

	<b>Seite</b>
<b>Hinweise zur Inbetriebnahme des Programms KML_EDIT Version 3.0 28. September 1987</b>	<b>2</b>
<b>Ergänzungen zur Bedienung des Programms KML_EDIT Version 3.0 28. September 1987</b>	<b>4</b>
<b>Hinweise zur Bedienung des Programms KML_EDIT Version 2.1 21. September 1987</b>	<b>6</b>
<b>Programmierung der FPS 3</b>	
<b>Operationen, Operanden und Parameter</b>	<b>13</b>
<b>Alphabetische Liste der Operationen</b>	<b>18</b>
<b>Zulässigkeit der Kombinationen von Operationen</b>	<b>21</b>
<b>Alphabetische Liste der Operanden</b>	<b>23</b>
<b>Ausgang</b>	<b>24</b>
<b>Ausgangswort</b>	<b>25</b>
<b>Eingang</b>	<b>26</b>
<b>Eingangswort</b>	<b>27</b>
<b>Konstante</b>	<b>28</b>
<b>Konstante im Wortformat</b>	<b>29</b>
<b>Lokale Marke</b>	<b>30</b>
<b>Merker</b>	<b>32</b>
<b>Merkerwort</b>	<b>33</b>
<b>Programmbaustein</b>	<b>34</b>
<b>Zähler</b>	<b>36</b>
<b>Zeitglieder</b>	<b>38</b>

## **Hinweise zur Inbetriebnahme des Programms KML-EDIT**

### **Version 3.0 28. September 1987**

1. Nach dem Start des PCs auf Laufwerk A umschalten,  
(Eingabe: "A: <RETURN>").
2. Legen Sie eine DOS-Systemdiskette in Laufwerk A ein. Auf der Diskette muss sich eine DOS-Version 3.10 oder höher befinden.
3. Erstellen Sie mit dem DOS-Systemprogramm FORMAT.COM eine Startdiskette mit Betriebssystem. Dazu legen Sie eine neue Diskette in Laufwerk B ein und rufen das Programm auf der Diskette in Laufwerk A wie folgt auf:  
"FORMAT B:/S <RETURN>".  
Fragen zur Handhabung der DOS-Systemprogramme und -befehle siehe das DOS-Handbuch.
4. Ist das Formatieren beendet, so sollte die neue Diskette neben den Betriebssystemdateien das Programm COMMAND.COM enthalten.
5. Von der DOS-Systemdiskette kopieren Sie mit dem DOS-Befehl COPY die Programme KEYBGR.COM, ANSI.SYS, MODE.COM und TIMER.COM.

Dazu geben Sie ein:           "COPY KEYBGR.COM B: <RETURN>",  
  "COPY ANSI.SYS B: <RETURN>",  
  "COPY MODE.COM B: <RETURN>",  
  und       "COPY TIMER.COM B: <RETURN>".

6. Kopieren Sie auf die so vorbereitete Arbeitsdiskette die Programmdateien der gelieferten Programmdiskette. Dazu legen Sie die Programmdiskette in Laufwerk A ein und geben folgende DOS-Befehle ein:

"COPY KML\_EDIT.EXE B: <RETURN>"  
"COPY CONFIG.SYS B: <RETURN>"  
"COPY AUTOEXEC.BAT B: <RETURN>".

7. Überprüfen Sie den Inhalt der Arbeitsdiskette nun mit dem DOS-Befehl DIR:

"DIR B: <RETURN>".

Das Inhaltsverzeichnis der Arbeitsdiskette erscheint nun auf dem Bildschirm.  
Folgende Dateien müssen aufgelistet werden:

```
COMMAND.COM  
KEYBGR.COM  
ANSI.SYS  
MODE.COM  
TIMER.COM  
KML_EDIT.EXE  
CONFIG.SYS  
AUTOEXEC.BAT
```

8. Ihre Arbeitsdiskette ist nun fertig. Sie wird beim Neustart des PC anstelle der DOS-Systemdiskette verwendet. Die Diskette in das Laufwerk A eingelegen und den PC neu starten. Sie befinden sich dann direkt im Programm KML\_EDIT.
9. Verwenden Sie für Ihre eigenen Programme nicht die Arbeitsdiskette, sondern eine Kopie!
10. Entfernen Sie niemals den Schreibschutz vom Original der Programmdiskette und bewahren Sie das Original an einem sicheren Ort auf!

## **Ergänzungen zur Bedienung des Programms** **KML\_EDIT Version 3.0 28. September 1987**

1. Bei den Punkten

- |    |                         |
|----|-------------------------|
| 4  | (Datei drucken),        |
| 7  | (Datei übersetzen),     |
| 8  | (Datei senden zur FPS), |
| 10 | (Datei bearbeiten)      |

kann durch Betätigen der Taste F2 das Inhaltsverzeichnis des aktuellen Laufwerks angezeigt werden. Dabei können Sie auswählen:

- |                    |   |
|--------------------|---|
| → CTRL PgUp        | bringt Sie zum Anfang des Verzeichnisses  |
| → CTRL PgDn        | bringt Sie ans Ende des Verzeichnisses  |
| → PgUp             | bringt Sie eine Seite (11 Einträge) zurück  |
| → PgDn             | bringt Sie eine Seite (11 Einträge) vor   |
| → Pfeil nach oben  | bringt Sie zum vorhergehenden Eintrag   |
| → Pfeil nach unten | bringt Sie zum nächsten Eintrag   |
| → RETURN           | übernimmt den Eintrag als aktuellen Namen<br>und führt die gewählte Funktion aus (4, 7, 8, 10). |

2. Bei jeder Auswahl "(J,N)" wird der voreingestellte Wert, den Sie durch Betätigen jeder beliebigen Taste wählen können, angezeigt; wollen Sie den jeweils anderen Wert eingeben, genügt der Anfangsbuchstabe.

