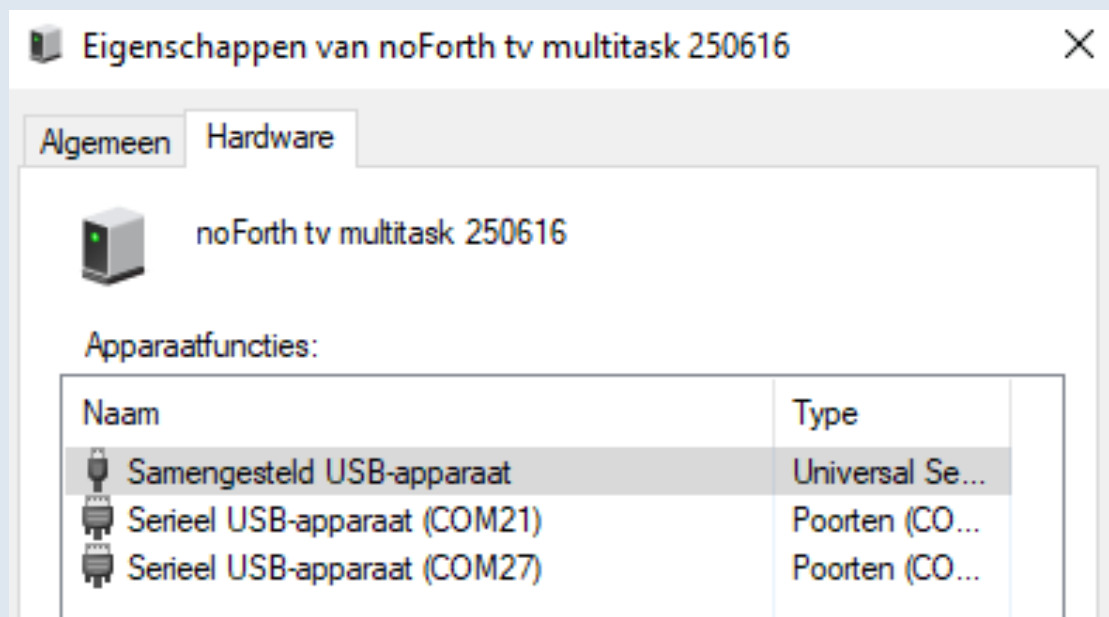


Werkgroepen verslag

USB werkgroep

De USB werkgroep is bijna afgerond, de huidige USB-driver draait onder Linux en Windows en Apple. We zijn nu bezig met het documenteren. We doen nu ook een poging om twee CDC poorten over één USB-kabeltje te laten lopen. Dat zou noForth t duo nog bruikbaar maken.



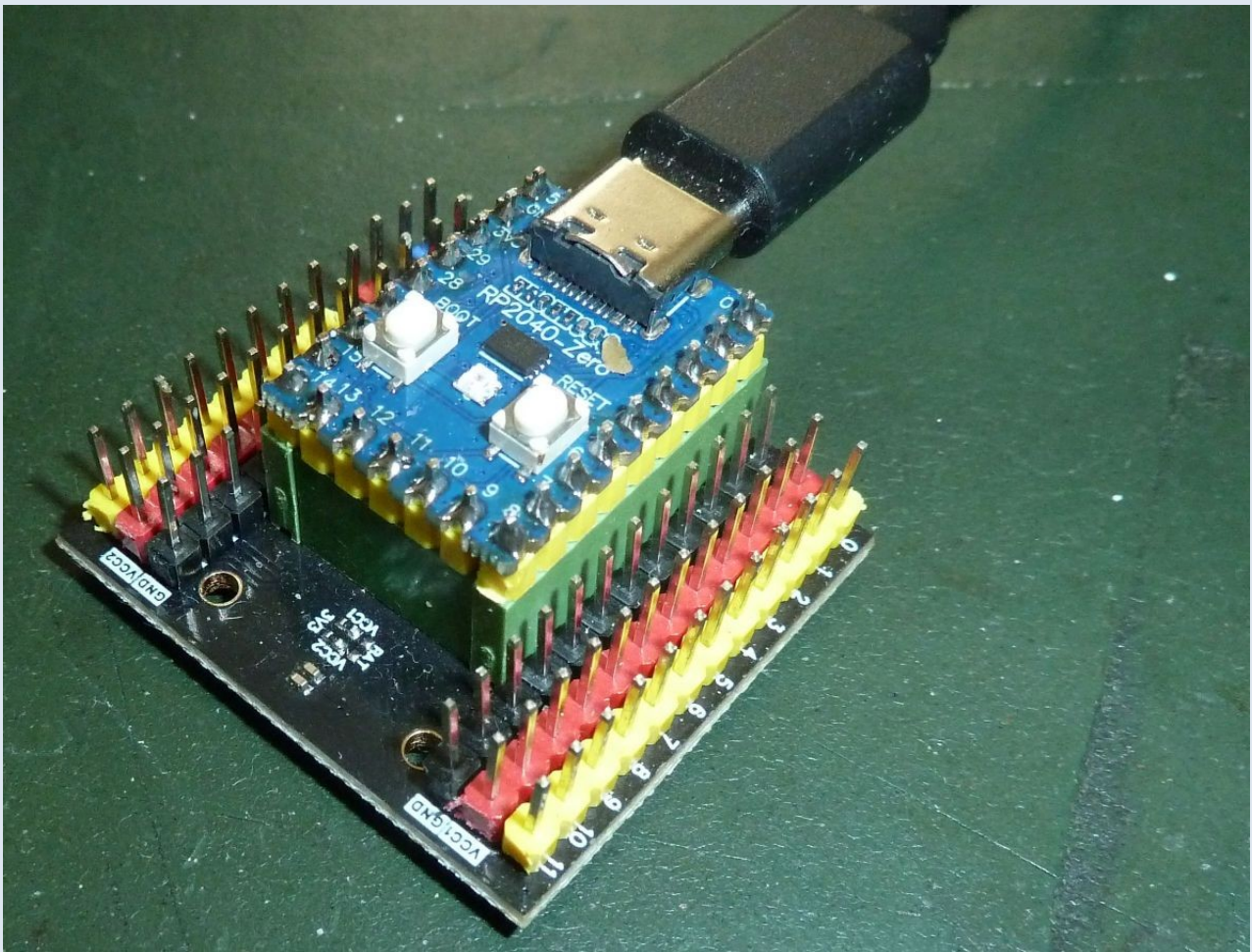
We lopen tegen een probleem aan, de pointers van de uitvoer buffer (EMIT) raken ongelijk, waarschijnlijk doordat twee processoren tegelijk in de ringbuffers te werken.

```
501006A0 000 0BA 0BA
501007AC 000 050 050
501006A0 000 0BA 0BA
501007AC 000 050 050
501006A0 000 0BA 0BA
501007AC 000 050 050
501006A0 000 0BA 0BA
501007AC 000 050 050 ← Ok
501006A0 000 0BB 0BB
501007AC 000 0C2 0CA ← Fout
501006A0 000 0BB 0BB
501007AC 000 0C2 0CA
```

