Building a The Things Network Outdoor Gateway

Tools needed

- Soldering iron
- Solder
- Side-cutter
- Small screw driver

The parts

- PCB by Charles Hallard. Reference https://github.com/ch2i/iC880A-Raspberry-PI
- IMST iC880A-SPI – LoRaWAN Concentrator 868MHz
- Raspberry Pi 3 Model B
- microSDHC Card 8-32 GB Class 10
- 2 Nylon Spacer 8mm incl. nuts
- 2x20p tall female header for Raspberry Pi
- 4 Resistor 330 Ω Axial 0207 0.25 W
- Optional 2 Resistor 10 kΩ Axial 0207 0.25 W
- Optional 2 Grove connector
- 4 LEDs 3mm blue, white, red, green
- Female connector 40p (1x 20pin, 2x 3pin)
- Male connector (2x 2pin)
- 2 Jumper/shorting blocks
- DC barrel connector, 5.5 mm 2.1 mm
- 7v-28v to 5v Dc-dc Step Down Buck Power Supply Module 3a Fixed Output
- 1N4001 diode
- RF Elements Stationbox Alu
- Pigtail UFL Hirose / IPX to N female
- Lightning protection N male - N female, Bulkhead, 5-6GHz
- Optional cable with N male 1m
- Optional NoName 150cm antenna 868Mhz or alternative
- Optional PoE Injector IEEE802.3af
- Optional PoE Splitter IEEE802.3af

Remark: this manual is describing v1.2 incl. Charles updated board v1.1. The main difference to the description v1.0 is the cut-out for the PI connectors and the outdoor case.

The PCB has several options we will NOT use in this workshop, but you can later on use them if needed - some of those are already listed in the parts list and can be soldered to be future proof:

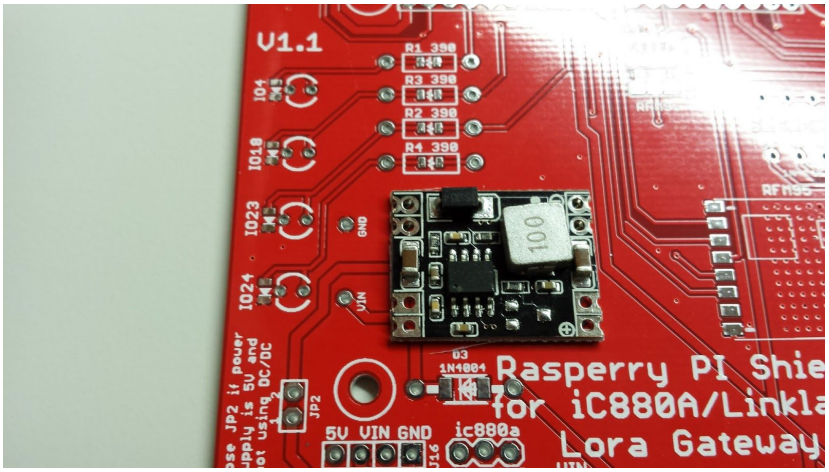
- Creating a single channel gateway with a RFM95 chip and D1, D2 and D3 or a dual channel gateway with the LinkLabs module (though the LinkLabs module is no longer produced).
- Adding Grove connectors to add Grove sensors by http://www.Seeedstudio.com , you can add your own Grove sensors to give your gateway extra local functionality. The I2C pull up resistors R5 and R11 will be mounted.

- We will mount the LEDs but they are only for test purposes. If you want to make them functional, you have to find a way to mount them in the casing somewhere.
- We will use the DC barrel connector for power and you can choose whether to power it directly with 5v (out of active PoE) or with more v (out of passive PoE) by setting a jumper.

We will mount all the parts EXCEPT for the 2x20p Raspberry PI connector and the DC-DC step down on the side were the parts are referenced on the silk screen (LEDs on the left, Power connector at the bottom)

Mount the DC-DC converter

Mount the Step Down converter at the frontside. Check the connections, there are several versions around. Input voltage on the right, output on the left, check the polarity! You may have to mount it on the backside. Some step down converters need to be adjusted. With a voltmeter, measure the output and regulate to 5.1V.



You can solder the converter with some small pieces of wire, which were cut off when mounting the resistors.

Mount the DC connector

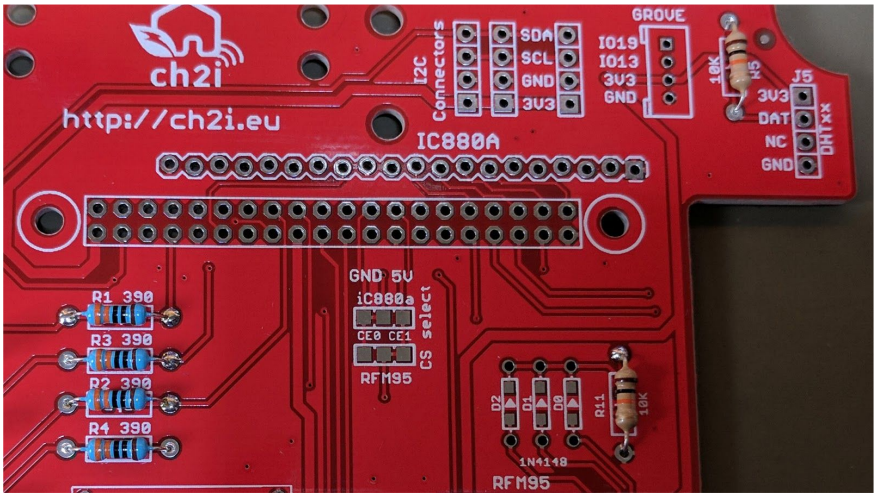
Solder the three pins of the DC connector as shown

Mount the diode?

D3 is used to protect the system for reversed polarity. IF YOU WANT TO TEST YOUR BOARD FIRST WITH A 5V ADAPTER, DO NOT MOUNT THIS DIODE YET! You can put jumpers on JP1 and JP2 to get the power to the PI and the radio board.

Mount the resistors

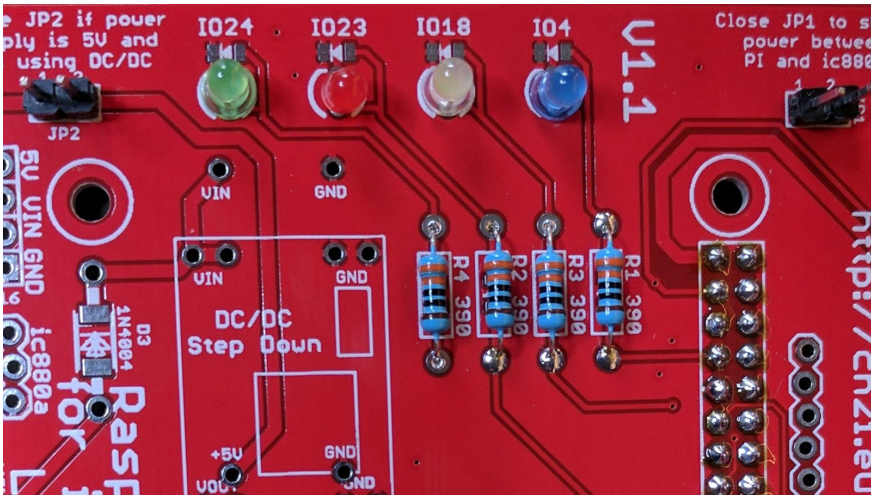
Mount the 4 resistors 330 ohm (orange – orange – brown) R1, R2, R3 and R4. (on the PCB it is 390, no problem here, just brighter LEDs).



Mount the 2 resistors 10K (brown-black-orange) R5 and R11.

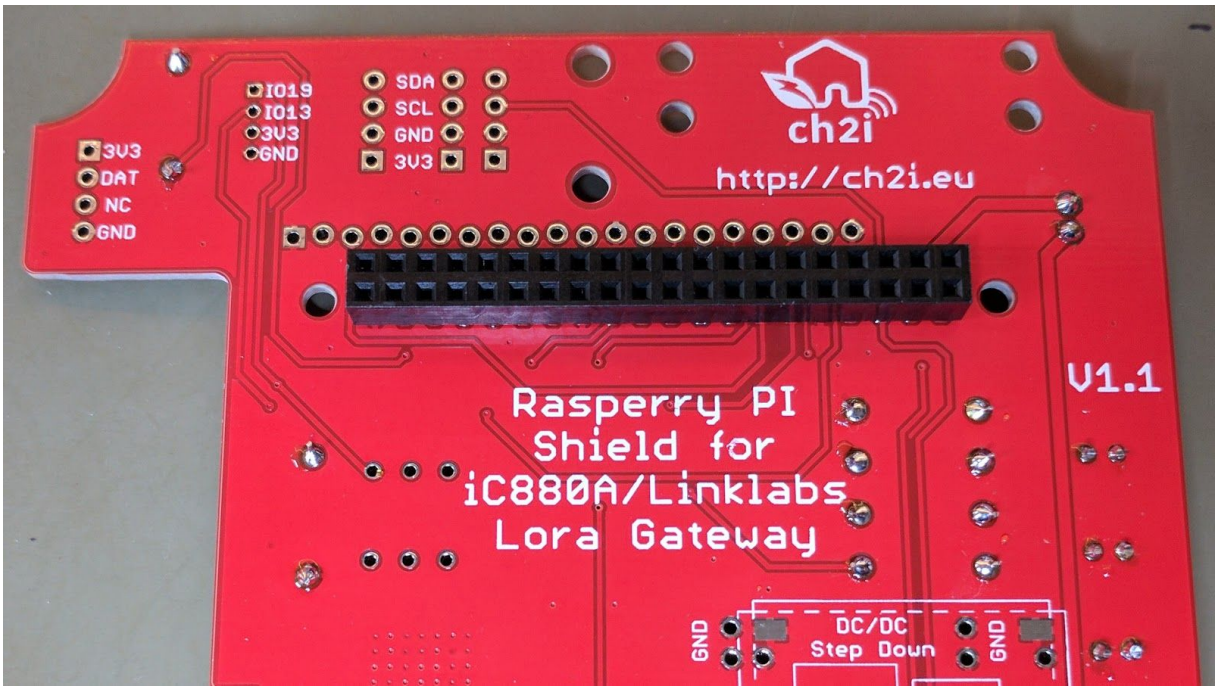
Mount the LEDs

Mount the LEDs. They have different colours (Red, White, Blue, Green). IO24 is green, IO23 is red, IO10 is white, IO4 is blue. The Led's cathode is the shortest wire and should be at the outside of the board.



