

**Zeitschrift:** Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association suisse des électriciens, de l'Association des entreprises électriques suisses

**Herausgeber:** Schweizerischer Elektrotechnischer Verein ; Verband Schweizerischer Elektrizitätsunternehmen

**Band:** 70 (1979)

**Heft:** 11

**Artikel:** Mikrocomputergesteuerter Strom-Zeit-Messplatz

**Autor:** Spiess, W.

**DOI:** <https://doi.org/10.5169/seals-905387>

### **Nutzungsbedingungen**

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. [Siehe Rechtliche Hinweise.](#)

### **Conditions d'utilisation**

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. [Voir Informations légales.](#)

### **Terms of use**

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. [See Legal notice.](#)

**Download PDF:** 13.05.2025

**ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>**

# Mikrocomputergesteuerter Strom-Zeit-Messplatz

Von W. Spiess

621.317.31::681.322-181.48;

Der mikrocomputergesteuerte Strom-Zeit-Messplatz misst automatisch die Kennlinie der Auslösezeit in Funktion des Stromes. Er ist für die Prüfung von Überstromauslösern (Sicherungsautomaten) wie Leitungs-, Geräte- und Motorschutzschalter geeignet und führt die zeitraubenden Messungen im 24-Stunden-Betrieb vollautomatisch durch. Für den Eingabedialog wird das für Realtime-Betriebssysteme geeignete Textverarbeitungsprogramm verwendet. Es erlaubt die tabellarische Textverarbeitung für die Text- oder Zahlenverschiebung, die Text-Ein- und -Ausgabe sowie die Eingabeprüfung für Text und Gleitkommazahlen.

Le poste piloté par microprocesseur mesure automatiquement la caractéristique du temps de déclenchement en fonction du courant. Il convient pour l'essai de coupe-circuit automatiques, soit de disjoncteurs de protection de lignes, d'appareils ou de moteurs, et exécute d'une façon tout à fait automatique, 24 heures sur 24, des mesures qui exigent beaucoup de temps. Pour le dialogue d'introduction, on utilise le programme de traitement de textes qui convient pour les systèmes à temps réel, ce qui permet de traiter sous forme de tableaux le décalage de textes ou de chiffres, l'entrée et la sortie de textes, ainsi que le contrôle d'entrée des textes et des nombres à virgule flottante.

## 1. Messaufgabe

Die Kennlinienmessungen der Überstromauslöser sind sehr zeitraubend. Für einzelne Messungen ist bis zu einer Stunde notwendig. Nach jeder Messung muss eine Abkühlzeit von 30 min...1 h eingehalten werden, damit die Prüflinge und speziell das Bimetall sich wieder auf die Raumtemperatur abgekühlt haben.

Die Kennlinie nach Fig. 1 stellt die Auslösezeit eines Prüflings in Funktion des Stromes dar. Jeder Prüflingstyp hat bis zu 40 verschiedene Nennströme, die alle praktisch die gleiche Kennlinie besitzen. Daher wird der Prüfstrom auf der Abszisse nicht in Ampere, sondern in Nennstromvielfachen aufgetragen. Derartige normierte Kennlinien lassen sich wesentlich besser vergleichen.

## 2. Messmethode

Bisher berechnet man den Prüfstrom für den einzelnen Messpunkt, stellt diesen am Messplatz ein und schickt diesen Strom durch die Prüflinge. Die Auslösezeit wird für jeden Prüfling gemessen. Anschliessend wird ein neuer Strom eingestellt, und nach der Abkühlzeit wird der nächste Punkt gemessen.

Die fortschreitende Miniaturisierung macht es nun möglich, den handbetriebenen Messplatz durch einen mikrocomputergesteuerten automatischen Strom-Zeit-Messplatz zu ersetzen (Fig. 2). Dieser Computermessplatz ist als rechnergesteuerter Labormessplatz konzipiert. Alle Funktionen werden vom Rechner gesteuert. Dieser verlangt die Eingaben von der Systemkonsole (Schreibmaschine) und benötigt keine zusätzlichen Steuertasten, da er automatisch die Messung startet, wenn alle notwendigen Daten eingegeben sind.

## 3. Gerätekonfiguration

Der Computermessplatz (Fig. 3) besteht aus einem Mikrocomputersystem CMS-80, einer Systemkonsole LA 36, einem Verteileinschub mit Zusatzelektronik, einem Stabilisator für den Prüfstrom und 2 Einschaltvorrichtungen, die die Prüflinge aufnehmen.

