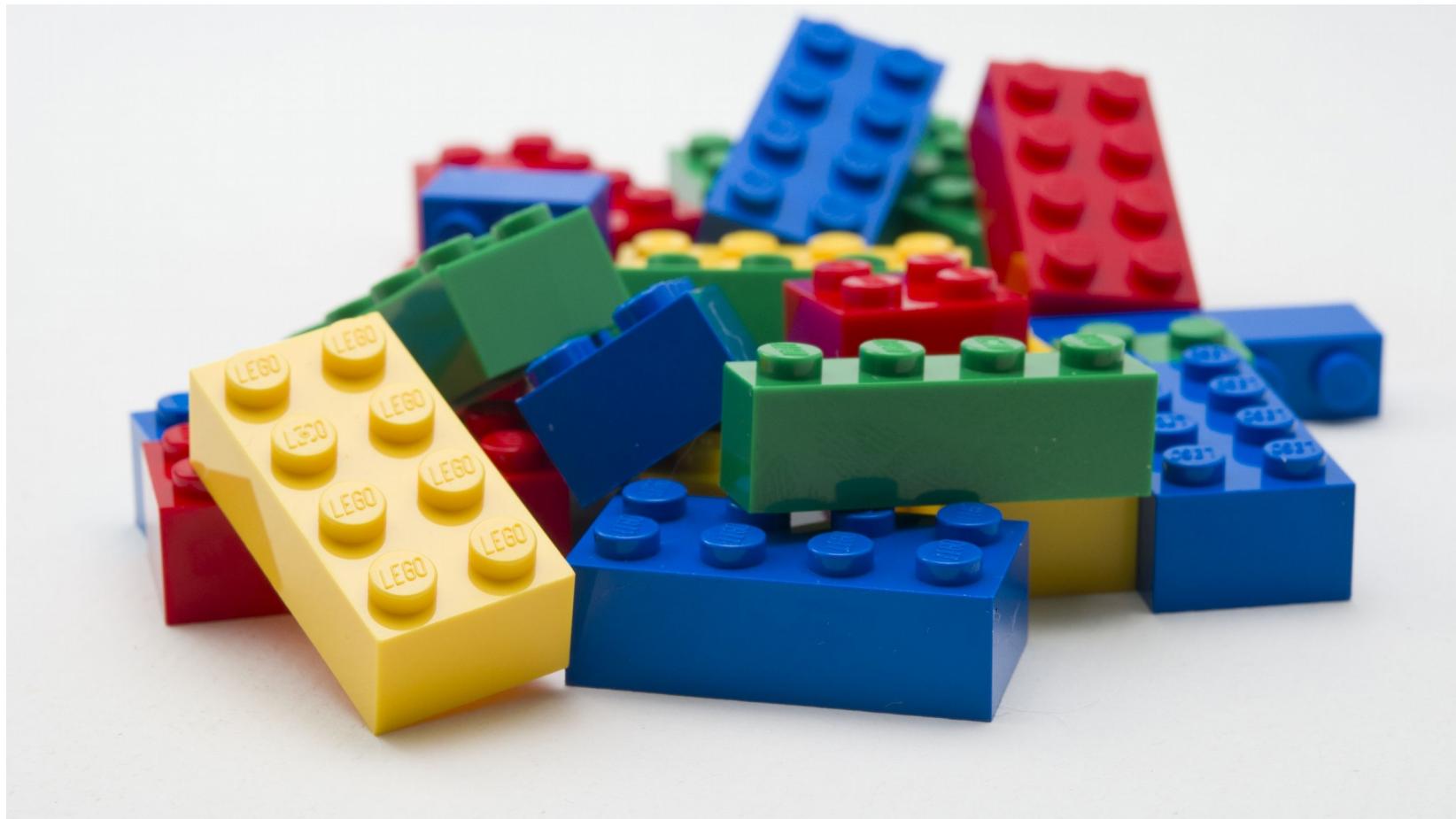


LinuxCNC workshop

Wat is LinuxCNC niet?



Wat wel?



Hoe maak je dan een systeem?

