

SHARP®

TASCHENCOMPUTER

MODELL
PC-1150

BEDIENUNGSANLEITUNG





EINLEITUNG

Der PC-1150 ist der neue Datenbank-Computer im Pocket-Format von Sharp, der speziell für die Verarbeitung von Geschäftsinformationen entwickelt wurde. Neben den normalen Taschenrechner-Funktionen bietet der PC-1150 eine praktische "Datenbank"- Funktion (für Telefonliste und Tabellen). Weiterhin verfügt der PC-1150 über einen BASIC-Befehlsvorrat, mit dem Sie Geschäftsinformationen auf die gleiche Weise wie mit einem Personal-Computer verarbeiten können.

Bitte lesen Sie diese Bedienungsanleitung sorgfältig durch, um sich mit der Bedienung vertraut zu machen. Heben Sie diese Bedienungsanleitung für zukünftiges Nachschlagen gut auf.

ZUR BEACHTUNG:

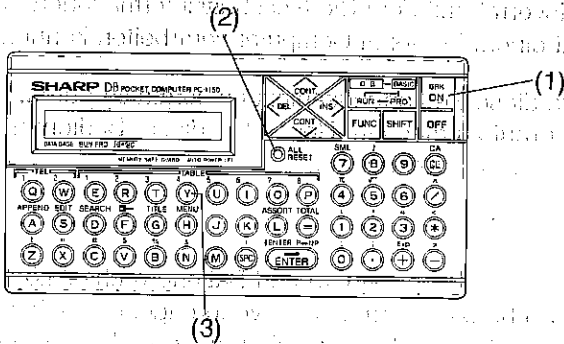
Bei Beschädigungen, Reparaturen und Batteriewechsel können alle im Rechner gespeicherten Informationen verlorengehen. Daher sollten wertvolle Daten auf Kasette gespeichert oder auf Papier notiert werden.

Inbetriebnahme

SHARP

Der Speicher des Rechners wird ab Werk gelöscht. Vor der ersten Verwendung des Rechners sollten Sie jedoch die folgende Bedienung durchführen, um sicherzugehen, daß der interne Speicher gelöscht ist.

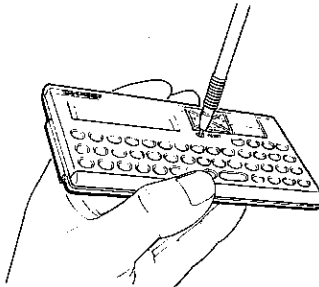
(1) Drücken Sie die Taste **BRK ON**.



Es wird angezeigt:

TEL1 ~ 2, TABLE 1 ~ 8 ?
DATA BASE

(2) Drücken Sie den ALL RESET-Knopf.



Auf dem Display wird angezeigt

MEMORY CLEAR OK ?
DATA BASE

(3) Drücken Sie die Taste **Y**.

Damit ist der Speicher des Rechners gelöscht, und die Meldung aus Schritt (1) erscheint wieder auf dem Display.

INHALTSVERZEICHNIS

Seite

KAPITEL 1	VERWENDUNG DIESER BEDIENUNGSANLEITUNG	5
KAPITEL 2	GRUNDSÄTZLICHE BEDIENUNG	7
	Vor der ersten Verwendung	7
	Darstellung der Tasten in dieser Bedienungsanleitung	8
	Einschalten	10
	Umschalten von einem Modus in einen anderen	10
	Ausschalt-Automatik	10
	Kontrastregelung	11
	ALL RESET-Knopf	11
	Symbole auf dem Display	12
KAPITEL 3	DATEN-EINGABE	14
	Eingabe von Zahlen, Buchstaben und Symbolen	14
	Korrektur der eingegebenen Daten	15
	Abrufen von versteckten Zeichen	17
	Anzahl der Eingabezeichen	18
KAPITEL 4	TELEFONLISTE-MODUS	19
	Eingabe von Name und Telefonnummer	19
	Lesen der Telefonliste	21
	Korrektur von Eintragungen	24
	Anhängen einer neuen Eintragung	25
	Löschen von Informationen	26
KAPITEL 5	TABELLEN-MODUS	28
	Eingabe von Daten	28
	Lesen der Daten	32
	Korrektur von Eintragungen	35
	Anhängen von Eintragungen	35
	Einfügen von Eintragungen	36
	Löschen von Eintragungen	37
	Reihen/Spalten-Summenbildung	38
	Summenbildung nach Posten	39
KAPITEL 6	GESCHÜTZTE DATEN UND DATEN-SPEICHERUNG	41
	Spezifizierung von Daten als geschützt oder nicht geschützt	41
	Sichern / Laden von Daten auf / von Kassette	42

	Seite
KAPITEL 7 RECHENFUNKTIONEN	44
Grundrechenarten	44
Korrektur von Tasteneingaben	45
Wiederabruf-Funktion	46
Anzeige des Berechnungsergebnisses	47
Speicher und Variable	48
KAPITEL 8 PROGRAMMIERUNG	51
Zeilen	51
Zeilennummern	51
Eingabetaste ENTER	51
Eintasten-Ausführung	52
Programmgröße	52
Eingabe und Ausführung von Programmen	52
KAPITEL 9 BASIC-REFERENZ	56
KOMMANDOS	58
BEFEHLE	67
FUNKTIONEN	87
KAPITEL 10 FEHLERSUCHE	93
Bedienung des Geräts	93
KAPITEL 11 PFLEGE DES COMPUTERS	94
ANHANG	
Anhang A : Fehlermeldungen	95
Anhang B : Auswechseln der Batterien	97
Anhang C : Erweiterte Funktion der Telefonliste	100
Anhang D : Beziehung zwischen den Daten in der Datenbank und den Basic- Feldvariablen	102
Anhang E : Numerische Genauigkeit des Computers	104
Anhang F : Zeichencode-Tabelle	106
Anhang G : Verwendung von programmen, die für andere PC-Modelle geschrieben wurden	108
Anhang H : Technische Daten	109
Anhang I : Beispielsprogramme	111

KAPITEL 1 VERWENDUNG DIESER BEDIENUNGSANLEITUNG

Diese Bedienungsanleitung ist als Einführung in die Funktionen und Möglichkeiten des Computers und als Nachschlagewerk konzipiert. Ob Sie ein "Anfänger" oder ein "alter Computerhase" sind, Sie sollten die Kapitel 2 bis 7 durcharbeiten, um sich mit dem Computer vertraut zu machen.

- * Kapitel 2 enthält eine Übersicht der Grundfunktionen.
- * Kapitel 3 erläutert die Tastenbetätigungen.
- * Kapitel 4 behandelt den Telefonliste-Modus.
- * Kapitel 5 erklärt den Tabellen-Modus.
- * Kapitel 6 beschreibt, wie der Kennwort-Schutz für die Datenbank gesetzt wird.
- * Kapitel 7 behandelt die Verwendung des Computers als Taschenrechner.

Die Kapitel 8 und 9 behandeln die Implementierung der Programmiersprache BASIC auf diesem Computer. Da sich alle BASIC-Dialekte geringfügig unterscheiden, sollten auch erfahrene BASIC-Programmierer diese Kapitel durchlesen, bevor Programmierarbeiten in Angriff genommen werden.

Kapitel 9 enthält alle BASIC-Befehle und -Funktionen in alphabetischer Reihenfolge innerhalb der Gruppen.

Falls Sie noch nicht in der BASIC-Programmierung geübt sind, empfehlen wir, ein Einführungswerk für BASIC zu erwerben oder einen entsprechenden Kursus zu besuchen, bevor Sie diese Kapitel durcharbeiten. Diese Bedienungsanleitung ist nicht als Einführung in die Programmierung konzipiert.

Die verbleibenden Kapitel sind:

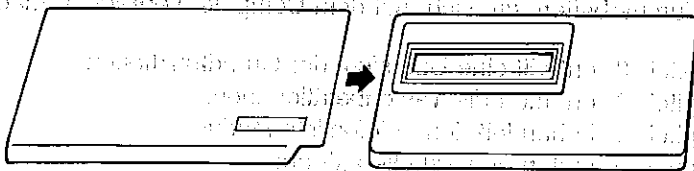
- * Kapitel 10 - Beschreibt Maßnahmen, wenn Probleme bei der Verwendung und Programmierung auftreten.
- * Kapitel 11 - Die Pflege des Computers.

Der ausführliche Anhangteil am Ende der Bedienungsanleitung enthält Tabellen und detaillierte Ausführungen zum Arbeiten mit dem Computer.

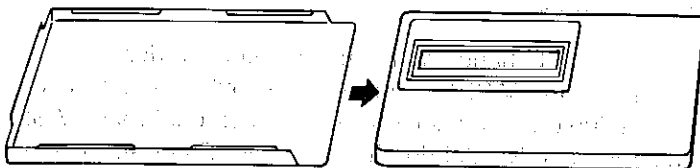
Verwendung der Tastaturabdeckung

Wenn der Computer nicht verwendet wird, sollte die Abdeckung über die Tastatur geschoben werden.

• Bei Nichtverwendung



• Bei Verwendung

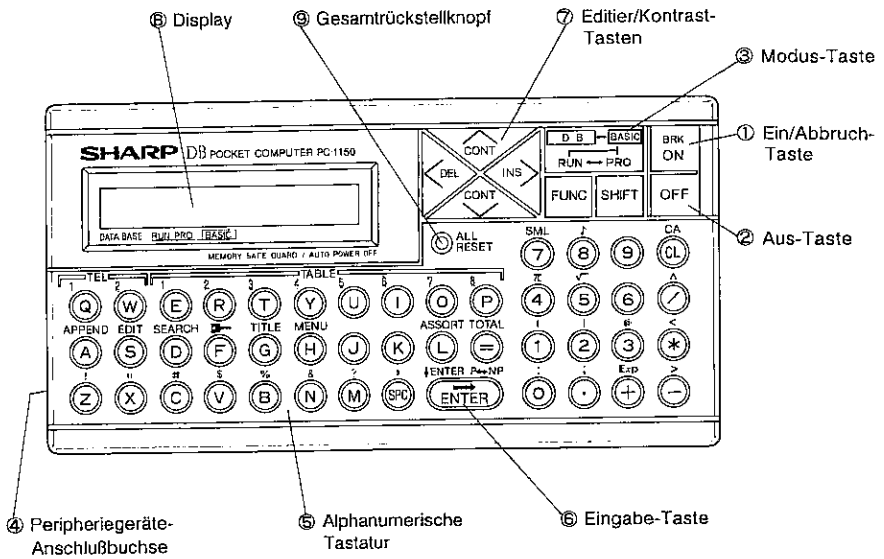


KAPITEL 2

GRUNDSÄTZLICHE BEDIENUNG

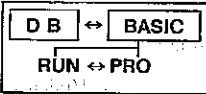

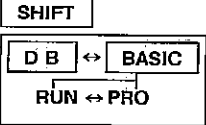
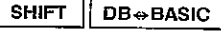




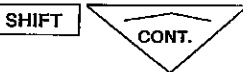
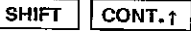



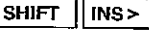


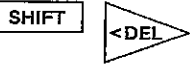



Vor der ersten Verwendung

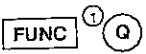




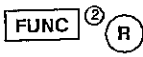



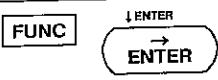




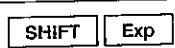
Machen Sie sich zunächst mit der Anordnung der Tasten und einigen Funktionen des Computers vertraut.



Darstellung der Tasten in dieser Bedienungsanleitung

Die Betätigung der Tasten wird in dieser Bedienungsanleitung wie folgt dargestellt. Wenn die Betätigung von zwei Tasten erforderlich ist, drücken Sie zunächst die erste Taste und geben sie wieder frei, bevor Sie die zweite Taste drücken. Betätigen Sie die beiden Tasten nicht gleichzeitig.



Tastenbetätigung	Darstellung	Beschreibung
		Umschalten zwischen RUN - und PRO - Modus
		Umschalten zwischen DATA BASE- (Telefonliste, Tabellen) und RUN-Modus
		Einschalten des Computers bzw. Unterbrechen der Programmausführung
		Bewegung des Cursors eine Zeile nach oben
		Erhöhung des Display-Kontrastes
		Bewegung des Cursors um eine Stelle nach rechts
		Erstellung einer Leerstelle für Einfügung eines Zeichens
		Bewegung des Cursors um eine Stelle nach links
		Löschung des Zeichens, auf dem der Cursor steht
		Spezifizierung von Tel 1 (nur bei Datenbank-Menü)

Key Operation	Notation	Description
		Spezifizierung von Tel 1 (außer bei Datenbank-Menü)
		Ausführung von Funktionstasten (nur im RUN-Modus)
		Spezifizierung von Tabelle 2 (nur bei Datenbank-Menü)
		Spezifizierung von Tabelle 2 (außer bei Datenbank-Menü)
		Dateneingabe
		Bewegung des Cursors eine Reihe nach unten nach Eingabe von Daten im Telefonliste- oder Tabellen-Modus
		Umschalten zwischen PRINT-Modus (Drucken) und NON-PRINT-Modus
		Spezifiziert das Exponentensymbol (E)

Für alphanumerische Zeichen und Symbole, die auf oder oberhalb der Tasten angegeben sind, werden in dieser Bedienungsanleitung Klammern ausgelassen.

Auf dem Display wird die Ziffer Null (0) als Ø angezeigt, um sie vom Großbuchstaben O unterscheiden zu können.

Tongenerator

Der PC-1150 erzeugt bei jeder Tastenbetätigung einen Signalton. Durch Drücken von **SHIFT**  kann dieser Signalton ausgeschaltet und durch erneutes Drücken von **SHIFT**  oder durch Drücken von **OFF** **ON** wieder aktiviert werden.

Einschalten

Der Rechner wird durch Drücken von **ON** eingeschaltet. Drücken Sie **TEL1** (Taste **Q**), anschließend einige Zahlentasten unter den Tasten **FUNC** und **SHIFT**, zum Beispiel:

1 **2** **3**

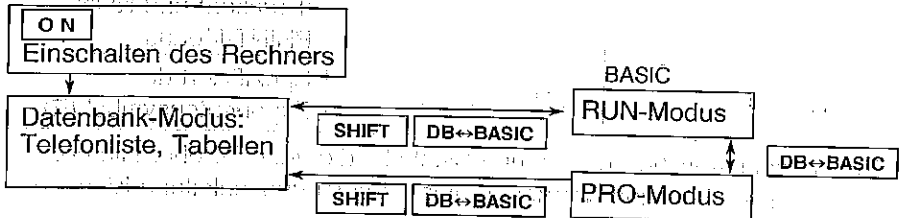
TEL1 TITLE = 123 _

DATA BASE

Die eingegebenen Zahlen erscheinen auf dem Display, gefolgt vom Aufforderungszeichen:

Umschalten von einem Modus in einen anderen

Der PC-1150 hat drei Grundmodi: der Datenbank-Modus zur Speicherung von Telefonliste oder Tabellen, der PRO-Modus zum Schreiben von BASIC-Programmen und der RUN-Modus zur Ausführung manueller Berechnungen oder Programme. Die Umschaltung von einem Modus zu einem anderen wird durch die folgenden Tastenbetätigungen durchgeführt:



Ausschalt-Automatik

Wenn etwa 11 Minuten lang keine Tastenbetätigung erfolgt, schaltet der Rechner automatisch das Display aus, um die Batterien zu schonen. Wird anschließend die Taste **ON** gedrückt, kehrt der Rechner in den Anfangsstatus zurück.

Kontrastregelung

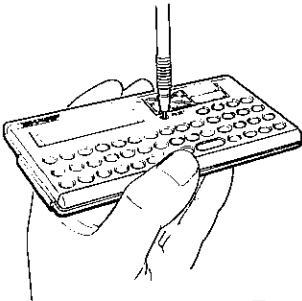
Zur Einstellung des Display-Kontrastes dienen die Tasten **CONT.↑** und **CONT.↓**. Durch Betätigung von **SHIFT** **CONT.↑** wird die Anzeige dunkler, durch Betätigung von **SHIFT** **CONT.↓** heller. Wird **CONT.↑** oder **CONT.↓** nach Drücken von **SHIFT** gedrückt gehalten, wird die Anzeige kontinuierlich dunkler bzw. heller.

Cursor

Die Symbole **_** und **■** auf dem Display werden als Cursor bezeichnet. Der Unterstrich-Cursor (**_**) kennzeichnet die Stelle, wo das nächste Zeichen eingegeben wird. Durch den blinkenden quadratischen Cursor (**■**) wird die Stelle gekennzeichnet, wo ein Zeichen editiert (korrigiert, gelöscht oder eingefügt) werden kann.

ALL RESET-Knopf

Wenn der Rechner einem harten Stoß ausgesetzt wird, kann ein Zustand eintreten, in dem keine der Tasten einschließlich der Taste **ON** wirksam ist. Sollte dies vorkommen, drücken Sie die Taste **ON** und halten dann den ALL RESET-Knopf für einige Sekunden gedrückt. Die folgende Meldung erscheint:



ZUR BEACHTUNG:

Verwenden Sie zum Drücken des ALL RESET-Knopfes einen Kugelschreiber oder einen ähnlichen Gegenstand. Verwenden Sie keinen Gegenstand, dessen Spitze scharf ist oder abbrechen kann.

MEMORY CLEAR OK ?

DATA BASE

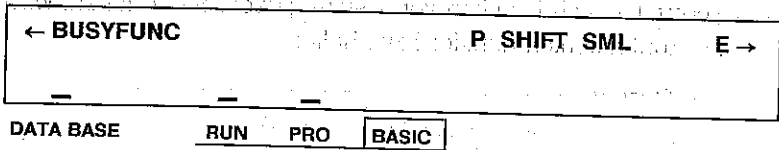
Wenn Sie jetzt die Taste **Y** drücken, werden alle Daten und Programme im Speicher des Rechners gelöscht. Durch Drücken der Taste **N** erscheint die folgende Meldung, dann erfolgt keine Löschung der Daten und Programme:

TEL 1 ~ 2, TABLE 1 ~ 8 ?

Wenn nach Durchführung der obigen Rückstellung das gleiche Problem erneut auftritt, drücken Sie den ALL RESET-Knopf erneut und dann die Taste **Y**, um den Speicher zu löschen. Dann müssen Sie alle Daten und/oder Programme erneut eingeben.

Symbole auf dem Display

Die auf dem Display erscheinenden Symbole und Status-Anzeigen haben die folgenden Bedeutungen:



- ←: Dieser Pfeil bedeutet, daß links von den angezeigten Daten noch weitere Daten vorhanden sind. Diese versteckten Daten können durch Drücken der Taste **<DEL** angezeigt werden (wirksam im DATA BASE-Modus).
- BUSY: Der Rechner ist mit Berechnungen befaßt.
- FUNC: Die Taste **FUNC** wurde gedrückt. Diese Anzeige erlischt, wenn die Taste **FUNC** erneut gedrückt wird.
- P: Der Rechner befindet sich im PRINT-Modus; angewählt durch **SHIFT** **P ↔ NP** im BASIC-Modus.
- SHIFT: Die Taste **SHIFT** wurde gedrückt. Diese Anzeige erlischt, wenn die Taste **SHIFT** erneut gedrückt wird.
- SML: Der Rechner befindet sich im SML-Modus (Kleinbuchstaben-Modus), angewählt durch **SHIFT** **SML**. Diese Anzeige erlischt, wenn **SHIFT** **SML** erneut gedrückt wird.
- E: Ein Fehler ist eingetreten. Zum Löschen des Fehlerzustandes die Taste **CL** drücken.
- : Dieser Pfeil bedeutet, daß rechts von den angezeigten Daten noch weitere Daten vorhanden sind. Diese versteckten Daten können durch Drücken der Taste **INS>** angezeigt werden (wirksam im DATA BASE-Modus).
- DATA BASE: Der Rechner befindet sich im DATA BASE-Modus. Wenn ein Überstrich () mit der RUN- oder PRO-Anzeige erscheint, **SHIFT** **DB ↔ BASIC** betätigen.

RUN: Der Rechner befindet sich im RUN-Modus. Durch Betätigung von **DB↔BASIC** wird in den PRO-Modus und durch Betätigung von **SHIFT** **DB↔BASIC** in den DATA BASE-Modus umgeschaltet.

PRO: Der Rechner befindet sich im PRO-Modus. Durch Betätigung von **DB↔BASIC** wird in den RUN-Modus und durch Betätigung von **SHIFT** **DB↔BASIC** in den DATA BASE-Modus umgeschaltet.

Hinweis: In dieser Bedienungsanleitung werden diese Display-Symbole und Status-Anzeigen nur angegeben, wenn sie zur Erklärung erforderlich sind.

KAPITEL 3

DATEN-EINGABE

In diesem Kapitel wird erklärt, wie Zeichen und Zahlen in den Rechner eingegeben werden.

Drücken Sie zuerst die Taste **OFF** und dann die Taste **ON**. Da wir die Eingabe anhand der Telefonliste üben wollen, drücken Sie **TEL1** (Taste **Q**).

