

```

10 LC=0:COLOR0,1:COLOR4,1:COLOR5,2
100 NAME$="{CLR,7RIGHT,7DOWN}C64 DISKETT
E"
110 PRINT "{CLR,5DOWN,9RIGHT}AUTOMATISCHE
R WECHSEL"
120 PRINT "{DOWN,11RIGHT}IN DEN 64'ER MOD
US"
130 OPEN15,8,15,"I0":A$=CHR$(0)
140 IFDS<>0THENBEGIN
150 PRINT "{2DOWN,RIGHT}DISKETTEN FEHLER:
";DS$
160 CLOSE8:CLOSE15
170 PRINT "{2DOWN,RIGHT}ABBRUCH":END
180 BEND
190 OPEN8,8,8,"#"
200 PRINT#15,"U1:8 0 18 0":PRINT#15,"B-P
";8;5
210 GET#8,BA$
220 PRINT#15,"U1:8 0 1 0"
230 FORI=0TO24:GET#8,S$:BL$=BL$+CHR$(ASC
(S$)):NEXT
240 IF (ASC(BA$)AND1)=0THENBEGIN
250 IFLEFT$(BL$,3)="CBM"THENPRINT "{2DOWN
,6RIGHT}BOOT-SECTOR SCHON VORHANDEN"
270 IFLEFT$(BL$,3)<>"CBM"THENPRINT "{2DOW
N,5RIGHT}PROGRAMM LIEGT AUF BOOT-SECTOR"
280 PRINT "{2DOWN,4RIGHT}INSTALLIERUNG FO
RTSETZEN? (J/N)":LC=1

```

```

290 GETKEYX$
300 IFX$<>"J"THENBEGIN
310 CLOSE8:CLOSE15
320 PRINT "{2DOWN,5RIGHT}ABBRUCH":END:BE
N
330 BEND
340 PRINT "{DOWN,5SPACE}INSTALLIEREN DES
BOOT-SEKTORS {CTRL-@}&{GREEN}Y {CTRL-A,LIG
.GREEN}" BITTE ENTFERNEN SIE EINEN EV
TL:
350 PRINT "{5SPACE}VORHANDENEN SCHREIBSCH
UTZ UND {CTRL-@}&{BLUE}X {CTRL-A,LIG.GREEN
}" DRUECKEN SIE EINE TASTE
370 GETKEYX$
380 PRINT#15,"B-P 8 0"
390 PRINT#8,CHR$(67);CHR$(66);CHR$(77);A
$;A$;A$;A$;NAME$;A$;
400 PRINT#8,A$;CHR$(DEC("20"));CHR$(DEC(
"4D"));CHR$(DEC("FF"));A$
410 PRINT#15,"U2: ";8;0;1;0
415 IFLC=0THENPRINT#15,"B-A 0 1 0"
420 CLOSE8:CLOSE15
430 PRINT "{4SPACE}BOOT-SECTOR INSTALLIER
T":END

```

664'er

Listing 1. Autochange 128/64. Bitte im C 128-Modus eingeben.

Das Maß der Dinge

Das exakte Ausmessen eines Unterprogrammes ist eine große Hilfe, um Laufzeiten zu verringern. Meist steht jedoch keine hinreichend genaue Uhr zur Verfügung. Wir bieten Ihnen zwei Lösungen an, eine für Maschinensprache- und eine für Basic-Programme.

Um die Laufzeiten von Programmen genau messen zu können, reicht die Stoppuhr für den 100-Meter-Lauf nicht mehr aus. Vor allem bei Maschinenprogrammen, bei denen die menschliche Reaktionsfähigkeit weit hinter der geforderten Genauigkeit bleibt. Doch auch die eingebaute interruptgesteuerte Uhr TI\$ im C 64 versagt, wenn es um das Auszählen sehr kurzer Maschinenprogramme geht. Ladevorgänge können damit überhaupt nicht erfaßt werden, da TI\$ während des Ladens stillsteht. Abhilfe schaffen hier die Timer der CIAs.

Taktzyklen zählen

Um Maschinensprache-Programme zu messen, ist es am besten, die Taktzyklen zu zählen. Dies liefert einen gut zu vergleichenden Wert. Geben Sie zunächst das Programm »Taktzyklen« (Listing 1) mit dem MSE ein und speichern Sie es. Aufgerufen wird die Routine durch den Befehl
SYS 49152,Startadresse <RETURN>

Die anzugebende Startadresse entspricht der des zu untersuchenden Maschinenprogrammes. Dieses Unterprogramm wird nun ausgeführt und die dafür benötigte Zeit bestimmt. Während dieses Ablaufes ist der Bildschirm ausgeblendet, er nimmt die Farbe des Rahmens an. Am Ende des zu messenden Programmes muß ein RTS stehen. Nach Programmende gibt das Programm den gemessenen Wert aus und springt in den Direkt-(READY-)Modus. Der angezeigte Zah-

lenwert drückt die Anzahl der Systemtakte aus, die während der Abarbeitung des Maschinen-Unterprogrammes durch den Mikroprozessor verstrichen sind. Da ein solcher Takt etwa 1,015 Mikrosekunden dauert, läßt sich durch Multiplizieren mit diesem Faktor die benötigte Zeit in Mikrosekunden errechnen. Bei eingeschaltetem Bildschirm vergrößert sich das Ergebnis um etwa 5 bis 6 Prozent.

