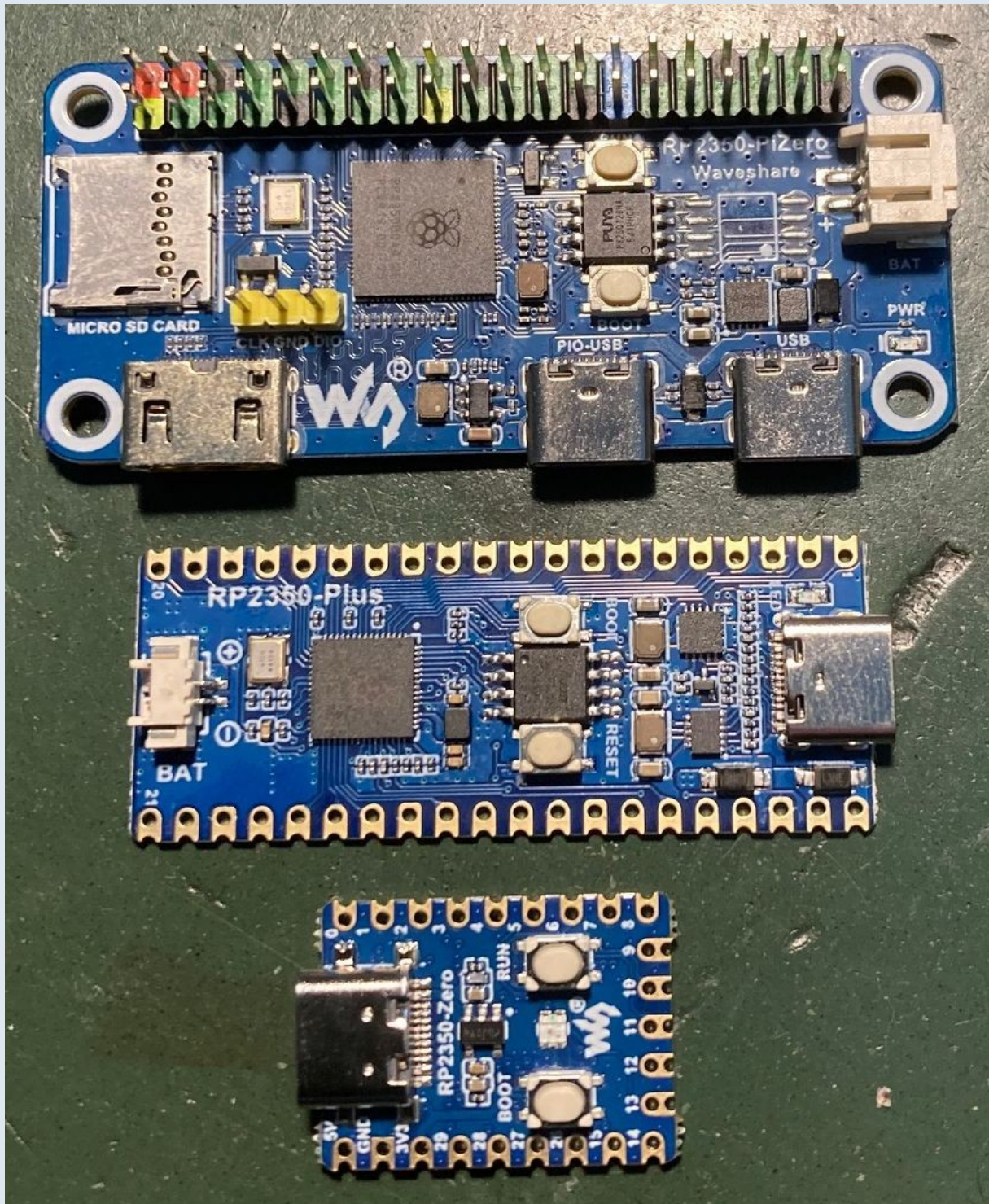


noForth t voor RP2350

0) Waarom voor de RP2350

Omdat de RP2350 is een heel snel en zeer veelzijdige chipje is. De wensenlijst bevat nog een snelle ARM versie, STC met inlining en een versie voor de ook aanwezige Hazard3 RiscV controller..



1) Wat is anders in noForth t op de RP2350

- Er is een image-definition nodig om de UF2 geaccepteerd te laten worden door de RP2350 bootloader.
- De secondary boot is volledig herschreven
- **COLD** maakt daarvan gebruik om in alle gevallen netjes te starten of herstarten
- Er zit dubbel zoveel RAM geheugen in
- Het noForth klok systeem is flexibel geïntegreerd met een aanpasbaar datastructuurtje **MAX-CLK VCORE PLL>** en **KHZ**
- De noForth t assembler en disassembler zijn uitgebreid met M33 opcodes
- Optimaliseringen in de assembly code aangebracht
- Woordenset voor de nieuwe snelle I/O co-processor toegevoegd aan bibliotheek
- Een veelzijdiger exception handler is mogelijk (Bus fout/Mem. fout)/Etc.)
- MS en US gebruiken nu een andere timer vanwege het nieuwe interne klok systeem met aparte TICK timers. Via het woord **TICKS** zijn deze tick generators ook beschikbaar voor de programmeur
- **CONFIG** is volledig herschreven en probeert op frequentie uit cel nul van **CFG** op te starten. Als dat niet lukt neemt hij de maximaal mogelijke frequentie. De frequentie is in cel nul van **CFG** genoteerd in stappen van 10kHz
- **FREEZE** is aangepast voor het grotere RAM geheugen
- De bibliotheek is volledig gecontroleerd en aangepast waar het nodig was. Sommige routines zijn overbodig geworden door het nieuwe klok systeem

