

---

# Der Heimcomputer robotron Z 9001



## 1. Was ist ein Heimcomputer?

Als Heimcomputer werden kleine, im allgemeinen sehr preiswerte Rechner bezeichnet, die in jüngster Zeit auch als Konsumgut in den Heimbereich Einzug halten. Sie können über ein Antennenkabel mit einem Fernsehgerät verbunden werden und sind dann sehr einfach als Computer oder auch zur Durchführung von Fernsehspielen nutzbar.

Heimcomputer besitzen eine Tastatur, über die Zahlen, Buchstaben und Zeichen eingegeben werden können, die der Rechner entsprechend ihrer Bedeutung verarbeitet. Auf dem Fernsehschirm werden alle dem Computer eingegebenen Befehle und Daten ebenso angezeigt wie die berechneten Ergebnisse und sonstige Informationen.

Für einfache Nutzungsfälle wird ein Heimcomputer in der Programmiersprache BASIC programmiert. Diese Sprache existiert in sehr vielen Varianten, die auf den verschiedenen Computer-Typen jedoch nur geringfügig voneinander abweichen. In den meisten Fällen wurde eine Anpassung der Sprache an die Möglichkeiten des konkreten Rechners vorgenommen.

Mit BASIC können, ähnlich wie beim Taschenrechner, einfache Rechenoperationen ausgeführt werden, wobei Aufgabe und Ergebnis am Bildschirm

sichtbar sind. Viel wichtiger ist aber die Tatsache, daß auch Programme eingegeben und anschließend beliebig oft abgearbeitet werden können.

Zur Speicherung dieser Programme über einen längeren Zeitraum dient dem Heimcomputer ein gewöhnliches (Audio-) Kassettenmagnetbandgerät, welches über ein Diodenkabel an den Rechner anschließbar ist. Durch spezielle Kommandos lassen sich die in den Rechner eingegebenen Programme auf Magnetband ausgeben, dort speichern und später wieder einlesen.

An viele Geräte können außerdem noch einfache Drucker angeschlossen werden, so daß auch ein Ausdrucken der Programme und Berechnungsergebnisse möglich wird. Häufig sind Spielhebel zur Steuerung von Bewegungsabläufen anschließbar.

BASIC-Programme für den Heimcomputer kann man auf Magnetbandkassetten kaufen oder aber auch selbst erstellen. Diese Programme können sehr unterschiedlichen Inhalts sein. Neben einfachen Rechenprogrammen für Aufgaben aus dem Schulbereich sind wissenschaftlich-technische Berechnungen für ingenieurtechnische Anwendungen ebenso denkbar wie Spielprogramme, z.B. in Form von Logik- und Reaktionsspielen, oder Programme zur Speicherung von Informationen und Statistiken.



Bild 1. Heimcomputer mit Fernsehgerät und Kassettengerät

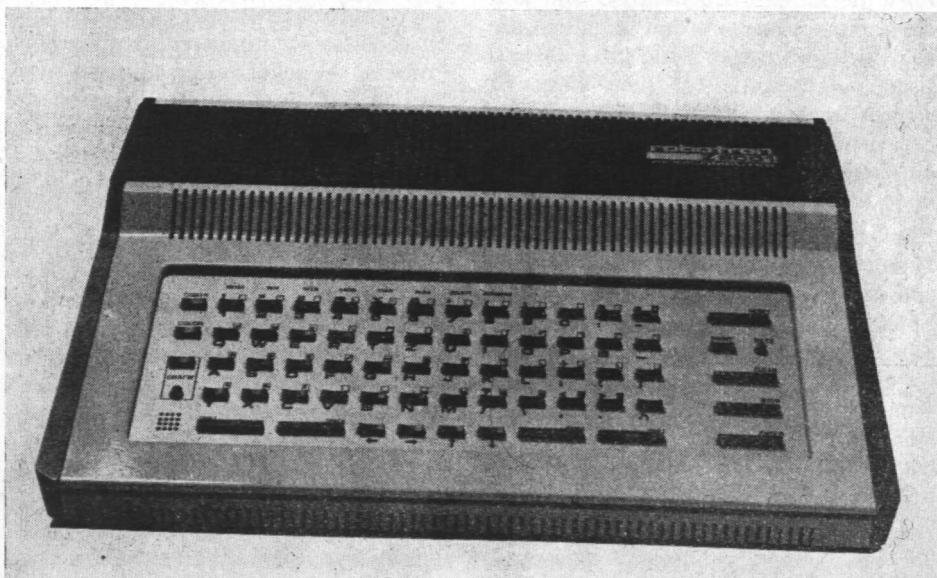


Bild 2. Tastatur

Besondere Bedeutung haben Heimcomputer für die Unterstützung der Ausbildung, zum Vertrautwerden mit der Rechentechnik. So lassen sich viele Unterrichtsaufgaben mit einem Heimcomputer anschaulich lösen und Gesetzmäßigkeiten aus vielen Fachgebieten demonstrieren.

Nicht zuletzt stehen dem technisch interessierten Amateur bei einigen Heimcomputern viele Möglichkeiten offen, selbstgebaute elektronische Baugruppen an den Computer anzuschließen und durch ein entsprechendes Programm zu steuern.