**Beispiel:** Im Hauptmenü drücken Sie "Esc", um das Programm zu beenden. Angezeigt wird "Nein" als voreingestellter Wert. Wollen Sie das Programm tatsächlich beenden, müssen Sie ein "J" eingeben. Bei jeder anderen Taste (auch "Esc") sind Sie wieder im Hauptmenü oder der ausgewählten Funktion (F1, F2, usw.).

3. Gleiches gilt für die Auswahl des aktuellen Laufwerks (F3). Dort wird das zur Zeit gewählte Laufwerk auch als voreingestellter Wert angezeigt.

4. Tritt beim Übersetzen (F7 oder F8) ein Fehler auf, so wird der Fehler angezeigt und Sie können mit einem Tastendruck zu der Stelle im Programm springen, an welcher der Fehler aufgetreten ist.

Dabei erscheint die entsprechende Fehlermeldung in der letzten Zeile des Bildschirms solange, bis Sie eine beliebige Taste drücken.

Solange sich die fehlerhafte Zeile im Programm an der gleichen Stelle befindet und die fehlerhafte Zeile selbst nicht geändert wurde, kann der Fehler durch Betätigen der Taste F10 erneut angezeigt werden, und der Cursor befindet sich wieder an der fehlerhaften Stelle.

## **Hinweise zur Bedienung des Programms**

### **KML\_EDIT Version 2.1 11. September 1987**

1. Das Programm wird durch Autostart mit der Datei AUTOEXEC.BAT oder durch Eingabe des Programmnamens "KML\_EDIT" in der DOS-Befehlszeile aufgerufen.
2. Es erscheint das Hauptmenü. Hier können Sie unter den verschiedenen Funktionen auswählen, indem Sie die angegebene Funktionstaste drücken.
3. So wird ein Programm zweckmäßig erstellt:
  - a) Vorüberlegungen aufschreiben. Je sorgfältiger Sie hier vorgehen, desto schneller kommen Sie zu einem lauffähigen Programm.
  - b) Programm erstellen, Taste F10.  
In Abständen von ca. 5 Minuten sichern (F9), verwendete Operanden aufschreiben (z. B. eine Liste aller Ein- und Ausgänge anfertigen). Die Schutzschaltungen einer Maschine nie von der Funktion der FPS abhängig machen.
  - c) Programm übersetzen, Taste F7.  
Tritt beim Übersetzen ein Fehler auf, befinden Sie sich automatisch wieder im Erstellmodus (wie bei F10) und die Stelle des ersten aufgetretenen Fehlers wird angezeigt.
  - d) Wenn das Programm fehlerfrei ist, den Programmcode zur FPS senden, Taste F8.

Steuern Sie nie direkt die fertige Maschine, sondern immer zunächst einen Testaufbau!

- e) Zur Fehlersuche den Programmtext ausdrucken (F4)
  - f) Läuft der Testaufbau zufriedenstellend, Programm dokumentieren und mit Datum und Versionsnummer versehen.
  - g) Von der Diskette mit dem erstellten Programm eine Kopie anfertigen.
4. Durch Betätigen von F1 werden Datum und Uhrzeit (Systemzeit des PCs) für etwa 5 Sekunden angezeigt.
  5. Durch Betätigen von F2 können Sie sich das Inhaltsverzeichnis des aktuellen Laufwerkes ansehen. Sie können sich alle Dateien oder nur bestimmte ansehen (z. B. alle Dateien mit der Erweiterung FPS: "\*.FPS").  
DOS-Unterverzeichnisse werden nicht angezeigt (siehe DOS-Handbuch).
  6. Mit F3 können Sie das aktuelle Laufwerk wechseln, auf das alle Angaben bezogen werden. Üblich ist A, B oder C im Fall einer Festplatte. Der DOS-Pfadname kann jedoch nicht gewechselt werden (siehe DOS-Handbuch).
  7. Erwartet das Programm von Ihnen eine Eingabe, so blinkt der Cursor. Schließen Sie Ihre Eingabe mit <RETURN> ab. Das gilt nicht für den Punkt F3: 'aktuelles Laufwerk wechseln', wo Ihre Eingabe direkt übernommen wird.
  8. Bei den Punkten F4, F7, ... F10 wird die Eingabe eines Dateinamens erwartet. Ein Dateiname besteht aus bis zu 8 Zeichen, gefolgt von einem Punkt und drei weiteren Zeichen, der Erweiterung.

Beispiel: "DREHZAHN.FPS".

Wenn Sie einen Dateinamen ohne eine Erweiterung eingeben, wird automatisch ".FPS" hinzugefügt.

9. Sicherheitsabfragen "(J, N)" können ausschließlich mit 'J' oder 'N' beantwortet werden. Wird eine andere Taste gedrückt, geht das Programm von dem jeweils voreingestellten Wert (z. B. bei "ESC Programm beenden" von "N") aus.
10. Wird der Punkt F4 gewählt, wird eine Datei gedruckt. Der Drucker muss an die Parallelschnittstelle LPT 1 angeschlossen sein. Wenn Sie alle auf dem Bildschirm darstellbaren Sonderzeichen drucken wollen, muss der Drucker IBM-kompatibel sein.

Verwenden Sie also in Ihren Dateien keine Zeichen, die Ihr Drucker nicht verarbeiten kann!

