



PHILIPS

COMPACT .

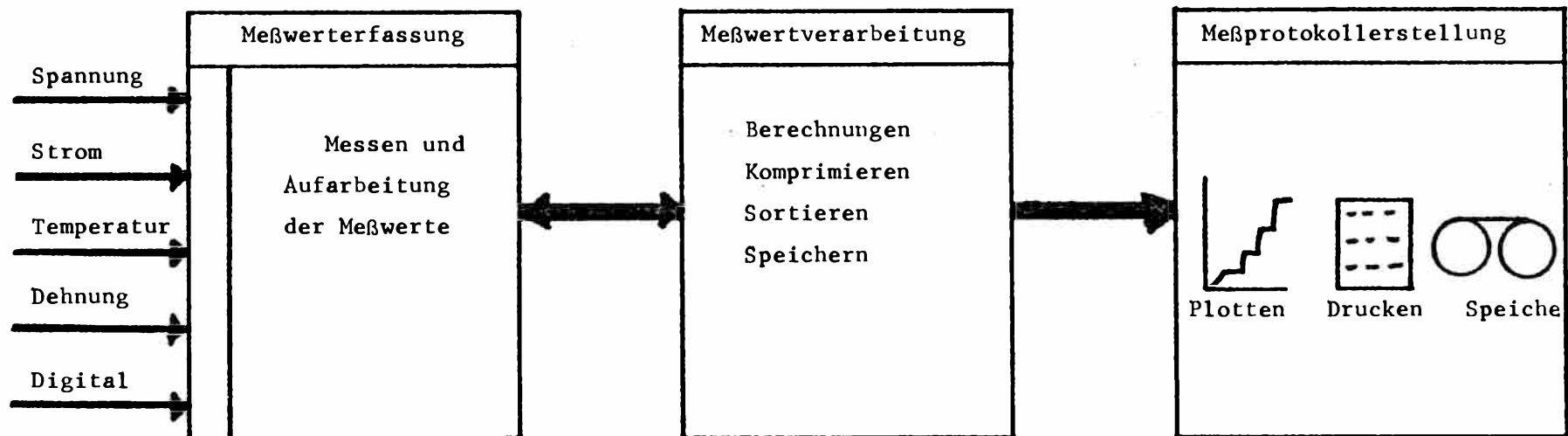
COMPUTER .

SYSTEM

P M 4400

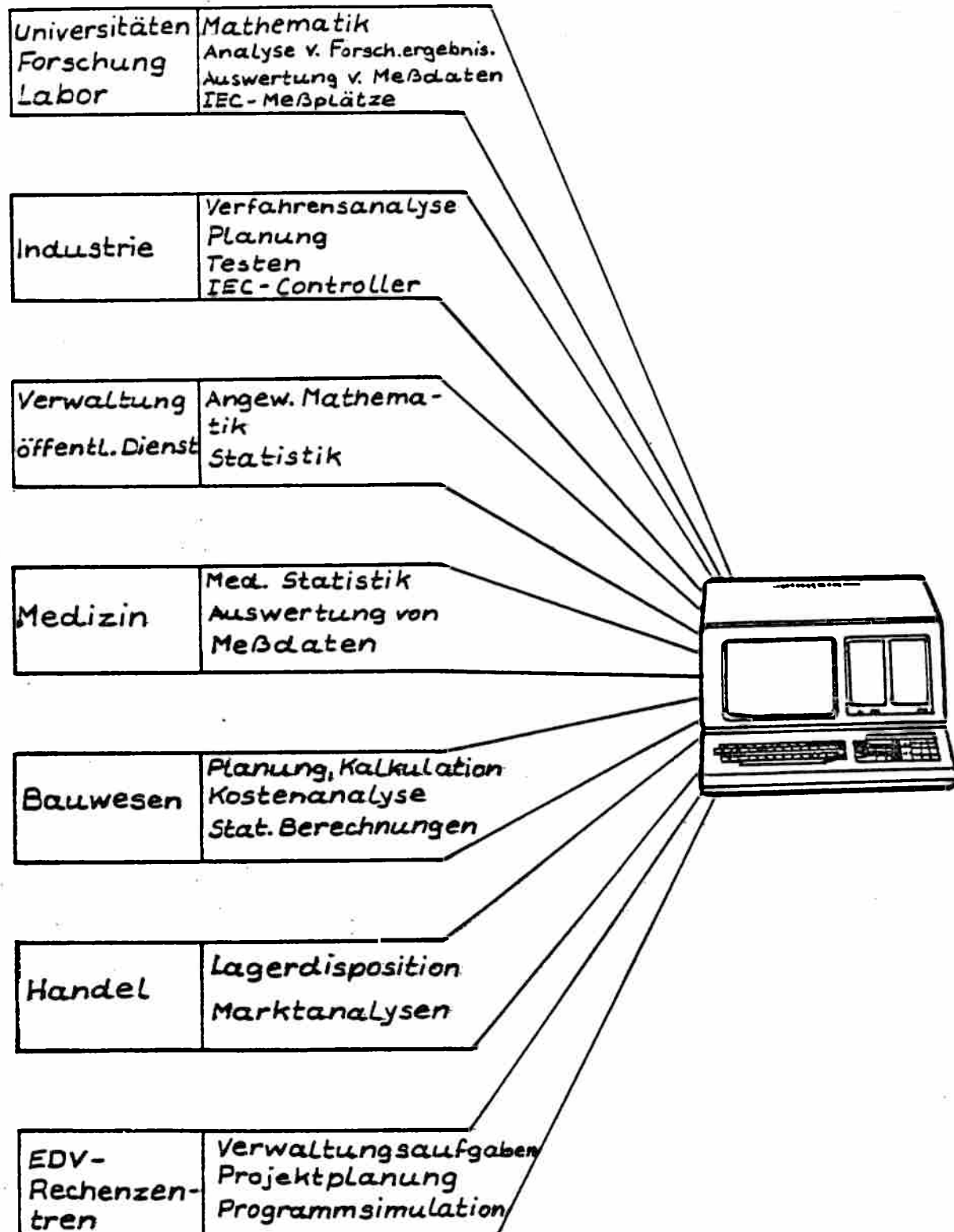
Anforderungen an einen modernen Meßplatz

- Variabel
- Aufwärtskompatibel
- "Intelligent"
- Zuverlässig
- Genau
- Vielseitig verwendbar