Der Heimcomputer robotron Z 9001 entspricht in seiner Grundgestaltung den oben geschilderten Prinzipien.

Seine speziellen technischen Daten und Eigenschaften sind nachfolgend erläutert:

- 49 Tasten für die Eingabe von Groß- und Kleinbuchstaben, Ziffern und Zeichen
- 16 Tasten zur Kommandoausführung bzw. zur Cursorsteuerung (vgl. Bild 2)
- 128 alphanumerische bzw. Steuerzeichen
- 128 Grafikzeichen (teilweise von der Tastatur aus ansprechbar)
- Darstellung von 24 Zeilen und 40 Spalten auf dem Bildschirm
- 16-KByte-RAM-Speicher (für den Anwender verfügbar). Damit sind etwa 800 bis 1000 BASIC-Programmbefehle speicherbar.

Das Grundgerät bietet die Möglichkeit, 4 Zusatzmodule zu stecken, die funktionelle bzw. Speicher-Erweiterungen beinhalten können (vgl. Bild 3). Damit ist es möglich, den BASIC-Interpreter sowohl von einer Magnetbandkassette zu laden, als auch als Modul zusätzlich zu stecken.

Neben dem BASIC-Modul gibt es unter anderem folgende Erweiterungsmög-

lichkeiten, die im Handel als Zubehör angeboten werden:

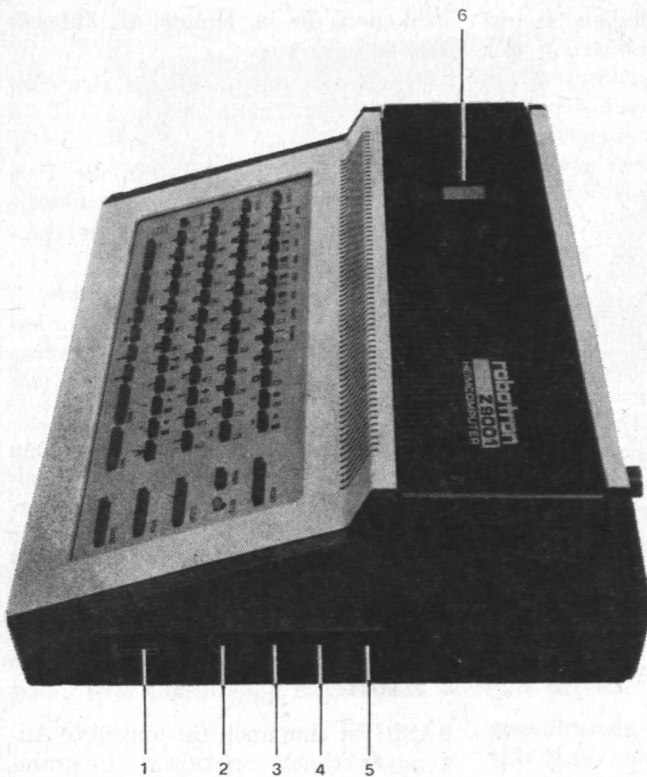
- Drucker-Modul (z.B. für den Anschluß des Thermodruckers TD 40/K 6303)
- ASSEMBLER-Modul (für die Programmierung in K1520-Assembler)
- Musik-Modul (3 Oktaven, frei programmierbar)
- Speicher-Modul (16-KByte-RAM)
- Farb-Modul (Je 8 Farben für Vordergrund und Hintergrund wahlweise. Dieser Modul wird in das Gerät eingebaut).

Über weitere Steckmöglichkeiten an der rechten Geräteseite können Spielhebel (für Bildschirmspiele) und verschiedene andere elektronische Geräte oder Baugruppen angeschlossen und gegebenenfalls gesteuert werden.

## 2. Die Programmiersprache BASIC des Z 9001

BASIC ist eine auch für ungeübte Anwender schnell erlernbare Programmiersprache, die vor allem den Dialog mit dem Rechner, z.B. die Einflußnahme auf die Programmabarbeitung, sehr gut unterstützt.

Mit Hilfe des BASIC des Z 9001 können sowohl einfache mathematische Operationen (im sogenannten „Taschenrechnermodus“) sofort ausgeführt, als auch Programme eingegeben, abgearbeitet und auf Magnetbandkassette gespeichert werden. Auch die Ausgabe von Programmen, Rechenergebnissen und anderen Informationen auf einen Drucker ist möglich. Ebenso können die Heimcomputer-spezifischen Ergänzungsmodule (Farbmodul, Musikmodul, Spielhebel u.a.) sowie der für die Gestaltung von Spielen bedeutsame Grafik-Zeichensatz (128 spezielle Grafiksymbbole) vom BASIC einfach angesprochen werden. Weiterhin lassen



- 1 – Anschluß für 8 E/A-Kanäle
- 2 – H/F-Anschluß
- 3 – Anschluß für Tonband
- 4 – Spielhebel 1
- 5 – Spielhebel 2
- 6 – 4 Modulsteckplätze über K 1520-BUS (abgedeckt)

Bild 3. Ansicht des Z 9001 mit Modulsteckplätzen und E/A-Anschlußbuchsen (Modellfoto)

sich digitale Informationen von angeschlossenen Meßgeräten abrufen und weiterverarbeiten und auch Steuerungssignale an entsprechende Geräte übertragen.

