

CLI — die zweite Haut des Amiga

Nachdem wir uns in der letzten Folge mit Befehlen beschäftigt haben, die dem Kopieren von Dateien und Formatieren von Disketten dienen, gehen wir in diesem Kursteil hauptsächlich auf den Zeileneditor des CLI ein. Bevor wir uns aber mit dem umfangreichen Befehl »Ed« beschäftigen, werden zuerst (zum Aufwärmen) einige leichter zu verstehende Kommandos des CLI behandelt. Der erste der neuen Befehle, die wir diesmal vorstellen, ist das Kommando »MakeDir«.

Im Regelfall arbeitet man bei einer Diskette oder Festplatte in der obersten Directory-Ebene, dem sogenannten »Root-Directory«, in welches Sie immer mit dem Befehl

cd :

gelangen (korrekterweise sollte »cd Laufwerk:« geschrieben werden, um sicherzugehen, daß auch das richtige Laufwerk angesprochen wird). Sie können in dieses Verzeichnis alle Programme speichern, die Sie möchten. Wenn Sie sich aber nun vorstellen, daß Sie (etwa bei Festplatten) bis zu mehrere hundert Programme in dieses Verzeichnis speichern können, kommt Ihnen sicher der Gedanke, daß beispielsweise die Directory-Anzeige dieser Diskette/Festplatte sehr unübersichtlich wird, da die einzelnen Einträge gar nicht alle auf den Bildschirm passen. Außerdem können weitere Probleme auftreten, wenn man etwa alle Dateien, die zu einem Programm gehören, kopieren möchte und diese erst mühsam zusammensuchen muß, da nicht klar ersichtlich ist, welche dies sind. Zu diesem Zweck schufen findige Entwickler Subdirectories. Dies sind Unterverzeichnisse, die einem anderen Directory untergeordnet sind, also Dateiverzeichnisse in einem Dateiverzeichnis. Damit ist es möglich, für bestimmte Anwendungen eigene »Abteilungen« auf der Diskette/Festplatte zu schaffen, in denen dann zu einem Gebiet gehörige Programme gespeichert werden (Bild). Bestes Beispiel dafür ist die Workbench-Diskette. Sie enthält eine Vielzahl

Wie Sie mit dem CLI-Editor »Ed« und weiteren Befehlen umgehen können, erfahren Sie in diesem Teil des CLI-Kurses.

verschiedener Subdirectories, in denen teilweise noch weitere Unterverzeichnisse eingebettet sind. Von diesen Ordnern sind aber auf der Workbench-Ebene nur die zu sehen, zu denen ein entsprechendes »info«-File existiert, welches beispielsweise die Grafikdaten für das Bildsymbol zur Verfügung stellt.

Der »c«-Ordner enthält nur Programme, die etwas mit dem CLI zu tun haben. Im Ordner »devs« dagegen befinden sich nur Programme, die etwas mit der Tastaturanpassung, Druckern oder anderen Gerätetreibern zu tun haben.

Auch Sie werden früher oder später in Situationen kommen, in denen Sie der Übersichtlichkeit halber Dateiverzeichnisse erstellen müssen. Das CLI stellt dafür den Befehl »makedir« zur Verfügung. Die Schreibweise für dieses Kommando lautet allgemein:

```
makedir (Verzeichnis)
Name
```

»Verzeichnis« muß nicht angegeben werden, wenn Sie in dem Directory, in dem Sie sich gerade befinden, ein Unterverzeichnis anlegen möchten. Soll aber ein Verzeichnis generiert werden, das beispielsweise zwei Ebenen tiefer liegt (Bild untere Reihe), ist der komplette Pfad anzugeben. Zwei Beispiele:

```
makedir df0:Anwendung
makedir df0:Anwendung/
Grafik
```

Im ersten Fall wird im aktuellen Verzeichnis das Verzeichnis »Anwendung« erstellt. Im zweiten Fall dagegen entsteht der Ordner »Grafik« im Verzeichnis »Anwendung«, das bereits dem aktuellen Directory untergeordnet ist. Das Directory »Grafik« ist also in der zweiten Verzeichnisebene angesiedelt (siehe dazu auch das Bild). Die Angabe des Laufwerks ist nicht unbedingt erforderlich, erhöht aber die Lesbarkeit. Ebenfalls ist darauf zu

achten, daß der Name des zu erzeugenden Verzeichnisses in Anführungszeichen gesetzt wird, wenn Leerzeichen oder Sonderzeichen enthalten sind.

