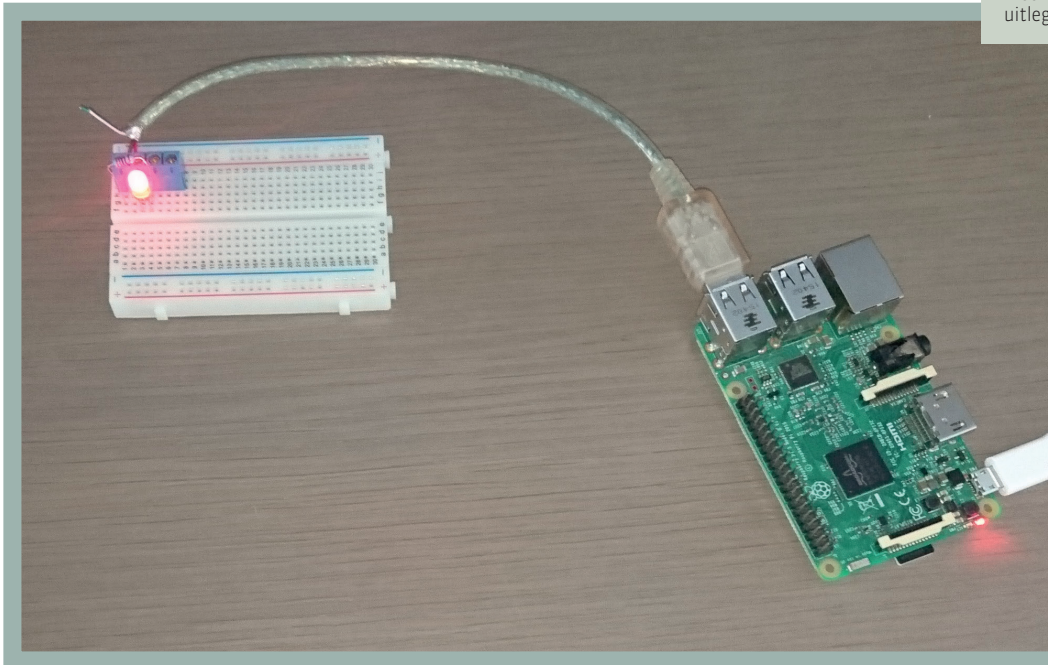


LISTING UITLEG

- > Start nieuwe regel
- regel met spatie na afbreking
- regel zonder spatie na afbreking

Meer informatie over de listing-uitleg vind je in de inhoudsopgave.



< Met hub-ctrl schakel je de USB-poorten van de Raspberry Pi aan en uit.

Arduino's als Chinese klonen hebt en je ze allemaal op dezelfde manier wilt kunnen identificeren, bijvoorbeeld via een udev-regel.

Installeer het programma met:

```
> sudo apt install ftdi-eeeprom
```

Creëer dan voor de te flashen Arduino een bestand `arduino_ftdi_USB.conf` met de volgende inhoud (pas deze aan je situatie aan):

```
> vendor_id=0x0403
> product_id=0x6001
```

```
> max_power=250 # Max. power
consumption: value * 2 mA.
```

```
> manufacturer="Arduino"
> product="Arduino Nano"
> serial="A700fmiq"
> use_serial=true
```

Daarna flash je deze gegevens naar de USB FTDI EEPROM van de Arduino met:

```
> sudo ftdi_eeeprom --flash-
eeeprom arduino_ftdi_usb.conf
```

Overigens kun je met dezelfde tool ook het EEPROM van de Arduino uitlezen en naar een bestand schrijven. Dat gaat als volgt:

```
> sudo ftdi_eeeprom --read-eeeprom
arduino_ftdi_usb.conf
```

In het configuratiebestand zet je dan een regel met de bestandsnaam, waarin het EEPROM moet worden geschreven:

```
> filename=arduino_nano_ftdi_
eeeprom.img
```