Das BASIC des Z 9001 ist in seinem Aufbau (Befehls- und Kommando-Vorrat, syntaktische Struktur) dem international weitverbreiteten Standard angeglichen, so daß eine Übertragung von Programmen vieler anderer Rechner auf den Z 9001 in der Regel leicht fällt. Wir wollen uns nun einen Überblick über die wichtigsten Anweisungen und Kommandos und ihre Wirkungsweise verschaffen.

### Einfache Rechenoperationen

Wie bereits erwähnt, können im BASIC ähnlich wie mit einem Taschenrechner einfache mathematische Rechenoperationen ausgeführt werden.

Dazu stehen die Grundoperationen  $+$ ,  $-$ ,  $\times$ ,  $/$  und  $\wedge$  (Potenzsymbol), sowie die Standardfunktionen

- ABS (X) – Absoluter Betrag
- ATN (X) – Arcustangens
- COS (X) – Cosinus (X in Bogenmaß)
- EXP (X) – Exponentialfunktion  $e^x$
- INT (X) – Ganzer Teil
- LN (X) – Natürlicher Logarithmus
- RND (X) – Zufallszahl
- SGN (X) – Signumfunktion

SIN (X) – Sinus (X in Bogenmaß)  
 SQR (X) – Quadratwurzel  
 TAN (X) – Tangens (X in Bogenmaß)  
 PI – Konstante  $\pi$

zur Verfügung. Diese Funktionen können beliebig auf numerische Größen (Zahlen, Variablen oder Ausdrücke) angewendet und miteinander verknüpft werden.

Dabei entsteht jeweils ein »numerischer Ausdruck«, zum Beispiel

$$5 * 7 + \text{SIN}(\text{PI} * 45/180)$$

oder

$$-7 * (4 + \text{EXP}(2)).$$

Will man das Ergebnis einer solchen Rechenoperation am Bildschirm anzeigen, so muß dem Ausdruck das Schlüsselwort PRINT (»Drucke«) vorangestellt werden.

Die Eingabe des gesamten Befehls erfolgt zeichenweise über die Tastatur, wobei jede Befehlszeile durch Betätigung der **ENTER**-Taste abgeschlossen wird.

Der Rechner arbeitet die Aufgabe sofort nach **ENTER** ab und zeigt das Ergebnis an.

*Beispiel:* Es wird eingegeben

$$\text{PRINT } 5 * 7 + 8 * 6$$

**ENTER**

Auf dem Bildschirm sind dann Aufgabe und Ergebnis in folgender Weise dargestellt

```
PRINT 5 * 7 + 8 * 6
83
OK
> ■
```

Ab der Position des Cursors (■) kann nun die Eingabe einer weiteren Aufgabe beginnen.

Zur Verkürzung der Eingabe kann das Schlüsselwort PRINT auch durch ein Fragezeichen (»?«) ersetzt werden. Eine Folge von Aufgaben und Ergebnissen ergibt am Bildschirm dann

```
?SIN(PI * 45/180)
.707107
OK
?-5 * 7
-35
OK
?0.5 * LN(10)/(5-3 * 5)
-.115129
OK
> ■
```

wobei die Zeilen 1, 4 und 7 jeweils über die Tastatur eingegeben und mit **ENTER** abgeschlossen werden.

### Variablen und Bezeichnungen

Bevor wir zur Erläuterung einiger einfacher BASIC-Befehle übergehen können, muß noch vorangestellt werden, daß im BASIC des Z 9001 die Verwendung von zwei Variablen-Typen möglich ist.

Zunächst gibt es die numerischen Variablen, die jeweils einen konkreten Zahlenwert (4-Byte-Gleitkommazahl) beinhalten, der der Variablen durch eine Eingabe- oder eine Ergibtanweisung zugewiesen wird. Außerdem können solche Variablen auch mit den im voranstehenden Abschnitt erläuterten Rechenoperationen zu numerischen Ausdrücken verknüpft werden. Im Gegensatz zur üblichen Schreibweise in der Mathematik, in der Variablen jeweils durch einen Buchstaben (z. B.:  $x, y, a, b, i, j, \dots$ ) bezeichnet werden, kann man im BASIC des Z 9001 bis zu zwei Zeichen (Großbuchstabe und Großbuchstabe bzw. Ziffer) zur Bezeichnung nutzen. Dadurch ist es möglich, aus-

reichend viele Variablen zu unterscheiden.

*Beispiel:* A, A1, AS, XX, R7, K9

Neben den numerischen Variablen gibt es beim Z 9001 noch den Typ der Zeichenkettenvariablen (auch String-Variablen), die beliebigen Text bzw. Zeichenfolgen (Groß- und Kleinbuchstaben, Sonderzeichen, Grafikzeichen) bis zu 255 Zeichen beinhalten können. Die Variablenbezeichnung erfolgt wie für numerische Variablen, nur wird noch das Zusatzzeichen »\$« (Dollarzeichen) angefügt.

