

```
*****
*
*   T   E   I   L       5
*
*****
*
*   Z   A   H   L   E   N
*
*   U   N   D   R   E   C   H   E   N   -
*
*   O   P   E   R   A   T   I   O   N   E   N
*
*****
```

```
*   EINLEITUNG
*   *****
*   ZAHLEN UND
*   GRUNDRECHENFUNKTIONEN
*   *****
*   AUSFÜHRUNG VON
*   RECHENOPERATIONEN
*   *****
*   WEITERE BEFEHLE ZUM
*   BILDSCHIRMAUSDRUCK
*   *****
*   VARIABLE
*   *****
```

* 5.1 EINLEITUNG *

Um die mathematischen Fähigkeiten des COMMODORE 116 zu verstehen, muß man kein Mathematikprofessor sein. Zusätzlich zu den vier Grundrechenarten Addition, Subtraktion, Multiplikation und Division ist der COMMODORE 116 auch für fortgeschrittene Rechenoperationen, wie Quadratwurzelziehen oder Sinuskurvenberechnungen, einzusetzen.

In diesem Teil des Handbuchs werden Sie die unterschiedlichsten Variablentypen und ihren richtigen Einsatz kennenlernen. Der COMMODORE 116 verarbeitet sowohl ganze Zahlen als auch Gleitkommazahlen mit bis zu 38 Stellen in Exponential-Schreibweise, wobei allerdings nur die ersten 9 Stellen berücksichtigt werden. Die Berechnungen können im DIREKT-Modus oder eingebunden in ein Programm erfolgen. Abschließend erhalten Sie eine kurze Anleitung, wie Sie selbstdefinierte Funktionen auf dem COMMODORE 116 implementieren können.

* 5.2 ZAHLEN UND GRUNDRECHENFUNKTIONEN *

Der COMMODORE 116 ist auch wie ein Taschenrechner einsetzbar. Neben den Standard-Operatoren für Addition ('+') und Subtraktion ('-') verwendet der COMMODORE 116 - wie viele andere Computer - für Multiplikation das 'Sternchen' (Asterisk, '*') und für Division bzw. Bruchstrich den Schrägstrich ('/').

Die Grundrechenfunktionen stehen im DIREKT-Modus und im PROGRAMM-Modus zur Verfügung. Im PROGRAMM-Modus werden die Rechenoperationen z.B. in PRINT-Befehlen ohne Anführungszeichen geschrieben.

ARITHMETISCHE OPERATOREN

LOGISCHE VERGLEICHOPERATOREN

Addition	+	Größer als	>
Subtraktion	-	Kleiner als	<
Division und Bruch	/	Gleich	=
Multiplikation	*	Größer oder gleich	>= oder =
Exponent	↑	Kleiner oder gleich	<= oder =
		Ungleich	<> oder ><

ANMERKUNG: Zahlen dürfen bei der Computereingabe kein Komma (',') enthalten - weder als (englische) Tausender-Trennung noch als Dezimalpunktersatz (21,00). Bei der Computereingabe bedeutet das Komma die Trennung zweier Zahlen; daher Eingabe von Zahlen so: 30359 oder so: 21.00

BRÜCHE UND DEZIMALSTELLEN

Ein Bruch kann so: .5

oder so dargestellt werden: 1/2

Im zweiten Fall wird der Computer das Ergebnis errechnen. Wenn die Rechenoperation in einen PRINT-Befehl eingebettet ist, wird das Ergebnis immer im Dezimalmodus bzw. als ganze Zahl ausgegeben.

Beispiel:

PRINT 139/493 + 5 und Taste <Return>:
5.28194726 ist das Ergebnis am Bildschirm.

DIE ' π '-TASTE *****

Ein Beispiel für eine Zahl, die sich nicht exakt als Bruch oder als Dezimalzahl mit endlich vielen Stellen schreiben läßt, ist die Zahl π (sprich:PI), die das Verhältnis des Kreisumfanges zu seinem Durchmesser ausdrückt. Die Zahl π ist mit einer Genauigkeit von acht Stellen nach dem Dezimalpunkt im COMMODORE 116 gespeichert und über die Taste < π > abrufbar. Ein Beispiel:

PRINT π /374

und Taste <Return>:

8.39998036E-03

erscheint auf dem Bildschirm.

