

Raspberry Pi installeren

Standaard installatie Raspbian

De rest van deze gids is beschreven voor de meest recente versie van Raspbian (distributie gebaseerd op debian).

<http://www.raspbian.org/>

Installatie wordt niet in dit document beschreven maar staat zeer duidelijk beschreven in de pdf hieronder vermeld:

http://www.raspberrypi.org/wp-content/uploads/2012/04/quick-start-guide-v2_1.pdf

Connecteren op command line (optioneel)

Alle setup is beschreven vanuit terminal-modus.

Als je de raspberry pi rechtstreeks op een scherm kan aansluiten (hdmi) open je een terminal-sessie, als je de RPi niet in scherm-modus hebt gestart kan je connecteren via ssh.

Bij de standaard installatie is er een ssh-server geactiveerd (bij de out-of-the-box-installatie via user pi en password raspberry hoewel het wel aangeraden dit te wijzigen bij installatie).

Voor Linux en Mac moet je niet specifiek een installatie doen, voor Windows gebruik je best een applicatie zoals Putty om te connecteren.

Voorbeeld Ubuntu/Mac (of andere OS waar OpenSSH is voorzien) zal je connecteren met het ssh-command onder de volgende vorm:

```
ssh ${user}@${hostname}
```

Dit kan bijvoorbeeld met een hostname of een ip-adress

```
ssh pi@bartsraspberrypi
```

```
ssh pi@192.168.0.206
```

Na het inloggen zal de Rpi je om een password vragen.

Editor

De voorbeelden gebruiken vi als editor, je bent natuurlijk vrij van eender welke editor te gebruiken zolang je deze via “sudo” opent (waar zo beschreven staat in de voorbeelden).

Nano bijvoorbeeld is een zeer gemakkelijk een eenvoudig (command-line) alternatief.

Internet-connectie

Voor de setup van i2c-connectiviteit, java en tomcat heb je enkel een netwerk-verbinding.

Voor de eerste installatie is het ook aangeraden dat geconnecteerd aangezien je een aantal installaties en updates moet doen vanuit de Raspbian-repositories.

Raspberry Pi voorbereiden voor i2c-connectie

Alvorens de SHT21 te kunnen aansluiten op de raspberry pi dienen er eerst een aantal zaken worden geconfigureerd op de Pi:

Uitschakelen van de blacklisting:

Daarvoor connecteer je op de RPi en navigeer je

```
cd /etc/modprobe.d/  
sudo vi raspi-blacklist.conf
```

Als je in deze file edit zet je vervolgens 'blacklist i2c-bcm2708' in deze file in commentaar:

```
# blacklist spi and i2c by default (many users don't need them)  
  
# blacklist spi-bcm2708  
# blacklist i2c-bcm2708
```

Activatie van 'i2c-dev'-module (Linux kernel)

Je navigeert als volgens en opent de modules-file

```
cd /etc  
sudo vi modules
```

Hier configureer je de modules die door de Linux-kernel moeten worden geladen bij opstart-tijd
Het volstaat hier om de alle modules startende met i2c uit commentaar te zetten

```
# /etc/modules: kernel modules to load at boot time.  
#  
# This file contains the names of kernel modules that should be loaded  
# at boot time, one per line. Lines beginning with "#" are ignored.  
# Parameters can be specified after the module name.  
  
snd-bcm2835  
i2c-bcm2708  
i2c-dev
```

Als laatste installeer je i2c-tools en voeg de user pi (of de user waarmee je bent ingelogd) aan de groep i2cc

```
sudo apt-get install i2c-tools  
sudo adduser pi i2cc
```

Herstarten en testen:

Om de installatie te beindigen kan je de Raspberry Pi herstarten via het volgende commando

```
sudo shutdown -r now
```

Daarna kan je testen of de er een de tools correct zijn geïnstalleerd door het commando in te geven dat de aanwezigheid test van een i2c-device.

```
sudo i2cdetect -y 1
```

Als je geen device heb aangesloten zal dit een lege tabel geven (tenzij je reeds een i2c-device hebt aangesloten):

[illegible]