Beschreibung des Programms »Taktzyklen«

Die beim Aufruf des Programms übergebene Startadresse des zu messenden Unterprogramms wird als Operand für einen Sprungbefehl gespeichert. Anschließend wird die Möglichkeit der Unterbrechung durch einen IRQ ausgeschaltet, weil sonst ein falsches Resultat geliefert werden kann. Nach dem Ausblenden des Bildschirms lädt das Programm beide Timer des CIA 2 mit dem Maximalwert und koppelt sie miteinander. Sobald die Zähler gestartet sind, erfolgt der Sprung in das Unterprogramm, nach dessen Ende die Zähler sofort wieder gestoppt werden. Schließlich werden neben der Einrichtung des Normalzustandes — den Bildschirm und den IRQ betreffend — die Timer-Bytes ausgewertet. Dazu invertiert das Programm sämtliche Bits der von den Timern gelieferten Werte, da beide Registerpaare heruntergezählt werden. Zusätzlich werden noch elf Takte subtrahiert, die zum Starten beziehungsweise Stoppen der Timer und zum Aufruf des Unterprogramms nötig sind. Damit erhält man einen Meßbereich, der zwischen 1 und $2^{32}-11$ Taktzyklen (entsprechend einer Zeit von 1 µs bis 1 Stunde 12 Minuten) liegt.

Laufzeit-Meß-System »LMS«

Das LMS wurde entwickelt, um die Laufzeit von Basic-Programmen zeilenbezogen messen zu können. Weiterhin sollte

die Anzahl einzelner Aufrufe von Programmzeilen festgestellt werden können.

Einsatzmöglichkeiten

Das LMS ist im wesentlichen als Werkzeug für folgende Anwendungsfälle konzipiert.

1. Performance-Verbesserung von Basic-Programmen

Durch den Einsatz des LMS ist sofort erkennbar, wo die zeitmäßig kritischen Stellen eines Programms sind. Oft kann durch einige Veränderungen von Basic-Statements wertvolle Laufzeit eingespart werden. Sollte das nicht ausreichen, so ist der Einsatz von Maschinenroutinen angebracht.

Die Erfahrung zeigt, daß es selten notwendig ist, ein Programm vollständig in Assembler zu schreiben. Meist bringt ein begrenzter Einsatz von Maschinenroutinen an zeitkritischen Stellen ausreichende Geschwindigkeit unter Beibehaltung der Vorteile von Basic. Diese sind vor allem wesentlich bessere Lesbarkeit und Wartung sowie geringerer Erstellungsaufwand.

2. Testhilfe

Das Laufzeit-Meß-System stellt die Anzahl der Aufrufe von Basic-Zeilen fest. Es ist daher damit leicht überprüfbar, ob Programmzeilen überhaupt durchlaufen werden und wenn ja, ob die Anzahl der Aufrufe stimmt.

Mit dem LMS können auch Statements gemessen werden, die mit TI\$ — abgesehen von der geringen Genauigkeit von TI\$ — zu erfassen sind (zum Beispiel INPUT#).

Bedienungsanleitung

Geben Sie zuerst die Listings 3 bis 5 ein und speichern Sie diese auf einer Diskette. Nach dem Laden und Starten des Programms »LMS« erscheint nach kurzer Zeit ein Menü mit folgenden Wahlmöglichkeiten:

Messen (m)

Durch die Auswahl der Funktion »Messen« wird das eigentliche Meßprogramm aktiviert. Dies ist immer erkennbar an der roten Rahmenfarbe. Es wird eine Einschaltmeldung geschrieben und danach in den READY-Modus verzweigt. Jetzt muß das zu messende Programm eingegeben oder geladen werden.

Bei jedem Start des Programms werden für jede Programmzeile Laufzeit und Aufrufanzahl gespeichert. Wichtig ist, daß das Programm mit RUN gestartet wird. Ein Start mit GOTO oder CONT ist möglich, führt aber nicht zur Messung.

