

waar 'USB' bij staat, niet degene waar 'PWR' bij staat. Voer dan in een terminalvenster de volgende opdracht uit:

```
> sudo rpiboot -d /usr/share/rpiboot/gpioexpand/
```

De directory `/usr/share/rpiboot/gpioexpand/` bevat de firmwarebestanden voor de Pi. Het programma `rpiboot` laat dan in enkele boodschappen zien dat de bootcode naar je Pi wordt gestuurd. Als dat is voltooid, sluit je het programma af en krijg je de opdrachtprompt terug te zien. Je Pi draait nu de aangepaste firmware en maakt zijn gpio-pinnen via het netwerk toegankelijk.

## NETWERK TESTEN

De Pi-firmware gebruikt alleen IPv6, dus je Pi is alleen via een IPv6-adres bereikbaar. Kijk met `ip address` wat de naam van de usb-interface van je Pi is. Waarschijnlijk is het de laatste interface in de lijst, omdat ze net pas is aangemaakt. Bij ons ziet dat er als volgt uit:

```
> 8: enp0s20f0u1i2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
> link/ether fa:00:fc:31:39:04 brd ff:ff:ff:ff:ff:ff
```



## SOLDEREN OF NIET

De Pi Zero en Pi Zero W komen standaard zonder gpio-header gesoldeerd. Je kunt dan op twee manieren elektronische componenten aansluiten. Of je soldeert de componenten rechtstreeks op de gaatjes. Dat is het minste werk, maar wel vrij drastisch: je kunt je Pi achteraf dan niet meer voor andere schakelingen gebruiken, tenzij je de componenten gaat desolderen. Of je soldeert een 40-pins header op de gaatjes. Dat is iets meer werk, maar met wat oefening goed te doen. Je kunt dan jumperwires op de gpio-pinnen aansluiten en dus gemakkelijk tijdelijke opstellingen met een breadboard maken om te experimenteren. Voor wie solderen een brug te ver is, zijn er gelukkig ook versies van de Pi Zero en Pi Zero W te koop met al een gpio-header op gesoldeerd. Die laatste versie staat ook bekend als de Pi Zero WH.

```
> inet6 fe80::751d:d87c:900d:d6c2/64 scope link tentative
> valid_lft forever preferred_lft forever
```

Nu kun je naar je Pi pingen om te zien of de netwerkverbinding het doet:

```
> ping6 fe80::1%enp0s20f0u1i2
```

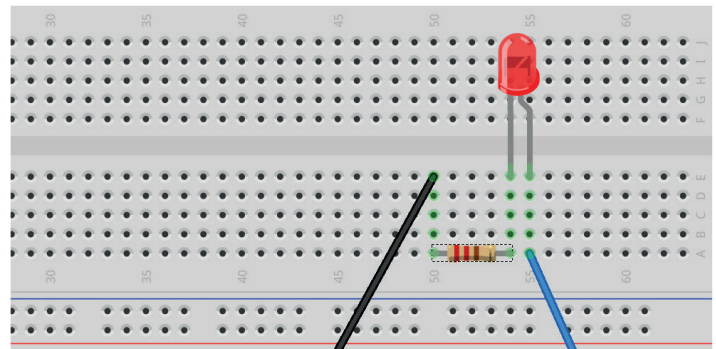
Je gebruikt daarvoor het IPv6-adres `fe80::1` en plakt het procentteken (%) erachter met daarna de naam van de interface (in ons geval `enp0s20f0u1i2`). Als alles goed gaat, krijg je antwoorden van de Pi op je ping-pakketjes. Druk op CTRL+C om de ping6-opdracht te stoppen.

## OMGEVINGSVARIABLEN INSTELLEN

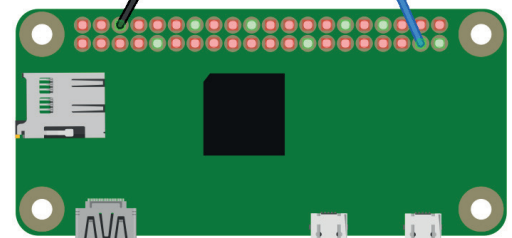
Gpiozero maakt gebruik van twee omgevingsvariabelen, die wel handig zijn om in te stellen. Plaats daarom de volgende twee regels in `~/.bashrc`:

```
> export GPIOZERO_PIN_FACTORY=pigpio
> export PIGPIO_ADDR=fe80::1%enp0s20f0u1i2
```

Op de tweede regel vervang je het adres uiteraard door dat van jou, inclusief het procentteken en de naam van de interface. Als je daarna een nieuw terminalvenster opent,



> De Pi Zero dient als interfacebordje om een led op je pc aan te sluiten.



fritzing

## > We gaan als test een led op de Pi aansluiten en die laten knipperen via een Python-script <

zijn de omgevingsvariabelen correct ingesteld en is alles klaar om de gpio-pinnen van je Pi aan te sturen.

### AAN DE SLAG!

Haal de usb-aansluiting van je Pi Zero nu uit je PC, zodat hier geen spanning meer op staat. We gaan nu als test een led op de Pi aansluiten en die laten knipperen via een Python-script. We creëren daarvoor een eenvoudige schakeling met één led op een breadboard. Neem daarvoor een led, een weerstand van 220 ohm en twee jumperwires. Plaats de led op het breadboard. Verbind de kathode (het korte pootje) via de weerstand van 220 ohm met een GND-pin op de Pi om de stroom door de led te begrenzen. De anode (het lange pootje) sluit je aan op pin BCM26 van de Pi. De locaties en namen van de gpio-pinnen van de Pi vind je op <https://pinout.xyz>. Bekijk het schema bij dit artikel voor de aansluiting en controleer goed of je alles correct hebt aangesloten voordat je de Pi terug op je PC aansluit.

Voer de `rpiboot`-opdracht opnieuw uit om de firmware op de Pi te starten. Het IP-adres blijft normaal gezien hetzelfde, dus dezelfde omgevingsvariabelen blijven werken. Schrijf dan in een bestand de volgende Python-code:

```
> from gpiozero import LED
> from time import sleep
> led = LED(26)
> while True:
>     led.on()
>     sleep(1)
>     led.off()
>     sleep(1)
```

Sla het bestand op als `gpiozero_blink.py` en voer het daarna uit:

```
> python3 gpiozero_blink.py
```

Als alles goed gaat, zie je nu de led op het breadboard knipperen. Je hebt je eerste elektronica-component via je PC aangestuurd! Bekijk zeker ook eens de uitgebreide documentatie van Gpiozero op <https://gpiozero.readthedocs.io>. Je vindt hier tientallen voorbeelden van Python-code om allerlei elektronische componenten aan te sturen via de gpio-pinnen. En dankzij `rpiboot` en een Pi Zero kun je dat nu ook op je PC.

## GPIO'S VAN EEN PI UITBREIDEN

Dezelfde aanpak is ook interessant als je meerdere HAT's of veel andere componenten op een Raspberry Pi wilt aansluiten en je te weinig vrije