### 3.1 Prozessgeräte

Der Prüfstromstabilisator liefert in 4 Stufen von 100 mA, 1 A, 10 A und 100 A stabilisierten Gleichstrom bei bis zu 180 V, und zwar einen lastunabhängig geregelten Gleichstrom von 10...100% des jeweiligen Bereiches.

Der Verteileinschub enthält die Meßshunts mit den Strombereichswahlschützen. Die gleichzeitig gemessenen Prüflinge

einer Gruppe sind elektrisch in Serie geschaltet. Überbrückungsschütze schliessen die nicht im Messkreis liegenden Prüflinge kurz. Mit den Anwahlschützen werden die Prüflinge mit dem Meßstromkreis verbunden.

Die Zusatzelektronik besteht im wesentlichen aus der Überbrückungselektronik. Bei der simultanen Messung wird die Spannung über den Prüflingen einzeln überwacht. Steigt sie an einem Prüfling über 30 V an, wird dieser als ausgelöst an den Rechner signalisiert. Gleichzeitig muss, infolge der Serieschaltung, der entsprechende Überbrückungsschütz betätigt werden.

Die Einschaltvorrichtungen dienen als Prüflingshalterung und schalten vor der Messung die Prüflinge ein. Als Prüflinge kommen Sicherungsautomaten in Frage, die Geräte und Leitungen vor Überströmen schützen. Der Computermessplatz ist so universell ausgelegt, dass sowohl Leitungsschutzschalter, Geräteschutzschalter, Wippschalter und auch Motorschutzschalter, je 2 Typen gleichzeitig gemessen werden können.

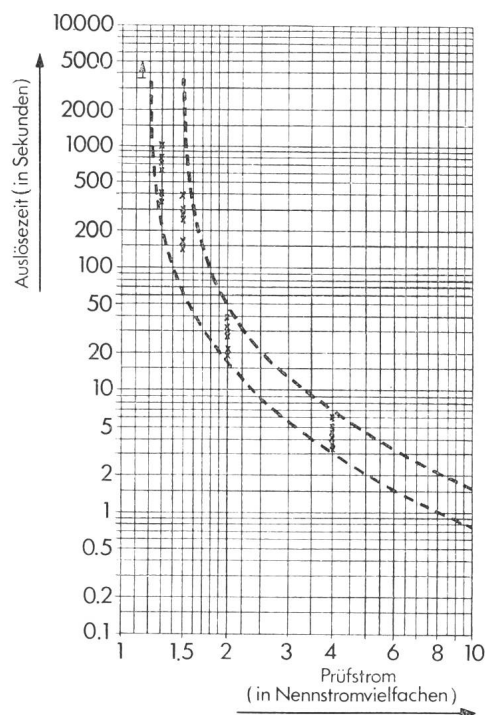


Fig. 1 Schaltkennlinie mit Streubereich

Geräteschutzschalter T 12  
Nennstrom 12 A

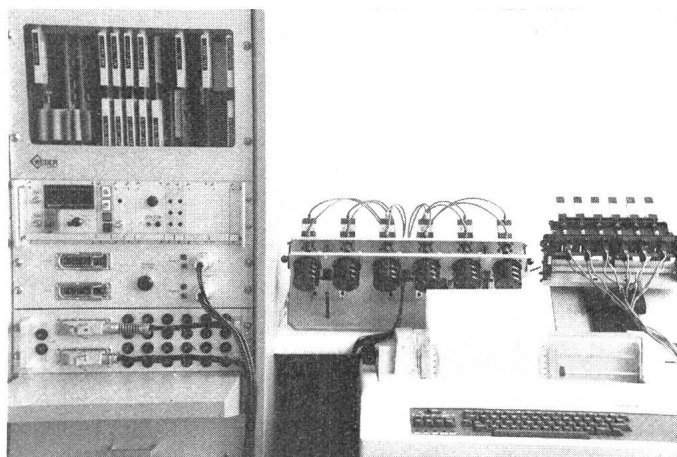


Fig. 2 Gesamtansicht des Strom-Zeit-Messplatzes

Der Geräteschrank links enthält das Mikrocomputersystem und die Prozessgeräte. Die beiden pneumatischen Einschaltvorrichtungen tragen 6 Motorschutzschalter (Mitte) und 6 Geräteschutzschalter (rechts). Davor befindet sich die Systemkonsole

### 3.2 Mikrocomputersystem

Das System besteht aus dem Mikrocomputer CMS 80 mit dem Interface, der Systemkonsole LA 36 und der Software.