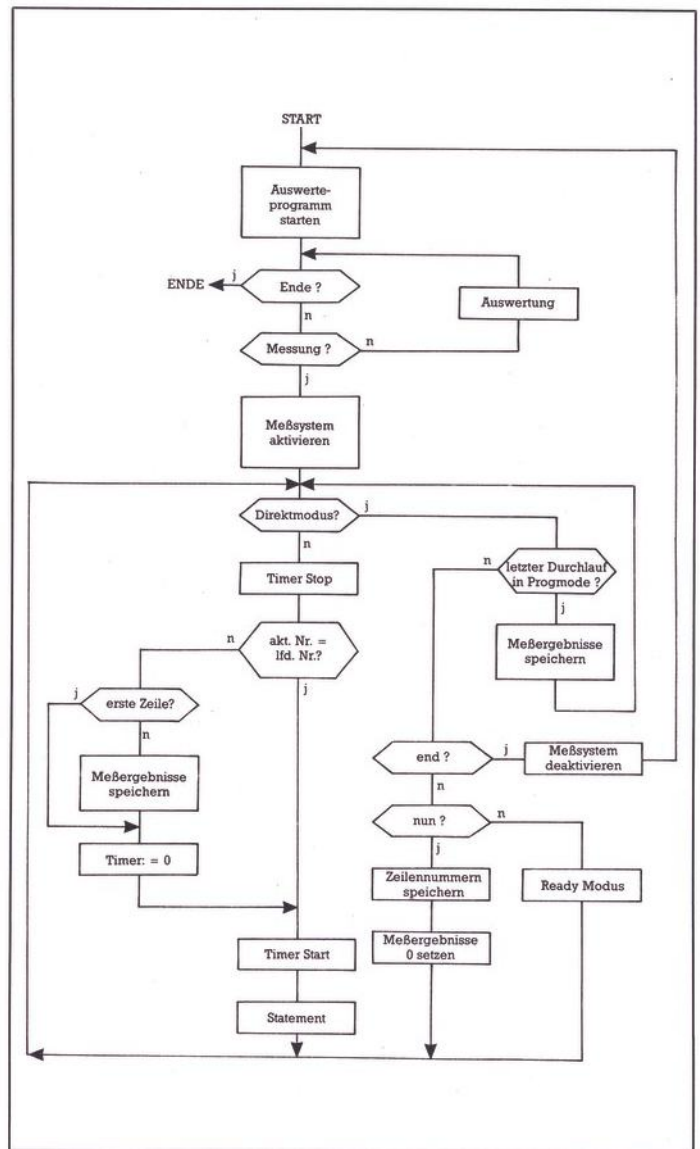


Bild 1. Flußdiagramm zum Laufzeit-Meß-System »LMS«. Nach diesem Schema berechnet man die Laufzeit eines Basic-Programms.

programm : taktzyklen c000 c061

```
c000 : 20 fd ae 20 8a ad 20 f7 55
c008 : b7 8c 2f c0 8d 30 c0 78 38
c010 : a9 0b 8d 11 d0 ad 12 d0 29
c018 : d0 fb a2 03 a9 ff 9d 04 08
c020 : dd ca 10 fa a9 11 a2 51 16
c028 : 8e 0f dd 8d 0e dd 20 ff b7
c030 : ff a0 00 8c 0e dd 8c 0f 31
c038 : dd a9 1b 8d 11 d0 58 38 cc
c040 : ad 04 dd 49 ff e9 0b 85 16
c048 : 65 a2 03 b9 05 dd 49 ff 5b
c050 : e9 00 95 61 c8 ca d0 f3 d9
c058 : 38 8a a2 a0 20 4f bc 4c 9a
=060 : d7 bd 00 ff 00 ff 00 ff 16
```

Listing 1. »Taktzyklen« geben Sie bitte mit dem MSE ein.

```
100 sys9*4096
110 .opt oo
120 ;
130 ;
140 ; taktzyklen eines
150 ; maschinenprogramms messen
160 ;
170 ;
180 ; von mark richters
190 ; allerstr.4
200 ; 2806 oyten
```

```
210 ; tel. 04207/1870
220 ;
230 ;
240 *=$c000
250 ;
260 fac =#62 ;fliesskomma-akku
270 frmnum =#ad8a ;ausdruck holen
280 chkcom =#aefd ;auf komma pruefen
290 getadr =#b7f7 ;2-byte integer
300 vicctrl =#d011 ;reg. fuer bs aus
310 raster =#d012 ;raster-zeile
320 timer =#dd04 ;timer a lo-byte
330 cra =#dd0e ;timer a control
340 crb =#dd0f ;timer b control
350 ;
360 jsr chkcom ;startadresse der
370 jsr frmnum ;zu messenden
380 jsr getadr ;routine holen
390 sty jsubr+1 ;und als sprung-
400 sta jsubr+2 ;adresse speichern
410 ;
420 sei ;irq sperren
430 lda #00001011 ;bildschirm aus
440 sta vicctrl
450 wait lda raster ;warten bis bild-
460 bne wait ;schirm ganz aus
470 ;
480 ldx #3 ;beide timer auf
490 lda #fff ;maximalwert
500 settimer sta timer,x
510 dex
520 bpl settimer
530 ;
540 lda #00010001 ;timer koppeln,
550 ldx #01010001 ;laden und starten
560 stx crb
570 sta cra
```

```
580 ;-----
590 jsubr jsr $ffff ;sprung auf routine
600 ;-----
610 ldy #0 ;beide timer stop
620 sty cra
630 sty crb
640 lda #00011011 ;bildschirm wieder
650 sta vicctrl ;an
660 cli ;irq wieder frei-
670 ;
680 sec ;4 timer-bytes
690 lda timer ;umrechnen
700 eor #fff ;fuer aufruf be-
710 sbc #11 ;noetigte zyklen
720 sta fac+3 ;abziehen
730 ldx #3 ;und die restlichen
740 ;
750 l1 lda timer+1,y ;3 timer-werte
760 eor #fff ;anpassen
770 sbc #0
780 sta fac-1,x
790 iny
800 dex
810 bne l1
820 ;
830 sec
840 txa ;4 byte werten
850 ldx #a0 ;ohne vorzeichen
860 jsr $bc4f ;in fließkommazahl
870 jmp $bdd7 ;fac ausgeben/rts

ready.
```