WISSENSCHAFTLICHE NOTATION *****

Was bedeutet eigentlich der Anhang 'E-03' in obigem Rechenbeispiel? Der COMMODORE 116 kann Zahlen im Bereich von -999 999 999 bis +999 999 999 direkt verarbeiten. Über neun Stellen hinausgehende Zahlen werden automatisch auf die ersten neun signifikanten Stellen (d.h. ohne führende Nullen) reduziert und in wissenschaftlicher Schreibweise angezeigt. Machen Sie den Versuch; der COMMODORE 116 übersetzt jede Eingabe mühelos! Die wissenschaftliche Schreibweise ist oft nützlich, da der Computer auch größere Zahlen in einer überschaubaren Stellenzahl auswirft. Ein Beispiel: Die Zahl 1 985 054 780 in wissenschaftlicher Schreibweise:

1.98505478E+09

Nur eine Stelle links vom Dezimalpunkt.

! Diese Zahl gibt an, um wieviele Stellen der Dezimalpunkt in dieser Schreibweise nach links verschoben wurde.

Zahlen, die kleiner als Ein-Hundertstel (' $< 1/100$ ') sind, werden vom COMMODORE 116 ebenfalls in wissenschaftlicher Schreibweise ausgegeben, mit dem Unterschied, daß der Exponent negativ erscheint, d.h. der Dezimalpunkt wird um entsprechend viele Stellen nach r e c h t s verschoben. Zum Beispiel: PRINT .0003359 ergibt 3.359E-04.

Beispiele zur wissenschaftlichen Schreibweise:

20	in wissenschaftlicher Notation	:	2E+01
105000	- " -	:	1.05E+05
.0666	- " -	:	6.66E-02

 * 5.3 AUSFÜHRUNG VON RECHENOPERATIONEN *

Um eine Rechenoperation auszuführen, ist PRINT und anschließend das mathematische Problem (o h n e Anführungszeichen!) einzugeben:

```
10 PRINT 1+2, 2-1
20 PRINT 2*2, 4/2
RUN
3          1
4          2
```

Anders als bei Textkonstanten (in Anführungszeichen) deckt sich die Ausgabe am Bildschirm nicht Zeichen für Zeichen mit den Parametern des PRINT-Befehls. Vielmehr errechnet der Computer selbständig die Lösung und druckt nur das Ergebnis aus. Wenn Sie also mit der PRINT-Anweisung rechnen wollen, so brauchen Sie bloß die Anführungszeichen wegzulassen.

Nächstes Experiment:

NEW und Taste <Return>

READY.

10 PRINT "2001/2010" und Taste <Return>

20 PRINT 2001 - 17 und Taste <Return>

RUN und Taste <Return>

2001/2010

1984 Die Leerstelle links von '1984' steht für
das Vorzeichen, das bei PLUS entfällt.

READY.

Die PRINT-Anweisung in Zeile 10 befindet sich zwischen Anführungszeichen, daher wird der dazwischen geschriebene Teil genauso am Bildschirm wiedergegeben. Es wird hierbei auch, im Gegensatz zu Zeile 20, keine Stelle für das Vorzeichen freigelassen. Im nächsten Beispiel ändern Sie Zeile 10 wie folgt und löschen Zeile 20:

10 PRINT "2001/2010=";2001/2010 Den Strichpunkt (Semikolon) nicht vergessen!

20 und Taste <Return>

RUN und Taste <Return>

2001/2010= .995522388 Die Leerstelle vor dem Dezimalpunkt steht wieder für das Vorzeichen.

Sollen Text und Ergebnis der Rechenoperation gemeinsam am Bildschirm erscheinen, so ist die Schreibweise entsprechend obigem Beispiel (Zeile 10) zu wählen.

