

FORTH AND pijFORTHos

- ▶ As all Forths is based on and fosters interactive programming
- ▶ Several Forth standards have been proposed, and somewhat adopted, but in the end each environment has its peculiarities
- ▶ Other, more recent languages, Python for instance, call this programming modality REPL (Read-Eval-Print Loop)
 - ▶ The pijFORTHos 0.1.8 environment is based on a small assembly implementation of a Forth interpreter called JonesForth written in literate x86 assembly by Richard W.M. Jones

```
rpi3bp — picocom --b 115200 /dev/cu.usbserial-00000000 — 80x33
Chute:rpi3bp dip$ picocom --b 115200 /dev/cu.usbserial-00000000
picocom v3.1

port is          : /dev/cu.usbserial-00000000
flowcontrol     : none
baudrate is     : 115200
parity is       : none
databits are    : 8
stopbits are    : 1
escape is       : C-a
local echo is   : no
noinit is       : no
noreset is      : no
hangup is       : no
nolock is       : no
send_cmd is     : sz -vv
receive_cmd is  : rz -vv -E
imap is         :
omap is         :
emap is         : crclrf,delbs,
logfile is      : none
initstring      : none
exit_after is   : not set
exit is         : no

Type [C-a] [C-h] to see available commands
Terminal ready
;-)
pijFORTHos 0.1.8 sp=0x00008000
2 3 + .
5 █
```

pijFORTHos and JonesForth

- ▶ The pijFORTHos 0.1.8 environment is based on a small assembly implementation of a Forth interpreter called JonesForth written in literate x86 assembly by Richard W.M. Jones
 - ▶ For its simplicity JonesForth has been adapted to many different architectures
 - ▶ It is defined in a single Assembly source file
 - ▶ For the user interface, JonesForth only needs two low-level subroutines to read from the stdin and write to the stdout a single char: getchar and putchar
- ▶ JonesForth is not particularly efficient nor it follows any Forth standard but, in spite of its small size, has a lot of features

```

rpi3bp — picocom --b 115200 /dev/cu.usbserial-00000000 — 80x33
Chute:rpi3bp dip$ picocom --b 115200 /dev/cu.usbserial-00000000
picocom v3.1

port is      : /dev/cu.usbserial-00000000
flowcontrol  : none
baudrate is  : 115200
parity is    : none
databits are : 8
stopbits are : 1
escape is    : C-a
local echo is : no
noinit is    : no
noreset is   : no
hangup is    : no
nolock is    : no
send_cmd is  : sz -vv
receive_cmd is : rz -vv -E
imap is      :
omap is      :
emap is      : crclrf,delbs,
logfile is   : none
initstring   : none
exit_after is : not set
exit is      : no

Type [C-a] [C-h] to see available commands
Terminal ready
;-)
pijFORTHos 0.1.8 sp=0x00008000
2 3 + .
5 █

```

FORTH AND pijFORTHos

► pijFORTHos source tree

```
Chute:pijFORTHos dip$ ls -la
total 416
drwxr-xr-x  28 dip  staff   896  4 Dic 08:44 .
drwxr-xr-x   8 dip  staff   256 28 Nov 19:45 ..
drwxr-xr-x  13 dip  staff   416  3 Dic 20:41 .git
-rw-r--r--   1 dip  staff    35 28 Nov 19:43 .gitignore
-rw-r--r--   1 dip  staff   471 28 Nov 19:43 AUTHORS
-rw-r--r--   1 dip  staff  7651 28 Nov 19:43 LICENSE
-rw-r--r--   1 dip  staff   879 28 Nov 19:43 Makefile
-rw-r--r--   1 dip  staff  4481 28 Nov 19:43 README.md
drwxr-xr-x   4 dip  staff   128 28 Nov 19:43 annexia
-rw-r--r--   1 dip  staff  2797 28 Nov 19:43 blinker.f
drwxr-xr-x   5 dip  staff   160 28 Nov 19:43 doc
drwxr-xr-x   4 dip  staff   128 28 Nov 19:43 firmware
-rw-r--r--   1 dip  staff  3485 28 Nov 19:43 jonesforth.f
-rw-r--r--   1 dip  staff 58385 28 Nov 19:43 jonesforth.s
-rwxr-xr-x   1 dip  staff 57064  3 Dic 23:59 kernel.sym
-rw-r--r--   1 dip  staff   200  3 Dic 23:58 loadmap
-rw-r--r--   1 dip  staff   200 28 Nov 19:43 loadmap.orig
-rw-r--r--   1 dip  staff   747 28 Nov 19:43 nqueens.f
-rw-r--r--   1 dip  staff  6259  3 Dic 23:59 raspberry.c
-rw-r--r--   1 dip  staff   535 28 Nov 19:43 raspi.h
-rw-r--r--   1 dip  staff  5481 29 Nov 18:17 serial.c
-rw-r--r--   1 dip  staff   941 28 Nov 19:43 serial.h
-rw-r--r--   1 dip  staff  1557 28 Nov 19:43 start.s
drwxr-xr-x  18 dip  staff   576 28 Nov 19:43 tests
-rw-r--r--   1 dip  staff  1145  3 Dic 23:54 timer.c
-rw-r--r--   1 dip  staff   482 28 Nov 19:43 timer.h
-rw-r--r--   1 dip  staff  3411 28 Nov 19:43 xmodem.c
-rw-r--r--   1 dip  staff   218 28 Nov 19:43 xmodem.h
```

FORTH AND pijFORTHos

- ▶ pijFORTHos
 - ▶ Features
 - ▶ JonesForth
 - ▶ Bootloader
 - ▶ Issues
 - ▶ Works on Pi 1 (FIXED!)
 - ▶ Edit line bug (FIXED!)
 - ▶ UART overrun (to be fixed)
 - ▶ ...

BLINKING LED IN FORTH

- ▶ To make the LED blink the phrases LED ON and LED OFF can be executed one after the other interleaving the executions with a delay. A possible way to do it simply requires the definition of the new word DELAY (n --):

```
LED ON 10000 DELAY LED OFF
```

- ▶ A second delay can be added to make the LED transition visible, for example by holding the LED on and off for the same amount of time:

```
LED ON 10000 DELAY LED OFF 10000 DELAY
```