Listing 2. Information für Profis. Quell-Code-Listing zu »Taktzyklen«


```

100 REM ----- LAUFZEIT-MESS-SYSTEM <146>
110 REM -----LADEPROGRAMM <109>
120 REM ----- <166>
130 REM ----- FRANZ STOIBER <150>
140 REM ----- BRAEUNLICHGASSE 30 <036>
150 REM ----- A-2700 WR.NEUSTADT <194>
160 REM ----- OESTEREICH <197>
170 REM ----- <216>
180 IF L=1 THEN 260 <045>
190 POKE 53280,15: POKE 53281,15: POKE 646
,6 <009>
200 GA=8 <033>
210 PRINT CHR$(14); "{CLR,11DOWN,6RIGHT}LAU
FZEIT-MESS-SYSTEM(2SPACE)V2.0" <122>
220 PRINT "{7RIGHT}(C) FRANZ STOIBER(2SPAC
E)11/85" <237>
230 PRINT "{2DOWN,12RIGHT}BITTE WARTEN" <096>
240 POKE 646,15 <224>
250 L=1: LOAD "LMS.PHA",GA,1 <085>
260 SYS 40698 <084>
270 PRINT "{HOME}"; <202>
280 PRINT "{2DOWN}SYS 40780" <137>
290 PRINT "{2DOWN}LOAD ";CHR$(34);"LMS.BAS
";CHR$(34);" ";GA <189>
300 PRINT "{4DOWN}RUN" <234>
310 FOR I=631 TO 633: POKE I,13: NEXT <226>
320 POKE 198,3 <103>
330 PRINT "{HOME}"; <006>
340 REM ----- <132>
350 REM FUER DAS ARBEITEN MIT DATASSETTE <110>
360 REM SIND FOLGENDE ZEILEN ZU AENDERN: <217>
370 REM <178>
380 REM *** 200 GA=1 <087>
390 REM *** 210 ENTFAEHLT <174>
400 REM *** 220 ENTFAEHLT <057>
410 REM *** 230 ENTFAEHLT <195>
420 REM *** 240 POKE 646,0 <247>
430 REM *** 270 PRINT "{CLR}"; <007>
440 REM *** 300 PRINT "{5DOWN}RUN" <103>
450 REM <002>
460 REM ----- <252>
470 REM AM BAND MUESSEN DIE PROGRAMME <214>
480 REM IN DER REIHENFOLGE <215>
490 REM <042>
500 REM LMS, LMS.PHA, LMS.BAS <074>
510 REM <062>
520 REM GESPEICHERT WERDEN. <123>

```

© 64'er

Listing 3. Das Ladeprogramm »LMS« geben Sie bitte mit dem Checksummer V 3 (Ausgabe 3/86, Seite 55) ein.

programm : lms.pha 9ce0 9fff4

```

9ce0 : 20 e1 9e a9 83 a2 9f 85 a4
9ce8 : f9 86 fa a2 53 20 c3 9e ba
9cf0 : ad 08 03 8d dc 9f ad 09 a7
9cf8 : 03 8d dd 9f a9 0c 8d 08 6e
9d00 : 03 a9 9d 8d 09 03 20 67 e9
9d08 : 9f 4c 74 a4 20 e1 9e a5 56
9d10 : 9d f0 03 4c 7b 9d ad 0e e7
9d18 : dd 29 fe 8d 0e dd ad 0f a0
9d20 : dd 29 fe 8d 0f dd a5 39 ec
9d28 : cd d6 9f d0 07 a5 3a cd 84
9d30 : d7 9f f0 32 ad d7 9f c9 05
9d38 : ff f0 03 20 80 9e a5 39 7a
9d40 : 8d d6 9f a5 3a 8d d7 9f 84
9d48 : a9 ff 8d 04 dd 8d 05 dd ef
9d50 : 8d 06 dd 8d 07 dd ad 0e 3c
9d58 : dd 29 d7 8d 0e dd ad 0f 16
9d60 : dd 29 d7 8d 0f dd 78 ad 96
9d68 : 0e dd 09 01 8d 0e dd ad e3
9d70 : 0f dd 09 41 8d 0f dd 58 52
9d78 : 4c e4 a7 20 73 00 c9 80 84
9d80 : f0 0a c9 8a d0 03 20 b6 4c
9d88 : 9d 4c 3d 9e 20 67 9f ad 86
9d90 : dc 9f 8d 08 03 ad dd 9f f5
9d98 : 8d 09 03 a9 52 8d 77 02 13
9da0 : a9 55 8d 78 02 a9 4e 8d 28
9da8 : 79 02 a9 0d 8d 7a 02 a9 36
9db0 : 04 85 c6 c6 02 03 20 43 f5
9db8 : 9e ad 02 03 8d da 9f ad 97
9dc0 : 03 03 8d db 9f a9 cf 8d c5
9dc8 : 02 03 a9 9e 8d 03 03 a9 da
9dd0 : ff 8d d6 9f 8d d7 9f a9 a9
9ddb : 00 85 fd a9 d0 85 fe a0 c6