OFF **ON** **TEL1**

TEL1 TITLE = ?
DATA BASE

Eingabe von Zahlen, Buchstaben und Symbolen

Auf dem Display erscheint die Aufforderung "TEL1 TITLE = ?". Geben Sie zuerst einige Zahlen ein:

1 **2** **3**

TEL1 TITLE = 123_

Die Zahlen "123" erscheint unmittelbar hinter der Aufforderung auf dem Display. Geben Sie jetzt einige Großbuchstaben ein:

A **B** **C**

L1 TITLE = 123ABC_

Bei der Eingabe wird die Meldung nach links aus dem Display verschoben. Lassen Sie sich dadurch nicht verwirren.

Zur Eingabe von Kleinbuchstaben müssen Sie zunächst **SHIFT** **SML** betätigen, um den SML-Modus anzuwählen (die Anzeige **SML** erscheint auf dem Display). Drücken Sie dann die gewünschten Zeichentasten:

SHIFT **SML** **D** **E** **F**

TITLE = 123ABCdef_ ^{SML}

Um wieder Großbuchstaben einzugeben, drücken Sie **SHIFT** **SML** erneut (die Anzeige **SML** verschwindet).

Zur Eingabe von Symbolen wie "@" oder "?" drücken Sie zuerst die Taste **SHIFT** und dann die entsprechende Symboltaste.

- * Durch einmaliges Drücken der Taste **SHIFT** wird der Rechner in den SHIFT-Modus versetzt, dann leuchtet die Anzeige **SHIFT**. Durch erneutes Drücken der Taste wird der SHIFT-Modus verlassen, die Anzeige **SHIFT** erlischt.

SHIFT **@** **SHIFT** **?**

T L E = 1 2 3 A B C d e f @ ? _

Drücken Sie jetzt die Taste **CL**. Dadurch werden die bislang eingegebenen Daten gelöscht, auf dem Display erscheint wieder die anfängliche Meldung:

CL

TEL1 TITLE = ?

- * Die Taste **CL** dient zum Löschen aller eingegebenen Daten und Zurücksetzen des Displays in den Anfangsstatus.

Wir wollen die Eingabe der "TELEPHONE"-Daten erneut beginnen:

T **E** **L** **E** **P** **H** **O** **N** **E**

TITLE = TELEPHONE _

ENTER

NAME = ?

Zum Speichern der eingegebenen Informationen in den Rechner müssen Sie die Taste **ENTER** drücken. Nach Drücken der Taste **ENTER** erscheint die nächste Aufforderung auf dem Display (NAME = ?).

Korrektur der eingegebenen Daten

Dieser Abschnitt beschreibt, wie bereits eingegebene Daten korrigiert werden. Dafür stehen drei Möglichkeiten zur Verfügung: Überschreiben, Löschen und Einfügen.

Überschreiben von Daten

Geben Sie auf die Aufforderung NAME den Namen "James" ein.

J **SHIFT** **SML** (Wahl des SML-Modus.)

A **M** **E** **S**

NAME = J a m e s _

Wenn Sie den Namen in "Jackson" ändern wollen, bringen Sie den Cursor mit der Taste **<DEL** auf den Buchstaben m und überschreiben dann die restlichen Zeichen.

- * Während der Datenkorrektur erscheint ein blinkender quadratischer Cursor.

NAME = James _

C K S O N

NAME = Jackson _

Löschen von Daten

Wenn Sie den Namen "Jackson" zu "Jason" ändern wollen, können Sie einfach das "ck" löschen. Bringen Sie den Cursor auf das zu löschende Zeichen (c oder k) und drücken dann **SHIFT** ****. Dadurch wird das Zeichen, auf dem der Cursor steht, gelöscht.

NAME = Jackson

SHIFT SHIFT

NAME = Jason

Der Cursor wird mit den Tasten **** und **INS>** bewegt. Zum Löschen von Zeichen drücken Sie zuerst die Taste **SHIFT** und dann ****.

INS> **INS>** **INS>** (bringt den Cursor nach rechts.)

NAME = Jason _

Einfügen von Daten

Wenn Sie den Namen "Jason" zu "Jefferson" ändern wollen, müssen Sie den Buchstaben "a" zu "e" ändern und die vier Buchstaben "ffer" vor dem "s" einfügen. Bringen Sie zunächst den Cursor mit der Taste **** auf den Buchstaben "a" und geben ein "e" ein

NAME = J~~a~~son

E

NAME = Je~~a~~son

Verwenden Sie dann **SHIFT** **INS>**, um Leerstellen für die benötigten Buchstaben zu schaffen, und geben Sie dann die Buchstaben in diese Leerstellen ein. Bringen Sie zuerst den Cursor auf den Buchstaben "s" (in diesem Beispiel ist dies nicht erforderlich, da der Cursor sich bereits auf dem Buchstaben "s" befindet);

NAME = Je~~a~~son

Schaffen Sie vier Leerstellen (□) für die einzufügenden Buchstaben:

SHIFT INS> SHIFT INS>
SHIFT INS> SHIFT INS>

NAME = Je □□□□ son

Geben Sie dann "ffer" in diese Leerstellen ein:

F F E R

NAME = Jefferson

Nach Durchführung aller Änderungen drücken Sie **SHIFT** **SML**, um den SML-Modus zu verlassen (die Anzeige SML erlischt).

Drücken Sie dann die Taste **ENTER**, um die eingegebenen Daten im Rechner zu speichern. Der Rechner stellt die nächste Frage:

ENTER

No. = ?

Abrufen von versteckten Zeichen

Geben Sie auf die Aufforderung No. "12345678901" ein:

1 2 3 4 5 6 7 8 9
0 1

No. = 12345678901 _

Damit ist das Display voll. Geben Sie weiter "234" ein. Bei der Eingabe werden die Zeichen nach links aus dem Display verschoben:

2 3 4

= 12345678901234 _

Mit der Taste **<DEL** können Sie die versteckten Zeichen, die nach links aus dem Display verschoben wurden, wieder zurückrufen:

<DEL

. = 12345678901234

<DEL

o. = 1234567890123

<DEL

No. = 123456789012

Dadurch wurden andere Zeichen nach rechts aus dem Display verschoben. Diese Zeichen können Sie mit der Taste **INS>** wieder in das Display zurückbringen.

INS> **INS>** **INS>**

= 1 2 3 4 5 6 7 8 9 0 1 2 3 4 _

Anzahl der Eingabezeichen

Geben Sie hinter den oben eingegebenen Zeichen "567" ein:

5 **6** **7**

2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7

Bei der Eingabe von "7" fängt der Cursor an zu blinken. Das bedeutet, daß Sie zu viele Zeichen eingegeben haben. Der Rechner nimmt das Zeichen, auf dem der Cursor blinkt und nachfolgende Zeichen nicht an.

Für jede Eintragung können Sie bis zu 16 Zeichen eingeben, das 17. und nachfolgende Zeichen werden ignoriert.

Wenn Sie am blinkenden Cursor Zeichen eingeben, wird das vorige durch das neue Zeichen ersetzt.

KAPITEL 4

TELEFONLISTE-MODUS

Der Rechner verfügt über zwei Telefonlisten: TEL1 und TEL2. Jede Liste ist in einen Abschnitt für Namen und Telefonnummern aufgeteilt. Die Namen in jeder Telefonliste werden alphabetisch geordnet (für die Reihenfolge der Zeichencodes siehe Seite 107).

In den Namen-Abschnitt können Sie bis zu 16 Zeichen für Name und Adresse eingeben.

- * Wenn Sie mehr als 16 Zeichen eingeben, ignoriert der Rechner das 17. und alle weiteren Zeichen.

Eingabe von Name und Telefonnummer

Als Beispiel wollen wir die folgende Liste von Namen und Telefonnummern verwenden:

(Beispiel)

TITLE	Business
NAME	No.
Y. SAITOO/SHARP	03-260-1161/315
M. JONES	212-758-0354
J. SMITH/LONDON	01-643-4431
M. BERRY	212-432-0012

Drücken Sie die Tasten **CL** **FUNC** und dann die Taste **MENU** (H), um das Menü auf das Display zu rufen (Sie können auch **OFF** **ON** drücken, um das Menü abzurufen):

CL **FUNC** **MENU**

TEL1 ~2, TABLE 1 ~ 8 ?

Wenn Sie die Taste **TEL1** drücken, erscheint der Titel "TELEPHONE", den Sie im vorigen Kapitel 3 bereits eingegeben haben:

TEL1

TEL1 TITLE = TELEP →

Um diesen Titel zu "Business" zu ändern, müssen Sie zuerst den vorigen Titel löschen. Die bereits im Computer gespeicherten Daten können nicht durch Drücken der Taste **CL** gelöscht werden. Führen Sie die folgenden Tastenbetätigungen durch:

SHIFT <DEL

DELETE OK?(Y/N)

Y

DELETED

TEL1 TITLE = ?

(Wenn Sie die vorigen Informationen löschen wollen, drücken Sie Y, sonst N.)

Geben Sie jetzt den neuen Titel (Business) ein:

B SHIFT SML (die Anzeige SML leuchtet)

U S I N E S S

TITLE = Business _

ENTER

NAME = ?

Jetzt erscheint die Meldung "NAME = ?" als Aufforderung, einen Namen einzugeben. Geben Sie "Y. SAITOO/SHARP" ein. In den folgenden Beschreibungen werden rechteckige Klammern für einzugebende Zeichen und Ziffern ausgelassen.

SHIFT SML (die Anzeige SML erlischt)

Y. SAITOO

NAME = Y. SAITOO _

/SHARP

= Y. SAITOO/SHARP _

Sie können bis zu 16 Zeichen für den Namen und andere relevante Informationen in der Telefonliste speichern. Drücken Sie die Taste

ENTER :

ENTER

No. = ?

Dann erscheint die Meldung "No. = ?" als Aufforderung, die Telefonnummer einzugeben.

Geben Sie die Telefonnummer ein:

03-260-1161

No. = 03-260-1161 _

Zum Speichern der Telefonnummer drücken Sie erneut **ENTER**. Der Name und die Telefonnummer werden als eine Einheit gespeichert:

ENTER

NAME = ?

Diese Aufforderung bedeutet, daß Sie einen weiteren Namen eingeben können.

Geben Sie den Rest der Telefonliste auf die gleiche Weise ein:

M. JONES

NAME = M. JONES _

ENTER

No. = ?

212-758-0354 **ENTER**

NAME = ?

J. SMITH/LONDON **ENTER**

No. = ?

01-643-4431 **ENTER**

NAME = ?

M. BERRY **ENTER**

No. = ?

212-432-0012 **ENTER**

NAME = ?

Nach der Eingabe werden die eingegebenen Namen automatisch in alphabetischer Reihenfolge sortiert.

Lesen der Telefonliste

Zum Lesen der Telefonliste gibt es zwei Möglichkeiten: sequentielle Suche, bei der die Namen in alphabetischer Reihenfolge abgerufen werden, und direkte Suche, bei der die Namen direkt ausgewählt werden können.

Sequentielle Suche

Für sequentielle Suche betätigen Sie die Taste **CONT.↑** oder **CONT.↓**, bis Sie den gewünschten Namen erreichen.

- Wenn Sie **FUNC** **CONT.↑** betätigen, während TITLE, NAME oder No. angezeigt wird, wird der Titel angezeigt. Durch Betätigung von **FUNC** **CONT.↓** erscheint der letzte Name in der Telefonliste auf dem Display. Der Rechner behandelt den Titel als Daten, die vor dem ersten Namen kommen.

Rufen Sie zuerst den Titel auf das Display:

FUNC **CONT.↑** (Wenn das Menü angezeigt wird, drücken Sie **TEL1** .)

CONT.↓

INS>

INS>

CONT.↓

CONT.↓

CONT.↓

TEL1 TITLE = Busin →

NAME = J. SMITH/LON →

← E = J. SMITH/LONDON

No. = 01-643-4431

NAME = M. BERRY

NAME = M. JONES

NAME = Y. SAITOO/SH →

Wenn Sie weiter als die gewünschte Zeile gehen, können Sie mit der Taste **CONT.↑** zurückgehen:

NAME = M. JONES

- Durch Gedrückthalten der Tasten **CONT.↑** und **CONT.↓** ist kontinuierliche Datensuche möglich.

Wenn bei Drücken von **CONT.↑** oder **CONT.↓** keine Veränderung der Anzeige erfolgt, befinden Sie sich am Titel oder am letzten Namen der Telefonliste.

CONT.↑ (keine Veränderung)

TEL1 TITLE = Busin →

(Titel erscheint)

CONT.↓ (keine Veränderung)

NAME = Y. SAITOO/SH →

(letzte Eintragung)

Ein Pfeil (→) oben rechts auf dem Display bedeutet, daß rechts von den angezeigten Daten weitere Daten vorhanden sind. Diese können mit der Taste **INS>** auf das Display gerufen werden:

INS>

NAME = Y. SAITOO/SH →

←
E = Y. SAITOO/SHARP

(Ein Pfeil (←) oben links auf dem Display bedeutet, daß links von den angezeigten Daten weitere Daten vorhanden sind. Diese können mit der Taste **<DEL** auf das Display gerufen werden.)

Drücken Sie die Taste **INS>** erneut, um die Telefonnummer anzuzeigen:

INS>

No. = 03-260-1161/ →

INS>

←
= 03-260-1161/315

- Wenn Sie den Namen erneut sehen wollen, der nach links aus dem Display verschoben wurde, verwenden Sie die Taste **<DEL** .

Direkte Suche

Durch Eingabe des oder der ersten Zeichen eines Namens kann der Name direkt abgerufen werden:

FUNC **SEARCH**

SEARCH WORD = ?

Geben Sie "M. J" ein, um den Namen M. JONES zu suchen:

M. J

SEARCH WORD = M. J _

ENTER

NAME = M. JONES

Wenn mehr als ein Name vorhanden ist, der mit "M. J" beginnt, wird der erstere in der alphabetischen Reihenfolge abgerufen.

Wenn der angezeigte Name nicht der gewünschte ist, drücken Sie die Taste **CONT.↓**, bis der richtige Name erscheint. Als nächsten sollen die Namen gesucht werden, die mit dem Buchstaben "M" beginnen:

FUNC SEARCH

M ENTER

NAME = M. BERRY

CONT. ↓

NAME = M. JONES

CONT. ↓

NOT FOUND

NAME = M. JONES

Wenn in der Telefonliste kein weiterer Name enthalten ist, der mit "M" beginnt, wird für etwa eine Sekunde "NOT FOUND" angezeigt; dann erscheint wieder der zuletzt gefundene Name.

- Wird während der direkten Suche **FUNC** **CONT. ↓** gedrückt, erscheint der zuletzt gefundene Name auf dem Display, durch Drücken von **FUNC** **CONT. ↑** wird der zuerst gefundene Name angezeigt.

Zum Beenden der Suche drücken Sie eine beliebige Taste außer **CONT. ↑** und **CONT. ↓**.

Korrektur von Eintragungen

Beispiel 1: Ändern der Nebenstellen-Nummer von Y. SAITOO von 315 zu 299.

Rufen Sie zuerst die Eintragung für Y. SAITOO durch die folgende Eingabe ab:

FUNC SEARCH

SEARCH WORD = ?

Y. S ENTER

NAME = Y. SAITOO/SH

Rufen Sie die Telefonnummer mit der Taste **INS>** auf das Display und wählen dann die EDIT-Funktion:

INS> INS>

No. = 03-260-1161/

FUNC EDIT

No. = 03-260-1161/

Bringen Sie den Cursor auf die zu ändernde Nummer und geben die neue Nummer ein:

INS> ... (12 mal) ... INS> 299

03-260-1161/299_

Nach der Änderung müssen Sie **ENTER** drücken, um die Änderung im Rechner zu speichern.

Auf dem Display erscheint wieder "NAME = ?".

Wenn Sie während der Korrektur die Taste **CONT.1**, **CONT.↓** oder **CL** drücken, wird der Korrektur-Modus verlassen und zum vorigen Modus zurückgekehrt.

Beispiel 2: Anhängen von "Designer" an die Eintragung für M. BERRY
Rufen Sie zunächst die Eintragung für M. BERRY ab und wählen dann die EDIT-Funktion:

FUNC **SEARCH**
M . B **ENTER**

NAME = M . BERRY

FUNC **EDIT**

NAME = M . BERRY

Bringen Sie den Cursor in die Position, wo die neue Information angehängt werden soll, und geben die Information dann ein:

INS> ... (7 mal) ... **INS>**

NAME = M . BERRY _

Zur Eingabe von Kleinbuchstaben brücken Sie **SHIFT** **SML**, so daß **SML** auf dem Display erscheint.

/ D **SHIFT** **SML** ESIGNER

. BERRY / Designer _

SHIFT **SML**

Zum Speichern der Änderung müssen Sie **ENTER** drücken:

ENTER

No. = 212-432-0012

Anhängen einer neuen Eintragung

Beispiel: Anhängen des Namens "P. PETERS" und der Telefonnummer "01-433-0056" an die Telefonliste.

Wählen Sie zunächst die APPEND-Funktion:

FUNC **APPEND**

NAME = ?

Geben Sie dann den Namen und die Telefonnummer auf die gleiche Weise wie bei der Eingabe der Telefonliste ein:

P. PETERS

NAME = P. PETERS _

ENTER

No. = ?

01-433-0056

No. = 01-433-0056 _

ENTER

NAME = ?

Wenn bei Anzeige eines Namens oder einer Telefonnummer der Cursor nicht auf dem Display ist, wird durch Eingabe eines beliebigen Zeichens der Computer in den APPEND-Modus versetzt. Wenn Sie beispielsweise, während "P. PETERS" angezeigt wird, "N. ITO" eingeben, wird "N. ITO" an das Ende der Telefonliste angehängt, ohne daß eine Änderung der Eintragung für "P. PETERS" erfolgt.

N. ITO

NAME = P. PETERS

NAME = N. ITO _

Löschen von Informationen

Wenn der Speicher voll ist und kein Raum für eine neue Eintragung (einschließlich der Nummer) mehr vorhanden ist, erscheint die Meldung "MEMORY FULL" auf dem Display, wenn die Taste **ENTER** nach Eingabe eines neuen Namens gedrückt wird. Zum Löschen einer nicht mehr benötigten Eintragung, beispielsweise die für Y. SATOO, müssen Sie zuerst diese Eintragung abrufen und dann **SHIFT** **<DEL** drücken:

FUNC **SEARCH**

Y. S **ENTER**

NAME = Y. SAITOO / SH →

SHIFT **<DEL**

DELETE OK? (Y/N)

Auf dem Display erscheint eine Abfrage, ob Löschung erfolgen soll. Wenn Sie die Eintragung löschen wollen, drücken Sie **Y**:

Y

NAME = P. PETERS

Nach Löschen der Eintragung erscheint die nächste Eintragung auf dem Display. Ist keine weitere Eintragung vorhanden, wird die vorige Eintragung angezeigt. Wenn überhaupt keine Eintragung vorhanden ist, wird der Titel angezeigt.

Wenn Sie auf die Abfrage, ob Löschung erfolgen soll, eine andere Taste als Y drücken, wird die Löschung nicht durchgeführt, und die Anzeige vor der Wahl des Lösch-Modus erscheint wieder.

Nur die jeweils angezeigte Eintragung wird gelöscht, die anderen Eintragungen werden dadurch nicht verändert.

Zum Löschen aller Eintragungen in der Telefonliste rufen Sie den Titel auf das Display ab und drücken SHIFT :

FUNC TITLE

TEL1 TITLE = Busin →

SHIFT

DELETE OK? (Y/N)

Wenn Sie auf die obige Abfrage, ob die Löschung durchgeführt werden soll, die Taste Y drücken, wird der gesamte Inhalt der Telefonliste TEL1 gelöscht. Auf dem Display wird dann für etwa eine Sekunde "DELETED" angezeigt, anschließend erscheint wieder die Aufforderung zur Titel-Eingabe.

Y

DELETED

TEL1 TITLE = ?

Durch Drücken einer anderen Taste als Y erfolgt keine Löschung, dann wird wieder der Titel angezeigt.

KAPITEL 5

TABELLEN-MODUS

Der Tabellen-Modus bietet eine effiziente, einfache Möglichkeit zur Tabellierung von Geschäftsdaten und Berechnung der Reihen- und Spaltensummen. Zwei Arten der Berechnung stehen zur Verfügung:
 Reihen-/Spalten-Berechnung der einzelnen Reihen- und Spaltensummen und dann der Gesamtsumme.
 Postensumme: Erfasst die Reihen mit dem gleichen Postennamen und tabelliert sie zu einer Reihe.

Für jeden Posten können Sie bis zu 16 alphanumerische Zeichen und Symbole als Daten eingeben.

Eingabe von Daten

Wir wollen die folgende Tabelle mit Verkaufsdaten aufstellen:

Titel

SALES	BRANCH	APR.	MAY	JUN.
TV	LONDON	1580	1890	2530
VCR	LONDON	2895	1900	3185
CD Player	NEW YORK	355	298	639
VCR	NEW YORK	2180	2513	2980
TV	HAMBURG	1855	1348	2000
VCR	HAMBURG	2730	2325	3294

← Spalten-
Postennamen

↑
Reihen-Postennamen

↑
Zelle

Beim PC-1150 können Sie bis zu acht Tabellen (Tabelle 1 - 8) erstellen. Verwenden Sie Tabelle 1 für die obige Tabelle.