Der *Mikrocomputer* basiert auf dem Motorola 8-bit-Mikroprozessor M 6800 (Fig. 4). Er ist auf einen Print inkl. Speicher und serielle Schnittstellen RS 232 aufgebaut. Mit den Speichererweiterungen ist das System auf 5 k Byte RAM (Daten) und 16 k Byte EPROM (Programmspeicher) ausgebaut.

Die *Interface-Karten* sind über den CMS 80-Bus mit dem Mikrocomputer verbunden. Das Rechner-Interface besteht aus dem Digitalteil mit 32 Optokopplereingängen und  $3 \times 16$  Relaisausgängen. Der Analogteil umfasst 16 galvanisch getrennte Spannungseingänge 0...20 V, überlastbar bis 200 V. Die Eingänge sind mit einem Vorverstärker für 0...100 mV dimensioniert.

An einer seriellen RS 232-Schnittstelle ist die Systemkonsole LA 36 angeschlossen. Sie dient zur Messdateneingabe und zum Ausdrucken der Resultate.

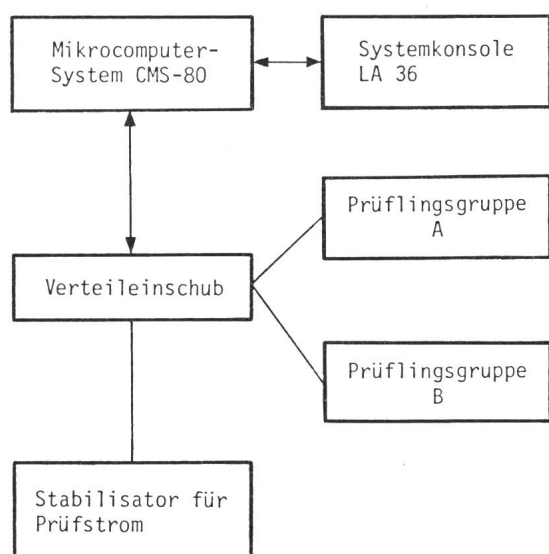


Fig. 3 Blockdiagramm des Mikrocomputer-gesteuerten Strom-Zeit-Messplatzes

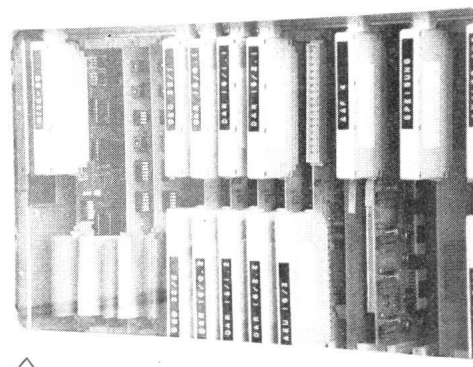


Fig. 4 Mikrocomputersystem CMS 80

Der Mikrocomputer links ist über den Prozessorbus (Flachkabel) mit den Speichererweiterungen verbunden. Die digitalen und analogen Interface-Karten schliessen rechts an

## 4. Software des Computermessplatzes

### 4.1 Eingabe-Dialogprogramm

Die Software (Programme) ist so konzipiert, dass die beiden Prüflingsgruppen vollständig unabhängig voneinander arbeiten können. Zu jeder Prüflingsgruppe gehört ein entsprechender Messbefehl-Eingabepuffer. Dies erlaubt mit nur einer Systemkonsole das unabhängige Arbeiten an zwei Prüflingsgruppen.

Der Fluss der Messdaten ist vereinfacht in Fig. 5 dargestellt. Der Mikrocomputer verlangt im Dialog mit dem Bediener die Angabe der Prüflingsgruppe. Durch Eingabe von A oder B werden die anschliessenden Eingabedaten in den Puffer A oder B gefüllt (Fig. 6).

Die Eingaben werden auf ein Minimum reduziert. Dadurch müssen die Prüflingsgrunddaten Auslöserart, Nennstrom und Prüflingsnummer nur bei der ersten Messung eingegeben werden. Die weiteren Messungen benötigen nur die Messpunkt-daten: Nennstromvielfaches (Meßstrom), maximale Prüfzeit, Abkühlzeit sowie die Anzahl Messzyklen.

