

nascocom journal

Zeitschrift für Anwender des NASCOM 1 oder NASCOM 2

3. Jahrgang · März/April 1982 · Ausgabe 3/4

Herausgeber:

MK-SYSTEMTECHNIK Michael Klein · Pater-Mayer-Straße 6 · 6728 Germersheim/Rhein
Telefon (0 72 74) 20 93 · Telex 453500 mks d

MK-SYSTEMTECHNIK Thomas Gräfenecker · Kriegsstraße 164 · 7500 Karlsruhe · Telefon (07 21) 2 92 43
MK-SYSTEMTECHNIK Michael von Keltz · Pfaffenberg 4 · 5650 Salingen 1 · Telefon (0 21 22) 4 72 67

Der Heftpreis beträgt DM 5,—. Ein Abonnement erhalten Sie für DM 48,— im Jahr. Dafür bekommen Sie 12 Hefte pro Jahr, bzw. 10 Hefte (zwei dicke Doppelausgaben).
Die Autoren sind für den Inhalt Ihrer Beiträge selbst verantwortlich.

INHALT

2	Leserbriefe	
	LOAD für NASSYS 3	Christian Peter
3	Cursor für DOS-BASIC	W.Mayer Gürr
	FORTH Teil 6	Günter Kreidl
9	Datenverwaltung	Günter Böhm
20	RAM-Graphik	J.C.Lotter
24	Schnelle Tastaturabfrage	Josef Zeller
25	Haushaltsbuchführung	Klaus Mombaur
32	Gewinner d.Preisausschr	
33	Spielautomat	Wolfgang Schröder
36	Bowling	Claus Stehlik
37	Spacepotatoes	Claus Stehlik
38	Groß/Laufschrift	Jürgen Loh
40	Doppelwurm	Michael Bach
41	BASIC-Hilfen	Wolfgang v.Jan
42	3-D-Irrgarten	Rüdiger Maurer
43	Bandlaufzähler	Erich Mehnert
44	NASSYS-Erweiterung	Otto Föbel
46	Flipper	Peter Brendel
47	NASCOM Journal INTERN	
48	Super-Adventure	Wolfgang Schröder
49	Türme von Hanoi	Eberhard Horch
50	Promille	Christian Peter
51	NASCOMPL/Impressum	
52	MK Produkte	

LESERBRIEFE

Meinen Glueckwunsch fuer die gelungene Niveaueanhebung des NASCOM Journals! Was mich besonders interessierte und begeisterte, waren die Beitraege ueber den PIO-Bus und die bereits dafuer vorhandenen Schaltungen, die MDCR-Beitraege von Herrn Lotter und und und ...

Ein bisschen enttaeuscht bin ich ueber den "leicht" unregelmassigen Erscheinungstermin und fast erschrocken bin ich ueber die Beitraege "Lottoprogramm" und "Lottozahlen-generator", deren Niveau doch hoffentlich nicht zur Gewohnheit wird.

Was ich auch sehr gut finde, sind die "Seiten fuer Einsteiger". Als ich mit dem Nascom anfang, habe ich genau diese Art von Hilfen vergebens gesucht. Nachdem ich das sehr begreusse, wuerde ich mich auch bereit erklaren, ein paar "kleine" Beitraege dazu zu liefern. Ich glaube schon, dass ich auf dem Hardware-Sektor kompetent genug dafuer bin. Sollten Sie einmal einen "Autor" fuer ein Thema suchen, koennen Sie es ja einmal mit mir versuchen.

Bis dann verbleibe ich mit freundlichen Gruessen

Otto Foessel, Erlangen

Meinen allerherzlichsten Glueckwunsch zur neuen Aufmachung des "Nascom Journals". Es hat wieder deutlich an Qualitaet gewonnen.

Ich muss meinen letzten Brief sehr missverstaendlich formuliert haben; gerade das Programm

Meinen allerherzlichsten Glueckwunsch zur neuen Aufmachung des "Nascom Journals". Es hat wieder deutlich an Qualitaet gewonnen.

Ich muss meinen letzten Brief sehr missverstaendlich formuliert haben; gerade das Programm, das ich 1:1 von den INMC-80 NEWS uebernommen habe, haben Sie unter meinem Namen abgedruckt, waehrend das von mir umgeschriebene Programm "LOAD" nicht veroeffentlicht wurde. Nachdem ich mich nicht mit fremden Lorbeeren schmuecken moechte, bitte ich Sie um den kleinen Hinweis, dass das in der Nummer 1/82 veroeffentlichte Programm "Pruefsumme" aus den INMC-80 NEWS stammt.

Ich freue mich, dass die veraenderte Version von "Yatzi" soviel Anklang findet. Es ist auf Seite 1 der beigefuegten Cassette enthalten. (Fuer die Leser steht es als "Rundlauf" zur Verfuegung. Red.)

Ich moechte mich uebrigens fuer die rege schriftliche Kommunikation bedanken, die mir z.B. beim INMC abgeht. (Alle Briefe an den INMC blieben bisher unbeantwortet. Ja, sogar Vorauszahlungen fuer Programme wurde einfach ignoriert!)

Christian Peter, Wien

Zu Ihrem Programm "LOAD": Wir wollten es zunaechst nicht veroeffentlichen, da kurz zuvor das Programm CONMOD mit gleicher Funktion gedruckt wurde. Nachdem die Editiermoeglichkeiten aber bei Ihrem Programm etwas anders sind, drucken wir es nun doch in diesem Heft ab.

Vielen Dank auch noch fuer die anderen Programme auf Ihrer Cassette. Sie werden im Laufe der Zeit noch im Journal erscheinen. Red.

PROGRAMM "LOAD"

```
START DES PROGRAMMS: E555 MMMM
SSSS: Beginn des Programms "LOAD"
MMMM: erste Speicheradresse des Listings

begin 0000 2A 0E 0C LD HL, (ARG2)
      0003 22 0C 0C LD (ARG1), HL
      0006 EF 0C 00 CLS
      0009 18 09 JMP "next1"
newad 000B 2A 0C 0C LD HL, (ARG1)
      000E D5 PUSH DE
      000F 11 08 00 LD DE, 0080
      0012 19 ADD HL, DE
      0013 D1 POP DE
next1 0014 DF 66 SCAL TBCD3
input 0016 DF 7B SCAL BLINK
      0018 FE 2E CP "."
      001A 28 49 JZ "endck"
      001C FE 0D CP CR
      001E 28 03 JZ "lnend"
outpt 0020 F7 RST ROUT
      0021 18 F3 JMP "input"
lnend 0023 2A 29 0C LD HL, (CURSOR)
      0026 DF 7C SCAL CPOS
      0028 EB EX DE, HL
      0029 DF 79 SCAL RLIN
      002B 38 2F JC "error"
      002D 3A 0B 0C LD A, (ARGN)
      0030 FE 0A CMP A
      0032 20 28 JNZ "error"
      0034 21 0C 0C LD HL, ARG1
      0037 AF XOR A
      0038 06 12 LD B, 12
chcks 003A 86 ADD A, (HL)
      003B 23 INC HL
      003C 10 FC DJNZ "chcks"
      003E BE CMP (HL)
      003F 20 1B JNZ "error"
      0041 2A 0C 0C LD HL, (ARG1)
      0044 11 0E 0C LD DE, ARG2
      0047 06 08 LD B, 08
```

trans	0049	1A	LD A, (DE)
	004A	77	LD(HL),A
	004B	23	INC HL
	004C	13	INC DE
	004D	13	INC DE
	004E	10 F9	DJNZ "trans"
	0050	EF 1B 00	ESC
	0053	2A OC OC	LD HL, (ARG1)
	0056	DF 66	SCAL TBCD3
	0058	DF 6A	SCAL CRLF
	005A	18 AF	JMP "newad"
error	005C	DF 6A	SCAL CRLF
	005E	DF 6B	SCAL ERRM
	0060	2A OC OC	LD HL, (ARG1)
	0063	18 AF	JMP "nextl"
endck	0065	F7	RST ROUT
	0066	DF 7B	SCAL BLINK
	0068	FE OD	CMP CR
	006A	20 B4	JNZ "outpt"
	006E	F7	RST ROUT
	006F	DF 5B	SCAL MRET

VERKAUFE NASCOM 1 , 32K RAM, BASIC
 ZEAP 4, TOOLKIT, Debugger auf EPROM,
 Kansas-City-Interf. VB 1000.-
 Siemens T100- TTY 100 Baud, Großschr.,
 100 Zeichen/Zeile VB 200.-
 Olivetti TE 318 ASCII- TTY 110Bd VB 600.-
 H.Molle Tel. [REDACTED]
 [REDACTED]
 [REDACTED]

Basic

von W. Mayer-Gür

Adressierbarer Cursor für DOS-Basic

Bei dem CLD-DOS Betriebssystem erfolgt die Ein- und Ausgabe über externe Geräte (Bildschirm, Tastatur, Drucker usw.) durch sogenannte Devicedriver. Die Daten laufen dabei durch eine Softwareschnittstelle. Hardwareänderungen erfordern nur eine Anpassung des entsprechenden Devicedrivers. Der Vorteil liegt in der Austauschbarkeit von Programmen. Deshalb sollte man es möglichst vermeiden, Eingriffe im Kern des Betriebssystems vorzunehmen.

Die Basic-Versionen, die unter CLD-DOS laufen, sind ursprünglich für Terminals, deren Cursor mit Escape Sequenzen gesteuert wird, geschrieben worden. Beim Nascom liegt die Bildschirmsteuerung getrennt vom TT-Devicedriver im RAM ab 0C00H; das Escapezeichen (ASCII 27) löscht nur den Bildschirm.

Für manche Zwecke ist ein Scrollen des Schirms unerwünscht (z.B. Schablonen für Formulare). Hier hilft ein adressierbarer Cursor. Das kleine Basic Unterprogramm greift in zwei Bereichen in das Betriebssystem ein. In der RAM Adresse 8408 liegt der Spaltenzähler für jedes beliebige Terminal. 3213/4 ergibt die aktuelle Position des Cursors im Videoram. Der Bildschirm wird mit seinen Koordinaten so organisiert, daß X=1 und Y=1 in der linken oberen Ecke liegt.

```

100 IF X<1 OR X>48 OR Y<1 OR Y>16
    THEN PRINT"Nicht moeglich!":STOP
110 IF Y=1 THEN Y=17
120 P=X-1+(Y-1)*64+1994
130 A=PEEK(3213)+PEEK(3214)*256
140 POKE 8408,X
150 POKE 3213,P MOD 256
151 REM * beim CLD-Basic statt Zeile 150
152 REM * 150 POKE 3213,P
160 POKE 3214,P/256
170 POKE A,32
180 RETURN

```

FORTH für den NASCOM

Teil 6 von Günter Kreidl

Mit diesem Beitrag und einigen im nächsten Heft erscheinenden Programmbeispielen soll diese Reihe zunächst einmal beendet werden. Um den Gebrauch des Systems zu erleichtern, habe ich ein ausführliches Verzeichnis aller verfügbaren Funktionen zusammengestellt. Dazu einige Hinweise:

In der Spalte "Parameter" sind die Eingangsdaten verzeichnet, die eine Routine auf dem Stack vorfinden muß, und zwar von links nach rechts in der Reihenfolge, wie sie (vom Benutzer oder von einer vorgehenden Routine) auf den Stack gegeben werden müssen. Die Reihenfolge auf dem Stack ist dann natürlich genau umgekehrt, d.h. der am weitesten rechts stehende Wert liegt oben auf dem Stack. Die Spalte Lit.(erale) Arg.(umente) ist deshalb nötig, weil in FORTH die UPN nicht vollständig durchgehalten wird. Werden als Argumente einer Funktion "Worte", z.B. Funktionsnamen, gebraucht, so folgen diese Worte auf die Funktion. In der Spalte "Ergebnis" sind die Parameter aufgeführt, die eine Funktion auf dem Stack hinterläßt, und zwar in der gleichen Reihenfolge wie die Eingangsdaten. Ich hoffe, daß die Kommentare ausführlich genug sind. Falls Fragen auftauchen sollten, stehe ich gerne zur Verfügung.