*Beispiele:* B\$, A1\$, Z\$, T5\$, KKS

Eine Wertzuweisung für alle Variablen erfolgt wie üblich durch das Gleichheitszeichen, wobei auf der rechten Seite auch Ausdrücke stehen dürfen.

*Beispiele:* A=5.7

C1=-12.6\*4

XX=3\*C1

A\$=»WORT1«

ZK\$=»Zeichenkette«

Außerdem können mittels der DIM-Anweisung Variablen-Felder (Matrizen) für numerische und Zeichenkettenvariable vereinbart werden:

DIM X (3, 4) definiert die Matrix

X (0, 0) . . . . . X (0, 4)

⋮ ⋮

X (3, 0) . . . . . X (3, 4)

und stellt Speicherplatz für sie bereit.

Die Elemente X(I, J), I=0, . . . , 3, J=0, . . . , 4 können dann im Programm wie einfache numerische Variable benutzt werden. Ebenso kann durch

DIM T\$(2, 4)

eine Matrix vereinbart werden, deren Elemente Zeichenketten sind.

## Einfache Programme

Nach den Vorbemerkungen wollen wir jetzt einige einfache Programme in BASIC formulieren und ihre Abarbeitung am Z 9001 beschreiben.

Dazu muß man wissen, daß jede Programmzeile mit einer Zeilennummer beginnt und eine oder mehrere Anweisungen (durch Doppelpunkt getrennt) enthält. Bei der Neuerstellung eines Programms wird jede Programmzeile über die Tastatur eingegeben und mit

**ENTER** abgeschlossen. Danach wird sie vom Rechner unmittelbar in den Programmspeicher übernommen. Nach beendeter Eingabe können die Programme durch das Kommando »RUN« gestartet werden. Hierzu steht die Taste **RUN** zur Verfügung, oder es wird

RUN **ENTER**

eingetippt. Die Abarbeitung der Programme erfolgt, wenn nicht anders angewiesen, in der Reihenfolge der Nummerierung.

Einfache Programme bestehen nur aus Eingabe-, Berechnungs- und Ausgabe-Anweisungen, die durch die Schlüsselworte

INPUT – Eingabeanweisung  
LET – Berechnungs- oder Ergibtanweisung  
(dieses Schlüsselwort kann auch entfallen)  
PRINT – Ausgabeanweisung

charakterisiert werden.

*Beispiel:* Soll der Mittelwert von drei Zahlen berechnet werden, so genügt folgendes Programm:

```
10 INPUT »3 Zahlen eingeben :« ;  
A,B,C  
20 LET M=(A+B+C)/3  
30 PRINT »Mittelwert :« ; M
```

Die Abarbeitung dieses einfachen Beispiels erfolgt in folgender Weise:

Das Programm wird durch **RUN** gestartet. Durch die Anweisung 10 wird dann der Text

**3 Zahlen eingeben: ■**

am Bildschirm angezeigt.

Der Rechner wartet jetzt auf die Eingabe der 3 Zahlen A, B, C, die durch Komma getrennt werden müssen, z. B.:

5, 7, 12 **ENTER**

Danach wird gemäß Zeile 20 der Mittelwert M berechnet und mit Hilfe der Anweisung 30 am Bildschirm ausgegeben. Insgesamt entsteht folgendes Bild:

3 Zahlen eingeben: 5, 7, 12  
Mittelwert : 8  
OK  
> ■

Durch »OK« wird das Ende der Programmabarbeitung angezeigt.

Diese kleine Rechnung läßt sich sicher auch auf einem Taschenrechner sehr schnell ausführen. Wollen wir aber zum arithmetischen Mittelwert gleichzeitig noch das geometrische und quadratische Mittel bestimmen, so ist das im BASIC wesentlich einfacher möglich.

Zur Realisierung dieser Programmweiterung wird zusätzlich zum bereits gespeicherten Programm folgendes eingegeben:

a) Zur Berechnung der gesuchten Werte:

21 MG =  $(A \times B \times C) \wedge (1/3)$

**ENTER**

22 MQ =  $\text{SQR}((A \times A + B \times B + C \times C)/3)$  **ENTER**

b) Zur Ausgabe der Werte:

40 PRINT »Geometr. Mittel: « ; MG

**ENTER**

50 PRINT »Quadrat.Mittel: « ; MQ

**ENTER**

Die Ausführung des jetzt im Rechner gespeicherten Programms nach **RUN** liefert am Bildschirm

3 Zahlen eingeben: 5, 7, 12  
Mittelwert: 8  
Geometr.Mittel: 7.48887  
Quadrat.Mittel: 8.52448  
OK  
> ■

Selbstverständlich kann dieses Programm beliebig oft abgearbeitet werden, wobei nach **RUN** stets nur die 3 Zahlenwerte einzugeben sind.