```

```

9de0 : 00 b9 7b 9f 91 fd c8 c0 3d
9de8 : 08 d0 f6 a5 2b 85 fb a5 e5
9df0 : 2c 85 fc 20 b5 9e a0 00 f5
9df8 : b1 fb 8d db 9f c8 b1 fb 24
9e00 : 8d d9 9f d0 12 ad d8 9f ad
9e08 : d0 0d a9 ff a0 00 91 fd 15
9e10 : c8 91 fd 20 4d 9e 60 a0 b1
9e18 : 02 b1 fb a0 00 91 fd a0 cb
9e20 : 03 b1 fb a0 01 91 fd a7 f7
9e28 : 00 c8 91 fd c0 07 d0 f9 2c
9e30 : ad d8 9f 85 fb ad d9 9f b6
9e38 : 85 fc 4c f3 9d 20 79 00 8e
9e40 : 4c e7 a7 48 78 a5 01 29 7e
9e48 : fc 85 01 68 60 48 a5 01 35
9e50 : 09 03 85 01 58 68 60 a9 fa
9e58 : 00 85 fd a9 d0 85 fe 20 45
9e60 : 43 9e d8 20 b5 9e a0 01 01
9e68 : b1 fd cd d7 9f d0 0c 88 48
9e70 : b1 fd cd d6 9f d0 04 20 3f
9e78 : 4d 9e 60 b0 fa 4c 63 9e 1f
9e80 : 20 57 9e 20 43 9e a0 02 a7
9e88 : 18 b1 fd 69 01 91 fd c8 4b
9e90 : b1 fd 69 00 91 fd 18 c8 95
9e98 : 20 4d 9e b9 00 dd 49 ff 52
9ea0 : 20 43 9e 71 fd 08 91 fd 9a
9ea8 : c8 07 f0 04 28 4c 97 9e 29
9eb0 : 28 07 fd 9e 60 18 a9 08 8d
9eb8 : 65 fd 85 fd a9 00 65 fe 6b
9ec0 : 85 fe 60 a0 00 b1 f9 20 a6
9ec8 : 16 e7 c8 ca d0 f7 60 20 ec
9ed0 : 80 9e ad da 9f 8d 02 03 da
9ed8 : ad db 9f 8d 03 03 c6 02 0a
9ee0 : 03 a9 02 8d 20 d0 60 20 34
9ee8 : 43 9e a0 00 b1 fd 99 de d2
9ef0 : 9f c8 c0 08 d0 f6 20 4d 04

```

```

9ef8 : 9e 60 20 43 9e a9 00 8d 89
9f00 : 00 d0 20 4d 9e 60 20 fd 83
9f08 : ae 20 9e b7 8a 48 20 fd cc
9f10 : ae 20 9e b7 68 a8 18 20 d9
9f18 : f0 ff 20 fd ae 4c a4 aa 05
9f20 : a9 00 8d 00 08 a9 01 85 0a
9f28 : 2b a9 08 85 2c a9 00 85 f6
9f30 : 37 a9 80 85 38 60 a9 00 3a
9f38 : 8d 00 80 a9 01 85 2b a9 57
9f40 : 80 85 2c a9 e0 85 37 a9 2d
9f48 : 9c 85 38 60 20 20 9f 20 83
9f50 : 44 a6 a2 00 b5 2b 9d e6 89
9f58 : 9f e8 e0 0e d0 f6 20 36 17
9f60 : 9f 20 44 a6 4c 74 a4 a2 35
9f68 : 00 bd e6 9f 48 b5 2b 9d 0e
9f70 : e6 9f 68 95 2b e8 e0 0e 8c
9f78 : d0 ef 60 4c 4d 53 2d 46 92
9f80 : 2e 53 2e 0d 4c 41 55 46 36
9f88 : 5a 45 49 54 2d 4d 45 53 5b
9f90 : 53 2d 53 59 53 54 45 4d 01
9f98 : 20 20 56 32 2e 30 20 c9
9fa0 : 31 31 2f 38 35 0d 28 43 20
9fa8 : 29 20 46 52 41 4e 5a 20 ed
9fb0 : 53 54 4f 49 42 45 52 0d cd
9fb8 : 11 52 55 4e 20 2d 20 4d 9b
9fc0 : 45 53 53 45 4e 0d 45 4e 2b
9fc8 : 44 20 2d 20 41 55 53 57 26
9fd0 : 45 52 54 45 4e 0d 00 00 49
9fd8 : 00 00 00 00 00 00 00 d9
9fe0 : 00 00 00 00 00 00 00 e1
9fe8 : 00 00 00 00 00 00 00 e9
9ff0 : 00 00 00 00 ff 00 ff 00 f0

```

Listing 4. Das Maschinenprogramm »LMS.PHA« bitte mit dem MSE eingeben.

zeigten Zeiten entsprechen den Laufzeiten ohne Meßprogramm. Es ist zu beachten, daß Programmschleifen innerhalb einer Basic-Zeile vom LMS nicht erkannt werden. Dies liegt an der zeilenorientierten Arbeitsweise des LMS.

```
10 GET A$: IF A$="" THEN 10
```

ergibt als Anzahl der Aufrufe der Zeile 10 die Zahl 1.

Um die exakte Aufrufanzahl zu erhalten, muß die Zeile geteilt werden.

```
10 GET A$
```

```
20 IF A$="" THEN 10
```

ergibt die genaue Anzahl der Aufrufe.

Die gleiche Vorgangsweise ist auch für Zeilen angebracht, in denen nach einem GOSUB noch weitere Statements folgen. In diesem Fall wird nämlich nach der Rückkehr vom Unterprogramm die Aufrufanzahl der Zeile um 1 erhöht. Die ausgewiesene durchschnittliche Laufzeit ist in diesem Fall mit Vorsicht zu genießen.

Im Rahmen von mehreren Messungen an einem Programm treten in der Regel unterschiedliche Laufzeiten für dieselben Statements auf. Diese zunächst etwas verblüffende Feststellung hat ihre Ursache darin, daß für die Abarbeitung eines Basic-Befehls nicht immer die gleiche Anzahl von Instruktionen durchlaufen wird. Dies ergibt sich zum Beispiel durch Bildschirm-Scrolling, Interrupt-Behandlung oder Garbage Collection. Bei der Festlegung der Meßgenauigkeit wurde diese Tatsache berücksichtigt; das heißt nicht aussagekräftige Dezimalstellen wurden weggelassen.

