

ATX-voeding voor Raspberry Pi

Sébastien Guerreiro de Brito (Frankrijk)

Heb je een Raspberry Pi die stroom nodig heeft, en een oude ATX-voeding? Een printplaatje, wat componenten en een ATtiny en het is gefixt.



Figuur 1: ATX-voeding. Bron: Shutterstock.

De ATX PC-voeding (**figuur 1**) werd in 1995 door Intel geïntroduceerd. Het is sinds de Pentium II-processoren de meest voorkomende schakelende voeding voor PC's. Alle elektronische, mechanische, milieu- en andere kenmerken zijn door Intel gespecificeerd om de ATX-norm te definiëren.

Het doel van deze schakeling is om een ATX-voeding te gebruiken om een Raspberry Pi-board en de vele randapparatuur die daarop kan worden aangesloten van stroom te voorzien. Het voordeel van deze voeding is het aantal verschillende spanningen dat beschikbaar is - op interessante vermogensniveaus.

ATX Voedingsaansluiting

In dit artikel gaan we echter niet de interne werking van de ATX schakelende voeding beschrijven, maar alleen proberen hem aan

de praat te krijgen. Laten we om te beginnen in detail kijken naar de connector (**figuur 2**) die moet worden aangesloten op het moederbord van de PC.

De connector bestaat uit

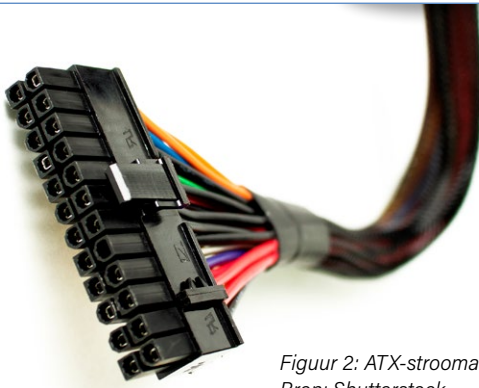
- een 24-polige, twee-rijige Molex Mini-Fit Jr vrouwelijke connector (ref: 39-01-2240) of gelijkwaardig
- Molex Mini-Fit HCS vrouwelijke contacten (ref: 44476-1112)

De pin-out van deze connector staat in **tabel 1**. De signalen die van belang zijn voor het aansturen van de voeding zijn PWR_OK en PS_ON#. De +5 VSB-voeding zal ook worden gebruikt in onze toepassing. Het PWR_OK signaal is een "Power good" signaal dat door de voeding wordt gebruikt

om het systeem te vertellen dat de +5 VDC, +3,3 VDC en +12 VDC uitgangen aanwezig zijn en voldoen aan de drempelwaarden.

Het PS_ON# signaal is het signaal waarmee we de voeding kunnen starten. Het is een TTL-compatibel signaal, actief-laag, waarmee het moederbord de voeding op afstand kan besturen. Dit maakt software-gestuurde voeding AAN/UIT, Wake-on-LAN, wake-on-USB activiteit, enz. mogelijk.

Wanneer PS_ON# laag wordt getrokken, moet de voeding de vier hoofdgelijkspanningen activeren: +12 VDC, +3,3 VDC, +5 VDC en -12 VDC. Wanneer PS_ON# naar een TTL-hoog niveau wordt getrokken, of naar een open circuit, mogen de DC-uitgangssporen geen stroom leveren en moeten ze op nul worden gehouden ten opzichte van massa. Het ATX-voedingscircuit voorziet in een intern



Figuur 2: ATX-stroomaansluiting.
Bron: Shutterstock.

anti-bounce circuit om aan/uit-schommelingen te voorkomen als het PS_ON#-signaal wordt geactiveerd door een mechanische schakelaar. De kenmerken van het PS_ON# signaal staan in **tabel 2**. De PS_ON# signaal-karakteristieken zijn in **figuur 3** als grafiek weergegeven.

Het PS_ON# signaal heeft geen effect op de +5 VSB uitgang. De +5 VSB (SB = stand-by) voeding is een voeding die aanwezig is wanneer de voeding op het lichtnet is aangesloten. Deze uitgang levert stroom voor circuits die operationeel moeten blijven wanneer de hoofdvoedingsspanningen zijn uitgeschakeld.

