dec2025

noForth

noForth web site

The multitasker in noforth t

Example

```
need TASKER\    \ Load the multitasker from the library.
task: ONE       \ Define a task named ONE with default action NOOP.
```

The multitasker is now loaded and a task has been defined, which has the default task action NOOP.

```
0 value COUNTER
: COUNTING      \ The task action we want to assign to task ONE.
    0 to counter
    begin 1 +to counter 1 ms
    again ;
```

If a task action contains an infinite loop, PAUSE (task switch) must always occur within that loop. PAUSE is already built into the following words:

```
US   ( u -- )    Wait 'u' microseconds
MS   ( u -- )    Wait 'u' milliseconds
KEY  ( -- ch )
EMIT ( ch -- )
```

```
' counting one start-task
```

`START-TASK ( xt task-name -- )` installs the new action in the task and starts it. Any action still running in the task will be removed.
`SLEEP ( task -- )` Temporarily pause a task,
`WAKE ( task -- )` resumes it.
Observe the behavior of task ONE :

```
@)decimal  OK.0
:)counter . one sleep 30 ms counter .  9341 9341  OK.0
:)counter . one wake  30 ms counter .  9341 9371  OK.0
```

Overview of basic words for the multitasker

```
TASK:      ( ccc -- )      Define a task named 'ccc'.
PAUSE      ( -- )          Perform a task switch.
SLEEP      ( task -- )     Pause the task (for a moment)
WAKE       ( task -- )     Turn the task back on.
START-TASK ( xt task -- ) Install and start 'xt' in 'task'.
```

A task that stops itself (via EXIT) enters SLEEP mode. You can resume it with WAKE.

Extension of previous example

We define two tasks (TWO and THREE). Each task gets its own task control block, data stack, return stack, and space for number conversion.

```
\ TWO
need BOOTKEY?   \ Load from the library
: -COUNTING
    begin bootkey? if -1 +to counter then   2 ms
    again ;
task: TWO
' -counting two start-task  \ Install and activate task TWO
```

`BOOTKEY? ( -- flag )` Redefines the BOOT button on each RP2040 board as a switch. We now use COUNTER from task ONE in task TWO to count down (as long as the BOOT button on the board is pressed). ONE and TWO thus operate against each other.

```
:)0 to counter 10 ms counter . 1000 ms counter .  10 1008  OK.0
```

Now with boot key pressed:

```
:)0 to counter 10 ms counter . 1000 ms counter .  5 508  OK.0
```

```
\ THREE
hex
D0000020 constant GPIO-OE   \ GPIO output enable
D0000010 constant GPIO-OUT  \ GPIO output value
: FLASHING  ( -- )                      \ 1 Hz flashing led
    5 dm 25 gpio!                       \ Enable SIO on pin 25
    dm 25 bitmask GPIO-OE **bis         \ Bit 25 is output
    begin dm 25 bitmask GPIO-OUT **bix  \ Toggle LED
          200 ms
    again ;
task: THREE
' flashing three start-task
```

While you continue working in noForth, task THREE causes the LED on the PICO board to flash in the background.

TASKS

TASKS (--) shows the status of all defined tasks.

```
tasks
 Task  ?    Stack    Rstack   BASE   Error Action
---------------------------------------------------
 MAIN  wake 064/000  160/016  016      0   noForth
THREE  wake 016/001  032/010  016      0   FLASHING
  TWO  wake 016/001  032/010  016      0   -COUNTING
  ONE  wake 016/001  032/010  016      0   COUNTING
 USB2  wake 016/000  032/010  016      0   USB-TX
 USB1  wake 016/001  032/008  016      0   USB-HANDLER   OK.0
```

```
Explanation:
  Task     Task name
  ?        Wake or sleep?
  Stack    Maximum/actual number of items on the stack
  Rstack   Maximum/actual number of items on the return stack
  BASE     The value of BASE in that task
  Error    Error number (or text), 0 means 'no error'
  Action   Name of the task action
```

There are even more words available for the multitasker. For more information, see Project Forth Works