

Tastatur-Umdefinierungsprogramm

UMTAST

Belegung beliebiger Tasten mit beliebigen Zeichen, Sequenzen und Texten auf beliebigen Tastaturen.

Ausnutzung von <ESC>-Sequenzen und Tasten mit Codes >80H neben den Standardtasten und CTRL-Zeichen.

Unterschiedliche Umkodierungstabellen für jedes Programm. Automatischer Ladevorgang.

ELEKTRONIKLADEN
Giesler & Danne GmbH & Co.KG
W. Mellies Str. 88 · 4930 DETMOLD 18

Telefon: 05232 - 8131

Telex: 931473 laden d

Geöffnet Mo - Fr 9-17h u.n. Vereinbarung

(C)Copyright 1982 ELEKTRONIKLADEN Giesler&Danne GmbH&Co KG

ELZET 80

UMTAST

Das Programm UMTAST ist Eigentum der ELEKTRONIKLADEN Giesler und Danne GmbH&Co.,KG in 4930 Detmold 18 (im folgenden "ELEKTRONIKLADEN") und urheberrechtlich geschützt. Mit der Benutzung werden die Bedingungen des Lizenz- und Nichtweitergabevertrags des ELEKTRONIKLADEN anerkannt.

ELEKTRONIKLADEN übernimmt keine Garantie für die Angaben in dieser Beschreibung und behält sich das Recht vor, jederzeit und ohne Kenntnisgabe Änderungen vorzunehmen.

ELEKTRONIKLADEN übernimmt keine Haftung für Schäden, die durch den Gebrauch von UMTAST entstehen, mittelbar oder unmittelbar.

Inhaltsverzeichnis

Kurzbeschreibung.....	Seite 3
Lieferumfang.....	Seite 4
Bild: UMTAST-Module für versch.Anwendungen.....	Seite 5
Anwendungsübersicht.....	Seite 6
Anwendungsbeispiele.....	Seite 7
Funktionsweise.....	Seite 10
Die Installation.....	Seite 11
Die Autoload-Funktion.....	Seite 12
Tastenprogrammierung.....	Seite 14
Änderungen nach Ende des Installationslaufs.....	Seite 16
CP/M-Autoload.....	Seite 17
ASCII-Maskierung.....	Seite 18

UMTAST

Wenn Sie das Handbuch nicht lesen wollen, lesen Sie bitte wenigstens diese Seite !

Aufruf:

UMTAST LEER.COM NEU.COM

benutzt als Eingabedatei:

LEER.COM

Enthält das eigentliche Wandlerprogramm und eine "leere" Tabelle entsprechend einer undefinierten Tastatur.

benutzt als Ausgabedatei:

NEU.COM

Das resultierende Umkodierungsprogramm mit Ihren Einträgen.

Erste Abfrage:

Filename fuer Autoload ?

Erwartet den Dateinamen des Programms, das automatisch durch "NEU.COM" geladen werden soll. z.B. ED.COM, WS.COM...

Alle weiteren Abfragen:

Funktionstaste ?

Erwartet Tastendruck, zeigt Hex-Wert der gedrückten Taste und fragt "Neubelegung (j/n) ?". Bei Antwort mit j wird eine Zeichenkette abgefragt. Steuerzeichen (CTRL) werden durch vorangehende Eingabe von ^ gespeichert. So ist auch z.B. die Speicherung von <CR> als ^M oder <ESC> als ^Å möglich. Eingabeende für die Zeichenkette mit <CR>.

Ende der Umdefinierung: ^C bei "Funktionstaste?"

UMTAST

Lieferumfang

UMTAST ist ein Programm zum Einsatz auf beliebigen CP/M-Systemen mit einem Z80-Prozessor. Bei Benutzung von Funktionstasten mit Zeichen >80 Hex muß die ASCII-Maskierung im CP/M-CBIOS entfernt werden.

