Sprungverteiler

SPRUNGVERTEILER SV: Hier handelt es sich nicht um ein Programm, wie Newcomer oft annehmen, es sind nur Standard-Adressen, in denen die Sprünge zu Adressen umgeleitet werden, die zu den aufgerufenen Standard-Routinen gehören. Ein Beispiel: Im KC-BASIC+ wird der Sprung zum HEADERSAVE auf den Adressen 149H (LORUF) und 157H (SARUF) eingetragen. Mein HEADERSAVE steht im Monitor, der eines anderen Rechner ab E000H. Steht nun im KC-BASIC+ im LORUF die Adresse FFF1H, und auf dieser die Adresse des jeweiligen LORUF vom HEADERSAVE, findet der KC-BASIC+ diese Routine ohne jede Änderung.

Der Sprungverteiler kann auch mit 'M' (modify) direkt eingetragen werden (RAM-Speicher muß aber dort vorhanden sein), wobei immer zuerst ein C3H (JMP), dann der niederwertige Teil der Adresse (letzten zwei Zchn.) und danach der höherwertige Teil (ersten zwei Zchn.) eingetragen wird. Wenn auf FFB7H ein C9H eingetragen wird, kann man die Sprünge zu Routinen, die nicht vorhanden sind, durch Eintrag dieser Adresse kurzschließen. (es geht dazu jedes C9H, was irgendwo im Monitor steht.)

Diesen eingetragenen SV kann man ausladen und nach jedem Kaltstart wieder einladen.

DIREKTANSPRÜNGE: Wer keine Möglichkeit hat, sich diesen SV einzutragen (z.B. 16-K), kann sich auch die Aufruf-Adressen der Standard-Routinen (z.B. LORUF/SARUF des HEADERSAVE) direkt eintragen. In allen Beschreibungen stehen die jeweiligen Adressen, wo diese Sprünge eingetragen werden. Hierbei bleiben ein eingetragenes CDH (CALL) oder C3H (JMP) stehen, es kommt der niederwertige, dann der höherwertige Teil der Rufadressen. Die Rufadressen stehen ebenfalls immer in den Beschreibungen zu diesen Standard-Programmen. Es ist deshalb immer sinnvoll, sich alle Beschreibungen gründlich durch zu lesen.

****

```
           Sprungverteiler fuer Z 1013-Erweiterungen
```

****

Der Zusatzsprungverteiler wurde entsprechend einer Einigung innerhalb der Interessengemeinschaft Heimcomputer des IZ an der TU-Dresden folgendermassen festgelegt:

Beginnt auf Adresse FFFFH abwaertsfuehrend:

FFFDH - JMP INKEY

```
holt ein Zeichen von Tastatur in den Akku;
                kommt beim 2. Aufruf nur zurueck, wenn Taste
                zwischendurch losgelassen wurde
```

FFFAH - JMP POLL

```
 bringt immer ein Zeichen im Akku zurueck, egal
                ob Taste losgelassen wurde oder nicht
```

FFF7H - JMP STAT

```
 uebergibt Tastaturstatus im Akku
                A=0   - keine Taste gedrueckt
                A=FFH - Taste gedrueckt
                die Abfrage erfolgt ohne Ruecksicht, ob die
                Taste schon vor dem Aufruf gedrueckt war und
                hinterlaesst trotz gedrueckter Taste den Status
                'letztes Zeichen war 0' ((Zelle 4)=0) um eine
                evtl. nachfolgenden INKEY-Routine nicht zu
                sperren
```

FFF4 - JMP SARUF

```
 ruft die SAVE-Routine des Headersave
                ! zerstoert 1. Registersatz + AF'
                Parameteruebergabe:
                Zellen 1BH - anfadr.
                       1DH - endadr.
                       23H - strtadr.
                Akku   3AH - Wiederholen der SAVE-Funktion mit
                             gleichem Kopf
                H(IY)  Typvorgabe (in ASCII), sonst 0
```