Connecteren van SHT21 en RPi

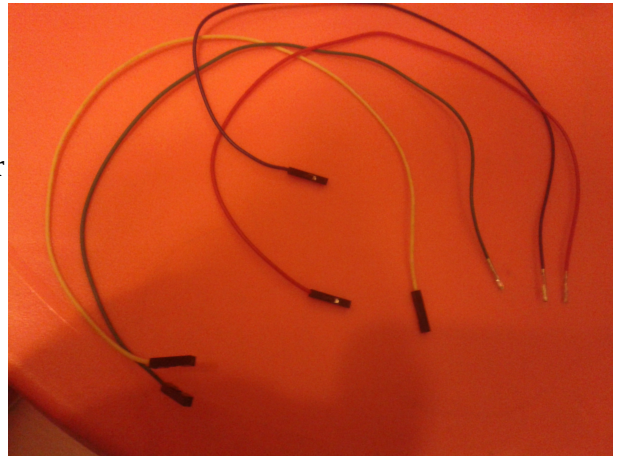
Aanpassen van de evaluatie-kit

Gebruikmakende van:

- Evaluation Kit EK-H5
- SHT21
- 4 jump-wires (female-to-female), bij voorkeur met een verschillende kleur

Stap 1: Ontkoppelen van de USB-module

Je kan de USB-module vrij eenvoudig loskoppelen via de connector-module in het midden. Door het klemmetje vanboven op deze connector in te duwen kan je beide modules voorzichtig uit elkaar trekken

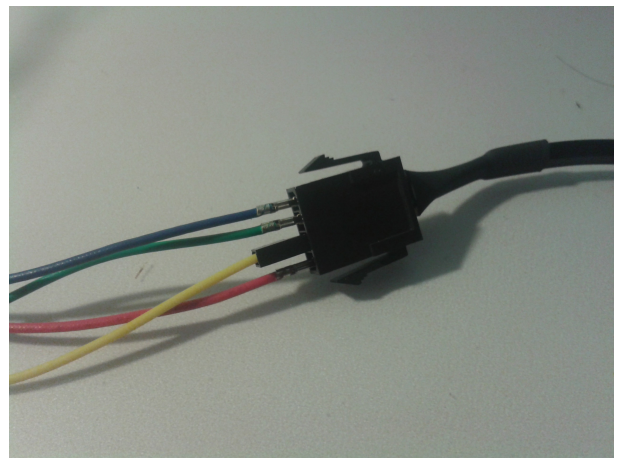


Stap 2: Strippen van de jump-wires

Een maal deze connector ontkoppeld heb je de 4 jumpwires nodig. Deze jumpwires dienen aan 1 kant te worden ontdaan van het plastic hoesje zodat enkel de ijzeren connector overblijft.

Stap 3: Connecteren van de jump-wires

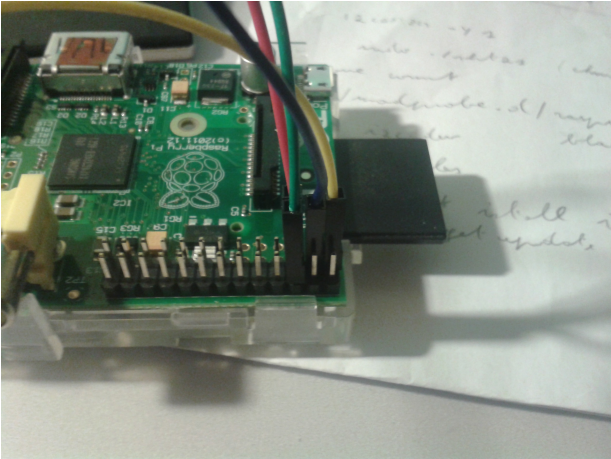
De jumpwires kan je normaal gezien nu in de connectoren plaatsen. Zoals je ziet of de foro moet kan 1 van de jumpwires geconnecteerd worden zonder deze te strippen. De foto hiernaast is genomen met het aanhechtpunt voor het klemmetje aan de achterkant.



Kleur	Omschrijving
Geel	Spannings-bron 3.3 v
Blauw	Data (SDA)
Rood	Clock (SCL)
Groen	Ground

Connecteren van de jumpwires of de Rpi

De volgende beschrijving er van uit dat de draden zijn geconnecteerd op de connector van de evaluatie-kit zoals beschreven hierboven (zelfde kleur en positie).