11. Mit dem Punkt F8 können Sie ihr Programm zur FPS übertragen, wobei die FPS mit dem mitgelieferten Kabel an die serielle Schnittstelle COM 1 angeschlossen sein muss. Die Schnittstellenparameter für die serielle Schnittstelle werden in der Datei AUTOEXEC.BAT mit dem MODE-Befehl eingestellt:

"MODE COM1:24,N,8,1,-".

Einzelheiten siehe DOS-Handbuch.

Stellen Sie unbedingt sicher, dass die serielle Schnittstelle IBM-kompatibel ist, denn sonst funktioniert die Datenübertragung zur FPS nicht!



12. Der Punkt "Sichern" (F9) kommt erst während des Bearbeitens von Dateien mit Punkt F10 richtig zur Geltung:

Sie können während der Arbeit am Programmtext durch Betätigung der Taste F9 den gesamten Text unter dem aktuellen Namen speichern. Ein gleichnamiges Programm (z. B. eine Datei der letzten Sicherung) wird unter gleichem Namen, jedoch mit der Erweiterung ".BAK" gespeichert, eine bestehende Datei gleichen Namens mit der Erweiterung ".BAK" wird überschrieben.

Wollen Sie die Datei auf mehreren Laufwerken sichern, ändern Sie mit F3 das aktuelle Laufwerk und speichern die Datei dann mit F9 auf dem neu gewählten Laufwerk. Dabei können Sie auch einen anderen Dateinamen angeben.

13. Zu jeder Zeit und an jeder Stelle können Sie den Programmablauf mit der Taste 'ESC' (bei deutschen Tastaturen mit der Taste ABBR) unterbrechen, wenn dies in der untersten Bildschirmzeile angezeigt wird. Bei manchen Programmteilen erfolgt ein Abbruch jedoch nicht sofort.

14. Geben Sie dem PC etwas Zeit, verschiedene Programmteile zu beenden, bevor sie ihn neu starten, weil Sie an einen Programmabsturz glauben.

Es ist uns bereits während der Entwicklungsphase des Programms nicht mehr gelungen, es zum Absturz zu bringen.

Die Punkte F3, F4, F7 und F8 können am Anfang eine etwas längere Bearbeitungszeit benötigen.

Es dauert immer etwas länger, wenn sie einen nicht vorhandenen Drucker, eine nicht vorhandene FPS oder ein Diskettenlaufwerk ohne eingelegte Diskette ansprechen wollen.

15. Ist Ihr Drucker angeschlossen und das erste Zeichen wurde übertragen, so wartet das Programm immer auf die Bereitmeldung des Druckers. Wenn es also einmal nicht weitergehen sollte, prüfen Sie bitte die Bereitschaft Ihres Druckers.
16. Bearbeitung des Programms mit F10:
- a) Sie können neue Dateien erstellen oder vorhandene bearbeiten. Wird die von Ihnen angegebene Datei auf dem aktuellen Laufwerk nicht gefunden, so wird eine neue Datei erstellt.
  - b) Wollen Sie eine Datei bearbeiten, die sich noch im Speicher befindet, so werden Sie gefragt, ob die Datei neu geladen werden soll oder nicht. Die Voreinstellung ist "NEIN".
  - c) Als "Cursor" bezeichnet man das blinkende Feld auf dem Bildschirm.
  - d) Müssen zwei Tasten gleichzeitig gedrückt werden, so hält man die erste (SHIFT oder CTRL) fest und drückt anschließend die zweite Taste.
  - e) Alle Tasten haben eine Wiederholfunktion (außer INS)
  - f) Folgende Tasten haben im Erstellmodus (F10) eine Sonderfunktion:

<b>ESC</b>	beendet Erstellmodus.
<b>F1</b>	zeigt Datum und Uhrzeit an.
<b>F9</b>	sichert den gesamten Speicherinhalt.
<b>DEL</b>	löscht das Zeichen unter dem Cursor.

<b>INS</b>	schaltet den Einfügemodus ein bzw. aus. Ist er eingeschaltet, so wird das nächste Zeichen an der Cursorposition eingefügt, ist er ausgeschaltet, wird das Zeichen unter dem Cursor überschrieben.
<b>BS</b>	(← Backspace) löscht das Zeichen links vom Cursor.  Die 4 Pfeiltasten bewegen den Cursor über den Bildschirm (nur bis zur letzten Zeile).
<b>HOME</b>	bringt den Cursor an den Zeilenanfang.
<b>END</b>	bringt den Cursor ans Textende der Zeile.
<b>CTRL HOME</b>	bringt den Cursor an den Anfang der Seite.
<b>CTRL END</b>	bringt den Cursor ans Ende der Seite.
<b>TAB</b>	bewegt den Cursor vier Zeichen nach rechts.
<b>SHIFT TAB</b>	bewegt den Cursor vier Zeichen nach links.
<b>RETURN</b>	bringt den Cursor an den Anfang der folgenden Zeile. Befindet sich der Cursor in der letzten Zeile, so wird eine neue Zeile hinzugefügt (nur bis zur max. Zeilenzahl).
<b>PGUP</b>	blättert um eine Seite zurück.
<b>PGDN</b>	blättert um eine Seite vor.
<b>CTRL PGUP</b>	bringt den Cursor an den Anfang des Textes.
<b>CTRL PGDN</b>	bringt den Cursor zum Ende des Textes.
<b>CTRL N</b>	fügt an der Cursorposition eine Zeile ein.
<b>CTRL Y</b>	löscht die Zeile, in der sich der Cursor befindet.

