

DESY-Bibliothek

8. JAN. 1969 ✓

Interner Bericht  
DESY F58-69/1  
Januar 1969

DESY - PDP5/PDP8/PDP8I - Programmiersystem

EDITORPROGRAMM SUPEDT

F. Akolk, G. Hochweller

# INHALTSVERZEICHNIS

=====

1	Einführung
2	Laden des Programms
3	Text- und Bandformate
4	SUPEDT-Kommandos
4.1	Kommando-mode
4.1.1	Display-Steuerung
4.1.2	Textpuffer-Kommandos
4.1.3	DEctape-Kommandos
4.1.4	DUMP- und LIST-Kommandos
4.1.5	Tabulator
4.1.6	Systemanrufe
4.2	Korrekturen
4.3	Übergang in den Text-mode
4.4	Text-mode
4.5	IBM-Programm 'SERVICE'
5	Anhang

1           EINFÜHRUNG  
              =====

Das Programm SUPEDT (SUPervisor EDiTor) dient zur Erzeugung und Verarbeitung von beliebigen Texten auf PDP-5/PDP-8/PDP-8I - Maschinen. Es setzt eine Konfiguration mit mindestens 8k-Speicherplätzen, DECTapes und Display (DSP30N oder DSP34B) voraus.

Grundsätzlich läßt sich mit SUPEDT jegliche Art von Texten bearbeiten; da es sich bei den meisten in Frage kommenden Texten jedoch um Programme handelt (PAL-4, IBM-Assembler, FORTRAN, etc.), enthält SUPEDT eine Reihe von Routinen, die den Umgang mit solchen 'formatierten' Texten erleichtern.

2           LADEN DES PROGRAMMS  
              =====

- a. Band "SYSTEM-Fxx" auf Einheit 8 montieren
- b. Einheit 8 auf "REMOTE" und "WRITE LOCK" (Kippschalter an der Bandeinheit) schalten
- c. Falls die Nummer 8 noch für eine andere Bandeinheit gewählt ist, diese ausschalten oder für diese Einheit eine andere Nummer wählen (Drehschalter an der Bandeinheit)
- d. "DATA FIELD" und "INST FIELD" (Schalter an der PDP) beide auf 0 einstellen
- e. "SWITCH REGISTER" (Schalter an der PDP) auf 7600 einstellen
- f. "LOAD ADD" (Taste an der PDP) drücken
- g. "START" (Taste an der PDP) drücken; das Band auf Einheit 8 läuft jetzt an den Anfang
- h. Nachdem der Fernschreiber ein Geräusch von sich gegeben hat, SUPEDT schreiben (auf dem Fernschreiber)
- i. "RETURN" (Taste am Fernschreiber) drücken

Jetzt wird das Programm SUPEDT vom Systemband in den Speicher geladen und gestartet.

3 TEXT- UND BANDFORMATE  
=====

Normalerweise werden mit Hilfe von SUPEDT erzeugte Texte auf DECTapes gespeichert. Solche texttragenden Bänder werden im folgenden als EDITORBÄNDER bezeichnet; jedes Editorband enthält 92 EDITORBLÖCKE (0...91). Ein Editorblock ist die Einheit, in der ein Editorband mit SUPEDT beschrieben oder gelesen wird. SUPEDT enthält hierfür einen Textspeicher, der genau einen Editorblock faßt (3600(oktal) = 1920(dezimal) Speicherplätze). Jeder Editorblock wiederum besteht aus 20 (oktal) BANDBLÖCKEN zu je 200 (oktal) Worten. Zur Abspeicherung des Editorblockes werden jedoch nur 17 (oktal) Bandblöcke benötigt; der erste Bandblock eines jeden Editorblockes wird nicht benutzt. Er enthält normalerweise den Namen des Editorbandes (siehe Hilfsprogramm 'COPY').