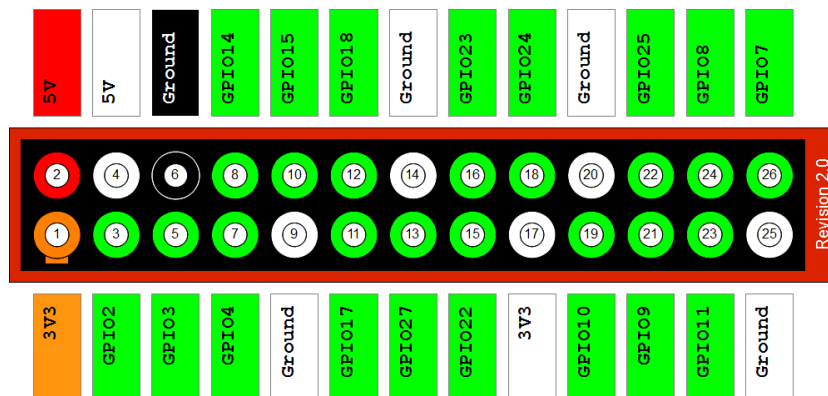


De draden dienen te worden geplaatst zoals geïllustreerd op de foto hiernaast.

Ter verduidelijking zie in de tabel hieronder de mapping naar de pin-nummering zoals gekend in de RPi-documentatie

Kleur	Pin	Omschrijving
Geel	1	Spannings-bron 3.3 v
Blauw	3	Data (SDA)
Rood	5	Clock (SCL)
Groen	6	Ground

Let wel, voor deze taak en voorgaande taak dien je nauwkeurig te werk te gaan om beschadiging aan de sensor te vermijden.



Geen pull-up weerstand nodig

Nota: Bij een klassieke i2c-setup die je normaal gezien een pull-up weerstand de plaatsen tussen de SDA – SCL en een spannings-bron omdat deze actief laag moet zijn.

De RPi heeft deze weerstand reeds intern ingebouwd.

Eerste test vanuit RPi

Als de vorige stappen zijn uitgevoerd zijn we klaar om een eerste test uit te voeren .

Test 1: Wordt de SHT21 gedetecteerd door de Rpi?

Een eerste test bestaat erin het commando te runnen dat een overzicht geeft van de i2c-devices die zijn aangesloten op de bus.

```
sudo i2cdetect -y 1
```

Als de SHT21 correct is aangesloten en de software-stappen zijn correct uitgevoerd moeten we het volgende resultaat hebben.

```
  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:  -- -- -- -- -- -- -- --
```

Deze tabel duidt aan dat de SHT21 is gedetecteerd (als slave-device) op de Rpi onder het adres 64 (40 in heximale codering).

Test 2: Meting uitvoeren via de RPi

Men kan ook via de command-line praten met de RPi en message doorsturen.

Het voorbeeld dat we hierbij gebruiken is het uitlezen van een configuratie-register:

```
i2cget -y 1 0x40 0xe7 b
```

Een i2c-commando bestaat altijd minstens uit 2 bytes (of meer) waarvan:

- de eerste byte het adres voorstelt
- de 2 de byte een commando (in dit geval opvragen van de inhoud van het user-register)

De 'b' in het commando geeft het formaat aan dat je wilt terugkrijgen in dit geval byte-data (zie ook man-paginas) in hexadecimaal formaat

De standaard (reset)-waarde die verkrijgt is:

```
0x0a
```

Later meer detail over het inhoud (binair equivalent *000 1010*) komt later nog aan de orde. Belangrijkste indicator zijn bit 7 (MSB) en bit 0 die de resolutie van de metingen aan geven (combinatie 00 betekent 12 bit voor RH en 14 bit voor T)

Raspberry Pi voorbereiden voor sensor-hub

Als de test-omgeving is opgezet en we hebben een sht21 kunnen connecteren kunnen we ons focussen op de setup van de sensor-hub.

De eigenlijke sensor-hub-applicatie is een java-applicatie, wordt gedeployed op een webcontainer (Tomcat, Jboss, Jetty, ...), houdt zijn data bij in een database en kan geconsulteerd worden via een rest-api (waarover later meer).

In de setup die we willen bereiken voert de RPi de volgende taken uit:

- Periodiek opvragen van metingen van de geconnecteerde sensors (in dit geval SHT21)
- Beheer van configuratie van de sensors
- Opslaan en aggregeren van de sensor-data in een database
- Het ondervragen van de sensor-data via een rest-api

Deze software (beheren en het ondervragen van sensors op de RPi) is geschreven in java.