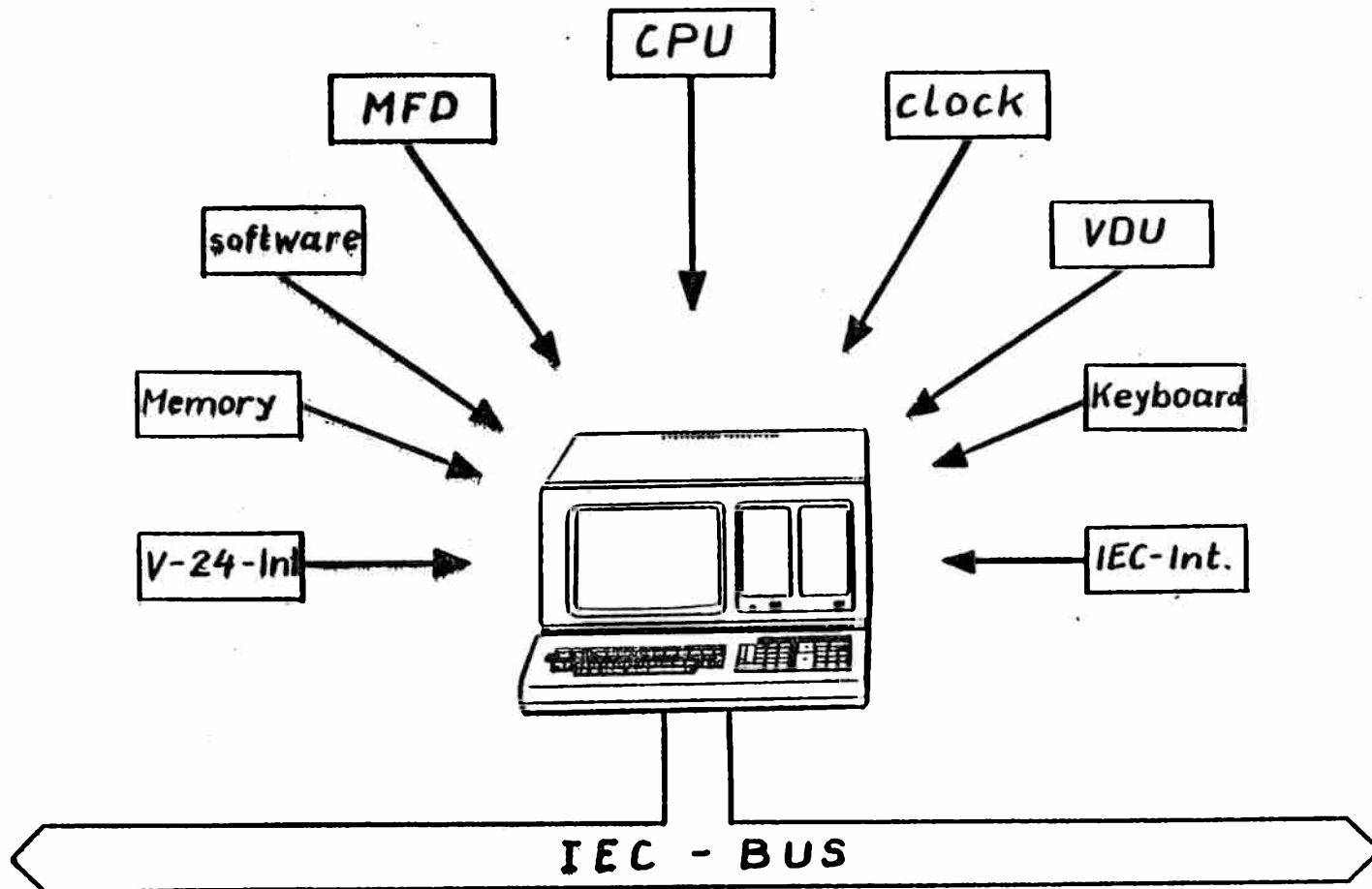
PHILIPS

Einsatzbereiche des PM 4410



Hardware - Konfiguration

Compact - Computer - System PM 4400





Hardwarekomponenten

Central Prozessing Unit

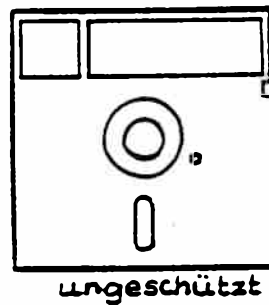
C P U

- Type: P 851
- 16 bit Minicomputer
- Aufgebaut mittels moderner LSI-Technologie (Large scale integration)
- Aufgebaut auf einer doppelten Europakarte
- Die CPU steuert sämtliche Funktionen
- Kompatibilität zu den größeren Rechnern der Philip-Serie P 800 ist gegeben

Arbeitsspeicher

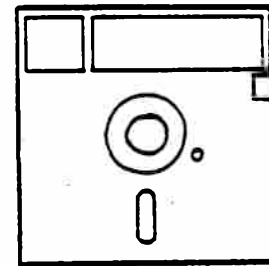
- Halbleiterspeicher mit RAM's (Random access memory) aufgebaut
- Speicherkapazität 32, 48 oder 64 k-Byte
- Zyklus-Zeit 450 ns für 16 bit
- Zugriffszeit ca. 300 ns
- Systemsoftware befindet sich auf Diskette
- Zum Laden der Systemsoftware existiert ein Bootstrap in ROM mit 64 Worten
- Durch RAM-Betrieb flexible Systemsoftware (z.B. Fortran; Assembler)
- Halbleiterspeicher ist auf einer doppelten Europakarte untergebracht

Mini - Floppy - Disk M F D



ungeschützt

ohne
Schreib-
sperre



Schreibgeschützt

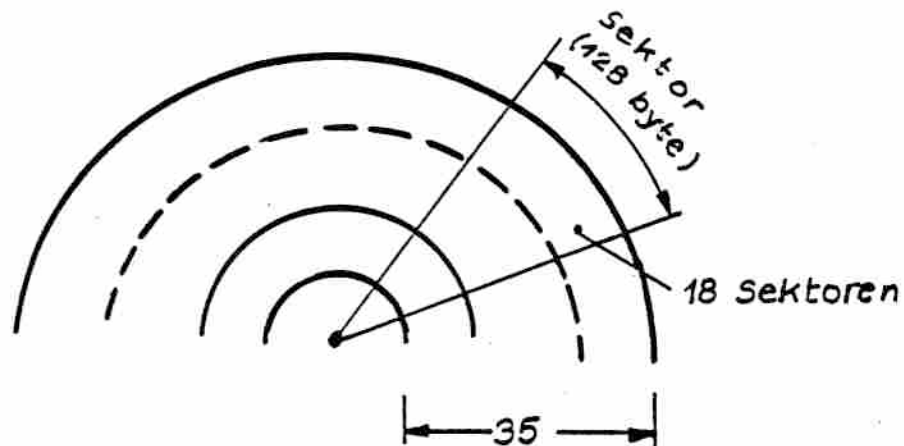
mit
Schreib-
sperre

- MFD - Laufwerk μ p-gesteuert
- Wahlweise 1 oder 2 Laufwerke
- Speicherkapazität 80 KByte formatiert für Speicherung von Betriebssoftware, Anwenderprogrammen und Daten
- Übertragungsrate 125 KBit /sec
- Mittlere Zugriffszeit 463 msec
- Zugriffszeit (Spur/Spur 40ms)
- Fehlerrate beim Lesen 1 auf 10^{11} bit
- Lebensdauer der Disketten ca. $30 \cdot 10^6$ Durchläufe
- Softwareunterstützung für 3 Laufwerke vorhanden

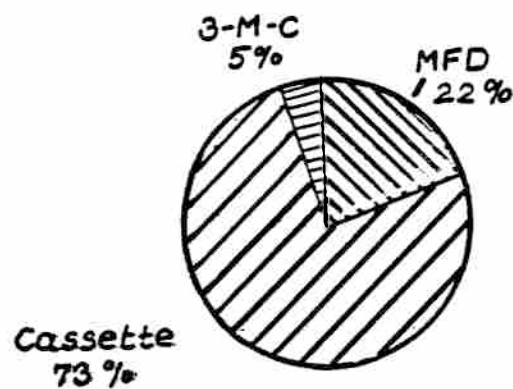
Speicher K:

$128 \times 18 \times 34$

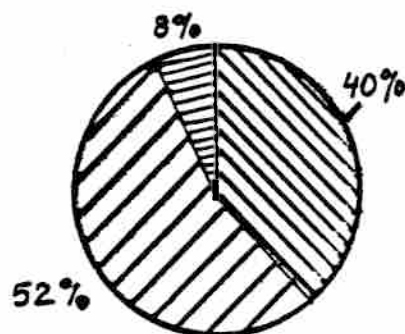
$= 78,3 \text{ Kbyte}$



Marktanteile für MFD



1975



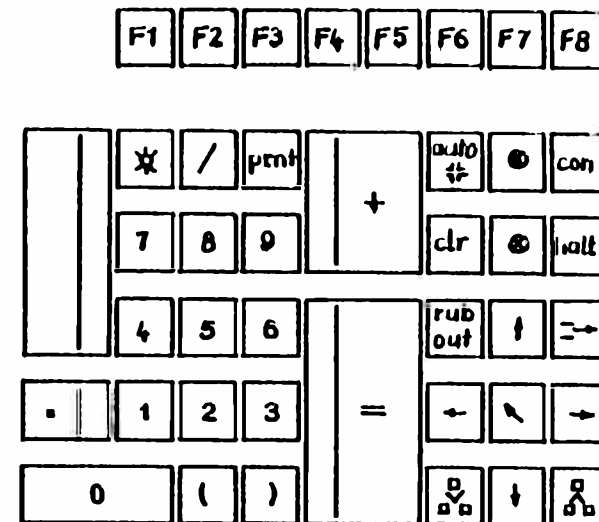
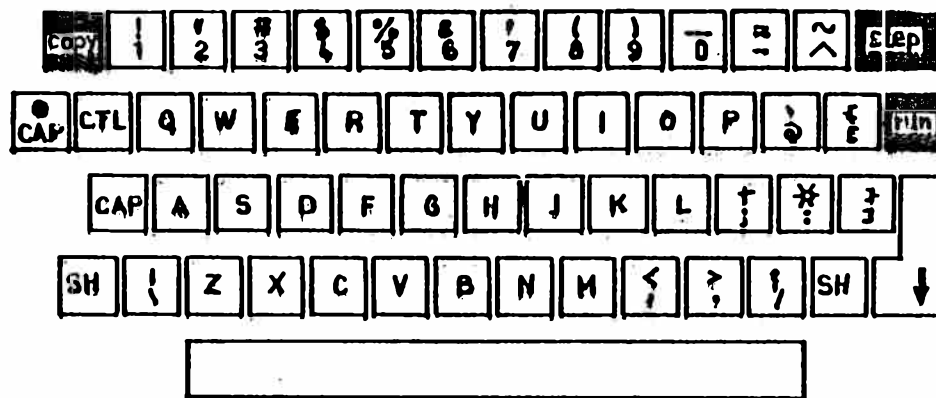
1980