- g) Alle anderen Zeichen-Tasten werden in den Speicher übernommen.
  - h) Bearbeiten Sie nie Dateien mit den Erweiterungen  
COM  
EXE  
SYS (außer die Datei CONFIG.SYS)
  - i) Das Bearbeiten von reservierten Dateien (z. B. "AUX") wird verhindert (siehe dazu das DOS-Handbuch). Die Programmdateien mit dem Namen "KML\_EDIT" können ebenfalls nicht bearbeitet werden.
  - k) Sie können Ihre Programme auch mit jedem anderen Textverarbeitungsprogramm (z. B. mit WORDSTAR) erstellen, wenn Sie sich an die hier getroffenen Vereinbarungen halten.
17. Nach Erstellung des Programms können Sie den Erstellmodus mit ESC verlassen. Sie werden dann gefragt, unter welchem Dateinamen Sie das Programm sichern wollen.

Wenn alles geklappt hat, befinden Sie sich anschließend wieder im Hauptmenü, ansonsten erhalten Sie eine Fehlermeldung.

## Operationen, Operanden und Parameter

1. Ein Befehl ist normalerweise wie folgt aufgebaut:

Operation	Operand	Parameter 1	Parameter 2
-----------	---------	-------------	-------------

Beispiele:

L	E	0	4
UN	A	0	6
=	A	0	9

Es gibt Befehle, die nur einen Parameter haben (z. B. **L MW 34**; **BA P 4**), und solche, die weder Operand noch Parameter haben (z. B. **NOP**; **BE**; **PE**).

2. Es gibt zwei Arten von Befehlen:

- a) Bit-Befehle (z. B. **ON M 4**)
- b) Wort-Befehle (z. B. **ADD KWD 1000**)

3. Eine Folge von Befehlen wird durch einen Befehl, der nur am Anfang einer Folge zulässig ist, eingeleitet, z. B. **L**. Sie wird durch einen Befehl, der nur am Ende einer Folge zulässig ist, abgeschlossen, z. B. **=**. Dazwischen können Befehle stehen, die nur innerhalb einer Folge von Befehlen zulässig sind, z. B. **U** oder **0**.

4. Mehrere Zuweisungen können einfach hintereinander geschrieben werden.

Beispiel:

=	A	0	0
S	A	0	5
ZV	Z	35	

5. Kombinationen von Bit- und Wortbefehlen innerhalb einer Folge sind nicht erlaubt, jedoch kann von einer Befehlsart zur anderen gewechselt werden, wenn eine Befehlsfolge beendet ist.
6. Beeinflussung von Verknüpfungsergebnis (**VKE**) sowie Arbeitsregister (**A**) und Statusregister (**M G P O**):

- a) Einige Bit-Befehle beeinflussen das VKE. Diese Befehle sind im Anhang aufgeführt. Dort ist dies in Spalte A vermerkt (z. B. **U, O**).  
Bei Bit-Befehlen steht das Ergebnis der letzten Operation im VKE.

Bei den Bit-Befehlen gibt es die beiden Zustände:

VKE = 1 (Bedingung erfüllt)

und

VKE = 0 (Bedingung nicht erfüllt).

Arbeits- und Statusregister der Wortverarbeitung werden durch die Bit-Verarbeitung verändert!

- b) Einige Wortbefehle beeinflussen A und M G P O. Diese Befehle sind im Anhang aufgeführt (z. B. **SUB**). Bei Wortbefehlen steht das Ergebnis der letzten Operation (eine Zahl) in A.

Das VKE der Bitverarbeitung wird durch die Wortverarbeitung verändert!

- c) In den Spalten A und M G P O im Anhang bedeuten:

x wird beeinflusst  
– wird nicht beeinflusst  
0 wird auf 0 gesetzt (also beeinflusst)

m Minus (kleiner als)  
G Gleich (gleich)  
P Plus (größer als oder gleich)  
O Bereichsüberschreitung (Overflow)

7. Zu den angegebenen Parametern:

Die theoretisch möglichen Grenzen sind angegeben, die praktisch zulässigen Grenzen hängen davon ab, aus welchen Komponenten Ihre FPS aufgebaut ist.

8. Programmbausteine:

Sie beginnen mit "**P n**:" (n bezeichnet die Nummer des Bausteins), sie enden mit "**BE**" und müssen mit einem der BA-Befehle aufgerufen werden (z. B. "**BA P 4**"). Der Doppelpunkt darf nur dort stehen, wo der Programmbaustein definiert wird, jedoch nie dort, wo er aufgerufen wird (also nicht so: "**BA P 4**").

Programmbausteine müssen vor dem Hauptprogramm definiert werden. Wenn es einen **P 0** gibt, so muss er an erster Stelle definiert werden.

In **P 0** werden alle Befehle eingetragen, die zu Beginn der Programmbearbeitung einmalig ausgeführt werden sollen (z. B. Startwerte für die Timer laden usw.). **P 0** muss nicht im Programm aufgerufen werden, er wird automatisch einmal bearbeitet, er kann jedoch zusätzlich wie jeder andere Programmbaustein aufgerufen werden und wird dann erneut bearbeitet.

Der Aufruf eines Programmbausteins ist nur im Hauptprogramm erlaubt.

9. Lokale Marken:

Sie dienen als Ziele für Sprungoperationen.

Eine lokale Marke besteht aus einer beliebigen Zeichenfolge, die mit einem Buchstaben beginnen muss (jedoch nie mit "P") und mit einem Doppelpunkt endet (z. B. "DREHZAHL:"). Reservierte Wörter (alle Operationen und Operanden) sind nicht erlaubt.