Das LMS funktioniert nicht mit Programmen

— die den vom LMS belegten Speicher benötigen (\$8000 bis \$9FFF und \$D000 bis \$FFFF; siehe auch Programmbeschreibung),

— die Basic-Zeilen generieren oder

— die den NMI-Timer verwenden (zum Beispiel RS232-Programme).

Programmbeschreibung

Das Laufzeit-Meß-System besteht aus dem Maschinenprogramm »LMS.PHA« und den Basic-Programmen »LMS« und »LMS.BAS.«. »LMS.PHA« ist das eigentliche Meßprogramm. Es liegt im Speicherbereich \$9CE0 bis \$9FFF und wird bei seiner Aktivierung in die Basic-Interpreterschleife eingebunden. Für die Zeitmessung wird der Timer im CIA2 benutzt. Als Zeitbasis wird der Systemtakt verwendet. Die Meßergebnisse werden im RAM ab \$D000 unter dem Betriebssystem abgelegt.

Der Ablauf einer Messung kann dem Flußdiagramm (siehe Bild 1) beziehungsweise dem Listing »LMS.ASS« entnommen werden. Die Aufbereitung und Auswertung der Meßergebnisse erfolgt im Programm »LMS.BAS«. Um ein abwechselndes Laden des in Basic geschriebenen Auswerteprogramms und des zu messenden Programms zu vermeiden, wird der Basic-Speicher derart geteilt, daß beide Programme gleichzeitig geladen sein können. LMS.BAS muß sich dabei mit weniger als 8 KByte ab Adresse \$8000 begnügen.

Die Umschaltung des Basic-Speichers erfolgt in »LMS.PHA«. Beim Aktivieren des Meßprogramms wird das zu messende Programm zur Verfügung gestellt, beim Desaktivieren wird zum Auswerteprogramm verzweigt.

Das Ladeprogramm »LMS« lädt zunächst »LMS.PHA«, damit die Maschinenroutinen zur Verfügung stehen. Danach teilt und initialisiert das Programm den Basic-Bereich. Zuletzt lädt und startet es das Auswerteprogramm.

Die Anpassung von »LMS.BAS« an unterschiedliche Drucker ist problemlos, da außer der Sekundäradresse 7 keine speziellen SteuerCodes verwendet wurden.

Für die Verwendung mit Datensette müssen die im Programmlisting LMS enthaltenen Hinweise beachtet werden.

(Mark Richters/Franz Stoiber/og)