Es wird üblicherweise geliefert auf einer einseitigen 8"-Floppy mit einfacher Bitdichte (single side/single density) oder auf den ELZET 80 5 1/4"-Formaten. Formate für andere Rechner können gegen Aufpreis geliefert werden.

UMTAST besteht aus drei Dateien:

UMTAST.COM
LEER.COM
WEG.COM

UMTAST erzeugt beliebig viele neue Dateien des Typs .COM, die die Umkodierungsinformation für zugehörige Programme beinhalten. UMTAST führt den Bildschirmdialog bei der Installation eines Umkodierungsprogramms, liest eine vordefinierte Datei ein und erstellt die Tabelle.

LEER ist die Basis für alle Umdefinierungsprogramme. LEER enthält die eigentliche Tabelle, den Relokator, das CONIN (Tastatureingabe) -Simulationsprogramm und das Ladeprogramm für das zugehörige Programm (Editor etc.)

WEG hat eine spezielle Funktion, die nur mit XSUB, DESPOOL o.ä. und gleichzeitigem Betrieb des Umkodierungsprogramms als CP/M-Autoload-File benötigt wird.

UMTAST

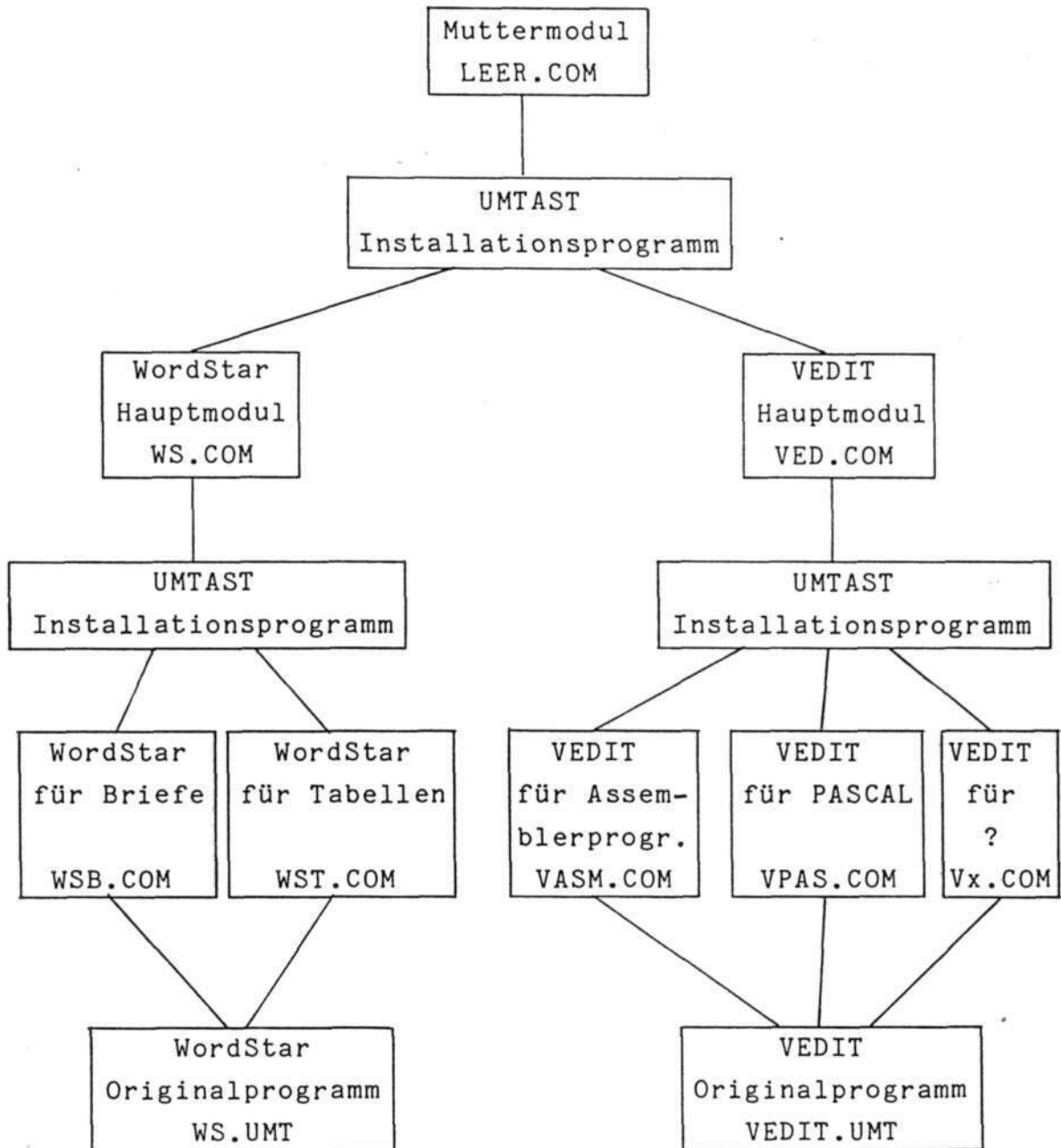


Bild: Der Aufbau von UMTAST-Modulen für verschiedene Anwendungen, teilweise mit gleichen Hauptprogrammen.

UMTAST

Anwendung

Moderne Terminals und Tastaturen stellen eine Vielzahl von Funktionstasten zur Verfügung, die in Anwenderprogrammen ausgewertet werden können. Dies geschieht jedoch fast ausschließlich im kommerziellen Bereich, wo ein Programm auf eine bestimmte Hardware zugeschnitten werden kann.

Der durchschnittliche Benutzer eines CP/M-Computers hat aber häufiger mit Standardprogrammen zu tun, die sich nicht auf seine Wünsche einstellen lassen.