VORRANGORDNUNG BEI BERECHNUNGEN

Es ist möglich (und zulässig), pro Zeile auch mehrere Berechnungen durchzuführen. Ein Beispiel:

PRINT 200*50+5 und Taste <Return>
10005 ... haben Sie mit diesem Ergebnis gerechnet?
Versuchen Sie als nächstes diese Eingabe:
PRINT 50+5*200 und Taste <Return>
1050 ... erwarteten Sie dieses Ergebnis?

Der COMMODORE 116 führt Berechnungen in einer ganz bestimmten Reihenfolge aus (Vorrangordnung der Algebra). Rechenoperationen werden von links nach rechts (in der Zeile) ausgeführt; innerhalb dieser Generalregel werden einige Rechenoperationen vorgezogen. Nachstehend diese Vorrangordnung:

- (1.) Prüft der COMMODORE 116 auf Negativ-Zahlen
- (2.) Werden sämtliche Exponential-Berechnungen ausgeführt
- (3.) Rechnet der COMMODORE 116 alle Multiplikationen und Divisionen (von links nach rechts)
- (4.) Rechnet der COMMODORE 116 alle Additionen und Subtraktionen (von links nach rechts)

ANMERKUNG: Im COMMODORE 116 haben Berechnungen in Klammern Priorität vor allen anderen Berechnungen. Auch mehrere Klammern lassen sich verschachteln:
 $36 * (12 + (A/3))$.
Die Berechnung der Klammern erfolgt von innen nach außen.

Eine Verbesserung der Übersicht bringt die Verpackung negativer Zahlen in Klammern. Ein Beispiel:
 $45 * (-5)$ ist übersichtlicher, obwohl der COMMODORE 116 diese Berechnung auch ohne Klammern richtig interpretiert.

```
*****  
* 5.4 WEITERE BEFEHLE ZUM BILDSCHIRMAUSDRUCK *  
*****
```

Sicher ist Ihnen aufgefallen, daß Komma (',') und Semikolon (';') ein unterschiedliches Ausdruckbild am Bildschirm bewirken. Der COMMODORE 116 interpretiert diese Interpunktionszeichen für den Zwischenraum beim Ausdruck.

Der Unterschied ist ähnlich den bereits beschriebenen Auswirkungen der Anführungszeichen bei Rechenoperationen und Text (sogenannten 'TEXT STRINGS') in Verbindung mit dem PRINT-Befehl. Ein Beispiel soll dies verdeutlichen:

```
NEW          und Taste <Return>  
READY.  
10 PRINT "O","K"  
20 PRINT "O";"K"
```

Beachten Sie, daß sich in beiden Programmzeilen die Interpunktionszeichen a u ß e r h a l b der Anführungszeichen (STRINGS) befinden.

```
RUN          und Taste <Return> bewirkt folgendes:  
O           K  
OK
```

Wie kommt dieser Unterschied im Ausdruck zustande, obwohl sich Zeile 10 und 20 nur im Interpunktionszeichen unterscheiden?

Werden STRINGS (zwischen Anführungszeichen stehende Zeichen) in einer PRINT-Anweisung durch Komma getrennt, so werden die Einzelstrings beim Ausdruck durch mehrere Leerstellen getrennt.

Werden STRINGS (zwischen Anführungszeichen stehende Zeichen) in einer PRINT-Anweisung durch Semikolon getrennt, so werden die Einzelstrings zusammenhängend ausgedruckt.

Wir rekapitulieren: Der Bildschirm des COMMODORE 116 hat 40 Zeichen pro Zeile und jede dieser Zeilen ist (unsichtbar) in vier Druck-Zonen von je 10 Zeichen eingeteilt. Werden nun PRINT-Parameter durch ein Komma getrennt, so druckt der COMMODORE 116 den ersten STRING in die erste Druckzone und den mit Komma getrennten zweiten STRING in die zweite Zone usw. Das Komma wirkt somit wie ein gesetzter Tabulator auf der Schreibmaschine.