Der Amiga 2000 enthält fest eingebaut eine batteriegepufferte Echtzeituhr, die auch nach dem Ausschalten des

Computers weiterläuft und somit beim nächsten Einschalten die aktuelle Uhrzeit zur Verfügung stellt. Auch ein Amiga 500, der mit der intern einzubauenden Speichererweiterungskarte (Erweiterung auf 1 MByte) ausgerüstet ist, verfügt über eine solche Echtzeituhr. Leider kann diese Uhr nicht mit »Date« gestellt und ausgelesen werden. Dazu existiert auf neueren Workbench-Versionen (ab Versionsnummer 33.4xx) der Befehl »Setclock«. Dieser Befehl ermöglicht das Übertragen der Amiga-Systemuhr von und zur batteriegepufferten Echtzeituhr.

Teil 3

KURSÜBERSICHT

Der CLI-Kurs erläutert Ihnen in mehreren Folgen den Umgang mit der Amiga-Benutzerschnittstelle CLI (Command-Line-Interface). Am Ende haben Sie alle wichtigen Befehle kennengelernt und können die Kommandos nach eigenen Wünschen einsetzen. Die folgende Kursübersicht zeigt Ihnen, in welchen Kursteilen welche CLI-Kommandos erklärt werden:

TEIL 1: behandelte Befehle zum Anzeigen des Directories und Stellen der Uhr (dir, list, cd, date, endcli)

TEIL 2: erklärte Ihnen, wie Sie Dateien kopieren, löschen, umbenennen und Disketten formatieren können (filenote, delete, relabel, rename, type, copy, diskcopy, info, format, install)

TEIL 3: zeigt unter anderem auf, wie Texte und Startdateien erstellt und bearbeitet werden (makedir, ed, prompt, wait, loadwb, setclock, execute)

TEIL 4: befaßt sich mit Befehlen, die sich zum Einbau in eine Batch-Datei anbieten (assign, echo, failat, if endif, then, else, stack, protect, bind-drivers)

TEIL 5: setzt die Erklärung von Kommandos für eine Batch-Datei fort (run, edit, join, lab, quit, skip, addmem, break)

TEIL 6: geht auf neue Spezialbefehle für Zusatzgeräte wie etwa Festplatten ein (search, newcli, sort, why, djmount, dpformat, mount, addbuffers, diskchange)

Aktivieren der (optionalen) Echtzeituhr

Das Stellen dieser Uhr geht relativ einfach vonstatten: zuerst ist mit dem »Date«-Befehl die Systemuhr des Amiga zu stellen (siehe dazu auch Teil 2 des Kurses, Ausgabe 8/9). Anschließend muß dieses Datum in die Echtzeituhr übertragen werden. Der Befehl dazu sieht folgendermaßen aus:

```
Setclock opt save
```

Das Datum (und damit auch die Uhrzeit) sind nun in die Echtzeituhr übertragen und bleiben auch nach dem Ausschalten des Amiga erhalten.

Nach einem erneuten Einschalten des Computers befindet sich das Datum noch in der Echtzeituhr, nicht aber in der Systemuhr des Amiga. Die Zeit muß also bei jedem Neustart wieder aus der Echtzeituhr ausgelesen und in die Systemuhr übertragen werden. Auch hierzu dient wieder der Setclock-Befehl, diesmal nur mit anderen Parametern:

```
Setclock opt load
```

Dies funktioniert aber nur dann, wenn Sie tatsächlich über die batteriegepufferte Uhr verfügen. Ansonsten meldet das Amiga-DOS (Betriebssystem) einen Fehler!

Damit dieser letztgenannte Befehl nicht jedesmal von Hand eingegeben werden muß, bietet er sich geradezu dafür an, in die Startdatei der

Workbench-Diskette eingebunden zu werden, die jedesmal beim Neustart des Amiga abgearbeitet wird. Wie das gemacht wird, erfahren Sie etwas später im Text, wenn der »Ed«-Befehl besprochen wird.

Ein ebenfalls leicht zu verstehender Befehl, der sich auch zum Einbinden in die Startdatei eignet, ist das Kommando »Prompt«.

