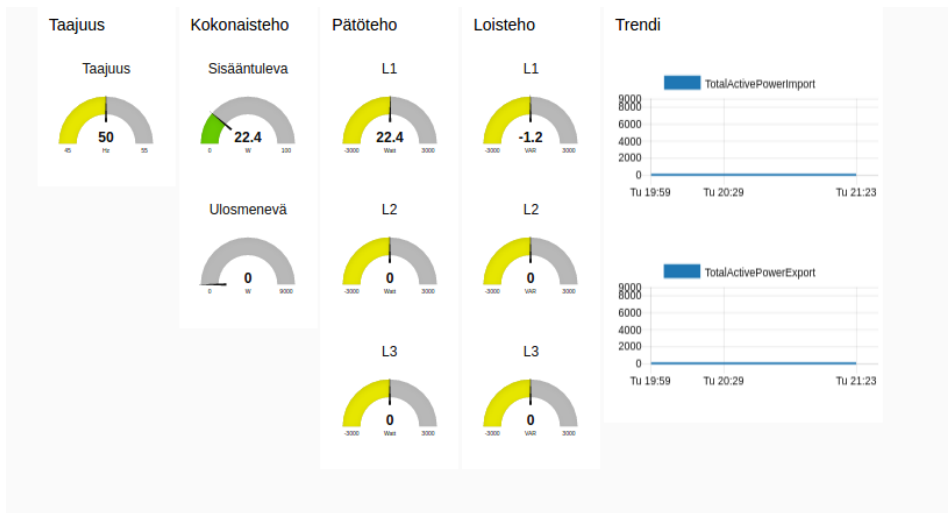


## Modbusin, mittarin ja näyttösivun konfigurointi

Tätä järjestelmää varten on luotu Modbus-RTU solmu, joka osaa kommunikoida Modbus-RTU väylää käyttäen Modbus-yhteensopivien laitteiden kanssa. Järjestelmässä käytetty EPR-04S mittari on sellainen laite. Kyseessä olevalle mittarille on rakennettu solmu, joka kerää mittarilta määrävälein mittaustiedot ja välittää ne eteenpäin Node-Red viesteinä. Mittariin on tehty useita portteja, jotka tarjoavat erikseen kokonaistehon tai vaihekohtaisen tehon mittaukset jne. Lista porteista on lopussa olevassa liitteessä.



Kaaviossa olevat numeroidut output-solmut tuottavat viestin ylemmällä tasolla olevaan sekvenssiin. Oikealla olevat solmut tuottavat kojelaudan (dashboard) instrumentit. Kts kuvakaappaus alla..



Tehonäytöt sisältävän kojelaudan saa näkyviin menemällä selaimella osoitteeseen: <http://<raspberrypi>:1880/ui/#/0>

Mittari luetaan kymmen sekunnin välein. Pollausvälin pituus on mahdollista muuttaa mittarisolmun asetussivulla, johon pääsee kaksois-klikkaamalla mittarisolmua (Meter-1):

**Edit epr04s node**

Buttons: Delete, Cancel, Done

**Name:** Meter-1

**Polling interval:** 10000

**info** | debug | dashboard

**Node**

Name	Meter-1
Type	epr04s
ID	9944c326.9c66a

**Properties**

pollingInterval	10000
-----------------	-------

Emet EPR04S energy meter on Modbus RTU

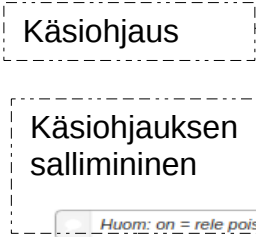
Polling interval is the interval in which

Pollausväli asetetaan millisekunneina. Oletusarvo on 10000 millisekuntia = 10 sekuntia. Kun arvo on muutettu halutuksi painetaan 'Done' nappulaa ja sen jälkeen 'Deploy' nappulaa, joka aktivoi muutokset sekvenssiin.