Parameter	Funktionsname	Lit.Arg.	Ergebnis	Kommentar
SYSTEMSCHLEIFEN (MODI)				
-	INTERACT	-	-	Benutzer-Interpreter; wartet auf Eingaben bis NEWLINE, führt dann die Operationen aus, druckt den obersten Stackwert und "-" als Promptzeichen. Wird bei Systemstart aufgerufen.
-	INTERPRETER	-	-	Vereinfachter Interpreter ohne Stackausdruck und Promptzeichen.
-	EDITOR	-	-	Bildschirm-Editor; erlaubt die Anwendung aller NAS-SYS-Editierbefehle; wird der Cursor an den Anfang einer Zeile gesetzt und "Control P" eingegeben, so wird der INTERPRETER aufgerufen, der nach der Ausführung aller Operationen zum EDITOR zurückkehrt. Siehe auch "EDITOR-Kommandos"!
-	ENTRY	Name	-	Ist selbst keine Schleife, sondern ruft den Modus (Name) auf; z.B.: ENTRY EDITOR.
EDITOR-KOMMANDOS				
-	,	-	-	Verknüpft 2 Zeilen miteinander, da sonst vom INTERPRETER nur jeweils eine Zeile abgearbeitet wird. Ausnahme Compilermodus
-	SAVESCREEN	-	-	Rettet den Bildschirminhalt nach 2800H.
-	GETSCREEN	-	-	Holt den Bildschirminhalt zurück.
-	SWAPSCREEN	-	-	Vertauscht den aktuellen Bildschirminhalt mit dem bei 2800H zwischengespeicherten.
-	READC	-	-	Liest eine Bildschirmseite von Cassette.
a b	WRITEC	-	-	Schreibt b Zeilen beginnend bei Zeile a vom Bildschirm auf Cassette.
a b	PRINT	-	-	Gibt b Zeilen beginnend bei Zeile a auf den Drucker aus. Bei C80H muß ein Sprung auf die Druckeroutine stehen.
DEFINIERENDE FUNKTIONEN				
-	;	Name	-	Ruft den Compiler auf, der (Name) in das Dictionary einträgt und die folgenden Funktionen und Zahlen compiliert, bis er auf ein ";" trifft.
-	MCODE	Name	-	Trägt (Name) in das Dictionary ein und ruft das MODIFY-Kommando von NAS-SYS auf. Dann können Programme im Maschinencode eingegeben werden. Sie dürfen nicht mit RET oder JP NEXT abgeschlossen werden, da dies MCODE selbst besorgt!
a	VARIABLE	Name	-	Erzeugt eine Variable mit dem Anfangswert a und trägt deren Namen in das Dictionary ein. Bei Aufruf wird nicht der Wert der Variablen, sondern deren Adresse auf den Stack gegeben!
a	CONSTANT	Name	-	Erzeugt eine Konstante mit dem Wert a und trägt (Name) in das Dictionary ein. Bei Aufruf der Konstante wird deren Wert auf den Stack gegeben.
-	MESSAGE	Name	-	Trägt (Name) ins Dictionary ein und erwartet die Eingabe eines beliebig langen

Parameter	Funktionsname	Lit.Arg.	Ergebnis	Kommentar
-	DEL	-	-	ASCII-Strings, der mit ''' abgeschlossen wird. Im EDITOR-Modus muß der String vorher ab der nächsten Zeile eingegeben werden. Bei Aufruf von (Name) wird die Adresse des Strings auf den Stack gegeben.
-	::	Name	-	Löscht die zuletzt definierte Funktion. Wie ":", doch (Name) wird ins Compiler-Dictionary eingetragen. Kann nicht mit DEL gelöscht werden!
BETRIEBSSYSTEM-SCHNITTSTELLE				
a	NAS-SYS	-	-	Ruft die NAS-SYS-Routine mit der Codenummer a auf. Die Argumente müssen vorher nach CoCH, CoEH, C1oH (= 3o84, 3o86 3o88 Dezimal) gepoked werden!
a	MODIFY	-	b	Ruft das M-Kommando von NAS-SYS auf ab Adresse a. Mit b wird anschließend die nächste freie Speicherstelle übergeben.
-	MOTOR	-	-	Schaltet den Cassettenrecorder ein/aus.
-	GRKL	-	-	Schaltet die Tastatur um von Groß- auf Kleinschrift und umgekehrt.
STACKOPERATIONEN				
a	DUP	-	a a	Dupliziert den Wert auf dem Stack.
a	POP	-	-	Entfernt den Wert auf dem Stack.
a b	SWAP	-	b a	Vertauscht a und b auf dem Stack.
a b c	ROT	-	b c a	Rotiert a nach oben.
-	CLEAR	-	-	Setzt den Daten-Stackpointer auf den Anfangswert - Alle Daten weg!
a b	OVER	-	a b a	Kopiert den zweiten Wert auf dem Stack "oben drauf".
a b	UNDER	-	b a b	Kopiert den obersten Wert nach unten.
ARITHMETISCHE, LOGISCHE und VERGLEICHOPERATIONEN				
a b	+	-	c	$c = a + b$
a b	"-"	-	c	$c = a - b$; Die Anführungszeichen sollen klarmachen, daß hier mit "-" sonst gemeint ist: kein Argument!
a b	*	-	c	$c = a * b$
a b	/	-	c	$c = a$ dividiert durch b
a b	MOD	-	c	c = Rest der Division a/b
a b	/MOD	-	c d	$c = a / b$; d = Rest von a/b
a	INC	-	b	$b = a + 1$
a	DEC	-	b	$b = a - 1$
a	MINUS	-	b	b = Komplement von a
a b	OR	-	c	$c = a$ ODER b; Byteweise logische Funktion a und b müssen kleiner als 256 sein!
a b	AND	-	c	$c = a$ UND b; Arg. wie oben
a b	XOR	-	c	$c = a$ EXCLUSIV-ODER b; Arg. wie oben
a b	EQ	-	c	$c = 0$ wenn a ungleich b, sonst $c = 65535$
a	EQZ	-	b	$b = 0$ wenn a ungleich 0, sonst $b = 65535$; entspricht auch der logischen Verneinung.
a b)	-	c	$c = 65535$, wenn a größer als b, sonst 0
a b	(=	-	c	$c = 65535$, wenn a kleiner oder gleich b, sonst $c = 0$.
a b	MIN	-	c	$c = a$, wenn a kleiner als b, sonst $c = b$.

Parameter	Funktionsname	Lit.Arg.	Ergebnis	Kommentar
a b	MAX	-	c	c = a, wenn a größer als b, sonst c = b
SPEICHEROPERATIONEN				
a	PEEKW	-	b	b = Wert an der Adresse a; 16-Bit-Wort
a	PEEKB	-	b	b = Wert an der Adresse a; 8-Bit-Byte
a b	POKEW	-	-	Schreibt 16-Bit-Wort a an die Adresse b.
a b	POKEB	-	-	Schreibt Byte a an die Adresse b.
a b	MEM+	-	-	Addiert a zum Wert an der Adresse b.
a b c	MOVEBYTES	-	-	Kopiert c Bytes von a nach b.
a b c	FILL	-	-	Schreibt b-mal das Byte c in den Speicher beginnend bei Adresse a.
a b	ERASE	-	-	Schreibt b Nullen in den Speicher ab Adresse a.
a b	BLANKS	-	-	Schreibt b Leerzeichen (20H) in den Speicher ab Adresse a.
EIN-/AUSGABE (siehe auch EDITOR-KOMMANDOS)				
-	CIN	-	a	a = Zeichen, das von der Eingaberoutine, deren Startadresse als Wert der Variablen INVAR gespeichert ist, geholt wird. Bei Systemstart BLINK (NAS-SYS!).
a	COUT	-	-	Gibt das Zeichen a aus über die Ausgaberroutine, deren Startadresse als Wert der Variablen OUTVAR gespeichert ist. Bei Systemstart Ausgabe auf Bildschirm.
-	BLINK	-	a	Wartet mit blinkendem Cursor auf Eingabe von der Tastatur bzw. vom UART und gibt das Zeichen (a) auf den Stack.
-	BLADR	-	a	Konstante, a = Startadresse von BLINK
-	NORMIN	-	a	a = Zeichen von der Eingaberoutine RIN (NAS-SYS)
-	NIN	-	a	Konstante, a = Startadresse von NORMIN
-	READSCR	-	a	Liest ein Zeichen (a) vom Bildschirm, die Adresse wird dem Cursorregister (C29H) entnommen. Die Zeichen werden gezählt, jedes 48. Zeichen durch NEWLINE ersetzt.
-	RSCR	-	a	Konstante, a = Startadresse READSCR
-	MISCHTEXT	-	a	wie CIN, nur werden Kleinbuchstaben und überflüssige Blanks ignoriert. Dadurch sind Kommentare in Kleinschrift möglich.
a	NORMOUT	-	-	Gibt das Zeichen a auf den Bildschirm aus, sofern der Ausgabezeiger von NAS-SYS nicht geändert wird.
-	OUTN	-	-	Setzt den Ausgabezeiger auf Bildschirm=ausgabe.
-	OUTC	-	-	Ausgabezeiger auf Bildschirm + Cassette.
-	OUTD	-	-	Ausgabezeiger auf Bildschirm + User-Out=put, UOUT auf C80H (Drucker!)
a	=	-	-	Druckt a als Dezimalzahl aus.
a	.	-	a	Wie "=", doch a bleibt auf dem Stack.
a	PRINTEX	-	-	Druckt a als Hex-Zahl aus.
a	PRINTM	-	-	Druckt den Wert an der Adresse a als Dezimalzahl aus.
a	PRINTS	-	-	Druckt den String auf den a zeigt; interessantes Format mit erstem Byte = Stringlänge.

Parameter	Funktionsname	Lit.Arg.	Ergebnis	Kommentar
a	TYPE	-	-	Druckt den String an der Adresse a aus; MESSAGE-Format, beliebig lang mit o als Ende-Zeichen.
a	PRINTDIC	-	-	Druckt das Dictionary mit der Startadresse a aus.
-	CR	-	-	Gibt NEWLINE aus.
-	CLS	-	-	Gibt CLEAR SCREEN aus.
-	SPACE	-	-	Druckt ein Leerzeichen aus.
a	SPACES	-	-	Druckt a Leerzeichen aus.
-	BACKSPACE	-	-	Gibt BACKSPACE (oBH) aus. .
-	PROMPT	-	-	Gibt "-" als Promptzeichen aus.
a	LINE	-	-	Setzt Cursor an den Anfang der Zeile a.
a	POSITION	-	-	Setzt Cursor an die Stelle a der jeweiligen Zeile.
a b	SCREEN	-	-	Setzt den Cursor in Zeile a, Position b.
-	WRITEM	-	-	Schreibt Programmcode und Dictionary in 2 Blöcken auf Cassette. (Beim Einlesen 2 mal R!)
-	'	Zeichen	a	a = (ASCII-)Wert von (Zeichen).
-	TERMINAL?	-	a oder a b	Fragt die Tastatur ab: wenn keine Eingabe vorliegt, a = o, sonst b = 65535 und a = Zeichen.