Im Sprungbefehl wird der Doppelpunkt weggelassen (z. B. "SPB DREHZAHL").

10. Sprünge müssen folgende Bedingungen erfüllen:

Immer nur vorwärts gerichtet, nie rückwärts, immer nur an den Anfang einer Befehlsfolge, nie in die Mitte oder ans Ende, nie derart, dass ein Programmbaustein verlassen wird.

11. Das Hauptprogramm gilt als eigener Programmbaustein, dem jedoch keine eigene Nummer zugewiesen wird. Wenn der letzte Programmbaustein mit BE abgeschlossen ist, und es folgt keine weitere Bausteindefinition, so beginnt das Hauptprogramm automatisch. Eine Kennzeichnung ist nicht erforderlich. Das Hauptprogramm wird von der FPS zyklisch bearbeitet. Ist "PE" abgearbeitet, wird wieder von vorne begonnen.
12. Nach einem Befehl kann ein Kommentar stehen. Dieser wird durch ein Semikolon ';' vom Befehl getrennt. Alles, was in einer Zeile nach dem ersten Semikolon folgt, wird als Kommentar angesehen und vom Übersetzer ignoriert.

Beispiel:

```
;L E 0 0 ;DER BEFEHL WIRD SOMIT WIRKUNGSLOS
```

13. Leerzeichen dürfen überall stehen. Durch sie werden einzelne Teile von Befehlen voneinander getrennt.



14. Eine Zeile sieht so aus:

- a) leer
- b) Definition eines Programmbausteins oder einer lokalen Marke (nie zusammen)
- c) Befehl
- d) b), dann c)
- e) wie a) . . . d), dann Kommentar

Beispiele:

```
...
b)  STUECK:
c, e)      L   Z   4   ; HOLT STUECKZAHL VON BAND 4
c, e)      GL  KWD 45  ; SCHON 45 STUECK?
a)
c, e)      BAG P   6   ; JA, BAND ANHALTEN
d, e)  PACKEN: L   MW  3   ; VERPACKUNG SCHLIESSEN
...
```

Es ist besser, b) und c) zu trennen, als d) zu verwenden. So sind Änderungen leichter durchzuführen.

Sparen Sie NIE mit Kommentaren!

15. Es wird nicht zwischen Groß- und Kleinschreibung unterschieden.

Beispiel: "marke1" und "MARKE1" sind identisch!

## Alphabetische Liste der Operationen

<b>A MGPO</b>	<b>Operation</b>	<b>Bedeutung</b>
x xxxx	<b>ADD</b>	Addiere Operand zum Arbeitsregister
- -----	<b>BA</b>	Baustein aufruf unbedingt, Operand = Marke des Bausteinanfangs
- -----	<b>BAB</b>	Baustein aufruf bedingt, der Aufruf erfolgt, wenn letztes VKE = 1
- -----	<b>BAG</b>	Baustein aufruf bedingt, der Aufruf erfolgt, wenn Ergebnis der letzten Operation = Null
- -----	<b>BAM</b>	Baustein aufruf bedingt, der Aufruf erfolgt, wenn Ergebnis der letzten Operation kleiner Null
- -----	<b>BAO</b>	Baustein aufruf bedingt, der Aufruf erfolgt, wenn Ergebnis der letzten Operation nicht mehr innerhalb des erlaubten Zahlenbereiches liegt
- -----	<b>BAP</b>	Baustein aufruf bedingt, der Aufruf erfolgt, wenn Ergebnis der letzten Operation größer oder gleich Null
- -----	<b>BE</b>	Baustein ende unbedingt
- -----	<b>BEB</b>	Baustein ende bedingt, der Rücksprung erfolgt, wenn letztes VKE = 1
- -----	<b>BEG</b>	Baustein ende bedingt, der Rücksprung erfolgt, wenn Ergebnis der letzten Operation = Null
- -----	<b>BEM</b>	Baustein ende bedingt, der Rücksprung erfolgt, wenn Ergebnis der letzten Operation kleiner Null
- -----	<b>BEO</b>	Baustein ende bedingt, der Rücksprung erfolgt, wenn Ergebnis der letzten Operation nicht mehr innerhalb des erlaubten Zahlenbereiches liegt
- -----	<b>BEP</b>	Baustein ende bedingt, der Rücksprung erfolgt, wenn Ergebnis der letzten Operation größer oder gleich Null

Alphabetische Liste der Operationen (Fortsetzung)

<b>A MGPO</b>	<b>Operation</b>	<b>B e d e u t u n g</b>
x xxxx	<b>DIV</b>	Dividiere das Arbeitsregister ganzzahlig durch den Operanden
- xxx0	<b>GL</b>	Vergleiche das Arbeitsregister mit dem Operanden (entspricht einer Subtraktion des Operanden vom Arbeitsregister, ohne dieses zu verändern)
x	<b>L</b>	Lade das Arbeitsregister mit dem Operanden (Bit)
x xxxx0	<b>L</b>	Lade das Arbeitsregister mit dem Operanden (Wort)
x	<b>LN</b>	Lade das Arbeitsregister mit dem negierten Operanden (Bit)
x xxxx	<b>MOD</b>	Rest der Ganzzahldivision
x xxxx	<b>MUL</b>	Multipliziere das Arbeitsregister mit dem Operanden
- ----	<b>NOP</b>	Nulloperation
x	<b>O</b>	Logische ODER-Verknüpfung des Arbeitsregisters mit dem Operanden
x	<b>ON</b>	Logische ODER-Verknüpfung des Arbeitsregisters mit dem Operanden, Ergebnis ist negiert
- ----	<b>PE</b>	Programmende
-	<b>R</b>	RESET (Rücksetzen des Operanden, wenn letztes VKE = 1)
-	<b>S</b>	SET (Setzen/Starten des Operanden, wenn letztes VKE = 1)
- ----	<b>SP</b>	Sprung vorwärts unbedingt, Operand = Marke der nächsten abzuarbeitenden Befehlsfolge