- MFD ist ein preisgünstiges Speichermedium
- Schnelle Zugriffszeit
- Einfaches Floppy-File-Management
- Zukunftssicheres Speichermedium

Video Display

- Fernsehschirm P4 weiß (standard)
- 1920 Zeichen (24 Zeilen / 80 Zeichen)
- Anzeige folgender Informationen
 - Ergebnis bei manueller Arbeitsweise
 - Programminstruktion
 - Systemmitteilungen
 - Daten u. Programmmeldungen
 - Eingabe über Tastatur
- Wahlweise beim Schriftbild schwarzer o. weißer Untergrund
- Darstellungsmöglichkeiten aller 128 ASII - Zeichen (Groß- u. Kleinbuchstaben, Sonderz. etc.)
- Schirmgröße 12" (diagonal)
- Darstellung 7*9 - Matrix
- Anschlußmöglichkeit für 10 externe Monitore über BNC - Buchse
- Automatischer Zeilenvorschub beim Erreichen des unteren Bildrandes
- Blinkender Cursor beliebig verschiebbar
- Gute Ablesbarkeit auch bei Sonnenlicht
- Helligkeit einstellbar, blendfreie Darstellung

Tastentfeld



- BASIC-Tastatur für die Eingabe von Programmen, Verarbeitungsanweisungen, BASIC-Worte für den direkten Aufruf
- Mit 99 Tasten sind sämtliche ASII - Zeichen darstellbar, Sondertasten geben Arbeitserleichterung
- Frei über ein Kabel mit dem Gerät verbunden
- Übertragungsrate bis 800 Zeichen/sec.

Echtzeituhr

- Bei 50 Hz wird alle 20 ms ein Programmunterbrechungssignal erzeugt
- Unterbrechungssignal steht zur Weiterverarbeitung auf einer der 16 Programmunterbrechungsebenen zur Verfügung

IEC - Interface

- Nach DIN IEC 66.22
- Standardmäßig eingebaut
- Funktionell kompatibel mit IEEE 488/75
- Durch BASIC einfach programmierbar
- Serielle Zustandsabfrage (seriell poll)
- Alle Controller-Funktionen sind implementiert

Serielle Schnittstelle

- V 24 oder Stromschleife (20 mA)
- Übertragungsrate getrennt für Sender und Empfänger auf 50-9600 Baud einstellbar
- Zeichenlänge 5-8 bit
- Leitungslänge bis 300 m
- Anschlußmöglichkeit für Schnelldrucker



Software

Die wichtigsten Funktionen der Software sind:

- Kompilierung der Systembefehle
- Kompilierung der Anweisungen
- Interpretation des BASIC-Programms

B A S I C

Beginners
All purpose
Symbolic
Instruction
Code

- Leicht erlernbar
- Leicht in der Handhabung
- problemorientiert



Daten

Numerische Daten:	Integerzahlen
	Dezimalzahlen
	Gleitkommazahlen
String Daten:	Alpha Numerisch (ASCII)
Felder:	Eindimensional
	Mehrdimensional

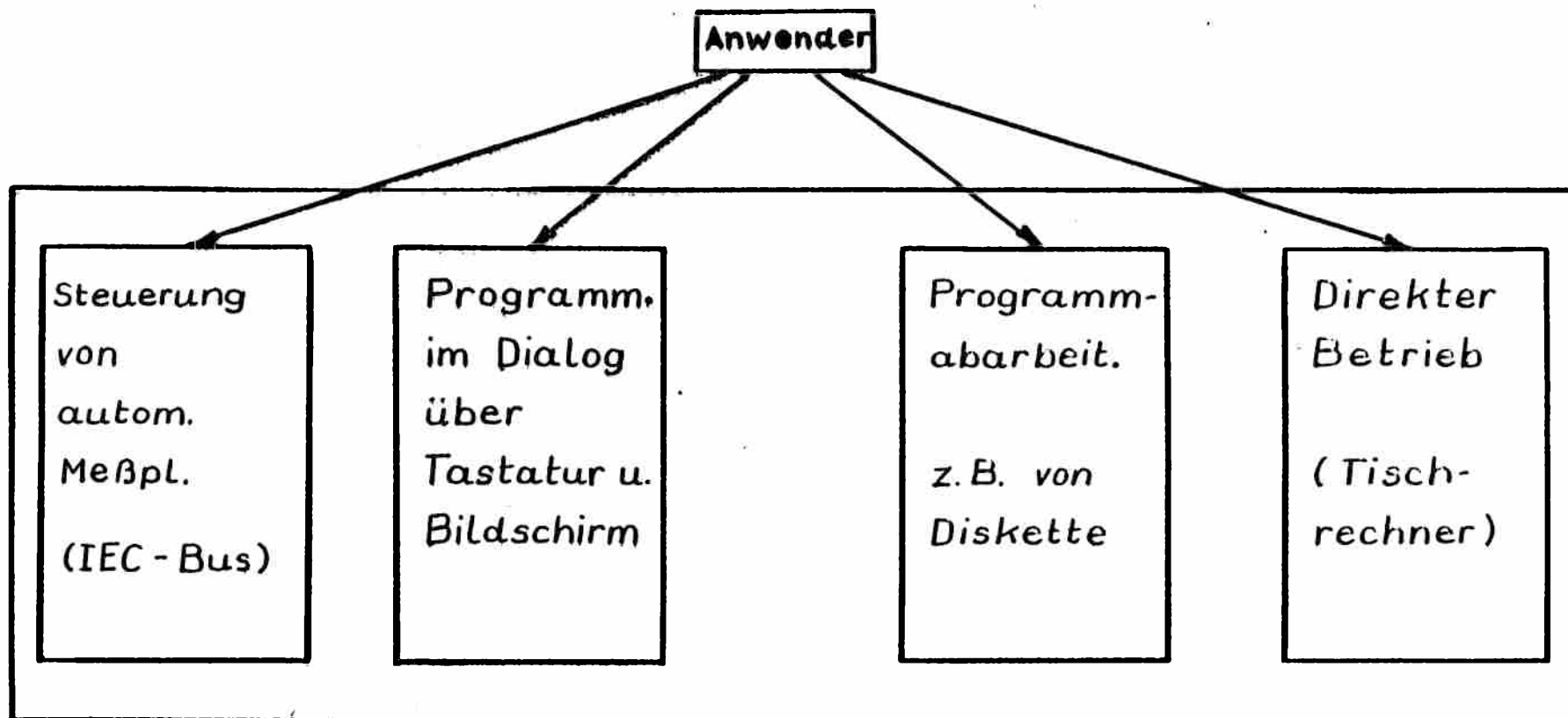
Operatoren

Arithmetisch:	+	Addition
	-	Subtraktion
	/	Division
	*	Multiplikation
	**	Exponent