	N	T	N	T		N	T	N	T	frei
BB	0	1-17	20	21-37	- - -	2640	2641-2657	2660	2661-2677	2700-2701
EB		0		1		90		91		

Abb. 1: Aufbau eines Editorbandes  
 BB = Bandblöcke (oktal)  
 EB = Editorblöcke (dezimal)  
 N = enthält Namen des Editorbandes  
 T = enthält Text

Sämtliche mit Hilfe von SUPEDT erzeugten Texte werden auf folgende Weise abgespeichert:  
 Die über den Fernschreiber eingegebenen Zeichen werden zunächst in einem Zeilenpuffer gesammelt; erst nach Beendigung der Zeile (Taste RETURN) wird der Inhalt des Zeilenpuffers an den jeweiligen Inhalt des Textspeichers angehängt. Während der Eingabe einer Zeile bestehen Korrekturmöglichkeiten; zur Kontrolle wird der Inhalt des Zeilenpuffers auf dem Display angezeigt. Die maximale Länge einer Zeile beträgt 126 Zeichen.

Die Speicherung der einzelnen Zeichen erfolgt im getrimmten ASCII-Code (siehe Anhang); jeweils zwei Zeichen benötigen einen Speicherplatz. Im ersten Wort einer jeden Zeile steht die Anzahl der Zeichen, aus denen sich die Zeile zusammensetzt. Besteht die Zeile aus einer ungeradzahlig Anzahl von Zeichen, so wird das letzte Wort mit Nullen aufgefüllt.

In dem ersten Wort des Textspeichers (und somit auch jeden Editorblocks) steht die jeweilige Anzahl der Zeilen in diesem Block; anschließend folgen die einzelnen Zeilen im oben beschriebenen Format.  
 Der Textspeicher beginnt auf Adresse 0 in bank 1.

Wort	oktaler Inhalt	Bedeutung
1	0013	11 (=Zeichenzahl)
2	0411	DI
3	0523	ES
4	0540	E
5	3205	ZE
6	1114	IL
7	0500	L

Abb.2: Beispiel für die Speicherplatzbelegung einer Textzeile  
Zeilentext: DIESE ZEILE

Außer dem Textspeicher und dem Zeilenpuffer enthält SUPEDT noch einige Hilfsspeicher, die zur Umsortierung, Korrektur und zur Ausgabe von Texten benötigt werden.

4 SUPEDT-KOMMANDOS  
=====

Das Programm SUPEDT kann sich in zwei verschiedenen Zuständen befinden, dem KOMMANDO- oder dem TEXT-mode. Im Kommando-mode werden alle eingegebenen Zeichen als Kommandos betrachtet, während sie im Text-mode als zum Text gehörig angesehen und somit in den entsprechenden Puffern abgespeichert werden.

Folgende Konventionen sollen im folgenden gelten:

Ein + hinter dem Kommando bedeutet, daß das Kommando mit "RETURN" abgeschlossen werden muß.

Falls die Zeichen m bzw. n Zeilennummern bedeuten, kann man folgende Abkürzungen verwenden, die dem Benutzer das Feststellen der genauen Zeilennummer ersparen:

- . für die Nummer der obersten auf dem Display angezeigten Zeile
- / für die Nummer der letzten Zeile des Textspeichers
- ; als Abkürzung für eine Kombination m,n - das Kommando bezieht sich dann auf den gesamten auf dem Display angezeigten Bereich

Einfache Additionen oder Subtraktionen in Kommandos sind möglich; will man z.B. mit der Kombination m,n die letzten 10 Zeilen des Textspeichers ansprechen, darf /-9,/ geschrieben werden.

Die Nummerierung der Zeilen des Textspeichers erfolgt immer von 1 an fortlaufend bis zur letzten Zeile. Sie wird automatisch korrigiert, wenn Zeilen gelöscht bzw. eingeschoben werden.

4.1 KOMMANDO-MODE  
-----

Beim Übergang in den Kommando-mode erscheint folgendes Bild auf dem Display:

KOMMANDO:

. = x / = y LB = z

Dabei bedeuten x die Nummer der obersten 'angezeigten' Zeile des Textspeichers, y die Nummer der letzten Zeile des Textspeichers, z die Nummer des zuletzt eingelesenen Editorblockes. Ebenfalls im Kommando-mode befindet man sich, wenn auf dem Display der Inhalt des Textspeichers angezeigt wird.