Beschrijving van de print

Om de ATX-voeding door de Raspberry Pi te laten aansturen, is een printplaatje ontworpen [1]. Dit bevat de benodigde logica om het opstarten en afsluiten van de Raspberry Pi mogelijk te maken door er een kleine controller aan toe te voegen. Dit geeft ook enkele andere leuke voordelen. Aangezien we een ATX-voeding hebben, zullen we de 5 V-rail gebruiken om een set USB-poorten aan de print toe te voegen, zodat randapparatuur die hogere stromen nodig heeft daar gebruik van kan maken. De PCB aan het werk is te zien in **figuur 4**.

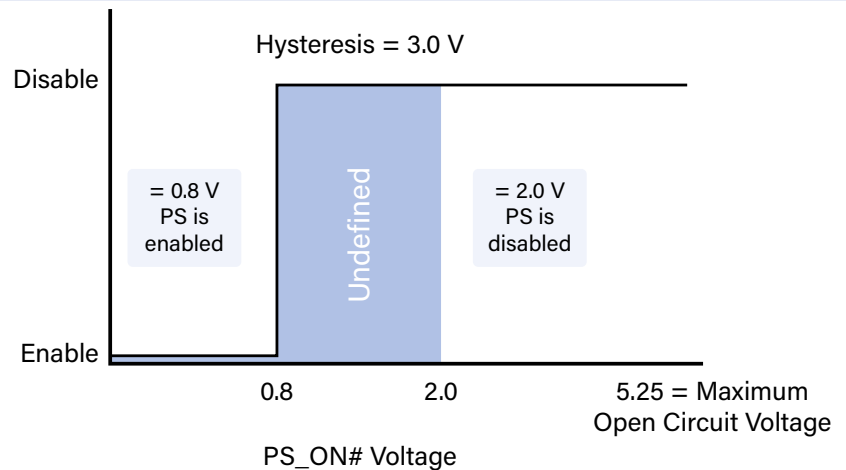
Parameter	Minimum	Maximum
V_{IL}	0 V	0,8 V
I_{IL}	-	-1,6 mA ¹
V_{IH}	2,0 V	-
$V_{IH, open circuit}$	-	-5,25 V
Rimpeling/ Ruis		400 mVpk-pk

¹ Opmerking: Negatieve stroom geeft aan dat er stroom loopt van de voeding naar het moederbord.

Tabel 2: PS_ON# signaalkarakteristieken.

Pin	Signal	Wire color
1	+3.3 VDC	
2	+3.3 VDC	
3	COM	
4	+5 VDC	
5	COM	
6	+5 VDC	
7	COM	
8	PWR_OK	
9	+5 VSB	
10	+12 V1DC	
11	+12 V1DC	
12	+3.3 VDC	
13	+3.3 VDC	
14	-12 VDC	
15	COM	
16	PS_ON#	
17	COM	
18	COM	
19	COM	
20	Reserved	NC
21	+5 VDC	
22	+5 VDC	
23	+5 VDC	
24	COM	

Tabel 1: ATX connector pinout.



Figuur 3: PS_ON# signaalkarakteristieken. (Bron: ATX/ATX12V Power Supply Design Guide versie 1.1, paragraaf 3.3.2, figuur 3)



Figuur 4: ATX-voeding aangesloten op de printplaat.

Figur 5: Schema.

Het schema van de printplaat is te zien in **figuur 5** en is relatief eenvoudig. Het is gebaseerd op het gebruik van een ATtiny85 microcontroller die rechtstreeks vanuit +5 VSB wordt gevoed. Het voordeel van het gebruik van een microcontroller is dat, hoewel de regeling van de voeding eenvoudig is, het de mogelijkheid biedt om andere functies toe te voegen.

Starten van de voeding

De ATtiny krijgt de opdracht de voeding te

starten door een externe drukknop die op jumper JP1 wordt aangesloten. Wanneer de drukknop wordt geactiveerd, stuurt de microcontroller transistor T1 aan, die de voeding start. Jumper JP8 maakt het mogelijk de voeding automatisch te starten.