Eingabe des Titels

Rufen Sie das Datenbank-Menü auf das Display.

OFF ON (oder betätigen Sie CL FUNC MENU im Datenbank-Modus)

TEL 1 ~ 2, TABLE 1 ~ 8 ?

Wählen Sie TABLE1.

TABLE1 (Taste E)

TABLE 1 TITLE = ?

Geben Sie den Titel "SALES" ein:

SALES

LE1 TITLE = SALES _

ENTER

COLUMN ITEM1 = ?

Dann erscheint die Aufforderung zur Eingabe des Postennamens für die 1. Spalte.

Eingabe von Spalten-Postennamen

Geben Sie den Postennamen (BRANCH) für die 1. Spalte ein:

BRANCH

MN ITEM1 = BRANCH _

Drücken Sie **ENTER** zur Speicherung des Namens. Dann erscheint die Aufforderung zur Eingabe des Postennamens für die 2. Spalte:

ENTER

COLUMN ITEM2 = ?

Geben Sie die restlichen Spalten-Postennamen auf die gleiche Weise ein:

APR. **ENTER**

MAY **ENTER**

JUN. **ENTER**

COLUMN ITEM5 = ?

Jetzt erscheint die Aufforderung zur Eingabe des Postennamens für die 5. Spalte.

Korrektur von Daten

Sie können die gespeicherten Daten auf ähnliche Weise wie bei der Telefonliste verändern (siehe Seite 35).

Beispiel: Änderung von "APR." in der 2. Spalte zu "MAR.":

Rufen Sie mit der Taste **<DEL** das "APR." der 2. Spalte auf das Display:

<DEL **<DEL** **<DEL**

2) APR.

FUNC **EDIT**

COLUMN ITEM2 = APR

Fügen Sie zuerst ein "M" vor "A" ein:

SHIFT INS>

COLUMN ITEM2 = MAP

M

COLUMN ITEM2 = MAP

Löschen Sie dann das "P":

INS>

COLUMN ITEM2 = MAP

SHIFT <DEL

COLUMN ITEM2 = MAP

ENTER

3) MAY

Das obige Verfahren ist nur eine Methode zur Korrektur von Daten, in einigen Fällen kann die folgende Methode schneller sein:
Beispiel: Änderung von "MAR." zurück zu "APR.":

<DEL

2) MAR.

FUNC EDIT

COLUMN ITEM2 = MAR

AP

COLUMN ITEM2 = APR

ENTER

3) MAY

Nach Durchführung der Änderung bringen Sie die rechte Spalte wieder auf das Display:

INS> INS>

COLUMN ITEM5 = ?

Ende der Spalten-Postennamen-Eingabe

Drücken Sie nach Eingabe aller Spalten-Postennamen bei der letzten Aufforderung die Taste **ENTER**. Damit wird die Eingabe der Spalten-Postennamen beendet, und Sie können die Postennamen für die Reihen eingeben. Die Eingabe der Spalten-Postennamen kann nicht ausgelassen werden.

ENTER

ROW ITEM1 = ?

Eingabe von Reihen-Postennamen und Daten

Geben Sie den Postennamen (TV) für die 1. Reihe ein:

TV

ROW ITEM 1 = TV _

ENTER

TV : BRANCH = ?

Nach Drücken der Taste **ENTER** zum Speichern des Postennamens für die 1. Reihe werden die Postennamen für die 1. Reihe und Spalte an Anhaltspunkt angezeigt.

Geben Sie jetzt die Daten für die 1. Spalte in der 1. Reihe ein:

LONDON

V : BRANCH = LONDON _

ENTER

TV : APR. = ?

Wiederholen Sie die Dateneingabe für die verbleibenden drei Spalten:

1580 ENTER

1890 ENTER

2530 ENTER

ROW ITEM 2 = ?

Nach Eingabe der Daten für die 1. Reihe erscheint die Aufforderung zur Eingabe des Postennamens für die 2. Reihe. Wiederholen Sie die Eingabe der Reihen- Postennamen und der zugehörigen Daten, bis Sie die 6. Reihe erreichen.

Wenn Sie einen Fehler machen, führen Sie die Korrektur nach der früheren Beschreibung durch.

VCR ENTER LONDON ENTER 2895 ENTER 1900 ENTER 3185 ENTER

CD_P SHIFT SML LAYER SHIFT SML ENTER NEW_YORK ENTER

355 ENTER 298 ENTER 639 ENTER

VCR ENTER NEW_YORK ENTER 2180 ENTER 2513 ENTER 2980

ENTER

TV ENTER HAMBURG ENTER 1855 ENTER 1348 ENTER 2000 ENTER

VCR ENTER HAMBURG ENTER 2730 ENTER 2325 ENTER 3294 ENTER

"_" kennzeichnet eine Leerstelle, drücken Sie die Taste **SPC**.

ROW ITEM 7 = ?

Auf dem Display erscheint jetzt die Aufforderung zur Eingabe des Postennamens für die 7. Reihe. Nach Registrierung von Daten für die letzte Reihe der letzten Spalten können Sie mit der nächsten Reihe weitermachen: - Falls Sie keine weiteren Daten registrieren möchten, kehren Sie durch die folgende Tastenbetätigung zum Datenbank-Menü zurück:

CL **FUNC** **MENU**

TABLE1 ~ 2, TABLE1 ~ 8 ?

Im obigen Beispiel haben Sie die Daten Spalte für Spalte innerhalb der gleichen Reihe eingegeben. Wenn Sie die Daten Reihe für Reihe innerhalb der gleichen Spalte eingeben wollen, drücken Sie **FUNC** **↓ENTER** anstelle von **ENTER** nach jeder Dateneingabe.

Lesen der Daten

Ein Ausschnitt der Tabelle, der Kreuzungspunkt einer Reihe und einer Spalte, wird als Zelle bezeichnet. Sie können die Daten jeder Zelle mit den Tasten **CONT.↑**, **CONT.↓**, **<DEL** und **INS>** abrufen:

TABLE1

TABLE1 TITLE = SAL

Wenn ein Pfeil (→) oben rechts auf dem Display erscheint, bestehen die angezeigten Daten aus mehr als 16 Zeichen, und rechts der angezeigten Zeichen befinden sich weitere, nicht angezeigte Zeichen. Diese versteckten Daten können mit der Taste **INS>** auf das Display gerufen werden:

INS>

←
BLE1 TITLE = SALES

Gleichermaßen läßt sich der Teil der Daten, der links von den angezeigten Zeichen nicht sichtbar ist, mit der Taste **<DEL** auf das Display bringen.

Wir wollen jetzt den Spaltennamen für die 1. Spalte abrufen:

INS>

1) BRANCH

Vor jedem Spalten-Postennamen steht eine Spaltennummer mit einer rechten Klammer ")".

Als nächstes wollen wir den Postennamen für die 1. Reihe abrufen.

Gehen Sie mit **<DEL** um eine Spalte nach links und dann mit **CONT.↓** nach unten zur nächsten Reihe:

<DEL **CONT.↓**

1 > TV

Vor jedem Reihen-Postennamen steht eine Reihenummer und ein ">".
Betätigen Sie die folgende Taste, um die Daten in der 1. Reihe und der 1. Spalte zu sehen:

INS>

TV : BRANCH = LONDON

Vor den Daten jeder Zelle stehen die Reihen- und Spalten-
Postennamen:

INS>

TV : APR. = 1580

Durch Drücken von **FUNC** **<DEL** wird der Inhalt der Zelle ganz links in der gegenwärtigen Reihe angezeigt, mit **FUNC** **INS>** erscheint die Zelle ganz rechts in der gleichen Reihe.

Zur Anzeige der obersten Zelle der gegenwärtigen Spalte drücken Sie

FUNC **CONT.↑**, für die unterste Zelle der gleichen Spalten

FUNC **CONT.↓** :

FUNC **INS>**

TV : JUN. = 2530

FUNC **CONT.↓**

VCR : JUN. = 3294

Suchen von Daten

Sie können einen Posten in der jeweiligen Spalte durch Spezifizierung des oder der ersten Zeichen suchen. Wir wollen "VCR" in der Spalte für Reihen-Postennamen suchen.

Rufen Sie zunächst einen Posten (z.B. CD Player) in der Reihen-
Postennamen-Spalte auf das Display:

FUNC **TITLE**

TABLE1 TITLE = SAL →

CONT.↓ **CONT.↓** **CONT.↓**

3 > CD Player

FUNC **SEARCH**

SEARCH WORD = ?

VC

SEARCH WORD = VC _

Durch Drücken von **ENTER** beginnt die Suche in Abwärtsrichtung, beginnend mit der ersten Reihe:

ENTER

2 > VCR

Die Suche stoppt bei "VCR". Zum Fortsetzen der Suche in Abwärts- oder Aufwärtsrichtung, müssen Sie **CONT.↑** bzw. **CONT.↓** drücken.

CONT.↓

4 > VCR

CONT.↑

2 > VCR

CONT.↑

NOT FOUND

2 > VCR

Wurde der spezifizierte Posten nicht gefunden, erscheint "NOT FOUND" auf dem Display, dann wird zur vorigen Anzeige zurückgekehrt. Zum Sehen der letzten Daten drücken Sie **FUNC** **CONT.↓**, zum Sehen der ersten Daten **FUNC** **CONT.↑**. Durch Drücken einer anderen Taste als **CONT.↑** und **CONT.↓** können Sie die Suche abbrechen.

INS> (beendet die Suche)

Wenn Sie zu Beginn der Suche nach Eingabe der Zeichen, nach denen gesucht werden soll, **CONT.↓** drücken (anstelle von **ENTER**), startet die Suche in Abwärtsrichtung, beginnend mit der nächsten Reihe. Wenn Sie entsprechend **CONT.↑** (anstelle von **ENTER**) am Beginn der Suche drücken, startet die Suche in Aufwärtsrichtung, beginnend mit der Reihe darüber.

<DEL **CONT.↓** (zeigt den Postennamen der 3. Reihe erneut an)

3 > CD Player

FUNC **SEARCH**

SEARCH WORD = ?

VC **CONT.↓**

4 > VCR

Korrektur von Eintragungen

Wenn sich der Cursor nicht auf dem Display befindet, können Sie die angezeigte Eintragung (einschließlich Titel sowie Reihen- und Spalten-Postennamen) einfach durch Eingabe neuer Daten ändern.

Beispiel: Änderung von TV, LONDON, Verkäufe April von 1580 zu 1680:

FUNC **TITLE** **CONT. ↓** **INS >** **INS >**

TV: APR. = 1580

1

TV: APR. = 1__

680 **ENTER**

TV: MAY = 1890

Die Eintragung wurde geändert. Prüfen Sie:

<DEL

TV: APR. = 1680

Zum Ändern eines Teils einer Eintragung drücken Sie **FUNC** **EDIT**, so daß der blinkende Cursor auf dem Display erscheint, während die zu ändernde Eintragung angezeigt wird:

FUNC **EDIT**

TV: APR. = 1680

Ändern Sie den Wert wieder zu 1580:

INS >

TV: APR. = 1580

5

TV: APR. = 1580

ENTER

TV: MAY = 1890

Damit ist die Eintragung geändert. Prüfen Sie:

<DEL

TV: APR. = 1580

Anhängen von Eintragungen

Durch Drücken von **FUNC** **APPEND** kann eine neue Eintragung an die letzte Reihe der Tabelle angehängt werden:

FUNC **APPEND**

ROW ITEM 7 = ?

Befolgen Sie zum Registrieren der Daten die Anweisungen unter "Eingabe von Daten". Sollen keine Daten angehängt werden, kehren Sie durch die folgende Tastenbetätigung zum Titel zurück:

FUNC **TITLE**

TABLE1: TITLE=SAL

Einfügen von Eintragungen

Durch Drücken von **SHIFT** **INS>** können Sie eine neue Reihe oder Spalte an beliebiger Stelle einfügen.

Einfügen einer Spalte

Wenn Sie **SHIFT** **INS>** drücken, während ein Spalten- Postenname angezeigt wird, werden diese Spalte und alle folgenden Spalten um eine Spalte nach rechts versetzt, so daß eine neue Spalte eingefügt werden kann.

Beispiel: Einfügen der Spalte "MAR." vor der Spalte "APR.": Wenn der Titel auf dem Display ist, drücken Sie:

INS> **INS>** **INS>**

2) APR.

SHIFT **INS>**

2)

M

COLUMN ITEM2 = M_

AR. **ENTER**

3) APR.

Einfügen einer Reihe

Wenn Sie **SHIFT** **INS>** drücken, während ein Reihen- Postenname oder Daten sich auf dem Display befinden, werden diese Reihe und alle nachfolgenden um eine Reihe nach unten versetzt, so daß eine neue Reihe eingefügt werden kann.

Beispiel: Einfügen der Reihen "TV" und "PARIS" über der 5. Reihe: Rufen Sie mit den Tasten **<DEL** und **CONT.↓** die 5. Reihe auf das Display:

5 > TV

SHIFT **INS>**

5 >

T

ROW ITEM5 = T_

V

TV:BRANCH = ?

PARIS

TV:MAR. = ?

Nach Einfügen einer Reihe oder Spalte erscheint eine Aufforderung zur Eingabe von Daten für diese Reihe bzw. Spalte. Wird versucht, eine Reihe oder Spalte einzufügen, während die Tabelle bereits aus 254 Reihen oder Spalten besteht, erscheint die Anzeige "LIMIT EXCEEDED" für etwa eine Sekunde, danach kommt wieder die vorige Anzeige.

Löschen von Eintragungen

Löschen einer Spalte

Wenn Sie drücken, während sich ein Spalten-Postenname auf dem Display befindet, erscheint eine Abfrage, ob diese Spalte gelöscht werden soll.

Beispiel: Löschen der Spalte "MAR.", die Sie gerade eingefügt haben:

Drücken Sie , so daß der Postenname der 2. Spalte auf dem Display erscheint:

2) MAR.

DELETE OK?(Y/N)

2) APR.

Wenn Sie die 2. Spalte löschen wollen, drücken Sie die Taste . Wenn Sie eine andere Taste als drücken, erfolgt keine Löschung, und die vorige Anzeige erscheint wieder.

Löschen einer Reihe

Wenn Sie drücken, während ein Reihen-Postenname sich auf dem Display befindet, erscheint eine Abfrage, ob Sie diese Reihe löschen wollen.

Beispiel: Löschen der Reihen "TV" und "PARIS", die Sie gerade eingefügt haben:

TV:BRANCH = PARIS

DELETE OK?(Y/N)

TV:BRANCH = HAMBUR →

Wenn Sie die Reihen löschen wollen, drücken Sie **Y**. Wird eine andere Taste als **Y** gedrückt, erfolgt keine Löschung, und die vorige Anzeige erscheint wieder.

Löschen einer Tabelle

Wenn sich der Titel einer Tabelle auf dem Display befindet, können Sie die gesamte Tabelle durch die folgende Tastenbetätigung löschen:

FUNC **TITLE**

TABLE1 TITLE = SAL

SHIFT **<DEL**

DELETE OK?(Y/N)

Es erscheint eine Abfrage, ob Sie die Tabelle löschen wollen. Durch Eingabe von **Y** wird die Tabelle gelöscht. Auf dem Display erscheint "DELETED" und dann die Aufforderung zur Eingabe eines Titels. In unserem Beispiel drücken Sie **N**, damit die Tabelle nicht gelöscht wird:

N

TABLE1 TITLE = SAL

Reihen/Spalten-Summenbildung

Durch Drücken von **FUNC** **TOTAL** werden die Summen der einzelnen Reihen und Spalten sowie die Gesamtsumme gebildet. Wir wollen die Summenbildung für Tabelle 1 (SALES) durchführen:

FUNC **TABLE1**

TABLE1 TITLE = SAL

FUNC **TOTAL**

TABLE1 TITLE = SAL

Das Ergebnis sieht wie folgt aus:

SALES	BRANCH	APR	MAY	JUN	ROW SUM
TV	LONDON	1580	1890	2530	6000
VCR	LONDON	2895	1900	3185	7980
CD Player	NEW YORK	355	298	639	1292
VCR	NEW YORK	2180	2513	2980	7673
TV	HAMBURG	1855	1348	2000	5203
VCR	HAMBURG	2730	2325	3294	8349
COLUMN SUM	0	11595	10274	14628	36497

(GESAMTSUMME)

Verwenden Sie die Tasten **INS>** , **<DEL** , **CONT.↑** und **CONT.↓** , um die einzelnen Summen auf das Display zu rufen.

INS> ...(6 mal)... **INS>**

ROW SUM

CONT.↓

TV SUM = 6000

Die Summen-Spalte und -Reihe können durch die folgenden Operationen gelöscht werden:

- 1) Anhängen oder Löschen von Eintragungen
 - 2) Änderung von Eintragungen (einschließlich Titel und Postennamen)
- Befinden sich keine Daten in der Tabelle, führt der Rechner keine Summenbildung durch.

Hinweis zum Umgang mit Daten

Die Spalten/Reihen-Summenbildung erfolgt mit allen Daten außer den Postennamen. Die Datenwerte werden nach den folgenden Regeln erfaßt. Stellen Sie vor Nummern und anderen Daten, die nicht summiert werden sollen, ein Zeichen oder Symbol (@, usw.).

Beispiel: LONDON → 0 (Daten, die mit einem Buchstaben beginnen, werden als Null angesehen.)
 1—2 → 12 Leerstellen werden ignoriert.)
 ↑
 Leerstelle
 2 * 3 → 2 (Nicht-numerische und alle nachfolgenden Zeichen werden ignoriert.)
 @123 → 0

Daten, die mit " * * " beginnen, werden als fehlerhafte Daten behandelt. Wenn der Rechenbereich bei der Spalten/Reihen-Summenbildung überschritten wird oder wenn eine Spalte oder Reihe, die fehlerhafte Daten (* *) enthält, in die Summenbildung einbezogen wird, werden zwei Sternchen " * * " in die Summen-Spalten/Reihe eingesetzt, um den Fehler anzuzeigen.

Summenbildung nach Posten

Sie können Daten für den gleichen Postennamen summieren und eine neue Tabelle erstellen, die Summen für die Reihen und Spalten enthält. Die neue Tabelle sieht dann wie folgt aus:

SALES	BRANCH	APR.	MAY	JUN.	ROW SUM
TV	0	3435	3238	4530	11203
VCR	0	7805	6738	9459	24002
CD Player	0	355	298	639	1292
COLUMN SUM	0	11595	10274	14628	36497

(GESAMTSUMME)

Wie Sie sehen, wurden die Daten für den Posten "TV" in der 1. und 5. Reihe und die für den Posten "VCR" in der 2., 4. und 6. Reihe aufsummiert.

Versuchen Sie es jetzt:

FUNC **TITLE**

TABLE1 TITLE = SAL →

FUNC **ASSORT**

RESULT TABLE No?

Wenn Sie **FUNC** **ASSORT** betätigen, sortiert der Rechner die Tabelle wie oben gezeigt und fragt, unter welcher Tabelle-Nr. Sie die sortierte Tabelle abspeichern wollen. Wenn Sie die Speicherung in Tabelle 1 durchführen, wird der vorige Inhalt von Tabelle 1 überschrieben. Wird eine andere Tabelle spezifiziert, bleibt Tabelle 1 unverändert. Wir wollen die sortierte Tabelle in Tabelle 2 speichern:

TABLE2

PLACED IN TABLE 2

TABLE1 TITLE = SAL →

Wenn das Sortieren beendet ist, erscheint wieder die Anzeige wie vor dem Sortieren.

Wenn die sortierte Tabelle in eine Tabelle gespeichert wird, die bereits Daten enthält, werden diese Daten gelöscht. Wenn die Tabelle, in der gespeichert werden soll (Tabelle 2 im obigen Beispiel), eine geschützte Tabelle ist, erscheint bei Drücken der Taste **TABLE2** die Meldung "TABLE 2 IS SECRET", und das Sortieren wird nicht durchgeführt.

Das Sortieren wird weiterhin nicht durchgeführt, wenn in der zu sortierenden Tabelle keine Daten vorhanden sind oder wenn die Speicherkapazität zum Speichern der sortierten Tabelle in einer anderen Tabelle nicht ausreicht. Wenn die Speicherkapazität nicht ausreicht, sollten Sie nicht mehr benötigte Daten löschen und dann das Sortieren erneut durchführen.

KAPITEL 6 GESCHÜTZTE DATEN UND DATEN-SPEICHERUNG

Spezifizierung von Daten als geschützt oder nicht geschützt

Wenn Sie Ihre Telefonliste oder Ihre Tabellen nicht jedermann zugänglich machen wollen, können Sie diese Daten schützen. Wenn die Daten als geschützt spezifiziert sind, sind sie nicht mehr zugänglich.