UMTAST dient dazu, ein benutzerspezifisches Interface zwischen beliebigen Eingabetastaturen und Standardprogrammen zu schaffen. Dazu kann jede Taste (außer ^C) eines Terminals oder einer Tastatur anders belegt werden. Neben den Standardtasten können auch alle Escape-Sequenzen (also <ESC> und ein folgendes Zeichen) und alle Zeichen mit gesetztem 8. Bit (größer als 127 Dez./80 Hex) verwendet werden.

Damit kann auch eine Tastatur ohne Funktionstasten Verwendung finden, wenn man in der Lage ist, das <ESC>-Zeichen (^Ä/27/1BH) zu erzeugen.

Zu Speichern sind anstelle des Originalcodes:

- a) ein anderer Code oder
- b) eine Mehrzeichen-Sequenz oder
- c) mehrere Wörter

Anwendungsbeispiele

a) Textprozessor (z.B. MicroPro's WordStar)

Textprozessoren verarbeiten hunderte von Befehlen für die Cursorsteuerung, Randbegrenzung, Tabulatorfunktion, Blockverschiebungen, Dateioperationen u.v.m.

Diese Befehle lassen sich nicht durch Drücken einer einzigen Taste erzeugen, die häufig benötigte Funktion "Datei sichern und zurück an die aktuelle Cursorposition" beispielsweise verlangt bei WS die 4 Zeichen: ^Ks^Qp. Um an das rechte Zeilenende zu gelangen, muß man ^Qd eingeben und um an der Cursorposition ein Tab zu setzen, erwartet WordStar die drei Zeichen ^Oi<ESC>.

All diese Funktionen und auch Kombinationen mehrerer Funktionen lassen sich in einzelnen Tasten speichern und erhöhen dadurch die Nutzungsrate von ungebräuchlichen Befehlen und einhergehend auch die Arbeitsgeschwindigkeit. Darüberhinaus lassen sich neben den WordStar-Befehlen auch häufige Wendungen wie "Mit freundlichen Grüßen", "Sehr geehrte", Markennamen (ELZET 80), eine Namenszeile und dergleichen in Funktionstasten speichern.

Wenn das Dezimalfeld einer Tastatur wahlweise andere Zeichen abgeben kann (wie bei ELZET 80 DIN-TAST), kann es gut als großes Cursorfeld Verwendung finden, mit allen Befehlen wie Wort rechts, Anfang Zeile usw.