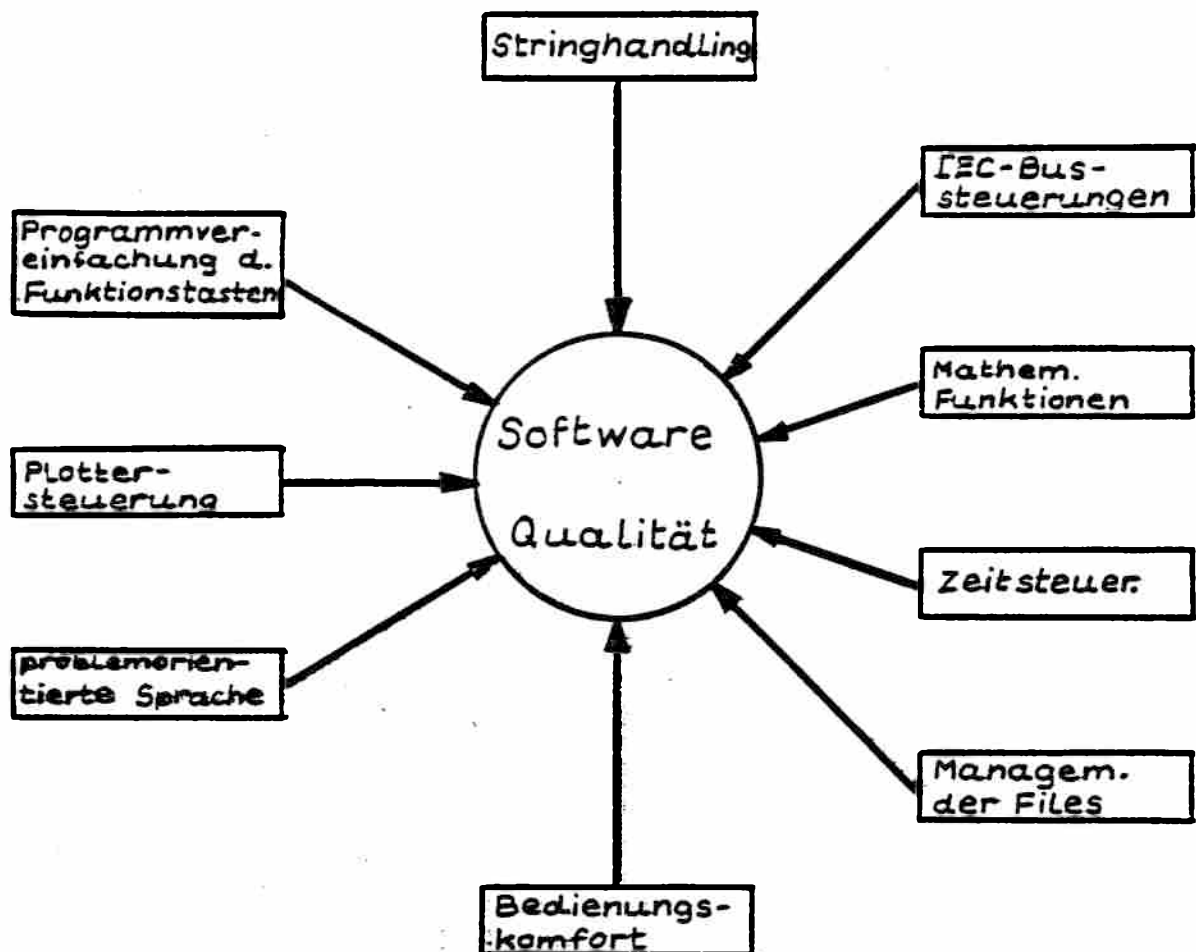
Vergleichsoperatoren:

=	gleich
< >	ungleich
>	größer
<	kleiner
>=	größer oder gleich
<=	kleiner oder gleich

Betriebsarten des Systems

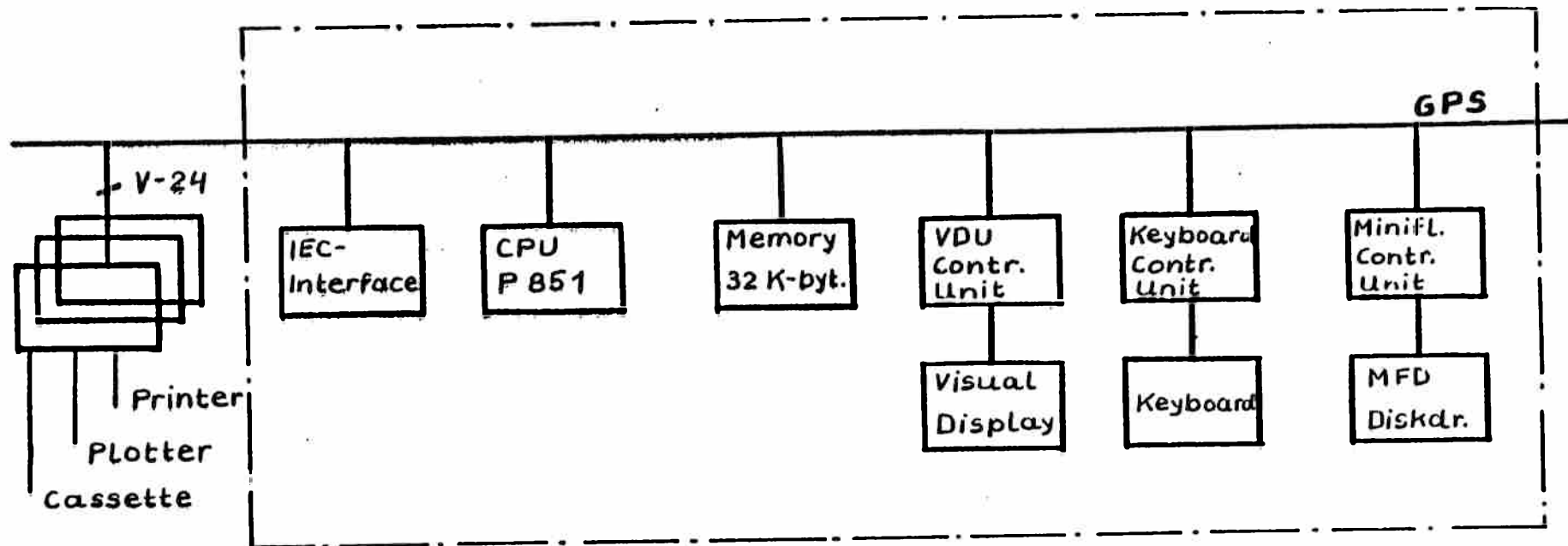


Über die standardmäßigen Befehle und Anweisungen gibt es eine Reihe von Erweiterungen



- Einführung durch Bedienungshandbuch
- Erleichterung durch BASIC-Handbuch
- Lerndisketten

Basis - System



- Nachträgliche Speichererweiterung durch *MODUL-Technik* leicht möglich
- 2. MFD-Laufwerk kann beim Kunden eingebaut werden
- Im Grundgerät enthalten sind Disketten, Programm- u. Operating-Manual



2.4. SUMMARY OF BASIC-COMMANDS

2.4.1. System control commands

CLEAR	to reinitialize the system
SEND	return from the BASIC-system to the monitor
PAUSE	temporary halt of the program
GOON	continue execution of a BASIC-program
RUN	starts execution of a BASIC-program
DELETE	to delete program lines
RESEQUENCE	to change the linenumbers in the program
FORMAT	to initialize a mini floppy disk
CALC	set calculator mode
UNCALC	reset calculator mode
COPY	to copy a mini floppy disk
LET	to assign values to variables in direct mode
TRACE	to list the program execution sequence
SPACE	return remaining spaces

2.4.2. Commands for input/output handling

LOAD	to load a program from the mini floppy disk
SOUTPUT	switches an external device for system output
LIST	to print a program source on an external device
SAVE	saves program on an external device
PRINT	to print values in direct mode
DFILE	declare a file
SFILE	scratch a file
RFILE	rewind a file
CFILE	close a file
AFILE	assign a filecode to a file
DIRECTORY	list the directory of a floppy disk