Alphabetische Liste der Operationen (Fortsetzung)

<b>A MGPO</b>	<b>Operation</b>	<b>B e d e u t u n g</b>
–	<b>SPB</b>	Sprung vorwärts, bedingt. Der Sprung erfolgt, wenn letztes VKE = 1
– -----	<b>SPG</b>	Sprung vorwärts, bedingt. Der Sprung erfolgt, wenn Ergebnis der letzten Operation = Null
– -----	<b>SPM</b>	Sprung vorwärts, bedingt. Der Sprung erfolgt, wenn Ergebnis der letzten Operation kleiner Null
– -----	<b>SPO</b>	Sprung vorwärts, bedingt. Der Sprung erfolgt, wenn Ergebnis der letzten Operation nicht mehr innerhalb des erlaubten Zahlenbereiches liegt
– -----	<b>SPP</b>	Sprung vorwärts, bedingt. Der Sprung erfolgt, wenn Ergebnis der letzten Operation größer oder gleich Null
x xxxxx	<b>SUB</b>	Subtrahiere den Operanden vom Arbeitsregister
x	<b>U</b>	Logische UND-Verknüpfung des Arbeitsregisters mit dem Operanden
x	<b>UN</b>	Logische UND-Verknüpfung des Arbeitsregisters mit dem Operanden, Ergebnis ist negiert
x	<b>XO</b>	Logische EXCLUSIV-ODER-Verknüpfung des Arbeitsregisters mit dem Operanden (Antivalenz)
x	<b>XON</b>	Logische EXCLUSIV-ODER-Verknüpfung des Arbeitsregisters mit dem Operanden, Ergebnis ist negiert (Äquivalenz)
–	<b>ZR</b>	Erniedrige den Operanden um 1 (zähle rückwärts). Es wird dekrementiert, wenn letztes VKE = 1
–	<b>ZV</b>	Erhöhe den Operanden um 1 (zähle vorwärts). Es wird inkrementiert, wenn letztes VKE = 1
– -----	<b>=</b>	Weise dem Operanden den Inhalt des Arbeitsregisters zu (bei Bit-Befehlen das VKE)

## Zulässigkeit der Kombination von Operationen

### 1. Bitverarbeitung

#### a) Nur als **erste** Anweisung einer Folge

<b>BA</b>	Bausteinanruf, unbedingt
<b>BE</b>	Bausteinende, unbedingt
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>LN</b>	Laden (Bereitstellen des ersten (negierten) Operanden)
<b>PE</b>	Programmende
<b>SP</b>	Sprung vorwärts, unbedingt

#### b) Nur zwischen erster und letzter Anweisung

<b>O</b>	ODER
<b>ON</b>	ODER NICHT
<b>U</b>	UND
<b>UN</b>	UND NICHT
<b>XO</b>	EXCLUSIV ODER
<b>XON</b>	EXCLUSIV ODER NICHT

#### c) Nur als letzte Anweisung einer Folge

<b>BAB</b>	Bausteinanruf, wenn letztes VKE = 1
<b>BEB</b>	Bausteinende, wenn letztes VKE = 1
<b>R</b>	RESET
<b>S</b>	SET
<b>SPB</b>	Sprung vorwärts, wenn letztes VKE = 1
<b>ZR</b>	Zählen rückwärts
<b>ZV</b>	Zählen vorwärts
<b>=</b>	Zuweisung

#### d) An jeder Stelle

<b>NOP</b>	Nulloperation
------------	---------------

## 2. Wortverarbeitung

a) Nur als erste Anweisung einer Folge

<b>BA</b>	Bausteinanruf, unbedingt
<b>BE</b>	Bausteinende, unbedingt
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>PE</b>	Programmende
<b>SP</b>	Sprung vorwärts, unbedingt

b) Nur zwischen erster und letzter Anweisung

<b>ADD</b>	Addieren
<b>DIV</b>	Ganzzahldivision
<b>GL</b>	Vergleichen
<b>MOD</b>	Rest der Ganzzahldivision
<b>MUL</b>	Multiplizieren
<b>SUB</b>	Subtrahieren

c) Nur als letzte Anweisung einer Folge

<b>BAG</b>	Bausteinanruf, wenn Ergebnis gleich Null
<b>BAM</b>	Bausteinanruf, wenn Ergebnis kleiner Null (Minus)
<b>BAO</b>	Bausteinanruf, wenn Ergebnis überläuft (Overflow)
<b>BAP</b>	Bausteinanruf, wenn Ergebnis größer/= Null (Plus)

<b>BEG</b>	Bausteinende, wenn Ergebnis gleich Null
<b>BEM</b>	Bausteinende, wenn Ergebnis kleiner Null (Minus)
<b>BEO</b>	Bausteinende, wenn Ergebnis überläuft (Overflow)
<b>BEP</b>	Bausteinende, wenn Ergebnis größer/= Null (Plus)

<b>SPG</b>	Sprung vorwärts, wenn Ergebnis gleich Null
<b>SPM</b>	Sprung vorwärts, wenn Ergebnis kleiner Null (Minus)
<b>SPO</b>	Sprung vorwärts, wenn Ergebnis überläuft (Overflow)
<b>SPP</b>	Sprung vorwärts, wenn Ergebnis größer/= Null (Plus)