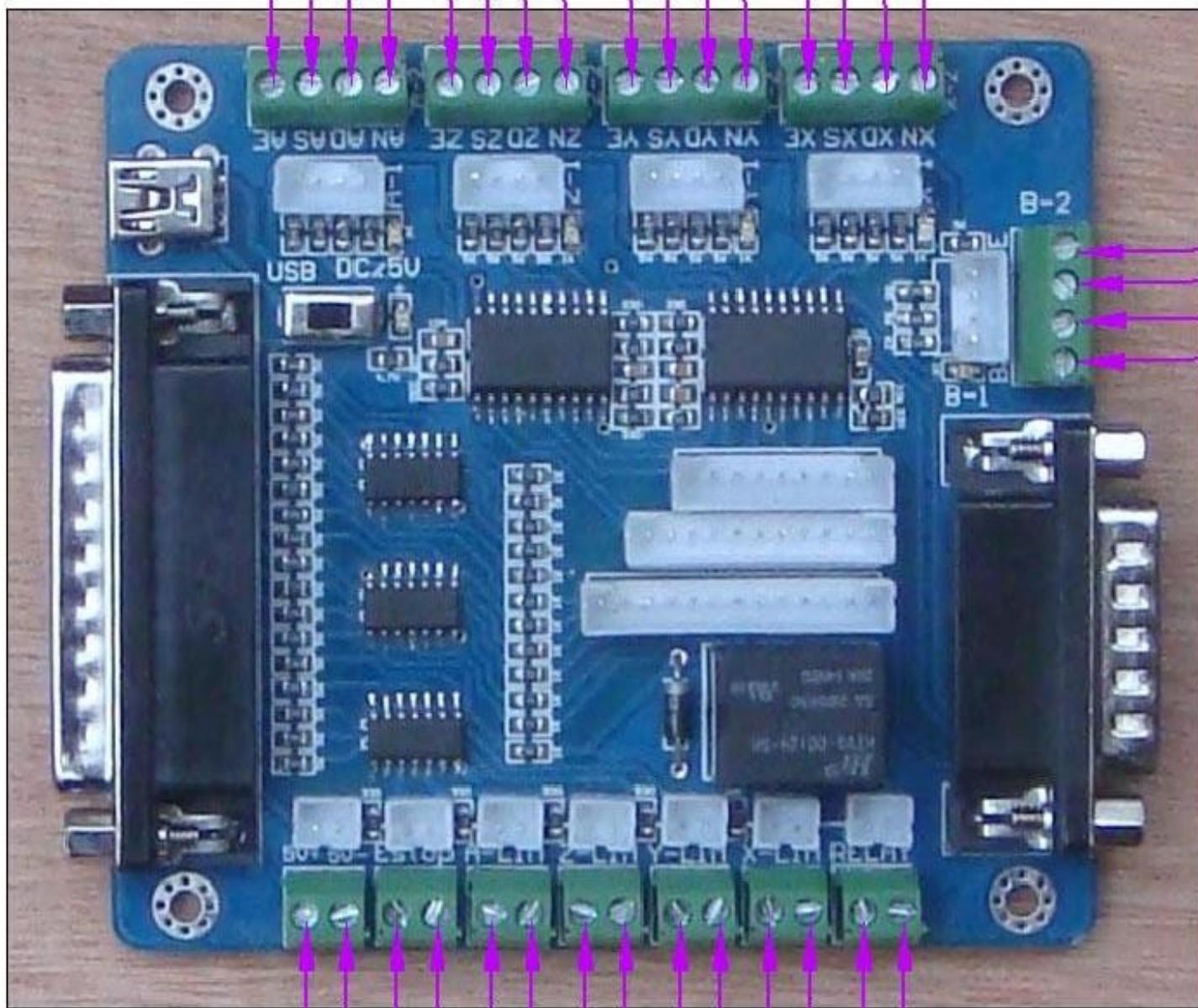


Ieks, hellup!



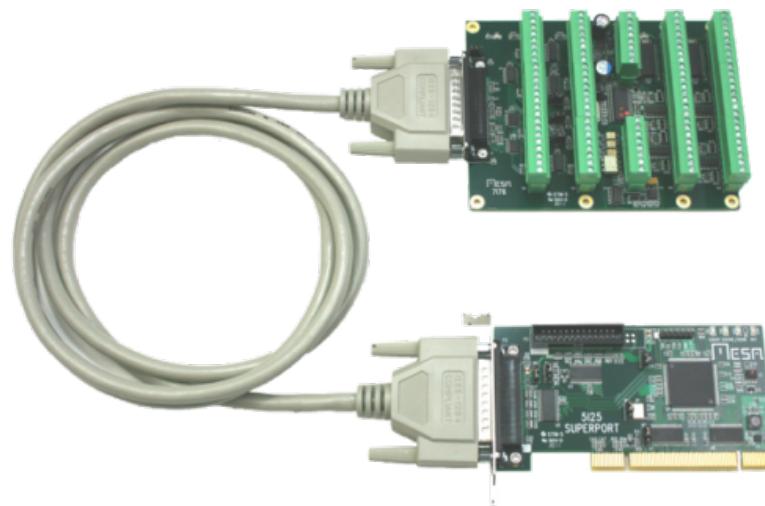
Gelukkig hebben we de Stepconf en PnCConf wizards.