COMPILERFUNKTIONEN

a	IF ELSE THEN	-	-	Wenn a ungleich o, wird der auf IF folgende Code ausgeführt, sonst erfolgt ein Sprung nach ELSE bzw. nach THEN.
-	REPEAT UNTIL LOOP	-	-	UNTIL testet den obersten Stackwert und läßt das Programm mit dem auf LOOP folgenden Code fortfahren, wenn der Wert ungleich o ist. LOOP gibt immer einen Rücksprung nach REPEAT. UNTIL kann an beliebiger Stelle zwischen REPEAT und LOOP stehen.
a b	FOR LOOP	-	-	Programmcode zwischen FOR und LOOP wird (a - b + 1) mal wiederholt. b = Startwert a = Endwert des Schleifenindex.
-	;	-	-	Beendet die Compilation.
-	!	Funktion	-	(Funktion) wird ausgeführt statt compiliert. VORSICHT!
-	"	Compiler=funktion	-	(Compilerfunktion) wird compiliert statt ausgeführt.

WICHTIGER HINWEIS: Alle diese Funktionen können nur im Compilermodus verwendet werden. Sie werden dann während der Compilierung sofort ausgeführt, nicht compiliert wie alle übrigen Funktionen.

SYSTEM- und HILFSFUNKTIONEN

a	TESTKL	-	b	Prüft, ob es sich bei a um einen Kleinbuchstaben handelt; dann b = o, sonst b = 65535.
a	DELAY	-	-	Verzögerung abhängig vom Wert a.
a b	OUTLINES	-	-	Liest b Zeilen beginnend bei Zeile a vom Bildschirm und gibt sie aus.
a	EXECUTE	-	-	Führt das Programm an der Adresse a aus.
a	VAREX	-	-	Holt den Wert b von der Adresse a und

Parameter	Funktionsname	Lit.Arg.	Ergebnis	Kommentar
-	CADR	Funktion	a	führt das Programm an der Adresse b aus. a = Startadresse von (Funktion).
-	READLINE	-	-	Liest eine Zeile in den Textpuffer ein. Im EDITOR-Modus jeweils eine Bildschirm= Zeile, sonst von der Tastatur bis NEWLINE mit Echo auf dem Bildschirm.
-	GETWORD	Name	a	Holt das nächste Wort aus dem Textpuffer; wenn dieser leer ist, wird READLINE auf= gerufen. a = Wortadresse
a	FIRST	-	b c	a muß auf ein Wort im Textpuffer zeigen; c = erstes Byte des Wortes (= Wortlänge), b = Zeiger auf den Rest des Wortes.
-	SCAN	-	a oder b c	Holt das nächste Wort im Textpuffer an den Zeilenanfang; wenn Puffer leer, a = 0 sonst c = 65535 und b = Zeiger auf Wort.
a b	MATCH	-	a b c	Stringvergleich; c = 65535, wenn gleich, sonst c = 0.
a b	LOOKUP	-	c d	Sucht das Wort, auf das a zeigt, im Dic= tionary, auf das b zeigt; d = 65535; wenn gefunden, und c = Funktionsadresse; sonst d = 0 und c = a.
a	CONAXB	-	b c	Dezimal-Binär-Umwandlung; a zeigt auf De= zimalstring. Wenn Umwandlung möglich, c = 65535 und b = binärer Wert, sonst c = 0 und b = a.
a	CONEXA	-	b	Wandelt a in Dezimalzahl, auf die b zeigt
a	ENTER	-	-	Trägt Funktionsnamen, auf den a zeigt und Adresse der neuen Funktion in das Inter= preter-DictionarY ein.
a	CMPLW	-	-	Compiliert a (= Adresse einer Funktion).
-	I	-	a	a = aktueller Wert des innersten Schlei= fenindex
-	J	-	a	wie I, aber "zweitinnerste Schleife"
-	K	-	a	wie I, aber "drittinnerste Schleife"
SYSTEM-VARIABLEN und KONSTANTEN				
-	VARBL	-	a	a = Adresse der VARIAB-Funktion, die die auf sie selbst folgende Adresse gibt.
-	CONS	-	a	a = Adresse der CONS-Funktion (wie VARIAB nur Inhalt der folgenden Adresse)
-	NEGONE	-	a	a = 65535
-	ONE	-	a	a = 1
-	ZERO	-	a	a = 0
-	MEMORY	-	a	Adresse a enthält Obergrenze des bel. RAM
-	EXEC	-	a	Adresse a enthält Startadresse der Sys= temschleife.
-	NAMES	-	a	Adresse a enthält Adresse des Int.-Dict.
-	CODEADR	-	a	Adresse a enthält nächste freie Programm= codeadresse.
-	NEXTADR	-	a	a = Adresse der Funktion NEXT
-	CNAMES	-	a	a = Adresse der Adresse des Comp.-Dict.
-	INVAR	-	a	Adresse a enthält Adresse für COUT.
-	OUTVAR	-	a	Adresse a enthält Adresse für CIN.
-	CURSADR	-	a	a = Adresse des Cursorregisters (C29H)

Datenverwaltung

von Günter Böhm

Eben frisch aus dem Assembler: ein Datenverarbeitungsprojekt, das noch nicht als fertiges Programm anzusehen ist, sondern als Grundstock für ein komfortables Programm, an dessen Fertigstellung hoffentlich viele Journal-Leser mitarbeiten werden. Keine Angst, nicht wieder einer der theoretischen Vorschläge, die nie in die Praxis umgesetzt werden. Das Programm ist schon in der vorliegenden Form praktisch anwendbar, sollte aber noch ausgebaut werden. Für die Anpassung an persönliche Bedürfnisse ist es modular aufgebaut und sollte auch im Ausbau flexibel sein.

Was das Programm schon kann:

Der Nachteil vieler mir bekannter Datenverwaltungsprogramme ist die Beschränkung auf bestimmte Datenformate, wie z.B. 25 Buchstaben bei Namen etc. Der Vorteil des vorliegenden Programms ist die Organisation nach "Files", deren Umfang nur durch die Bildschirmkapazität beschränkt wird, da der Schirm als Eingabepuffer verwendet wird. Die einzelnen Daten oder Namen können beliebig lang sein; sie werden durch Space oder New Line unterschieden. Beim Aufsuchen bestimmter Daten dient dann die Reihenfolge im File zur Unterscheidung. So kann also ein File zehn oder 500 Zeichen lang sein, es wird mit seiner Original-Länge gespeichert, ohne unnötigen Platz für überflüssige Spaces zu beanspruchen. Interessant ist auch die Möglichkeit, Files zu ändern und somit auf eine völlig andere Länge zu bringen; das Programm verkürzt oder verlängert seinen Platz und verschiebt die anderen Files entsprechend. Beim Löschen eines Files bleibt ein Space mit dem Bell-Zeichen (als Kennzeichnung für File-Ende) erhalten, so daß nach versehentlichem Löschen wenigstens durch die erhaltene Filenummer eine Wiederherstellung der verlorenen Daten möglich ist.

Nach der Eingabe von Daten können die Files in der vorliegenden Fassung folgendermaßen verarbeitet werden:

Alle Files werden ausgegeben. Dabei kann man zwischen der Ausgabe auf dem Bildschirm oder durch Drucker wählen. Bei der Druckausgabe besteht die Möglichkeit, die Files in der abgespeicherten Form oder in bestimmte Texte

oder Ausgabeformulare eingebunden zu drucken. Die Texte können genauso wie die Files eingegeben, geändert oder aufgelistet werden. Um das Programm zu verkürzen, werden dafür die gleichen Routinen verwendet, nur werden die Parameter für Text oder File zuvor vertauscht, da beide in verschiedenen Speichern abgelegt sind. Aber zur Aufbaubeschreibung kommen wir noch zu späterem Zeitpunkt! Das Schnecken-A und eine folgende zweistellige Ziffer bezeichnen die Nummer des Datenworts im File, unbezeichnete Buchstaben oder Ziffern werden wie im Formular ausgedruckt.

Es können bis zu 9 Formulare gespeichert sein, die mit Ihrer laufenden Nummer aufgerufen werden. Ihr Umfang kann bis zu kleinen Formschriften reichen und ist wiederum durch die Bildschirmkapazität begrenzt. Sie sehen, das Dateisystem geht ein gutes Stück über die Anlage eines Telefonverzeichnisses mit 20 Nummern hinaus.

Wie schon erwähnt, können Files oder Formulare geändert oder angehängt werden. Dabei wird eine mögliche Speicherüberschreitung durch "String too long" angezeigt, und die Änderung kann wiederholt werden.

Noch ein interessanter Aspekt ist die Suchroutine. Hier ist nicht nur ein Suchbegriff möglich, sondern eine Kombination von Wörtern mit UND- bzw. ODER- Verknüpfung in beliebiger Anzahl und Länge, nur durch Bildschirmkapazität eingeschränkt (klingt so bekannt!). Dabei gilt ein Space als UND, ein "/" als ODER. Beispiel: 1947 Max/Moritz gibt alle Files aus, die entweder Max oder Moritz UND 1947 enthalten.

Ein Return zu NASSYS und das Abspeichern der (evt. geänderten Datei) sind ebenfalls vorgesehen. Beim Speichern wird zweckmäßigerweise das Programm mit abgespeichert. Auf die paar Bytes kommt es auch nicht mehr an, und man spart sich so getrenntes Laden von Programm und Datei. Die Dateiparameter, wie Dateianfang - und Ende, Speicherende etc. werden ebenfalls auf Cassette festgehalten.

Was das Programm noch können soll:

Bisher ging es nur um die Verwaltung und das Aufsuchen von Files, aber nicht um eine wirkliche Verarbeitung der Daten. Es müßten also Routinen eingebaut werden, die Daten innerhalb eines Files berechnen, manipulieren, ablegen. Ebenso müßten Daten aller Files rechnerisch verglichen werden können.

Durchschnitt berechnen usw. Für manche Fälle muß eine formatierte Eingabe vorhanden sein, bestimmte Datenwörter müssen durch alle Files mit Cursor leicht geändert werden können (nicht Eingabe mit freibeweglichem Cursor wie vorliegend, sondern mit Cursor auf bestimmtem Wort.) Interessant wäre auch eine einfache "Programmiersprache" für die Verwaltung z.B. SUCHE X Y/Z PRINT NAME TELEFON, eine Sortieroutine (schneller als BASIC!) und und und....

Hier sind Sie als Leser aufgefordert, noch weitere Ideen beizusteuern oder sogar auszuarbeiten.

Zur Orientierung ein kleiner Hinweis zum Programmaufbau, eine ausführliche Beschreibung ist hier nicht möglich, kann aber auf Wunsch nachgeliefert werden.