**=** Zuweisung

d) An jeder Stelle

**NOP** Nulloperation

## Alphabetische Liste der Operanden

	<b>Bedeutung</b>
<b>A</b>	Ausgang
<b>AW</b>	Ausgangswort
<b>E</b>	Eingang
<b>EW</b>	Eingangswort
<b>K</b>	Konstante
<b>KW</b>	Konstante im Wortformat
<b>LM</b>	Lokale Marke (Text beliebig)
<b>M</b>	Merker
<b>MW</b>	Merkerwort
<b>P</b>	Programmbaustein
<b>TA</b>	Zeitglied mit Ausschaltverzögerung
<b>TE</b>	Zeitglied mit Einschaltverzögerung
<b>TI</b>	Zeitglied als Impulszähler
<b>Z</b>	Zähler

## Ausgang

Funktion: 1 Hardware – Bit, vom Anwender nutzbar

Befehlsformat:

Eingabe Anwender					Bedeutung
A	0	0 ... A	0	255	Ausgänge Karte 0
A	1	0 ... A	1	255	Ausgänge Karte 1
...					
A	255	0 ... A	255	255	Ausgänge Karte 255

Zulässige Operationen:

Befehl	Bedeutung
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>LN</b>	Laden, Operand wird negiert
<b>O</b>	ODER
<b>ON</b>	ODER NICHT
<b>R</b>	RESET
<b>S</b>	SET
<b>U</b>	UND
<b>UN</b>	UND NICHT
<b>XO</b>	EXCLUSIV ODER
<b>XON</b>	EXCLUSIV ODER NICHT
<b>=</b>	Zuweisung



## Ausgangswort

Funktion: Hardware – Analogausgang, vom Anwender nutzbar

Befehlsformat:

Eingabe Anwender				Bedeutung
AW 0	0 ...	AW 0	255	D/A-Kanäle Karte 0
AW 1	0 ...	AW 1	255	D/A-Kanäle Karte 1
...				
AW 255	0 ...	AW 255	255	D/A-Kanäle Karte 255

Zulässige Operationen:

Befehl	Bedeutung
<b>ADD</b>	Addieren
<b>DIV</b>	Ganzzahldivision
<b>GL</b>	Vergleichen
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>MOD</b>	Rest der Ganzzahldivision
<b>MUL</b>	Multiplizieren
<b>SUB</b>	Subtrahieren
<b>=</b>	Zuweisung

## Eingang

Funktion: 1 Hardware – Bit, vom Anwender nutzbar

Befehlsformat:

Eingabe Anwender					Bedeutung
E	0	0 ... E	0	255	Eingänge Karte 0
E	1	0 ... E	1	255	Eingänge Karte 1
...					
E	255	0 ... E	255	255	Eingänge Karte 255

Zulässige Operationen:

Befehl	Bedeutung
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>LN</b>	Laden, Operand wird negiert
<b>O</b>	ODER
<b>ON</b>	ODER NICHT
<b>U</b>	UND
<b>UN</b>	UND NICHT
<b>XO</b>	EXCLUSIV ODER
<b>XON</b>	EXCLUSIV ODER NICHT

## Eingangswort

Funktion: Hardware – Analogeingang, vom Anwender nutzbar

Befehlsformat:

Eingabe Anwender				Bedeutung
EW 0	0 ...	EW 0	255	A/D-Kanäle Karte 0
EW 1	0 ...	EW 1	255	A/D-Kanäle Karte 1
...				
EW 255	0 ...	EW 255	255	A/D-Kanäle Karte 255

Zulässige Operationen:

Befehl	Bedeutung
<b>ADD</b>	Addieren
<b>DIV</b>	Ganzzahldivision
<b>GL</b>	Vergleichen
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>MOD</b>	Rest der Ganzzahldivision
<b>MUL</b>	Multiplizieren
<b>SUB</b>	Subtrahieren

## Konstante

Funktion: 1 Bit, hardwareunabhängig

Befehlsformat:

Eingabe Anwender Bedeutung

K 0 Konstante 0 (1 Bit)

K 1 Konstante 1 (1 Bit)

Zulässige Operationen:

Befehl	Bedeutung
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>LN</b>	Laden, Operand wird negiert
<b>O</b>	ODER
<b>ON</b>	ODER NICHT
<b>U</b>	UND
<b>UN</b>	UND NICHT
<b>XO</b>	EXCLUSIV ODER
<b>XON</b>	EXCLUSIV ODER NICHT

## Konstante im Wortformat

Funktion: 32 Bit, hardwareunabhängig

Besonderheiten: - Eingabe von negativen Zahlen möglich (nur KWD)

Befehlsformat:  
Eingabe Anwender

**KWD** (Wert als Dezimalzahl)

**KWH** (Wert als Sedezimalzahl)

Zulässige Operationen:

Befehl	Bedeutung
<b>ADD</b>	Addieren
<b>DIV</b>	Ganzzahldivision
<b>GL</b>	Vergleichen
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>MOD</b>	Rest der Ganzzahldivision
<b>MUL</b>	Multiplizieren
<b>SUB</b>	Subtrahieren

## Lokale Marke

Funktion: kennzeichnet das Ziel von Vorwärtssprüngen

- Besonderheit:
- muss mit einem Buchstaben beginnen (nicht mit P)
  - darf kein reserviertes Wort sein
  - es darf nur an den Anfang einer Folge gesprungen werden
  - LM n steht stellvertretend für alle Marken

Befehlsformat Wortoperationen:

Eingabe Anwender

LM 0

...