UMTAST

Bei Verwendung mit WordStar bietet sich die Ausnutzung der UMTAST-Autoload-Funktion an. Bei Spezifikation dieser Option im Installationsprogramm wird nach Laden der Umkodierungstabelle der Textprozessor automatisch nachgeladen. So wird für den Benutzer kein Unterschied zum Laden eines WordStar ohne Umkodierung merklich.

b) Bildschirmeditor (wie z.B. VEDIT)

Hier bietet VEDIT von sich aus schon hohe Flexibilität, da die Cursorsteuerungstasten dialoggesteuert zugewiesen werden können.

In der Bildschirm-Betriebsart kommt nur ein Teilaspekt von UMTAST zur Geltung, die Speicherung ganzer Wörter. VEDIT arbeitet dann mit den Schlüsselwörtern Ihrer bevorzugten Programmiersprache auf den Funktionstasten.

Da bieten sich z.B. für BASIC an:

INPUT "	FOR	= 1 TO	NEXT	
GOSUB	THEN	LIST	RUN	usw.

In der Befehlsbetriebsart von VEDIT lassen sich, wie bei WS, ganze Befehlsketten auf einer Taste vereinen.

Man kann VEDIT natürlich von verschiedenen UMTAST-Programmen aus laden, so daß ein VEDBAS neben VEDPLI oder VEDC existiert, alle mit VEDIT.COM als Autoload-File. Für PL/I beispielsweise bieten sich so längliche Eingaben wie "put skip list(" oder "get edit(" an.

UMTAST

c) Terminal ohne deutsche Umlaute

Da die Produkte auf dem deutschen Computermarkt hauptsächlich von englischsprachigen Herstellern stammen, findet man häufig Terminals, bei denen Z und Y vertauscht sind, Ä, Ö und Ü garnicht existieren usw. Hier bietet UMTAST die Möglichkeit zur "Eindeutschung" ausländischer Terminals.

Zwar können ä und ü durch UMTAST nicht dargestellt werden, aber die Eingabe dieser Zeichen (entsprechend den geschweiften und eckigen Klammern) kann zumindest normalisiert werden. Häufig sind nämlich auf einer Taste die Ä und Ü entsprechenden Zeichen (normal und Shift), auf einer anderen ä und ü. Andere Terminals haben zwar beide Ä's auf einer Taste, aber Ä normal und ä bei Shift (eckige Klammer unten, geschweifte oben). Ändert man diese Belegungen auf eine korrekte Anordnung, so hat man zwar immer noch Hieroglyphen auf dem Schirm. an die gewöhnt man sich aber leichter als an falsche Tastenbelegungen. Für den Drucker sind ja preiswert deutsche Typenräder erhältlich, wenn nicht, wie bei den meisten Matrixdruckern, der Zeichensatz wählbar ist.

In deutschen Texten ungebräuchliche Zeichen, wie z.B. das Nummernzeichen #, können dann noch mit Sonderfunktionen belegt werden.

Ein letztes Anwendungsbeispiel: CP/M-Befehle wie PIP B:= oder STAT *.* auf Tasten legen, eventuell mit CP/M-Autoload.

UMTAST

Funktionsweise

UMTAST besteht aus einem Installationsprogramm (UMTAST.COM) und einem benutzerspezifischen Programm beliebigen Namens, das mit UMTAST.COM aus einem Mutterprogramm ohne Tabelleneinträge (LEER.COM) generiert wird.

Das mit UMTAST aus LEER.COM erstellte Programm, beispielsweise NEU.COM, wird bei Aufruf aus der CP/M-Befehlszeile in den Speicher ab 100H geladen.