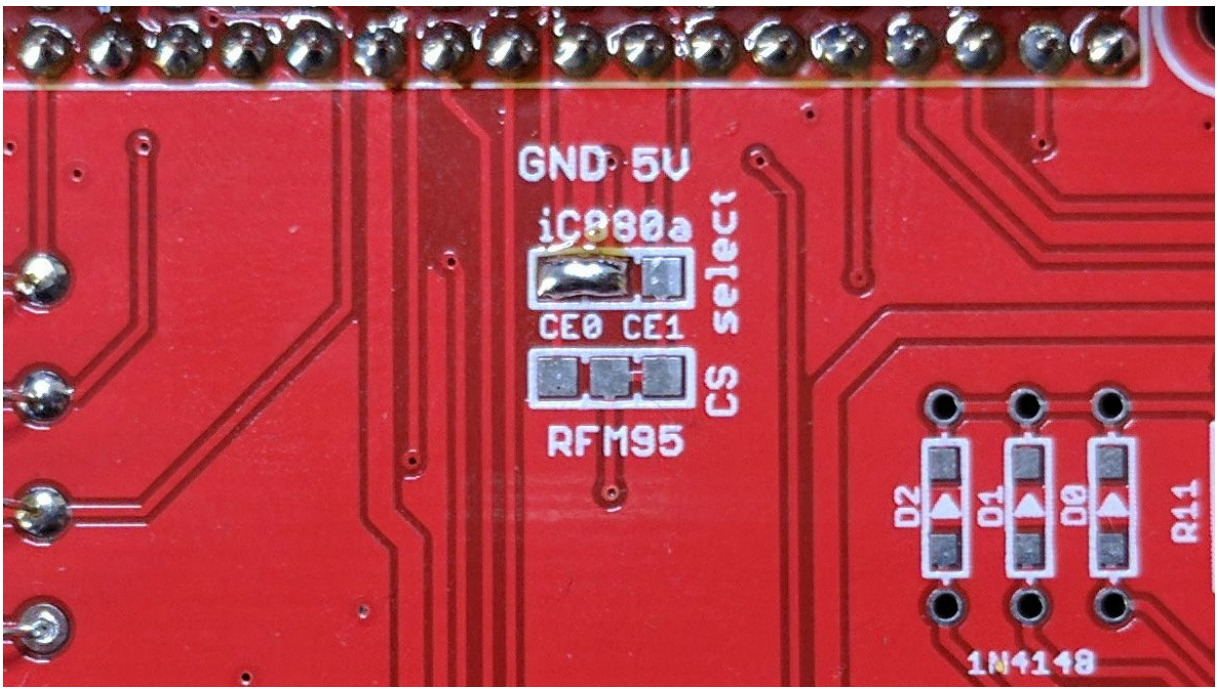
Mount the Raspberry Pi connector 2x20p

Mount the connector ON THE BACKSIDE.



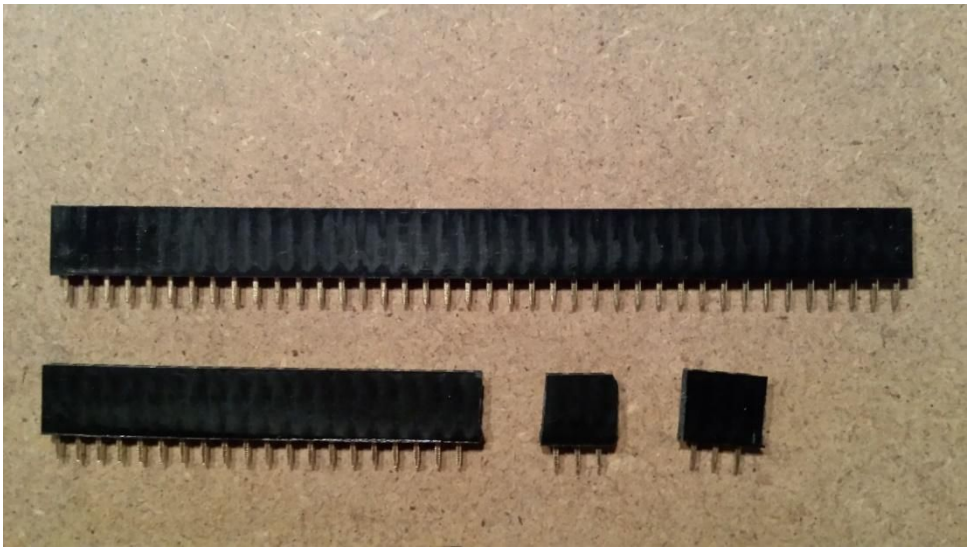
(Solder a bridge on ‘CE0’ on the ic880a select connector)

Put a little solder on the CE0 connection of the ic880a select connector to make a ‘bridge’. Make sure there is a good connection! On the newer v1.1 boards, the CE0 connection is already made, so you may skip this step!



Cut the 40p connector

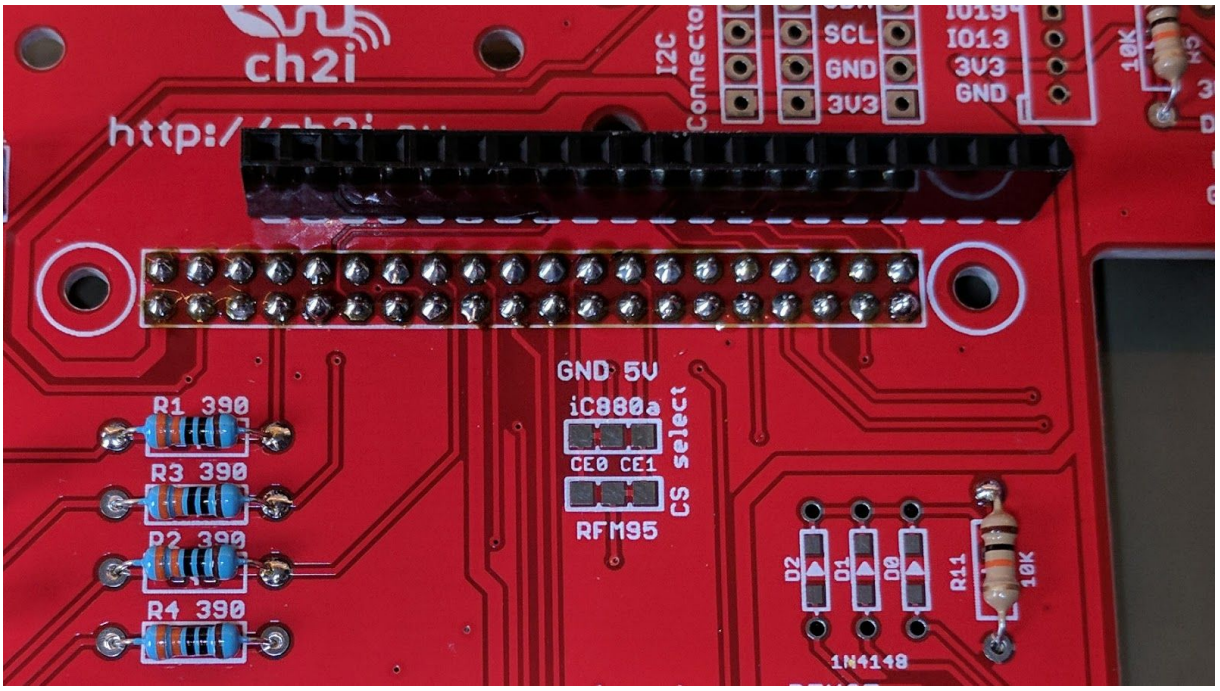
The three connector for the imst ic880a board can be cut from one 1x40p connector. 2x 3p connector and 1x 20p connector is needed. The strip is easily cut on a pin, this will destroy one pin, but we need in total only 26 (not 40).