Extra temperatuurbewaking voor de Raspberry Pi

Wij bieden extra luchtkoeling voor het Raspberry Pi bord. Hiervoor wordt een 10 kΩ NTC thermistor geplaatst en afhankelijk van

de gemeten temperatuur zal het board een ventilator activeren. Met behulp van jumper JP7 kunnen we kiezen voor een ventilatorspanning van 12 V of 5 V.

Communicatie met de Raspberry Pi

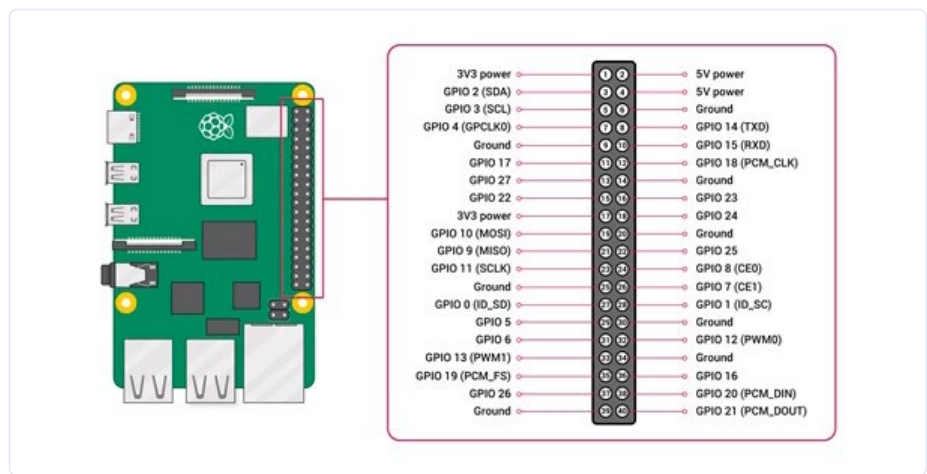
Om de ATX-voeding van de Raspberry Pi te kunnen uitschakelen, zoals op een PC, hebben we een kleine list bedacht. Het RPI_GPIO signaal is verbonden met GPIO27 op het Pi bord. De GPIO-connector van de Raspberry Pi is te zien in **figuur 6**. Wat het besturingssysteem betreft moeten we er eerst voor zorgen dat GPIO27 toegankelijk is. Daartoe maken we het GPIO-initialisatiebestand aan (dat bij het opstarten wordt uitgevoerd) met de nano-editor:

nano S75gpioint

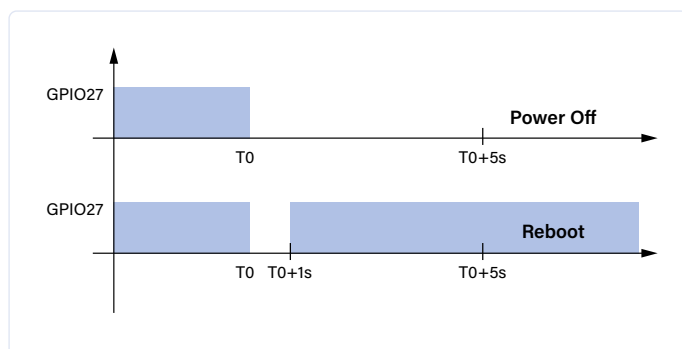
Dan voeren we de code van **Listing 1** in. We maken het bestand uitvoerbaar en plaatsen het in de initialisatiemap.

```
sudo chmod +x S75gpioinit
sudo mv S75gpioinit /etc/init.d
```

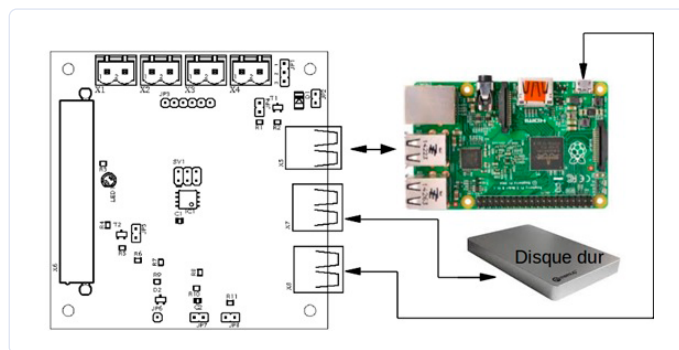
Je zult in het initialisatiescript zien dat GPIO27 is geïnitieerd met een waarde van 1. Dus, wanneer we de Raspberry Pi uitschakelen



Figuur 6: Raspberry Pi pinout (Bron: [2]).