Spezifizierung von Daten als geschützt

Sie können die Daten als geschützt spezifizieren, wenn sich der Titel auf dem Display befindet. Rufen Sie den Titel von Tabelle 1 auf das Display, um diese Tabelle als geschützt zu spezifizieren:


CL FUNC MENU



TEL 1 ~ 2, TABLE 1 ~ 8 ?

TABLE 1

TABLE 1 TITLE = SA →

FUNC 

TABLE 1 TITLE =  SA →

Durch Drücken von **FUNC**  erscheint das Symbol "" unmittelbar vor dem Titel, dann ist die Tabelle geschützt. Wir wollen zum Menü zurückkehren, um dies zu überprüfen:


CL FUNC MENU

TEL 1 ~ 2, TABLE 1 ~ 8 ?

Wählen Sie Tabelle 1:

TABLE 1

TABLE 1 IS SECRET

TABLE 1 TITLE =  ?

Nach der Meldung "TABLE 1 IS SECRET" erscheint eine Aufforderung zur Eingabe des Titels der geschützten Tabelle, der Inhalt der geschützten Tabelle wird nicht angezeigt.

Spezifizierung von Daten als nicht geschützt

Um die Daten als nicht geschützt zu spezifizieren, geben Sie den Titel für diese Daten ein. Bei Eingabe eines falschen Titels wird folgendes angezeigt:

SELL

CANNOT OPEN

TABLE1 TITLE = 5 ?

Nach der Meldung "CANNOT OPEN" erscheint die Aufforderung zur Eingabe des richtigen Titels. Geben Sie jetzt den richtigen Titel ein:

SALES

OPENED

TABLE1 TITLE = 5 SA

Der Titel wird angezeigt. Wenn Sie jetzt drücken, verschwindet das Symbol "5", und die Tabelle ist nicht mehr geschützt:

TABLE1 TITLE = SAL

Die Daten können nur als nicht geschützt spezifiziert werden, indem das Symbol "5" ausgeschaltet wird.

1 > TV

Sichern/Laden von Daten auf/von Kassette

Sie können die Telefonliste und die Tabellen vom Speicher des Rechners auf Kassette sichern. Auch Daten, die BASIC-Feldvariablen zugewiesen wurden, werden zusammen mit den Telefonlisten- und Tabellen- Daten gesichert und geladen.

Sichern von Daten auf Kassette

Drücken Sie , um den RUN-Modus zu wählen.

Zum Sichern aller Telefonlisten- und Tabellen- Daten führen Sie den folgenden Befehl aus:

CSAVE MEM "Dateiname"

Als "Dateiname" können Sie einen beliebigen Namen verwenden. Notieren Sie sich den Dateinamen, da Sie ihn benötigen, wenn Sie die Daten zurück in den Rechner laden. Nach Sichern der Daten führen Sie eine Überprüfung durch Ausführung von CLOAD MEM? "Dateiname" aus.

Laden von Daten von Kassette

Drücken Sie , um den RUN-Modus zu wählen. Zum Laden der Telefonliste und der Tabellen von Kassette führen Sie den folgenden Befehl aus:

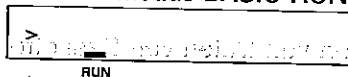
CLOAD MEM "Dateiname"

Der "Dateiname" muß der gleiche Name sein, den Sie beim Sichern der Daten verwendet haben.

KAPITEL 7 RECHENFUNKTIONEN

Bei Verwendung des Computers als Taschenrechner werden die rechts angeordneten Tasten verwendet.

Drücken Sie **SHIFT** **DB→BASIC** (Stellen Sie den Modus BASIC RUN ein.)



Drücken Sie vor jeder Berechnung die Taste **CL**.

Grundrechenarten

Beispiel: $12 \times 3 + 5 =$

12 ***** 3 **+** 5 **ENTER**

41.

Hinweis: Dieser Computer verwendet die Tasten *****, **/** und **ENTER** anstelle von \times , \div und $=$.

Beispiel: $5 \times - 6 + 7 =$

5 * - 6 + 7

5 * - 6 + 7 _

ENTER

- 23.

Das Symbol "-" unmittelbar nach einem arithmetischen Operator oder "(" wird als Minussymbol angesehen, nicht als Operator für Subtraktion.

Beispiel: $-5.2 \times (10^{-2} + 1.7) =$

CL - 5.2 ***** **SHIFT** **(** 1

SHIFT **Exp** - 2 + 1.7 **SHIFT** **)**

ENTER

- 5.2 * (1 E - 2 + 1.7) _

- 8.892

Korrektur von Tasteneingaben

Bei der Eingabe unterlaufen immer mal Fehler, sei es daß eine falsche Ziffer eingegeben wurde, sei es daß eine Ziffer vergessen wurde. Mit den nach folgend beschriebenen Methoden können Sie solche Fehler korrigieren. Die Wiederabruf- Funktion wird verwendet, wenn schon die Taste **ENTER** betätigt wurde. (Siehe Seite 46.)

Bei sehr vielen Fehlern

Drücken Sie die Taste **CL**, um die Eingabe zu löschen, und beginnen Sie erneut vom Anfang.

Wenn die falsche Taste gedrückt wurde

Bringen den Cursor mit der Taste **<DEL** auf das falsche Zeichen, so daß dieses blinkt. Drücken Sie dann die richtige Taste.

Beispiel: $2 + 3 * 4$ wurde als $2 + 3/4$ eingegeben

$2 + 3/4$	<input type="text" value="2 + 3 / 4 _"/>
<DEL <DEL	<input type="text" value="2 + 3 / 4"/>
*	<input type="text" value="2 + 3 * 4"/>
INS>	<input type="text" value="2 + 3 * 4 _"/>

Wenn nicht benötigte Tasten gedrückt wurden

Bewegen Sie den Cursor mit der Taste **<DEL** auf das überschüssige Zeichen, so daß es blinkt. Drücken Sie dann **SHIFT** **<DEL**.

Beispiel: $2 + 3 * 4$ wurde als $2 + 43 * 4$ eingegeben

$2 + 43 * 4$	<input type="text" value="2 + 4 3 * 4 _"/>
<DEL <DEL <DEL <DEL	<input type="text" value="2 + 4 3 * 4"/>
SHIFT <DEL	<input type="text" value="2 + 3 * 4"/>
ENTER	<input type="text" value="14."/>

Wenn eine Tasteneingabe ausgelassen wurde

Bringen Sie den Cursor mit der Taste **<DEL** auf das Zeichen rechts von dem fehlenden Zeichen, so daß es blinkt. Drücken Sie dann **SHIFT** **INS>**, um an der Cursor-Position eine Leerstelle zu schaffen, umgeben das fehlende Zeichen ein.

Beispiel: $2 + 3 * 4$ wurde als $2 + 34$ eingegeben

2 + 34

<DEL

SHIFT

INS>

*

ENTER

2 + 34 _

2 + 34

2 + 3 4

2 + 3 * 4

14.

Wiederabruf-Funktion

Nach Ausführung einer Berechnung mit der Taste **ENTER** kann die ausgeführte Rechnung durch Drücken der Tasten **<DEL** oder **INS>** wiederabgerufen werden. Diese Funktion ist praktisch, um die ausgeführte Berechnung zu überprüfen oder um einen Teil der Berechnung für erneute Ausführung zu ändern.

Die Wiederabruf-Funktion ist besonders dann praktisch, wenn eine Berechnung in einem Fehler resultiert. Der Cursor kennzeichnet die Stelle, wo der Fehler auftrat, so daß die Fehlerursache leicht gefunden werden kann. (Die Wiederabruf-Funktion kann jedoch nicht eingesetzt werden, wenn das Aufforderungszeichen (>) angezeigt wird, weil dann die Berechnung bereits gelöscht ist.)

Beispiel: Fehler tritt auf, wenn $36/1 + 2$ **ENTER** eingegeben wird.

36/1 + 2 **SHIFT** **)** **ENTER**

<DEL

ERROR 1 E

36 / 1 + 2)

Korrigieren Sie jetzt die Berechnung durch Eingabe von $36/(1 + 2)$.

36/ **SHIFT** **(** 1 + 2 **SHIFT** **)** **ENTER**

12.

Rufen Sie die Berechnung mit der $\boxed{\langle \text{DEL} \rangle}$ Wiederabruhf-Funktion zurück auf die Anzeige.

$\boxed{\langle \text{DEL} \rangle}$

3 6 / (1 + 2) _

$\boxed{\text{ENTER}}$

1 2 .

Oder verwenden Sie die Wiederabruhf-Funktion mit der Taste $\boxed{\text{INS} >}$.

$\boxed{\text{INS} >}$

3 6 / (1 + 2)

Anzeige des Berechnungsergebnisses

Zahlen werden normalerweise in der Dezimaldarstellung angezeigt, z.B. 120000 oder 0.00123. Sehr große bzw. sehr kleine Zahlen werden jedoch in Exponential-Darstellung (oder Fließpunktdarstellung) angezeigt, z.B. 1.2×10^5 oder 1.23×10^{-3} .

Zur Anzeige einer Zahl in exponentieller Darstellung müssen Sie vor der Eingabe des Exponenten **E** eingeben, z.B. $1.2\text{E}05$ oder $1.23\text{E} - 03$.

1. 23 **E** - 03

Mantisse Exponent

Werden mehr als 2 Stellen für den Exponenten eingegeben, werden nur die beiden letzten 2 Stellen verwendet.

Beispiel: 5 $\boxed{\text{SHIFT}}$ $\boxed{\text{Exp}}$

1230 $\boxed{\text{ENTER}}$

5 . E 3 0

Fehleranzeige

Wenn in einer Berechnung ein Fehler auftritt, wird eine Fehlernummer wie **ERROR 1** oder **ERROR 2** angezeigt.

Ein Fehler kann mit der Taste $\boxed{\text{CL}}$ gelöscht werden, Löschen mit der Taste $\boxed{\langle \text{DEL} \rangle}$ oder $\boxed{\text{INS} >}$ ist aber auch möglich. Beim Löschen eines Fehlers mit der Taste $\boxed{\langle \text{DEL} \rangle}$ oder $\boxed{\text{INS} >}$ wird die Stelle angezeigt, wo der Fehler auftrat. (Siehe die Beschreibung der Wiederabruhf-Funktion.) Eine Beschreibung der Fehlernummern findet sich auf Seite 95.

Speicher und Variable

Zahlen und Zeichen werden im Datenspeicher gespeichert. Beispielsweise wird für eine Gleichung wie $A + 3$ der Datenspeicher A verwendet. Wenn dann der Wert von A geändert (2, 3, 7, ...) und die Gleichung berechnet wird, werden von der gleichen Gleichung unterschiedliche Ergebnisse erhalten. Da der Wert des Datenspeichers in einer Gleichung geändert werden kann, wird der Datenspeicher als Variable bezeichnet. Variablen sind in folgende Gruppen unterteilt:

Variable

Dieser Computer verwendet drei Arten von Variablen:

Numerische Variable für Zahlen: 10, 124, 5 E 12.

Zeichenvariable für Zeichenfolgen: ABC, cat?

Feldvariable, siehe die Anweisung DIM auf Seite 69.

Numerische Variable können in feste numerische Variable (A-Z) und numerische Feldvariable (A()-Z()) aufgeteilt werden. Zeichenvariable lassen sich in feste Zeichenvariable (A\$, Z\$) und Zeichenfeldvariable (A\$()-Z\$()) unterteilen.

Variablenamen

Die alphabetischen Großbuchstaben werden als Variablenamen verwendet. Die kleinen Buchstaben finden keine Verwendung.

Speichern von Daten in Variable

- 1) Speichern eines numerischen Wertes in eine numerische Variable
Numerische Variable = Ausdruck (Der Ausdruck repräsentiert einen numerischen Wert, eine Gleichung oder eine andere numerische Variable.)

Beispiel:

$A = 3$ **ENTER**

Speicherung von 3 in Variable A

$B = A + 2$ **ENTER**

Speicherung des Ergebnisses der Addition von 2 zur Variablen A in Variable B

In eine numerische Variable kann ein numerischer Wert mit einer Mantisse von bis zu 10 Stellen und einem Exponenten von bis zu 2 Stellen gespeichert werden.

- 2) Speichern eines Zeichens in eine Zeichenvariable
Zeichenvariable = "Zeichen"
Zeichenvariable = Zeichenvariable

Durch Anfügen des Dollarsymbols (\$) an einen Variablennamen wird die Variable als Zeichenvariable gekennzeichnet.

Beispiel:

A\$ = "The" Speicherung der Zeichen The in A\$.
 B\$ = "cat" Speicherung der Zeichen cat in B\$.
 C\$ = A\$ + " " + B\$ Speicherung der Zeichen The cat in C\$.

Zeichenketten können durch das Plussymbol (+) miteinander verbunden werden.

Hinweise:

1. In dieser Bedienungsanleitung wird eine Leerstelle durch dargestellt. Drücken Sie zur Eingabe dieses "Zeichens" die Taste .
2. Feste numerische Variable und feste Zeichenvariable belegen den gleichen Speicher. Daher können beispielsweise die Variable A und A\$ nicht gleichzeitig verwendet werden.

Beispiel:

A = 123 Speicherung von 123 in Variable A
 A\$ = "ABC" Löschung von A und Speicherung von ABC in A\$
 A\$ Anzeige von ABC
 A Fehler

Ein Fehler tritt ein, wenn eine Zeichenvariable als numerische Variable spezifiziert wird.

Die Speicherplätze für feste Variable sind, wie schon der Name sagt, fest, während die Speicherplätze für Feldvariable vor der Verwendung eingestellt werden müssen.

Die folgende Tabelle zeigt auf, wieviel Speicherplatz (in Byte) bei der Definition von Feldern belegt wird.

Variable	Variablenname	Daten
Numerische Feldvariable	6 Bytes	8 Bytes
Zeichenfeldvariable	6 Bytes	Spezifizierte Anzahl Byte

Wenn beispielsweise DIMB\$(2,3)*10 spezifiziert wird, werden 12 Variable spezifiziert, wobei jede Variable 10 Zeichen aufnehmen kann. Die Anzahl der verwendeten Byte ist:

$$6 \text{ Byte} + 10 \text{ Byte pro Variable} \times 12 \text{ Variable} = 126 \text{ Byte}$$

(Variablenname) (Zeichenkette)

Löschen von Variablen

Führen Sie zum Löschen von Variablen die folgenden Eingaben durch.

Numerische Variable werden durch Eingabe von 0 gelöscht.

Zeichenvariable werden durch Eingabe von "" gelöscht.

Die Eingabe von "" in eine Variable bedeutet, daß die Variable keine Zeichen enthält. Dies wird als Nullvariable bezeichnet.

Befehl CLEAR

Durch Ausführung des Befehls CLEAR können alle Variable gleichzeitig gelöscht werden.

CLEAR

Durch Ausführung dieses Befehls werden alle Feldvariable, die mit der Anweisung DIM definiert wurden, und der Inhalt aller festen Variablen gelöscht. (Siehe den Befehl CLEAR.)

KAPITEL 8 PROGRAMMIERUNG

Ein Programm besteht aus einer Gruppe von Befehlen, die den Rechner anweisen, ein Verfahren, beispielsweise eine Berechnung, durchzuführen. Das Schreiben dieser Befehle in einer Form, die für den Computer verständlich ist, wird als Programmieren bezeichnet. Dieses Kapitel beschreibt die Grundlagen der Programmierung.

Zeilen

Die Grundeinheit von BASIC-Programmen ist die Zeile. Eine Zeile beginnt nach der Zeilennummer und endet an der Stelle, wo die Taste **ENTER** gedrückt wird. Sie kann eine oder mehrere Befehle oder Ausdrücke enthalten. (Mehrere Befehle oder Ausdrücke müssen durch Doppelpunkt voneinander getrennt werden.)

Die Länge einer Zeile (wobei die Taste **ENTER** als ein Zeichen zählt) kann maximal 80 Tastenbefehle (oder maximal 80 Byte) betragen.

Zeilennummern

Zeilennummern stehen am Anfang jeder Zeile und kennzeichnen die Reihenfolge der Programmausführung. Programme werden in aufsteigender Reihenfolge der Zeilennummern ausgeführt. Zur Rückkehr zu einer früheren Zeile oder zum Auslassen einer Zeile in einem Programm kann die Ausführung durch Spezifizierung einer Zeilennummer zu dieser verzweigt werden (z.B. GOTO 80). Als Zeilennummern können ganze Zahlen von 1 bis 999 verwendet werden.

Eingabetaste **ENTER**

Durch Drücken der Taste **ENTER** werden die eingegebenen Zeichen im Computer gespeichert.

10 INPUT A

```
10 INPUT A _  
PRO
```

ENTER

```
10: INPUT A  
PRO
```

Der Computer setzt automatisch einen Doppelpunkt hinter die Zeilennummer und trennt den Befehl (INPUT) von der Variablen (A).

Eintasten-Ausführung

Ein BASIC-Programm kann ausgeführt werden, indem die Taste **FUNC** und danach eine alphabetische Taste (A-Z) gedrückt wird.

Wenn beispielsweise der Buchstabe "A" unmittelbar hinter einer Zeilennummer in einem Programm gesetzt wird, wird das Programm beginnend mit dieser Zeile ausgeführt, indem die Eintasten-Ausführung für A (Taste **FUNC** und Taste **A**) betätigt wird.

Drücken Sie bei Eingabe des Programms die Tasten **SHIFT** **"** **A** **SHIFT** **"**.

Wenn entsprechend der Buchstabe "B" unmittelbar hinter einer Zeilennummer in einem Programm gesetzt wird, wird das Programm beginnend mit dieser Zeile ausgeführt, indem die Eintasten-Ausführung für B (Taste **FUNC** und Taste **B**) betätigt wird.

Programmgröße

Die folgende Tabelle enthält die Anzahl Byte, die Zeilennummern und Befehle belegen.

Element	Zeilennummer	Befehl & Funktion	ENTER und andere
Anzahl der belegten Byte	2 Bytes	1 Byte	1 Byte

Für die Speicherkapazität des Computers siehe die Technischen Daten (Seite 109). Der noch freie Speicherplatz kann mit dem Befehl MEM überprüft werden (siehe Seite 87).

Eingabe und Ausführung von Programmen

Programme werden im BASIC/PRO-Modus eingegeben und im RUN-Modus ausgeführt.

Wir wollen jetzt ein Programm eingeben, das Tabellendaten ausdrückt (siehe die Programmliste auf Seite 112).

Wenn die Eintasten-Ausführung für A (Taste **FUNC** und Taste **A**) betätigt wird, wird das Programm ausgeführt und der Inhalt der Tabelle (Tabelle 1) ausgedruckt.

In der Programmliste befindet sich der Buchstabe "A" hinter der Zeilennummer 10. Daher kann das Programm ausgeführt werden, indem die Eintasten-Ausführung für A (Taste **FUNC** und Taste **A**) betätigt wird.

Eingabemethode

Drücken Sie die Taste **DB↔BASIC**, um den PRO-Modus zu aktivieren.

DB↔BASIC

> _____
PRO

Geben Sie NEW ein und drücken dann die Taste **ENTER**. Dadurch werden alle gespeicherten Programme gelöscht, und ein neues Programm kann eingegeben werden. Fangen wir mit der ersten Zeile an.

10 **SHIFT** " A **SHIFT** "
SHIFT : PRINT = LPRINT

A" : PRINT = LPRINT _

Zur Eingabe des Doppelpunktes (:) müssen Sie die Taste **SHIFT** und dann die Taste **0** drücken. Die anderen Symbole werden auf die gleiche Weise eingegeben: zuerst die Taste **SHIFT** und dann die entsprechende Taste (die mit dem benötigten Symbol markiert ist) drücken.

ENTER

10 : "A" : PRINT =

Sie müssen am Ende jeder Zeile die Taste **ENTER** drücken.

• **Für die Tasteneingabe siehe Kapitel 3 Daten-Eingabe.**

Zur Eingabe von kleinen Buchstaben müssen Sie die Tasten **SHIFT** und **SML** und dann die entsprechende Buchstabentaste betätigen. Nach Eingabe des Kleinbuchstabens wird durch erneutes Drücken der Tasten **SHIFT** und **SML** wieder auf Großbuchstaben-Eingabe umgeschaltet. Zur Erläuterung der Fehlerkorrektur geben Sie IIF anstelle von IF in Zeile 30 und J=J1 anstelle von J=J+1 in Zeile 50 ein.

Hinweis: Wenn kein Drucker angeschlossen ist, siehe den Hinweis auf Seite 112.

Überprüfung und Korrektur des Programms

Überprüfen Sie das Programm nach der Eingabe mit der Programmliste. Mit den Tasten **CONT.↓** und **CONT.↑** können Sie die einzelnen Zeilen auf das Display bringen.

Wenn eine Zeile mehr als 16 Zeichen enthält, werden die Zeichen hinter dem 16. Zeichen nicht angezeigt.

30 : "B" : I I F D 1 \$ (

Drücken Sie die Taste **INS>**, um den Rest der Zeile anzuzeigen.

INS> ... **INS>**

```
30 + 1: GOTO "B"
```

Wenn Sie einen Fehler entdecken, korrigieren Sie ihn nach dem folgenden Verfahren.