De volgende componenten zijn nodig om het systeem:

- Java versie 7
- Apache Tomcat webserver
- PostgreSQL-database

Java installeren en configureren

Bij de standaard-installatie van een Raspbian-distributie is een jdk voorzien (huidige versie 7)

Om dit te testen kan je 2 volgende commando's uitvoeren:

- Versie van Java?

```
pi@raspberrypi ~ $ java -version
java version "1.7.0_40"
Java(TM) SE Runtime Environment (build 1.7.0_40-b43)
Java HotSpot(TM) Client VM (build 24.0-b56, mixed mode)
```

- JDK geïnstalleerd ipv enkel een JVM?

```
pi@raspberrypi ~ $ javac -version
javac 1.7.0_40
```

In je geen vergelijkbaar resultaat kan je deze installeren via apt-get

```
sudo apt-get install openjdk-7-jdk
```

Apache Tomcat installeren

Tomcat is de applicatie/webserver die we gaan gebruiken om de sensor-hub-functionaliteit te installeren en te deployen.

Deze kan je op de volgende manier installeren.

```
sudo apt-get install tomcat7
```

Bij de installatie van tomcat dien je een aantal configuratie-parameters aan te passen:
Dit wordt geconfigureerd in de file “/etc/default/tomcat7”

```
cd /etc/default  
vim tomcat7
```

Daarin pas je de volgende waarden aan:

```
JAVA_HOME=/usr/lib/jvm/jdk-7-oracle-armhf  
TOMCAT7_USER=root  
TOMCAT7_GROUP=root
```

De meest eenvoudige manier om de war-file te deployen is gebruik te maken van de admin-tool van tomcat.

Deze kan geïnstalleerd worden via het volgende commando:

```
sudo apt-get install tomcat7-admin
```

Na de installatie kan je nog niet in de management-console (zie ook later), hiervoor moet je nog een user toevoegen die gerechtigd is om de admin-console te gebruiken.

Dit kan door in de volgende file

```
sudo vi /etc/tomcat7/tomcat-users.xml
```

een user toe te voegen met de rol “manager-gui”.

```
<user username="system" password="raspberry" roles="manager-gui"/>
```

Uiteindelijk voer je een restart van Tomcat aan om deze configuratie te activeren:

```
sudo service tomcat7 restart  
sudo service tomcat7 stop
```

Postgresql (nog niet nodig)

Deze stap is optioneel aangezien we in deze fase een embedded (in jvm) database gebruiken (hsqldb)

```
sudo apt-get install postgresql
```


Applicatie bouwen (optioneel)

Om de applicatie te kunnen deployen moet de applicatie eerst worden gecompileerd en gepackaged in een war file.

De WAR-file is reeds beschikbaar via de volgende URL:

<https://www.dropbox.com/s/yhjr9xasajp8mi7/restapi.war>

Als je echter de meest recente versie wilt hebben kan je dit doen mits de volgende configuratie.

Installatie Maven

Hiervoor heb je (bovenop de vorige java installatie) Maven nodig.

Op RPi en andere debian-gebaseerde systemen zoals ubuntu kan je dit installeren met het volgende commando.

```
sudo apt-get install maven
```

Gedetailleerde instructies ter installatie van maven zijn te vinden op

<http://maven.apache.org/download.cgi>.

```
~$ mvn --version
Apache Maven 3.0.4
Maven home: /usr/share/maven
Java version: 1.7.0_51, vendor: Oracle Corporation
Java home: /usr/lib/jvm/java-7-openjdk-amd64/jre
Default locale: nl_BE, platform encoding: UTF-8
OS name: "linux", version: "3.11.0-15-generic", arch: "amd64", family: "unix"
```

Source downloaden (optioneel met git-clone)

Download en unzip <https://github.com/bartvoet/sensors/archive/master.zip>

Optioneel maar je kan natuurlijk ook git gebruiken om deze source code te clonen

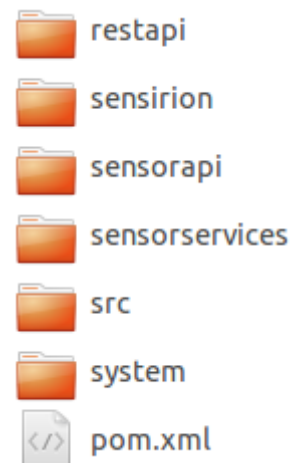
```
git clone https://github.com/bartvoet/sensors.git
```

Build uitvoeren

Eenmaal de voorgaande stappen zijn uitgevoerd kan je overgaan tot de eigenlijke compilatie en build van het systeem.