Doch was ist ein »Prompt«? Wie Sie sicher bereits festgestellt haben, steht im CLI-Fenster immer die Kennung des jeweiligen Command-Line-Interfaces (Sie können mehrere CLI-Fenster gleichzeitig öffnen) vor dem Cursor (1>). Sie können diese Meldung aber auch ändern, wenn eine »eigene« Bereitschaftsanzeige gewünscht wird. Nehmen wir an, der normale Prompt soll so geändert werden, daß er eine persönlichere Note erhält:

Prompt "Hallo, User> "

Nach Eingabe dieses Kommandos würde nun im aktuellen CLI-Fenster anstelle des »1>«-Prompts die Bereitschaftsanzeige »Hallo, User>« erscheinen.

Wenn Sie aber zusätzlich zu dieser neuen Meldung auch noch die Nummer des aktuellen CLI-Tasks sehen möchten, müssen Sie nur den Parameter »%N« einfügen. Für das obige Beispiel sähe dies so aus:

Prompt "Hallo, User %N> "

Das Ergebnis auf dem Bildschirm hätte die Form »Hallo, User 1>«. Den normalen, gewohnten Prompt bringt Ihnen folgender Befehl zurück:

Prompt %N> "

Ein weiterer Befehl, der eigentlich nur sehr selten zum Einsatz kommt, ist das Kommando »Wait«. Es ist im Regelfall nur für Besitzer eines Amiga 2000 mit PC-/AT-Karte oder eines Sidecar interessant, die eine Festplatte besitzen und bei denen der Controller in einem PC-Steckplatz steckt. Hier muß der Amiga so lange mit Festplattenzugriffen warten (etwa 40 Sekunden), bis der PC nach dem Booten die Platte nicht mehr benötigt (sonst kommen sich beide Systeme bei Zugriffen auf die Platte ins Gehege).

Der Wait-Befehl kann jedoch durch seine vielfältigen Möglichkeiten auch für andere Zwecke eingesetzt werden. Beispielsweise, wenn eine mit »Echo« ausgegebene Meldung (siehe Teil 4) länger am Bild-

schirm sichtbar bleiben soll und erst danach das Startprogramm weiter abgearbeitet wird. Im einfachsten Fall wartet der Amiga 1 Sekunde, bis er mit der Abarbeitung der Startdatei fortfährt. Dies erreicht man durch Eingabe von »Wait« ohne weitere Parameter. Soll länger gewartet werden, muß, durch ein Leerzeichen getrennt, die Zeitdauer in Sekunden (SEC) oder Minuten (MIN) angegeben werden. Zwei Beispiele:

Wait 32sec
Wait 3min

Im ersten Fall hält das System für 32 Sekunden, beim zweiten Beispiel wartet der Amiga drei Minuten. Es ist aber auch möglich, den Computer bis zum Erreichen einer bestimmten Uhrzeit anzuhalten. Das nachstehende Beispiel würde den Amiga warten lassen, bis die angegebene Uhrzeit erreicht wird:

Wait UNTIL 12:00

Die Diskettensymbole werden sichtbar

Ein dagegen sehr wichtiger Befehl, der eigentlich in jede Startdatei gehört, ist »Load WB«. Erst dieser Befehl ermöglicht es dem Anwender, daß er mit den Icons (Bildsymbolen) auf der Workbench-Ebene arbeiten kann.

LoadWB erlaubt noch die Angabe eines speziellen Zusatzparameters:

LoadWB -debug

Hiermit wird eine Spezialfunktion (für »Eingeweihte«: ROMWack) des Amiga aktiviert, die nur für Programmierer interessant ist, die sich sehr gut mit dem Amiga-Betriebssystem auskennen und auch Maschinensprache beherrschen müssen. Deshalb soll auf diese Zusatzfunktion nicht näher eingegangen werden, da bei falscher Anwendung die Gefahr eines »endgültigen Systemabsturzes« besteht.

Nach dieser »Aufwärmübung« besprechen wir nun einen der umfangreichsten Befehle, die uns das CLI zur Verfügung stellt: den Texteditor »Ed«. Er ermöglicht das Erzeugen und Bearbeiten von Texten, die im ASCII-Format (!) vorliegen müssen. Ein Beispiel für eine solche Textdatei ist die in diesem Kurs bereits öfter erwähnte Startdatei »Startup-Sequence«, welche sich im