Um die Parameterübergabe bei den einzelnen Programmteilen zu erleichtern, wurde ein Speicher angelegt, der die benötigten Informationen enthält. Sie müssen bei den Programmen darauf achten, welche Informationen aus dem Speicher geholt oder darin abgelegt werden. Die Daten sind folgendermaßen verteilt: "DATEI" enthält den Beginn des Speichers für die Parameter

"ASTFIL" enthält die Anfangsadresse des gerade bearbeiteten Files

"ALEFIL" enthält die File Länge (inclusive BELL als Endzeichen)

"DATEND" zeigt auf das augenblickliche Ende der Datei

"RAMEND" zeigt auf das Ende des verfügbaren Speicherplatzes (im Augenblick 5FFF)

"AUSGAB-1" enthält C3 und wird aufgerufen, wenn eine Ausgabe (z.B. Bildschirm, Drucker oder auch Ändern! des Files) benötigt wird)

"AUSGAB" enthält die jeweilige Adresse der Ausgaberroutine

Daran schließen sich die entsprechenden Parameter für die Formulare an; so entspricht "FORMPA" dem Speicher DATEI, er enthält den Beginn der Formulardatei

An die 5 Formular-Parameter schließt sich eine Adresse an, die die Nummer des augenblicklich benötigten Formulars enthält

Die folgenden 10 Bytes dienen dazu, die Parameter der Filedatei aufzunehmen, wenn gerade mit Formularen gearbeitet wird. Nach erfolgter Manipulation werden die Formulardaten wieder nach FORMPA etc geschleust und die alten Parameter wieder aus dem "Rettungspuffer" geholt. (Beachten Sie hier die

Routinen RETTEN und NORMAL). Ein einfacher Trick mit großer Wirkung.

Ein zusätzlicher Puffer bei C80 nimmt die Suchbegriffe auf. Dabei enthält BUFFER den Beginn des Suchspeichers, ENDZ enthält das Zeichen, bis zu dem ein File durchsucht werden soll (im Augenblick BELL für File Ende), BUFWD1 enthält die Adresse des Wortes im Buffer, nach dem gerade gesucht wird, und SUCHAN enthält die Anzahl der zu suchenden Wörter.

Alle Routinen sind als Unterprogramme angelegt (auch wenn sie im Listing als Hauptprogramm bezeichnet werden), die entweder vom Menue aufgerufen werden, oder sich gegenseitig aufrufen. Zukünftige Routinen sollten ebenfalls mit Return abschließen, um die Flexibilität zu erhalten.

EINGAB (1000) Einlesen von Daten und Speichern in Datei; Abschluß durch ESCAPE. Freibeweglicher Cursor, NL erzeugt ASCII 09 auf dem Schirm, um das Abspeichern unnötiger Spaces zu verhindern.

INIT (114D) lädt Ausgangsparameter zum Anlegen neuer File- und Form- Dateien; ruft EINGAB auf

ISTEND (11E3) vergleicht Filebeginn mit Dateiende. Ist Dateiende erreicht; Message und Ret zum Menue

ALLFLS (120F) gibt alle Files aus, errechnet die jeweilige Filenummer und gibt sie (wie bei SUCHEN) auf dem Schirm aus.

SCREEN (10ED) lädt File von Speicher auf Bildschirm. Verwandelt #09 in New Line.

SEARCH (122C) durchsucht ein File. Es kehrt zurück mit B=0, wenn alle Suchbedingungen erfüllt sind, sonst enthält B einen anderen Wert.

INITSE (1286) lädt die Suchbegriffe in den Puffer und errechnet die Anzahl aus der UND/ODER Verknüpfung.

SUCHEN (12EC) ruft SEARCH so lange, bis alle Files gefunden oder Dateiende erreicht ist.

PRTALL (133b) gibt alle Files aus.

PRTNBR (1378) druckt das Wort, dessen Nummer in B steht (BCD-Code)

FORM (13AD) druckt ein Formular und darin die angegebenen Wörter aus den Files.

CHAFOR (141D) lädt die gewählte Ausgaberroutine in den Speicher AUSGAB.

EXTEND (14C0) ermöglicht das Anhängen von Files an bestehende Datei.

CHANGE (14C9) springt ins Suchprogramm und wählt als Ausgaberroutine AENDER zum Ändern

bestehender Files.
 MENUE (Start des ganzen Programms bei 1510) erklärt sich von selbst
 RETTEN (16E8) und NORMAL (16FF) werden zum Arbeiten mit File/Formular benötigt (bereits zu Beginn erklärt)
 PRINT (1716) enthält eine Verzögerung nach der Ausgabe von NEW LINE für TTY. Hier kann eine beliebige Druckerroutine eingebaut werden.
 Hinzuweisen wäre noch auf die Verwendung von #0A für NEW LINE in den Druckausgaben. Hier muß wohl für die meisten Drucker #0D eingetragen werden. Die entsprechenden Stellen sind im Listing durch KSR im Kommentarteil gekennzeichnet.
 Und nun wünsche ich mir nur noch einige Leser, die durch Ihre Beteiligung dieses Programm noch vielseitiger machen.
 Der Assembler und das Hexlisting können als Rundlauf bezogen werden.

E1510

DATEIVERWALTUNG

Neue Datei anlegen	1
Alle Files ausgeben	2 ←
File suchen	3
File anhängen	4
File ändern	5
Formular erstellen	6
Formular ändern	7
Alle Formulare ausgeben	8
Zurück zu NASSYS	9
Progr.u.Datei auf Cass.	0

AUSGABEFORMAT:

Bildschirm	1
Ganze Files drucken	2 ←
Daten in best.Format	3

FILE No 01
 ALBERT DICK GUTMOTIG KAHL UND SCHWAMMIG
 FILE No 02
 BERTA DÜNN HITZIG LOCKIG
 FILE No 03
 CASAR HAGER RUHIG STRÄHNIG
 FILE No 04
 DORA MOLLIG LUSTIG KRAUS
 FILE No 05
 EMIL MAGER LÜSTERN RÖTLICH
 FILE No 06
 END OF DATA 2897 hex

FILE No 01 (Formulare)
 NS01 IST S02 UND EIN BISSCHEN S03.
 DAS S04E HAAR WIRKT ANZIEHEND.

FILE No 02
 NAME : S01
 STATUR : S02
 GEMÜT : S03
 HAAR : S04

FILE No 03
 GELIEBTE/R S01,
 DU BIST ZWAR S03, ABER DAFÜR SEHR S02, UND
 DAS MAG ICH AN DIR.
 MIT S04EN GRÜSSEN
 DEIN NASCOMPL

Suchbegriff ALBERT
 Formular "NAME" No2
 NAME : ALBERT
 STATUR : DICK
 GEMÜT : GUTMÜTIG
 HAAR : KAHL

Suchbegriff DICK/MOLLIG
 Formular No1
 ALBERT IST DICK UND EIN BISSCHEN GUTMOTIG.
 DAS KAHLE HAAR WIRKT ANZIEHEND.
 DORA IST MOLLIG UND EIN BISSCHEN LUSTIG.
 DAS KRAUSE HAAR WIRKT ANZIEHEND.

NUMMER des Formulars?
 Suchbegriffe eingeben !
 HAGER/LUSTIG GELIEBTE/R CASAR,
 DU BIST ZWAR RUHIG, ABER DAFÜR SEHR HAGER, UND
 DAS MAG ICH AN DIR.
 MIT STRÄHNIGEN GRÜSSEN
 DEIN NASCOMPL
 GELIEBTE/R DORA,
 DU BIST ZWAR LUSTIG, ABER DAFÜR SEHR MOLLIG, UND
 DAS MAG ICH AN DIR.
 MIT KRAUSEN GRÜSSEN
 DEIN NASCOMPL
 All Files Searched
 DATEIVERWALTUNG

VERKAUFE NASCOM 2 64K
 im Veroframe mit Netzteil
 Assembler, Disassembler, DEBUG, TOOLKIT
 und eine Menge Software
 2 mal 32K BOARDS (1 auf BASIC), Bufferb.

evt. auch einzeln
 Tel. [REDACTED] Schmidt

VERKAUFE oder TAUSCHE für NASCOM 2 & NASSYS
 Sargon-Schach mit Softwarepflege DM 50.-
 MpdK-Life (max.3 Generationen/sec) DM 25.-
 Haufenweise BASIC-Spiele ab 10.- u.a.
 (Liste gegen Freiumschlag). Bepieoste Cas-
 sette und Geld belegen. Cass.format ang.
 David Kastrup

VERKAUFE NASCOM 1 , 32K RAM
 erw. Tastatur, 8K Tape-BASIC
 Siemens Monitor, Fernschreiber mit Anst.
 zus. Speicherplatine 32K (noch nicht
 eingebaut), ZEAP, NASPEN u.a. Software
 VB DM 1300,- Tel. [REDACTED]