2.6. LIST OF STATEMENTS OF THE KERNEL COMPONENT

DIM	Declares the dimension of arrays or strings
DEF	To define internal functions
LET	Assigns a value of an arithmetic or string expression to a variable
IF	Conditional branch
FOR	Head of FOR-loop
NEXT	Tail of a FOR-loop
GOTO	Absolute branch
GOSUB	Internal subroutine call
RETURN	Transfers control back from an internal subroutine
ON	Selected conditional branch
PAUSE	Sets program in the wait-state
STOP	Termination of program execution
END	Termination of program execution and closure of source stream
REMARK	Adds comment to source
DATA	Defines an internal sequential "Data file"
READ	Assigns next values from "Data file" to variables
RESTORE	Rewinds internal "Data file"
INPUT	Requests values from external file and assigns them to variables
PRINT	Visualizes values on a device
RANDOMIZE	Initializes the random-generator
FIX	Numerics in fixed point notation
FLOAT	Numerics in floating point notation
PAGE	Erase screen
HOME	Cursor to home-position
SET	Enable/disable user definable keys
INIT	Reset initial value
DELAY	Suspend program execution
WAIT	Wait for SRQ or EOI
POLL	Issue serial poll on IEC-bus
PHYSICAL WRITE	Physical output access on IEC-bus
PHYSICAL READ	Physical input access on IEC-bus
CHANGE END CHAR	Change end-of-transmission character
OFF SRQ	Undo the SRQ Declarative
OFF EOI	Undo the EOI Declarative



Anweisungen: (Standard BASIC)

DIM:	Speicherplatzreservierung
DEF:	Definition beliebiger Funktionen
LET:	Arithmetische Zuweisung
IF:	Vergleichen
FOR:	z. Formulieren einer Programmschleife
NEXT:	" " " "
GOTO:	unbedingte Programmverweigerung
GOSUB:	Sprunganweisung in ein U.P.
RETURN:	Rücksprunganweisung in das H.P.
PAUSE:	Warten im Programm
STOP:	Sprung zum Programmende
END:	physikalische Ende eines Programms
REMARK:	zum Einfügen von Anmerkungen
DATA:	zum Einlesen von Daten (Datenfiles)
READ:	Einleseanweisung für progr. Daten
RESTORE:	Anweisung zum mehrmaligen Lesen eines Datenblocks
INPUT:	Eingabeanweisung
PRINT:	Schreibbefehl
RANDOMIZE:	Zur Erzeugung eines neuen Startpunkts für RND
FIX:	Fixkommazahl
FLOAT:	Ausgabe mit Formatspezifikation
PAGE:	Cursorbewegung
HOME:	"



Standard Funktionen

ABS (X):	Absoluter Betrag
SQR (X):	Quadratwurzel
SGN (X):	Vorzeichen von X: 1 für $x > 0$ 0 * $x = 0$ -1 * $x < 0$
INT (X):	Integr. Zahl X (ganzzahlig)
SIN (X):	Sinus X
COS (X):	Cosinus X
TAN (X):	Tangens X
ATN (X):	Arcustangens
EXP (X):	Exponent (e^x)
LOG (X):	$\ln X$
RND (X):	Zufallszahlen zw. 0 u. 1
TAB (X):	Tabellieren



MATRIX - Anweisungen

MAT COPY:	Kopieren einer Matrix
MAT ADD:	Matrizen - Addition
MAT SUBTRACT:	Matrizen - Subtraktion
MAT MULTIPLY:	Matrizen - Multiplikation
MAT SCALAR:	Scalare - Multiplikation
MAT ZERO:	Elemente einer Matrix zu 0
MAT CONSTANT:	Elemente einer Matrix zu 1
MAT INVERSE:	Matrix - Inversion
MAT INPUT:	Eingabeanweisung für Matrixelemente
MAT READ:	Einleseanweisungen für Matrizen
MAT PRINT:	Ausgabeanweisungen für Matrizenelemente



Plotter - Anweisungen

MOVE:	Stiftbewegung
DRAW:	Schreiben
RELATIVE MOVE:	Stiftbewegung (rel. Position)
RELATIVE DRAW:	" (rel. Position)
SPACE:	Abstand Zeichen / Zeichen
AXIS:	Horizontal- u. Vertikalzeichen
VIEWPORT:	Vergrößerung u. Verschiebung
WINDOW:	Anschnitt (Kurven)
ROTATE:	Schrägstellung (Schriftbild)
ALPHASCALE:	Buchstabengröße festlegen
ALPHAROTATE:	Schrägstellung d. Schriftbildes
PRINT:	Schreiben von Buchstaben



Floppy - File - Management

Schreiben auf MFD

FO # 48 → Formatieren/Floppy wird gelöscht

DF # 48, NAME → File festlegen

SA # 48, NAME → Beschreibung des Files

LO NAME → Laden von MFD ins RAM

Neue Belegung eines bestimmten Files

AF # 48, NAME, 51 128

RF # 48

Anzeige der Filenamen

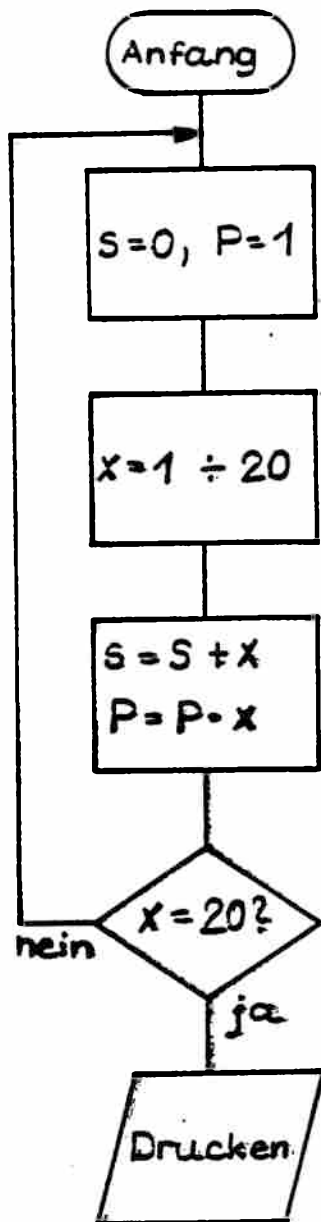
DI # 48

Löschen eines Files

SF # 48, NAME, P oder D

Programmbeispiel

Berechne Summe und Produkt der ganzen Zahlen von 1 bis 20.



```

100 LET P=1
110 LET S=0
120 FOR X = 1 TO 20
130 LET S = S + X
140 LET P = P * X
150 NEXT X
160 PRINT
170 PRINT
180 PRINT "Summe d.Zahlen 1 bis 20 ="; S
185 PRINT
190 PRINT "PRODUKT d.ZAHLEN 1 BIS 20="; P
200 END
  
```

AUSGABE:

Summe der Zahlen 1 bis 20 = 210

Produkt der Zahlen 1 bis 20 = 2.4329020 E+18

Programmierbeispiel

Um die Richtigkeit des Programms überprüfen zu können, müssen die Testbeispiele alle möglichen Fälle enthalten.

Wir wählen folgende Gleichungen:

$$x^2 + x + 1 = 0 \quad \text{Wurzeln: } x_1 = -\frac{1}{2} + \frac{\sqrt{3}}{2}j; \quad x_2 = -\frac{1}{2} - \frac{\sqrt{3}}{2}j$$

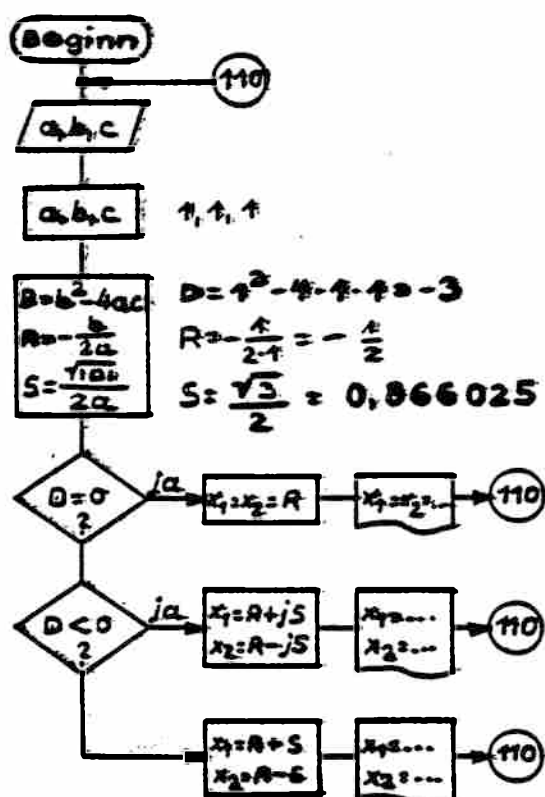
$$x^2 - 4x + 3 = 0 \quad x_1 = 3$$

$$x^2 - 2x + 1 = 0 \quad x_1 = x_2 = 1$$

Quadratische Gleichung

Die Lösungszahlen (Wurzeln) der quadratischen Gleichung $ax^2 + bx + c = 0$ sind zu ermitteln.