Rufen Sie die zu korrigierende Zeile auf das Display, und drücken die Taste **INS>**, um den Cursor zu bewegen.

Wir wollen jetzt $J = J + 1$ in Zeile 50 zu $J = J + 1$ korrigieren. Bringen Sie den Cursor mit der Taste **INS>** auf die Position, wo das "+" eingefügt werden muß, und drücken dann die Tasten **SHIFT**, **INS>** und **+**.

INS> ... **INS>**

SHIFT **INS>**

+

```
50: "C": PRINT "
```

```
50 "D": J = J
```

```
50 "D": J = J 1
```

```
50 "D": J = J + 1
```

Nach der Korrektur müssen Sie die Taste **ENTER** drücken.

ENTER

```
50: "C": PRINT "
```

Löschen

Das Zeichen oder der Befehl, auf dem der blinkende Cursor steht, wird durch Drücken der Tasten **SHIFT** und **** gelöscht.

Fehler

Die Programmiersprache BASIC, wie jede Sprache, hat bestimmte Syntaxregeln. Wenn beispielsweise ein Wort einen Schreibfehler enthält oder eine Zahl vorhanden ist, wo BASIC ein Zeichen erwartet, tritt ein Fehler ein. Wenn dies vorkommt, wird während der Ausführung ERROR angezeigt. Drücken Sie die Tasten **DB-BASIC** (Anwahl des PRO-Modus) und **CONT**. Dann wird die Zeile, in der der Fehler auftrat, angezeigt, und der Cursor markiert die Fehlerstelle.

Korrektur

Wenn ein neues Zeichen an der Stelle, wo der Cursor blinkt, eingegeben wird, wird das alte Zeichen durch das neue ersetzt.

Da Befehle wie INPUT und PRINT in abgekürzter Form gespeichert werden, bei der Korrektur die Anzeige überprüfen.

Ausführung

Eingabefehler können auch durch Ausführung des Programms gefunden werden. Wenn eine Zeile einen Fehler enthält, wird ERROR angezeigt, und die Ausführung wird unterbrochen. Korrigieren Sie die Fehler und versuchen erneut das Programm auszuführen. Das Programm ist in Ordnung, wenn keine weiteren Fehler angezeigt werden und das Programm einwandfrei bis zum Ende ausgeführt wird.

Drücken Sie die Taste **DB↔BASIC**, um den RUN-Modus zu aktivieren.
Drücken Sie die Tasten **FUNC** und **A**, um das Programm auszuführen.

```
ERROR 9 IN 30
RUN
```

Da in Zeile 30 ein Fehler vorhanden ist, wird eine Fehlermeldung angezeigt. Das bedeutet, daß in Zeile 30 ein Fehler vorhanden ist.

Drücken Sie **CL**, **DB↔BASIC** und **CONT.↑**.
(Dadurch wird der PRO-Modus aktiviert.)

Die Zeile (30), die den Fehler enthält, wird angezeigt, und der blinkende Cursor befindet sich auf der Stelle, an der der Fehler auftrat.

Vergleichen Sie die Anzeige mit der Programmliste. In diesem Fall müssen Sie ein I löschen, da ein I zuviel vorhanden ist.

SHIFT **<DEL**

```
30 "B": F D 1 $( I
```

Nachdem Sie sichergestellt haben, daß die korrigierte Zeile jetzt mit der Programmliste übereinstimmt, drücken Sie die Taste **ENTER**, um die korrigierte Zeile abzuspeichern.

Drücken Sie die Taste **DB↔BASIC**, um den RUN-Modus zu aktivieren, und führen das Programm erneut aus.

Fehlersuche

WIC
Ein Syntaxfehler oder ein Fehler in einem Ausdruck, der verhindert, daß das richtige Ergebnis erhalten wird, wird als Bug (engl. Käfer) bezeichnet. Daher ist Debugging die Aufgabe, Fehler im Programm zu finden und zu korrigieren.

KAPITEL 9

BASIC-REFERENZ

Dieses Kapitel ist in drei Abschnitte aufgeteilt:

Kommandos: Anweisungen, die außerhalb von Programmen zur Programm-Ablaufsteuerung, für Peripheriegeräte usw. verwendet werden.

Befehle: Werden in BASIC-Programmen als Anweisungen verwendet.

Funktionen: Spezielle Operatoren, die in BASIC-Programmen zur Umwandlung von Variablen verwendet werden.

Die Kommandos und Befehle sind alphabetisch angeordnet, für jede Eintragung ist eine Seite vorgesehen. Die Funktionen sind in drei Kategorien und dort jeweils alphabetisch angeordnet.

<u>Kommandos</u>	<u>Befehle</u>	<u>Funktionen</u>
CLOAD	BEEP	(Pseudovariable)
CLOAD ?	CLEAR	INKEY\$
CONT	DIM	MEM
CSAVE	END	PI
GOTO	FOR-TO-STEP,NEXT	(Numerische Funktionen)
LIST	GOSUB, RETURN	ABS
LLIST	GOTO	ACS
NEW	IF-THEN	ASN
RUN	INPUT	ATN
	LET	COS
	LPRINT	DEG
	PRINT	DMS
	STOP	EXP
	USING	INT
	WAIT	LOG
		LN
		SGN
		SIN
		SQR
		TAN
		&

Funktionen
(Zeichenfunktionen)
ASC
CHR\$
LEFT\$
LEN
MID\$
RIGHT\$
STR\$
VAL

Die Bedeutung der Terminologie in den Formaten wird folgend erklärt.

Ausdruck: Bedeutet einen numerischen Wert, eine numerische Variable oder eine Gleichung, die solche enthält.

Variable: Bedeutet eine numerische Variable oder Zeichenvariable, die Feldelemente enthält.

Zeichenkette: Bedeutet ein "Zeichen" oder eine Zeichenvariable.

"Zeichen": Bedeutet die Zeichen (Buchstaben, Ziffern, Symbole), die in Anführungsstriche eingeschlossen sind.

**{A
B}:** A oder B kann gewählt werden.

[]: Der Inhalt der eckigen Klammern kann ausgelassen oder wiederholt werden.
Falls darauf jedoch ein anderer Parameter folgt, muß ein Komma oder Semikolon als Trennzeichen gesetzt werden.

KOMMANDOS

- 1 CLOAD "Dateiname"
- 2 CLOAD
- 3 CLOAD MEM "Dateiname"
- 4 CLOAD MEM

ANWENDUNG:

Das Kommando CLOAD dient zum Laden von Programmen oder Daten (Telefonliste, Tabellen und Daten für Feldvariable) von Kassette. Wenn die Telefonliste oder Tabellen durch ein Passwort geschützt sind, werden sie in geschützter Form geladen.

Format 1 des CLOAD-Kommandos löscht den Speicher, sucht das Band nach dem Programm ab, dessen Name mit "Dateiname" angegeben ist und lädt das Programm.

Format 2 löscht die existierenden Programme aus dem Speicher und lädt das erste Programm auf dem Band, angefangen bei der aktuellen Position.

Format 3 sucht das Band nach der Datei mit dem "Dateinamen" ab und lädt sie in den Computer.

Format 4 lädt die erste auf dem Band bei der Wiedergabe gefundene Datei in den Computer.

Während des Ladevorgangs werden ein Stern (für Programme) oder zwei Sterne (für Daten) oben rechts auf dem Display angezeigt. Nach dem Laden verschwinden die Sterne, und die Aufforderung (>) erscheint.

Wird der spezifizierte Dateiname nicht gefunden, sucht der Computer auch nach Erreichen des Bandendes weiter nach dem Dateinamen. In diesem Fall ist die Taste drücken, um die Suche abzubrechen.

Tritt während der Ausführung ein Fehler auf, sind die geladenen Programme oder Daten nicht verwendbar.

Beispiel 1: **CLOAD "PRO-1"**

Der Computer sucht und lädt das Programm mit dem Dateinamen "PRO-1".

Beispiel 2: **CLOAD MEM "TEL.DAT"**

Der Computer sucht und lädt die Daten mit dem Dateinamen "TEL.DAT".

- 1 CLOAD ? "Dateiname"
- 2 CLOAD ?
- 3 CLOAD MEM ? "Dateiname"
- 4 CLOAD MEM ?

ANWENDUNG:

Das Kommando CLOAD? dient zum Vergleichen von Programmen und Daten (Telefonliste, Tabellen oder Daten für Feldvariable), die auf Band gesichert wurden, mit den entsprechenden Daten im Speicher des Computers.

Format 1 des CLOAD?-Kommandos sucht nach dem mit "Dateinamen" bezeichneten Programm und vergleicht es dann mit dem Speicherinhalt.

Format 2 vergleicht das im Speicher enthaltene Programm mit dem ersten Programm auf dem Band, angefangen bei der aktuellen Position.

Format 3 sucht das Band nach dem "Dateinamen" ab und vergleicht die darin enthaltenen Daten mit denen im Speicher des Computers.

Format 4 vergleicht die erste bei der Wiedergabe gefundene Datei auf dem Band mit den Daten im Speicher des Computers.

Falls beim Vergleich ein Fehler gefunden wird, wird ERROR 8 angezeigt.

Während des Vergleichens werden ein Stern (für Programme) oder zwei Sterne (für Daten) oben rechts auf dem Display angezeigt. Nach Beendigung des Ladevorgangs verschwinden die Sterne, und die Aufforderung (>) wird angezeigt.

Beispiel 1: CLOAD? "PRO-1"

Der Computer sucht das Programm mit dem Dateinamen "PRO-1" und vergleicht es mit dem im Computer gespeicherten Programm.

Beispiel 2: CLOAD MEM? "TEL.DAT"

Der Computer sucht die Daten mit dem Dateinamen "TEL.DAT" und vergleicht sie mit den im Computer gespeicherten Daten.

1 CONT

ANWENDUNG:

Mit dem CONT-Kommando setzen Sie die Ausführung eines zeitweise unterbrochenen Programms fort.

Der Befehl CONT kann nur im RUN-Modus verwendet werden.

Wenn die Ausführung eines Programms mit dem Befehl STOP unterbrochen wurde, so kann sie mit der Anweisung CONT fortgesetzt werden.

Wurde ein Programm mit der -Taste unterbrochen, so kann es nach Eingabe des CONT-Befehls am Aufforderungszeichen fortgesetzt werden.

1 CSAVE "Dateiname"

2 CSAVE

3 CSAVE MEM "Dateiname"

4 CSAVE MEM

ANWENDUNG:

Mit dem Kommando CSAVE wird ein Programm auf dem Band gesichert.

Format 1 des Kommandos CSAVE speichert alle Programme im Speicher des Rechners auf dem Band und weist den entsprechenden Dateinamen zu.

Format 2 speichert alle Programme des Rechners ohne spezielle Dateinamen auf dem Band.

Format 3 schreibt die Daten aus dem Speicher des Computers mit dem spezifizierten Dateinamen auf Band.

Beispiel: CSAVE MEM "TEL.DAT"

Die Daten im Speicher werden in die Datei "TEL.DAT" auf Band geschrieben.

Format 4 schreibt die Daten aus dem Speicher des Computers auf Band, ohne einen Dateinamen zu spezifizieren.

Auf der gleichen Kassettenseite (A oder B) sollten nicht mehrere Dateien den gleichen Dateinamen aufweisen, weil sonst beim Laden die falsche Datei geladen werden kann.

Das Laden wird vereinfacht, wenn man sich die Position des Bandzählwerks vor den einzelnen Dateien notiert.

Dateinamen

Ein Dateiname ist der Name einer Datei (die ein Programm oder Daten enthält). Wenn eine Datei mit Dateiname gesichert wird, kann der Computer die Datei anhand des Dateinamens finden und dann laden. Der Dateiname kann aus bis zu 7 Zeichen, Buchstaben, Ziffern und Symbolen, bestehen.

1 GOTO { Ausdruck "Etikett" }

ANWENDUNG:

Mit dem Kommando GOTO wird die Ausführung eines Programms begonnen.

Das Kommando GOTO kann anstelle des Kommandos RUN benutzt werden, um die Ausführung eines Programms an der durch Ausdruck bestimmten Zeilennummer zu starten.

GOTO unterscheidet sich von RUN auf die folgende Weise:

- 1) Der Wert des WAIT-Intervalles wird nicht zurückgesetzt.
- 2) Wurde die Anzeige durch USING-Anweisung formiert, wird sie nicht gelöscht.
- 3) Variablen und Felder bleiben erhalten.
- 4) Der FOR-NEXT- und der GOSUB-Stapel werden nicht gelöscht.

Die Ausführung eines Programms mit dem Kommando GOTO ist die gleiche wie die Ausführung mit der **FUNC**-Taste.

BEISPIEL:

GOTO 100

Die Ausführung des Programms wird bei Zeile 100 gestartet.

1 LIST

2 LIST Ausdruck

3 LIST "Etikett"

ANWENDUNG:

Das Kommando LIST kann nur im PRO-Modus verwendet werden.

Die erste Form des LIST-Kommandos zeigt die Zeile mit der kleinsten Zeilennummer an.

Die zweite Form zeigt die Zeile mit der niedrigsten Zeilennummer an, die größer als der Wert des Ausdrucks ist.

Dann kann das Programm mit den Aufwärts- und Abwärts-Tasten durchlaufen werden.

Die dritte Form zeigt die Anweisung mit dem spezifizierten Etikett an.

1 LLIST

2 LLIST Ausdruck 1 , Ausdruck 2

ANWENDUNG:

Druckt auf dem Drucker ein Programm aus.

Das LLIST-Kommando kann nur im PRO-oder RUN-Modus verwendet werden.

Format 1 druckt alle im Speicher befindlichen Programme aus.

Format 2 druckt die Anweisungen von der niedrigsten Zeilennummer gleich oder größer dem Wert von Ausdruck 1 bis zur höchsten Zeilennummer gleich oder kleiner dem Wert von Ausdruck 2. Zwischen den beiden Nummern müssen sich mindestens zwei Zeilen befinden.

1 NEW

ANWENDUNG:

Das Kommando NEW kann nur im PRO-Modus verwendet werden.
Das NEW-Kommando löscht alle Programme und Daten, die sich im Speicher befinden.

Die Telefonliste und die Tabellen bleiben jedoch erhalten.

Das NEW-Kommando ist im RUN-Modus nicht definiert und hat die Fehlermeldung ERROR 9 zur Folge.

1 RUN

2 RUN { Ausdruck
"Etikett" }

ANWENDUNG:

Mit dem RUN-Kommando wird das Programm im Speicher gestartet.

Die erste Form des RUN-Kommandos startet ein Programm mit der Zeile, die die niedrigste Zeilennummer hat.

Die zweite Form des RUN-Kommandos startet ein Programm mit der angegebenen Zeilennummer.

Das RUN-Kommando unterscheidet sich von GOTO auf die folgende Weise:

- 1) Der Wert des WAIT-Intervalles wird zurückgesetzt.
- 2) Wurde die Anzeige durch USING-Anweisungen formatiert, so wird sie gelöscht.
- 3) Variablen und Felder, mit Ausnahme der festen Variablen, werden gelöscht.
- 4) Der FOR-NEXT- und der GOSUB-Stapel werden gelöscht.

Die Ausführung eines Programms mit dem GOTO-Kommando ist die gleiche wie die Ausführung mit der -Taste.

BEISPIEL:

RUN 100 Beginnt die Ausführung des Programms ab Zeile 100.

BEFEHLE

1 BEEP Ausdruck

ANWENDUNG:

Erzeugt Töne der über den internen Lautsprecher des Computers.

Die Anzahl der Signaltöne wird durch den Ausdruck bestimmt, welcher numerisch sein muß. Der Ausdruck wird berechnet, aber nur der ganzzahlige Teil wird zur Bestimmung der Anzahl Signaltöne verwendet.

BEEP kann auch als Kommando mit numerischen Literalen und vordefinierten Variablen verwendet werden. In diesem Fall ertönt der Signalton unmittelbar nach Drücken der Taste **ENTER**.

BEISPIEL:

10 A = 5
20 BEEP 3 erzeugt 3 Signaltöne
30 BEEP A erzeugt 5 Signaltöne

1 CLEAR

ANWENDUNG:

Der Befehl CLEAR setzt Speicherplatz frei, in dem Variablen gespeichert waren. Dies ist praktisch, wenn die im ersten Teil eines Programms verwendeten Variablen im zweiten Teil nicht mehr benötigt werden und der vorhandene Speicherplatz begrenzt ist. CLEAR kann auch am Anfang eines Programms angewendet werden, wenn mehrere Programme im Speicher vorhanden sind und der bei der Ausführung früherer Programme verwendete Speicherplatz gelöscht werden soll.

CLEAR setzt die Speicherbereiche, die durch die Variablen A-Z oder A\$-Z\$ benutzt wurden, nicht frei, da diese fest vorgegeben sind. Der Befehl CLEAR setzt numerische Variable auf 0 und Zeichenvariable auf NULL zurück.

Die Telefonliste und die Tabellen (D0\$(,)-D9\$(,)) werden jedoch nicht gelöscht. Verwenden Sie im Datenbank-Modus zum Löschen der Telefonliste und der Tabellen.

BEISPIEL:

```
100 "A" : CLEAR : DIM B (4)
```

- 1 DIM numerischer Variablenname (Größe) [, numerischer Variablenname (Größe)]
- 2 DIM numerischer Variablenname (Reihen, Spalten) [, numerischer Variablenname (Reihen, Spalten)]
- 3 DIM Zeichenfolgenname (Größe) * Länge [, Zeichenfolgenname (Größe) * Länge]
- 4 DIM Zeichenfolgenname (Reihen, Spalten) * Länge [, Zeichenfolgenname (Reihen, Spalten) * Länge]

ANWENDUNG:

DIM wird benutzt, um Speicher für numerische und Zeichenfolgen-Feldvariablen frei zu halten.

Eine Feldvariable und eine einfache Variable können den gleichen Namen haben. Ein Zeichenfeld und ein numerisches Feld können mit Ausnahme des Dollarsymbols den gleichen Namen haben.

Ein Feld kann maximal zwei Dimensionen haben; die maximale Größe einer Dimension ist 255. Zu der angegebenen Zahl von Elementen, die im DIM-Befehl definiert ist, addiert sich ein "nulltes" Element. So hat z.B. das Feld DIM B(3) die Elemente B(0), B(1), B(2) und B(3). In zweidimensionalen Feldern hat es eine extra "nullte" Reihe und Spalte.

In Zeichenfolgen-Feldern kann zusätzlich zur Anzahl der Elemente die Größe der einzelnen Zeichenfolge-Elemente vorgegeben werden. So reserviert z.B. DIM B\$(3) * 12 Speicherraum für 4 Zeichenfolgen mit je 12 Zeichen Länge. Wird die Länge der Zeichenfolgen nicht vorgegeben, kann jede Zeichenfolge maximal 16 Zeichen enthalten.

Bei der Initialisierung eines Feldes werden alle Werte eines numerischen Feldes gleich Null und eines Zeichenfolgen-Feldes gleich leer gesetzt.

Beispielsweise setzt DIM D1\$(8, 4) die Anzahl Spalten auf 5 (einschließlich der Titel- und der Gesamtsumme-Spalte) und die Anzahl Reihen auf 9 (einschließlich der Titel- und der Gesamtsummen-Reihe) für die Datenvariable D1\$() von Tabelle 1.

Die Feldvariablen können mit dem Befehl CLEAR (auf ihren nicht definierten Zustand) gelöscht werden. Die Feldvariablen werden auch

gelöscht, wenn die Programmausführung mit dem Kommando RUN gestartet wird. (Siehe das CLEAR-Kommando.)

Nach der Definition kann ein Feldname nicht umdefiniert werden. Zuerst mit dem Kommando CLEAR löschen und dann neu definieren.

BEISPIEL:

- 10 DIM B(10) Schafft Platz für ein numerisches Feld mit 11 Elementen.
- 20 DIM C\$(4, 4) * 30 Schafft Platz für ein zweidimensionales Zeichenfolgen-Feld mit 5 Reihen und 5 Spalten; jede Zeichenfolge hat maximal 30 Zeichen.

1 END

ANWENDUNG:

Ein END-Befehl signalisiert das Ende eines Programms.

Sind mehrere Programme in den Speicher eingegeben, so ist es nötig, den einzelnen Programmen Endmarkierungen zu geben, damit bei der Ausführung eines Programms der Rechner nicht in ein anderes Programm hineinläuft. Dies wird mit dem END-Befehl als letzte Anweisung im Programm erreicht.

BEISPIEL:

```
10 PRINT "HELLO"  
20 END  
30 PRINT "GOODBYE"  
40 END
```

Mit diesen Programmen im Speicher drückt ein Kommando "RUN 10" das Wort "HELLO", aber nicht das Wort "GOODBYE". Diese wird auf das Kommando "RUN 30" ausgegeben.

1	FOR	numerische Variable = Ausdruck 1	TO	Ausdruck 2
	STEP	Ausdruck 3		
		}		
	NEXT	numerische Variable		

ANWENDUNG:

FOR wird in Verbindung mit NEXT benutzt, um eine Reihe von Operationen mehrmals zu wiederholen.