## BASIC-Kommandos

Zur Eingabe, Abarbeitung und Änderung der BASIC-Programme stehen Kommandos zur Verfügung, die ebenfalls mit **ENTER** abgeschlossen und anschließend vom Rechner sofort ausgeführt werden. Für einige Kommandos stehen Tasten zur Verfügung, die eine sofortige Ausführung dieser Kommandos bewirken (z.B. für **RUN**, **STOP**, **LIST**, **CONT**, **PAUSE** – vgl. Tastatur). Die Handhabung der wichtigsten Kommandos wollen wir kurz beschreiben.

## LIST

Mit dem LIST-Kommando kann ein Programm ausgegeben (»gelistet«) wer-

den. Dies erfolgt beim Heimcomputer standardmäßig auf dem Bildschirm. Wird das Programm zur Mittelwertberechnung wie S.56 eingegeben und ergänzt, so wird es nach

LIST **ENTER**

am Bildschirm wie folgt angezeigt:

```

10 INPUT »3 Zahlen eingegeben:«,
   A, B, C
20 LET M=(A+B+C)/3
21 MG=(A*B*C) (1/3)
22 MQ=SQR
   ((A*A+B*B+C*C)/3)
30 PRINT »Mittelwert :« ;M
40 PRINT »Geometr. Mittel:« ;MG
50 PRINT »Quadrat. Mittel :« ;MQ
OK
> ■

```

Zu Auslösung des Kommandos kann ebenso die **LIST**-Taste gedrückt werden.

## EDIT

Durch das EDIT-Kommando wird die Änderung von Programmen einfach möglich. Nach

EDIT 20 **ENTER**

wird vom beschriebenen Mittelwertprogramm die Zeile mit der Nummer 20 angezeigt, also

```
20 LET M=(A+B+C)/3 ■
```

Nun kann diese Zeile mit Hilfe der Kursortasten **→**, **←** sowie der Taste zum Steichen und Einfügen von Zeichen (Taste **DEL** **INS**) korrigiert werden, z.B. kann also das überflüssige

Schlüsselwort **LET** herausgestrichen werden. Nach **ENTER** wird die korrigierte Zeile wieder in das Programm übernommen und anschließend Zeile 21 zur Änderung bereitgestellt.

Der EDIT-Modus kann durch **STOP** verlassen werden.

## CSAVE/CLOAD

Da man vor allem längere Programme nicht mehrmals in den Rechner eintippen möchte, sollte man diese auf einer Magnetbandkassette speichern. Mit dem Kommando

CSAVE »PROGR1« **ENTER**

erfolgt die Speicherung des gesamten BASIC-Programmes auf der Kassette unter dem Namen »PROGR1«. Als Name kann hierbei jede Zeichenkette mit maximal 8 Zeichen benutzt werden.

Zu einem späteren Zeitpunkt läßt sich das gespeicherte Programm nach Positionierung des Magnetbandes vor den Programmanfang durch

CLOAD »PROGR1« **ENTER**

wieder in den Rechner laden und dann abarbeiten.

Der Vollständigkeit halber sei abschließend noch der komplette Kommandosatz des Z 9001 angeführt:

- |       |  |
|-------|--|
| AUTO  | - Automatische Zeilennumerierung bei Eingabe |
| BYE   | - Verlassen des BASIC-Interpreters           |
| CLEAR | - Löschen aller Variablen                    |
| CONT  | - Fortsetzung unterbrochener Programme       |
| CSAVE | - Abspeichern von Programmen auf Magnetband  |



- CLOAD** – Laden von Programmen vom Magnetband
- DELETE** – Löschen von Programmzeilen
- EDIT** – Programmänderungsmodus
- LINES** – Anzahl der bei LIST ausgegebenen Programmzeilen
- LIST** – Programmausgabe (abschnittsweise, Fortsetzung jeweils nach **ENTER**)
- LIST#n** – Programmausgabe auf Drucker oder Magnetband
- LOAD** – Laden von LIST-gespeicherten Programmen vom Magnetband
- NEW** – Löschen des gesamten BASIC-Programmes
- NULL** – Legt Anzahl auszugebender Nullen bei LIST# fest
- RENUMBER** – Automatische Nummerierung des BASIC-Programmes
- RUN** – Programmstart
- TROFF** – Ausschalten TRACE-Modus
- TRON** – Einschalten TRACE-Modus
- WIDTH** – Länge der Ausgabezeilen festlegen

### REM – Anweisung

Zur besseren Übersichtlichkeit der Programme können nach dem Schlüsselwort »REM« beliebige Kommentare in die Programme eingefügt werden, die keinen Einfluß auf deren Abarbeitung haben. »REM« kann auch durch »!  
« ersetzt werden.