Figuur 7: Beheer van de GPIO27-pin.



Figuur 8: Aansluiten van de Raspberry Pi.

via het OS (poweroff, bijvoorbeeld), zal dit signaal veranderen in 0 zodat de microcontroller weet dat het tijd is om de ATX-voeding uit te schakelen.

Maar, gewoon omdat het kan, willen we een reboot van de Raspberry Pi mogelijk maken met behulp van de reset van de voeding.

Hiervoor gebruiken we de truc om GPIO27 te variëren om de microcontroller het signaal te geven dat niet een poweroff, maar een reset wordt gevraagd.

Zoals je kunt zien in **figuur 7**, is het beheer van GPIO27 als volgt gedetailleerd: Als de microcontroller de GPIO27-pin op 0 ziet

staan, wacht hij 5 seconden en als de pin dan nog steeds op 0 staat, betekent dit dat de Raspberry Pi is uitgeschakeld en dus schakelt hij de ATX-voeding uit. Zo niet, dan gaat het om een reboot commando en zorgt hij ervoor dat de ATX-voeding 2 seconden uitschakelt, voordat hij opnieuw opstart.



Listing 1: Shell-Script

```
#!/bin/sh
#####
# IO Init Script #
# Author : Sebastien Guerreiro #
# Versions : #
# ----- #
# Mars 2019 V1.0 Création #
#####
CHEM_GPIO=/sys/class/gpio
#-----
register()
{
#Registers the outputs
if [ $SENS="out" ]; then
#Outputs
echo "$NUM" > $CHEM_GPIO/export
echo "$SENS" > $CHEM_GPIO/gpio$NUM/direction
echo "$VALEUR" > $CHEM_GPIO/gpio$NUM/value
chmod g+w $CHEM_GPIO/gpio$NUM/value
else
#Inputs
echo "$NUM" > $CHEM_GPIO/export
echo "$SENS" > $CHEM_GPIO/gpio$NUM/direction
fi
ret=$?;
if [ $ret -eq 0 ]; then
#echo_success;
echo "Registering GPIO$NUM : OK"

else
# echo_failure;
echo "Registering GPIO$NUM : ERROR"
exit $ret;
fi
}
#-----
unregister()
{
echo "$NUM" > /sys/class/gpio/unexport
ret=$?;
if [ $ret -eq 0 ]; then
echo_success;
else
echo_failure;
exit $ret;
fi
}
#-----
start()
{
echo "Registering GPIO--Setting Power GPIO ON"
#Output for power GPIO
NUM=27; SENS=out ; VALEUR=1
register;
}
#-----
start
exit 0
```



Listing 2: Het shutdown script.

```
#!/bin/sh
#####
# Reboot Poweroff Management #
# Author : Sebastien Guerreiro (www.sebelectronic.com)#
# Versions : #
# ----- #
# Fev 2020 V1.0 Création #
#####
case $1 in
-r)
echo 0 > /sys/class/gpio/gpio27/value
sleep 1
echo 1 > /sys/class/gpio/gpio27/value
sleep 1
/sbin/shutdownSeb $@
;;
*)
/sbin/shutdownSeb $@
;;
esac
```

Om dit alles mogelijk te maken moeten we ingrijpen met enkele Linux commando's. Aan de kant van het besturingssysteem hernoemen we de `shutdown` en `reboot` instructies in `sbin` naar respectievelijk `shutdownSeb` en `rebootSeb`.

```
sudo mv /sbin/shutdown /sbin/shutdownSeb
sudo mv /sbin/reboot /sbin/rebootSeb
```

Vervolgens maken we twee scripts: `shutdown` en `reboot`, die we in de map `/sbin` plaatsen. Typ in de terminal:

```
nano shutdown
```

en voer het volgende (**Listing 2**) in het bestand in. Sla het bestand op en sluit nano af.

