

Raspberry Pi – saunavahti

Tämän projektityön tarkoitus on rakentaa järjestelmä, jolla tarkkaillaan saunan lämpenemistä. Saunan lämpenemistä tarkkaillaan digitaalisen lämpötilamittarin avulla, sekä videokuvaa hyödyntäen. Tarkoituksena olisi, että ohjelmalle asetettaisiin tavoitelämpötila, jonka saunan halutaan saavuttavan. Lämpötilamittarilla tarkkaillaan ennen kaikkea nykyistä lämpötilaa, mutta myös sitä, miten nopeasti sauna lämpiää. Jos sauna lämpenee tietyllä nopeudella, voidaan laskea koska saavutetaan haluttu lämpötila. Videokuvaa käytetään siksi, että järjestelmä helpottaisi puusaunan lämmittämistä. Kaikki jotka ovat puusaunaa lämmittäneet, tietävät miten turhauttavaa on, jos pesästä on tuli hiipunut.

Raspberry Pille tehtäisiin siis ohjelma, joka tarkkailee lämpötilaa ja sen kehitystä, sekä laskee kehityksen perustella kauanko menee tavoitelämpötilan saavuttamiseen. Aika ajoin, esim kerran sekunnissa, Raspberry Pihin kytketty kameramoduuli ottaisi kuvan tulipesän tilasta. Kuvaan yhdistettäisi informaatio lämpötilasta, että sen kehityksestä. Kielenä käytetään Pythonia.

Raspberry Pi – Käyttöönotto

Kattavat käyttöönotto-ohjeet löytyvät esimerkiksi osoitteesta http://elinux.org/RPi_Beginners. Käytettävät muistikortit ovat esiasennettuja Raspbian Jessie Lite -käyttöjärjestelmällä. Raspin voi joko kytkeä suoraan näyttöön ja liittää hiiren+näppäimistön siihen, tai sitten kytkee Raspin reitittimeen kiinni ethernetkaapelilla ja muodostaa ssh-yhteyden Raspiin esimerkiksi puttyllä. Esiasetetut tunnukset ovat,

```
user: pi
password: raspberry
```

Raspi kannattaa pitää kiinni Internetissä, jotta pakettien ja päivitysten asentaminen onnistuu helposti. Käyttöjärjestelmä kannattaa ensin päivittää komennolla **sudo apt-get update**.

Kannattaa ajaa myös komennot: **sudo apt-get install python-dev** ja **sudo apt-get install python-rpi.gpio** ja **sudo apt-get install i2c-tools** ja **sudo apt-get install python-smbus**.

Raspin saa myös wifi-donglalla verkkoon kiinni, tällöin pitää lukeista ohjeita miten se hoituu. Joka tapauksessa, Raspin käyttäminen ssh-yhteyden kautta on pakollista työn kannalta.

I2C-väylä ja Lämpötilamittarin Liittäminen Raspiin

Lämpötilamittari kytketään Raspissa I2C-väylään. Raspissa on vakiona I2C-väylä poistettu käytöstä. Kirjoita käsky **sudo nano /etc/modprobe.d/raspiblacklist.conf**. Avautuvasta tiedostosta voidaan I2C-väylä ottaa käyttöön kommentoimalla blacklist i2cbcm2708 risuaidalla pois. Tämän lisäksi I2C-moduli on lisättävä kernelin boottilistaan. Kirjoita **sudo nano /etc/modules** ja lisää avautuvaan tiedostoon rivi **i2c-dev**.

Ohjelman konfigurointia varten lisätään käyttäjä pi I2C:n käyttöoikeusryhmään käskyllä **sudo adduser pi i2c**. Oletuskäyttäjä on siis pi, jolla olet kirjautunut sisälle.

Käynnistä järjestelmä uudelleen **sudo reboot**. Rpi:ssä on kaksi I2C kanavaa, 0 ja 1. Kurssin Rpi mallit käyttävät oletuksena 1 kanavaa.

Testataksesi onko väylässä kiinni mitään kirjoita `i2cdetect -y 1`, joka näyttää seuraavalta, jos mitään ei ole kytketty

```
0 1 2 3 4 5 6 7 8 9 a b c d e f
00: -----
10: -----
20: -----
30: -----
40: -----
50: -----
60: -----
70: -----
```

Sensorissa on siis neljä linjaa: käyttöjännite (punainen), maa (musta), data (sininen, SDA) ja kellosignaali (punakeltainen, SCL). Lämpötilasensori toimii 3.3V jännitteellä. Raspberry Pi:n GPIO-pinoutin löytää esimerkiksi osoitteesta http://elinux.org/RPi_Low-level_peripherals. Kytkenän helpottamiseksi on järkevää kytkeä kaikki ensin koekytkentälevylle, josta alkaa jakaa johdotuksia oikeiin GPIO-pinneihin. Kun kytkentä on valmis, kannattaa ajaa komento **i2cdetect**, josta nähdään missä osoitteessa sensori sijaitsee. Jos sensori ei tunnistaudu tarkista kytkentä ja kokeile vielä vaihtaa tarkistettavaa kanavaa.