	DRUCK-ZONE 1	DRUCKZONE 2
SPALTE:	1 2 3 4 5 6 7 8 9 10	11 12 13 14 15 16 17 18 19 20
	O	K
	DRUCK-ZONE 3	DRUCKZONE 4
SPALTE:	21 22 23 24 25 26 27 28 29 30	31 32 33 34 35 36 37 38 39 40

DRUCKZONENGESTALTUNG BEIM PRINTBEFEHL MIT KOMMA

Umfaßt der PRINT-Befehl mehr als vier durch Kommata getrennte Strings, so setzt der COMMODORE 116 automatisch in der nächsten Zeile fort.

Ein Beispiel:

```
PRINT "A","B","C","D","E","F"    und Taste <Return>
                                   bringt folgenden Ausdruck

      1      11      21      31      40      SPALTE
ZEILE 1  A          B          C          D
      2  E          F
```

Trennen Sie die PRINT-Befehle durch Semikolons, dann werden die Druckzonen vom COMMODORE 116 ignoriert, und der Ausdruck erfolgt zusammenhängend.

Ein Beispiel:

```
PRINT "A";"B";"C";"D";"E";"F"    und Taste <Return>
                                   bringt folgenden Ausdruck

      1      11      21      31      40      SPALTE
ZEILE 1  ABCDEF
```

Ist der erste zu druckende STRING länger als 9 Zeichen und der zweite Ausdruck mit einem Komma getrennt, dann passiert folgendes:

```
PRINT "ABCDEFGHIJKL","M"          und Taste <Return>
                                   bringt folgenden Ausdruck

      ZONE 1      ZONE 2      ZONE 3      ZONE 4
      1      11      21      31      40      SPALTE
ZEILE 1  ABCDEFGHIJKL          M
```

Was geschieht, wenn für die Eingabe einer Programmzeile die 40 Zeichen einer Zeile nicht ausreichen. Ein Beispiel:

```
10 PRINT "DER COMMODORE 116 IST EIN HANDLICHER, KLEINER COMPUTER"
```

Beim Eintippen dieser Programmzeile reichen die 40 Zeichen einer Zeile nicht aus. Schreiben Sie dennoch weiter; beim COMMODORE 116 springt der CURSOR automatisch am Ende einer Zeile an den Anfang der nächsten Zeile. Programmzeilen können auf diese Weise das max. übliche Maß von 80 Zeichen Länge erreichen. Dabei ist zu beachten, daß sich der CURSOR innerhalb einer der beiden Zeilen befindet, wenn - zur Übergabe an den Computer - die Taste <Return> gedrückt wird. Für den Computer ist nicht das Ende der Zeile, sondern das <RETURN>-Signal das entscheidende Kriterium.

Mit 'RUN' und Drücken der Taste <Return> wird dieser String auch ausgedruckt. Der dabei über 40 Zeichen hinausgehende Teil wird automatisch in der darauffolgenden Zeile ausgedruckt. Auch hier empfehlen sich einige Übungsbeispiele - auch mit mehr als 80 Zeichen.

```
*****  
* 5.5 VARIABLE *  
*****
```

Das Beispiel $36 \cdot (12 + (A/3))$ zeigt u.a. eine der wesentlichen Eigenschaften des COMMODORE 116. Den Einsatz eines Buchstabens in einer mathematischen Formel - den Einsatz einer VARIABLEN. Eine VARIABLE stellt stets einen Wert dar:

```
10 A = 3
```

```
20 PRINT "ERGEBNIS: "; A * 4
```

```
RUN
```

und Taste <Return>, dann erscheint
am Bildschirm:

```
ERGEBNIS: 12
```

Es sind drei VARIABLEN-Typen, die es zu unterscheiden gilt:

VARIABLE	SYMBOL	BESCHREIBUNG	BEISPIEL	WERT-BEISPIELE
FLIESS- KOMMA		Ganze oder Dezimalzahlen	X, AB, T4	23.5, -12, 1.3E+02
INTEGER	%	Ganze Zahlen	X%, A1%	15, 102, -32700
TEXT- STRING	\$	Buchstaben, Zahlen und alle anderen Zeichen zwischen An- führungszeichen	X\$, MS\$	"SUMME:", "DM" "1. TAG", "COMMODORE"

Eine VARIABLE, die als ganze Zahl betrachtet werden soll, wird mit dem Prozentzeichen ('%') gekennzeichnet.