4.1.1 DISPLAY-STEUERUNG

- - - - -
- nA+ Die Zeile n des Textspeichers wird auf dem Display angezeigt.
  - m,nA+ Die Zeilen m bis n (einschließlich) werden auf dem Display angezeigt.
  - SPACE Das angezeigte Stück des Textspeichers wird um eine Zeile nach oben verschoben.
  - nSPACE Das angezeigte Stück des Textspeichers wird um n Zeilen nach oben verschoben.
  - B Das angezeigte Stück des Textspeichers wird um eine Zeile nach unten verschoben.
  - nB Das angezeigte Stück des Textspeichers wird um n Zeilen nach unten verschoben.
  - CTRL+F Das angezeigte Stück des Textspeichers wird um eine Zeile nach unten verlängert.
  - CTRL+B Das angezeigte Stück des Textspeichers wird um eine Zeile verkürzt.
  - n0+ Der Inhalt des Speicherplatzes n wird oktal, dezimal und alphabetisch angezeigt.  
Soll die angegebene Adresse n oktal angenommen werden, so muß 0n statt n geschrieben werden.
  - m,n0+ Der Inhalt der Speicherplätze m bis n (einschließlich) wird oktal, dezimal und alphabetisch angezeigt. Sollen die angegebenen Adressen m oder n oktal angenommen werden, so muß 0m bzw. 0n statt m bzw. n geschrieben werden.
  - CTRL+T Die PAL-4 - Formatierung des angezeigten Textes wird ein- bzw. ausgeschaltet.

4.1.2 TEXTPUFFER-KOMMANDOS

- K+ Der gesamte Textspeicher wird gelöscht.
- nD+ Die Zeile n des Textspeichers wird gelöscht.
- m,nD+ Die Zeilen m bis n (einschließlich) des Textspeichers werden gelöscht.
- G+ Der Inhalt des Textspeichers wird in den Hilfsspeicher übernommen. Der Textspeicher wird dabei nicht gelöscht.
- nG+ Die Zeile n wird in den Hilfsspeicher übernommen. Sie wird dabei im Textspeicher nicht gelöscht.
- m,nG+ die Zeilen m bis n (einschließlich) des Textspeichers werden in den Hilfsspeicher übernommen. Sie werden dabei im Textspeicher nicht gelöscht.
- H+ Der Inhalt des Textspeichers wird in den Hilfsspeicher übernommen. Der Textspeicher wird dabei gelöscht.
- nH+ Die Zeile n des Textspeichers wird in den Hilfsspeicher übernommen und dabei im Textspeicher gelöscht.
- m,nH+ Die Zeilen m bis n (einschließlich) des Textspeichers werden in den Hilfsspeicher übernommen und dabei im Textspeicher gelöscht.
- U+ Der Inhalt des Hilfsspeichers wird an den Inhalt des Textspeichers angehängt. Der Inhalt des Hilfsspeichers bleibt dabei erhalten.
- nU+ Der Inhalt des Hilfsspeichers wird vor Zeile n in den Textspeicher eingesetzt. Der Inhalt des Hilfsspeichers bleibt dabei erhalten.
- m,nU+ Die Zeilen m bis n (einschließlich) des Textspeichers werden gelöscht; an ihre Stelle wird der Inhalt des Hilfsspeichers eingesetzt. Der Inhalt des Hilfsspeichers bleibt dabei erhalten. m und n dürfen gleich sein.

4.1.3 DECTAPE-KOMMANDOS

- 
- m,nE+ Der Editorblock n des DECTapes auf Bandeinheit m wird in den Textspeicher gelesen.
  - m,nX+ Der Inhalt des Textspeichers wird auf den Editorblock n des DECTapes auf Bandeinheit m geschrieben. Der Textspeicher wird dabei nicht gelöscht.
  - m,nR+ Von dem DECTape auf Bandeinheit 7 werden n Bandblöcke (zu je 128 Worten) ab Bandblock m in den Textspeicher gelesen. Sollen die Zahlen m oder n oktal angenommen werden, so muß Om bzw. On statt m bzw. n geschrieben werden.
  - m,nW+ n Bandblöcke (zu je 128 Worten) ab Bandblock m des DECTapes auf Bandeinheit 7 werden mit dem Inhalt des Textspeichers beschrieben. Sollen die Zahlen m oder n oktal angenommen werden, so muß Om bzw. On statt m bzw. n geschrieben werden.

