noForth

13nov2025

Special features of noForth t

Hard faults

NoForth provides a brief explanation when a hard fault occurs. After

```
need hardfault
```

you will receive more information about the circumstances in which hard errors occur. Example:

```
: t1 noop 2 @ ;
: t2 noop t1 ;
: t3 noop t2 ;
```

Now execute t3 .

```
@)t3

A hardfault occurred @ used in T1  TOS: 2
Called by: T2 <- T3 <- INTERPRET
 OK.0
```

In T1, an @ is performed on a non-cell-aligned address. The second line shows the word nesting in reverse order.

SWITCH

With `SWITCH ( -- )`, you can switch back and forth between the noForths on core-0 and core-1 in the same terminal (only for noForth t duo).

```
 switch <enter> <enter>
```

Yes, you got that right, you have to press twice to get the prompt from the other core.

IRQ!

`IRQ! (routine-address irq-number --)` puts the routine address in the IRQ vector.

- For more information, see page 60 of "rp2040-datasheet.pdf".
- The vector table can be found on page 30 of "DUI0662B_cortex_m0p_r0p1_dgug.pdf".
- Examples of use can be found on [the noForth GitHub page.]

GROW

With `GROW ( n -- )`, you can increase or decrease the dictionary space by n bytes (only for noForth t solo). You can use it, for example, to keep the binary for a rounded APP as small as possible.

`hx 10000 grow` makes noForth 10000 bytes larger. The protected limit is the end of the available RAM memory.

`hx -10000 grow` makes noForth 10000 bytes smaller. The protected lower limit is HERE.

COLD and FREEZE in noForth t solo/duo

The image of noForth t solo is stored in ROM at address FROZEN. For noForth t duo, the images of the noForths for the two cores are stored consecutively in ROM at address FROZEN. These two noForths can therefore be different, but COLD and FREEZE always treat them as a single entity.

`COLD ( -- )` copies the FROZEN image(s) to RAM and starts the noForth(s) there. The same thing happens with an autostart after a power failure.
`FREEZE ( -- )` does the opposite: it copies the running noForth(s) to FROZEN in ROM, overwriting the old image(s).
`FREEZE2 ( -- )` copies the running noForth(s) to FROZEN2 (in ROM). Autostart will never use the image(s) in FROZEN2.
`COLD2 ( -- )` can copy the FROZEN2 image(s) to RAM and start them.

IMAGE - how do you create a UF2 file from a frozen noForth t?

```
need image
```

The Intel-Hex generator is now loaded.

```
core image
```

An Intel-Hex text will now appear in the terminal.

1. Copy the Intel-Hex text and save it on your computer as a text file `ccc.hex` .
2. Include `BUILD-UF2-V3.F` in Win32Forth.
3. Type `HEX>UF2 ccc` in Win32-Forth. Leave out the extension of ccc, because the UF2 generator does not accept extensions. The result is saved as `ccc-CURRENTDATE.UF2` . You may distribute this file without rights.

`CORE+ IMAGE` creates an Intel-Hex file from the noForth(s) saved with FREEZE plus the noForth(s) saved with FREEZE2.
With `CORE+LIB IMAGE` the library is included as well.

*