ZEAP Z80 Assembler - Source Listing

```

0010 ;DATENVERWALTUNG
0020 ;MASCH.UNTERPROGR.
0030 ;(C)G. BOHM, KARLSRUHE
0040 ;Ver.2.0/10.4.82
0050 BLINK EQU #7BDF
0060 CURSOR EQU #C29
0070 DATEI EQU #2000
0080 ASTEIL EQU DATEI+2;File Beginn
0090 ALEFIL EQU DATEI+4;File Länge
0100 DATEND EQU DATEI+6;Datei-Ende
0110 RAMEND EQU DATEI+8;Speicher-Ende
0120 AUSGAB EQU DATEI+11;Adr.d.Ausgaberroutine
0130 FORMPA EQU DATEI+13;Param.f."Formulare"
0140 FORNUM EQU DATEI+23;Nummer.d."Formulars"
0150 RETTBE EQU DATEI+24;"Rettungsuffer"Filepa
0160 FORBEG EQU DATEI+34;Beg.d.Formulardatei
0170 RAMBOT EQU DATEI+2048;Beginn der Filedatei
0180 RAMTOP EQU #5FFF
0190 BUFFER EQU #C80
0200 ENDZ EQU BUFFER+2
0210 BUFWD1 EQU BUFFER+4
0220 SUCHAN EQU BUFFER+3
0230 TOPLIN EQU #BCA
0240 ;
0250 ORG #1000
0260 EINGAB LD HL,#80A
0270 LD (CURSOR),HL
0280 LOOP1 DEFW BLINK
0290 FE1B CP #1B;ESC
0300 CP #D;FILE
0310 CP #D;OUT
0320 CP #D;OUT
0330 LD A,#9;Zeich. f. NL
0340 RST #30
0350 LD A,#D
0360 RST #30
0370 JR LOOP1
0380 LD A,#20
0390 RST #30
0400 LD A,7;BELL
0410 RST #30
0420 LD BC,#00
0430 LD HL,#809
0440 XOR A
0450 INC BC
0460 INC HL
0470 CP (HL)
0480 JR Z LOOP3 ;MARGINES
0490 LD A,9
0500 CP (HL)
0510 JR NZ BELL
0520 XOR A
0530 INC HL
0540 BE (HL)
0550 NZ NULL1
0560 LOOP2
0570 A,7
0580 (HL)
0590 NZ LOOP2
0600 ;Part File in Speicher?
0610 XOR A
0620 LD HL,(ALEFIL)
0630 SBC HL,BC
0640 JR Z LOAD
0650 JR NC REDUJ
0660 PUSH BC
0670 POP HL
0680 XOR A
0690 DE,(ALEFIL)
0700 SBC HL,DE;Diff.
0710 EX DE,HL
0720 LD HL,(DATEND)
0730 ADD HL,DE
0740 LD DE,(RAMEND)
0750 EX DE,HL
0760 SBC HL,DE
0770 JR NC SHIFT
0780 RST #28
0790 DEFM "FILE TOO LONG"
0800 DEFB 0
0810 DEFW BLINK
0820 CP #D;ENTER
0830 JR NZ AGAIN
0840 JR EINGAB
0850 ;Platz für File vergrößern
0860 PUSH BC
0870 LD DE,(ASTFIL)
0880 LD HL,(DATEND)
0890 XOR A
0900 SBC HL,DE
0910 PUSH BC
0920 PUSH BC
0930 POP HL
0940 LD DE,(ALEFIL)
0950 SBC HL,DE
0960 PUSH HL
0970 POP BC;Differenz
0980 LD HL,(DATEND)
0990 PUSH HL
1000 POP DE
1010 ADD HL,BC;neues Dateilende
1020 LD (DATEND),HL
1030 EX DE,HL
1040 POP BC
1050 LDDR
1060 POP BC
1031 BE
1032 20FC
1034 18EE
1036 3E07
1038 BE
1039 20E9
103B AF
103C 2A0420
103F ED42
1041 2854
1043 3076
1045 C5
1046 E1
1047 AF
1048 ED5B0420
104C ED52
104E EB
104F 2A0620
1052 19
1053 ED5B0820
1057 EB
1058 ED52
105A 3017
105C EF
105D 46494C45
20544F4F
204C4F4E
47
106A 00
106B DF7B
106D FE0D
106F 20FA
1071 188D
1073 C5
1074 ED5B0220
1078 2A0620
107B AF
107C ED52
107E E5
107F C5
1080 E1
1081 ED5B0420
1085 ED52
1087 E5
1088 C1
1089 2A0620
108C E5
108D D1
108E 09
108F 220620
1092 EB
1093 C1
1094 ED88
1096 C1
0800 AGAIN
0810 DEFB 0
0820 CP #D;ENTER
0830 JR NZ AGAIN
0840 JR EINGAB
0850 ;Platz für File vergrößern
0860 PUSH BC
0870 LD DE,(ASTFIL)
0880 LD HL,(DATEND)
0890 XOR A
0900 SBC HL,DE
0910 PUSH BC
0920 PUSH BC
0930 POP HL
0940 LD DE,(ALEFIL)
0950 SBC HL,DE
0960 PUSH HL
0970 POP BC;Differenz
0980 LD HL,(DATEND)
0990 PUSH HL
1000 POP DE
1010 ADD HL,BC;neues Dateilende
1020 LD (DATEND),HL
1030 EX DE,HL
1040 POP BC
1050 LDDR
1060 POP BC

```

```

1097 ED430420 1070 LOAD (ALEFIL),BC
1098 ED5B0220 1080 LD DE,(ASTFIL)
109F 210908 1090 LD HL,#809
10A2 AF 1100 XOR A
10A3 23 1110 INC HL
10A4 BE 1120 CP (HL)
10A5 28FB 1130 JR Z LOOP4
10A7 7E 1140 LD A,(HL)
10A8 12 1150 LD (DE),A
10A9 13 1160 INC DE
10AA 0B 1170 DEC BC
10AB FE09 1180 CP 9
10AD 2007 1190 JR NZ NEWL
10AF AF 1200 XOR A
10B0 25 1210 INC HL
10B1 BE 1220 CP (HL)
10B2 20FC 1230 JR NZ NULL2
10B4 18EC 1240 LD A,B
10B6 78 1250 NEWL
10B7 B1 1260 OR C
10B8 20E8 1270 JR NZ LOOP4
10BA C9 1280 RET

10BB C5 1290 ;REDUZ
10BC E5 1300 PUSH BC ;Screenlänge
10BD 2A0220 1310 PUSH HL ;Differenz
10C0 ED5B0420 1320 LD HL,(ASTFIL)
10C4 19 1330 LD DE,(ALEFIL)
10C5 E5 1340 ADD HL,DE
10C6 C1 1350 PUSH HL ;altes Fileende
10C7 ED5B0620 1360 POP BC ;nach BC
10CB EB 1370 LD DE,(DATEND)
10CC AF 1380 EX DE,HL
10CD ED52 1390 XOR A
10CE EB 1400 SBC HL,DE
10CF E1 1410 EX DE,HL ;Dateilänge in DE
10D0 E1 1420 POP HL
10D1 C5 1430 PUSH BC
10D2 E5 1440 PUSH HL
10D3 C1 1450 POP BC
10D4 2A0620 1460 LD HL,(DATEND)
10D7 ED42 1470 SBC HL,BC
10D9 220620 1480 LD HL,(DATEND),HL ;neues Dateieinde
10DC E1 1490 POP HL
10DD C1 1500 POP BC
10DE C5 1510 PUSH BC
10DF E5 1520 PUSH HL
10E0 2A0220 1530 LD HL,(ASTFIL)
10E3 09 1540 ADD HL,BC ;neues Fileende
10E4 D5 1550 PUSH DE
10E5 C1 1560 POP BC
10E6 D1 1570 POP DE
10E7 EB 1580 EX DE,HL
10E8 ED80 1590 LDIR
10EA C1 1600 POP BC
10EB 18AA 1610 JR LOAD
1620 ;File auf Screen darstellen

10ED 2A0220 1630 ;SCREEN
10F0 E5 1640 LD HL,(ASTFIL)
10F1 EF 1650 PUSH HL
10F2 0C00 1660 RST #28
10F4 2B 1670 DEFW #C ;CLRS
10F5 010000 1680 DEC HL
10F8 3E07 1690 LD BC,0
10FA 23 1700 LD A,7
10FB 03 1710 INC HL ;Fileende suchen
10FC BE 1720 INC BC
10FD 20FB 1730 CP (HL)
10FE ED430420 1740 JR NZ LOOP6
1103 110A08 1750 LD (ALEFIL),BC
1106 ED53290C 1760 DE,#80A
110A E1 1770 LD (CURSOR),DE
110B 7E 1780 POP HL
110C FE09 1790 LD A,(HL)
110E 2002 1800 CP 9
1110 3E0D 1810 JR NZ OUT1
1112 F7 1820 LD A,#D
1113 23 1830 RST #30 ;ROUT
1114 0B 1840 TNC HL
1115 78 1850 DEC BC
1116 B1 1860 LD A,B
1117 20F2 1870 OR C
1119 11CA0B 1880 JR NZ LOOP7
111C ED53290C 1890 ;In HL = Start next File
1120 EF 1900 LD DE,TOPLIN
1121 46494C45 1910 LD (CURSOR),DE
1129 00 1920 RST #28
112A 08 1930 DEFW "FILE No "
112B F5 1940 DEF B 0
112C DF68 1950 EX AF,AF'
112E F1 1960 PUSH AF,AF'
112F 08 1970 DEFW #68DF ;B2HEX - PRINT A
1130 EF 1980 POP AF,AF'
1131 20202020 1990 EX AF,AF'
1131 436F6E74 2000 RST #28
1131 696E7565 2010 DEFW " Continue or ESCAPE"
1131 206F7220 2020 DEF M
1131 45534341 2030 POP AF,AF'
1131 5045 2040 EX AF,AF'
1147 00 2050 DEF B 0
1148 E5 2060 PUSH HL
1149 DF7B 2070 DEFW BLINK
114B E1 2080 POP HL
114C C9 2090 RET

114D 212220 2090 ;Hauptprogramme
1150 220D20 2100 ;DATEI INITIIEREN
2110 INIT LD HL,FORMREG ;FORMULARE
2120 LD (FORMPA),HL

```

```

1153 220F20 LD (FORMPA+2),HL
1156 221320 LD (FORMPA+6),HL
115A 210100 LD HL,1
115D 221120 LD (FORMPA+4),HL
1160 21FE27 LD HL,RAMBOT-1
1163 221520 LD (FORMPA+8),HL
;
1166 210028 LD HL,RAMBOT ;FILES
1169 220020 LD (DATEI),HL
116C 220220 LD (ASTFIL),HL
116F 110100 LD DE,1 ;min,File Länge
1172 ED530420 LD (ALEFIL),DE
1176 23 INC HL ;Dateiende=Fileende+1
1177 220620 LD (DATEI),HL
117A 21FF5F LD HL,RAMTOP
117D 220820 LD (RAMEND),HL
1180 3EC3 LD A,#C3
1182 320A20 LD (AUSGAB-1),A
1185 21ED10 LD HL,SCREEN
1188 220B20 LD (AUSGAB),HL
118B EF RST #28
118C 0C00 DEFW #C ;CLRS
118E 21CA0B LD HL,TOPLIN
1191 22290C LD (CURSOR),HL
1194 EF RST #28
1195 44415441 DEFM "DATA INPUT"
20494E50
5554
119F 00 DEFB 0
11A0 C00010 CALL ETINGAB
11A3 EF RST #28
11A4 0C00 DEFW #C ;CLRS
11A6 21CA0B LD HL,TOPLIN
11A9 22290C LD (CURSOR),HL
11AC EF RST #28
11AD 4E657874 DEFM "Next File - or ESCAPE ?"
2046696C
65202D20
6F722045
53434150
45203F
11C4 00 DEFB 0
11C5 DF7B DEFW 8LINK
11C7 FE1B CP #1B
11C9 2001 JR NZ NOESC
11CB C9 2520 RET
11CC 2A0420 LD HL (ALEFIL) ;Länge
11CF ED5B0220 LD DE,(ASTFIL)
11D3 19 2550 ADD HL,DE
11D4 220220 LD (ASTFIL),HL ;next File start
11D7 23 2570 INC HL
11D8 220620 LD HL,1
11DB 210100 LD (ALEFIL),HL
11DE 220420 LD HL,18A8
11E1 18A8 2620 JR LOOP5
;
2130 LD (FORMPA+2),HL
2140 LD (FORMPA+6),HL
2150 HL,1
2160 LD (FORMPA+4),HL
2170 HL,RAMBOT-1
2180 LD (FORMPA+8),HL
;
2200 LD HL,RAMBOT ;FILES
2210 LD (DATEI),HL
2220 LD (ASTFIL),HL
2230 LD DE,1 ;min,File Länge
2240 LD (ALEFIL),DE
2250 INC HL ;Dateiende=Fileende+1
2260 LD (DATEI),HL
2270 LD HL,RAMTOP
2280 LD (RAMEND),HL
2290 LD A,#C3
2300 LD (AUSGAB-1),A
2310 LD HL,SCREEN
2320 LD (AUSGAB),HL
2330 RST #28
2340 DEFW #C ;CLRS
2350 LD HL,TOPLIN
2360 LD (CURSOR),HL
2370 RST #28
2380 DEFM "DATA INPUT"
2390 44415441
20494E50
5554
2400 DEFB 0
2410 CALL ETINGAB
2420 RST #28
2430 DEFW #C ;CLRS
2440 LD HL,TOPLIN
2450 LD (CURSOR),HL
2460 RST #28
2470 DEFM "Next File - or ESCAPE ?"
2046696C
65202D20
6F722045
53434150
45203F
2480 DEFB 0
2490 DEFW 8LINK
2500 CP #1B
2510 JR NZ NOESC
2520 RET
2530 LD HL (ALEFIL) ;Länge
2540 LD DE,(ASTFIL)
2550 ADD HL,DE
2560 LD (ASTFIL),HL ;next File start
2570 INC HL
2580 LD HL,1
2590 LD (ALEFIL),HL
2600 LD HL,18A8
2610 18A8
2620 JR LOOP5
;
2630 ;Unterprf. "Ist nächstes File vorhanden?"
2640 LD DE,(DATEI) ;Datei Ende?
2650 DEC DE
2660 XOR A
2670 PUSH HL
2680 SBC HL,DE
2690 POP HL
2700 RET C
2710 RST #28
2720 DEFW #C
2730 LD DE,TOPLIN
2740 LD (CURSOR),DE
2750 RST #28
2760 DEFM "END OF DATA "
2770
2780 DEFB 0
2790 DEFW #66DF ;TBCD3 - PRINT HL (next Adr.)
2800 RST #28
2810 DEFM "hex"
2820 DEFB 0
2830 POP HL ;Throw Away Return Address
2840 RET
2850 ;
2860 ;Hauptprogr. ALLE FILES AUSGEBEN.
2870 ALLFLS LD HL,(DATEI)
2880 LD (ASTFIL),HL
2890 LD A,1 ;COUNTER
2900 EX AF,AF
2910 CALL AUSGAB-1.
2920 CP #1B
2930 JR NZ 600N
2940 RET
2950 LD (ASTFIL),HL ;Start next File
2960 LD (ASTFIL),HL ;Start next File
2970 CALL ISTDND
2980 EX AF,AF ;INC COUNTER
2990 INC A
3000 DAA
3010 EX AF,AF
3020 JR SCR
3030 ;Unterprogr. "Suchen eines Files"
3040 SEARCH LD A,(SUCHAN)
3050 LD B,A ;Anzahl der Suchbegriffe
3060 LD HL,BUFWDI
3070 LD HL,(BUFFER),HL
3080 LD HL,(ASTFIL)
3090 FILBEG LD DE,(BUFFER)
3100 BUFBEGLD A,(HL)
3110 ISBELL LD HL
3120 LD HL,ENDZ
3130 OP (HL)
3140 POP HL
3150 JR Z NEXTWD ;Endzeichen des Files erreicht
3160 LD A,(DE)
3170
3180
11E3 ED5B0620
11E7 TB
11E8 AF
11E9 E5
11EA ED52
11EC E1
11ED D8
11EE EF
11EF 0C00
11F1 11CA0B
11F4 ED53290C
11F8 EF
11F9 454E4420
4F462044
41544120
1205 00
1206 DF66
1208 EF
1209 68578
120C 00
120D E1
120E C9
120F 2A0020
1212 220220
1215 3E01
1217 08
1218 CD0A20
121B FE1B
121D 2001
121F C9
1220 220220
1223 CDE311
1226 08
1227 3C
1228 27
1229 08
122A 18EC
122C 3A830C
122F 47
1230 21840C
1233 22800C
1236 2A0220
1239 ED5B800C
123D 7E
123E E5
123F 21820C
1242 8E
1243 E1
1244 2830
1246 1A