Probeer maar: Applications Menu → CNC → Stepconf Wizard



P1 P6
P5 GND
P1 P7
P8 GND
P1 P16
P3 GND
P1 P2
P14 GND
P1 P4
P17 GND
GND
5V- 5V-
GND
P15 GND
P13 GND
P12 GND
P11 GND
P10 GND
GND
GND

Mesa 5i25 + 7i76



Ook een ‘robotje’ beschikbaar om
de legoblokjes te stapelen:
‘PnCConf’

Ja, leuk, maar....

- Ik heb een toolchanger..
- Ik heb een vacuumtafel en wil zones sturen
- Ik heb een speciale 8-assige machine
-

Dan moet je toch zelf blokjes gaan stapelen...

Hoe werkt LinuxCNC?

- Elke milliseconde word een nieuwe positie uitgerekend en naar de motoren gestuurd.
- Strak elke milliseconde....
- EdingCNC, Mach3, etc. doen dat op externe hardware.
- LinuxCNC doet dat met een ‘realtime OS’.
- ‘Smart PC, dumb hardware’.

Hoe werkt LinuxCNC?

- De 1-milliseconde ‘stopwatch’ noemen ze ‘servo-thread’
- Standaard 1kHz, maar aan te passen naar behoefte.
- Meerdere ‘stopwatches’ mogelijk, bijvoorbeeld base-thread.

Hoe werkt LinuxCNC? HAL-file

- LinuxCNC is een bouwwerk van bouwstenen
- De HAL-file beschrijft welke bouwstenen er gebruikt worden
- En hoe ze aan elkaar geknoopt zijn

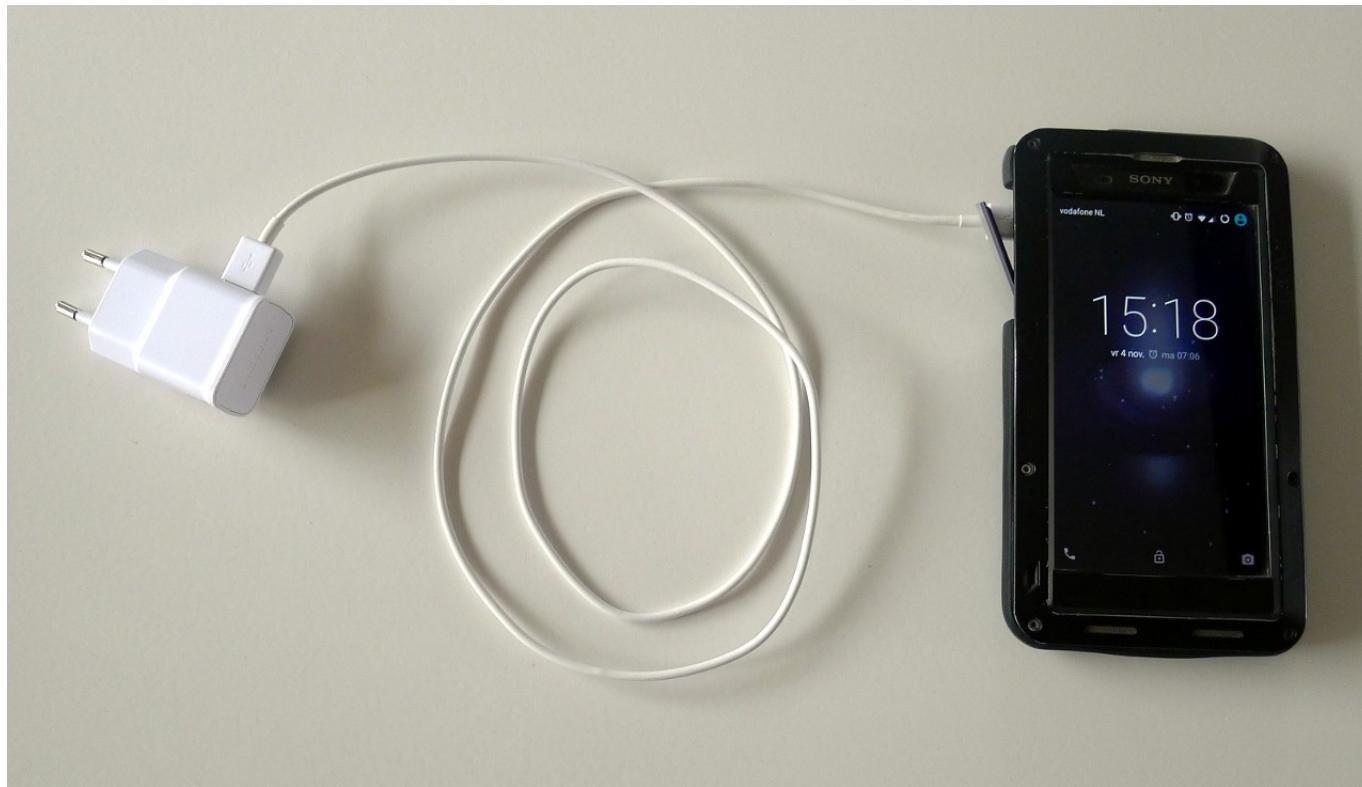
Hoe werkt LinuxCNC?