FOR und NEXT schließen eine Gruppe von Anweisungen ein, die wiederholt werden sollen. Wird diese Gruppe das erste Mal ausgeführt, so hat die Schleifenvariable (die Variable, die unmittelbar nach FOR genannt wird) ihren Anfangswert (Ausdruck 1).

Erreicht die Ausführung nun die NEXT-Anweisung, so wird die Schleifenvariable durch den Ergänzungswert erhöht und dann mit dem Schlußwert (Ausdruck 2) verglichen. Ist der Wert der Schleifenvariablen weniger oder gleich dem Ausdruck 2, so wird die eingeschlossene Gruppe ein weiteres Mal ausgeführt, beginnend mit dem Befehl hinter FOR. Die Schrittweite wird durch Ausdruck 3 gegeben. Wenn Ausdruck 3 ausgelassen wird, wird die Schrittweite gleich 1 gesetzt.

Schleifenvariablen größer als Ausdruck 2, so wird die Ausführung unmittelbar hinter NEXT fortgesetzt. Da der Vergleich am Ende durchgeführt wird, werden die Anweisungen innerhalb eines FOR...NEXT Paares mindestens einmal ausgeführt.

Die Befehle FOR und NEXT müssen immer paarweise verwendet werden. Die numerische Variable hinter dem FOR-Befehl und die numerische Variable hinter dem NEXT-Befehl müssen gleich sein.

Ausdruck 2 und Ausdruck 3 müssen im folgenden Bereich liegen: - 32768 bis 32767 (Nur der ganzzahlige Teil ist gültig. Schrittweite \neq 0)

Die Schleifenvariable kann in der umschlossenen Gruppe von Anweisungen benutzt werden, z.B. als ein Feld-Index. Man sollte aber vorsichtig sein, wenn der Wert der Schleifenvariablen verändert wird.

Programme sollen so geschrieben werden, daß die Ausführung nicht aus dem FOR...NEXT Paar herauspringt, bevor der Zähler den Schlußwert erreicht. Um aus einer Schleife herauszukommen, bevor sie die bestimmte

Anzahl von Malen wiederholt wurde, setzen Sie die Schleifenvariable höher als den Schlußwert.

Die Gruppe von Anweisungen, die von dem FOR...NEXT Paar eingefasst werden, kann ein weiteres Paar von FOR...NEXT-Anweisungen enthalten, das eine andere Schleifenvariable haben muß, solange das eingeschlossene Paar vollständig eingeschlossen ist. Wenn also eine FOR-Anweisung in der Gruppe auftaucht, so muß auch der dazugehörige NEXT-Befehl eingeschlossen sein. So können bis zu fünf FOR...NEXT Paare ineinander "verschachtelt" sein.

BEISPIEL:

```
10 FOR I=1 TO 5
20 PRINT I
30 NEXT I
40 FOR N=10 TO 0 STEP -1
50 PRINT N
60 NEXT N
70 FOR N=1 TO 10
80 X=1
90 FOR F=1 TO N
100 X=X * F
110 NEXT F
120 PRINT X
130 NEXT N
```

(10-30) Diese Gruppe von Anweisungen druckt die Zahlen 1,2,3,4,5 aus.

(40-60) Diese Gruppe von Anweisungen zählt rückwärts 10,9,8.....0.

(70-130) Diese Gruppe von Anweisungen berechnet und druckt die Fakultäten N der Zahlen von 1 bis 10 aus.

```

1 GOSUB { Ausdruck }
        { "Etikett" }
        }
RETURN

```

ANWENDUNG:

Verzweigt die Programmausführung zu einer BASIC-Subroutine.

Wenn die gleiche Gruppe von Anweisungen in einem Programm mehrere Male ausgeführt werden soll oder wenn diese Gruppe von Anweisungen in verschiedene Programme eingesetzt werden soll, so empfiehlt sich die Verwendung der BASIC-Subroutinen mit Hilfe von GOSUB und RETURN.

Die Gruppe von Anweisungen wird im Programm an der gleichen Stelle eingeschlossen, an der sie normalerweise nicht erreicht werden kann. Eine gemeinsame Stelle folgt der END-Anweisung, die das Ende des Hauptprogramms angibt. An diesen Stellen des Hauptprogramms, an denen die Subroutinen ausgeführt werden sollen, fügen Sie die GOSUB-Anweisung mit der Zeilennummer der Startzeile der Subroutine als Ausdruck ein. Die letzte Zeile der Subroutine muß einen RETURN-Befehl enthalten. Wird der GOSUB-Befehl nun ausgeführt, so springt der Computer an die angegebene Zeilennummer und verarbeitet die Anweisungen, bis ein RETURN-Befehl erreicht wird. Der Rechner springt dann in die Anweisung hinter dem GOSUB-Befehl zurück.

In einer Subroutine kann ein GOSUB vorkommen. Maximal können 10 Subroutinen ineinander "verschachtelt" werden.

BEISPIEL:

```

10   GOSUB 100
20   END
100  PRINT "HELLO"
110  RETURN

```

In Ausführung druckt dieses Programm HELLO einmal.

1 GOTO

{ Ausdruck
"Etikett" }

ANWENDUNG:

GOTO veranlaßt den Rechner, von einer bestimmten Stelle in einem BASIC-Programm an eine andere Stelle zu springen. Anders als beim GOSUB "erinnert" sich GOTO nicht, von wo dieser Sprung ausgeführt wurde.

Wird ein numerischer Wert benutzt, so muß er als eine gültige Zeilennummer bewertet werden, d. h. zwischen 1 und 65279, oder Sie bekommen die Fehlermeldung ERROR 4.

Gute Programme laufen normalerweise von Anfang bis Ende ohne Sprünge, abgesehen von Subroutinen, die während des Programms ausgeführt werden. Der prinzipielle Gebrauch des GOTO-Befehls ist deshalb in der Anweisung IF...THEN zu finden.

BEISPIEL:

```
10 INPUT A$
20 IF A$ = "Y" THEN GOTO 50
30 PRINT "NO"
40 GOTO 60
50 PRINT "YES"
60 END
```

Dieses Programm druckt das Wort "YES" aus, wenn Sie "Y" eingeben, und druckt das Wort "NO", wenn Sie etwas anderes eingeben.

- 1 IF Bedingung THEN Anweisung
- 2 IF Bedingung Anweisung

ANWENDUNG:

IF...THEN wird benutzt, um eine Anweisung auszuführen oder nicht auszuführen, wenn bestimmte Bedingungen bei der Programmausführung erfüllt sind.

Im normalen Ablauf von BASIC-Programmen werden die Anweisungen in der Reihenfolge ihres Auftretens abgearbeitet. Das IF...THEN-Befehlspar erlaubt es, eine bestimmte Anweisung nur dann auszuführen, wenn eine bestimmte Bedingung erfüllt ist. Ist der Bedingungsteil der IF-Anweisung wahr, so wird die Anweisung ausgeführt, ist die Bedingung falsch, wird sie übersprungen.

Wenn nach dem THEN-Befehl eine Zeilennummer oder ein Etikett gesetzt wird, springt die Ausführung zur Zeile mit der Zeilennummer oder dem Etikett.

Wenn eine Zuweisungs-Anweisung für eine ausführbare Anweisung geschrieben wird, ist der Befehl LET immer erforderlich.

Eine ausführbare Anweisung kann ohne den Befehl THEN geschrieben werden. Zur Kennzeichnung eines Sprungs ist der Befehl THEN oder GOTO immer erforderlich.

Beispiel: IF A < 5 LET C = A * B : GOTO 50

Wenn A kleiner als 5 ist, wird C das Ergebnis von $A * B$ zugewiesen, und die Ausführung springt zu Zeile 50.

Beispiel: IF B = C + 1 GOTO 60
oder

IF B = C + 1 THEN 60

Wenn B und C + 1 gleich sind, springt die Ausführung zu Zeile 60.

Der Ausdruck nach dem Befehl IF (z.B. $A < 5$) wird als Bedingungsausdruck bezeichnet. Ein Bedingungsausdruck wird normalerweise im folgenden Format geschrieben:

Bedingungsäusdruck	Entscheidung
$OO = XX$	gleich (ist OO gleich XX ?)
$OO > XX$	größer als (ist OO größer als XX ?)
$OO \geq XX$	größer oder gleich (ist OO größer oder gleich XX ?)
$OO < XX$	kleiner als (ist OO kleiner als XX ?)
$OO \leq XX$	kleiner oder gleich (ist OO kleiner oder gleich XX ?)
$OO < > XX$	ungleich (ist OO ungleich XX ?)

Hinweis: OO und XX repräsentieren Ausdrücke oder Variable. (5 * 4, A, 8, usw.)

Vergleichen von Zeichenfolgen

Zeichenfolgen können in einem Bedingungsausdruck verglichen werden.

Beispiel: IF A\$ > = " ABC " THEN.....

Ziffern, Zeichen und Symbole werden anhand ihrer Zeichencodes verglichen. Siehe die Zeichencode-Tabelle auf Seite 107. Die Länge einer zusammenhängenden Zeichenfolge oder die Länge einer Zeichenfolge in einem Ausdruck darf 80 Zeichen nicht überschreiten.

- | | | |
|---|-------|---|
| 1 | INPUT | Variable [, Variable] |
| 2 | INPUT | "Zeichenfolge", Variable [, "Zeichenfolge" , Variable] |
| 3 | INPUT | "Zeichenfolge"; Variable [, "Zeichenfolge" ; Variable] |

ANWENDUNG:

INPUT wird benutzt, um einen oder mehrere Werte über die Tastatur einzugeben.

Wollen Sie verschiedene Werte für jeden Programmablauf eingeben, benutzen Sie INPUT, um diese Werte über die Tastatur einzugeben.

In der ersten Form hat die INPUT-Anweisung keine Aufforderungs-Zeichenfolge, sondern ein Fragezeichen erscheint in der linken oberen Ecke des Displays. Sie geben dann einen Wert, gefolgt von **ENTER** , ein. Dieser Wert wird dann der ersten Variablen der Variablenliste zugewiesen. Sind mehrere Variablen in der gleichen INPUT-Anweisung enthalten, so wird dieser Prozeß so oft wiederholt, bis die Liste erschöpft ist.

Wenn ein Aufforderungszeichen in der INPUT-Anweisung enthalten ist, so ist der Prozeß genau derselbe, nur daß anstelle eines Fragezeichens ein Aufforderungszeichen in der linken oberen Ecke des Displays erscheint. Wenn dem Aufforderungszeichen ein Semikolon folgt, erscheint der Cursor nach ausgeführter Anweisung unmittelbar nach dem Aufforderungszeichen. Wenn dem Aufforderungszeichen ein Komma folgt, erscheint nach ausgeführtem Befehl das Aufforderungszeichen, und wenn Daten eingegeben wurden, wird das Display gelöscht, so daß das erste Zeichen der Eingabe links angezeigt wird.

Wenn ein Aufforderungszeichen angegeben wird und die darauf folgende Variablenliste mehr als eine Variable enthält, so werden die zweite und die folgenden Variablen mit einem Fragezeichen gekennzeichnet. Beinhaltet die Liste ein zweites Aufforderungszeichen, so wird es für die unmittelbar darauf folgende Variable angezeigt.

Wenn die Taste **ENTER** ohne Eingabe gedrückt wird, behält die Variable den Wert, den sie vor der Anweisung INPUT hatte.

BEISPIEL:

10 INPUT A

20 INPUT "A = "; A

30 INPUT "A = ", A

40 INPUT "X = ?"; X, "Y = ?"; Y

[10] Löscht das Anzeigenfeld und setzt ein Fragezeichen an den linken Rand.

[20] Gibt "A = " aus und wartet auf Eingabe.

[30] Gibt "A = " aus. Bei der Eingabe verschwindet "A = " und der eingegebene Wert wird am linken Rand angezeigt.

[40] Gibt "X = ?" aus und wartet auf die erste Eingabe. Nach dem **ENTER** wird die Anzeige gelöscht und "Y = ?" wird am linken Rand angezeigt.

1 LET numerische Variable = numerischer Ausdruck

2 LET Zeichenvariable = Zeichenausdruck

ANWENDUNG:

Mit LET weisen Sie der angegebenen Variablen den Wert des Ausdrucks zu. Die Form des Ausdrucks muß mit der Variablen übereinstimmen, d. h. nur numerische Ausdrücke können numerischen Variablen und nur Zeichenfolge-Ausdrücke können Zeichenfolgen-Variablen zugewiesen werden. Um Werte aus der einen Form in die andere zu übertragen, muß eine der beiden expliziten Umsetzfunktionen (STR\$ oder VAL) eingesetzt werden.

Der LET-Befehl kann in allen LET-Anweisungen weggelassen werden, es sei denn, die Anweisungen folgen auf das THEN einer IF...THEN-Anweisung.

BEISPIEL:

10 I = 10

Weist I den Wert 10 zu.

20 A = 5 * I

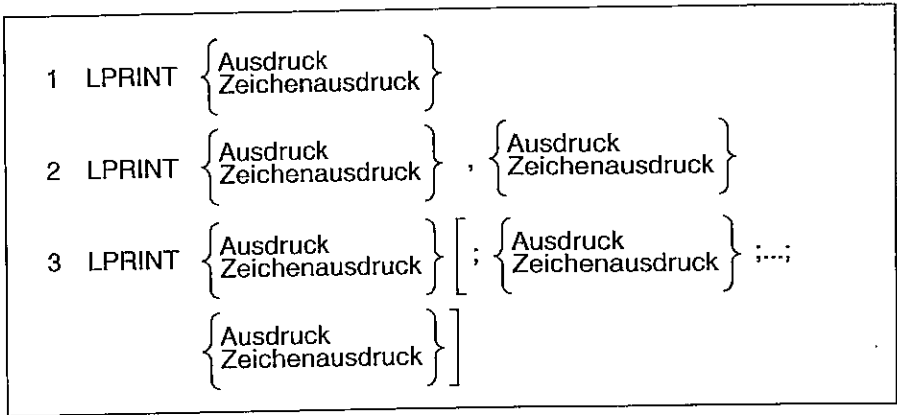
Weist A den Wert 50 zu.

30 X\$ = STR\$(A)

Weist X\$ den Wert 50 zu.

40 IF I >= 10 THEN LET Y\$ = X\$ + ".00"

Weist Y\$ den Wert 50.00 zu.



ANWENDUNG:

Der Befehl LPRINT dient zum Drucken von Informationen, Berechnungsergebnissen usw.

Die erste Form der LPRINT-Anweisung druckt einen einzelnen Wert. Wenn der Ausdruck numerisch ist, wird der Wert an der rechten Papierseite gedruckt. Bei einem Zeichenausdruck erfolgt der Ausdruck an der linken Papierseite.

Bei der zweiten Form wird das Papier in zwei Hälften für je 12 Zeichen aufgeteilt, die beiden Werte werden entsprechend den obigen Regeln in die Hälften gedruckt.

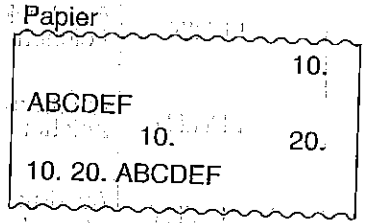
Bei der dritten Form beginnt der Druck immer an der linken Papierseite, und die Werte werden aufeinanderfolgend von links nach rechts ohne Zwischenräume gedruckt.

Es ist möglich, PRINT-Anweisungen als LPRINT-Anweisungen fungieren zu lassen. Für Einzelheiten siehe den Befehl PRINT.

Wenn eine LPRINT-Anweisung mehr als 24 Zeichen enthält, werden die ersten 24 auf einer Zeile gedruckt, die nächsten 24 auf die nächste Zeile und so weiter. Anders als bei PRINT erfolgt nach der Ausführung der LPRINT-Anweisung keine Unterbrechung.

BEISPIEL:

```
10 A = 10: B = 20: X$ = " ABCDEF "  
20 LPRINT A  
30 LPRINTX$  
40 LPRINT A, B  
50 LPRINT A; B; X$
```



1	PRINT	{ Ausdruck Zeichenausdruck }	
2	PRINT	{ Ausdruck Zeichenausdruck }	, { Ausdruck Zeichenausdruck }
3	PRINT	{ Ausdruck Zeichenausdruck }	[; { Ausdruck Zeichenausdruck } ; ... ; { Ausdruck Zeichenausdruck }]
4	PRINT = LPRINT		
5	PRINT = PRINT		

ANWENDUNG:

Mit PRINT werden Texte, Ergebnisse usw. auf dem Display angezeigt.

Die erste Form gibt einen einzelnen Wert aus. Wenn es sich dabei um einen numerischen Ausdruck handelt, so wird der Wert von der rechten Seite des Bildschirms angezeigt. Handelt es sich um einen Zeichenfolge-Ausdruck, beginnt die Anzeige ganz links.

Bei der zweiten Form wird das Display in zwei Hälften für je 8 Zeichen aufgeteilt, die beiden Werte werden entsprechend den obigen Regeln in die Hälften gedruckt.

Bei der dritten Form beginnt die Anzeige an der linken Seite, und die Werte werden aufeinanderfolgend von links nach rechts ohne Zwischenräume angezeigt.

Die vierte Form bewirkt, daß alle folgenden PRINT- Anweisungen als LPRINT-Anweisungen ausgeführt werden. Die fünfte Form hebt diesen Zustand wieder auf, dann bewirken PRINT-Anweisungen Anzeige auf dem Display.

Es ist möglich, PRINT-Anweisungen zu schreiben, die mehr als 16 Zeichen anzeigen würden, jedoch erscheinen nur die ersten 16 Zeichen von links auf dem Display. Dann gibt es keine Möglichkeit, die anderen Zeichen zu sehen.

BEISPIEL:

Anzeige

10 A=10 : B = 20 : X\$ = " ABCDEF "

20 PRINT A

30 PRINT X\$

40 PRINT A, B

50 PRINT A; B; X\$

10.

ABCDEF

10.

20.

10. 20. ABCDEF

1 STOP

ANWENDUNG:

Wenn STOP während des Programmablaufs angetroffen wird, stoppt die Ausführung, und es erscheint eine Meldung wie "BREAK IN 200", wobei 200 die Nummer der Zeile ist, die STOP enthält. STOP wird während der Entwicklung eines Programms benutzt, um den Ablauf des Programms zu kontrollieren oder den Zustand der Variablen zu überprüfen. Sie können die Ausführung mit dem CONT-Kommando fortsetzen.

BEISPIEL:

10 STOP Läßt die Meldung "BREAK IN 10" auf der Anzeige erscheinen.

- 1 USING
- 2 USING " Editier-Spezifikation "
- 3 USING Zeichenvariable

ANWENDUNG:

USING wird gebraucht, um die angezeigte oder gedruckte Ausgabe zu formatieren.

USING kann einzeln oder als Erweiterung eines LPRINT-oder PRINT-Befehls eingesetzt werden. USING erstellt eine Ausgabeformatierung, die für alle folgenden Ausgaben Gültigkeit hat, bis die Formatierung durch ein neues USING geändert wird.

Wenn daher kein Anzeigeformat benötigt wird, sollte es mit Form 1 aufgehoben werden. Das spezifizerte Format kann auch mit dem Kommando RUN oder durch Drücken von **SHIFT** **CA** gelöscht werden.

Die Editier-Spezifikation ist aus einer Kombination folgender Zeichen zusammengesetzt:

#	<p>Spezifiziert eine numerische Stelle.</p> <p>Anzahl der Stellen im ganzzahligen Teil: 2 bis 11 Stellen (einschließlich Vorzeichen)</p> <p>Wenn ein numerischer Wert weniger als die spezifizierte Anzahl Stellen enthält, werden die überzähligen Stellen mit Leerstellen gefüllt. Enthält er mehr als die spezifizierte Anzahl Stellen, tritt ein Fehler auf.</p> <p>Werden 12 oder mehr Stellen spezifiziert, werden 11 Stellen angenommen.</p> <p>Anzahl der Stellen des Dezimalteils : 0 bis 12 Stellen (0 bis 9 Stellen bei exponentieller Darstellung)</p> <ul style="list-style-type: none"> • Wenn ein numerischer Wert weniger als die spezifizierten Anzahl Stellen enthält, werden die überzähligen Stellen mit 0 gefüllt. Enthält er mehr als die spezifizierte Anzahl Stellen, werden die überzähligen Stellen abgeschnitten. <p>Spezifiziert die Anzeige des Dezimalpunktes (die Trennung des ganzzahligen Teils vom Dezimalteil).</p>
^	<p>Spezifiziert die Anzeige für exponentielle Darstellung:</p> <p>Unabhängig von der spezifizierten Anzahl Stellen beträgt die Anzahl der Stellen der Mantisse 2 Stellen (einschließlich Vorzeichen) .</p> <p>Werden 9 Stellen oder mehr für den Dezimalteil spezifiziert, wird die Mantisse auf 9 Stellen gesetzt.</p>
&	<p>Spezifiziert die Anzahl der Zeichen in einer Zeichenfolge.</p> <p>Wenn die Zeichenfolge weniger als die spezifizierte Anzahl Zeichen enthält, werden die überzähligen Stellen mit Leerstellen gefüllt. Enthält sie mehr als die spezifizierte Anzahl Stellen, wird nur die spezifizierte Anzahl Stellen angezeigt.</p>

BEISPIEL:

- (1) USING " ### "
Anzeige von 2 ganzzahligen Stellen (3 Stellen einschließlich dem Vorzeichen)
- (2) USING " ###. "
Anzeige von 2 ganzzahligen Stellen (3 Stellen einschließlich dem Vorzeichen) und des Dezimalpunktes
- (3) USING " ###.## "
Anzeige von 2 ganzzahligen Stellen (3 Stellen einschließlich dem Vorzeichen) , des Dezimalpunktes und 2 Dezimalstellen
- (4) USING " ##.## ^ "
Anzeige in exponentieller Darstellung mit 2 Dezimalstellen (der ganze Zahl des Festpunktteils einschließlich dem Vorzeichen wird automatisch auf 2 Stellen gesetzt, der Exponent einschließlich dem Vorzeichen besteht aus 4 Stellen (E – 00))
- (5) USING " &&&&&& "
Anzeige von 6 Zeichen
- (6) USING " ###&&&& " "
Gleichzeitige Spezifikation von Zahlen und Zeichen

Der USING-Befehl kann auch in einem PRINT-Befehl eingesetzt werden.