Beispiel: A> NEU

Das Hauptprogramm in NEU.COM untersucht nun, wo der CP/M-Eingabeprozessor (CCP) anfängt. Anschließend wird die Länge der Umkodierungstabelle festgestellt und diese zusammen mit einem kurzen Programm zur Eingabesimulation unter den CCP geschoben. Ist bereits eine andere UMTAST-Tabelle im Speicher, wird diese überschrieben. Dieses Verfahren ist das gleiche wie bei XSUB und DESPOOL. Daher kann UMTAST nicht gleichzeitig mit diesen Programmen betrieben werden.

Der BDOS-Sprungvektor wird geändert, um einem nachfolgenden Programm den korrekten reduzierten Arbeitsspeicher anzuzeigen und ein Überschreiben des Tastaturprogramms zu verhindern.

Nun werden die CBIOS-Sprungvektoren für CONIN und CONST (also Tastatureingabe und Statusabfrage) auf das UMTAST-Programm umgeleitet.

UMTAST

Als letzter Schritt wird untersucht, ob eine Datei nachgeladen werden muß (Siehe unten "Autoload"). Nötigenfalls wird ein kleines Programm in den Pufferbereich bei 80H geschrieben, welches den Ladevorgang startet.

Muß kein Programm geladen werden, dann springt das Tastaturprogramm ohne Warmboot in den CP/M-Eingabeprozessor CCP.

Der nächste Warmboot entfernt automatisch das UMTAST-Programm.

Die Installation

Das anwendungsspezifische Umkodierungsprogramm (z.B. NEU.COM) wird im Bildschirmdialog erstellt. Als Grundlage kann eine Datei verwendet werden, die einer unbelegten Tastatur entspricht oder eine vorher für einen anderen Zweck definierte Datei. Bei der Erstinstallation ist also LEER.COM als Eingabedatei zu verwenden, bei weiteren Installationen kann entweder LEER.COM oder eine vorher installierte Version benutzt werden.

Der Installationsvorgang wird mit folgender Befehlszeile aufgerufen:

UMTAST <Eingabedatei> <Ausgabedatei>

also z.B. UMTAST LEER.COM NEU.COM (Erstinstallation)
oder UMTAST NEU.COM NEU1.COM (Folgeinstallation)

UMTAST

Die Umkodierungsprogramme müssen immer den Extent ".COM" haben, nur so lassen sie sich durch einfachen Aufruf laden und ausführen.

Die Autoload-Funktion

Nach korrektem Aufruf meldet sich UMTAST mit:

UMTAST Installationsprogramm

Filename für Autoload: --- kein Eintrag ---
Neubelegung (j/n):

Der abgefragte Dateiname dient zur Ausnutzung einer optionalen Funktion von UMTAST, dem automatischen Nachladen des zur Umkodiertabelle gehörenden Programms, also z.B. des Editors. Verlangt wird an dieser Stelle die Eingabe des Dateinamens des Editors, z.B. ED.COM.

Gibt man auf die Abfrage "Neubelegung (j/n):" ein "j" ein, dann fragt UMTAST:

Filename ? :

und erwartet die Eingabe eines gültigen Dateinamens nach der CP/M-Konvention. Da die Datei von UMTAST geladen wird, ist es nicht notwendig, den Extent .COM zu verwenden. Diese

UMTAST

Eigenschaft von UMTAST erlaubt es, das Programm WordStar, das üblicherweise WS.COM heißt, umzubenennen in z.B.: WS.UMT. Dann kann das Umkodierungsprogramm wieder WS.COM genannt werden. Mit dem gewohnten Aufruf "WS" wird dann das UMTAST-Programm geladen und lädt WS.UMT hinterher, so daß für den Benutzer kein Unterschied in der Handhabung besteht. Bevor Sie im UMTAST-Installationslauf WS.COM bzw. einen anderen Originaldateinamen als Ausgabedatei spezifizieren, müssen Sie die Originaldatei umbenannt haben, da Sie diese sonst zerstören.