100 REM ----- LMS-AUSWERTEPROGRAMM V2.0	<031>	380 IF A\$="D" THEN 720	<217>
101 REM ----- 11/85 FRANZ STÖIBER	<181>	390 IF A\$="H" THEN 1010	<211>
110 PRINT CHR\$(14);CHR\$(8): POKE 808,225:		400 IF A\$="E" THEN 1030	<123>
ZF=6	<072>	410 GOTO 300	<100>
120 E\$="***** ENDE *****"		420 REM ----- MESSUNG	<090>
*****	<058>	430 POKE 808,237: PRINT CHR\$(9);"CLR";:	
130 POKE 53280,15: POKE 53281,15: POKE 646		SYS 40160	<057>
,ZF: GOSUB 1440	<240>	440 REM ----- ANZEIGE AB	<208>
140 BL\$="(40SPACE)"	<040>	450 GOSUB 1930: IF F THEN 300	<252>
150 RA=53248: PU=253: L=22: CL=985.248: R=		460 GOSUB 1440: GOSUB 1480	<002>
.05	<086>	470 P=P-8: GOSUB 1130: GOTO 300	<116>
160 K0=2+8: K1=2+16: K2=2+24: K3=2+32: K4=		480 REM ----- ZURUECK	<085>
K1-1: K5=63999: K6=10	<076>	490 GOSUB 1930: IF F THEN 300	<036>
170 RD=40679: AT=40710: AB=40535: NR=40918	<128>	500 P=P-(Z+L-1)*8	<057>
180 P1=40926: P2=P1+1: P3=P1+2: P4=P1+3: P		510 IF P<RA THEN P=RA	<096>
S=P1+4: P6=P1+5: P7=P1+6: P8=P1+7	<044>	520 GOSUB 1130: GOTO 300	<157>
200 J=0: K=80: P=RA	<194>	530 REM ----- VOR	<145>
210 KZ\$="(6SPACE)R(3SPACE)R(8SPACE)MSEK		540 GOSUB 1930: IF F THEN 300	<088>
(4SPACE)MSEK/R(8SPACE)MSEK	<218>	550 GOSUB 1130: GOTO 300	<187>
220 SYS AT,6,5,"MESSUNG (3SP		560 REM ----- SUMME	<221>
ACE)M"	<103>	570 GOSUB 1930: IF F THEN 300	<118>
230 SYS AT,6,7,"ANZEIGE AB R (3SP		580 GOSUB 1440	<220>
ACE)A"	<155>	590 SYS AT,5,4,"SUMME": SYS AT,5,5,"TTTT"	<170>
240 SYS AT,6,9,"VORHERGEHEND SEITE .. (3SP		600 GOSUB 1480: GOSUB 1580	<021>
ACE)↑"	<194>	610 P=P-8: S=0	<089>
250 SYS AT,6,11,"NÄCHSTE SEITE RE		620 GOSUB 1080	<069>
TURN"	<111>	630 IF Z>ZB THEN 670	<167>
260 SYS AT,6,13,"SUMME (3S		640 GOSUB 1370	<104>
PAGE)S"	<154>	650 S=S+T: SYS AT,30,10,Z	<203>
270 SYS AT,6,15,"DRUCKEN (3S		660 GOTO 620	<152>
PAGE)D"	<235>	670 SYS AT,30,10,"(6SPACE)"	<233>
280 SYS AT,6,17,"HILFE (3S		680 T=S: T\$="": GOSUB 1400	<215>
PAGE)H"	<171>	690 SYS AT,5,17,"SUMME =" ; T\$; " MSEK"	<243>
290 SYS AT,6,19,"ENDE (3S		700 P=RA: GOTO 300	<136>
PAGE)E"	<137>	710 REM ----- DRUCKEN	<067>
300 POKE 198,0: SYS AT,0,24,"(RVSON,4SPACE		720 GOSUB 1930: IF F THEN 300	<012>
)WEITER MIT: M/A/↑/RETURN/S/D/H/E(3SPA		730 GOSUB 1440	<114>
CE,RVDF)";	<144>	740 SYS AT,5,4,"DRUCKEN": SYS AT,5,5,"TTTT	
310 POKE 2023,160: POKE 56295,ZF	<206>	TTTT	<085>
320 GET A\$: IF A\$="" THEN 320	<131>	750 GOSUB 1480: GOSUB 1580: P=P-8: S=0	<078>
330 IF A\$="M" THEN 430	<039>	760 SYS AT,5,15,"DRUCKER D. L. ?"	<216>
340 IF A\$="A" THEN 450	<059>	770 SYS AT,5,16,"WEITER MIT BELIEBIGER JAS	
350 IF A\$="↑" THEN 490	<244>	TE"	<232>
360 IF A\$=CHR\$(13) THEN 540	<156>	780 POKE 198,0: WAIT 198,1: POKE 198,0	<121>
370 IF A\$="S" THEN 570	<118>	790 OPEN 4,4,7: POKE 768,185: PRINT#4,"";:	


```

POKE 768,139
800 IF ST THEN SYS AT,8,22,"(RVSON,SPACE)
RUCKER NICHT BEREIT (SPACE,RVOFF)": CLO
SE 4: GOTO 300
810 IF K<80 THEN PL$="": GOSUB 1830: GOTO
830
820 SYS AT,0,15,BL$: SYS AT,0,16,BL$:
830 F=0: PL$=""
840 GOSUB 1080
850 IF Z>ZB THEN 960
860 GOSUB 1320
870 GOSUB 1370
880 S=S+T: T$="": B$="(12SPACE)"
890 GOSUB 1400
900 IF AZ<2 THEN 920
910 B$="": T=T/AZ: AZ=0: GOTO 890
920 PL$=PL$+" (2SPACE)"+Z$+AZ$+T$+B$
930 IF F=0 THEN F=1: GOTO 840
940 GOSUB 1830
950 GOTO 830
960 IF F THEN GOSUB 1830
970 T=S: T$="": GOSUB 1400
980 PL$="(7SPACE)SUMME ="+"T$+" MSEK": GOSU
B 1830
990 CLOSE 4: P=RA: GOTO 300
1000 REM ----- HILFE
1010 GOSUB 1440: GOTO 220
1020 REM ----- ENDE
1030 GOSUB 1440: SYS AT,5,13,"BITTE ENDE B
ESTÄTIGEN (J/N): (SPACE,RVSON,SPACE,R
VOFF)":
1040 GET A$: IF A$="" THEN 1040
1050 IF A$="J" THEN SYS AT,35,13,"J": SYS
64738
1060 SYS AT,0,13,BL$: GOTO 220
1070 REM ----- WERTE AUS RAM LESEN
1080 P=P+B: PH=INT(P/K0): PL=P-PH*K0
1090 POKE PU,PL: POKE PU+1,PH: SYS RD
1100 Z=PEEK(P1)+PEEK(P2)*K0
1110 RETURN
1120 REM ----- AUSGABE BILDSCHIRM
1130 PRINT "(CLR,RVSON,4SPACE)NR(3SPACE)BN
Z(8SPACE)MSEK(4SPACE)MSEK/BNZ(4SPACE)
"
1140 M1=0: M2=0: C1$="": C2$=""
1150 FOR ZZ=1 TO L
1160 :GOSUB 1080
1170 :IF Z=K4 THEN P=P-B: PRINT E$: GOTO
1280
1180 :GOSUB 1320
1190 :GOSUB 1370
1200 :IF T>M1 THEN M1=T: L1=ZZ: C1$="+"
1210 :T$=""
1220 :GOSUB 1400
1230 :IF AZ<2 THEN 1260
1240 :T=T/AZ: IF T>M2 THEN M2=T: L2=ZZ: C2
$="+"
1250 :AZ=0: GOTO 1220
1260 :PRINT Z$:AZ$:T$
1270 NEXT
1280 IF C1$="+" THEN SYS AT,24,L1+1,C1$:
1290 IF C2$="+" THEN SYS AT,36,L2+1,C2$:
1300 RETURN
1310 REM ----- AUFBEREITUNG NR,ANZ
1320 Z$=RIGHT$(" (5SPACE)"+STR$(Z),6)
1330 AZ=PEEK(P3)+PEEK(P4)*K0
1340 AZ$=RIGHT$(" (5SPACE)"+STR$(AZ),6)
1350 RETURN
1360 REM ----- ZEIT ERRECHNEN
1370 T=(PEEK(P5)+PEEK(P6)*K0+PEEK(P7)*K1+P
EEK(P8)*K2)/CL
1380 RETURN
1390 REM ----- AUFBEREITUNG ZEIT
1400 TR=T+R: TH=INT(TR): TL=INT((TR-TH)*K6
)
1410 T$=T$+RIGHT$(" (9SPACE)"+STR$(TH),10)+
". "+RIGHT$(STR$(TL),1)
1420 RETURN
1430 REM ----- KOPFZEILEN
1440 PRINT "(CLR,RVSON,2SPACE)LAUFZEIT(2SP
ACE)-(2SPACE)MESS(2SPACE)-(2SPACE)SYS
TEM(3SPACE)2.0(3SPACE)":
1450 PRINT "(RVSON,2SPACE)(C) 11/85 (SPACE,
SHIFT-SPACE)FRANZ STOIBER(14SPACE)"
1460 RETURN
1470 REM ----- EINGABE ZEILENNR AB