#### 4.1.4 DUMP- UND LIST-KOMMANDOS

bg bedeutet bei den DUMP-Kommandos eine Gruppe von Editorblöcken. Wenn bg beispielsweise 4,7-9,15,5-6 ist, so werden folgende Editorblöcke ausgegeben (in dieser Reihenfolge):

4, 7, 8, 9, 15, 5, 6

Die Ausgabe auf der Schreibmaschine erfolgt direkt vom Editorband über einen eigenen kleinen Hilfsspeicher; der Inhalt des Textspeichers wird dabei nicht verändert. Neben der Ausgabe auf der Schreibmaschine kann mit dem Textspeicher ganz normal weitergearbeitet werden.

PDUMP:bg+ Die durch die Blocknummern-Liste angegebenen Editorblöcke des Editorbandes auf Bandeinheit 5 werden im PAL4-Format auf der Schreibmaschine ausgegeben.  
Schreibmaschineneinstellung:  
linker Rand: 10  
Tabulator: 16, 25, 45, 55, 65, 75

ODUMP:bg+ Die durch die Blocknummern-Liste angegebenen Editorblöcke des Editorbandes auf Bandeinheit 5 werden unformatiert auf der Schreibmaschine ausgegeben.

XDUMP:bg+ Die durch die Blocknummern-Liste angegebenen Editorblöcke des Editorbandes auf Bandeinheit 5 werden als normaler Text auf der Schreibmaschine ausgegeben. Auf eine Seite kommen dabei 66 Zeilen (einzeilige Ausgabe). Zur Darstellung von Schreibmaschinen-Sonderzeichen im ASCII-Code siehe Anhang.

XDUMP1:bg+ Die durch die Blocknummern-Liste angegebenen Editorblöcke des Editorbandes auf Bandeinheit 5 werden als normaler Text auf der Schreibmaschine ausgegeben. Auf eine Seite kommen dabei 44 Zeilen (anderthalbzeilige Ausgabe). Zur Darstellung von Schreibmaschinen-Sonderzeichen im ASCII-Code siehe Anhang.

DSTOP+ Die Ausgabe auf der Schreibmaschine wird nach Beendigung der nächsten Zeile gestoppt.

nL+ Die Zeile n des Textspeichers wird auf dem Fernschreiber ausgegeben.

m,nL+ Die Zeilen m bis n (einschließlich) des Textspeichers werden auf dem Fernschreiber ausgegeben.

S+ Die Ausgabe auf dem Fernschreiber wird abgebrochen.

4.1.5      TABULATOR  
- - - - -

Beim Schreiben von 'formatierten' Texten wie IBM-Assembler, FORTRAN, etc., bei denen eine festgelegte Spalteneinteilung eingehalten werden muß, kann man sich die Eingabe der Zwischenräume ersparen, wenn man die Tabulator-Simulierung benutzt. Bis zu fünf Tabulatorwerte können vorgegeben werden. Die Taste "ALT MODE" (im Text-mode) bewirkt, daß alle Zeichenpositionen bis zum nächsten Tabulatorwert mit Zwischenräumen gefüllt werden.

CTRL+Y+    Setzen der Tabulatorwerte  
            Es erscheint folgendes Bild auf dem Display:

```
TABS:    t1   t2   t3   t4   t5  
NEXT:    0
```

Dabei bedeuten t1 bis t5 die augenblickliche Stellung der Tabulatorwerte. Sollen diese geändert werden, so sind nun die neuen Werte einzugeben (in monoton steigender Folge). Der jeweils nächste Wert erscheint dabei in der Zeile 'NEXT:' und wird beim Drücken der Taste "RETURN" auf den nächsten Platz in der Tabelle übernommen. Der Wert in der Zeile 'NEXT:' kann durch Drücken der Taste "RUBOUT" gelöscht werden. Die Eingabe wird automatisch nach dem fünften Wert beendet; soll sie vorher abgebrochen werden, so ist CTRL+L zu drücken.

4.1.6      SYSTEMANRUF  
- - - - -

CTRL+S    Das 'SYSTEM' wird geladen und gestartet. Die Systemprozedur, die ausgeführt werden soll, muß im Textspeicher stehen (siehe Beschreibung "SYSTEMANWEISUNGEN").

CTRL+A    Der 'bootstrap-loader' wird gestartet. Das Kommando hat dieselbe Wirkung wie Starten der PDP bei Adresse 7600.