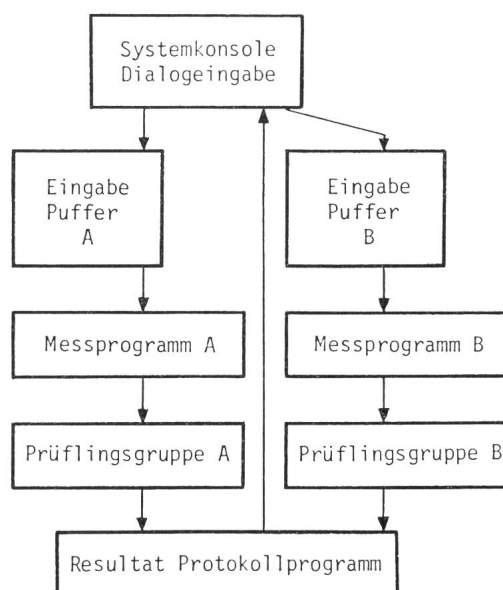


Fig. 5 Messdatenflussdiagramm

Diese Daten werden nach der positiven Beantwortung der Abfrage «SPEICHERN?» als Messdatenblock ans Ende des entsprechenden Eingabepuffers A oder B geschrieben.

Der jeweils erste eingegebene Messdatenblock wird, sobald die vorherige Messung beendet ist, vom Messprogramm übernommen und im Eingabepuffer gelöscht. Dieser ist so dimensioniert, dass die vollständige Prüfung einer Prüflingsgruppe darin gespeichert werden kann.

#### 4.2 Messprogramm

Das Messprogramm führt mit den erhaltenen Daten die gewünschte Art und Anzahl Messungen durch. Aus dem Nennstrom und den Nennstromvielfachen wird der Prüfstrom berechnet. Die gewünschten Prüflinge werden in den Messkreis geschaltet. Anschliessend wird der Prüfstromstabilisator eingeschaltet und der Meßstrom auf den Sollwert hochgeregelt. Vom Beginn des Stromflusses bis zum Auslösen des Prüflings wird dessen Auslösezeit gemessen.

Alle Prüflinge einer Gruppe sind bei kleinem Strom in Serie geschaltet (Simultanmessung). Darum muss jeder ausgelöste Prüfling sofort kurzgeschlossen werden, damit die anderen Prüflinge weiter vom Strom durchflossen werden. Bei grossen Strömen bzw. sehr kurzen Auslösezeiten werden die Prüflinge einzeln geprüft. Nach der Messung einer Prüflingsgruppe werden die Resultate ausgedruckt (Fig. 7).

```

PRUEFLINGSGRUPPE : A
AUSLOESERART      : T      NENNSTROM      : 1,0 A
PRUEFLING NR.     : A1-A6
NENNSTROMVIELFACHES : 1,1 THAX          : 60 M
ABKUEHLZEIT       : 1 S  ANZ. MESSZYKLEN : 1

SPEICHERN ? J
PRUEFLINGSGRUPPE : B
AUSLOESERART      : T      NENNSTROM      : 0,5 A
PRUEFLING NR.     : B1-B6
NENNSTROMVIELFACHES : 2,0 THAX          : 50 S
ABKUEHLZEIT       : 1 S  ANZ. MESSZYKLEN : 1

SPEICHERN ? J
PRUEFLINGSGRUPPE :