Zunächst überlegen wir uns die möglichen Fälle und halten dann den Programmablauf graphisch fest.



$$x_1 = -\frac{1}{2} + j \cdot 0,866025$$

$$x_2 = -\frac{1}{2} - j \cdot 0,866025$$



Programm:

```
100 PRINT " ", "A * X**2 + B * X + C = 0"
110 A, B, C
120 PRINT
130 PRINT "A ="; A, "B ="; B, "C ="; C
140 D = B*B - 4*A*C
150 R = -B/2/A
160 S = SQR(ABS(D))/2/A
170 IF D = 0 THEN 210
180 IF D < 0 THEN 230
190 PRINT "X1 ="; R+S, "X2 ="; R-S
200 GOTO 110
210 PRINT "X1 = X2 = "; R
220 GOTO 110
230 PRINT "X1 ="; R, "+ J*"; S, "X2 ="; R, "- J*"; S
240 GOTO 110
250 DATA 1,1,1, 1,-4,3, 1,-2,1
```

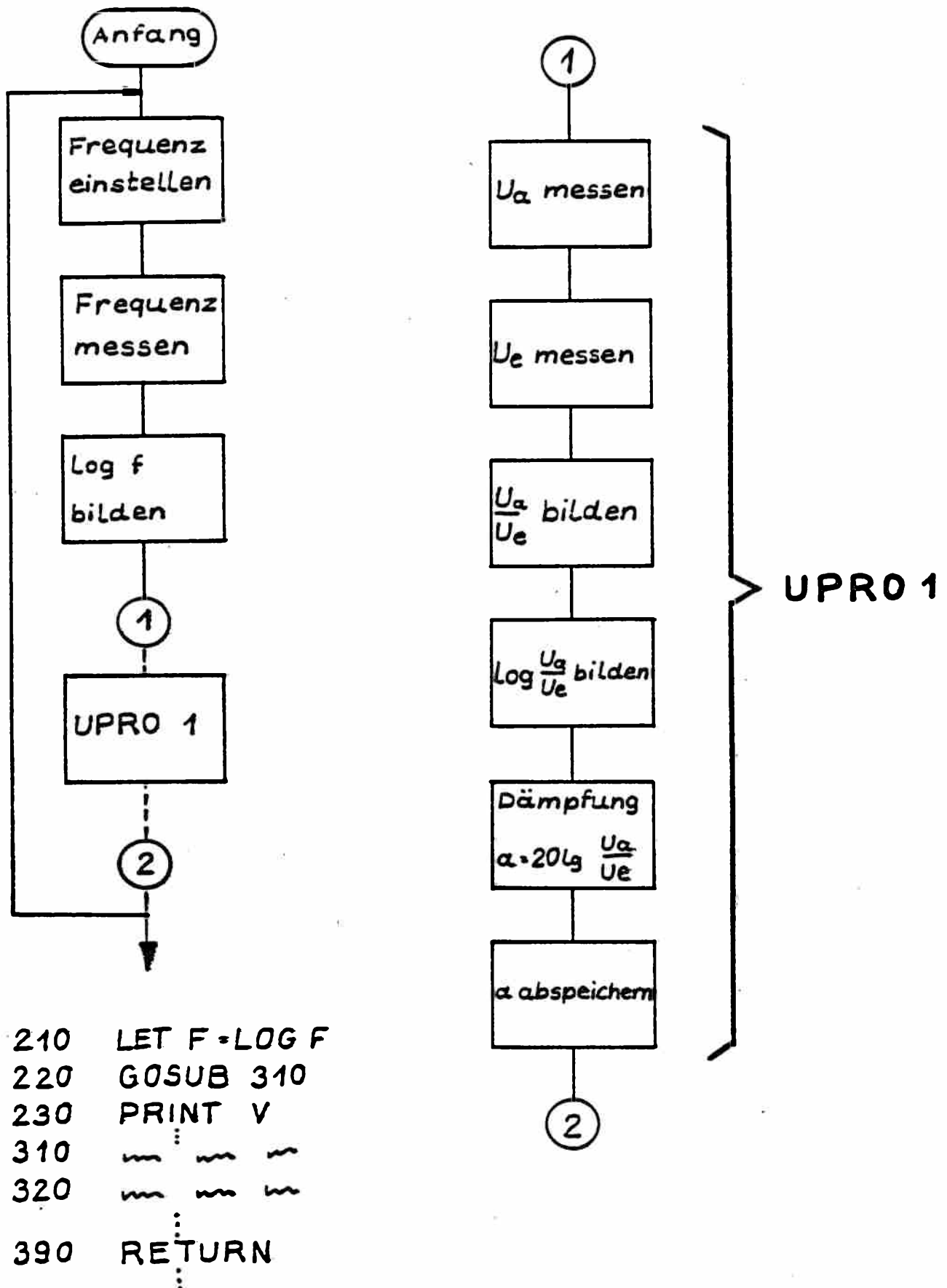
Ausgabe: $A * X^{**2} + B * X + C = 0$

A = 1	B = 1	C = 1
X1 = -.5	+ J* .866025	X2 = -.5 - J* .866025

A = 1	B = -4	C = 3
X1 = 3	X2 = 1	

A = 1	B = -2	C = 1
X1 = X2 = 1		

Beispiel für Sprung in Unterprogramm (GOSUB - RETURN)





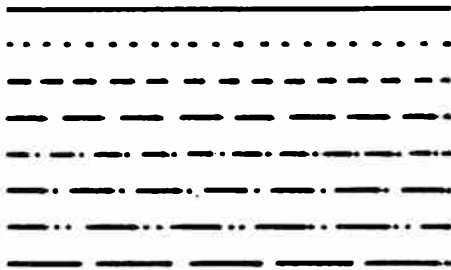
Plotter PM 8150

- DIN A3 -Format
- μP gesteuert
- IEC-Interface o. V-24 Interface
- Buffer 1600 bytes
- Window scaling
- 8 verschiedene Linientypen
- 8 versch. Symbole (Markierungen)
- Plott Rate > 400 mm/s per axis
- Addressierb. Auflösung 0,1 mm
- Stift-Bewegung 5 cycl./s
- Papierhaltung elektrost.
- Plott Fläche: x-Axe > 381 mm
y-Axe > 254 mm

PM 8150

Über IEC - Bus als Talker und Listener zu betreiben (z.B. Abfahren von Kurven)

- Zeichnen von Kurven
- Kurven abfahren
- Schriftgröße verstellbar in Höhe u. Breite
- Verstellbare Schriftlage
- 8 verschiedene Linientypen, dadurch Übersichtlichkeit beim Kopieren