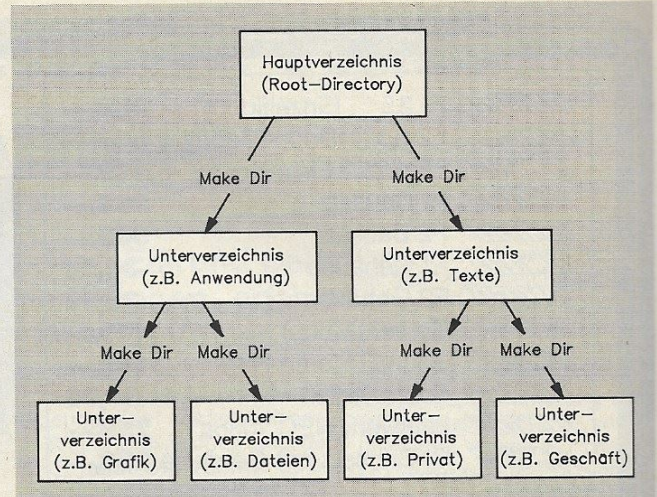


Bild. Ein Beispiel für eine Diskette/Festplatte, auf der nach Themen geordnet Subdirectories erzeugt wurden

»s«-Directory befindet. Bei jedem Neustart des Amiga wird diese Datei geladen und alle in ihr enthaltenen Befehle der Reihe nach abgearbeitet. Sie können sich den Inhalt dieser Datei beispielsweise mit dem Befehl

Type :s/startup-sequence

ansehen. Bearbeiten läßt sie sich jedoch nur mit dem Editor.

Der Aufruf von Ed erfolgt sehr einfach. Es ist nur der Name der zu bearbeitenden Datei sowie, wenn erforderlich, die Größe des zu reservierenden Textspeichers anzugeben. Wird keine Größe angegeben, reserviert Ed von sich aus 40000 Byte Textspeicher, was völlig genügen müßte.

Versuchen wir doch gleich einmal, eine Datei zu bearbeiten beziehungsweise zu erzeugen. Geben Sie folgendes ein:

Ed :Test-Ed SIZE 50000

Die Angabe von »size 50000« wäre nicht unbedingt erforderlich gewesen, sondern soll nur die Parameterübergabe zur Größenbestimmung des zu reservierenden Textspeichers demonstrieren. Ebenso ist auch der Doppelpunkt vor dem Dateinamen nicht nötig. Er stellt nur sicher, daß die Datei im Root-Directory (Hauptverzeichnis) des aktuellen Laufwerks erzeugt wird.

Da die Datei Test-Ed noch nicht existiert, gibt Ed eine Meldung aus, die besagt, daß eine neue Datei erzeugt wird. Doch widmen wir uns nun den Funktionen des Editors:

Im normalen Betrieb des Editors, in dem Sie sich nach dem Start befinden, können Sie ganz normal Texte eingeben. Da Ed ein »Fullscreen-Editor« ist, läßt sich der Cursor

ohne Probleme im ganzen Fenster bewegen. Ausnahme ist hier nur die letzte Zeile des Fensters, die für Editor-Meldungen reserviert ist.

Außer der Eingabe des ASCII-Textes beherrscht Ed eine Vielzahl von Befehlen, die dem Laden, Speichern und Bearbeiten des Textes dienen. Alle diese Befehle (von Commodore als »erweiterter Modus« bezeichnet) werden durch einen Druck auf die ESC- und CTRL-Taste eingeleitet. Da die einzelnen Steuerbefehle im mitgelieferten Handbuch ausführlich genug beschrieben sind, soll Ihnen die Tabelle zusammenfassend einen Überblick über die wichtigsten Funktionen geben.

Doch wozu kann man nun den Editor verwenden, werden Sie sich vielleicht fragen? Mit einer Textverarbeitung kann doch viel komfortabler gearbeitet werden! Doch dazu dient Ed ja auch gar nicht. Vielmehr ist er dazu da, kleine ASCII-Files wie beispielsweise die Startdatei oder ähnliche, ausführbare Dateien, sogenannte Batch-Dateien, zu erstellen.

Das Betriebssystem des Amiga erlaubt es, verschiedene Befehle des CLI zu einem kleinen Programm zusammenzufassen, das den Sinn hat, einzelne Kommandofolgen, die häufig gebraucht werden, durch einen einzigen Befehl zu ersetzen. Versuchen wir doch gleich einmal, eine solche ausführbare Datei zu erzeugen! Da wir uns bereits im Editor befinden, können wir gleich loslegen. Als Beispiel werden wir mit den Befehlen, die wir bis jetzt kennengelernt haben, eine Batch-Datei erstellen.

