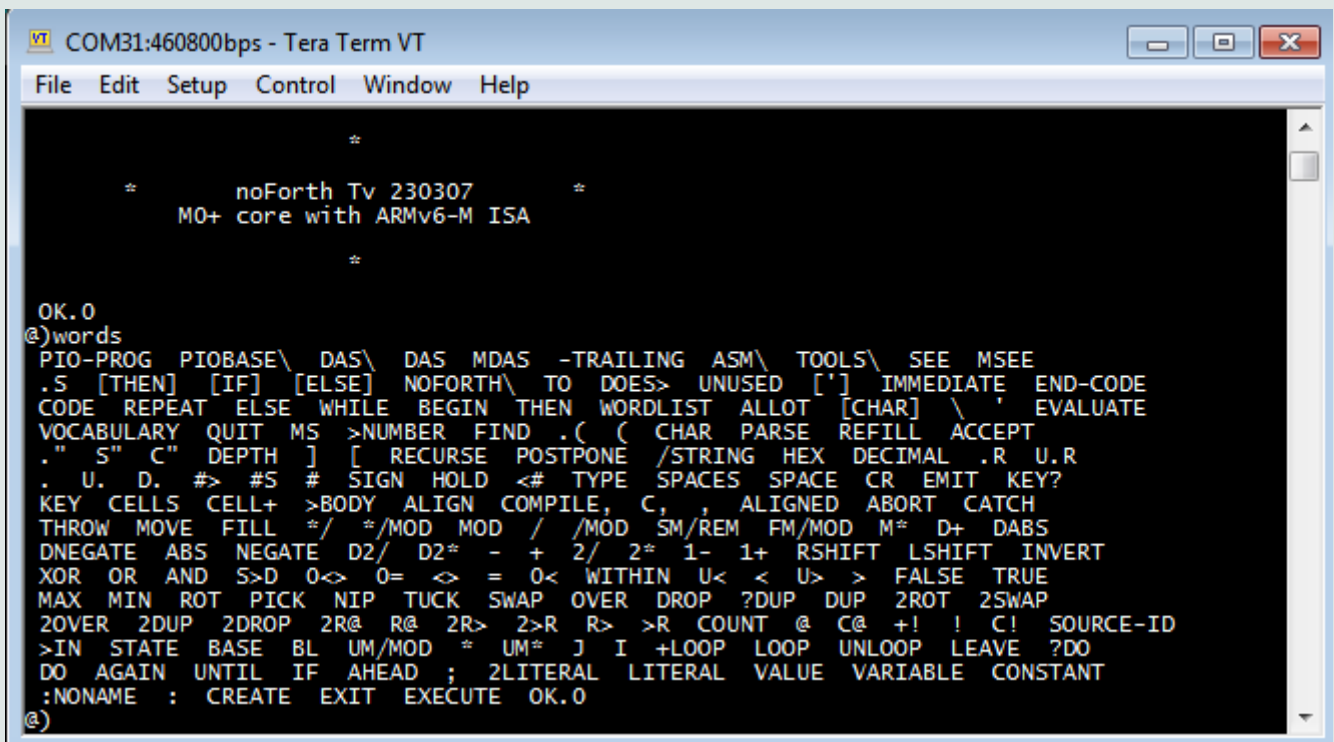


noForth T introduction



```
COM31:460800bps - Tera Term VT
File Edit Setup Control Window Help

*
*      noForth Tv 230307      *
M0+ core with ARMv6-M ISA
*

OK.0
@)words
PIO-PROG PIOBASE\ DAS\ DAS MDAS -TRAILING ASM\ TOOLS\ SEE MSEE
.S [THEN] [IF] [ELSE] NOFORTH\ TO DOES> UNUSED [''] IMMEDIATE END-CODE
CODE REPEAT ELSE WHILE BEGIN THEN WORDLIST ALLOT [CHAR] \ ' EVALUATE
VOCABULARY QUIT MS >NUMBER FIND .( ( CHAR PARSE REFILL ACCEPT
." S" C" DEPTH ] [ RECURSE POSTPONE /STRING HEX DECIMAL .R U.R
. U. D. #> #S # SIGN HOLD <# TYPE SPACES SPACE CR EMIT KEY?
KEY CELLS CELL+ >BODY ALIGN COMPILE, C, , ALIGNED ABORT CATCH
THROW MOVE FILL */ */MOD MOD / /MOD SM/REM FM/MOD M* D+ DABS
DNEGATE ABS NEGATE D2/ D2* - + 2/ 2* 1- 1+ RSHIFT LSHIFT INVERT
XOR OR AND S>D 0<> 0= <> = 0< WITHIN U< < U> > FALSE TRUE
MAX MIN ROT PICK NIP TUCK SWAP OVER DROP ?DUP DUP 2ROT 2SWAP
2OVER 2DUP 2DROP 2R@ R@ 2R> 2>R R> >R COUNT @ C@ +! ! C! SOURCE-ID
>IN STATE BASE BL UM/MOD * UM* J I +LOOP LOOP UNLOOP LEAVE ?DO
DO AGAIN UNTIL IF AHEAD ; 2LITERAL LITERAL VALUE VARIABLE CONSTANT
:NONAME : CREATE EXIT EXECUTE OK.0
@)
```