4.2

KORREKTUREN

- - - - -

nV+ Die Zeile n des Textspeichers soll korrigiert werden. Sie wird in einen zweiten Zeilenpuffer übernommen, dessen Inhalt während der Korrektur unter dem ersten auf dem Display angezeigt wird. Das Programm geht für die Dauer der Korrektur in den Text-mode; alle dort beschriebenen Kommandos sind auch hier gültig. Der (erste) Zeilenpuffer kann also ganz normal mit Zeichen gefüllt, gelöscht und korrigiert werden. Zusätzlich bestehen noch folgende Möglichkeiten:

ALT MODE Das erste Zeichen aus dem zweiten Zeilenpuffer wird dort gelöscht und dafür an den Inhalt des ersten Zeilenpuffers angehängt.

CTRL+N Das erste Zeichen im zweiten Textspeicher wird gelöscht.

CTRL+F Der Inhalt des zweiten Zeilenpuffer wird an den Inhalt des ersten Zeilenpuffers angehängt.

RETURN Der Inhalt des ersten Zeilenpuffers wird an die Stelle der Zeile n im Textspeicher eingesetzt.  
Das Programm kehrt in den Kommando-mode zurück.

nS+ Der Inhalt des Speicherplatzes n auf bank 1 soll geändert werden. Es erscheint folgendes Bild auf dem Display:

```
n(oktal)   alter Wert (oktal)   REPL:   0
```

Der neue Wert kann nun eingegeben werden (oktal) und erscheint während der Eingabe in dem Feld hinter 'REPL:'. Durch Drücken der Taste "RETURN" wird der neue Wert übernommen; das Programm kehrt in den Kommando-mode zurück. Während der Eingabe kann der neue Wert durch Drücken der Taste "RUBOUT" gelöscht werden.

Soll die Speicherplatzadresse oktal angenommen werden, muß On statt n geschrieben werden.

### 4.3 ÜBERGANG IN DEN TEXT-MODE - - - - -

Die folgenden Kommandos bewirken den Übergang vom Kommando- in den Text-mode. Hierin bleibt das Programm so lange, bis "CTRL+L" gedrückt wird.

- A+        Der anschließend eingegebene Text wird an den Inhalt des Textspeichers angehängt.
- nI+      Der anschließend eingegebene Text wird vor Zeile n des Textspeichers eingeschoben.
- nC+      Die Zeile n des Textspeichers wird gelöscht und der anschließend eingegebene Text wird an ihre Stelle im Textspeicher eingeschoben. Der neue Text darf dabei aus beliebig vielen Zeilen bestehen.
- m,nC+   Die Zeilen m bis n (einschließlich) des Textspeichers werden gelöscht und an ihre Stelle wird der anschließend eingegebene Text eingeschoben. Der neue Text darf dabei aus beliebig vielen Zeilen bestehen.

4.4        TEXT-MODE  
- - - - -

Wenn sich das Programm im Text-mode befindet, wird folgendes Bild auf dem Display angezeigt:

TEXT MODE        LL=    x

Inhalt des Zeilenpuffers

x bedeutet dabei die momentane Anzahl von Zeichen im Zeilenpuffer.

RUBOUT        Das letzte Zeichen des Zeilenpuffers wird gelöscht.

CTRL+O        Der gesamte Zeilenpuffer wird gelöscht.

RETURN        Der Inhalt des Zeilenpuffers wird in den Textspeicher übernommen (siehe 4.3). Der Zeilenpuffer wird dabei gelöscht.

CTRL+L        Das Programm kehrt in den Kommando-mode zurück.

#### 4.5 IBM-Programm 'SERVICE'

- - - - -

Auf der PDA.MVT.LIBRARY in der IBM existiert ein Modul (rent) mit dem Namen 'SERVICE', der allen On-line-Benutzern zur Verfügung steht. Er wird benötigt, um Benutzermoduln für die einzelnen kleinen Maschinen zu 'laden'. Nebenbei enthält er noch eine Reihe nützlicher Funktionen, die im wesentlichen darin bestehen, beliebige Texte von DECTapes auf bestimmte IBM-Datensätze zu überspielen. Von dort aus können die Texte dann mit (existierenden) Off-line-Programmen weiterverarbeitet werden.

Der Hilfsmodul 'SERVICE' ist immer für alle die Maschinen geladen, für die kein anderer Benutzermodul angefordert wurde. Er wird ebenfalls immer dann geladen, wenn ein Benutzermodul ernsthafte Fehler zeigt und abnorm endet (ABEND xxx).