```

```

1480 RO=8: SYS AT,5,RO,"NR(2SPACE)?(2SP
ACE)": GOSUB 1650
1490 IF A$="(5SPACE)" THEN ZA=0: SYS AT,14
,RO,ZA: GOTO 1520
1500 ZA=VAL(A$): IF ZA<=K5 THEN SYS AT,0,2
2,BL$: GOTO 1520
1510 SYS AT,6,22,"(RVSON,4SPACE)ZEILENNUM
MER ZU GROSS(4SPACE,RVOFF)": GOTO 148
0
1520 ZH=INT(ZA/K0): ZL=ZA-ZH*K0: POKE NR,Z
L: POKE NR+1,ZH
1530 SYS AB: P=PEEK(PU)+PEEK(PU+1)*K0
1540 P=P-B: GOSUB 1080
1550 IF Z<K4 THEN RETURN
1560 SYS AT,6,22,"(RVSON,SPACE)ZEILENNUMME
R NICHT GEFUNDEN (SPACE,RVOFF)": GOTO
1480
1570 REM ----- EINGABE ZEILENNR BIS
1580 RO=10: SYS AT,5,RO,"BIS NR?(2SPACE)"
: GOSUB 1650
1590 IF A$="(5SPACE)" THEN ZB=K5: SYS AT,1
4,RO,ZB: RETURN
1600 ZB=VAL(A$): IF ZB>=ZA THEN 1620
1610 SYS AT,5,22,"(RVSON,2SPACE)ZEILENNR B
IS < ZEILENNR AB(2SPACE,RVOFF)": GOT
O 1580
1620 IF ZB>K5 THEN ZB=K5
1630 SYS AT,0,22,BL$: RETURN
1640 REM ----- ZAHL EINGEBEN
1650 N=1: RC=1024+14+40*RO: CO=RC+54272: P
OKE 198,0
1660 FOR I=1 TO 5: POKE CO+I,ZF: NEXT
1670 IF N>5 THEN N=5
1680 IF N<1 THEN N=1
1690 PO=RC+N: POKE PO,PEEK(PO)+128
1700 GET C$: IF C$="" THEN 1700
1710 IF C$>="0" AND C$<="9" THEN POKE PO,A
SC(C$): N=N+1: GOTO 1670
1720 IF C$=CHR$(157) THEN POKE PO,PEEK(PO)
-128: N=N-1: GOTO 1670
1730 IF C$=CHR$(29) THEN POKE PO,PEEK(PO)-
128: N=N-1: GOTO 1670
1740 IF C$=CHR$(20) THEN SYS AT,15,RO,"(5S
PACE)": GOTO 1650
1750 IF C$=CHR$(13) THEN POKE PO,PEEK(PO)-
128: GOTO 1770
1760 GOTO 1700
1770 A$=""
1780 FOR I=1 TO 5
1790 :A$=A$+CHR$(PEEK(RC+I))
1800 NEXT
1810 RETURN
1820 REM ----- ZEILE DRUCKEN
1830 IF K<68 THEN 1900
1840 IF K<73 THEN PRINT# 4,"": K=K+1: GOTO
1840
1850 J=J+1: PRINT# 4,"": PRINT# 4,""
1860 PRINT# 4,"(3SPACE)LAUFZEIT - MESS - S
YSTEM(2SPACE)2.0":
1870 PRINT# 4,"(5SPACE)(C) FRANZ STOIBER 1
1/85(6SPACE)SEITE: ";RIGHT$(STR$(J+10
0),2)
1880 PRINT# 4,"": PRINT# 4,"": PRINT# 4,KZ
$:KZ$: PRINT# 4,""
1890 K=8
1900 PRINT# 4,PL$: K=K+1
1910 RETURN
1920 REM ----- TEST AUF MESSUNG
1930 PP=P: F=0: P=RA-B: A$=""
1940 GOSUB 1080
1950 FOR I=P1 TO P8: A$=A$+CHR$(PEEK(I)):
NEXT
1960 IF A$="LMS-F.S." THEN 1990
1970 SYS AT,6,22,"(RVSON,SPACE)KEINE MESSU
NG GELAUFEN! (SPACE,RVOFF)"
1980 F=1: RETURN
1990 GOSUB 1080
2000 IF Z<K4 THEN P=PP: RETURN
2010 SYS AT,6,21,"(RVSON,SPACE)WÄHREND DE
R MESSUNG WAR "
2020 SYS AT,6,22,"(RVSON,2SPACE)KEIN PROGR
AMM GELADEN! (2SPACE,RVOFF)"
2030 GOTO 1980

```