```

1247 BE	3190	(HL)	CP	INC	HL	ISBELL	2A0620	3690	LD	HL,(DATEND)
1248 23	3200	HL	JR	NZ	ISBELL	E5	3700	PUSH	LD	HL,(RAMEND)
1249 20F2	3210	HL	JR	ISBELL	HL	2AD	3710	PUSH	LD	HL,(RAMEND)
124B E5	3220	HL	JR	ISBELL	HL	E5	3720	PUSH	LD	HL,(RAMEND)
124C 13	3230	HL	JR	ISBELL	HL	2B0	3730	LD	HL, BUFWD1 ; Beginn des Puffers	
124D 1A	3240	HL	JR	ISBELL	HL	2B4	3740	LD	HL, (ASTFIL), HL	
124E FE20	3250	HL	JR	ISBELL	HL	2B7	3750	LD	HL, (ASTFIL), HL	
1250 280B	3260	HL	JR	ISBELL	HL	2B8	3760	LD	HL, (DATEND), HL	
1252 FE2F	3270	HL	JR	ISBELL	HL	2B8	3770	LD	HL, 1	
1254 2807	3280	HL	JR	ISBELL	HL	2B8	3780	LD	HL, (CALEFIL), HL	
1256 BE	3290	HL	JR	ISBELL	HL	2BE	3790	LD	HL, #F00	
1257 23	3300	HL	JR	ISBELL	HL	2C1	3800	LD	HL, (RAMEND), HL	
1258 28F2	3310	HL	JR	ISBELL	HL	2C4	3810	LD	HL, (RAMEND), HL	
125B 18DC	3320	HL	JR	ISBELL	HL	2C7	3820	LD	HL, (RAMEND), HL	
125D F1	3330	HL	JR	ISBELL	HL	2CA	3830	LD	HL, (RAMEND), HL	
125E 05	3340	HL	JR	ISBELL	HL	2CC	3840	LD	HL, (RAMEND), HL	
125F C8	3350	HL	JR	ISBELL	HL	2CF	3850	LD	HL, (RAMEND), HL	
1260 1A	3360	HL	JR	ISBELL	HL	2D1	3860	LD	HL, (RAMEND), HL	
1261 FE2F	3370	HL	JR	ISBELL	HL	2D1	3870	LD	HL, (RAMEND), HL	
1263 2006	3380	HL	JR	ISBELL	HL	2D4	3880	LD	HL, (RAMEND), HL	
1265 13	3390	HL	JR	ISBELL	HL	2D5	3890	LD	HL, (RAMEND), HL	
1266 1A	3400	HL	JR	ISBELL	HL	2D6	3900	LD	HL, (RAMEND), HL	
1267 FE20	3410	HL	JR	ISBELL	HL	2D8	3910	LD	HL, (RAMEND), HL	
1269 20FA	3420	HL	JR	ISBELL	HL	2D8	3920	LD	HL, (RAMEND), HL	
126B 13	3430	HL	JR	ISBELL	HL	2D8	3930	LD	HL, (RAMEND), HL	
126C 1A	3440	HL	JR	ISBELL	HL	2D8	3940	LD	HL, (RAMEND), HL	
126D FE07	3450	HL	JR	ISBELL	HL	2D8	3950	LD	HL, (RAMEND), HL	
126F C8	3460	HL	JR	ISBELL	HL	2D8	3960	LD	HL, (RAMEND), HL	
1270 ED53800C	3470	HL	JR	ISBELL	HL	2D8	3970	LD	HL, (RAMEND), HL	
1274 1800	3480	HL	JR	ISBELL	HL	2D8	3980	LD	HL, (RAMEND), HL	
1276 13	3490	HL	JR	ISBELL	HL	2D8	3990	LD	HL, (RAMEND), HL	
1277 1A	3500	HL	JR	ISBELL	HL	2D8	4000	LD	HL, (RAMEND), HL	
1278 FE20	3510	HL	JR	ISBELL	HL	2D8	4010	LD	HL, (RAMEND), HL	
127A C8	3520	HL	JR	ISBELL	HL	2D8	4020	LD	HL, (RAMEND), HL	
127B FE2F	3530	HL	JR	ISBELL	HL	2D8	4030	LD	HL, (RAMEND), HL	
127D 20F7	3540	HL	JR	ISBELL	HL	2D8	4040	LD	HL, (RAMEND), HL	
127F 13	3550	HL	JR	ISBELL	HL	2D8	4050	LD	HL, (RAMEND), HL	
1280 ED53800C	3560	HL	JR	ISBELL	HL	2D8	4060	LD	HL, (RAMEND), HL	
1284 18B0	3570	HL	JR	ISBELL	HL	2D8	4070	LD	HL, (RAMEND), HL	
	3580	HL	JR	ISBELL	HL	2D8	4080	LD	HL, (RAMEND), HL	
	3590	HL	JR	ISBELL	HL	2D8	4090	LD	HL, (RAMEND), HL	
	3600	HL	JR	ISBELL	HL	2D8	4100	LD	HL, (RAMEND), HL	
1286 EF	3610	HL	JR	ISBELL	HL	2D8	4110	LD	HL, (RAMEND), HL	
1287 0000	3620	HL	JR	ISBELL	HL	2D8	4120	LD	HL, (RAMEND), HL	
1289 21CA0B	3630	HL	JR	ISBELL	HL	2D8	4130	LD	HL, (RAMEND), HL	
128C 22290C	3640	HL	JR	ISBELL	HL	2D8	4140	LD	HL, (RAMEND), HL	
128F EF	3650	HL	JR	ISBELL	HL	2D8	4150	LD	HL, (RAMEND), HL	
1290	3660	HL	JR	ISBELL	HL	2D8	4160	LD	HL, (RAMEND), HL	
	3670	HL	JR	ISBELL	HL	2D8	4170	LD	HL, (RAMEND), HL	
	62656772	HL	JR	ISBELL	HL	2D8	4180	LD	HL, (RAMEND), HL	
	69666665	HL	JR	ISBELL	HL	2D8	4190	LD	HL, (RAMEND), HL	
	2065696E	HL	JR	ISBELL	HL	2D8	4200	LD	HL, (RAMEND), HL	
	67656265	HL	JR	ISBELL	HL	2D8	4210	LD	HL, (RAMEND), HL	
	6E202120	HL	JR	ISBELL	HL	2D8	4220	LD	HL, (RAMEND), HL	
12A8 00	3680	HL	JR	ISBELL	HL	2D8	4230	LD	HL, (RAMEND), HL	
		HL	JR	ISBELL	HL	2D8	4240	LD	HL, (RAMEND), HL	

```

1313 ED580620 4250
1317 1B 4260
1318 AF 4270
1319 ED52 4280
131B 38DB 4290
131D EF 4300
131E 0C00 4310
1320 21CA0B 4320
1323 22290C 4330
1326 EF 4340
1327 41606C20 4350
46696055
73205365
61726368
6564
1339 00 4360
133A C9 4370
4380
4390 ; Unterprog. GANZE FILES AUSDRUCKEN
4400 PRTALL LD A,#A ;KSR NL
4410 CALL PRINT
4420 LD HL,FNUMB ;Stringadresse
4430 LD B,9
4440 DEFW #6DDF ;SOUT
4450 EX AF,AF' ;Counter
1347 08 4460
1348 47 4470
1349 08 4480
134A 78 4490
134B CB2F 4500
134D CB2F 4510
134F CB2F 4520
1351 CB2F 4530
1353 C830 4540
1355 CD1617 4550
1358 78 4560
1359 E60F 4570
135B C830 4580
135D CD1617 4590
1360 3E0A 4600
1362 CD1617 4610
4620
4630 L00P12
4640
4650
4660
4670
4680
4690
4700 DRUCK1
4710 CALL LOOP12
4720 ;
4730 ;
4740 ; Unterprog. X-TES WORT DRUCKEN
4750 ; Wortnummer vor Aufg. in B
4760 PRINBR LD HL,(ASTFIL)
4770 LOOP14 LD A,B
4780 DEC A
1313 137E 27
1317 137F 47
1318 2819 2819
1319 23 23
1320 7E 7E
1321 FE07 FE07
1322 2008 2008
1323 216703 216703
1324 0605 0605
1325 DF6D DF6D
1326 C9 C9
1327 FE20 FE20
1328 2804 2804
1329 FED9 FED9
1330 20EA 20EA
1331 23 23
1332 18E1 18E1
1333 7E 7E
1334 FE09 FE09
1335 2002 2002
1336 3E0A 3E0A
1337 23 23
1338 FE0A FE0A
1339 C8 C8
1340 FE20 FE20
1341 C8 C8
1342 FE20 FE20
1343 C8 C8
1344 C8 C8
1345 FE20 FE20
1346 C8 C8
1347 C8 C8
1348 CD1617 CD1617
1349 18ED 18ED
5070 ;
5080 ;
5090 ; Unterprog. Formatierte Druckausgabe
5100 ; Muster (Nr d. Formulars) in FORNUM
5110 ; ruft PRINBR auf
5120 FORM LD HL,(FORMPA);Beginn des Formulars
5130 LD A,(FORNUM)
5140 B,A
5150 LOOP16 LD DE,(FORMPA+6)
5160 DEC DE
5170 XOR A
5180 PUSH HL,DE
5190 SBC HL,DE
5200 POP HL
5210 JR NC NOFORM
5220 DEC B
5230 Z L00P15
5240 LD A,(HL)
5250 LOOP17 LD HL
5260 INC HL
5270 CP Z ;BELL
5280 JR Z L00P16
5290 RST #28 L00P17
5300 NOFORM DEFM "Form No. not existing"
466F726D
204E6F2E
206E6F74
2067869
7374696E
67
1378 2A0220
137B 78
137C 3D
4790 DAA
4800 LD B,A
4810 JR Z DRUCK2
4820 LOOP13 INC HL
4830 LD A,(HL)
4840 CP Z ;BELL
4850 JR NZ NOEND
4860 LD HL,#367 ;ERRM
4870 B,5
4880 DEFW #6DDF ;SOUT
4890 RET
4900 CP " ;wortlende?
4910 JR Z WORTEN
4920 CP 9
4930 NZ LOOP13
4940 INC HL
4950 LD A,(HL)
4960 DRUCK2 CP 9
4970 NZ DRUCK3
4980 LD A,#A ;NL KSR
4990 INC HL
5000 CP #A
5010 RET Z
5020 CP Z
5030 RET Z
5040 CALL PRINT
5050 DRUCK2 JR
5060 ;
5070 ;
5080 ;
5090 ; Unterprog. Formatierte Druckausgabe
5100 ; Muster (Nr d. Formulars) in FORNUM
5110 ; ruft PRINBR auf
5120 FORM LD HL,(FORMPA);Beginn des Formulars
5130 LD A,(FORNUM)
5140 B,A
5150 LOOP16 LD DE,(FORMPA+6)
5160 DEC DE
5170 XOR A
5180 PUSH HL,DE
5190 SBC HL,DE
5200 POP HL
5210 JR NC NOFORM
5220 DEC B
5230 Z L00P15
5240 LD A,(HL)
5250 LOOP17 LD HL
5260 INC HL
5270 CP Z ;BELL
5280 JR Z L00P16
5290 RST #28 L00P17
5300 NOFORM DEFM "Form No. not existing"
466F726D
204E6F2E
206E6F74
2067869
7374696E
67