Solder the ic880a connectors

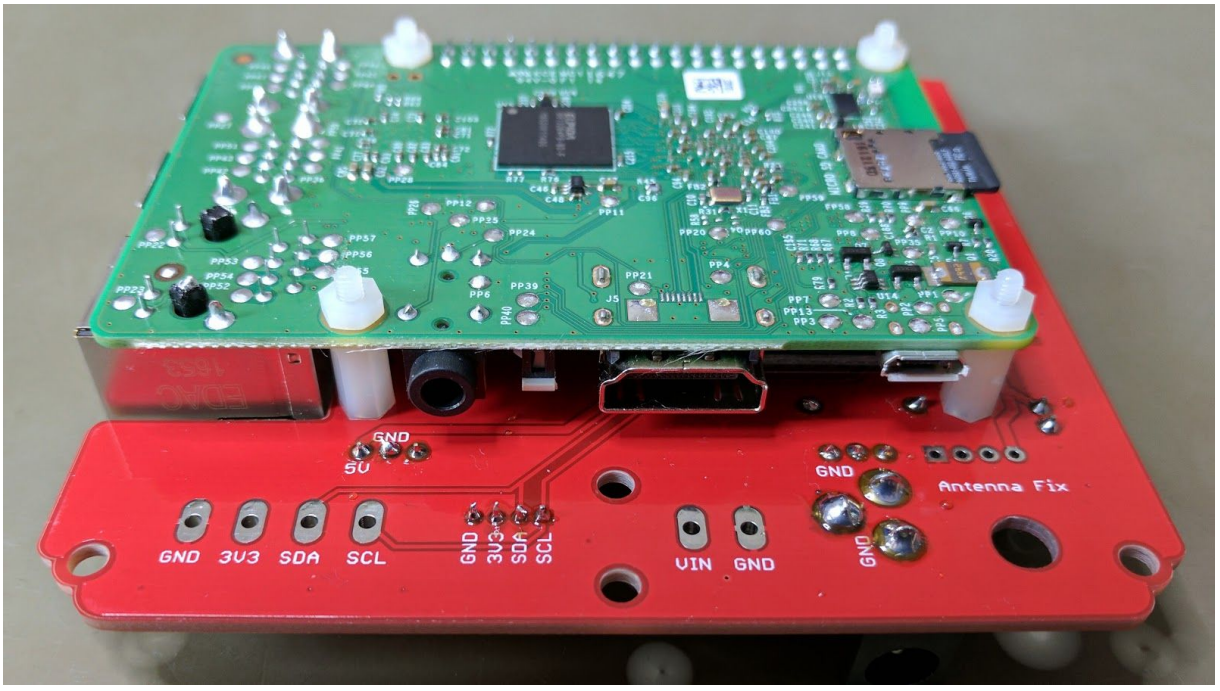
Solder first the two small ones. Solder just one pin, and adjust them to be straight and tight to the PCB. Then solder the other two pins.

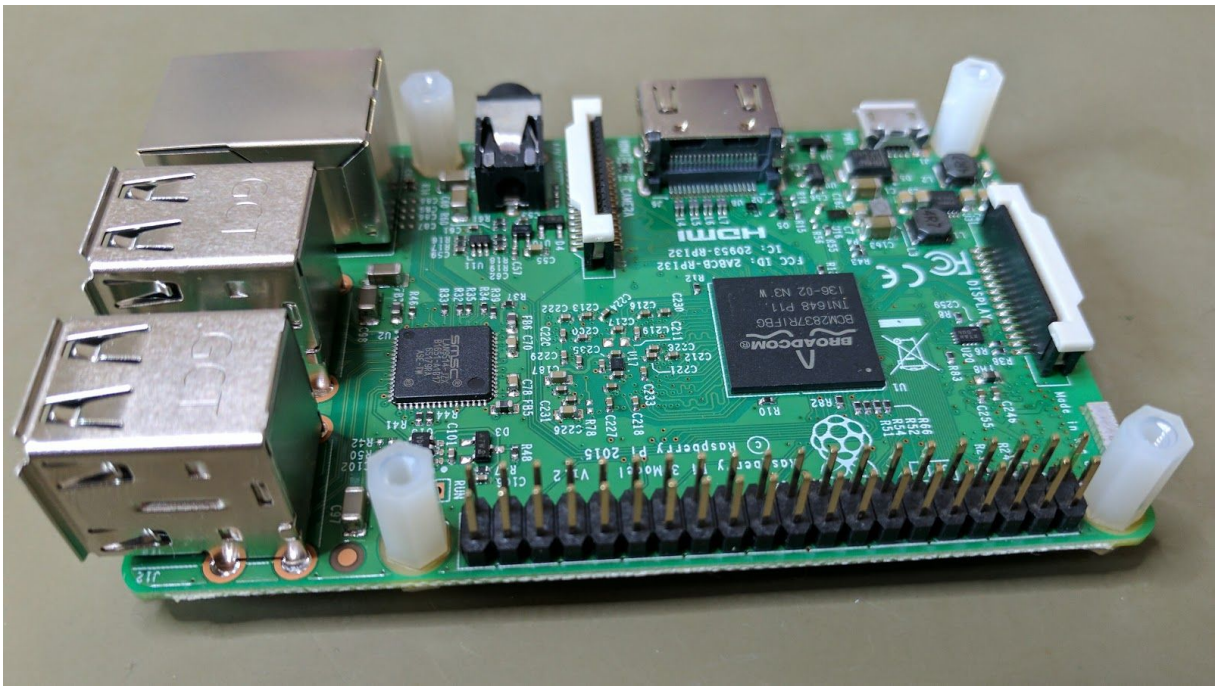
The 20 pin connector can be placed, solder again just one outside pin, adjust and solder the rest. Be careful for any short circuits here. It can destroy your boards!



Mount two distance holders on the PCB. They will fit tight, so you have to put some pressure on them. Fix them with a nut on the other side. The PI can be mounted with two screws.

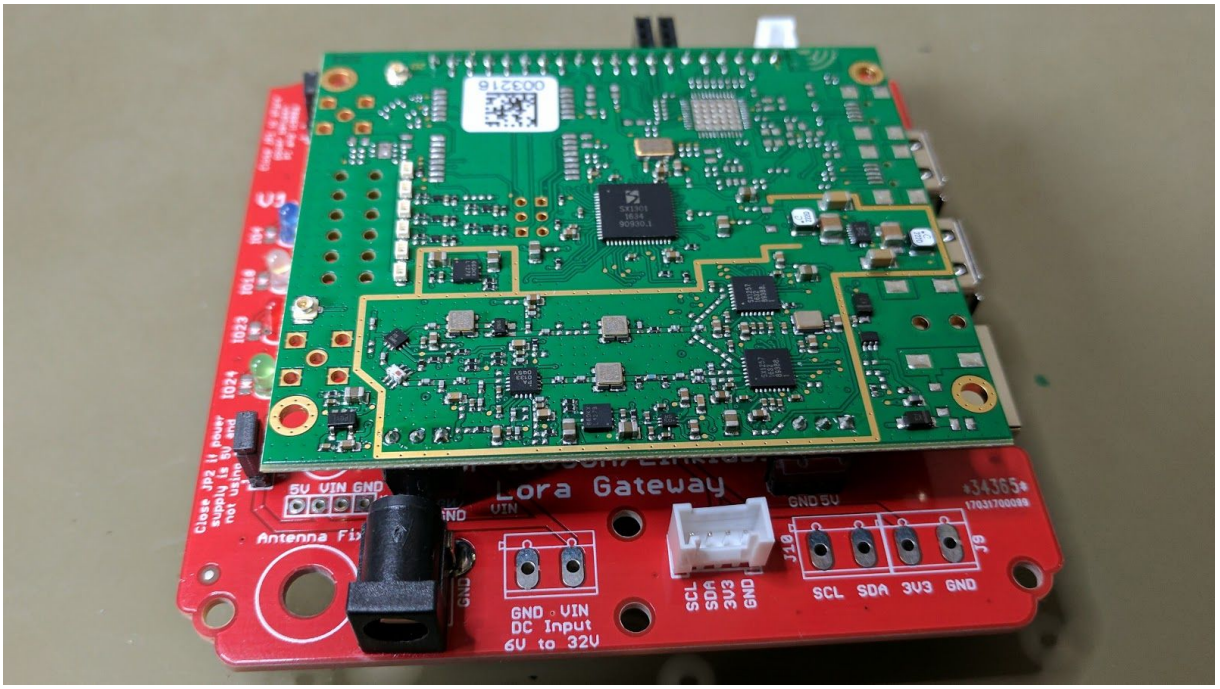
Mount the Raspberry PI to the underside of the board. Just after the software test we will mount the ic880a board.