Lämpötilamittarin Käyttäminen

Tässä lyhyt oheistus ja koodinpätkä siitä, miten Pythonilla saadaan smbussin avulla luettua/kirjoitettua I2C-väylään.

1. Raspissa luo uusi kansio komennolla **mkdir raspiharkka**
2. Siirry kansioon komennolla **cd raspiharkka**
3. Luo kansioon uusi tekstitiedosto **nano luelampo.py**
4. Kirjoita oheinen koodi ja tallenna tiedosto

```
#!/usr/bin/python
import smbus
import sys

REG_TEMP = 0b00
I2C_ADDRESS = 0x48

bus = smbus.SMBus(1)
temperature = bus.read_byte_data(address, REG_TEMP)
print(temperature)
```

5. Suorita ohjelma komennolla **python luelampo.py**

Koodissa I2C_ADDRESS on se, joka saatiin i2cdetect komennolla esille. REG_TEMP taas on rekisteri lämpömittarissa, jonne lämpötilan on luettavaksi. Muuttuja bus liitetään I2C-kanavaan, joko 0 tai 1, riippuu kummalla lämpötilamittarin havaitsi. Lämpötila luetaan seuraavalla rivillä ja sitten se tulostetaan. Lämpötila on kokonaislukuna, jos tarkempia mittauksia haluaa tehdä, niin sitten pitää tutustua mittarin manuaaliin ja ottaa selvää siitä, jos haluaa desimaleja esille.

<http://ww1.microchip.com/downloads/en/DeviceDoc/21935a.pdf>

Kameran Liittäminen Raspberry Pihin

Koska Raspin community on hyvin laaja, liitä tähän vain linkkejä mistä löytyy tietoa kameran kytkemiseksi ja kokeilemiseksi:

Asennus:

<https://www.raspberrypi.org/documentation/usage/camera/README.md>

Käyttäminen:

<https://www.raspberrypi.org/documentation/raspbian/applications/camera.md>

Esim komento. **raspistill -o kuva1.jpg** ottaa kuvan ja tallentaa sen nimikkeellä. Löytyy myös komentoja jolla kuva otetaan aika ajoin, esim 10 kertaa sekunnissa.

Muuta hyödyllistä

Koska käytössä ei ole graafista käyttöliittymää, ja voi ehkä haluta tarkistaa tallentaako Raspi kuvat oikein. Ohjelmalla WinSCP saa muodostettua yhteyden Raspiin siten, että sen sisältöä voi selata kuin Windowsin Resurssien hallinnalla. Raspissa olevat tekstieditorit kuten, Vi tai Nano, eivät välttämättä avaudu samantien logiikaltaan: joten jos tykkää ennemmin koodata jollain tutummalla tekstieditorilla, niin WinSCP:n kautta halutut kooditiedostot voi viedä Raspiin tai avata ne suoraan editoitavaksi.

Toiminta

Ohessa on tullut jo aika paljon apuja työn suorittamista varten. Lyhykäisyydessään:

1. Käynnistä ohjelma haluamallasi tavoitelämpötilalla
2. Lue nykyinen lämpötila
3. Muodosta luetuista lämpötiloista ennuste kauanko menee saavuttaa tavoitelämpötila
4. Ota kuva
5. Liitä tiedot kuvaan
6. Streamaa kuvaa Raspilta, siten että sitä voi tarkastella toisella laitteella

Raspin community on hyvin laaja, googlailulla pääsee hyvin pitkälle kun alkaa työtä tekemään. Ongelma on siinä, että pitää tietää mitä etsiä. Jos Raspi tai Linux-ympäristö ei ole ennestään tuttu, niin lisäähän se aina haastetta. Mutta jos tuntuu että työtä tehdessä ei pääse eteenpäin, niin viestiä ohjaajalle.

Toteutuksen testaaminen on oma juttunsa, Raspia ei välttämättä kannata viedä saunaan testattavaksi eikä sensori varmaan kauheasti kosteuttakaan/vettä kestä. Luovia ratkaisuja voi käyttää toiminnan varmistamiseksi.

Lisäksi työstä kevyt dokumentaatio, miten työ on toteutettu ja toiminta varmistettu.

Hannu-Pekka Hedman

hanhed@utu.fi

(Alkuperäinen Sami Teräväinen)