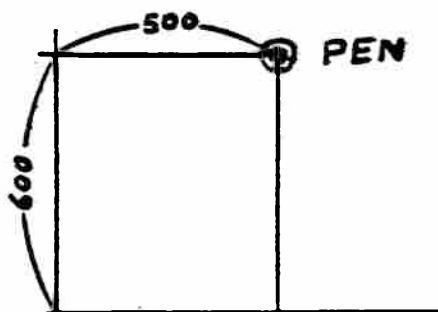


- 8 verschiedene Markierungszeichen in (ROM's) zur Kennzeichnung



Kommandos

MOVE #3: 500, 600

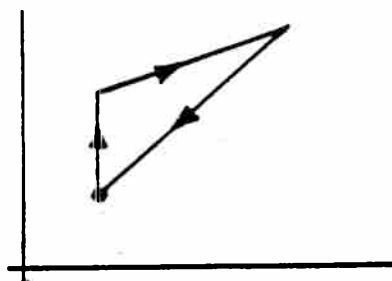


Stiftbewegung
zu einem vorgewählten
Punkt



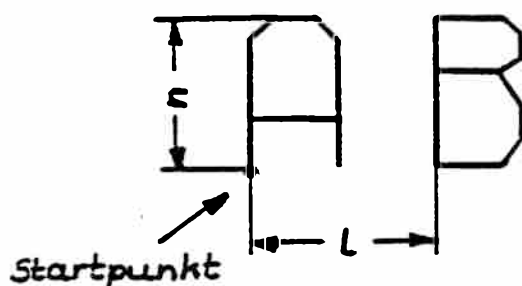
PHILIPS

DRAW #3: 500, 500, 1500, 2500, 2500



ALPHA SCALE #3: 50

Höhe 5mm

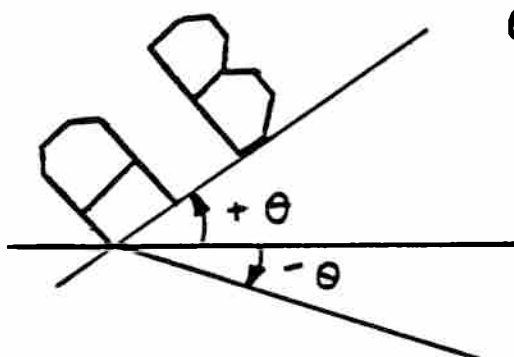


ALPHA SPACE #3: 60

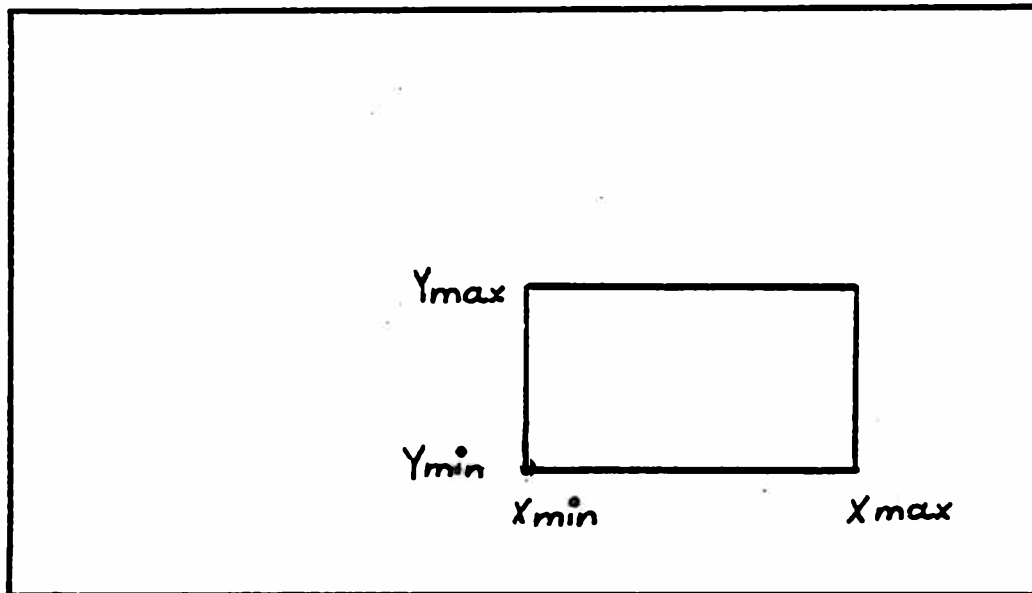
Abstand 6mm

ALPHA ROTATE #3: θ

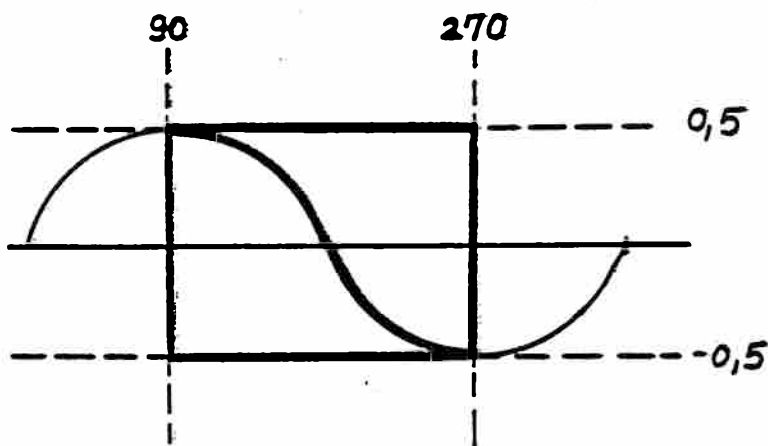
$\theta : 0^\circ - + 360^\circ$



VIEWPORT # LA: X1, X2, Y1, Y2
in mm x 10



WINDOW # LA: Xmin, Xmax, Ymin, Ymax



WINDOW # LA: 90; 270; - 0,5; + 0,5

Vergrößerung und Verkleinerung wird
selbständig durchgeführt

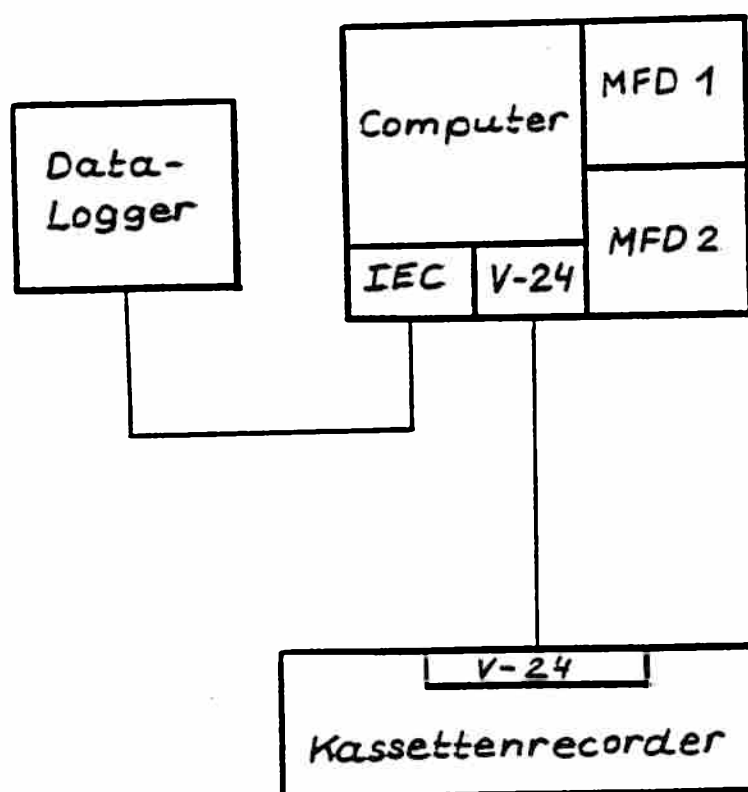


Schnelldrucker PM 4490

- Druckgeschwindigkeit: 160 Z/sec
- Matrix 9x9
- Zeichen / Zeile: 136
- Line Feed: Einzelschaltung
u. Dauerfunktion
- Übertragungsrate: 75, 110, 300, 1200,
1800, 2400, 4800 oder 9600
bits/sec
- Interface: V24 (RS 232C)
- Papierbreite: einstellbar zw. 4 u. 15 inch
- Papierform: Faltpapier
- Papiervorschub: durch Traktoren