LM 65535

Zulässige Wortoperationen:

Befehl	Bedeutung
<b>SP</b>	Sprung vorwärts, unbedingt
<b>SPG</b>	Sprung vorwärts, wenn Ergebnis = Null
<b>SPM</b>	Sprung vorwärts, wenn Ergebnis kleiner Null
<b>SPO</b>	Sprung vorwärts, wenn Ergebnis überläuft
<b>SPP</b>	Sprung vorwärts, wenn Ergebnis größer/= Null

## Lokale Marke

Befehlsformat Bitoperationen:

Eingabe Anwender

LM 0

⋮  
LM 65535

Zulässige Bitoperationen:

Befehl	Bedeutung
<b>SP</b>	Sprung vorwärts, unbedingt
<b>SPB</b>	Sprung vorwärts, wenn letztes VKE = 1

## Merker

Funktion: 1 Bit Zwischenspeicher, vom Anwender nutzbar

Besonderheiten:

- normale Merker 0 ... 255
- Die Merker 256 ... 511 sind besondere Merker, in denen der Zustand der Steuerung gespeichert ist
- Die Merker 512 ... 767 sind Textmerker, bei deren Aufruf ein Text in der Anzeige dargestellt wird

Befehlsformat: Eingabe Anwender

```
M    0
...
M    65535
```

Zulässige Operationen:

Befehl	Bedeutung
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>LN</b>	Laden, Operand wird negiert
<b>O</b>	ODER
<b>ON</b>	ODER NICHT
<b>R</b>	RESET
<b>S</b>	SET
<b>U</b>	UND
<b>UN</b>	UND NICHT
<b>XO</b>	EXCLUSIV ODER
<b>XON</b>	EXCLUSIV ODER NICHT
<b>=</b>	Zuweisung



## Merkerwort

Funktion: 32 Bit Zwischenspeicher, vom Anwender nutzbar

Befehlsformat:

Eingabe Anwender

MW 0

...  
MW 65535

Zulässige Operationen:

Befehl	Bedeutung
<b>ADD</b>	Addieren
<b>DIV</b>	Ganzzahldivision
<b>GL</b>	Vergleichen
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>MOD</b>	Rest der Ganzzahldivision
<b>MUL</b>	Multiplizieren
<b>SUB</b>	Subtrahieren
<b>=</b>	Zuweisung

## Programmbaustein

Funktion: vom Anwender frei definierbare Anzahl  
von Steuerungsanweisungen

Besonderheit: - Kennzeichen des Beginns eines Programmbausteins  
ist die (reservierte!) Marke zu Beginn '**P n**:',  
die als 'global' aufgefasst wird

- ein Baustein muss mit einem der BE-Befehle enden,  
da er mit einem der BA-Befehle aufgerufen  
werden muss

Befehlsformat Wortoperationen:

```
Eingabe Anwender
P      0
...
P      65535
```

Zulässige Wortoperationen:

Befehl	Bedeutung
<b>BA</b>	Bausteinanruf, unbedingt
<b>BAG</b>	Bausteinanruf, wenn Ergebnis = Null
<b>BAM</b>	Bausteinanruf, wenn Ergebnis kleiner Null
<b>BAO</b>	Bausteinanruf, wenn Ergebnis überläuft
<b>BAP</b>	Bausteinanruf, wenn Ergebnis größer/= Null

## Programmbaustein

Befehlsformat Bitoperationen:

Eingabe Anwender

P 0

...

P 65535

Zulässige Bitoperationen:

Befehl	Bedeutung
<b>BA</b>	Bausteinanruf, unbedingt
<b>BAB</b>	Bausteinanruf, wenn letztes VKE = 1

## Zähler

Funktion: 32 Bit - Softwarezähler

Besonderheiten:

- kann mit "=" auf Anfangswert gesetzt werden, Wert muss im Akku stehen
- alle Wortoperationen mit dem Zählerstand
- Zählen (vor-/rückwärts) sowie SET / RESET: Bitoperation!

Befehlsformat Wortoperationen:

Eingabe Anwender

Z 0

· · ·  
Z 65535

Zulässige Wortoperationen:

Befehl	Bedeutung
<b>ADD</b>	Addieren
<b>DIV</b>	Ganzzahldivision
<b>GL</b>	Vergleichen
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>MOD</b>	Rest der Ganzzahldivision
<b>MUL</b>	Multiplizieren
<b>SUB</b>	Subtrahieren
<b>=</b>	Zuweisung

## Zähler

Befehlsformat Bitoperationen:

Eingabe Anwender

Z 0

...

Z 65535

Zulässige Bitoperationen:

Befehl	Bedeutung
<b>R</b>	Rücksetzen (auf 0)
<b>S</b>	Setzen (auf Anfangswert)
<b>ZR</b>	Zählen rückwärts
<b>ZV</b>	Zählen vorwärts



## Zeitglieder

Befehlsformat Bitoperationen:

Eingabe Anwender

Tx	0
...	
Tx	65535
TI	0 0
...	
TI	255 255

Zulässige Bitoperationen:

Befehl	Bedeutung
<b>L</b>	Laden (Bereitstellen des ersten Operanden)
<b>LN</b>	Laden, Operand wird negiert
<b>O</b>	ODER
<b>ON</b>	ODER NICHT
<b>R</b>	RESET
<b>S</b>	SET
<b>U</b>	UND
<b>UN</b>	UND NICHT
<b>XO</b>	EXCLUSIV ODER
<b>XON</b>	EXCLUSIV ODER NICHT