Eine VARIABLE, die Text enthält, wird mit dem Dollarzeichen ('\$') gekennzeichnet.

Fehlen diese Zusätze, so interpretiert der COMMODORE 116 die VARIABLE als Fließkomma-Variable, d.h. als reellen Zahlenwert ('REAL'). Die Integer-Variablen stellen eine Untergruppe der Fließkomma-Variablen dar; sie enthalten keinen Dezimalpunkt.

Es ist wichtig, daß immer der richtige Variablen-Typ zur Anwendung kommt. Wollen Sie z.B. ein Textwort einer Integer-Variablen zuordnen, so wird dies nicht funktionieren. Das nachstehende Beispiel wird Ihnen die Unterschiede demonstrieren.

```
10 PRINT "GEBEN SIE EINE ZAHL EIN"
20 INPUT X%
30 PRINT "ICH LESE IHRE ZAHL ALS";X%
40 PRINT "GEHT PRIMA!"
50 END
```

Mit INPUT übernimmt
der COMMODORE 116
Werte von außen.

Starten Sie dieses Programm mit 'RUN' und versuchen Sie die nachstehenden Werte (nach jedem Neu-Start) bei Aufforderung einzutippen (Taste <Return> nicht vergessen).

```
.043
10
EIN VIERTEL
```

NUMERISCHE FUNKTIONEN *****

Die Computersprache BASIC des COMMODORE 116 enthält, wie die wissenschaftlichen Rechner auch, eine Vielzahl numerischer Funktionen (wie z.B. SINUS, COSINUS, TANGENS, usw.).

Die Mehrzahl dieser Funktionen wird mit Eingabe des Funktionsnamens und des zu rechnenden Zahlenwertes (in Klammern) berechnet:

FUNCTION(X)

Um zum Beispiel den Sinus einer Variablen zu finden, genügt es zu schreiben:

```
PRINT SIN(X),
```

wobei X jeden gewünschten Zahlenwert annehmen kann. Der Winkel ist übrigens bei allen Winkelfunktionen im 'Bogenmaß' anzugeben (siehe bei SIN im LEXIKON).

Numerische Funktionen lassen sich auch in Programmzeilen einbinden, wie das folgende Beispiel zeigt:

```
10 FOR X = 1 to 5
20 PRINT "DIE QUADRATWURZEL AUS";X;"IST"; SQR(X)
30 NEXT X
```

Beachten Sie auch die vollständige Aufzählung aller im COMMODORE 116 implementierten Numerischen Funktionen im BASIC 3.5 Lexikon.

SELBSTDEFINIERTER FUNKTIONEN *****

Einen sehr wirkungsvollen Einsatz der mathematischen Fähigkeiten des COMMODORE 116 stellt die Möglichkeit dar, eigene Funktionen zu definieren. Diese selbst-definierten Funktionen können sehr hilfreich bei umfassenden Berechnungen sein. Das Prinzip besteht darin, daß der Benutzer seine eigene Formel entwickelt und der Computer die Rechenwerte dazu liefert.

Um z.B. die Fläche eines Kreises zu berechnen, definiert der Benutzer folgende Funktion, die in der Folge nur noch die Eingabe der 'X'-Werte benötigt:

```
10 DEF FNF(X)=2* $\pi$ *X*X
```

Mit dieser Funktion läßt sich für jeden Wert 'X' die Fläche des Kreises mit Radius X errechnen, wobei 'FNF' der Name dieser definierten Funktion ist. Im Anhang finden Sie eine Reihe mathematischer Funktionen, die nicht im BASIC 3.5 enthalten sind, und auf diese Weise dennoch zur Anwendung kommen können.