Booted with tools, assembler, disassembler & PIO-base

noForth T a little different

VALUE	With initialization value
GROW	Modifying dictionary size
FREEZE & FREEZE2	You can save two (different) images
COLD & COLD2	And boot up two (different) images
noForth T duo	Launches a 2nd noForth on CPU core-1
PIO (dis)assembler	All options, export feature & (mini-pio.f)
Core-1.f	Run and stop code on second CPU core

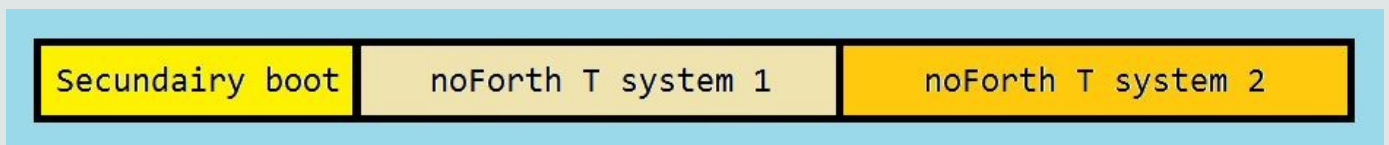
noForth T variants

- **noForth T** (Without vocabularies)
- **noForth Tv** (With vocabularies)
- **noForth T duo** (Two noForth's on one RP2040)
- **noForth Tv duo** (Ditto with vocabularies)

Details:

- **FREEZE** saves the (startup) image
- **FREEZE2** saves a second (alternative) image
- Switching between images is done with **COLD** and **COLD2**
- System clock 12MHz to 250MHz
- **Authors:** J.J. Hoekstra, Willem Jager, Albert Nijhof and myself
- The metacompiler runs on Win32Forth
- A number of extensions like: Assembler, disassembler, comma code, dozens of PIO examples, mini-PIO, config, ADC, etc.

Structure of noForth T duo image:



To do:

- Interrupt examples
- Calling other ROM functions (QSPI, Floating point)
- Getting the NOF file system working properly (QSPI)
- noForth serial port and power supply through USB-port?

noForth T configuration:

(* Changing the configuration of noForth T duo

12MHz I=8.3mA, 30MHz I=12.5mA, 132MHz I=29.5mA, 250MHz=38.9mA

CFG = Clock frequency in MHz
CFG 04 + = Used UART (0 or 1) only 0 is valid for now
CFG 08 + = Baudrate in bits per second
CFG 0C + = Used GPIO pin for S?
CFG 10 + = Boot method

Valid data for these parameters are:

Clock = 12, 30, 60, 120, 125, 132, 200 & 250 MHz
Uart = 0 (will be upgraded when this version is stable)
Baudrate = Any baudrate like 9600, 115200 until 921600 was tested ok
S? pin = GPIO 24, but any free GPIO pin will do
Boot = 0 = Boot only low image, 1 = Boot low & load second image
 2 = Load & boot both images

*)

decimal cfg

250 over ! cell+ \ Set max. freq. in MHz

0 over ! cell+ \ Use UART-0

460800 over ! cell+ \ Baudrate is 460k8

24 over ! cell+ \ Use GPIO-24 for S?

2 over ! drop \ Load & start second image too

hex

cfg config \ Test new configuration

\ freeze \ Save new config, boots at startup & when you type COLD

\ freeze2 \ Save as spare system, boots when you type COLD2

\ End

Code part that boots noForth T

```
code WARM      ( -- )
  here ivecs cell+ vec! \ Install WARM at second location in IVECS

  apool,          \ 4 - Save PC in W & jump over data
    ramadr: RP0 , \ HOP = Bottom return stack
    ramadr: SP0 , \ DAY = Bottom data stack
    origin ,      \ SUN = Address of DOES-code
    amsterdam ,   \ MOON = Address of high level WARM part
  then,
  w { hop day sun moon } ldm, \ 5 - Read all startup data
  hop hop ) ldr,      \ 2 - Read RP0
  rp hop mov,        \ 1 - Set return stack to RP0
  sp day ) ldr,      \ 2 - Set data stack to SP0
  does sun mov,      \ 1 - Set DOES register
  ip moon mov,       \ 1 - Set IP to noForth high level start
  next,             \ 6
end-code
```

Start program on second core

```
\ Start assembly code routine on core-1
\ Core-1 access sequence: 0, 0, 1, vectortable, sp, pc

: BOOT1      ( code-addr -- )
  1 or >r reset1 \ Set thumb bit & reset core-1
  begin begin begin begin begin begin
    empty-fifo \ Clear incoming FIFO
  0 >cmd? until \ Start with access sequence, succeed?
  0 >cmd? until \ Second step, succeed?
  1 >cmd? until \ Third step, succeed?
  20000000 >cmd? until \ Sent interrupt table, succeed?
  tib/ 100 + >cmd? until \ Sent stack pointer for core-1, succeed?
  r@ >cmd? until rdrop ; \ Sent core-1 PC address, succeed?
```

The change in CONFIG

```
dup h@ ?dup if \ Multi core?
  2 = if \ Start other core?
    memtop cell+ @ boot1 \ Yes, do so
  then
then
```