Folgende Befehle, die natür-

lich auch im Direktmodus eingegeben werden können, eignen sich gut für diese Datei:

```
dir
info
loadwb
setclock opt load
```

Der letzte Befehl ist natürlich nur dann sinnvoll, wenn Sie über die batteriegepufferte Echtzeituhr verfügen.

Geben Sie nun die drei beziehungsweise vier Befehle im Ed-Fenster ein und schließen jeden Befehl durch <RETURN> ab. Wenn Sie anschließend <ESC> <X> <RETURN> eingeben, wird der Editor beendet, nachdem die kleine Batch-Datei unter dem Namen »Test-Ed« in das Hauptdirectory des aktuellen Laufwerks gespeichert wurde. Sie können sich den Inhalt dieser ASCII-Datei noch einmal mit dem Befehl

Type :Test-Ed

anzeigen lassen. Jedoch können Sie ohne weiteres nichts mit dieser Datei anfangen. Zum Aufruf eines Batchfiles stellt das CLI einen weiteren Befehl zur Verfügung: das Kommando »Execute«.

Nach dem Aufruf des Execute-Befehls, hinter dem der Name der auszuführenden Datei angegeben werden muß, startet der Amiga mit der Ausführung der Batch-Datei. Die Arbeit kann, wenn erforderlich, mit <CTRL D> abgebrochen werden.

Um die soeben erzeugte Datei zu starten, geben Sie also folgendes ein:

Execute :Test-Ed

Der Amiga wird anschließend das Directory des aktuellen Laufwerks anzeigen, danach den Zustand und die Belegung der angeschlossenen

Geräte (Laufwerke) ausgeben und schließlich nochmals die Workbench laden. Wenn Sie den vierten Befehl auch noch eingetippt haben, zeigt der Amiga nochmals die aktuelle Zeit aus der batteriegepufferten Uhr an. Der Befehl Execute kann zwar noch viel mehr als nur eine Batch-Datei aufrufen, doch diese weiteren Möglichkeiten werden erst besprochen, wenn Sie noch weitere Befehle kennengelernt haben, die sich zur Einbindung in eine Batch-Datei eignen.

Sie können sich natürlich auch die Startdatei der Workbench-Diskette mit dem Kommando

Ed :s/startup-sequence

betrachten und mit ihr herumexperimentieren. Achten Sie aber vorher darauf, daß die Diskette schreibgeschützt ist, da Sie sonst unter Umständen schwer zu behebende Fehler

machen könnten, wenn die möglicherweise veränderte Datei auf die Diskette zurückgeschrieben wird. Deshalb sollte in diesem Fall der Editor mit <ESC> <Q> verlassen werden, um ein Zurückschreiben der veränderten Batch-Datei zu verhindern.

Wir werden aber in den nächsten Folgen des CLI-Kurses weitere Befehle kennenlernen, die es uns dann erlauben, eigene Startdateien zu schreiben beziehungsweise bereits bestehende zu modifizieren. Bis dahin können Sie aber versuchen, mit den bereits kennengelernten Befehlen eigene Batch-Dateien zu erstellen oder die Kommandos in Ihre Startdatei einzubinden (natürlich nur auf einer Kopie Ihrer Workbench-Diskette).

(dm)

Unterstützende Literatur: Das Amiga-Handbuch, Markus Breuer, Markt & Technik-Verlag, 49 Mark