Bei Benutzung der Autoload-Funktion sind zwei Besonderheiten zu beachten. Zunächst die Einschränkung: Es können keine Parameter an das zu ladende Programm übergeben werden. Beispiel: WordStar kann man mit einem Aufruf der Art "WS HUGO.BRF" veranlassen, gleich die Datei HUGO.BRF zu bearbeiten, statt das Hauptmenü zu zeigen. Diese Möglichkeit kann wegen des knappen Platzes für die Autoloadfunktion im CP/M-Buffer (80H) nicht angeboten werden.

Die zweite Besonderheit: Durch Angabe eines Dateinamens als Parameter in der Aufrufzeile kann statt der im Installationslauf vordefinierten die angegebene Datei geladen und ausgeführt werden. Beispiel: Das Umkodierungsprogramm "WS.COM" hat als Autoload-Datei "WS.UMT" definiert. Soll eine spezielle Version des WordStar mit der gleichen Umkodierungstabelle geladen werden, reicht die Eingabe von "WS WSSPEZ.COM", um statt WS.UMT den WSSPEZ.COM zu laden.

UMTAST

Tastenprogrammierung

Nach der Abfrage einer Autoload-Datei folgt die eigentliche Tastenprogrammierung. Der Programmierdialog beginnt mit:

Funktionstaste:

Das nächste Zeichen, das mit der Tastatur erzeugt wird, wird mit seinem hexadezimalen Wert dargestellt, also 71 für ein kleines q, 51 für ein großes Q und 11 für ^Q. Wird statt des Zeichens eine Escape-Sequenz eingegeben, z.B. also <ESC>q, so erscheint die Angabe:

Tastencode: 71 <ESC>

Aktuelle Belegung: --- kein Eintrag ---

Neubelegung (j/n):

Erfolgt auf diese Abfrage die Eingabe j, dann verlangt UMTAST durch Ausgabe von:

Zeichenkette:

die Eingabe der gewünschten Tastenprogrammierung. Im folgenden können dann die gewünschten Zeichen eingegeben werden.

Beispiel: (Benutzereingaben unterstrichen)

Funktionstaste: #

Tastencode: 23

Aktuelle Belegung: --- kein Eintrag ---

Neubelegung (j/n): J

Zeichenkette: ELZET 80^M

UMTAST

Im Beispiel wird die Taste "#", die in deutschen Texten selten gebraucht wird, als Funktionstaste verwendet. UMTAST zeigt zunächst den Wert mit "23" an, dann die aktuelle Belegung. Da die Taste programmiert werden soll, wird auf die Abfrage nach der Neubelegung ein "J" eingegeben. Folglich fragt UMTAST nach einer Zeichenkette, die dann als ELZET 80 eingegeben wird. Im Beispiel folgt ein ^M auf das ELZET 80. was einem <CR> entspricht.

UMTAST verlangt für die Programmierung von Steuerzeichen (Control-Codes) das Drücken von zwei Tasten hintereinander, zuerst ^ . entsprechend 5E Hex, dann den Buchstaben. Die Zeichenkombination ^ und § ist nicht erlaubt, da UMTAST den entsprechenden Hexwert 00 als Begrenzungszeichen verwendet. Soll ESC programmiert werden, sind ^ und Ä (1B Hex) einzugeben.

Die einzige Taste, die nicht umprogrammiert werden kann, ist die CTRL/C-Taste (bzw. die Tastenkombination CTRL und C gleichzeitig). Die Eingabe von CTRL/C bei der Abfrage "Funktionstaste" bewirkt den Abbruch des Installationslaufs und den Aufbau der Ausgabedatei.

Wird von einem Programm die Eingabe von ESC erwartet (Word-Star z.B. als Fehlerquittung), so muß eine Taste mit ESC programmiert werden, da ansonsten ESC als Einleitung für eine Funktionstaste betrachtet wird und UMTAST auf das folgende Zeichen wartet. Es empfiehlt sich, die Kombination ESC ESC mit ESC zu programmieren, dann wird ESC durch zweimaliges Drücken von ESC ausgegeben. Dazu bei der Abfrage

UMTAST

"Funktionstaste" zweimal ESC drücken, dann die Taste "^", dann "Ä" bzw. eckige Klammer links. Wird diese Programmierung nicht vorgenommen, dann gibt Umtast bei ESC ESC auch zweimal ESC weiter, da nicht programmierte Tasten bis auf eine ASCII-Maskierung unverändert übergeben werden.

