Multitasking

das Paket MTASK erlaubt kooperatives Multitasking.

Laden mit

```
1 GET MTASK
1 5 THRU
```

Glossar

MULTI (–)
Aktiviert das Multitasking

SINGLE (–)
Stoppt Multitasking, der aktuelle Task bleibt aktiv

TASK: (size –)
erzeugt einen neuen Task und initialisiert die User-Variablen
size ist der Speicherplatz für User-Variablen und den lokalen Stack. Hinter TASK: folgt der Taskname.
Beispiel:

```
1000 TASK: COUNTER1
```

ACTIVATE (task –)
Aktiviert einen Task. ACTIVATE darf nur in Colon-Definitionen genutzt werden. Hinter ACTIVATE folgt der im Task auszuführende Code.
Beispiel:

```
  : COUNT!   COUNTER1 ACTIVATE
    BEGIN PAUSE 1 COUNTS +!  AGAIN ;
```

SLEEP (task –)
Versetzt den Task in den Ruhezustand

WAKE (task –)
Reaktiviert den Task aus dem Ruhezustand

STOP (–)
Versetzt den aktuellen Task in den Ruhezustand

BACKGROUND: (–)
Definitionswort. Alternative zu ACTIVATE zum Erzeugen eines Tasks. Hinter BACKGROUND: folgt der Taskname und danach der auszuführende Code.
Der Task wird jedoch nicht gleich gestartet, sondern muss mit WAKE aktiviert werden.
Beispiel:

```
BACKGROUND: COUNTER2  BEGIN
 PAUSE 1 COUNTS +!  AGAIN ;
```

```
COUNTER2 WAKE
```

PAUSE (–)
Weiterschalten zum nächsten Task.
Damit kooperatives Multitasking überhaupt funktioniert, muss in jedem Task regelmäßig PAUSE aufgerufen werden. Mit PAUSE wird der nächste Task aktiv. In allen Basis-I/O-Worten ist PAUSE bereits enthalten ((KEY), (CONSOLE), (PRINT), BUFFER, BLOCK) und damit indirekt in (EMIT), (CR), „.“ etc., so dann man in interaktiven Tasks PAUSE nicht direkt einfügen muss. Bei Tasks ohne I/O muss dagegen PAUSE in den Taskschleifen explizit stehen.

Literatur

- https://www.bradrodriguez.com/papers/mtasking.html
- C.H. Thing „Inside F83“ http://forth.org/OffeteStore/1003_InsideF83.pdf
- R. Zech, Forth 83
- http://www.forth.org/fd/FD-V05N4.pdf ff.

Beispiel

```
VARIABLE COUNTS
BACKGROUND: COUNTER  BEGIN
    PAUSE 1 COUNTS +!  AGAIN ;
COUNTER WAKE  MULTI

COUNTS ?
COUNTS ?
COUNTS ?
SINGLE
```

Funktionsweise

Das Multitasking-System basiert auf dem des Laxen/Perry-F83 und wurde geringfügig ans FG FORTH83 angepasst. Die Beschreibung zur Funktionsweise und Nutzung des Multitasking-Systems entspricht daher der Dokumentation in C.H. Thing „Inside F83“ (http://forth.org/OffeteStore/1003_InsideF83.pdf).

Abweichungen:

- Beim Z80 wird statt INT80 der RST30 (RST 6) genutzt.
- instruction pointer ist Register BC
- return stack pointer ist Register IY

Das F83 ist bereits von vornherein auf Multitasking vorbereitet. Ein kooperatives Multitasking muss beim Taskwechsel so nur 4 Register sichern/restaurieren: SP (parameter stack pointer), RP (return stack pointer), IP (interpreter pointer) und UP (user area pointer). Das erledigen die Worte PAUSE und RESTART.

Jeder Task hat seinen eigenen User-Variablen-Bereich. Mit den User-Variablen ENTRY und LINK wird eine geschlossene Taskkette (round-robin-Task list) aufgebaut. Je nach Zustand des Task steht hier ein Sprung zum nächsten Task (wenn der Task im Ruhezustand ist) oder mittels RST 30 wird das Wort RESTART ausgeführt, dass den Task reaktiviert. Der Task wird dann bis zum nächsten PAUSE abgearbeitet, dann ist der folgende Task an der Reihe.

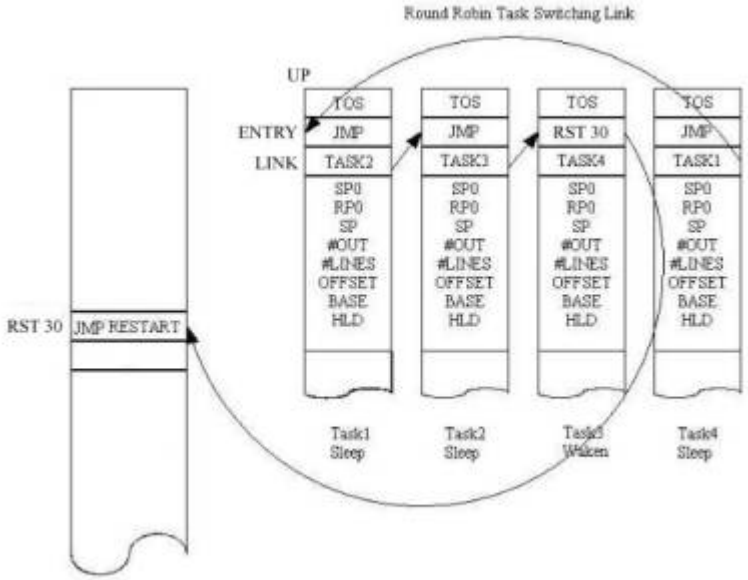


Bild nach „Inside F83“, leicht modifiziert

Hinweis: das Paket ist unabhängig von der Lage des Forth83 im Speicher; wird das Forth83 auf einen anderen Adressbereich gelinkt, braucht das Paket nicht angepasst zu werden!

From:
https://hc-ddr.hucki.net/wiki/ - **Homecomputer DDR**

Permanent link:
https://hc-ddr.hucki.net/wiki/doku.php/forth/fgforth/multitasking

Last update: **2025/07/21 08:51**