Typ in de terminal:

```
sudo cp shutdown /sbin
```

Dit kopieert uw script naar `/sbin` zodat u het later kunt uitvoeren. Vervolgens moeten we het `reboot` script uitvoeren. Ook in de terminal, voer in:

```
nano reboot
```

en voer het volgende script (**Listing 3**) in dat bestand in. Sla het bestand op en sluit nano af. Typ dan in de terminal:

```
sudo cp reboot /sbin
```

Dit zal uw herstartscript kopiëren naar `/sbin` en het later laten uitvoeren.

Dat is het!



Lijst 3: Het herstartscript.

```
#!/bin/sh
#####
# Reboot Management #
# Author : Sebastien Guerreiro #
# Versions #
# ----- #
# Fev 2020 V1.0 Création #
#####
echo 0 > /sys/class/gpio/gpio27/value
sleep 1
echo 1 > /sys/class/gpio/gpio27/value
sleep 1
#/sbin/rebootSeb
/sbin/shutdownSeb -r now
```



VERWANTE PRODUCTEN

- > **Raspberry Pi 4 2 GB (SKU 18965)**
www.elektor.com/18965
- > **Velleman VTSS220 Temperatuurgestuurd Soldeerstation (SKU 19865)**
www.elektor.com/19865

WEB LINKS

- [1] Project on Elektor Labs: www.elektormagazine.com/labs/atx-powersupply-for-rpi
- [2] Image Source: www.raspberrypi.com/documentation/computers/images/GPIO-Pinout-Diagram-2.png

De randapparatuur aansluiten

Het gebruik van een ATX-voeding maakt het mogelijk het bord aan te sluiten op bijvoorbeeld een IDE harde schijf, of in mijn geval een IDE CD-ROM drive. Kortom, het voordeel is dat je de randapparatuur van je oude PC

kunt gebruiken voor je nieuwe projecten. De aansluiting gebeurt zoals je ziet in **figuur 8**.

Wie meer wil weten over het project kan terecht op de Elektor Labs pagina [1].

191205-03

Vragen of opmerkingen?

Heeft u vragen of opmerkingen over dit artikel? Neem contact op met Elektor via editor@elektor.com of laat een reactie achter op de projectenpagina van Elektor Labs.



ONDERDELENLIJST

Weerstanden (0805, 0,1 W)

R1, R3, R6, R7, R8 = 1 kΩ
R2, R4, R5, R9, R10 = 100 kΩ
R11 = 10 kΩ

Condensatoren (0805)

C1 = 100 nF
C2 = 10 nF

Halfgeleiders

D1 = SM4007
D2 = BAT54
IC1 = ATTINY85

T1 = SI2318CDS

T2 = 2N7002

LED1 = LED, groen, 3 mm

Diversen

SV1, JP7 = Header 3 pennen, 2.54 mm pas

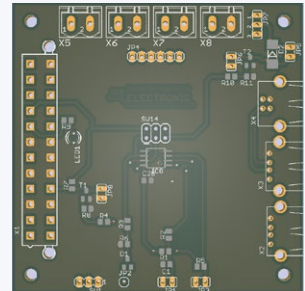
JP1, JP3, JP5, J6, JP8 = Header 2 pennen,
2.54 mm pas

JP2 = Header 1 pen

SV14 = Header 2 rijen van 3 pennen, 2.54 mm
pas

X1 = Molex Mini-Fit Jr verticale connector, 5566,
24-way, 2 rijen

X2, X3 = USB Type A connector



X4 = USB Type B connector

X5, X6, X7, X8 = MSTBVA 2.5 / 2-G-5.08 2-pins
header

Print

Advertisement

Word lid van de Elektor Community

- ✓ Een compleet web-archief t/m 1980!
- ✓ 8x Elektor Magazine (Print)
- ✓ 8x digitaal (PDF)
- ✓ 10% korting in onze webshop, en exclusieve aanbiedingen
- ✓ Toegang tot meer dan 5000 Gerberfiles



Neem nu een
lidmaatschap!



www.elektormagazine.nl/Abonnement

elektor
design > share > earn