```

Fig. 6 Beispiel Eingabedialog

Der Computer-Messplatz benötigt nur die kurzen Eingaben nach dem Doppelpunkt. Nach jeder richtigen Eingabe verlangt er die nächste Eingabe. Sobald ein Block eingegeben ist, wird die 1. Messung gestartet

```

MESSUNG : 1 THERMOAUSLOESER IN = 1,000 A I = 1,10*IN
DATUM 26. 3.79 ZEIT 0. 6 ZYKLEN : 1
A1 I = 1,12 *IN T = 00 H 00 M 55 S 35 C F = 1,9 W
A2 I = 1,12 *IN T = 00 H 00 M 42 S 27 C F = 1,7 W
A3 I = 1,12 *IN T = 00 H 00 M 45 S 09 C F = 1,7 W
A4 I = 2,33 *IN T = 00 H 00 M 00 S 08 C F = 1,0 W
A5 I = 1,12 *IN T = 00 H 00 M 52 S 55 C F = 1,8 W
A6 I = 1,11 *IN T = 00 H 00 M 57 S 72 C F = 1,7 W

MESSUNG : 2 THERMOAUSLOESER IN = 0,500 A I = 2,00*IN
DATUM 26. 3.79 ZEIT 8. 8 ZYKLEN : 1
B1 I = 2,04 *IN T = 00 H 00 M 34 S 21 C F = 3,1 W
B2 I = 2,04 *IN T = 00 H 00 M 15 S 85 C F = 3,1 W
B3 I = 2,04 *IN T = 00 H 00 M 12 S 29 C F = 3,1 W
B4 I = 2,03 *IN T = 00 H 00 M 03 S 24 C F = 3,0 W
B5 I = 1,97 *IN T = 00 H 00 M 02 S 59 C F = 3,0 W
B6 I = 2,03 *IN T = 00 H 00 M 08 S 21 C F = 3,0 W

```

Fig. 7 Beispiel Messresultate

Die Messresultate enthalten im Kopf die Hauptdaten sowie die Zeitangabe. Zu jedem Prüfling A1...A6, B1...B6 gehört anschliessend eine Resultatzeile. Sie enthält den gemessenen Strom (I) in Nennstromvielfachen, die Auslösezeit (T) in h, min, s und cs sowie die Verlustleistungen (E)

```

00213 0015 C0022 > LDX $INITIA : INITIALISIER TEXTABELLENADRESSE
00214 0016 00000 > JSR TEXMOV

```

Fig. 8 Aufruf des Textverarbeitungsprogrammes

Durch die beiden Befehle LDX Lade Index-register und JSR Jump to Subroutine wird das Programm aufgerufen. Damit wird dann die ganze Texttabelle abgearbeitet

```

00222          : INITIALISIER TEXTABELLE
00223          : 1. BLOCK U
00224 0022 > INITIA EQU $
00225 0022 0036 > WORD INIDAA : SOURCE
00226 0024 0000 > WORD INFA : DESTINATION
00227 0026 13   > BYTE 19 : LENGTH
00228 0027 00   > BYTE 0 : CODE 0= TEXTUMLADEN
00229          :
00230          : 2. BLOCK U
00230 TEXMOV     H6600 ASH V3.3 DIRPRO PAGE 2

00231 0028 0034 > WORD INIDAB
00232 002A 0160 > WORD INFA
00233 002C 0400 > BYTE 4,0
00234          : 3. BLOCK U
00235 002E 003C > WORD INIDAT
00236 0030 0164 > WORD INFB+4
00237 0032 0F   > BYTE 15 : LENGTH
00238 0033 60   > BYTE 60H : CODE AD= TEXTUMLADEN + ENDE

```

Fig. 9 Texttabelle für die tabellarische Programmierung

Die Texttabelle besteht aus mehreren gleichlangen Blöcken (Beispiel 3 Blöcke). Der erste Block mit der Anfangsadresse «INITIA» enthält die Source-Adresse «INIDAA» (2 Byte) und die Destination-Adresse «INFA», die Length (Länge) 19 Byte und den Code 0 = Text umladen. Dieser Block bewirkt, dass 19 Zeichen (Byte) des bei «INIDAA» beginnenden Textes in den Datenspeicher «INFA» umgeladen werden

### 5. Tabellarische Programmierung für Mikrocomputer

Die stürmische Entwicklung der Mikrocomputer-Hardware hat bereits sehr preisgünstige Mikrocomputer auf den Markt gebracht. Leider hat die Softwareentwicklung bisher nicht immer Schritt gehalten.

Das *Textverarbeitungsprogramm* zeigt, wie mit wenig Aufwand an Speicher auch für kleine Mikrocomputersysteme ohne Massenspeicher, wie z.B. Floppydisc, trotzdem die Vorteile der einfachen tabellarischen Programmierung angewendet werden.

Dieses Programm dient zur einfachen Programmierung bei der Softwareerstellung auf einem Softwareentwicklungssystem. Die in Tabellenform programmierten Textverarbeitungen werden mit den anderen Programmen zusammen assembliert, d.h. in Objektcode übersetzt. Anschliessend werden die einzelnen übersetzten Programme durch den Linker zusammengesetzt. Diese Programme werden dann in den Programmspeicher eingebrannt. Der Systembenutzer erhält so preisgünstige Programme, die auf einfache Art und Weise abgeändert werden können. Es ist bisher nicht vorgesehen, dass der Anwender diese Programme direkt ändern kann. Dies wäre aber im Sinne einer Weiterentwicklung ohne weiteres denkbar.