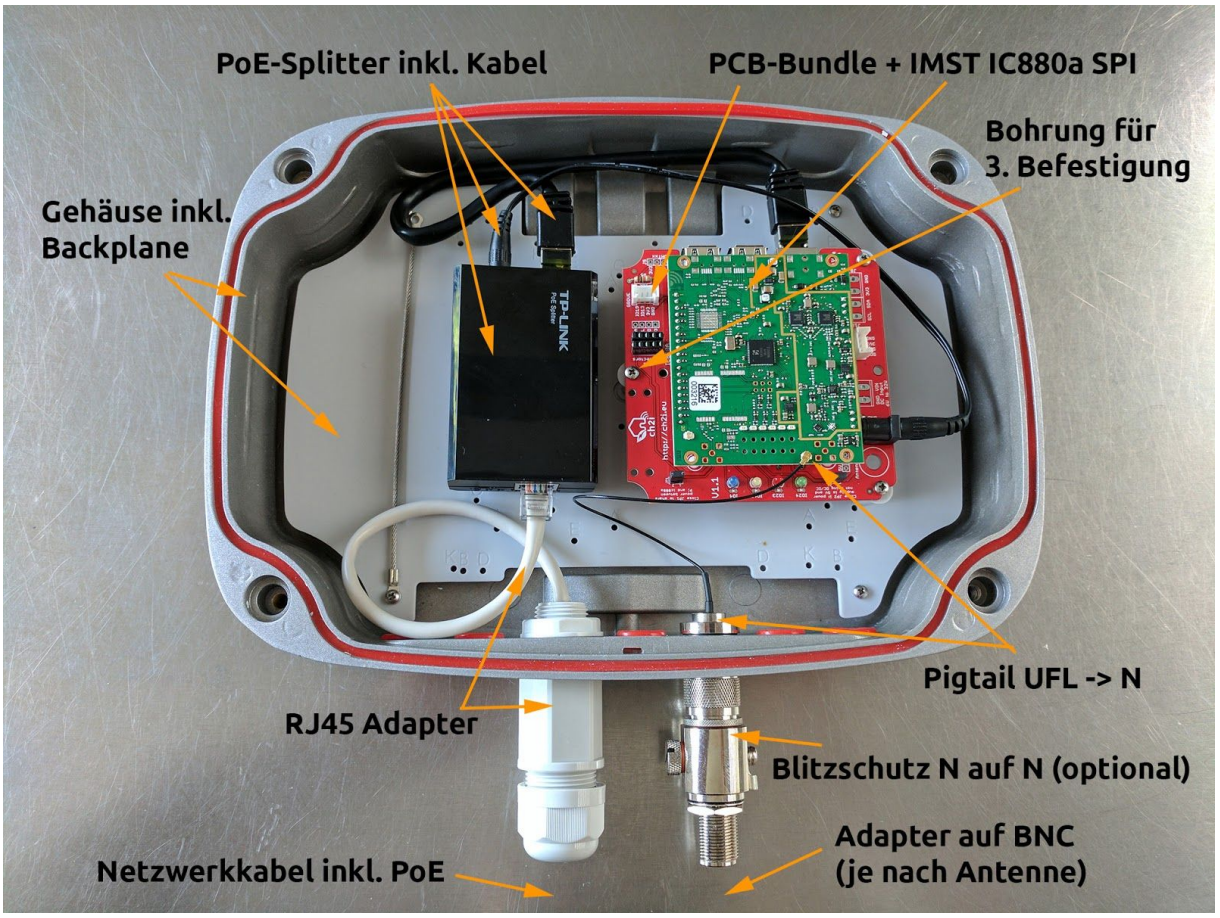


Mount the ic880a board. Be careful! This is a very EMD sensitive (and expensive) board. So ‘discharge’ yourself or even better use ESD equipment. Place the board onto the three connectors. Check visually if you mounted it the right way!

Mount the pigtail to the antenna connector. Look at the picture to identify the right connector. Near pin 20 is another connector, this one is for the future GPS extension and we will not use this one, but the one near the 12 drilled holes. The connector is very small and can break easily.



Drill a 3rd hole (2mm) in the middle of the backplane to screw down the pcb on the backplane of the case. Glue velcro on the backplane and fleece to the back of the poe-splitter to make a removeable connection. Connect the power from the poe with the DC barrel connector and make sure the output of the poe is set to 5V. Use a short cat5/6/7 cable to connect the network port from the raspberry pi to the poe splitter. Attach the pigtail to the UFL connector on the concentrator board (use the right connector as shown in the picture!) and tighten the N connector to the case.



Screw the rj45 adapter to the case and connect it to the poe splitter. Now on the outside you can optional use some lightning protection and adapters to the connector from your antenna and connect the rj45 to your local network for internet access. Do not power up the gateway without an antenna connected!

Software

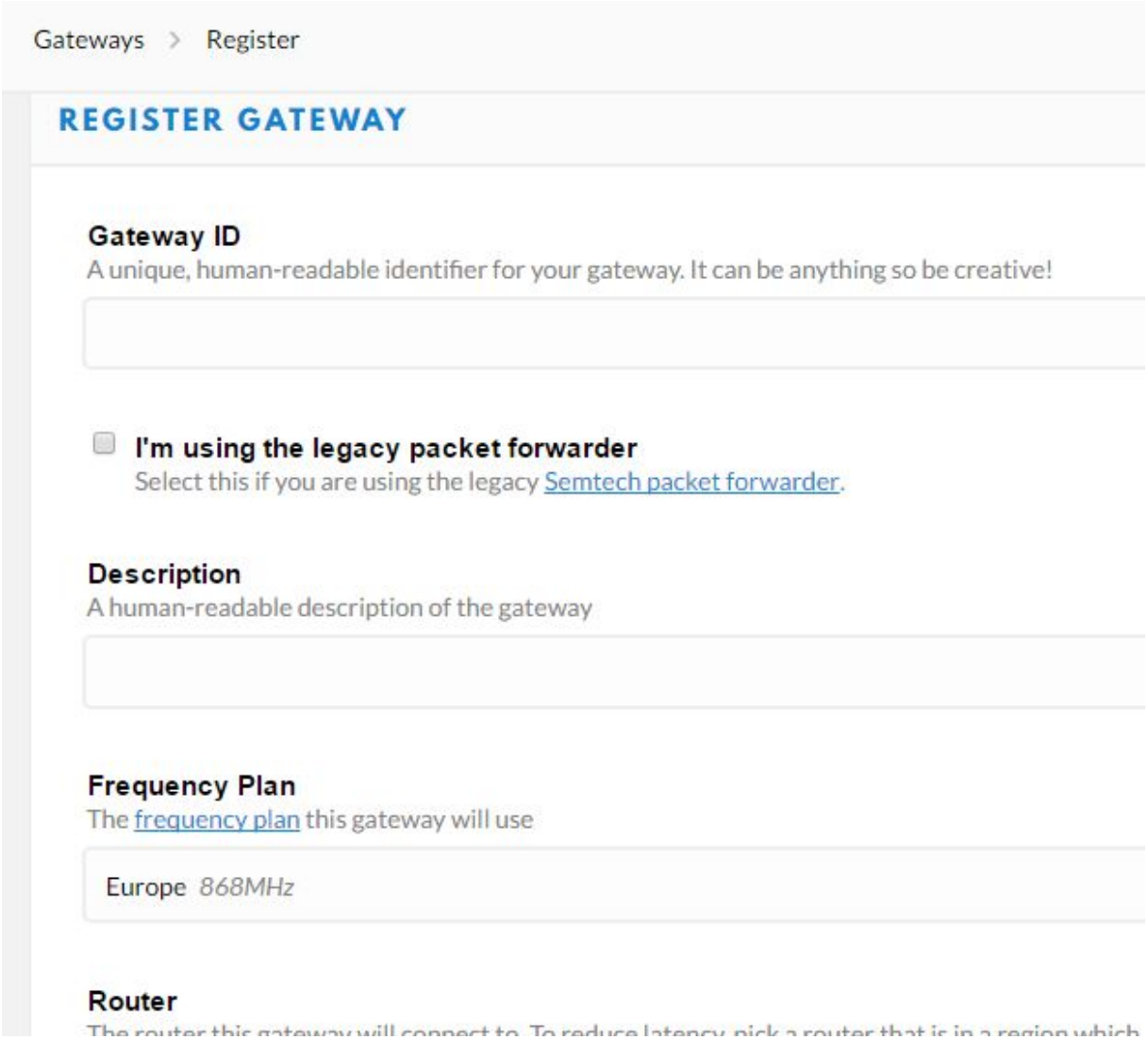
You will need:

- Laptop with internet connection (MAC/Windows/Linux) and SD card read/writer
- Ethernet Cable connection to connect the Raspberry PI to the Internet
- The Things Network account
- Github account with a SSH key
- Resin.io account

Creating The Things Network account

Go to https://account.thethingsnetwork.org/register

Fill in the form to create an account:

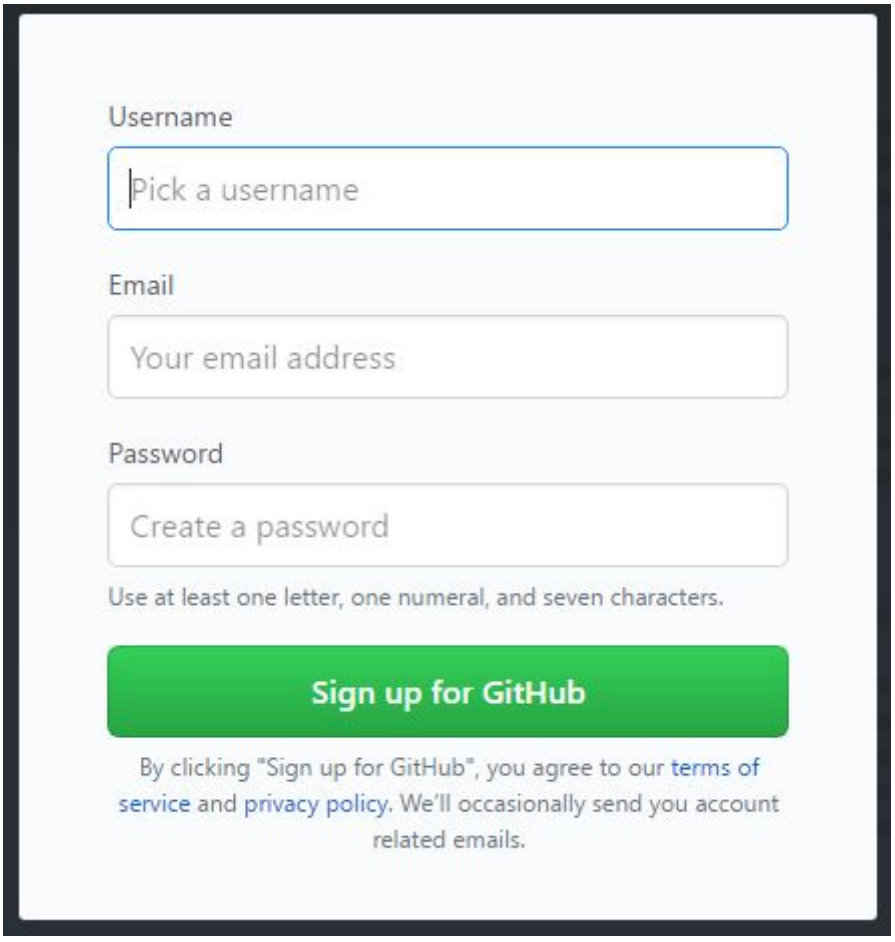


Give your gateway a name, select the European frequency plan, leave the Semtech box unchecked and place your gateway on the map where you want to install your gateway.

Creating Github Account

To upload the proper configuration we will use Github. An account is needed. If you have already an account login with this account and go to the next step.