Die Kommunikation mit dem SERVICE-Modul erfolgt in ähnlicher Weise wie mit dem System in der PDP. Die einzelnen auszuführenden Anweisungen werden als normaler Text in den SUPEDT-Textspeicher geschrieben (ein Kommando per Zeile); sie bilden zusammen den Steuer-Record für den SERVICE-Modul. Durch das Kommando

"CTRL+I"

wird dann der gesamte Steuer-Record zur IBM geschickt. Die einzelnen Anweisungen werden nacheinander ausgeführt. Nach Abarbeitung aller Anweisungen befindet sich der Steuer-Record mit zusätzlichen Kommentaren wieder im SUPEDT-Textspeicher. Die maximale Länge eines Steuer-Records beträgt 100 PDP-Worte.

Vor der Benutzung des SERVICE-Moduls sollte man sicherstellen, daß er auch für die betreffende kleine Maschine geladen ist. Zu diesem Zweck gibt es eine Anweisung, die - unabhängig vom gerade geladenen Programm - das Laden des SERVICE-Moduls erzwingt (siehe Anweisung 'CHANGE').

Als Return-Codes (Inhalt des ersten zurückgesandten Wortes) benutzt der SERVICE-Modul die Zahlen 1 bis 199 (dezimal); diese Codes sollten also in anderen Benutzer-Moduln nicht verwendet werden.

Beschreibung der erlaubten Kommandos  
=====

A. CHANGE  
-----

Das Kommando 'CHANGE' erzwingt das Laden des SERVICE-Moduls für die betreffende Maschine. Dabei ist es gleichgültig, welcher Modul vorher geladen war. 'CHANGE' darf NICHT MIT ANDEREN Kommandos kombiniert werden, sondern muß als gesonderter Kommando-Record zur IBM geschickt werden.

B. LOAD name(,mode)  
-----

Durch das Kommando 'LOAD' wird der Modul 'name' für die betreffende Maschine geladen. 'name' darf aus maximal 8 alphanumerischen Zeichen bestehen. Für 'mode' gibt es zwei Möglichkeiten:

mode=TRANSIENT    der Modul wird transient  
mode=PERMANENT    der Modul wird permanent

Falls man 'mode' nicht angibt, wird TRANSIENT angenommen.

Beispiel:

LOAD F58MODUL                    der F58MODUL wird transient geladen  
LOAD F58MODUL,TRANSIENT        der F58MODUL wird transient geladen  
LOAD F58MODUL,PERMANENT        der F58MODUL wird permanent geladen

Die Bearbeitung der Anweisungen eines Status-Records wird nach einem LOAD-Kommando abgebrochen; evtl. noch folgende Anweisungen werden nicht mehr ausgeführt.

C. UNIT n  
-----

Alle folgenden Kommandos, die die Benutzung einer DECTape-Einheit zur Folge haben, sollen sich auf die Einheit 'n' beziehen. 'n' muß eine Zahl zwischen 1 und 8 (einschließlich) sein.

D. Überspielen von Texten von Editorbändern zur IBM  
-----

Sämtliche Kommandos haben folgendes Format:

command bg,bg,...,bg

Die Texte auf den Editorblöcken 'bg' werden zur IBM überspielt. 'bg' bedeutet dabei die Nummer eines Editorblockes oder eine Gruppe von aufeinanderfolgenden Editorblöcken.

Beispiel:

command 3,30-31,14,8-10

Die Texte auf den Editorblöcken 3,30,31,14,8,9,10 werden in der angegebenen Reihenfolge überspielt.

Die überspielten Texte werden in der IBM zunächst auf Platten-Datensätze geschrieben. Von dort müssen sie durch (auf einer Bibliothek existierende) Off-line-Programme ausgegeben werden. Da aber alle Texte unabhängig davon, von wem und von wo sie überspielt wurden, auf den gleichen Datensätzen landen, muß die erste Zeile eines jeden Textabschnitts bestimmte Kenninformation enthalten, die es dem Ausgabeprogramm ermöglicht, den entsprechenden Textabschnitt wiederzufinden. Diese Textzeile hat folgendes Aussehen:

USER=name

'name' darf aus bis zu 8 beliebigen alphanumerischen Zeichen bestehen.