Alle Befehle des Editors »Ed« auf einen Blick

<CTRL A> — an der aktuellen Cursorposition wird eine Zeile eingefügt
 <CTRL B> — die Zeile, in der sich der Cursor gerade befindet, wird gelöscht
 <CTRL D> — der Text im Ed-Fenster wird um einige Zeilen nach unten gescrollt
 <CTRL E> — setzt den Cursor entweder an die obere oder untere Begrenzung des Ed-Fensters
 <CTRL F> — damit läßt sich der Buchstabe, auf dem sich der Cursor befindet, von Kleinschrift auf Großschrift und umgekehrt wechseln
 <CTRL G> — das letzte ESC-Kommando wird noch einmal ausgeführt
 <CTRL H> — das Zeichen, das links vom Cursor steht, wird gelöscht (Funktion wie die Backspace-Taste)
 <CTRL I> — der Cursor springt auf die nächste Tabulatormarke (normalerweise drei Zeichen, Funktion wie TAB-Taste)
 <CTRL O> — Löscht ein Wort oder Leerstellen bis zum Beginn des nächsten Wortes oder Leerzeichens
 <CTRL R> — bewegt den Cursor auf den Schluß des vorhergehenden Wortes
 <CTRL T> — bewegt den Cursor auf den ersten Buchstaben des nächsten Wortes
 <CTRL U> — der Text im Ed-Fenster wird um einige Zeilen nach oben gescrollt
 <CTRL V> — das Ed-Fenster wird neu gezeichnet (>refresh<)
 <CTRL Y> — der dem Cursor folgende Text wird bis zum Zeilenende gelöscht
 <CTRL J> — positioniert den Cursor auf den Anfang oder das Ende der aktuellen Zeile
 <ESC> <A> »Text« — der in den Anführungszeichen stehende Text wird in der nächsten Zeile eingefügt
 <ESC> — der Cursor springt an das Ende des gesamten Textes
 <ESC> <E> — die aktuelle Cursorposition wird als das Blockende markiert
 <ESC> <S> — die aktuelle Cursorposition wird als Blockanfang markiert
 <ESC> <D> — die Zeile, in der sich der Cursor gerade befindet, wird gelöscht
 <ESC> <D> — der mit »BS«/»BE« markierte Block wird gelöscht
 <ESC> <E> »Text1« »Text2« — der Text »Text1« wird durch »Text2« ersetzt
 <ESC> <E> <Q> »Text1« »Text2« — wie »E«, nur wird vorher eine Bestätigung verlangt
 <ESC> <E> <X> — die rechte Randmarkierung wird ignoriert, es

kann also über das eingestellte rechte Randende hinausgeschrieben werden

<ESC> <F> »Text« — von der Cursorposition ab vorwärts wird der Text nach einer bestimmten Zeichenfolge durchsucht

<ESC> <I> »Text« — vor der Zeile, in der sich der Cursor gerade befindet, wird ein Text eingefügt

<ESC> <I> — der mit »BS«/»BE« markierte Block wird der der Cursorposition folgenden Zeile nochmals eingefügt

<ESC> <I> <F> »Name« — dieser Befehl gestattet es dem Anwender, in der nach dem Cursor folgenden Zeile die Datei »Name« einzufügen

<ESC> <L> <C> — bei der Suche nach einem Text (>F<) soll zwischen Groß- und Kleinschreibung unterschieden werden

<ESC> <N> — der Cursor wird an den Anfang der nächsten Zeile gesetzt

<ESC> <P> — der Cursor wird an den Anfang der vorhergehenden Zeile gesetzt

<ESC> <Q> — der Editor wird verlassen, ohne daß der bearbeitete Text gespeichert wird. Vorher erfolgt aber eine Sicherheitsabfrage.

<ESC> <R> <P> — der nach RP eingegebene Befehl wird so lange wiederholt, bis ein Fehler auftritt oder das Textende erreicht ist. Beispiel: RP D

<ESC> <S> <A> »Name« — der Text wird gespeichert, ohne den Editor zu beenden. Wird »Name« weggelassen, erfolgt die Speicherung unter dem Namen, der beim Aufruf von Ed angegeben wurde.

<ESC> <S> — der mit »BS«/»BE« markierte Block wird im Ed-Fenster angezeigt

<ESC> <S> <H> — mit diesem Kommando kann man sich die aktuellen Einstellungen des Editors anzeigen lassen

<ESC> <S> <L> x — Der linke Randsteller wird auf Spalte »x« gesetzt

<ESC> <S> <R> x — Der rechte Randsteller wird auf Spalte »x« gesetzt

<ESC> <S> <T> x — dieser Befehl dient zum Einstellen der Tabulatorstopps (der Normalwert beträgt x=3)

<ESC> <T> — der Cursor wird an den Textanfang gesetzt

<ESC> <U> — damit werden Änderungen an der aktuellen Zeile rückgängig gemacht. Dies funktioniert aber nur, wenn noch nicht <RETURN> gedrückt wurde.

<ESC> <U> <C> — bei der Suche nach Text (>F<) wird nicht zwischen Groß- und Kleinschreibung unterschieden

<ESC> <W> »Name« — der mit »BS«/»BE« markierte Block wird in die Datei »Name« gespeichert

<ESC> <X> »Name« — mit diesem Befehl wird der Text gespeichert und Ed beendet. Wird dabei »Name« nicht angegeben, so nimmt Ed den Namen her, der beim Editor-Aufruf mit angegeben wurde.

Tabelle 1. Der Texteditor »Ed« beherrscht alle Funktionen, die zum Bearbeiten von ASCII-Texten erforderlich sind