INI-file

- De INI-file beschrijft instellingen van een blokje
- Bijvoorbeeld de maximale snelheid van een as.
- Simpele structuur: **INSTELLING = waarde**

Hoe werkt dat met die blokjes in HAL?

- LinuxCNC term voor een blokje: **component**
- LinuxCNC term voor een aansluiting: **pin**
- Blokje uit de kast pakken: **loadrt**
- Blokje toevoegen aan het systeem: **addf**
- Blokjes aan elkaar knopen met draadje: **net**



loadrt telefoonlader
loadrt telefoon

addf telefoonlader servo-thread
addf telefoon servo-thread

net usbkabel telefoonlader.usb => telefoon.usb



loadrt telefoonlader

loadrt telefoon

loadrt koptelefoon names=linkerdop,rechterdop

addf telefoonlader servo-thread

addf telefoon servo-thread

addf linkerdop servo-thread

addf rechterdop servo-thread

net usbkabel telefoonlader.usb => telefoon.usb

net audiosnoertje telefoon.geluiduit => linker.dop.in =>
rechterdop.in

Welke bouwstenen hebben we?

- Een lijst is te vinden in
LinuxCNC_Documentation.pdf, pagina 636

De motioncontroller?

- Deze bestaat uit een aantal bouwstenen:
 - motion-command-handler
 - motion-controller
 - kinematica

De gebruikersinterface?