Unter einem Textabschnitt ist der gesamte Text zu verstehen, der durch ein Kommando übertragen wird.

Folgende Kommandoworte sind zulässig:

- a. PDUMP Der Text wird in PAL4-Formatierung auf dem Zeilendrucker ausgegeben. Dieses Kommando ist also nur für PDP-Programmtexte sinnvoll.
- b. ODUMP Der Text wird unformatiert auf dem Zeilendrucker ausgegeben. Jede Seite enthält in der Überschrift den Namen des Editorbandes, die Blocknummer, die Nummer der Seite innerhalb des Blocks und das Datum. Die einzelnen Textzeilen werden nummeriert.
- c. TDUMP Wie ODUMP, jedoch ohne Zeilennummerierung.
- d. XDUMP Der Text wird unformatiert und ohne Seitenüberschrift und Zeilennummerierung auf dem Zeilendrucker ausgegeben. Jede Druckseite besteht aus 60 Zeilen.
- e. PUNCH Der Text wird unformatiert auf dem Kartenstanzer ausgegeben; die maximale Länge einer Textzeile beträgt 80 Zeichen.
- f. CARDS Der Text wird unformatiert auf dem Kartenstanzer ausgegeben; die maximale Länge einer Textzeile beträgt 72 Zeichen. Die einzelnen Karten werden (in den Spalten 73-80) durchnummeriert.
- g. SYSIN Der Text ist ein auf der IBM ausführbares Programm, welches als normaler Monitor-Job laufen soll. Dieses Kommando setzt also FORTRAN-, Assembler-, PL1- oder COBOL-Programmtexte voraus.

Off-line-Programme für die Weiterverarbeitung der überspielten  
Texte

=====

Die Off-line-Programme, die für die Weiterverarbeitung der  
überspielten Texte benötigt werden, befinden sich auf der  
PDA.AKLIB. Die für diese Programme benötigten Steuerkarten werden  
in folgenden beschrieben. Wichtig ist der zweite Parameter auf  
der EXEC-Karte ('name'); hier muß das gleiche Kennwort stehen wie  
in der ersten Zeile des überspielten Textabschnittes, der  
angesprochen werden soll (USER='name').

Steuerkarten für die Ausgabe auf dem Zeilendrucker (Kommandos:  
PDUMP, ODUMP, TDUMP, XDUMP):

```
//... JOB ...,programmer's name,MSGLEVEL=1
//JOB LIB DD DSNAME=PDA.AKLIB,UNIT=2314,VOL=SER=DISK04,DISP=SHR
// EXEC PGM=F58T,PARM='PRINT,name'
//INPUT DD DSNAME=SERVICE,UNIT=2314,VOL=SER=DISK04,DISP=OLD
//SERVTEMP DD UNIT=SYSDA,SPACE=(3000,(50,50))
//OUTPUT DD SYSOUT=A
```

Steuerkarten für die Ausgabe auf dem Kartenstanzer (Kommandos:  
PUNCH, CARDS):

```
//... JOB ...,programmer's name,MSGLEVEL=1
//JOB LIB DD DSNAME=PDA.AKLIB,UNIT=2314,VOL=SER=DISK04,DISP=SHR
// EXEC PGM=F58T,PARM='PUNCH,name'
//INPUT DD DSNAME=SERVICE,UNIT=2314,VOL=SER=DISK04,DISP=OLD
//SERVTEMP DD UNIT=SYSDA,SPACE=(3000,(50,50))
//OUTPUT DD SYSOUT=B
```

Steuerkarten für die Ausführung eines überspielten Programms in  
Monitor (Kommando: SYSIN):

```
//... JOB ...,programmer's name,MSGLEVEL=1
//JOB LIB DD DSNAME=PDA.AKLIB,UNIT=2314,VOL=SER=DISK04,DISP=SHR
//PREP EXEC PGM=F58T,PARM='SYSIN,name'
//INPUT DD DSNAME=SERVICE,UNIT=2314,VOL=SER=DISK04,DISP=OLD
//SERVTEMP DD UNIT=SYSDA,SPACE=(3000,(50,50))
//OUTPUT DD UNIT=SYSDA,SPACE=(80,(1000,500)),DISP=(NEW,PASS)
//USE EXEC gewünschte Prozedur
//xxx.SYSIN DD DSNAME= .PREP.OUTPUT,DISP=(OLD,DELETE)
//evtl. weitere benötigte Steuerkarten wie bei einem 'normalen'
Monitor-Job
```