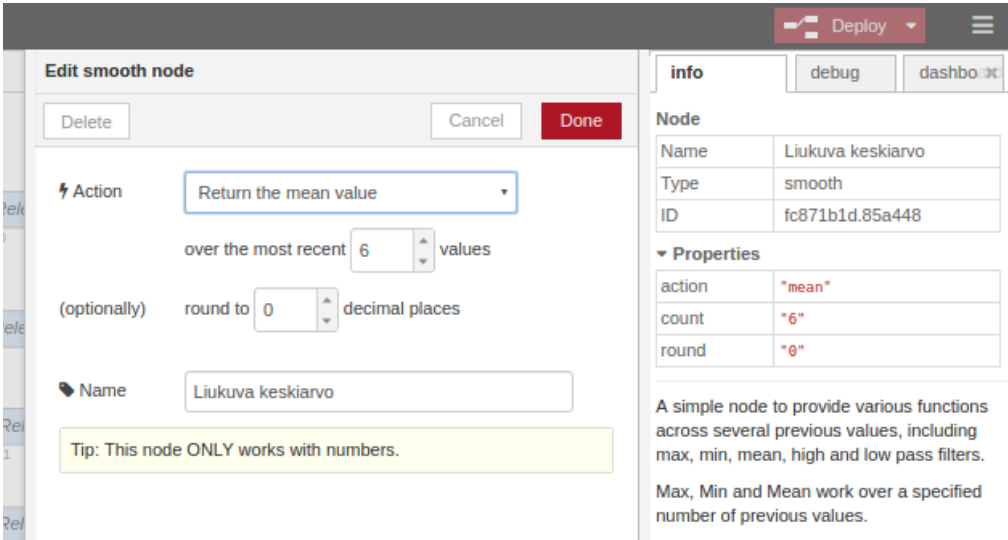
## Releiden ohjauksen konfigurointi

Releiden ohjaus tehorojojen mukaan tapahtuu kytkemällä haluttu mittarin ulostuloportti suoraan tai liukuvan keskiarvosuotimen kautta releen ohjaussomulle. Releen ohjaussolmu kytketään releen GPIO liitäntäolmuun. Releen veto kontrolloidaan Raspberryn GPIO liitännän tilalla. GPIO tilassa looginen 1 (~5V) rele on lepotilassa ja GPIO tilassa looginen 0 (~0V) rele vetää. Releet ovat

Liikuvan keskiarvon solmu      Releenohjaussolmu      GPIO solmu



Liukuvalla keskiarvolla voidaan tasoittaa mittarilta luettavia lukemia. Keskiarvotus on konfiguroitavissa kaksois-klikkaamalla keskiarvosolmua:



Konfiguroitavissa on kuinka monen peräkkäisen mittauksen arvot otetaan huomioon ja arvon pyöristyksen tarkkuus. Tässä käytössä arvo on syytä pyöristää kokonaisluvuksi. Solmulla saa aikaan



# Releenohjaussolmu

Edit power-limit node

Delete
Cancel
Done

**Name for the node**

**Lower power limit**

**Upper power limit**

**Turn relay on in, out, below or above the range lower->higher**

**Initial mode auto or manual**

info

debug

dashboar

Node	
Name	Rele-1-ohjaus
Type	power-limit
ID	d77ea7f5.880f88

**▼ Properties**

lower_limit	"20"
upper_limit	"24"
range	"in"
initial_mode	"auto"

Node for controlling relays based on power limits.

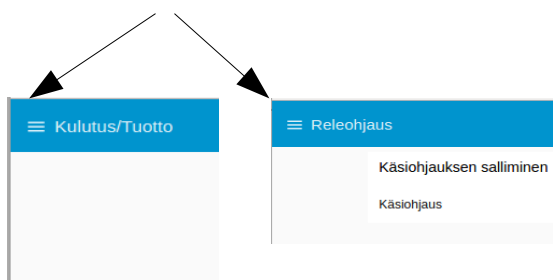
Set lower and higher power limits in Watts. Use 'in', 'out', 'below' or 'above' to specify how the relay should be turned on.

- 'in': relay is turned on when the power is in given range
- 'out': relay is turned on when the power is outside given range
- 'below': relay is turned on when the power is below lower limit
- 'above': relay is turned on when the power is above upper limit

Releet on mahdollista asettaa kojetaulusta käsiohjaustilaan, jolloin releiden tilaa on mahdollista kontrolloida manuaalisesti. Sekvenssi on oletusarvoisesti tehty niin, että kaikki releet menevät käsiohjaus- tai automaattitilaan. Tämä käytös on myös helppo muuttaa rele- tai releryhmäkohtaiseksi tekemällä sekvenssiin useampia käsiohjauksen kontrollinappuloita ja langoittamalla ne releenohjaussolmuihin kuten tämä yksi keskitetty nappula.



Käsiohjaukset sisältävän kojelaudan saa näkyviin menemällä selaimella osoitteeseen <http://<raspberry IP osoite>:1880/ui/#/0> ja valistemalla vasemmalla olevasta valikosta releohjauksen:



# Raspberry PI:n Linux etäkäyttö

## Konsoli

Raspberryn konsolia voi etäkäyttää kytkeytymällä siihen toisesta Linux tai Windows koneesta. Linux koneesta helpoin tapa on kytkeytyä käyttämällä rsh -ohjelmaa:

```
rsh <käyttäjätunnus>@<raspberryn IP osoite>
```

Esim rsh [pi@192.168.100.43](https://pi@192.168.100.43)

Komento kysyy käyttäjän salasanaa (joka on konfiguroitu raspberryn). Ensimmäisellä kytkeytymiskerralla komento esittää raspberry:n sormenjälkiluvun, jota voidaan käyttää varmistamaan, että kytkentä on tehty oikeaan koneeseen.