Das Textverarbeitungsprogramm besteht aus vier Teilen. Der erste Teil ist das Programm selbst, der zweite der Aufruf (Fig. 8); der dritte Teil umfasst die Texttabelle, die das ganze Programm steuert. Der vierte Teil besteht aus den einzelnen Textstücken. Diese Teile werden bei der Programmentwicklung zusammengesetzt und ergeben zusammen mit der restlichen Anwender-Software das ganze Softwaresystem.

Der Softwareaufwand für Dialogprogramme wird durch das einfache Textverarbeitungsprogramm wesentlich reduziert. Dieses kann direkt die folgenden Funktionen durchführen:

1. Text im Speicher umladen
2. Text-Input
3. Text-Output
4. Zahlen von ASCII in Binär wandeln (ASCII amerikanischer genormter Zeichen-Code)
5. Zahlen von Binär in ASCII-Code wandeln
6. Text prüfen
7. Zahlen prüfen
8. Gleitkomma prüfen
9. Kurzzahlen mit Komma und Nullen ergänzen inkl. Einheiten schieben

Diese Funktionen können für einen Programmaufruf in beliebiger Reihenfolge wiederholt werden.

Das Programm wird durch die *Texttabelle* gesteuert. Diese hat eine Filestruktur und besteht aus einer beliebigen Anzahl von Texttabellenblöcken. Die Anfangsadresse des ersten Blockes wird dem Textverarbeitungsprogramm im Indexregister übergeben. Der letzte Block muss immer die Endemarke enthalten.

Jeder Texttabellenblock (Fig. 9) ist 6 Byte lang und besteht aus Sourceadresse (2 Byte), Destinationadresse (2 Byte), Length sowie Steuercode.

Die Sourceadresse gibt an, woher der Text oder die Zahl kommt, und die Destinationadresse, wohin er gespeichert wird. Die «Length» = Länge (0...256 Byte oder Zeichen) gibt an, wie lang der zu bearbeitende Text oder die Zahl ist. Der Steuercode bestimmt die gewünschte Funktion und enthält die Endemarke der Texttabelle.

Die Texte werden direkt in ASCII-Strings definiert und durch den Assembler in den entsprechenden Zahlencode übersetzt (Fig. 10).

Das Textverarbeitungsprogramm (TEXMOV) wird als Subroutine aufgerufen. Mit der Anfangsadresse (INITTA) im Indexregister (X) wird die als Blockfile aufgebaute Texttabelle komplett abgearbeitet. Der letzte Texttabellenblock enthält im

---

00242	0034	>	INITAB	EGU	\$	
00243	0034	01640165>		WORD	INFB+4.INFB+5	
00244	0036	>	INITAA	EGU	\$	
00245	0036	0004	>	WORD	INFA+4	: ANFANGSPONTER
00246	003A	0005	>	WORD	INFA+5	: END PONTIER -4
00247	003C	>	INITAT	EGU	\$	
00248	003C	05		BYTE	5	: LAENGE
00249	003D	454E4445		ASCII	"ENDE"	
00250	0041	00000000		WORD	0.0.0.0	
00250	0045	00000000				
00250	0047	0000				

---

**Fig. 10** Texte

Diese Texte werden als Binäre Zahlen oder als ASCII-Zeichen umgeladen

Code die Endemarke. Anschliessend wird im Hauptprogramm wieder weitergefahren.

Das Textverarbeitungsprogramm TEXMOV hat gegenüber den höheren Programmiersprachen (FORTRAN, BASIC) den Vorteil, dass nicht ein sehr grosser Speicherteil durch die notwendigen Sprachmodule belegt wird. Es kann in Multitask-Betriebssystemen verwendet werden. Die kurzen Input- und Output-Makros werden für das Betriebssystem entsprechend angepasst. Dadurch läuft das Programm TEXMOV sowohl auf dem Tektronix-Entwicklungssystem als auch auf dem CMS-80-Multitask-Realtime-Betriebssystem.

#### Adresse des Autors

W. Spiess, El.-Ing. ETH, Weber AG, Compac, 6020 Emmenbrücke.