Beispiel: 10 B = -10, C = 10.7703
20 PRINT USING " &&&### " ; " B = " B ;
" _C = " ; USING " ###.### " ; C

Hinweise:

1. Wenn die Gesamtzahl der mit USING spezifizierten Stellen für " PRINT Ausdruck " 16 überschreitet, tritt ERROR 7 auf.
Wenn der spezifizierte ganzzahlige Teil jedoch 11 Stellen überschreitet, werden die überzähligen Stellen ignoriert.
2. Wenn die Anzahl der Stellen für den ganzzahligen Teil (einschließlich Vorzeichen und Dezimalpunkt) bei Verwendung des Festpunktsystems für " PRINT Ausdruck, Ausdruck" 8 überschreitet, tritt ERROR 7 auf.
Wenn die Zeichenfolge des Ausdrucks in der Form " PRINT Ausdruck, Ausdruck " 8 Stellen überschreitet, wird der überzählige Teil nicht angezeigt.
3. Wenn der Anzeigehalt der Form " PRINT Ausdruck; Ausdruck " 16 Stellen überschreitet, wird der überzählige Teil nicht angezeigt.

1 WAIT Ausdruck

2 WAIT

ANWENDUNG:

WAIT wird benutzt, um die Zeit festzusetzen, die die Ausgabe einer Information auf der Anzeige während des Programmablaufs wiedergegeben werden soll.

Bei normaler Programmausführung wartet der Computer nach einem PRINT-Kommando, bis die **ENTER**-Taste gedrückt wird. WAIT weist den Computer an, eine Ausgabe für eine vorgegebene Zeit anzuzeigen und dann die Ausführung des Programms automatisch fortzusetzen. Der Ausdruck nach WAIT setzt die Länge des Intervalls fest. Der Intervall kann jeden Wert von 0 bis 65535 haben. Dabei bedeutet jede Erhöhung um 1 eine Verlängerung um 1/64 Sekunde. WAIT 0 ist zu schnell, als daß die Ausgabe gelesen werden könnte. WAIT 65535 ist etwa 17 Minuten. WAIT ohne Ausdruck dahinter setzt den Computer in seinen ursprünglichen Wartezustand, bis die **ENTER**-Taste gedrückt wird.

BEISPIEL:

10 WAIT 64 Der Rechner wartet nach PRINT etwa 1 Sekunde.

FUNKTIONEN

Pseudovariablen

Pseudovariablen sind eine Gruppe Funktionen, die keine Argumente benötigen und bei Bedarf wie einfache Variablen verwendet werden.

1 INKEY\$

INKEY\$ ist eine Zeichen-Pseudovariablen, die der spezifizierten Variablen den Wert der gedrückten Taste während der Ausführung der Funktion INKEY\$ zuweist. INKEY\$ wird zum Ansprechen auf das Drücken einzelner Tasten verwendet, ohne auf die Taste **ENTER** zum Beenden der Eingabe zu warten. Der Computer fragt laufend ab, bis er eine Meldung von der Tastatur erhält.

```
10 A$ = INKEY$
20 B = ASC A$
30 IF B = 0 THEN GOTO 10
40 PRINT B
```

Wird eine andere Taste als eine der folgenden gedrückt, erfolgt die Eingabe als Null.

A - Z, SPC, =, 0-9, ., +, /, *, -,

Mit der Taste **BRK** kann das Programm unterbrochen werden.

Die Funktion INKEY\$ kann nicht zum Lesen einer Tastenfunktion verwendet werden, die nach der Taste **SHIFT** oder **FUNC** gedrückt wurde.

1 MEM

MEM ist eine numerische Pseudovariablen, die die Anzahl der verbleibenden Zeichen im Programmspeicher ergibt. Der verbleibende Programmspeicher ist der Gesamtspeicher ohne den Speicherplatz, der von Daten und Programmen eingenommen wird. MEM kann auch als Kommando verwendet werden.

Der freie Bereich kann auch im Datenbank-Modus verwendet werden.

1 PI

PI ist eine numerische Pseudovariablen mit dem Wert von PI. Sie ist identisch zur Verwendung des PI-Symbols (π) auf der Tastatur. Wie andere Zahlen weist der Wert eine Genauigkeit von 10 Stellen auf (3.141592654).

Numerische Funktionen

Numerische Funktionen sind eine Gruppe mathematischer Operationen, die einen numerischen Wert als Argument annehmen und einen numerischen Wert als Ergebnis liefern. Dazu gehören trigonometrische Funktionen, logarithmische Funktionen und Funktionen für den ganzzahligen Teil und das Vorzeichen von Zahlen. Viele BASIC-Dialekte fordern, daß die Argumente von Funktionen in Klammern eingeschlossen werden. Bei diesem Computer sind die Klammern nur dann erforderlich, wenn gekennzeichnet werden muß, welcher Teil eines komplexen Ausdrucks zum Argument gehört.

LOG 100 + 100 wird interpretiert als:
(LOG 100) + 100 und nicht als LOG (100 + 100)

1 ABS numerischer Ausdruck

Zeigt den absoluten Wert des numerischen Arguments an. Der absolute Wert ist der Wert einer Ziffer unabhängig von seinem Vorzeichen. ABS-10 ist 10.

1 ACS numerischer Ausdruck

Zeigt den Arcuscossinus des numerischen Arguments an. Der Arcuscossinus ist der Winkel, dessen Cosinus gleich dem Ausdruck ist. ACS0.5 beträgt 60 in Dezimalgrad.

1 ASN numerischer Ausdruck

Zeigt den Arcussinus des numerischen Arguments an. Der Arcussinus ist der Winkel zwischen dem der Sinus gleich dem Ausdruck ist. ASN0.5 beträgt 30 in Dezimalgrad.

1 ATN numerischer Ausdruck

Zeigt den Arcustangens des numerischen Ausdrucks an. Der Arcustangens ist der Winkel, dessen Tangens gleich dem Ausdruck ist.
ATN 1 beträgt 45 in Dezimalgrad.

1 COS numerischer Ausdruck

Zeigt den Cosinus des Winkelarguments an. COS 60 ist .5.

1 DEG numerischer Ausdruck

Wandelt das Argument eines Winkels in der DMS-Form (Alt-Grad, Minuten, Sekunden) in die DEG-Form (Dezimal-Grad) um. In der DMS-Form repräsentiert die ganze Zahl die Gradzahl, die erste und zweite Stelle die Minuten und die dritte und vierte Stelle hinter dem Komma repräsentieren die Sekunden. Jede weitere Stelle gibt Dezimalsekunden an. DEG 55 . 10445 ist 55.17902778.

1 DMS numerischer Ausdruck

Wandelt das Winkelargument in DEG auf DMS um . (siehe DEG) .
DMS 55 . 17902778 ist 55.10445.

1 EXP numerischer Ausdruck

Zeigt den Wert von e (2.718281828-die Basis des natürlichen Logarithmus) erhoben zum Wert des numerischen Arguments an.
EXP 1 ist 2.718281828.

1 INT numerischer Ausdruck

Zeigt die ganze Zahl des eigenen Arguments an. Die ganze Zahl von PI ist 3.

1 LOG numerischer Ausdruck

Zeigt den Logarithmus der Basis 10 seines eigenen numerischen Arguments an. LOG 2 ist 2.

1 LN numerischer Ausdruck

Zeigt den Logarithmus der Basis e (2.718281828...) des eigenen numerischen Arguments an. LN 100 ist 4.605170186.

1 SGN numerischer Ausdruck

Zeigt den auf das Zeichen des Arguments basierenden Wert an.
Wenn $x > 0$, wird 1 angezeigt.
Wenn $x < 0$, wird -1 angezeigt.
Wenn $x = 0$, wird 0 angezeigt.

1 SIN numerischer Ausdruck

Zeigt den Sinus des Winkelarguments an. SIN 30 ist 0.5.

1 SQR numerischer Ausdruck

Zeigt die Quadratwurzel des Arguments an. Benutzen Sie entweder das SQR-Kommando oder das Wurzelzeichen ($\sqrt{\quad}$) auf der Tastatur.

1 TAN numerischer Ausdruck

Zeigt den Tangens seines Winkelarguments an. TAN 45 ist 1.

1 & numerischer Ausdruck

& ist eine numerische Funktion, die Hexadezimalzahlen in Dezimalzahlen umwandelt. Das Hexadezimalsystem beruht auf der Zahl 16 und nicht auf 10 wie beim Dezimalsystem. Hexadezimalzahlen werden mit den Ziffern 0 bis 9 und den Buchstaben A, B, C, D, E und F dargestellt, welche den Zahlen 10, 11, 12, 13, 14, 15 und 16 entsprechen. Damit der Computer Hexadezimalzahlen als solche erkennt, muß diesen das Symbol "&"

vorangestellt werden. &10 ist 16 und &FFFF ist 65535.

Zeichenfunktionen

Zeichenfunktionen sind eine Gruppe von Operationen zum Umgang mit Zeichenfolgen. Einige haben eine Zeichenfolge als Argument und liefern einen numerischen Wert als Ergebnis, bei anderen ist es umgekehrt. Wieder andere haben eine Zeichenfolge und einen oder zwei numerische Werte als Argument und liefern eine Zeichenfolge als Ergebnis. Viele BASIC-Dialekte fordern, daß das Argument in Klammern eingeschlossen wird. Bei diesem Computer muß das Argument nur dann in Klammern eingeschlossen werden, wenn gekennzeichnet werden muß, welcher Teil eines komplexen Ausdrucks zum Argument gehört. Bei Zeichenfunktionen mit zwei oder drei Argumenten sind die Klammern erforderlich.

1 ASC numerischer Ausdruck

ASC ist eine Zeichenfunktion, die den numerischen Wert des ASCII-Codes des ersten Zeichens im Argument liefert. Die Tabelle der ASCII-Codes und ihre Beziehung zu den Zeichen findet sich im Anhang F. ASC "A" ist 65.

1 CHR\$ numerischer Ausdruck

CHR\$ ist eine Zeichenfunktion, die das Zeichen liefert, das dem ASCII-Code des Arguments entspricht. Die Tabelle der ASCII-Codes und ihre Beziehung zu den Zeichen findet sich im Anhang F. CHR\$ 65 ist "A".

1 LEFT\$ (Zeichenausdruck, numerischer Ausdruck)

LEFT\$ ist eine Zeichenfunktion, die den linken Teil des Zeichenausdrucks im Argument liefert. Dabei wird die Anzahl Zeichen durch den numerischen Ausdruck bestimmt. LEFT\$ ("ABCDEF", 2) ist "AB".

1 LEN Zeichenausdruck

LEN ist eine Zeichenfunktion, die die Länge des Zeichenausdrucks liefert. LEN "ABCDEF" ist 6.

1 MID\$ (Zeichenausdruck, numerischer Ausdruck 1, numerischer Ausdruck 2)

MID\$ ist eine Zeichenfunktion, die den mittleren Teil des Zeichenausdrucks im Argument liefert. Der erste numerische Ausdruck bestimmt die Position des ersten Zeichens des Ergebnisses und der zweite numerische Ausdruck die Anzahl Zeichen des Ergebnisses. MID\$ ("ABCDEF", 2, 3) ist "BCD".

1 RIGHT\$ (Zeichenausdruck, numerischer Ausdruck)

RIGHT\$ ist eine Zeichenfunktion, die den rechten Teil des Zeichenausdrucks im Argument liefert. Dabei wird die Anzahl Zeichen durch den numerischen Ausdruck bestimmt. RIGHT\$ ("ABCDEF", 3) ist "DEF".

1 STR\$ numerischer Ausdruck

STR\$ ist eine Zeichenfunktion, die den numerischen Ausdruck des Arguments in eine Zeichenfolge umwandelt, und damit die Umkehrung von VAL. STR\$ 1.59 ist "1.59".

1 VAL Zeichenausdruck

VAL ist eine Zeichenfunktion, die den numerischen Wert des Zeichenausdrucks im Argument liefert, und damit die Umkehrung von STR\$. Bei nicht-numerischen Stellen im Argument wird im Ergebnis 0 gesetzt. VAL "1.59" ist 1.59.

Hinweis: Die Zeichenfolgen, die durch die Funktion VAL in numerische Werte umgewandelt werden können, bestehen aus Ziffern (0 bis 9), Symbolen (+ und -) und dem Symbol (E) als Kennzeichnung der exponentiellen Darstellung. Andere Zeichen und Symbole sind nicht eingeschlossen. Wenn eine Zeichenfolge andere Zeichen und Symbole enthält, wird der Teil rechts von diesem Zeichen oder Symbol ignoriert. Leerstellen in einer Zeichenfolge werden normalerweise als nicht vorhanden verarbeitet. Wenn jedoch eine Leerstelle im Exponent (rechts von E) vorhanden ist, werden die Zeichen rechts von der Leerstelle ignoriert.

KAPITEL 10 FEHLERSUCHE

Dieses Kapitel enthält Hinweise darüber, was Sie unternehmen können, wenn Ihr **Computer** nicht tut, was Sie von ihm erwarten. Für jedes Problem schlagen wir eine Reihe von Lösungen vor. Sie sollten jede von ihnen nacheinander ausprobieren, bis Sie das Problem gelöst haben.

Bedienung des Geräts

Wenn:

Sie das Gerät einschalten aber nichts in der Anzeige erscheint.

Die Anzeige zwar funktioniert, aber keine Reaktion auf Tastendruck erfolgt.

Sie eine Rechnung oder Antwort eingeben, aber keine Reaktion erfolgt.

Sie ein BASIC-Programm abarbeiten lassen, etwas angezeigt wird und das Programm dann anhält.

Sie eine Rechnung eingeben und diese in Form eines BASIC-Statements angezeigt wird (Doppelpunkt nach der ersten Zahl).

Sie keinerlei Reaktion auf Tastenbetätigung erhalten.

Dann sollten Sie:

1. Die **BRK ON** -Taste betätigen, um festzustellen, ob sich das Gerät automatisch abgeschaltet hatte, oder **OFF** drücken.
2. Die Batterien wechseln.

1. **CL** drücken, um die Anzeige zu löschen.
2. **CA** (**SHIFT** **CL**) drücken, um die Anzeige zu löschen.
3. Das Gerät aus-und wieder einschalten.
4. Den RESET-Schalter betätigen.

1. Drücken Sie **ENTER** .

1. Drücken Sie **ENTER** .

1. Schalten Sie vom PRO-in den RUN-Modus um.

1. Drücken Sie den RESET-Schalter.

KAPITEL 11

PFLEGE DES COMPUTERS

Um reibungsloses Funktionieren des Computers zu gewährleisten, empfehlen wir die Beachtung folgender Punkte:

- Gehen Sie immer vorsichtig mit dem Computer um, da die Flüssigkristallanzeige aus Glas gefertigt ist.
- Halten Sie den Computer von starken Temperaturschwankungen fern, ebenso von Feuchtigkeit und Staub. Wenn Sie bei warmem Wetter Ihren PKW längere Zeit in der prallen Sonne stehen lassen und sich im Wageninnern hohe Temperaturen aufbauen, kann das Ihren Computer beschädigen.
- Benutzen Sie zur Reinigung des Computers nur ein trockenes, weiches Tuch. Verwenden Sie niemals Lösungsmittel, Wasser oder ein feuchtes Tuch.
- Um das Auslaufen der Batterien zu vermeiden, entfernen Sie diese, wenn Sie den Computer für längere Zeit nicht benutzen wollen.
- Wenn eine Reparatur dieses Gerätes erforderlich sein sollte, wenden Sie sich ausschließlich an einen SHARP- Fachhändler, eine SHARP- Kundendienststelle oder einen SHARP- Reparaturdienst.
- Wenn der Computer starker elektrischer Aufladung oder auch starken elektromagnetischen Wellen ausgesetzt ist, kann er sich unter Umständen "aufhängen" (d. h. auf Tastendruck nicht mehr reagieren). Falls dies auftritt drücken Sie den RESET-Schalter (siehe Fehlersuche).
- Bewahren Sie dieses Handbuch gut auf, falls Sie später einmal etwas nachschlagen wollen.

ANHANG A

Fehlermeldungen

Es Gibt neun verschiedene Fehler-Codes, die im Computer eingebaut sind. Die folgende Tabelle erläutert diese Codes.

**Fehler-
Nummer**

Bedeutung

1 **Syntax-Fehler**

Der Computer kann nicht verstehen, was Sie eingegeben haben. Prüfen Sie die Eingabe auf Dinge wie Semikolon am Ende einer PRINT-Anweisung, falsch geschriebene Wörter und fehlerhafte Anwendungen.

z.B. 3 * /2

Unzulässiger Feldname für die Datenbank.

Beispiel: D1 (,) wurde anstelle von D1\$ (,) spezifiziert.

2 **Rechenfehler**

Hier haben Sie vermutlich eines der drei folgenden Dinge getan:

- 1) Versucht, eine zu große Zahl zu benutzen. Das Rechenergebnis ist größer als 9.999999999E 99.
- 2) Versucht, durch 0 zu teilen.
z.B. 5/0
- 3) Versucht, eine unlogische Rechnung auszuführen.
z.B. LN-30 oder ASN 1.5

3 **Unzulässige Funktion** (DIMensionierungsfehler/Argumentfehler)

Die Feldvariable existiert bereits.

Das Feld wurde spezifiziert, aber nicht dimensioniert.

Der Index des Feldes übersteigt die in der DIM-Anweisung vorgegebene Größe.

z.B. DIM B(256)

Ein Beispiel ist Spezifizierung einer Obergrenze für eine FOR...NEXT Schleife, die grösser als 32767 ist. Grund für den Fehler ist in diesem Fall, daß die Obergrenze für eine FOR...NEXT Schleife in nur zwei Bytes gespeichert wird. Die grösste positive ganze Zahl mit Vorzeichen, die in zwei Bytes untergebracht werden kann, ist 32767.

- 4 **Zu hohe Zeilennummer.**
Hier haben Sie vermutlich folgendes getan:
1) Versucht mit GOTO, GOSUB, RUN, LIST oder THEN usw. eine nicht vorhandene Zeile anzusprechen.
2) Versucht eine zu große Zeilennummer zu verwenden. Die höchstmögliche Zeilennummer ist 999.
- 5 **Verschachtelungs-Fehler**
Die Verschachtelung einer Subroutine übersteigt 10 Stufen.
Die Verschachtelung einer FOR-Schleife übersteigt 5 Stufen.
Die Befehle RETURN ohne GOSUB oder NEXT ohne FOR: Fassungsvermögen des Puffers überschritten.
- 6 **Speicherkapazität überschritten.**
Diese Fehlermeldung erhalten Sie normalerweise, wenn Sie versuchen, ein Feld zu DIMensionieren, das zu groß für die Speicherkapazität ist. Sie kann aber auch auftreten, wenn ein Programm zu lang ist.
- 7 **USING-Formatfehler**
Das bedeutet, daß in einer USING-Anweisung eine unzulässige Formatspezifikation enthalten ist.
- 8 **E/A-Anschluß-Fehler**
Dieser Fehler kann auftreten, wenn Sie einen optionalen Drucker und/oder Kassettenrekorder an den Computer angeschlossen haben. Es bedeutet, daß Sie ein Kommunikationsproblem zwischen dem Gerät und dem Computer haben.
- 9 **Andere Fehler**
Dieser Fehlercode wird angezeigt, wenn der Computer ein Problem hat, das mit den anderen Fehlercodes nicht zu erfassen ist. Eine der häufigsten Ursachen für das Auftreten dieser Meldung ist der Versuch, Daten einer Variablen unter einem bestimmten Namen (z.B. A\$) anzusprechen, während die Daten der Variablen unter einem anderen Namen (d.h.A) abgespeichert sind.
Es würde versucht, eine geschützte Datenbank anzusprechen.