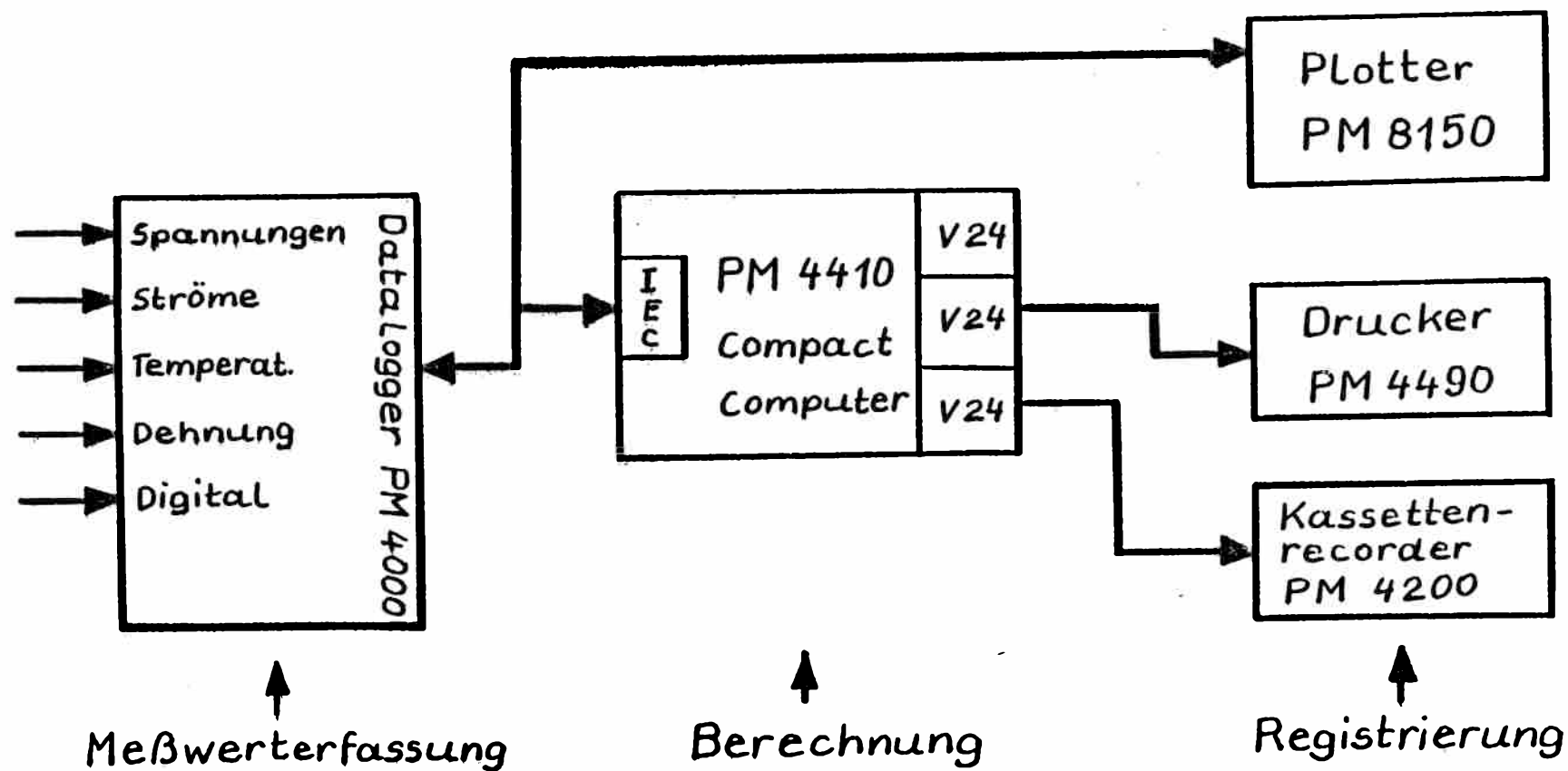
Kassettenrecorder PM 4200

- Speichereinheit für digitale Daten
- Datenträger auf Kassette nach ECMA34
- Fehlerrate 1 auf 10^9 Bit
- Daten zeichenseriell/bitparallel oder zeichenseriell/bitseriell



- Speicherkapazität 500 KByte/Kassette

Autom. Meßplatz





available versions	options	peripherals	standard software	software options	cable connections
PM 4410/11 16k memory 1 diskdrive	PM 4400/20 16k memory	PM 4490 fast character printer	system diskettes (5 equal diskettes), with 5 empty diskettes packed by for every PM 4410	PM 4410/90 BASIC software std. (10 diskettes)	PM 9480 IEC bus cable length 1m
PM 4410/12 32k memory 1 diskdrive	PM 4400/21 8k memory	PM 8150 intelligent digital plotter		PM 4410/91 BASIC software std. + matrix support	PM 9481 IEC bus cable, length 2m
PM 4410/13 16k memory 2 diskdrives	PM 4400/30 1 diskdrive	PM 4200/50 cassette recorder		PM 4410/92 BASIC software + plotter support	PM 9482 IEC bus cable, length 4m
PM 4410/14 32k memory 2 diskdrives	PM 4400/40 serial control unit V24/current loop interface	PM4491 paper tape recorder		PM 4410/93 BASIC software std. + matrix support + plotter support	PM 9483 IEC/IEEE cable adapter, length 1m
	PM 4400/50 IEC bus interface	PM 4492 PROM programmer			

- examples:
1. The option PM 4400/30, minifloppy diskdrive, converts the PM 4410/12, which was factory equipped with 32k memory and 1 diskdrive, to PM 4410/14.
 2. For the compact computer with plotter you need PM 4400/... with standard software packed by, PM 8150 intelligent digital plotter PM 4410/92 BASIC software + plotter support PM 9480 (9481, 9482) IEC bus cable, length 1m (2m, 4m).

fig.

available versions
options
peripherals
standard software
software options
cable connections

PHILIPS



Programmbeispiel

PM 4400, Plotter, Drucker

```
100 REM          plott-programm:
105 PAGE
106 INIT
107 REM          setzen v. konstanten und anfangswerten:
110 LET PI=4*ATN(1)
130 DIM A(59)
140 LET MAX=-1E+99
150 LET MIN=1E+99
155 REM          loop zum einnehmen der werte:
160 FOR I=1 TO 59
170 LET A=COS(PI/59*2*I)

175 REM          max u. min erkennen und speichern:
180 IF A>=MIN THEN 200
190 LET MIN=A
200 IF A<=MAX THEN 220
210 LET MAX=A
215 REM          eingew. wert ins array:
220 LET A(I)=A
230 NEXT I
235 REM          normierung der vertikalen v. 1 b. 20:
240 LET K1=20-19*MAX/(MAX-MIN)
250 LET K2=19/(MAX-MIN)
255 REM          loop fuer die "vert.-ablenkung":
260 FOR I=20 TO .9 STEP -.1
270 PRINT
275 REM          bedingung fuer die beschriftung d. vert.-achse:
280 IF I=20 THEN 310
290 IF I=2 THEN 330
300 GOTO 340
310 PRINT ///IMAX;TAB(18);
320 GOTO 340
330 PRINT MIN;TAB(18);
340 PRINT TAB(18);"I";
345 REM          loop zur quantisierung u. "horiz.-abl.":
350 FOR J=1 TO 59
360 LET Y=K1+K2*A(J)
370 IF Y>I THEN 400
380 IF Y<=I-1 THEN 400
390 PRINT TAB(18+J);"*"
400 NEXT J
410 NEXT I
420 PRINT TAB(18);"-----"
430 END
:EOS
```



```
100 REM ***** plott-programm mit pm2526
110 PAGE
120 INIT
130 REM ***** setzen v. konstanten und anfangswerten:
140 DIM A(59)
150 LET MAX=-1E+99
160 LET MIN=1E+99
170 REM ***** loop zum einnehmen der werte:
180 FOR I=1 TO 59
190 PRINT #1,8:
200 INPUT #1:A
210 REM ***** max u. min erkennen und speichern!
220 IF A>=MIN THEN 240
230 LET MIN=A
240 IF A<=MAX THEN 270
250 LET MAX=A
260 REM ***** eingen. wert ins array!
270 LET A(I)=A

280 NEXT I
290 REM ***** normierung der vertikalen v. 1 b. 20:
300 LET K1=20-19*MAX/(MAX-MIN)
310 LET K2=19/(MAX-MIN)
320 REM ***** loop fuer die "vert.-ablenkung":
330 FOR I=20 TO .9 STEP -.1
340 PRINT
350 REM ***** bedingung fuer die beschriftung d. vert.-achse:
360 IF I=20 THEN 390
370 IF I=1 THEN 410
380 GOTO 420
390 PRINT ///;MAX;TAB(18);
400 GOTO 420
410 PRINT MIN;TAB(18);
420 PRINT TAB(18);"I"
430 REM ***** nested loop zur quantisierung u. "horiz.-abl.":
440 FOR J=1 TO 59
450 LET Y=K1+K2*A(J)
460 IF Y>I THEN 490
470 IF Y<=I-1 THEN 490
480 PRINT TAB(18+J);" "
490 NEXT J
500 NEXT I
510 PRINT
520 PRINT TAB(18);" "
530 END
:E05
```