Go to http://www.github.com and sign up:

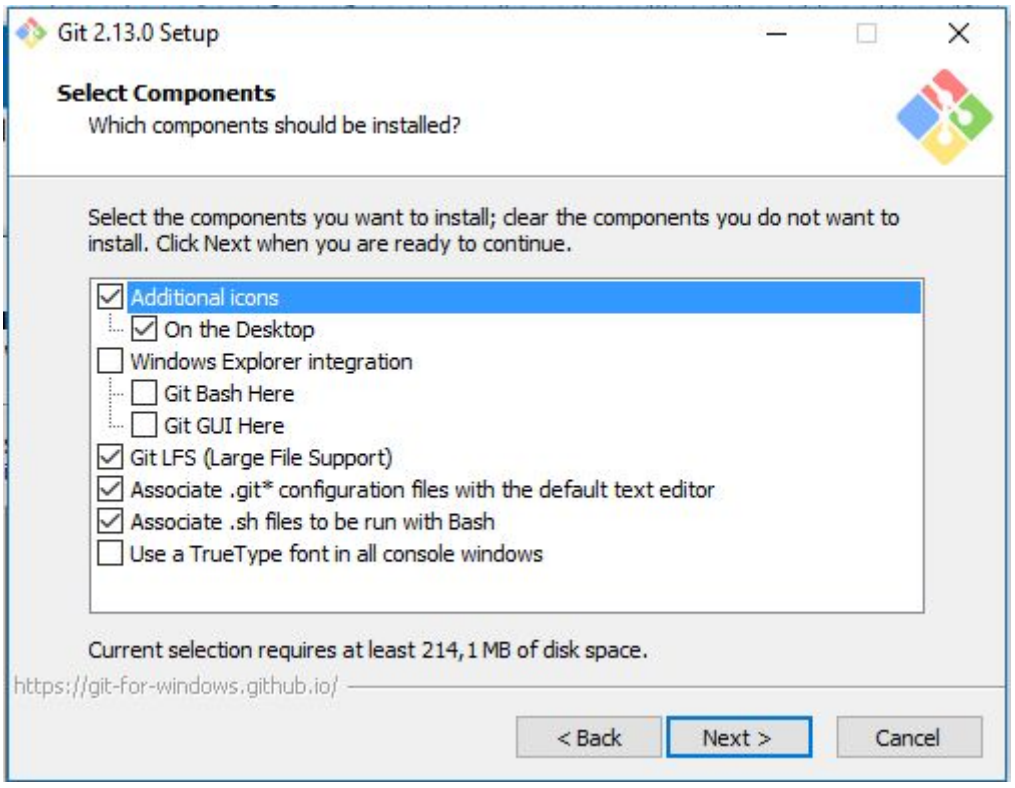


After setup you have to verify your email address.

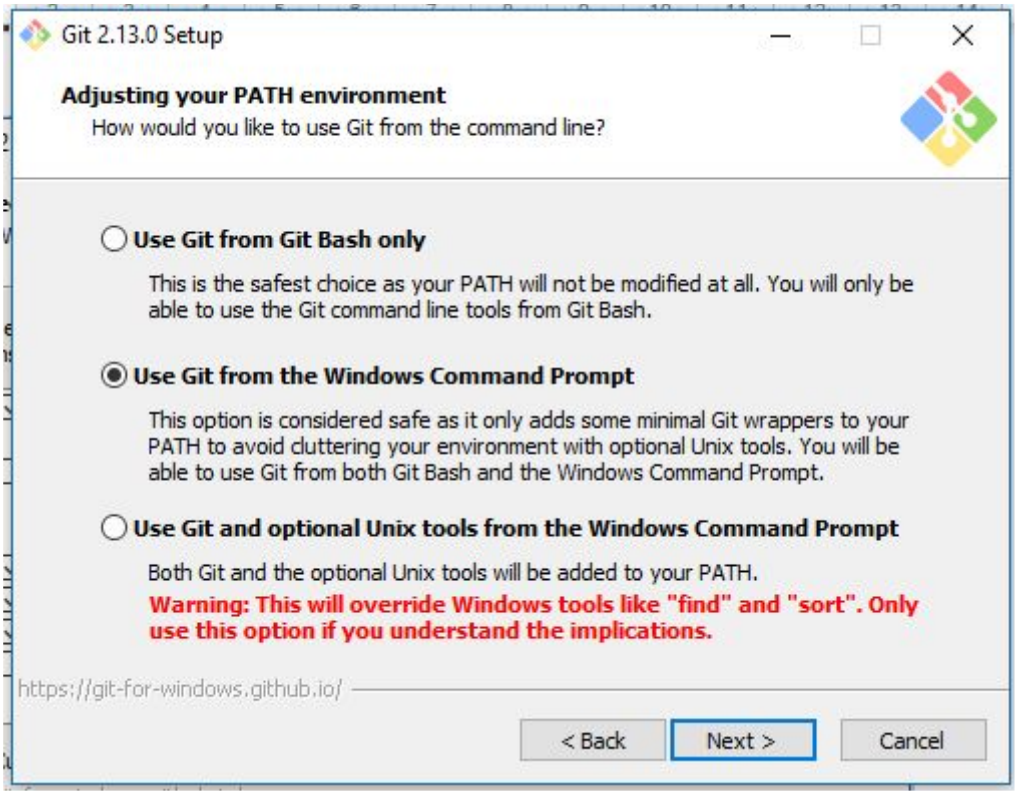
Setup GIT client

Download the client from https://git-scm.com/downloads , choosing your right platform

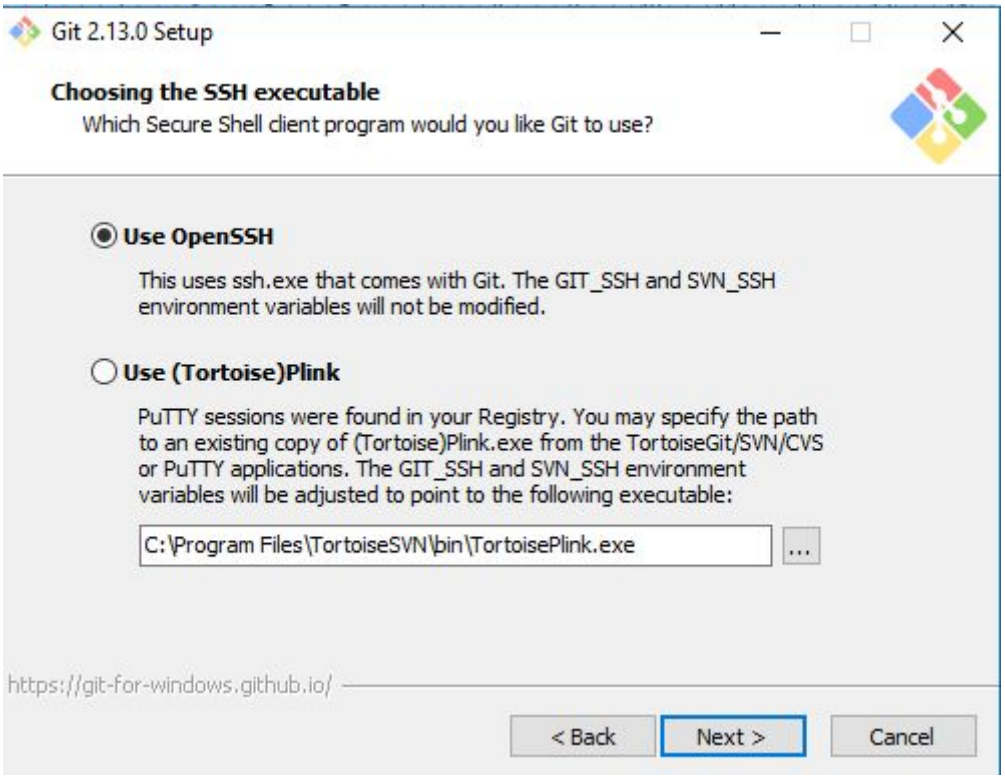
Use the default install settings, but add the additional icons:



Use Git from the Windows Command prompt (other platforms may show similar options.

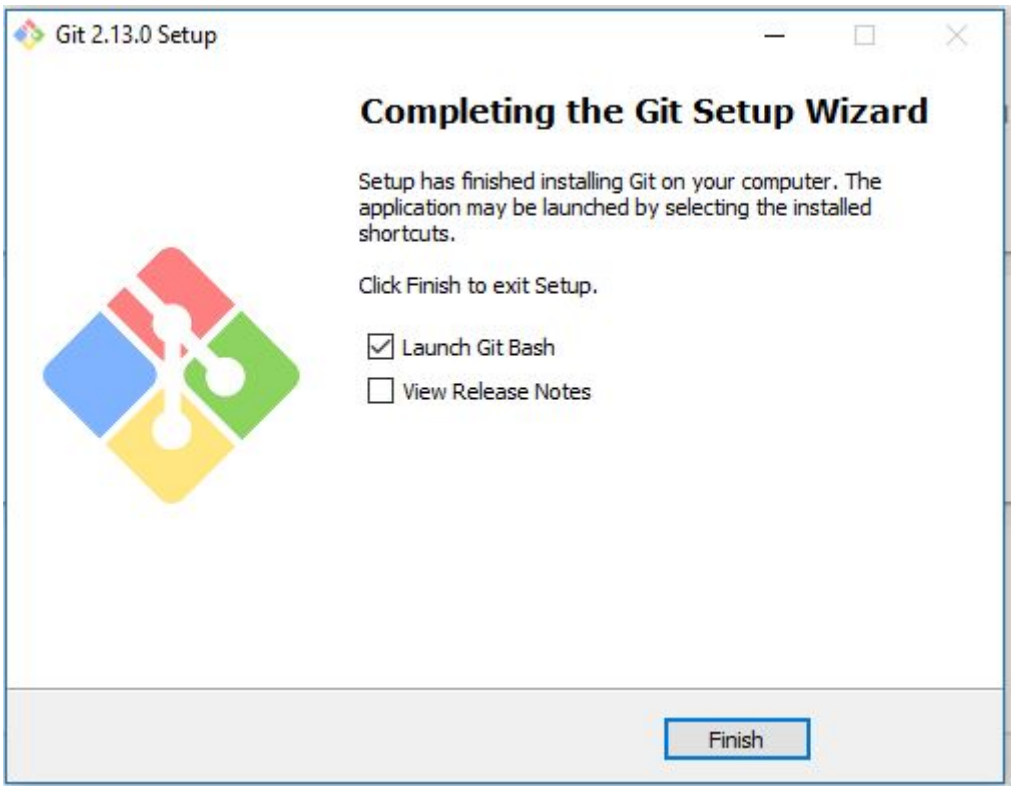


Use openSSH:

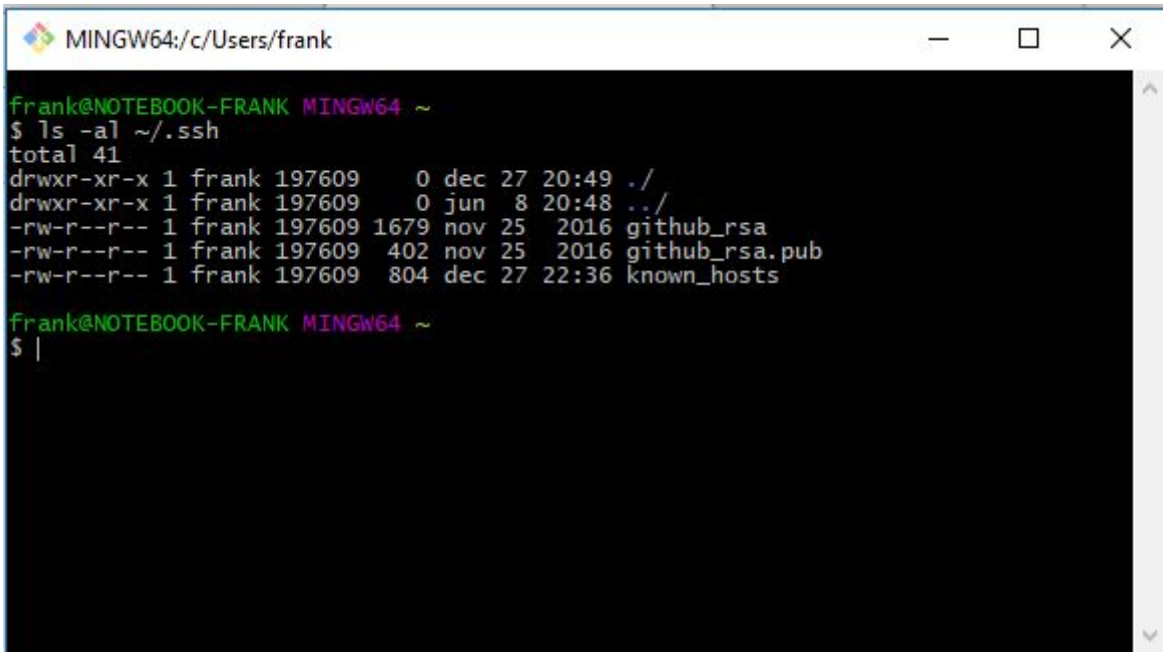


- In the upcoming options choose the defaults:
- Use the OpenSSL library
- Checkout Windows-Style, commit UNIX-style line endings
- Use minTTY
- Enable file system caching, enable git credential manager

Choose Install, at the end launch git bash:



Bash will open, you may open bash on your favourite platform

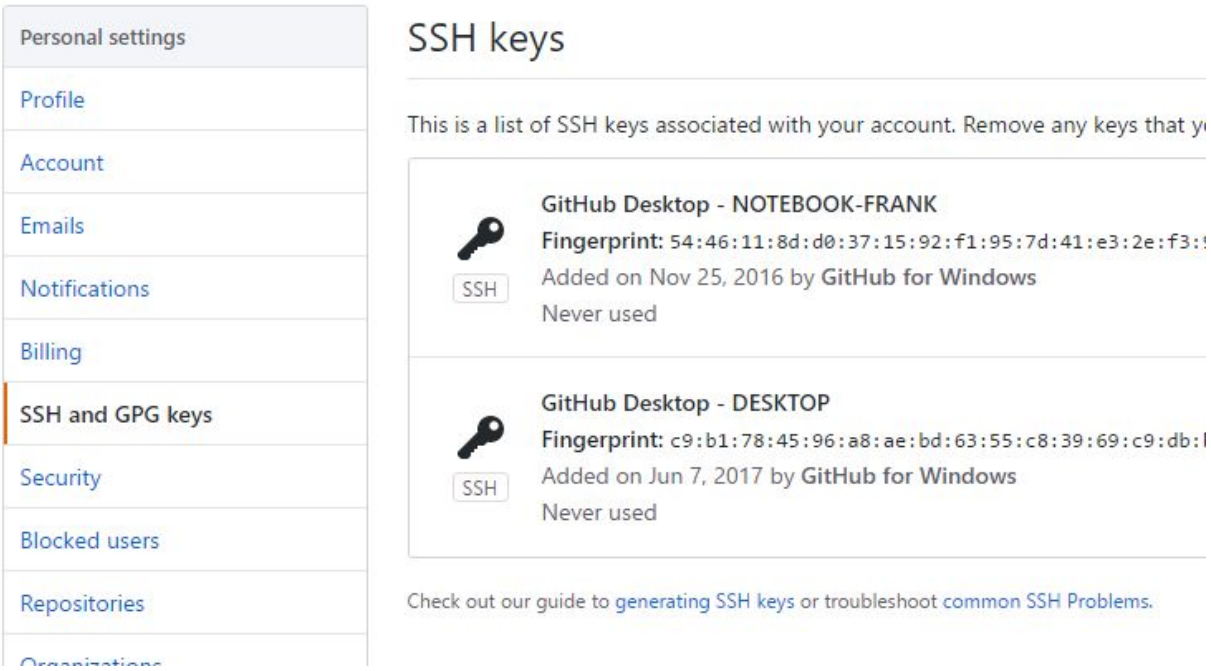


Show your SSH key(s):

- `ls -al ~/.ssh`

the file listed as .pub is your public SSH key. This one should be the same as your listed github key.

You can check this by clicking on your upper right account logo, choose settings, SSH and GPG keys.



If you do not have a SSH key, you can create one in the console:

- `ssh-keygen -t rsa -b 4096 -C "mygithubmailaddress@xxx.yy"`
- `eval $(ssh-agent -s)`
- `ssh-add ~/.ssh/id_rsa`

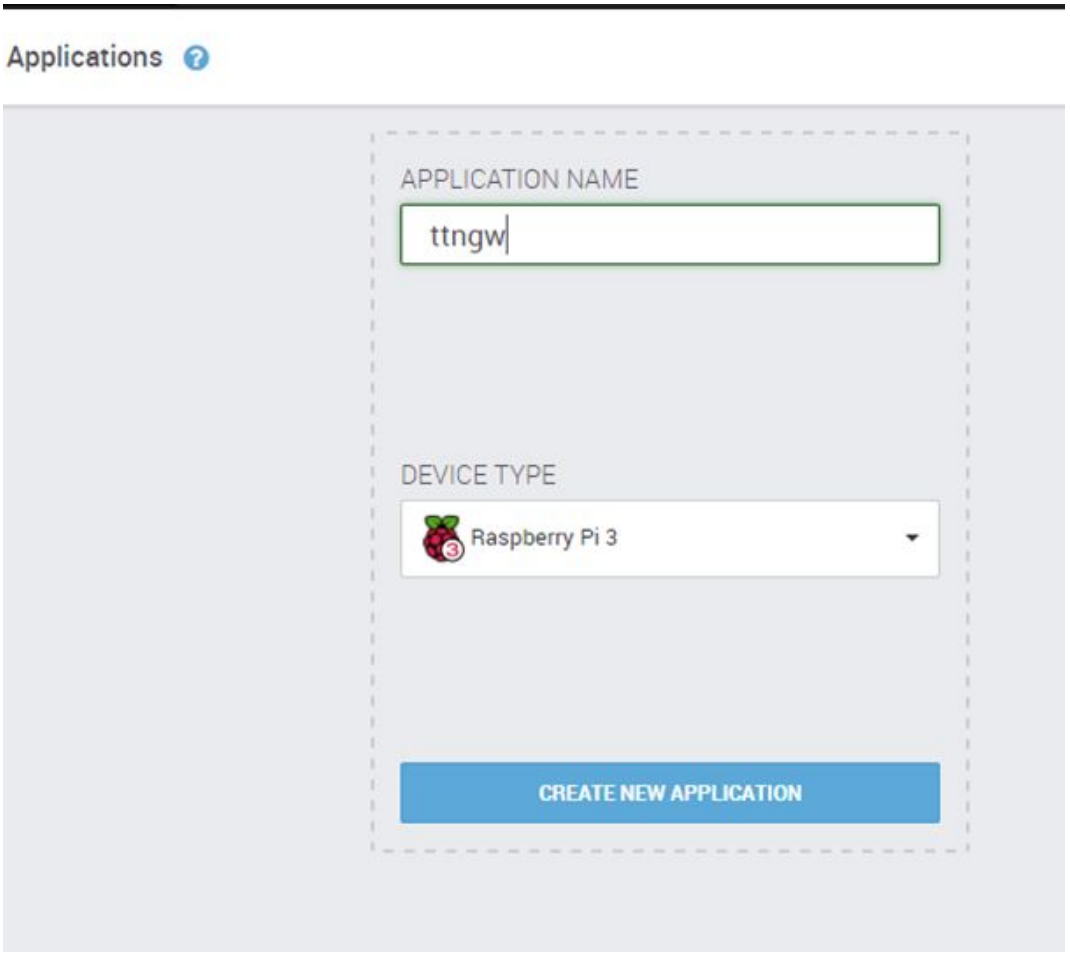
Now you can re-issue the ls command again to check your key:

- `ls -al ~/.ssh`

Create Resin.io account

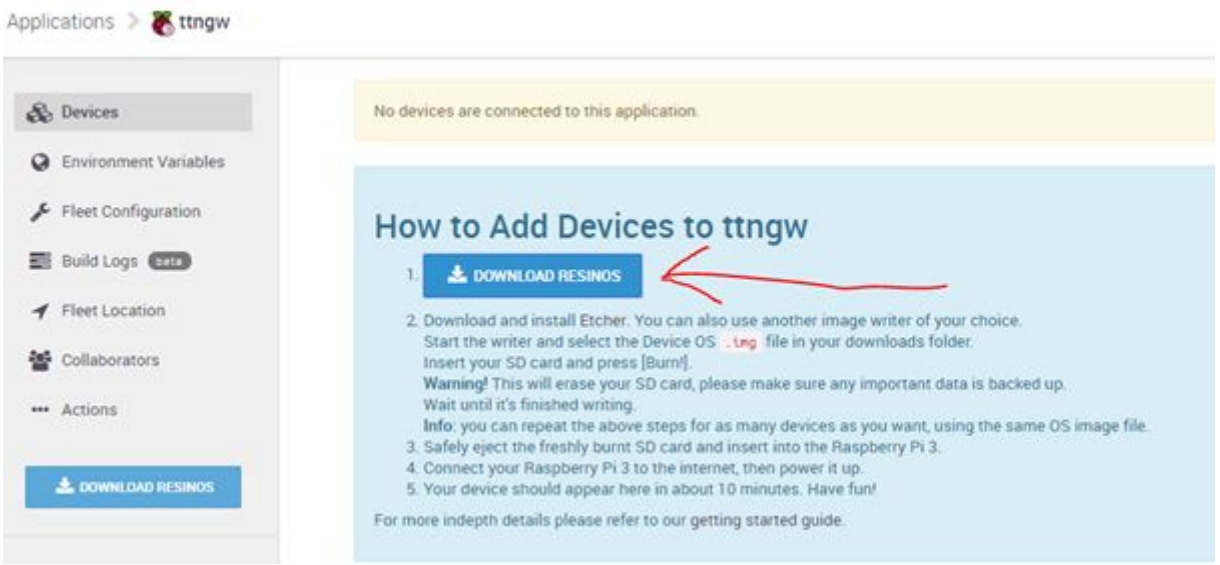
At this moment Resin.io is free for up to five devices per user. At http://resin.io create an account (sign up).

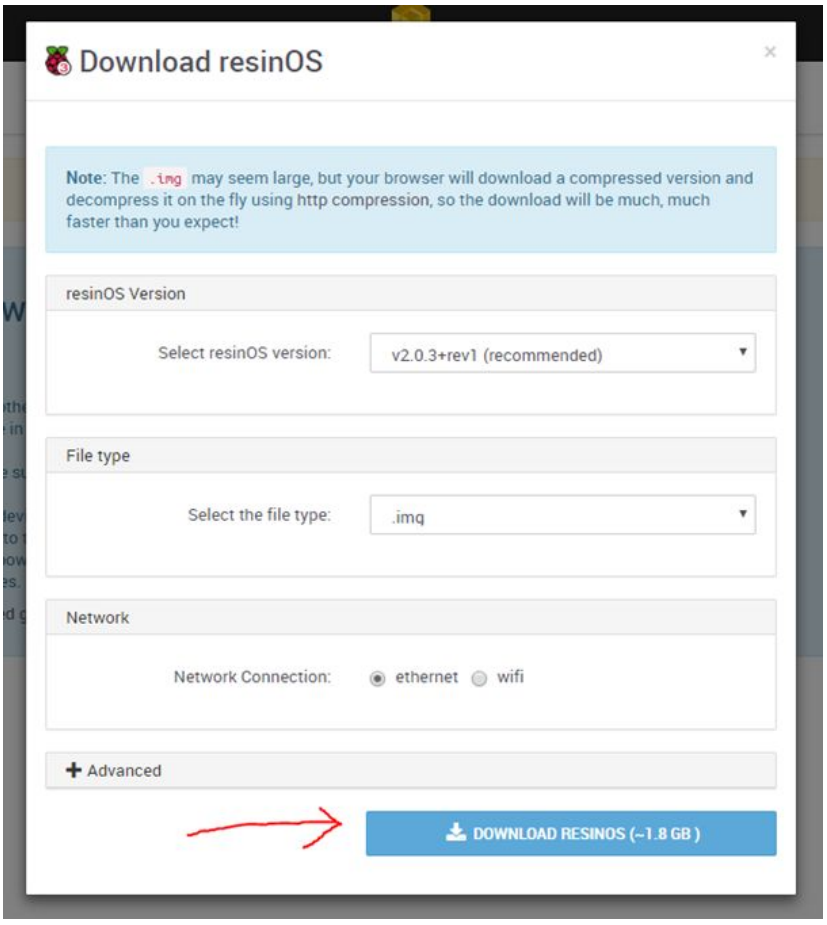
Now you can create your first application, name your application ttngw (or whatever you like) and select Raspberry PI 3.



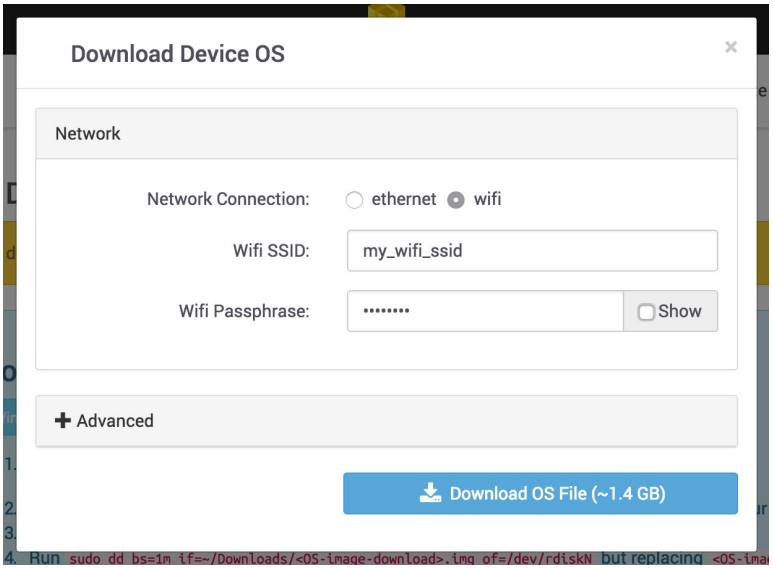
Now download the RESINOS

Leave the settings to default

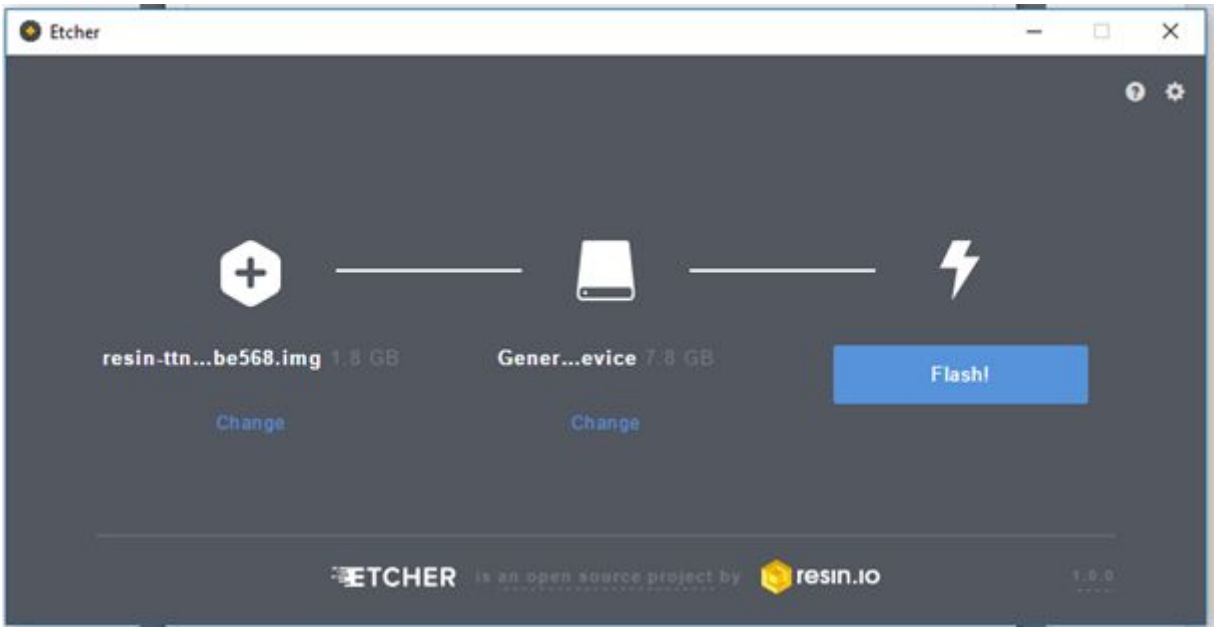




If you want to use a wifi connection, you have to enter the wifi credentials before the image is created:



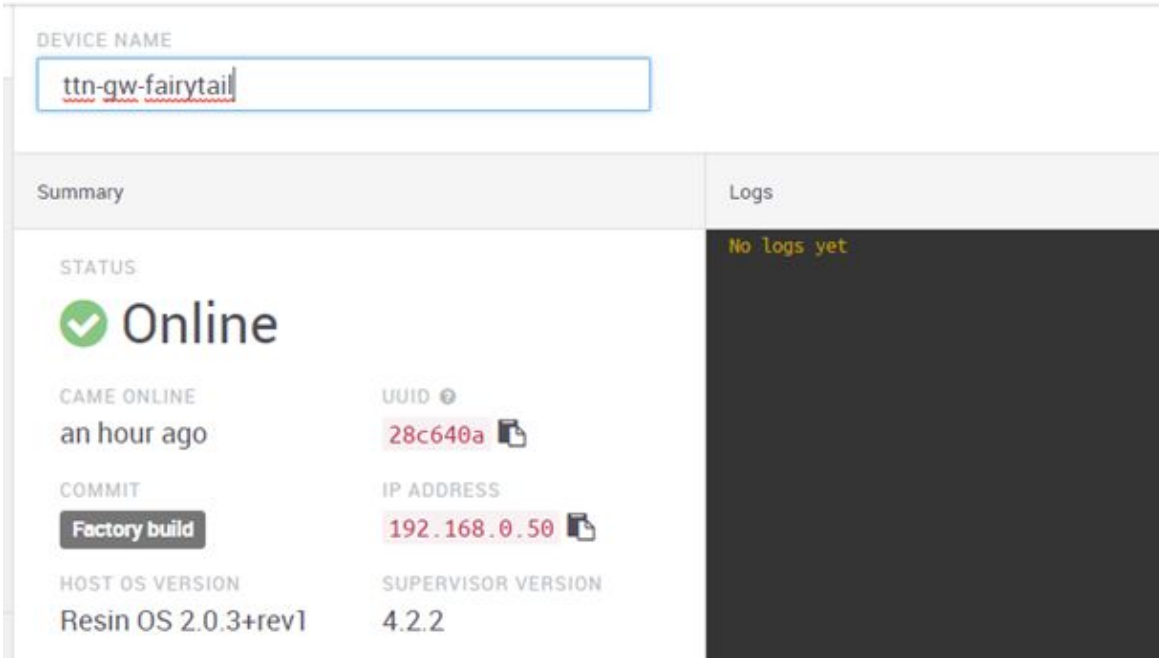
You can use the shown link to 'Etcher' to install the flash writer application and flash your SD card with the previous download resinos image. You may use win32diskimager as well.



Now you may boot your Raspberry PI, connected with the Internet, with the created SD card.

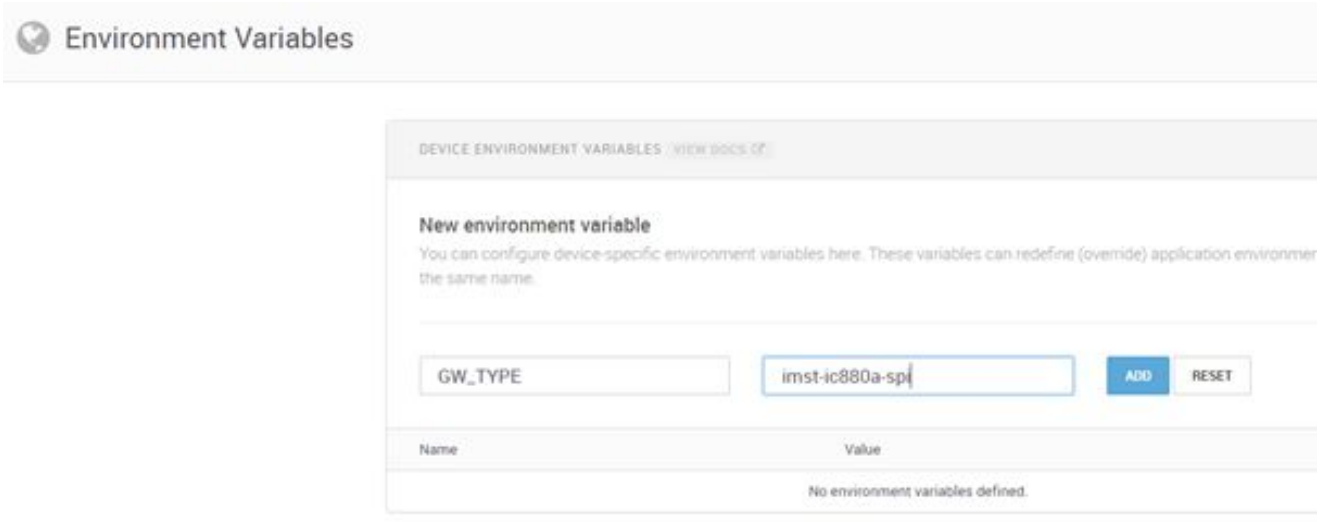
After some minutes your device will appear in the resin.io interface:

Give your device his own name (use location in this name to select this gateway from others you may install later on).



Create the environment variables

Now create the environment variables. You can get the info from your ttn gateway registration page. There are two types ‘application’ (for all your gateways), and ‘device’ for specific parameters per device.



Name	Value	Environment
GW_TYPE	imst-ic880a-spi	application
GW_CONTACT_EMAIL	yourname@yourdomain.com	application
GW_ID	The gateway ID from the TTN console	device
GW_KEY	The gateway KEY from the TTN console	device
GW_RESET_PIN	11	application

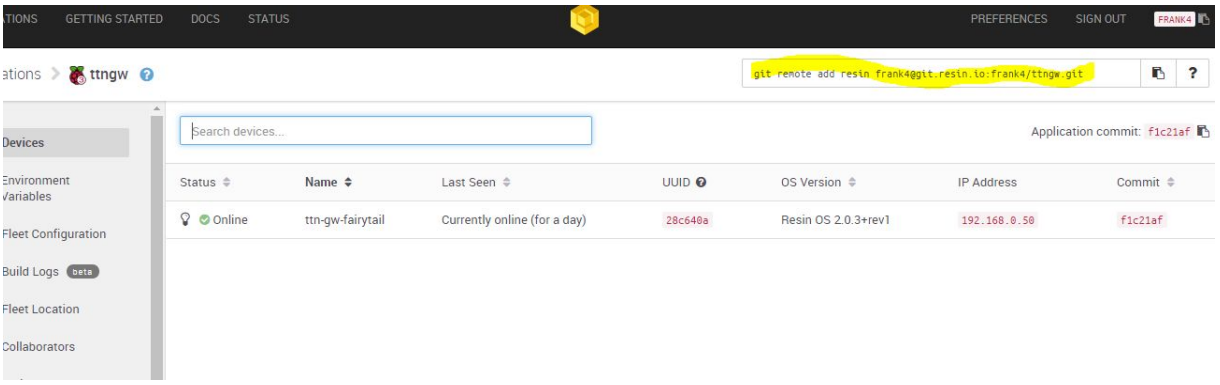
Clone the Repo

On your computer, clone this git repo. For example in a terminal on Mac or Linux type, or in git bash on Windows:

- `git clone https://github.com/jpmeijers/ttn-resin-gateway-rpi.git`
- `cd ttn-resin-gateway-rpi/`

Now, type the command that you'll see displayed in the edit control in the upper-right corner of the Resin devices dashboard for your device. This command "connects" your local directory to the resin GIT service, which uses GIT to "receive" the gateway software from TTN, and it looks something like this:

- `git remote add resin youraccount@git.resin.io:youraccount/yourapplication.git`



Add your SSH public key to the list at https://dashboard.resin.io/preferences/sshkeys. You may need to search the internet how to create a SSH key on your operating system, where to find it afterwards, copy the content, and paste the content to the resin.io console. Most easily copy your github key by using the button.

Type the following commands into your terminal to "push" the TTN files up to resin.io:

- `git add .`
- `git commit -m "first upload of ttn files to resin"`
- `git push -f resin master`

What you'll now see happening in terminal is that this "git push" does an incredible amount of work:

- It will upload a Dockerfile, a "build script", and a "run script" to resin
- It will start to do a "docker build" using that Dockerfile, running it within a QEMU ARM virtual machine on the resin service.
- In processing this docker build, it will run a "build.sh" script that downloads and builds the packet forwarder executable from source code, for RPi+iC880A-SPI.
- When the build is completed, you'll see a unicorn ҉ ASCII graphic displayed in your terminal.
- Now, switch back to your device dashboard, you'll see that your Raspberry Pi is now "updating" by pulling the Docker container from the resin.io service. Then, after "updating", you'll see the gateway's log file in the window at the lower right corner. You'll see it initializing, and will also see log output each time a packet is forwarded to TTN. You're done!

Common problems

- SPI bus not enabled in RASPI-CONFIG
- CE880 bridge not soldered (v1.0 board only)
- SD Card problems: use good quality SD card (class 10), reinstall card if you have still problems

here

- Power problems, use a good power supply!

Additional information

- The original instruction from JP Meijers: https://github.com/jpmeijers/ttn-resin-gateway-rpi
- Creating SSH keys (mac, linux, Windows): https://help.github.com/articles/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent/
- Creating a Resin.io image: https://docs.resin.io/raspberrypi3/cpp/getting-started/
- Support from the community: https://www.thethingsnetwork.org/forum/

Antenna

For test usage you can use the small inside antenna. You can connect outside antenna's like:

- http://webshop.ideetron.nl/GP901C
- http://webshop.ideetron.nl/SD868L

You can connect the antenna with a SMA connector or use an adapter like this:

Credits

- Gonzalo Casas on the iC880a-based gateway
- Ruud Vlaming on the Lorank8 installer
- Jac Kersing on the Multi-protocol packet forwarder
- Ray Ozzie on the original ResinIO setup
- The Team at resin.io
- JP Meijers for the resin.io install walk through
- Pictures and text: Caspar Armster and Frank Beks