ANHANG B

Auswechseln der Batterien

Bei Verwendung ohne Peripheriegeräte erfolgt die Spannungsversorgung des Computers aus den eingesetzten Batterien. Bei Anschluß am CE-125 oder CE-126P kann die Spannungsversorgung des Computers auch vom CE-125 oder CE-126P erfolgen. Dadurch werden die Batterien des Computers geschont.

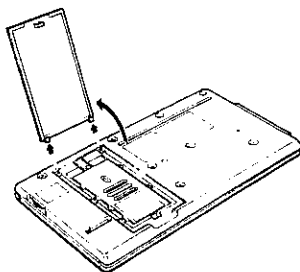
Zeitpunkt zum Auswechseln der Batterien

Wenn das Display bei maximalem Kontrast schwach und schwer ablesbar ist, sind die Batterien entladen und müssen ausgewechselt werden.

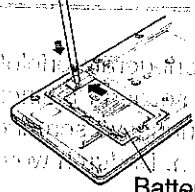
Hinweis: Durch Entnehmen der Batterien gehen die im Computer gespeicherten Daten und Programme verloren. Speichern Sie Daten und Programme, die erhalten bleiben sollen, in einem optionalen Peripheriegerät wie CE-124, CE-125, CE-126P oder CE-123P oder auf einem Datenrecorder (CE-152).

Wechseln Sie die Batterien nach dem folgenden Verfahren aus.

1. Drücken Sie die Taste OFF .
2. Nehmen Sie den Batteriefachdeckel ab.

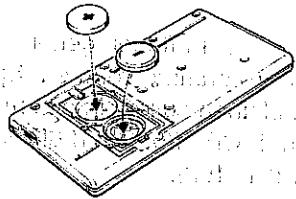


3. Halten Sie die Verriegelung gedrückt und schieben den Batteriehalter in Pfeilrichtung, um ihn abzunehmen.



Batteriehalter

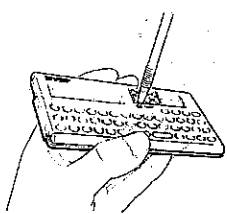
4. Entnehmen Sie die alten Batterien und setzen zwei neue ein. Wischen Sie die Batterien vorher ab und setzen sie so ein, daß die Plus- und Minus-Symbole auf den Batterien mit denen im Batteriefach übereinstimmen.



5. Bringen Sie den Batteriehalter durch Umkehrung von Schritt 3 wieder an.

6. Bringen Sie den Batteriefachdeckel wieder an.

7. Drücken Sie die Taste **ON** und den Rückstellknopf, um den Computer zu initialisieren.



8.

TEL 1 ~ 2, TABLE 1 ~ 8?
DATA BASE

Wenn diese Anzeige nicht erscheint, entnehmen Sie die Batterien und setzen sie nach dem obigen Verfahren erneut ein.

HINWEIS:

- Batterien können durch Erhitzung oder falschem Umgang explodieren.
- Wenn entladene Batterien im Computer belassen werden, können sie durch Auslaufen der Batterieflüssigkeit zu Beschädigungen des Computers führen. Daher sollten entladene Batterien immer entnommen werden.
- Die Batterien werden ab Werk in den Computer eingesetzt. Daher ist es möglich, daß die Batterien nach dem Erwerb des Computers nicht die volle Lebensdauer (etwa 120 Stunden) aufweisen.

WARNUNG: Batterien aus der Reichweite von Kindern fernhalten.

ANHANG C

Erweiterte Funktion der Telefonliste

In der Telefonliste für die Datenbank kann nur eine Telefonnummer pro Name gespeichert werden. Unter Verwendung des BASIC-Befehls DIM können jedoch bis zu vier Telefonnummern gespeichert werden. Auf diese Weise können Sie zusätzlich zur Firmenummer auch noch die Nebenstellennummer oder die private Telefonnummer speichern.

In dem folgenden Verfahren tritt ein Fehler auf, wenn Daten in TEL1 vorhanden sind. In diesem Fall siehe den Abschnitt zum Löschen der Telefonliste und löschen Sie TEL1.

Beispiel: Erweiterung der Telefonliste für TEL1 auf drei Nummern pro Namen.

Speichern von Telefonnummern

SHIFT **DB↔BASIC** (Stellen Sie den Modus BASIC RUN ein.)

DIM D9\$ (0, 3) **ENTER**

Hierdurch werden vier Speicherplätze (0-3) bereitgestellt, einer für den Namen und drei für die Telefonnummern. Setzen Sie die erste Nummer innerhalb der Klammern auf 0. Bei der Eingabe von Daten in die Telefonliste werden automatisch Speicherplätze für die Telefonnummern bereitgestellt.

SHIFT **DB↔BASIC** (Einstellung des Datenbank-Modus)

- | | | |
|----|------------------------------|-----------------------|
| 1) | TEL1 | TEL1 TITLE = ? |
| 2) | Titel ENTER | NAME = ? |
| 3) | Name ENTER | No. = ? |
| 4) | Telefonnummer 1 ENTER | No. = ? |
| 5) | Telefonnummer 2 ENTER | No. = ? |
| 6) | Telefonnummer 3 ENTER | No. = ? |

Führen Sie die Eingabe unter Wiederholung von 3) bis 6) durch.

Abrufen von Telefonnummern

Drücken Sie zum Abrufen von Telefonnummern die Taste **INS >**, während der Name angezeigt wird. Dann können Sie die Telefonnummer 1, Telefonnummer 2 und Telefonnummer 3 abrufen.

Wenn ein Name zu lang ist, so daß er nicht vollständig auf das Display paßt, drücken Sie die Taste **INS >**, um die zweite Hälfte anzuzeigen, und drücken Sie die Taste erneut, um den nächsten Posten abzurufen. Durch Drücken der Taste **CONT. ↓** können Sie zur nächsten Person weitergehen.

ANHANG D

Beziehung zwischen den Daten in der Datenbank und den Basic-Feldvariablen

Der Inhalt von TEL1 wird in D9\$() und der Inhalt von TEL2 in D0\$() gespeichert. Die Inhalte der Tabellen 1 bis 8 werden in D1\$() bis D8\$() gespeichert. Die Telefonliste und die Tabellendaten werden in den unten dargestellten Bereichen gespeichert.

TEL1 (5 Personen) Tabelle 1 (5 Reihen und 2 Spalten)

Titel	Telefonnummer	Titel	Spaltenname 1	Spaltenname 2	Reihen-Gesamtsumme
Name 1	TEL1	Reihenname 1			
Name 2	TEL2	Reihenname 2			
Name 3	TEL3	Reihenname 3			
Name 4	TEL4	Reihenname 4			
Name 5	TEL5	Reihenname 5			
		Spalten-Gesamtsumme			Gesamtsumme

D9\$(0,0)	D9\$(0,1)	D1\$(0,0)	D1\$(0,1)	D1\$(0,2)	D1\$(0,3)
D9\$(1,0)	D9\$(1,1)	D1\$(1,0)	D1\$(1,1)	D1\$(1,2)	D1\$(1,3)
D9\$(2,0)	D9\$(2,1)	D1\$(2,0)	D1\$(2,1)	D1\$(2,2)	D1\$(2,3)
D9\$(3,0)	D9\$(3,1)	D1\$(3,0)	D1\$(3,1)	D1\$(3,2)	D1\$(3,3)
D9\$(4,0)	D9\$(4,1)	D1\$(4,0)	D1\$(4,1)	D1\$(4,2)	D1\$(4,3)
D9\$(5,0)	D9\$(5,1)	D1\$(5,0)	D1\$(5,1)	D1\$(5,2)	D1\$(5,3)
		D1\$(6,0)	D1\$(6,1)	D1\$(6,2)	D1\$(6,3)

Hinweis: ● D9\$(0,1), D1\$(0,3) und D1\$(6,0) sind Nullen.

- Wenn die Telefonliste und Tabellendaten im Datenbank-Modus geschützt sind, können sie auch in BASIC nicht gelesen werden.

Beachten Sie bei der Definition von Datenbank-Feldern mit dem DIM-Befehl folgendes:

1. Setzen Sie die Anzahl der Spalten für die Telefonliste auf 2 bis 5 Spalten (spezifizieren Sie die Werte 1 - 4).

Beispiel: DIM D9\$(0, 4)

↑ 4 ist das maximum.

2. Setzen Sie die Anzahl der Spalten in der Tabelle auf 3 oder mehr (spezifizieren Sie den Wert 2 oder höher).

Beispiel: DIM D1\$ (0, 3)

↑ Spezifizieren Sie 2 oder höher.

ANHANG E

Numerische Genauigkeit des Computers

Genauigkeit bei Berechnungen

Die Anzeige von Ergebnissen erfolgt mit 10 Stellen; für die internen Berechnungen werden 12 Stellen verwendet.

Beispiel: $5/9$ ergibt intern $5.555555555555E-01$ und wird auf die 10. Stelle gerundet, dann ist die Anzeige $5.555555556E-01$

$5/9 * 9$ ergibt intern $4.999999999999E+00$ und wird auf die 10. Stelle gerundet, dann ist die Anzeige 5.

Für die Funktionen wird ein entsprechender Algorithmus verwendet.

Beispiel: $SIN 30$ ergibt intern $5.000000000001E-01$ und wird auf die 10. Stelle gerundet, dann ist die Anzeige 0.5

Dies ist sehr wichtig für logische Ausdrücke. Da der interne Wert in logischen Ausdrücken verwendet wird, ergibt $SIN30 = 0.5$ den Wert Falsch (0).

Wenn Sie daher einen logischen Ausdruck in einer IF- Anweisung verwenden, geben Sie zuerst das Ergebnis in eine Variable ein und führen dann den Vergleich durch. Bei der Zuweisung des Wertes zu einer Variablen wird gerundet.

10: INPUT A

20: B = SIN A

30: IF B = 0.5 THEN ...

Spezielle Begrenzungen

Zusätzlich zu den oben beschriebenen Begrenzungen haben bestimmte Funktionen des Computers eigene spezielle Begrenzungen, die in der folgenden Tabelle aufgeführt sind.

Funktion	Rechenbereich
$y \wedge x$ (y^x)	$-1 \times 10^{100} < x \log y < 100$ $\left(\begin{array}{l} y = 0, x \leq 0: \text{ERROR 2} \\ y = 0, x > 0: 0 \\ y < 0, x \neq \text{ganze Zahl: ERROR2} \end{array} \right) \left(\begin{array}{l} (\text{Ex.}) 0 \wedge 0 \\ 0 \wedge 5 \\ (-4) \wedge 0.5 \end{array} \right)$ <div style="display: flex; justify-content: space-between; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">ENTER</div> → ERROR 2 <div style="border: 1px solid black; padding: 2px;">ENTER</div> → 0. <div style="border: 1px solid black; padding: 2px;">ENTER</div> → ERROR 2 </div> <p>Der Wert von Y kann nur negativ sein, wenn X eine ganze Zahl ist.</p>
SIN x COS x TAN x	$ x < 1 \times 10^{10}$ <div style="display: inline-block; vertical-align: middle; margin-left: 10px;"> $\left. \begin{array}{l} \text{Bei TAN x sind jedoch die folgenden} \\ \text{Fälle ausgeschlossen.} \\ x = 90 (2n - 1) \\ (n: \text{ganze Zahl}) \end{array} \right\}$ </div>
ASN x (SIN ⁻¹ x) ACS x (COS ⁻¹ x)	$-1 \leq x \leq 1$
ATN x (TAN ⁻¹ x)	$ x < 1 \times 10^{100}$
LN x LOG x	$1 \times 10^{-99} \leq x < 1 \times 10^{100}$
EXP x	$-1 \times 10^{100} < x \leq 230.2585092$
\sqrt{x}	$0 \leq x < 1 \times 10^{100}$
& x	$0 \leq x \leq \text{FFFF}$

Andere Funktionen als die oben aufgeführten können nur berechnet werden, wenn x im folgenden Bereich bleibt.

$$1 \times 10^{-99} \leq |x| < 1 \times 10^{100} \text{ und } 0$$

Im obigen Rechenbereich ist der Fehler von Funktionsberechnungen kleiner als ± 1 an der untersten Stelle des angezeigten numerischen Wertes (an der untersten Stelle der Mantisse bei wissenschaftlicher Darstellung).

ANHANG F

Zeichencode-Tabelle

Die folgende Tabelle zeigt die Umwandlungswerte für den Einsatz von CHR\$ und ASC. Die Spalte zeigt das erste hexadezimale Zeichen oder die ersten 4 Binärbits, die Reihe zeigt das zweite hexadezimale Zeichen oder die zweiten vier Binärbits. Die obere linke Ecke jedes Kastens enthält die Dezimalzahl für das Zeichen. Die untere rechte Ecke zeigt das Zeichen. Wird kein Zeichen angegeben, bedeutet es, daß das Zeichen unzulässig ist auf dem Computer.

Zum Beispiel das Zeichen "A" ist dezimal 65 oder hexadezimal 41 oder binär 01000001.

Der Computer erkennt nicht die Codes im schraffierten Bereich. Wird eine Codenummer aus dem schraffierten Bereich eingegeben, tritt ein Fehler ein.

Erste 4 Bits

Zweite 4 Bits

Hex	0	1	2	3	4	5	6	7
Binary	0000	0001	0010	0011	0100	0101	0110	0111
0	0	16	32	48	64	80	96	112
0000	NUL		SPACE	0	@	P	E	p
1	1	17	33	49	65	81	97	113
0001			!	1	A	Q	a	q
2	2	18	34	50	66	82	98	114
0010			"	2	B	R	b	r
3	3	19	35	51	67	83	99	115
0011			#	3	C	S	c	s
4	4	20	36	52	68	84	100	116
0100			\$	4	D	T	d	t
5	5	21	37	53	69	85	101	117
0101			%	5	E	U	e	u
6	6	22	38	54	70	86	102	118
0110			&	6	F	V	f	v
7	7	23	39	55	71	87	103	119
0111			□	7	G	W	g	w
8	8	24	40	56	72	88	104	120
1000			(8	H	X	h	x
9	9	25	41	57	73	89	105	121
1001)	9	I	Y	i	y
A	10	26	42	58	74	90	106	122
1010			*	:	J	Z	j	z
B	11	27	43	59	75	91	107	123
1011			+	;	K	√	k	
C	12	28	44	60	76	92	108	124
1100			,	<	L	¥	l	
D	13	29	45	61	77	93	109	125
1101			-	=	M	π	m	
E	14	30	46	62	78	94	110	126
1110			.	>	N	∧	n	
F	15	31	47	63	79	95	111	127
1111			/	?	O	-	o	

ANHANG G

Verwendung von Programmen, die für andere PC-Modelle geschrieben wurden

Unter geringen Programm-Modifikationen können auf dem PC-1150
Programme von den folgenden Computern laufen:

PC-1246, PC-1246S, PC-1247, PC-1248

Beachten Sie jedoch folgende Punkte:

- 1) Wenn Befehle verwendet werden, über die der PC-1150 nicht verfügt, tritt während der Ausführung ein Fehler auf. Programme, die solche Befehle enthalten, müssen geändert werden. Diese Befehle werden in der Programmliste mit dem Symbol "~" angezeigt.
- 2) Geschützte Programme können nicht von Band geladen werden.
- 3) Daten können nicht verwendet werden.

ANHANG H

Technische Daten

Modellname:	PC-1150			
Display:	16-stellige Flüssigkristallanzeige mit 5 x 7 Matrix			
CPU:	CMOS 4 Bit Mikroprozessor			
System-ROM:	ca. 20 KByte			
Speicherkapazität:	Systembereich:	ca. 0,5 KByte		
	Datenbereich:	208 Byte		
	Programm/Daten- Bereich:	9470 Byte		
Editierfunktionen:	Cursor-Steuerung	<table border="1"><tr><td><DEL</td><td>, INS ></td></tr></table>	<DEL	, INS >
	<DEL	, INS >		
	Einfügung	<table border="1"><tr><td>SHIFT</td><td>INS ></td></tr></table>	SHIFT	INS >
	SHIFT	INS >		
Löschung	<table border="1"><tr><td>SHIFT</td><td><DEL</td></tr></table>	SHIFT	<DEL	
SHIFT	<DEL			
Zeilen-Scrolling	<table border="1"><tr><td>CONT. ↑</td><td>, CONT. ↓</td></tr></table>	CONT. ↑	, CONT. ↓	
CONT. ↑	, CONT. ↓			
Speicherschutz:	Daten bleiben nach Ausschalten erhalten			
Abmessungen:	135(B) x 70(T) x 11(H) mm			
Gewicht:	85 g (mit Batterien)			
Spannungsversorgung:	6 V Gleichstrom, 2 Lithiumbatterien (CR2032)			
Leistungsaufnahme:	0,05 W			
Betriebstemperatur:	0 bis 40°C			
Batterie-Lebensdauer:	ca. 120 Stunden bei Dauerbetrieb (bei 10 Minuten Programmausführung und 50 Minuten Anzeige pro Stunde bei 20°C), abhängig von verwendeter Batteriesorte und Betriebsarten			
Zubehör:	Tastaturabdeckung, 2 Lithiumbatterien (im Rechner eingesetzt), Bedienungsanleitung			

Datenbank-Spezifikationen

- Telefonliste:** Speicherung und Anzeige von Namen und Telefonnummern (sequentielle und direkte Suche)
- Name und Telefonnummer:** bis zu 16 Zeichen
ca. 293 Eintragungen
(nur eine Telefonnummer pro Eintragung)
- Tabellen:** Summenbildung/Sortieren für Reihen/Spalten
Daten: bis zu 16 Zeichen pro Element
Kapazität: ca. 591 Zellen, 5 Tabellen mit 13 Reihen und 9 Spalten enthalten 585 Zellen.
- Die oben angegebene Anzahl der Datenelemente setzt voraus, daß nur die relevanten Datenelemente registriert werden.

Rechner-Spezifikationen

- Rechenstellen:** 10 Stellen (Mantisse) + 2 Stellen (Exponent)
- Grundrechenarten:** Addition, Subtraktion, Multiplikation und Division
- Funktionen:** trigonometrische Funktionen, inverse trigonometrische Funktionen, Vorzeichen-Funktionen, Absolutwert, Logarithmus, Exponent, Potenzierung, Quadratwurzel, Runde auf ganze Zahl, Pi und Winkelumwandlung.

Programmierspezifikationen

- Programmiersprache:** BASIC

ANHANG I

Beispielsprogramme

1. Telefonlisten-Druckprogramm

Dieses Programm druckt die Daten in TEL1.

Eingabe und Ausführung des Programms

Stellen Sie den Computer in den PRO-Modus und geben das Programm ein. Für die Eingabe von Programmen siehe Seite 52. Die Doppelpunkte (:) nach den Zeilennummern brauchen nicht eingegeben zu werden. Drücken Sie für Ausführung des Programms die Taste **DB↔BASIC**, um den RUN-Modus zu aktivieren.

Schließen Sie einen optionalen Drucker an und bereiten Sie alles zum Drucken vor (Drucker einschalten, Papier einsetzen usw.). Durch Drücken von **FUNC** **A** wird der Inhalt von TEL1 ausgedruckt. Nach dem Drucken tritt ein Fehler auf, und das Programm stoppt.

1. Wenn kein Drucker verwendet werden soll, löschen Sie die unterstrichenen Teile in den Zeilen 10, 20 und 50. Dann stoppt das Programm nach Anzeige der Daten. Drücken Sie zum Fortsetzen die Taste **ENTER**. Das Programm endet, wenn ein Fehler auftritt.
2. Die Telefonliste kann zum Aufnehmen von bis zu 4 Telefonnummern erweitert werden (siehe ANHANG C). Wenn zwei Telefonnummern vorhanden sind, ändern Sie $K=1$ in Zeile 10 zu $K=2$. Für drei Telefonnummern ändern Sie $K=1$ zu $K=3$.
3. Zum Drucken des Inhaltes von TEL2 ändern Sie $D9\$(0,0)$ in Zeile 20 zu $D0\$(0,0)$ sowie $D9\$(J,I)$ in den Zeilen 40 und 50 zu $D0\$(J,I)$.

Programmliste

```
10: 'A': PRINT = LPRINT
   : K=1: J=0
20: PRINT "TITLE="; D9$(0
   ,0): PRINT ""
30: J=J+1: FOR I=0 TO K
40: IF I=0 THEN PRINT D9
   $(J,I): GOTO 60
50: PRINT " "; D9$(
   J,I)
60: NEXT I: GOTO 30
```

Beispielsausdruck

Für
Telefonnummern-Teil
mit einer
Telefonnummer

```
TITLE=Business
J.SMITH/LONDON
   01-643-4431
M.BERRY
   212-432-0012
M.JONES
   212-758-0354
Y.SAITO/SHARP
   03-260-1161/315
```

Für
Telefonnummern-Teil
mit zwei
Telefonnummern

```
TITLE=Business
J.SMITH/LONDON
   01-643-4431
   1111
M.BERRY
   212-432-0012
   2222
M.JONES
   212-758-0354
   3333
Y.SAITO/SHARP
   03-260-1161/315
   4444
```

SHARP CORPORATION

OSAKA, JAPAN

1987 © SHARP CORPORATION
PRINTED IN JAPAN/IMPRIMÉ AU JAPON
8B2T(TINSG1141ECZZ)®