Bekijk zeker ook de man-pagina van `ftdi_eeeprom` voor de betekenis van de verschillende parameters in het configuratiebestand. <

Download de software met:

```
> git clone https://github.com/
codazoda/hub-ctrl.c.git
```

Ga daarna in de directory van het project en compileer het programma:

```
> cd hub-ctrl.c
> gcc -o hub-ctrl hub-ctrl.c -lUSB
```

Nu kun je de USB-poorten van je Pi uitschakelen met:

```
> sudo ./hub-ctrl -h 0 -P 2 -p 0
```

En terug inschakelen doe je met:

```
> sudo ./hub-ctrl -h 0 -P 2 -p 1
```

Overigens werkt dit niet bij alle modellen van de Raspberry Pi. We hebben dit uitgeprobeerd en kwamen tot de volgende vaststellingen:

- Pi Zero en Pi Zero W: werkt niet, want deze modellen hebben geen interne USB-hub.
- Pi 1 Model B: werkt niet.
- Pi 1 Model B+, 2 en 3B: werkt.
- Pi 3B+: werkt, maar je dient de optie `-h 1` in plaats van `-h 0` te gebruiken.

Let op: je schakelt alle USB-poorten tegelijkertijd uit, dus als je een toetsenbord of een WiFi-adaptor hebt aangesloten, schakel je die ook uit. En zorg dat je niet per ongeluk de optie `-P 1` in plaats van `-P 2` gebruikt, want dan schakel je

de ethernetpoort uit, die immers ook op de interne USB-hub is aangesloten...

UDEV-REGELS VOOR ARDUINO

Elk serieel USB-apparaat dat je op je pc aansluit, krijgt in Linux een apparaatpad, zoals `/dev/ttyUSB0` of `/dev/ttyUSB1`. Als je meerdere Arduino's tegelijk aansluit, krijgt het eerste apparaatpad `/dev/ttyUSB0`, de tweede `/dev/ttyUSB1`, enzovoorts. Maar wat als je een specifieke Arduino een apparaatpad als `/dev/arduino_nano` wilt geven? Dat kan...

Sluit eerst je Arduino aan en kijk met `lsusb` vóór en nadat je de Arduino aansluit wat het vendor ID en product ID is. Dat zie je in de uitvoer van `lsusb` in de vorm `0403:6001`, waarbij 0403 het vendor ID en 6001 het product ID is. Dat is in dit geval van een officiële Arduino Nano. Gebruik je een Chinese kloon, dan zullen de waardes verschillen. Als je meerdere Arduino's van hetzelfde type ook wilt onderscheiden, dien je hun serienummer te weten. Dat vraag je op met:

```
> sudo -v -d0403:6001
```

Het serienummer krijg je na `iSerial` te zien en dit is het tweede getal.

Nu ken je alle informatie, die je nodig hebt om een specifieke Arduino een eigen apparaatpad te geven. Creëer een bestand `/etc/udev/rules.d/90-arduino.rules`

en plaats daarin voor elke Arduino de volgende regel:

```
> SUBSYSTEMS=="USB",
ATTRS{idVendor}=="0403",
ATTRS{idProduct}=="6001",
SYMLINK+="arduino_
nano_${attr{serial}}"
```

Als je nu je Arduino Nano aansluit, krijgt die als apparaatpad `/dev/arduino_nano_serienummer`.

> Wat als je een apparaat met je Pi wilt in- en uitschakelen, maar je de GPIO's niet kunt of wilt gebruiken? Dan gebruik je gewoon USB <

FTDI EEPROM VAN DE ARDUINO FLASHEN

De USB FTDI-chip van een Arduino, die je gebruikt om firmware naar te flashen, heeft EEPROM dat je met de tool `ftdi_eeeprom` kunt beschrijven. Hiermee kun je zelf het vendor ID, product ID, de beschrijving van de fabrikant en het product- en het serienummer aanpassen. Dat kan handig zijn als je zowel originele