2) Voorbeelden

Met het woord **KHZ** kun je de frequentie en daarmee het stroomverbruik op elk moment aanpassen. De overige kloksystemen voor USB, UART, MS, etc. worden hierdoor niet beïnvloed. Let op, kloksnelheden onder de 0,5 MHz leveren geen winst in het stroomverbruik op.

```
\ Change CPU klok, minimal valid frequency here is 10kHz,
\ The maximum is the configured PLL clock freq.
extra: : KHZ          ( kHz -- div ) \ Set scaled clock freq. in 1kHz
steps
  MAX-CLK @ 3E8 / >r          \ Freq. in kHz                f1 f1
  r@ umin 0A umax r>         \ Keep frequency in valid range  fn f1
  over 0A / 0 cfg 2 + h!     \ Store it in CFG in 10kHz steps  fn f1
  over /mod >r              \ Calc. q & remainder            fn r  q
  10000 rot */              \ Build fractional part           fraq q
  r> h+h 40010040 ! ;       \ Set new freq.
```

Als je het stroomverbruik wilt minimaliseren, kun je zoveel mogelijk devices uitschakelen en de core voeding-spanning zo laag mogelijk zetten. Dat pas je aan met **VCORE**. Let op, als je de spanning te laag zet, herstart de RP2350 en daarmee noForth t.

De laagst bruikbare PLL-frequentie met de USB actief is 48MHz, de hoogst bruikbare tot nu toe 480MHz. Maar ook daar kun je zelf mee experimenteren.

```
v: inside
MAX-CLK decimal
48,000,000 over ! cell+      \ PLL clock
hx 90 over h! 2 +           \ Core voltage 1.0V
hx 3C53 over h! 2 +         \ PLL dividers xt * 60 / 15
1 1 b+b swap !             \ Dividers for peripheral & USB/ADC
clock

4 cfg @ [if]                \ Is it a duo version?
  ramborder hx 3C44 + >r     \ MAX-CLK of core1
  MAX-CLK @+ r@ !           \ Copy MAX-CLK clock table
  @+ r@ cell+ !
  @ r> cell+ cell+ !
[then]

max-clk @ 10000 / 0 cfg 2 + h! \ Set maximum clock freq.
4 cfg @ abs 4 cfg !          \ Enable restart
v: fresh
config led                   \ Test new PLL clock, LED off
```

3) Github pagina's

noForth-T-for-the-RP2350

noForth t for the RP2350 sources and examples. There will be more UF2 files for more solo & duo versions of noForth t also more examples. The UF2 solo version with library is already quite complete!

These sources are here to document noForth t, and the hardware examples are there to learn how to use the the RP2350. All code should run on noForth t for the RP2350 solo & duo from version 260826 onwards. More info on noForth, click the link below.



-
- [Metacompilers for noForth](#) ; Metacompiler for noForth t solo multi, written for win32forth.
 - [noForth t UF2 files](#) ; noForth t solo & duo UF2 files, the most complete versions have USB, tools & source code library added
-
- [Configuration](#) ; Setting the system clock, baudrate & pin for S? and

noForth t staat nu compleet op een tweetal Github pagina's en wel:

- <https://github.com/WillemOuwerkerk/noForth-T-for-the-RP2040>
- <https://github.com/WillemOuwerkerk/noForth-T-for-the-RP2350>

4) Finale

- noForth moet nu goed aan de tand gevoeld worden
- Gaarne de bibliotheek inhoud goed testen
- Een eerste bug in /MOD is al gevonden
- Floating point bibliotheek schrijven en toevoegen
- Optie-1: noForth t voor de RiscV erbij maken
- Optie-2: noForth t met subroutine bedrading en inlining uitwerken voor de snelheidsmaniakken onder ons

5) Enkele demo's

**s" spi" need oled-app\ \ Laad en OLED DEMO
words**

**task: DEMO \ Voeg taak voor OLED toe
' oled demo start-task**

**need tasks \ Multitasker tools erbij
tasks**

need .cfg \ Toon noForth configuratie

**run 480MHZ \ Wijzig PLL klok naar 480MHz
.cfg**

**dm 480,000 khz \ Zet systeem klok op 480MHz
.cfg**

**dm 48,000 khz \ Systeem klok naar 48MHz
.cfg**

6) Nog vragen?