```


13E1 00	DEFB 0	HL ;Rücksprungadr. , wegwerfen	1449 00	5760	DEFB #D	2"
13E2 E1	POP	START	144A 47616E7A	5770	DEFM "Ganze Files drucken	
13E3 C31015	JP		65204669			3"
5340	LD	A, (HL)	60657320			
5350	CP	Z ;BELL	64727563			
5360	JR	Z SEAEAD	68656E20			
5370	CP	#40 ;Kennung für Wortnummer	20202020			
5380	JR	Z NUMMER	2032			
5390	CP	9 ;NL	00	5780	DEFB #D	
5400	JR	NZ DRUCK4	1464 44617465	5790	DEFM "Daten in best.Format	
5410	LD	A, #A ;KSR	1465 6E20696E			
5420	JR	PRINT	20626573			
5430	CALL	HL	742E466F			
5440	INC	HL	726D6174			
5450	JR	LOOP15	20202020			
5460	INC	HL	2033			
5470	LD	B, (HL)	147F 0D00	5800	DEFB #D	
5480	SLA	B	1481 DF7B	5810	DEFW BLINK	
5490	SLA	B	1483 FE31	5820	CP #31	
5500	SLA	B	1485 2005	5830	JR NZ W02	
5510	SLA	B	1487 21ED10	5840	LD HL, SCREEN	
5520	INC	HL	1488 1830	5850	JR LADEN	
5530	LD	A, (HL)	148C FE32	5860	CP #32	
5540	AND	#F	148E 2005	5870	JR NZ W03	
5550	ADD	A, B	1490 213B13	5880	LD HL, PRTALL	
5560	LD	B, A	1493 1827	5890	JR LADEN	
5570	INC	HL ;nächstes Wort	1495 FE33	5900	CP #33	
5580	PUSH	HL	1497 20E8	5910	JR NZ WAHL	
5590	CALL	PRTNBR	1499 EF	5920	RST #28	
5600	POP	HL	149A 0C	5930	DEFB #C	
5610	JR	LOOP15	149B 4E554D4D	5940	DEFM "NUMMER des Formulars?"	
5620	SEAEND	LD HL, (ASTFIL)	45522064			
5630	LOOP18	LD A, (HL)	65732046			
5640	CP	7 ;BELL	6F726D75			
5650	INC	HL	60617273			
5660	JR	NZ LOOP18	3F			
5670	RET					
5680 ;						
5690 ;						
5700 ;Hauptprogr. , AUSGABEFORMAT ANDERN						
5710	CHAFOR	RST #28	1480 0D00	5950	DEFB #D	
5720	DEFB #C		1482 DF7B	5960	DEFW BLINK	
5730	DEFM "AUSGABEFORMAT:"		1484 E60F	5970	AND #F ;nur 1 bis 9 zugelassen	
			1486 321720	5980	LD (FORNUM), A	
141D EF				5990 ;		
141E 0C				6000	LD HL, FORM	
141F 41555347				6010	LD (AUSGAR), HL	
41424546				6020	RET	
4F524D41				6030 ;		
543A				6040 ;		
142D 0D0D				6050 ;Hauptprogr. , NEUE FILES ANHANGEN		
142F 42696C64				6060	EXTEND LD HL, (DATEND)	
73636869				6070	LD (ASTFIL), HL	
726D2020				6080	JP INPUT	
20202020				6090 ;		
20202020				6100 ;		
20202020				6110 ;Hauptprogr. , FILES ANDERN		
2031				6120	CHANGE LD HL, AENDER	
				6130	LD (AUSGAR), HL	
				6140	JP SUCHEN	
				6150 ;folgendes UP wird nun von SUCHEN		
				6160 ;als Ausgabe aufgerufen		

14D2	CDIED10	6170	AENDER	CALL SCREEN	1574	46696C65	6450	DEFM "File anhängen	4"
14D5	E5	6180		PUSH HL,Start next File		20615F68			
14D6	FE1B	6190		OP #1B ;ESCAPE?		7B6E6765			
14D8	282E	6200		JR Z,NORM		6E202020			
14DA	21CA0B	6210		LD HL,TOPLIN		20202020			
14DD	22290C	6220		LD (CURSOR),HL		20202020			
14E0	EF	6230		RST #28		2034			
14E1	4348414E	6240		DEFM "CHANGE ! Put CURSOR to End of File!		158E 0D	6460	DEFB #D	5"
	47452021					158F	6470	DEFM "File ändern	
	20507574					46696C65			
	20435552					207B6E64			
	534F5220					65726E20			
	746F2045					20202020			
	6E64206F					20202020			
	66204669					20202020			
	6C6521					2035			
1504	00	6250		DEFB 0		15A9 0D	6480	DEFB #D	6"
1505	CD0010	6260		CALL EINGAB		15AA	6490	DEFM "Formular erstellen	
1508	21ED10	6270	NORM	LD HL,SCREEN ;Ausgabe wieder normal		466F726D			
150B	220B20	6280		LD (AUSGAB),HL		756C6172			
150E	E1	6290		POP HL		20657273			
150F	C9	6300		RET ;ZU MENUE		74656C6C			
		6310				656E2020			
		6320		;MENUE-Programm DATEIVERWALTUNG		20202020			
		6330		ENT		2036			
1510	EF	6340	START	RST #28		15C4 0D	6500	DEFB #D	7"
1511	0D	6350		DEFB #D		15C5	6510	DEFM "Formular ändern	
1512	44415445	6360		DEFM "DATEIVERWALTUNG"		466F726D			
	49564552					756C6172			
	57414C54					207B6E64			
	554E47					65726E20			
1521	0D	6370		DEFB #D		20202020			
1522	00	6380		DEFB #D		2037			
1523	4E657565	6390		DEFM "Neue Datei anlegen	1"	15DF 0D	6520	DEFB #D	8"
	20446174					15E0	6530	DEFM "Alle Formulare ausgeben	
	65692061					20466F72			
	6E606567					6D756C61			
	656E2020					72652061			
	20202020					75736765			
	2031					62656E20			
153D	0D	6400		DEFB #D		2038			
153E	416C6C65	6410		DEFM "Alle Files ausgeben	2"	15FA 0D	6540	DEFB #D	9"
	2046696C					15FB	6550	DEFM "Zurück zu MASSYS	
	65732061					5A75727D			
	75736765					636B207A			
	62656E20					75204E41			
	20202020					53535953			
	2032					20202020			
1558	0D	6420		DEFB #D		2039			
1559	46696C65	6430		DEFM "File suchen	3"	1615 0D	6560	DEFB #D	
	20737563					1616	6570	DEFM "Progr.,u.Datei auf Cass. 0"	
	68656E20					50726F67			
	20202020					722E752E			
	20202020					44617465			
	20202020					69206175			
	20202020					66204361			
	2033					75732E20			
1573	0D	6440		DEFB #D		2030			
						1630 0000	6580	DEFM #D	

```

1632 DEFZ 6590 DEFW BLINK
1633 FE31 6600 CP #1
1634 6610 JR NZ Z2
1635 6620 CALL INIT
1636 CD4011 6630 START
1637 C31015 6640 JP #2
1638 6650 JR NZ Z3
1639 CD1D14 6660 CALL CHAFOR
1640 2009 6670 CALL ALLEFLS
1641 CD0F12 6680 START
1642 C31015 6690 JP #3
1643 FE33 6700 JR NZ Z4
1644 2009 6710 CALL CHAFOR
1645 CD1D14 6720 CALL SUCHEN
1646 6730 JP #4
1647 6740 CP #4
1648 6750 JR NZ Z5
1649 2006 6760 CALL EXTEND
1650 CD0C14 6770 JP #5
1651 C31015 6780 JR NZ Z6
1652 FE35 6790 CALL CHANGE
1653 2006 6800 JP #6
1654 CD0C14 6810 JR NZ Z7
1655 C31015 6820 CALL RETTEN
1656 FE36 6830 CALL EXTEND
1657 200C 6840 CALL NORMAL
1658 CD0E16 6850 JP #7
1659 CDF16 6860 JR NZ Z8
1660 C31015 6870 CALL RETTEN
1661 FE37 6880 CALL CHANGE
1662 200C 6890 CALL NORMAL
1663 CD0E16 6900 JP #8
1664 CD0C14 6910 JR NZ Z9
1665 CDF16 6920 CALL CHAFOR
1666 C31015 6930 CALL RETTEN
1667 FE38 6940 CALL ALLEFLS
1668 200F 6950 CALL NORMAL
1669 CD1D14 6960 JP #9
1670 C2A616 6970 JR NZ Z0
1671 DF58 6980 DEFW #5BDF ;RET NASSYS
1672 FE30 6990 CP #0 ;Speicher auf Cassette
1673 21015 7000 JP NZ START
1674 21000 7010 LD HL,#1000 ;Programmbeginn
1675 220C0C 7020 LD (#C0C),HL ;ARG1
1676 2A0620 7030 LD HL,(DATEND)
1677 220E0C 7040 LD (#C0E),HL ;ARG2
1678 DF66 7050 DEFW #66DF ;PRINT HL
1679 EF 7060 RST #28
1680 44617465 7070 DEFM "Dateiende"
1681 69656E64 7080
1682 65 7090
1683 0D 7100
1684 7130 DEFB #D