### FOR ... NEXT – Anweisung

Zur mehrfachen Abarbeitung von Anweisungsgruppen (z.B. bei der Berechnung von Tabellen, Folgen, Reihen oder bei Iterationszyklen) ist eine sogenannte »Laufanweisung« erforderlich, die im BASIC durch die FOR ... NEXT-Anweisung gebildet wird. Sollen z.B. die Funktionen

$$f(x) = x \cdot e^x, \quad g(x) = \lfloor 3x + 10 \rfloor$$

im Intervall  $[-2,2]$  im Abstand von 0,5 tabelliert werden, so wird dies durch folgendes Programm erreicht:

```
40 REM Tabellenberechnung
50 FOR X=-2 TO 2 STEP 5
60 F=X*EXP(X)
70 G=SQR(3*X+10)
80 PRINT X, F, G
90 NEXT X
```

Dabei bedeutet die Anweisung 50, daß die folgenden Anweisungen mehrfach ausgeführt werden sollen, und zwar für  $X=-2$  bis  $X=2$  in Abständen von 0,5, d.h. für

$$X = -2, -1.5, -1, -0.5, 0, 0.5, 1, 1.5, 2.$$

Die Anweisung 90 beendet diese Anweisungsgruppe.

Zur Zeile 80 muß noch erläutert werden, daß die durch PRINT auszugebenden Elemente durch Komma oder Semikolon getrennt werden können. Das Komma bewirkt die Ausgabe in einem festen Raster (Tabelle), während

### Weitere BASIC-Anweisungen

Neben den bereits genannten fundamentalen Anweisungen INPUT, LET und PRINT stehen eine Reihe weiterer Anweisungen zur Verfügung, die zur rationellen Gestaltung vieler Programme nützlich sind. Einige sollen kurz erläutert werden:

durch ein Semikolon die Werte fortlaufend ausgegeben werden.

Die Ausführung dieses Programmes liefert am Bildschirm:

-2	-.270671	2
-1.5	-.334695	2.34521
-1	-.367879	2.64575
-.5	-.303265	2.91548
0	0	3.16228
.5	.824361	3.39117
1	2.71828	3.60555
1.5	6.72253	3.80789
2	14.7781	4
OK		
> ■		

### IF ... THEN - Anweisung

Häufig ist es erforderlich, einige Anweisungen nur unter gewissen Bedingungen auszuführen (z. B. Berechnung von  $SQR(X)$  nur für  $X \geq 0$ ). Dazu dient die Anweisung

IF bedingung THEN anweisung.

Hierbei steht für »bedingung« ein logischer Ausdruck (z. B.  $A < B$ ,  $X > 5$ ,  $AS = »A«$ ), der wahr oder falsch sein kann.

Nur wenn er wahr ist, wird die auf THEN folgende »anweisung« ausgeführt. Wesentlich für die Nutzung ist, daß die »bedingung« auch Vergleiche von Zeichenketten beinhalten darf.

*Beispiel:* 90 INPUT »Gib Wert für A ein: «; A  
100 IF A > 0 THEN PRINT »A ist positiv «  
110 IF A < 0 THEN PRINT »A ist negativ «

### GOTO - Anweisung

Durch die Anweisung

GOTO zeilennummer

kann ein Programm, abweichend von der normalen Reihenfolge, in einer anderen Programmzeile entsprechend der »zeilennummer« fortgesetzt werden. Diese »zeilennummer« muß als Zahlenwert konkret angegeben werden, beispielsweise

210 GOTO 40.

Ein Beispiel für die Anwendung der beschriebenen Befehle ist das folgende Ratespiel, in dem im Befehl 30 durch einen Spieler (A) eine Zahl eingegeben werden muß, die ein anderer (B) dann erraten soll.

```
10 REM Zahlenratespiel
20 REM Spieler A gibt vor (Zeile 30)
30 INPUT »Vorgabezahl(Spieler A):«; A
35 REM Loeschen des Bildschirmes in 36
36 CLS
40 REM Spieler B darf raten (Zeile 50)
50 INPUT »Gib einen Tip ein: «; B
60 IF A < B THEN PRINT B, »ist zu groß«
70 IF A > B THEN PRINT B, »ist zu klein«
80 IF A = B THEN 100
90 GOTO 40
100 PRINT B, »ist die gesuchte Zahl«
110 INPUT »Neues Spiel (J/N)? «; AS
120 IF AS = »J« THEN GOTO 30
130 PRINT »ENDE«
```

Unter Nutzung der beschriebenen Standardfunktionen kann sehr leicht auch der Rechner als »Spieler A« eingesetzt werden. Ersetzt man Zeile 30 durch

30 A = INT(100 \* RND(1)),

so wird automatisch eine ganzzahlige zweistellige Zufallszahl erzeugt.

Der Ablauf des Programms am Bildschirm vollzieht sich in folgender Weise:

```
Gib einen Tip ein: 50
50 ist zu groß
Gib einen Tip ein: 10
10 ist zu klein
Gib einen Tip ein: 25
25 ist zu klein
Gib einen Tip ein: ■
```

Neben den mathematischen Standardfunktionen stehen im BASIC des Z 9001 einige Funktionen zur Zeichenkettenverarbeitung zur Verfügung. Von diesen sollen hier CHR\$, LEN und STRING\$ erläutert werden.

CHR\$(N) – Liefert das Zeichen mit der Nummer N in der ASCII-Code-Tabelle.

LEN(A\$) – Liefert die Anzahl der Zeichen, die die Zeichenkette A\$ enthält.

STRING\$(N, A\$) – Setzt die Zeichenkette A\$ genau N-mal aneinander.