Windows koneesta kytkeytyminen käy helpoiten PuTTY -ohjelman avulla. Asennusohjelman saa osoitteesta: <https://the.earth.li/~sgtatham/putty/latest/w32/putty-0.68-installer.msi>. Käyttöohjeet löytyvät täältä: <https://tartarus.org/~simon/putty-snapshots/htmldoc/>

## Solarem ohjelman käynnistys ja pysäytys

Node-Red pohjaista Solarem järjestelmää ajetaan [Docker](#) kontissa (container). Se on asennettu käynnistymään automaattisesti virran kytkemisen jälkeen. Normaalisti ohjelmistoa ei tarvitse erikseen käynnistää tai pysäyttää.

Jos kuitenkin ohjelmisto halutaan pysäyttää ja käynnistää se tapahtuu antamalla konsolilta komennot:

Pysäytys: *docker stop solarem*

Käynnistys: *docker start solarem*

## Linux ohjelmiston päivitys

Linux ohjelmisto on syytä ajoittain päivittää mm tietoturvasyistä uusimpiin versioihin. Päivitys tapahtuu antamalla konsolilta komennot:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

Ensimmäinen komento noutaa tiedon päivittyneistä ohjelmistopaketeista ja jälkimmäinen päivittää ohjelmiston. Komennon alkuosa 'sudo' mahdollistaa ohjelmien ajon järjestelmämävalvojan oikeuksilla, jota ne vaativat Sudo kysyy käyttäjän salasanaa aina jonkin ajan kuluttua uudelleen.

Järjestelmää ei välttämättä tarvitse uudelleenkäynnistää. Tarve riippuu päivityksen laajuudesta ja tyypistä. Yleensä on kuitenkin hyvä uudelleenkäynnistää päivityksen jälkeen sen varmistamiseksi, että kaikki uudet versiot tulevat käyttöön.

## Järjestelmän sammutus ja uudelleenkäynnistys

Järjestelmä ei normaalisti vaurioidu, vaikka sen sammuttaa katkaisemalla siitä virran. On kuitenkin hyvä sammuttaa se ohjelmallisesti, jos tarkoituksellisesti se otetaan pois käytöstä tai siitä katkaistaan virta.

Järjestelmän voi sammuttaa ja uudelleenkäynnistää antamalla konsolilta seuraavat komennot:

Sammutus virrankatkaisua varten: *sudo shutdown -h now*

Uudelleenkäynnistys (esim päivityksen jälkeen): *sudo shutdown -r now*

Järjestelmä käynnistyy automaattisesti kun siihen kytketään virta. Käynnistyminen kestää normaalisti noin 30 sekuntia. Jos järjestelmää ei suljettu ohjelmallisesti vaan virrat katkaisemalla käynnistys voi kestää muutaman minuutin järjestelmän tarkastaessa ja korjatessa muistikortin mahdolliset virheet.

## Wlan verkon konfiguroiminen

Kuuluvilla olevat Wlan verkot voi etsiä antamalla konsolilta seuraavan komennon:

```
sudo iwlist wlan0 scan
```

tai

```
sudo iwlist wlan0 scan | grep ESSID
```

Komennolla saadaan selville kyseiseen kohtaan kuuluvat WLAN verkot. Ensimmäinen muoto komennosta listaa paljon muutakin tietoa verkoista. Jälkimmäinen poimii vain verkon nimet (ESSID).

Halutun verkon nimi ja salasana on lisätä suojattuun tiedostoon `/etc/wpa_supplicant/wpa_supplicant.conf`.

Sen voi tehdä esim komenolla:

```
sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

Komento avaa yksinkertaisen editorin, jolla tiedostoa voi editoida. Tiedoston loppuun pitää lisätä seuraava:

```
network={
    ssid="<verkon nimi>"
    psk="<verkon salasana>"
}

esim:

network={
    ssid="RANKAMETTA"
    psk="23KRUHAZOOXI"
}
```

Editori kirjoittaa tiedoston ctrl-o näppäilyllä ja ctrl-x näppäilyllä poistuu editorista.

Järjestelmä on uudelleenkäynnistettävä, jotta se kytkeytyy verkkoon.

Liite: EPR04S Node ports

1: Measurement commands for the meter (via modbus node)

**2: L1 Active Power**

**3: L2 Active Power**

**4: L3 Active Power**

**5: L1 Reactive Power**

**6: L2 Reactive Power**

**7: L3 Reactive Power**

8: L1 Apparent Power

9: L2 Apparent Power

10: L3 Apparent Power

11: L1 Cos Phi

12: L2 Cos Phi

13: L3 Cos Phi

**14: Total Import Active Power**

**15: Total Export Active Power**

16: Total Inductive Reactive Power

17: Total Capacitive Reactive Power

18: Total Apparent Power

19: Average Inductive Cos Phi

20: Average Capacitive Cos Phi

**21: Frequency**

**Korostetut** mittaukset ovat näkyvissä kojetaulunäytöllä.