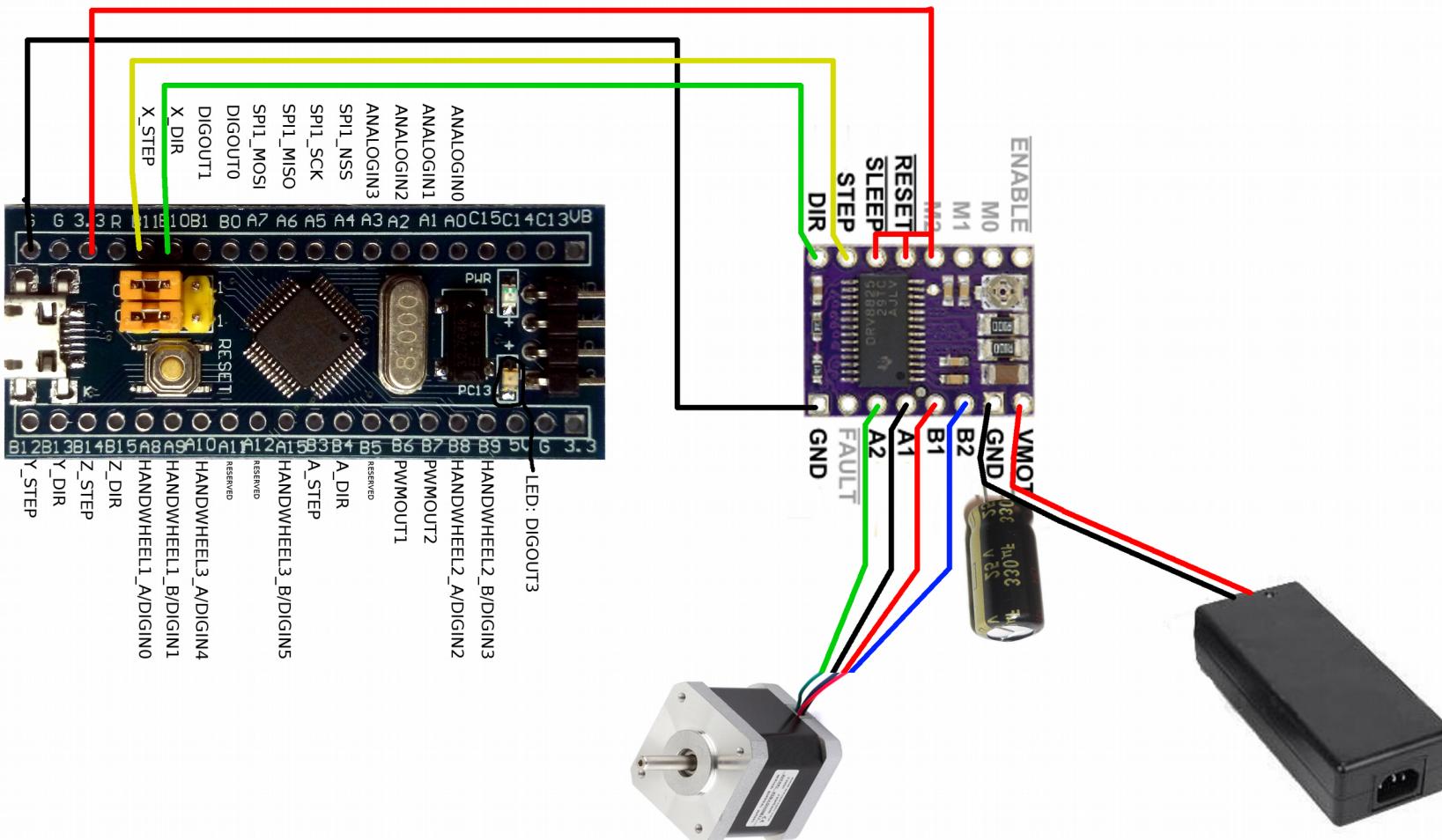
- Ook al een bouwsteen. Het word saai.
 - Meerdere beschikbaar.
 - Je kunt dus zelf kiezen welke je wil.
 - Of zelfs je eigen gebruikersinterface maken...
 - Veel gebruikte: AXIS, gmoccapy, touchy
-
- Sluit de USB-controller aan, koppel ‘m in de VM, en speel met AXIS en GMOCCAPY.

Hoe zitten die HAL en INI in elkaar?

- Open de HAL file: ‘configs’ shortcut → axis_kaal → usbmotion.hal
- Wat doet setp?
- Waarom hebben we ook een pos-fb?
- Bekijk de INI file ook eens, maar besteed niet teveel aandacht aan het snappen ervan.

Draadjes...

CNCZone.nl LinuxCNC workshop bedradingsschema



Bekijken van bouwstenen

- Meestal doet het niet meteen wat je wil.
- Meten is weten:
 - halshow: lijst van alle bouwstenen en pinnen
 - halscope: traditionele oscilloscoop

In AXIS:

Machine → Show HAL Configuration

Machine → HAL Scope

Bekijken van bouwstenen

- Open kaal_axis
- Aanzetten (F1, F2)
- Machine → Show HAL Configuration
- ‘WATCH’ aanklikken
- Pins → dabitusb → digitalin1
- Maak nu met een draadje verbinding tussen een ‘3.3’ rondje en rondje ‘A9’.

LinuxCNC pimpen

- We doen niet allemaal enkel ‘druk op de groene knop, koffie!’
- Er zijn veel door anderen gemaakte uitbreidingen beschikbaar. Teach-in, camera, toolsensor, probing, etc.
- En zelf kun je ook eenvoudig uitbreidingen maken, zelfs zonder programmeerkennis.

Zo, nu gaan we leuk spelen :)