Users Return Codes der Off-line-Programme  
=====

Die in den Programmen vorgesehenen Return Codes haben folgende Bedeutung:

- U0001 = auf der EXEC-Karte fehlen Parameter
- U0002 = illegaler Parameter auf der EXEC-Karte
- U0003 = Fehler beim OPEN (OUTPUT-Datensatz)
- U0004 = Fehler beim ersten RENAME
- U0005 = Fehler beim OPEN (INPUT-Datensatz)
- U0006 = Fehler beim zweiten RENAME
- U0007 = der zweite Parameter auf der EXEC-Karte fehlt oder ist länger als 8 Zeichen
- U0008 = es existiert kein Textabschnitt mit der auf der EXEC-Karte angegebenen Kenninformation
- U0009 = Fehler beim OPEN (SERVTEMP-Datensatz)

Die Return Codes 3,4,5,6,9 sollten normalerweise nicht vorkommen können; bei ihrem Auftreten bitte F-58 verständigen!

5 ANHANG  
=====

A. getrimmter ASCII-Code  
-----

Code	Zeichen	Sonder- bedeutung	Code	Zeichen	Sonder- bedeutung
00	@	'CR'	40	'SPACE'	
01	A		41	!	
02	B		42	"	
03	C		43	#	ß
04	D		44	\$	'SPACE'
05	E		45	%	
06	F		46	&	'COMM'
07	G		47	'	
10	H		50	(	
11	I		51	)	
12	J		52	*	'BACK'
13	K		53	+	
14	L		54	,	
15	M		55	-	
16	N		56	.	
17	O		57	/	
20	P		60	0	-
21	Q		61	1	
22	R		62	2	
23	S		63	3	
24	T		64	4	
25	U		65	5	
26	V		66	6	
27	W		67	7	
30	X		70	8	
31	Y		71	9	
32	Z		72	:	
33	[	Ä	73	;	
34	\	Ö	74	<	'NEW PAGE'
35	]	Ü	75	=	
36	↑	'UC'	76	>	'STOP'
37	←	'LC'	77	?	

ASCII = American Standard Code for Information Interchange

Die Sonderbedeutung der ASCII-Zeichen gilt für Ausgabe auf der Schreibmaschine und Benutzung des Typesetting-Programms.

Es bedeuten:

- 'CR' = Wagenrücklauf, Zeilenvorschub
- 'UC' = Umschaltung auf große Buchstaben
- 'LC' = Umschaltung auf kleine Buchstaben
- 'COMM' = Kommandozeichen (Typesetting)
- 'NEW PAGE' = Beginn einer neuen Seite
- 'STOP' = Stoppen der Ausgabe bis "CTRL+G"

B. Bootstrap-loader

PDP-8/PDP-8I			PDP-5		
Adr.	okt. Inhalt	Bedeutung	Adr.	okt. Inhalt	Bedeutung
7600	6224	RIF	7600	1223	TAD RV
7601	6774	DTLB	7601	6757	MMMM
7602	1220	TAD RV	7602	4216	JMS FD
7603	4212	JMS FD	7603	4216	JMS FD
7604	1221	TAD BA	7604	1224	TAD RD
7605	6201	CDFO	7605	6756	MMMF
7606	3623	DCA I AM	7606	4216	JMS FD
7607	1222	TAD RD	7607	1225	TAD BA
7610	4212	JMS FD	7610	6766	MMML
7611	7402	HLT	7611	4216	JMS FD
7612	0000	FD,0	7612	7402	HLT
7613	6766	DTCA DTXA	7613	0000	
7614	3354	DCA 7754	7614	0000	
7615	6771	DTSF	7615	0000	
7616	5215	JMP .-1	7616	0000	FD,0
7617	5612	JMP I FD	7617	6761	MMSF
7620	0600	RV,600	7620	5217	JMP .-1
7621	7577	BA,7577	7621	6772	MMCF
7622	0220	RD,220	7622	5216	JMP I FD
7623	7755	AM,7755	7623	1030	RV,1030
			7624	0022	RD,22
			7625	7600	BA,7600