Zusätzlich sei erwähnt, daß man Zeichenketten durch »+« aneinanderfügen kann und in einer IF...THEN-Anweisung wie bei Zahlen die Vergleichsoperatoren <, >, =, <=, >= anwendbar sind. Damit kann man mit Zeichenketten praktisch »rechnen«, wobei die Vergleichsoperatoren die Zeichenketten bezüglich ihrer alphabetischen Ordnung (allgemeiner: entsprechend der ASCII-Code-Numerierung) sortieren, wie das folgende Programmbeispiel zeigt:

```
5 ! Sortierung zweier Namen
10 INPUT »Gib 2 Namen ein: « ;
N1$, N2$
20 IF N1$ < N2$ THEN GOTO 60
```

```
30 H$ = N2$
40 N2$ = N1$
50 N1$ = H$
60 PRINT »Alphabetische Reihenfolge: «
70 PRINT N1$; CHR$(44); N2$
80 GOTO 10
```

Werden nach **RUN** in Anweisung 10 die Namen KARL, ERICH eingegeben so ist am Bildschirm folgendes zu sehen:

```
Gib 2 Namen ein: KARL, ERICH
Alphabetische Reihenfolge:
ERICH, KARL
Gib 2 Namen ein: ■
```

Zur Erläuterung der Zeile 70 sei vermerkt, daß der Rechner alle Buchstaben, Ziffern, Sonder- und Grafikzeichen in codierter Form speichert. Jedem dieser Zeichen entspricht eine Zahl (0 bis 255). Durch die Funktion CHR\$(N) kann umgekehrt einer Zahl auch das zugehörige Zeichen zugeordnet werden.

In Zeile 70 liefert der Wert CHR\$(44) das Kommazeichen, welches zwischen ERICH und KARL ausgegeben wird. Die Anwendungsmöglichkeiten der beschriebenen Zeichenkettenfunktionen sind sehr vielfältig. Neben der Analyse von Zeichenketten können besonders Probleme der Bildgestaltung zweckmäßig gelöst werden.

Soll eine Überschrift vorher unbekannter Länge unterstrichen werden, so läßt sich dies mit den obigen Funktionen leicht realisieren. Im folgenden Programmteil soll die Textzeile

```
ABRECHNUNG FUER monatsname
für beliebigen »monatsnamen« entsprechend der Textlänge mit dem Zeichen Nr.160 (■) unterstrichen werden.
10 ! Anwendung der ZK-Funktionen
20 A$ = »ABRECHNUNG FUER «
```

30 INPUT »Gib Monatsnamen ein: « ;  
 MOS  
 35 CLS: REM Bildschirm loeschen  
 40 N = LEN(MOS) + 16  
 50 PRINT AS + MOS  
 60 PRINT STRING\$(N, CHR\$(160))  
 70 ! Datenauswertung fuer lfd. Monat  
 ⋮

### Zusammenstellung der BASIC-Anweisungen

Da eine vollständige Beschreibung aller Anweisungen des Z 9001 den Rahmen dieses Abschnittes sprengen würde, soll wenigstens eine Übersicht der Schlüsselwörter aller Funktionen und Anweisungen aufgeführt werden:

- a) Logische Operatoren  
 AND, NOT, OR
- b) Mathematische Funktionen  
 vgl. S. 54
- c) Zeichenketten-Funktionen
  - ASC – ASCII-Code eines Zeichens
  - CHR\$ – Zeichen entsprechend ASCII-Nummer
  - INSTR – Position einer ZK in einer anderen
  - LEN – Länge einer Zeichenkette
  - LEFT\$ – Linker Teil einer Zeichenkette
  - MID\$ – Mittlerer Teil einer Zeichenkette
  - RIGHT\$ – Rechter Teil einer Zeichenkette
  - STR\$ – Wandelt Zahl in Zeichenkette um
  - STRING\$ – Vervielfacht eine Zeichenkette
  - VAL – Wandelt Zeichenkette in Zahl um
- d) Sonderfunktionen
  - FRE – Gibt noch freien Speicherplatz an

POS – Liefert Kursorposition  
 SPC – Ausgabe von Leerzeichen  
 USR – Aufruf nutzereigener Maschinenprogramme  
 TAB – Setzt Kursor auf TAB-Position

### e) Anweisungen

BEEP – Tonerzeugung  
 BORDER – Farbuweisung für Randbereich  
 CALL – Aufruf eines Maschinenprogramms  
 CLS – Bildschirm löschen  
 DATA – Datenbereitstellung  
 DEEK – 2-Byte-Eingabe  
 DEF FN – Definition einzelner Funktionen  
 DIM – Feldvereinbarung  
 DOKE – 2-Byte-Ausgabe  
 END – Phys. Programmende  
 FOR...NEXT – Laufanweisung  
 GOSUB – Sprung zu einem Unterprogramm  
 GOTO – Sprung zu einer Zeile  
 IF...THEN...ELSE – Bedingte Anweisungen  
 INK – Festlegung Vordergrundfarbe  
 INKEY\$ – Zeicheneingabe von Tastatur  
 INPUT – Eingabe Zahlen oder Zeichenkette  
 INP – Lesen einer Port-Adresse  
 JOYST – Spielhebelabfrage  
 LET – Ergibtanweisung  
 ON...GOTO – Verzweigungsanweisung  
 ON...GOSUB – Verzweigung zu Unterprogrammen  
 OUT – Ausgabe auf Port-Adresse