Dit kan door via de command line naar de root van het (maven-) project te navigeren (zie printscreen).

In deze folder vindt je een file genaamd pom.xml die de meta-info bevat gerelateerd aan deze software (build-instructies, dependencies ...)



```
mvn clean package
```

Na de installatie vindt je in het pad *sensors/restapi/target* een *restapi.war*.

Deze war-file zal je in de volgende stap moeten gebruiken voor de installatie op de RPi.

Sensor-hub installeren (war-file)

Als de voorgaande instructies aangaande installatie Java en Tomcat hebt opgevolgd ben je klaar om de applicatie op de RPi te deployen.

De eenvoudigste (maar niet de enige) manier om te deployen is via een web-interface van Tomcat (meer bepaald de administratie-interface).

Bij een out-of-the-box-installatie kan je deze applicatie via de url “`http://{host}:8080/manager/html`” gebruiken.

Je vervangt host door de hostname (of ip-adres) van je Rpi, Tomcat zal by default op poort 8080 van de RPi luisteren.

Bij het aanloggen zal je wel een user en password moeten ingeven, als je de installatie-instructies hebt gevolgd zal je met de combinatie system/raspberry moeten kunnen inloggen.

The screenshot shows the Tomcat Web Application Manager interface in a web browser. The address bar shows the URL `192.168.0.206:8080/manager/html`. The page title is "Tomcat Web Application Manager". Below the title is a message bar with "Message:" and "OK". The main content area is divided into two sections: "Manager" and "Applications". The "Manager" section has links for "List Applications", "HTML Manager Help", and "Manager Help". The "Applications" section contains a table with columns: Path, Version, Display Name, Running, Sessions, and Commands. The table lists four applications: `/`, `/host-manager`, `/manager`, and `/restapi`. Each application has a "Start" button and a "Stop" button. The `/manager` application has a "Reload" button and an "Undeploy" button. The `/restapi` application has a "Start" button and a "Stop" button. Below the table is a "Deploy" section with a form for deploying a WAR file. The form has fields for "Context Path (required)", "XML Configuration file URL", and "WAR or Directory URL". There is a "Deploy" button. Below the form is a "WAR file to deploy" section with a "Select WAR file to upload" button and a "Bestand kiezen" button. The "Bestand kiezen" button is highlighted, and the text "restapi.war" is visible next to it. There is also a "Deploy" button.

Path	Version	Display Name	Running	Sessions	Commands
<code>/</code>	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
<code>/host-manager</code>	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
<code>/manager</code>	None specified	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
<code>/restapi</code>	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Deploy
Deploy directory or WAR file located on server

Context Path (required):
XML Configuration file URL:
WAR or Directory URL:

WAR file to deploy

Select WAR file to upload Geen bestand gekozen

Vanonder op deze webinterface heb je een button “Bestand kiezen”, als je deze hebt geselecteerd verschijnt er een dialoog waar je de eigenlijke war-file moet selecteren

The screenshot shows a dialog box titled "WAR file to deploy". It has a "Select WAR file to upload" button and a "Bestand kiezen" button. The "Bestand kiezen" button is highlighted, and the text "restapi.war" is visible next to it. There is also a "Deploy" button.

WAR file to deploy

Select WAR file to upload restapi.war

Na deze selectie druk je op de button “Deploy” en de applicatie zal worden gedeployed op de Tomcat Webcontainer, het deployen hiervan kan een 5-15 seconden in beslag nemen.

Om een test uit te voeren gaan we over naar het volgende hoofdstuk.

Gebruik Sensor-hub voor metingen uit te voeren

De sensor-hub-applicatie bestaat uit 2 onderdelen:

- Een asynchroon process dat metingen uitvoerd op een bepaalde interval en deze naar een database wegschrijft
- Een restapi (http) die de data (geregistreerd door het asynchroon process) visualiseerd onder json-formaat.

Gebruik van REST-api om SHT21 te ondervragen (direct of database)

```
curl http://192.168.1.137:8080/restapi/api/sensors/1/overview
```

Gebruik van REST-api om SHT21 te configureren

```
curl http://192.168.1.137:8080/restapi/api/sensors/1/configuration
```