FFF1H - JMP LORUF

```
ruft LOAD-Routine des Headersave
                ! zerstoert 1. Registersatz + AF'
                Parameteruebergabe:
                Zellen 1BH - neue Anfangsadresse des Files
                             sonst 0
                Akku   0   - ohne signifikante Kontrolle
                       4EH - mit signifikanter Kopfkontrolle
                             (Typ) + Namenabfrage
                H(IY)  0   - Typ wird abgefragt
                       Typkennzeichen (in ASCII), keine Typabfr.
                L(IY)  20H - Freigabe Autostart bei COM-Files
```

FFEEH - JMP ZMINI

```
Initialisierung der Z-Monitorrufe auf B0H
```

FFEBH - JMP DRDEL

```
setzt den logischen Druckertreiber zurueck
```

FFE8H - JMP DRAKK

```
uebergibt den Akkuinhalt an den logischen
                Druckertreiber
```

FFE5H - JMP BSDR

```
 druckt den Inhalt des BWS und kehrt in das
                rufende Programm zurueck
```

FFE2H - JMP HARDC

```
uebergibt den Akkuinhalt an logischen Drucker-
                treiber wenn ein Flag im Rechner gesetzt ist;
                wandelt CR (0DH) in NL (1EH/0DH-0AH)
                ! nur verwenden, wenn Programm eine eigene
                Bildschirmverwaltung hat
```

FFDFH - JMP DRZEL

```
wie DRAKK, nur das der Inhalt von 1BH uebergeben
                wird (vorgesehen, um im BASIC mit POKE zu drucken)
```

FFDCH - JMP BEEP

```
 erzeugen eines kurzen Signals
```

FFD9H - JMP ASTA

```
 Ausgabe Akkuinhalt als ASCII-Zeichen an PUNCH
```

FFD6H - JMP BSTA

```
 Ausgabe Akkuinhalt als Byte an PUNCH
```

FFD3H - JMP AIN

```
  Eingabe eines ASCII-Zeichens vom LBL in den Akku
```

FFD0H - JMP BIN

```
  Eingabe eines Bytes vom LBL in den Akku
```

FFCDH - JMP DRINI

```
Initialisierung des logischen Druckertreibers
```

FFCAH - JMP ZEIDR

```
uebergibt ein Zeichen im Akku an physischen
                Druckertreiber
```

FFC7H - JMP BLMK

```
 Lesen eines Blocks vom Headersave
                Parameteruebergabe:
                Zellen 25H/26H * Kopfinhalt des zu lesenden Bl.
                HL             * Ladeadresse des Blocks
               Return:
                Zellen 25H/26H * Kopfinhalt + 20H
                HL             * HL:=HL+20H
               Abbruch des Lesens bei Kopfinhalt=0FFFFH
               oder DMA > Endadr in ARG2 (1DH)
```

FFC4H - JMP BSMK

```
 Schreiben eines Blocks im Headersave
                Parameteruebergabe:
                HL  * Quelladresse Block
                IX  * Kopfinhalt
                DE  * Anzahl der Sync.-Bits
               Return:
                HL  * HL:=HL+20H
```

FFC1H - JMP SUCHK

```
Suchen eines Kopfblocks und Uebergabe des Inhalts
                im Kopfpuffer (E0-FF), keine Auswertung
```

FFBEH - JMP AKP

```
  Aufbereitung Kopfpuffer mit Namenabfrage
                Parameteruebergabe wie bei SARUF
```

FFBBH - JMP GETST

```
Abfrage der Joysticks und Uebergabe des Er-
                gebnisses in BC (B-links,C-rechts) mit folgen-
                den Bit-Bedeutungen (Belegung mit 1):
                Bit 0 - links
                    1 - rechts
                    2 - runter
                    3 - hoch
                    4 - Aktionstaste
                 Z-Flag=1, wenn keine Betaetigung vorliegt
                CY-Flag=1, wenn Spielhebel nicht angeschlossen
```

FFB8H - JMP SOUND

```
Ausgabe einer vollen Periode auf die Tonband-
                buchse, sowie auf Bit 7 vom Systemport (User-P)
                Uebergabe der Periodendauer in C mit
                T=n*33mks+20mks (2MHz)
```

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

Permanent link:
https://hc-ddr.hucki.net/wiki/doku.php/z1013/software/sprungverteiler?rev=1316098505

Last update: **2011/09/15 14:55**