PAPER	- Festlegung Hintergrundfarbe
PAUSE	- Programmunterbrechung
PEEK	- Lesen eines Bytes von einer Speicheradresse
POKE	- Schreiben eines Bytes in eine Speicheradresse
PRINT	- Standardausgabe von Zahlen und Zeichen
PRINT AT	- Ausgabe ab spezieller Bildschirmposition
READ	- Lesen der DATA-Anweisungen
REM	- Bemerkung (Kurzzeichen: !)
RESTORE	- Setzen des Zeigers für READ-Anweisung
RETURN	- Abschluß eines Unterprogrammes
STOP	- Programmabbruch
WAIT	- Zeitweilige Programmunterbrechung
WINDOW	- Festlegung des Rollbereiches am Bildschirm

### 3. Erweiterungsmöglichkeiten des Heimcomputers Z 9001

Wie bereits angedeutet wurde, besitzt der Heimcomputer Z 9001 eine Reihe von Möglichkeiten, zusätzliche Peripherie anzuschließen.

Das wichtigste externe Gerät außer Fernsehgerät und Kassettentonband ist sicher ein Drucker, der beim Z 9001 über einen V24-Druckermodul angeschlossen werden kann. In der DDR steht z.B. der Thermodrucker TD 40 (robotron K 6303) zur Verfügung.

Da der Drucker über den Ausgabekanal 2 angesprochen wird, ist es möglich, mit

```
LIST#2 ENTER
```

Programmausdrucke auf den Drucker auszugeben oder innerhalb der Programme mit PRINT#2;... Daten zu drucken.

*Beispiel:* Ersetzt man im Beispiel zur FOR...NEXT-Anweisung (Abschnitt 2) Zeile 80 durch

```
80 PRINT#2; X, F, G
```

so wird die gesamte Tabelle nicht auf dem Bildschirm, sondern auf dem Drucker ausgegeben.

Typisch für Heimcomputer ist die Möglichkeit des Anschlusses von 2 Spielhebeln. Die Stellung dieser Spielhebel (9 Positionen) kann durch die Funktion JOYST(X) abgefragt und in einem Spielprogramm z. B. zur Erzeugung von Bewegungen (Autorennen, Katze und Maus u.ä.) genutzt werden.

*Beispiel:* 100 ! Spielhebel 1 abfragen  
110 A=JOYST(1)  
120 IF A=4 THEN...!  
Reaktion

Falls der Farbmodul im Heimcomputer eingebaut ist, können alle Zeichen farbig dargestellt werden.

Dafür stehen 8 Farben zur Verfügung, die unabhängig voneinander für die Farbe der Schrift (Befehl INK) bzw. des Schriftgrundes (Befehl PAPER) ausgewählt werden können. Den Farben sind die Zahlen 1, ..., 8 in der Reihenfolge schwarz, rot, grün, gelb, blau, purpur, hellblau, weiß zugeordnet, so daß die Anweisungen

```
200 N=2
```

```
210 INK N
```

```
220 PAPER 1
```

```
230 PRINT »rot auf schwarz«
```

eine rote Schrift auf schwarzem Hintergrund hervorrufen.

Weiterhin steht ein Musik-Modul zur Verfügung, mit dem über den Befehl

BEEP N, M, L

auch Tonfolgen programmiert werden können.

Dabei codieren N – die Tonhöhe

M – die Tonlänge

L – die Lautstärke.

Interessante Anschlußmöglichkeiten bietet der Heimcomputer Z 9001 durch die 8 parallelen E/A-Kanäle, über die verschiedene elektronische Geräte und Hobby-Anlagen anschließbar sind.

Weiterhin können z.B. mit dem zum Computer erhältlichen Analog-Digital-Wandler Meßwerte (Spannungen, Temperaturen, Kräfte, Zeiten usw.) in den

Rechner übernommen und dort verarbeitet werden. Ebenso kann der Rechner Signale ausgeben, mit denen andere Geräte und Anlagen (elektrische Eisenbahnen usw.) gesteuert werden können.

Neben den als Zubehör angebotenen Ergänzungsteilen bleibt es dem Bastler überlassen, weitere Geräte unter Beachtung der Anschlußbedingungen für den Heimcomputer ansprechbar zu machen.

Autor:

*Dr. rer. nat. Gert Keller*

DDR 8023 Dresden  
Eisenberger Straße 9

VEB Robotron Meßelektronik  
„Otto Schön“ Dresden

---

© VEB Fachbuchverlag Leipzig 1984

1. Auflage

Lizenznummer 114-210/50/84

LSV 1083

Verlagslektor: Helga Fago

Gestaltung: Lothar Gabler

Printed in GDR

Satz und Druck:

Messedruck Leipzig, Bereich Borsdorf

III-18-328

Redaktionsschluß: 15. 3. 1984

Bestellnummer: 546 921 6

00780

---