Änderungen nach Ende des Installationslaufs

Wenn Sie nach dem Test Ihres Umkodierungsprogramms feststellen, daß eine oder mehrere Tasten falsch belegt sind, dann ist eine Folgeinstallation vorzunehmen. Bei einer Folgeinstallation wird eine vorhandene Tabelle verwendet. Dazu ist lediglich LEER.COM als Eingabedatei abzulösen durch die zu ändernde Datei.

Folgeinstallation:

Aufruf: UMTAST <falsche Datei> <neue Datei>

Beispiel: UMTAST NEU.COM NEU1.COM

UMTAST verfährt bei der Folgeinstallation prinzipiell wie bei der Erstinstallation, lediglich die Zeile "Aktuelle Belegung:" zeigt nicht mehr "--- kein Eintrag ---", sondern die im vorherigen Lauf programmierte Zeichenkette.

Im Beispiel auf der nächsten Seite wird die Autoload-Datei WS.COM aus der Altdatei NEU.COM übernommen, # wird von "ELZET 80^M" in "Hugo Gammel" geändert.

UMTAST

Beispiel: (Benutzereingaben unterstrichen)

UMTAST Installationsprogramm

Filename für Autoload: WS.COM

Neubelegung (j/n): n

Funktionstaste: #

Tastencode: 23

Aktuelle Belegung: ELZET 80^M

Neubelegung (j/n): J

Zeichenkette: Hugo Gammel

Funktionstaste:

...usw

Automatisches Laden des Umkodierungsprogramms bei CP/M-Start. (CP/M-Autoload)

CP/M kennt eine Spezialfunktion, bei der nach einem Kalt- oder Warmstart automatisch ein Programm nachgeladen wird. Diese CP/M-Autoload-Funktion, die nicht mit der Autoload-Funktion von UMTAST zu verwechseln ist, wird häufig im kommerziellen Bereich angewandt, wo der Benutzer keinen Zugang zu CP/M haben soll.

UMTAST

Ein UMTAST-Umkodierungsprogramm kann auch als CP/M-Autoload definiert werden. Dann wird die Tabelle bei jedem Kalt- oder Warmstart von CP/M automatisch geladen und lädt, wenn ein UMTAST-Autoload definiert ist, das zugehörige Programm hinterher. Diese Funktion ist nützlich, wenn Standardtasten umdefiniert werden müssen (Z/Y-Tausch o.ä.) oder wenn CP/M-Funktionen damit aufgerufen werden sollen, z.B. Tastenbelegung mit "PIP B:=A:*. *" . Zum Verfahren des CP/M-Autoloads lesen Sie bitte in den CP/M-Unterlagen nach.

Das Programm "WEG.COM" dient dazu, eine UMTAST-Tabelle, die über einen CP/M-Autoload geladen wurde, aus dem Speicher zu entfernen, z.B. um mit DESPOOL oder XSUB arbeiten zu können. Es wird ohne weitere Angaben aufgerufen.

Beispiel: A> WEG

ASCII-Maskierung

Im CP/M-CBIOS wird üblicherweise das 8.Bit des von der Tastatur kommenden Zeichens auf 0 gesetzt, da der ASCII-Code in den unteren 7 Bit Platz hat (128 Zeichen). Dieses Rücksetzen nennt man "ASCII-Maskierung", es wird in der CONIN-Routine vorgenommen, meist durch den Assemblerbefehl "AND A, 7F". Für Tastaturen, die Zeichen >128 (80H) ausgeben können, muß die Maskierung entfernt werden (AND A, FF), wenn UMTAST davon Gebrauch machen soll. Am Günstigsten nehmen Sie die Änderung nach Rücksprache mit ihrem CP/M-Händler durch einen Patch im MOVCPM vor.