Een oplossing zou kunnen zijn een nummer die aangeeft wie in de buffer bezig is (ook wel semafoor genoemd).

noForth's

We zijn nog steeds niet klaar met de nieuwe release van noForth. Er zijn veel kleine wijzigingen en enkele bugs opgelost. Dat betekent vooral veel documentatie schrijven. Er is een multitasking versie bij is gekomen, een USB-driver, de bibliotheek, een harde fout afhandelaar en tenslotte is er een turnkey generator voor noForth t toegevoegd.



Mini noForth t ontwikkel systeem

NEED

De bibliotheek bevat nu ook de USB-driver, meer inzichtelijke harde fout afhandeling, tijd meting en een extra PIO-driver voor de DHT22 vochtigheid- en temperatuur-sensor. Zie hier:

DHT22 PIO driver

```
clean-pio decimal      \ Empty code space, start at zero
0 0 {pio               \ Use state machine-0 on PIO-0
  160000 =freq          \ On 4 * 40 kHz frequency (6.25 µs ticks)
  05 1 =out-pins        \ GPIO 5 for OUT & SET
  05 1 =set-pins
  05 =in-pin           \ GPIO 5 for IN & JMP
  05 =jmp-pin

  0 =in-dir             \ Shift left
  32 1 =autopush

  0 pindirs set,        \ Pin is input
  wrap-target
    4 [] begin, again,  \ Wait for start command
  one ( Sensor readout )
    1 pindirs set,      \ Pin is output
    3 y set, begin,     \ Generate start pulse low .8 ms
      31 [] 0 pins set,
    y--? until,
    6 [] 1 pins set,    \ 44 µs start pulse high

    6 [] 0 pindirs set, \ Now input, wait 44 µs, skip response low
    6 [] high 0 pin wait, \ Wait for response high 44 µs

    31 y set,           \ Now read the 32-bits answer 6.25 µs
    begin,              \ This contains the sensor data
      low 0 pin wait,
      5 [] high 0 pin wait, \ Wait for high bit 37.5 µs
      1 pins in,         \ Shift in low or high bit 6.25 µs
    y--? until,         \ Count all 32 bits 6.25 µs)
    7 y set,            \ Read 8-bit checksum 6.25 µs
    begin,
      low 0 pin wait,
      5 [] high 0 pin wait, \ Wait for high bit 37.5 µs
      1 pins in,         \ Shift in checksum bit 6.25 µs
    y--? until,         \ Count 8 bits 6.25 µs
    push,               \ Result 8-bits to fifo 6.25 µs
  wrap
  0 =exec               \ Start with wait loop
pio}

: READ) ( -- data chk )
  one> 0 exec-opc      \ Start new measurement
  begin 0 rx-depth 1- until \ Fifo minimal two deep?
  0 rxf> 0 rxf> ;      \ 32-bits data & 8-bits checksum
```

Het Forth klasje

*/	123.45	.F	2DROP	procenten	D+	Celsius
D.	dubbels	DNEGATE	Fahrenheit	π	inhoud	12345.
tweecellig	2>R	schaling	coördinaten	breuken	S>D	
2DUP	vaste komma	vorm	eencellig	gemengd		

Onderwerpen:

- Turnkey in (no)Forth
- Parsing [IF] etc.
- */ gebruik voor schaling en vaste komma
- EVALUATE
- WORDLIST (VOCABULARY)
- EXECUTE
- INVERT vs NOT (13 april 2024)
- MIN vs UMIN (13 april 2024)
- Dubbele getallen
- CATCH gebruiken
- KEY
- OK lus
- Getal uitvoer bouwen met <# # #S #>
- +LOOP voor string of COUNT
- CREATE DOES>
- Control flow stack
- S"
- BEGIN WHILE WHILE REPEAT THEN
- []
- COMPILE,

Vandaag behandelen we de
programmeer opdracht: Woorden raden