```

```

16C4 41627370 7140 DEFM "Abspeichern mit ENTER"
65696368
65726E20
6D697420
454E5445
52
16D9 0D00 DEFW #D
16DB DE7B DEFW BLINK
16DD FE0B CP #D
16DF 2802 JR Z ABSP
16E1 DE58 DEFW #5BDF ;NASSYS
16E3 DF57 DEFW #57DF ;WRITE
16E5 C31015 JP START
ABSP
7150
7160
7170
7180
7190
7200
7210
7220 ;
7230 ;
7240 ;Unterprogf.,RETTEN DER FILEPARAMETER
7250 RETTEN LD HL,DATEI
7260 LD DE,RETTBF
7270 LD BC,10
7280 LDIR
7290 LD HL,FORMPA
7300 LD DE,DATEI
7310 LD BC,10
7320 LDIR
7330 RET
7340 ;
7350 ;
7360 ;Unterprogf.,RESTORE FILEPARAMETER
7370 NORMAL LD HL,DATEI
7380 LD DE,FORMPA
7390 LD BC,10
7400 LDIR
7410 LD HL,RETTBF
7420 LD DE,DATEI
7430 LD BC,10
7440 LDIR
7450 RET
7460 ;PRINT
7470 PUSH AF
7480 DEFW #6FDF ;SRLX
7490 POP AF
7500 CP #A
7510 RET NZ
7520 LD B,#30 ;DELAY
7530 JP #40 ;Upro.in Nassys3 mit Ret

```

RAM-GRAPHIK

von Johannes C. Lotter

Viele Beiträege im Nascom-Journal befassten sich in letzter Zeit mit der Erzeugung von Graphik, und ich will hier keine Ausnahme machen. Normalerweise wird die NASCOM-Graphik ueber einen zweiten Zeichensatz erzeugt, der in einem 2k-Eprom gespeichert ist. Umgeschaltet zwischen erstem und zweitem Zeichensatz wird ueber Bit 7 des

Video-RAM. So weit, so gut; bald wird man sich jedoch wuenschen, man haette mehr Zeichen zur Verfuegung als die 128 zusaetzlichen. Die BASIC - Blockgraphik beansprucht 64 Zeichen, die Schach Graphik 64 weitere, In-

dizes und mathematische Sonderzeichen nochmal 64, und ein griechisches Alphabet waere auch nicht schlecht. Die beste Loesung ist ein Zeichensatz, der sich beliebig softwaremaessig definieren laesst.

laesst. Der Zeichengenerator muss also ein RAM sein, auf das von der CPU zugegriffen werden kann. Es erweist sich als sinnvoll, auch ein Lesen aus dem Grafik-RAM zuzulassen.

Die hier vorgestellte Schaltung erweitert den NASCOM-Zeichengenerator von 2k auf 4k. Der erste Teil ist der originale NASCOM 1 Zeichengenerator, der zweite Teil ein 2k RAM. Als ich die Schaltung vor 2 Jahren aufbaute, nahm ich vier 2114'er; heute waere vermutlich ein 6116'er guenstiger. Der Zeichengenerator mit nachgeschaltetem Buffer laesst sich auch durch ein entsprechend programmiertes Eprom ersetzen. Dann kaeme die Schaltung mit 7 IC's aus.

Die ganze Logik ist auf einer Lochrasterplatte auf der RAM-Karte aufgebaut. Dort

ist genuegend Platz direkt am Busstecker, wo auch die Adressen- und Datenleitungen sehr schoen zugaenglich sind.

Die Verbindung zur NASCOM-Grundplatte erfolgt ueber ein Kabel mit 24-poligem Stecker, der in die Fassung des Zeichengenerators eingesteckt wird. Zusaetzlich muss noch eine Verbindung zwischen folgenden IC's hergestellt werden:

IC 20 Pin 12 - IC 17 Pin 18

IC 17 Pin 19 - IC 16 Pin 10

IC 16 Pin 14 - IC 15 Pin 6; von Masse abtrennen!

Mit der gewaehlten Dekodierung liegt der Grafik-RAM-Bereich von 9000 - A000. Die folgenden Programme sind fuer diese Dekodierung ausgelegt.

Wenn die Schaltung eingebaut ist, kann man

mit der Grafik-Taste (Bit 7 gesetzt) die 128 frei definierbaren Zeichen auf den Schirm bringen. Zunaechst wird man nur wirres Gekringel sehen, da noch nichts vernuenftiges im RAM steht. Sie koennen nun Ihre Grafik testen, indem Sie eingeben:

I 9000 9800 800

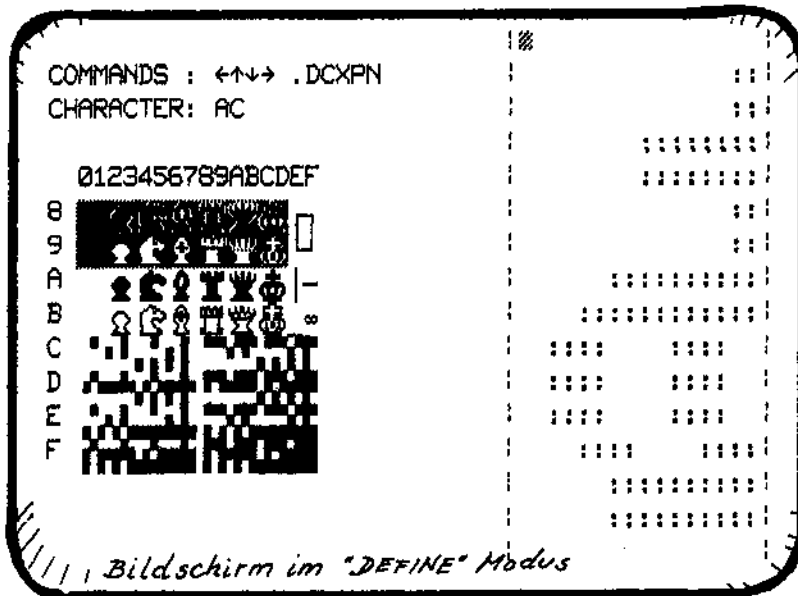
Dies kopiert den normalen Zeichensatz ins RAM. Folgendes kleine Programm erzeugt invertierte Zeichen:

21 00 90 11 00 98 01 00

08 7E 2F 12 03 13 23 78

B1 20 F6 E7

Anschliessend noch zwei weitere Programme. BSET erzeugt die BASIC-Blockgraphik und ist auf jeder Adresse lauffaehig. Mit DEFINE koennen Sie Ihren eigenen Zeichensatz definieren. Nach dem Start auf Adresse D00 wird der Bildschirm aufgeteilt (siehe Bildschirmausdruck); links steht der koeplette Zeichensatz, rechts koennen Sie einzelne Punkte in einem Zeichen setzen oder loeschen. Sie haben folgende Befehle zur Ver-



fuegung:

Space loescht einen Punkt

"," setzt einen Punkt

"X" wechselt den Cursor zwischen linkem und rechtem Feld

"C" kopiert ein Zeichen vergroessert ins rechte Feld

"D" bestimmt das zu definierende Zeichen

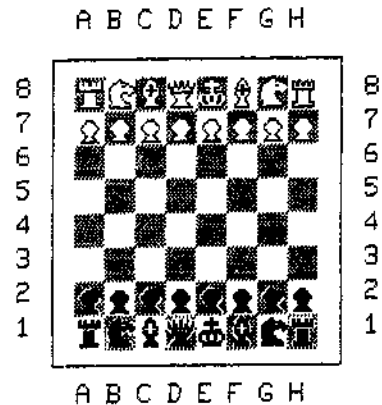
"N" springt zurueck nach NAS-SYS

Mit den Pfeiltasten laesst sich der Cursor bewegen. Sie wirken ebenso wie der "X"-Befehl in beiden Feldern. "D" und "C" wirken nur im linken, "." und Space nur im rechten Feld.

Der punktgetreue Bildschirmausdruck wurde uebrigens mit dem Seiko-Drucker erstellt. Er waere ohne lesbaren Zeichengenerator nicht moeglich. Der hier dargestellte Zeichensatz besteht aus der Schach- und der Blockgrafik.

```

0E70 58 88 0E 00 21 4A 0B 22 04
0E78 29 0C CD E8 0D C3 00 A0 E0
0E80 21 4A 0B 22 29 0C DF 58 95
0E88 2A 29 0C ED 5B 1A 0C E9 4E
0E90 22 29 0C ED 53 1A 0C C9 24
0E98 2A 29 0C 7E 32 1C 0C 26 03
0EA0 09 6F 29 29 29 C9 CD 60
0EA8 CE 0D 0D D7 E8 EB CD A9 84
0EB0 0D C3 0D 0E A5 A5 A5 A5 3D
  
```



01.Zug : Weiss zieht von -- nach --

```

DEFINE
TD00 E88
0D00 3E 80 CD 9C 0E 22 1E 0C 8E
0D08 21 E9 0B 22 1A 0C CD 43 82
0D10 0D 21 0C 09 22 29 0C DF 96
0D18 7B 21 52 0E 4F 7E B7 20 CD
0D20 F6 89 28 05 23 23 23 18 8A
0D28 F4 07 05 CD E8 0D 18 E7 C6
0D30 23 7E 23 66 6F E9 CD CE 5A
0D38 0D 00 CD 98 0E 22 1E 0C E1
0D40 CD 88 0E EF 0C 43 4F 4D 8A
0D48 4D 41 4E 44 53 20 3A 20 42
0D50 00 2A 29 0C 36 0D 23 22 44
0D58 29 0C EF 5E 08 09 20 2E 49
0D60 44 43 58 50 4E 0D 43 48 82
0D68 41 52 41 43 54 45 52 3A B1
0D70 20 00 3A 1C 0C DF 68 EF 35
0D78 0D 0D 20 20 00 0E 08 16 08
0D80 80 06 10 1E 00 7B DF 7A 15
0D88 1C 10 FA 1E 08 DF 6A 7B A5
0D90 DF 7A 1C DF 69 06 10 7A EA
0D98 F7 14 10 FB 0D 20 EE 21 F7
0DA0 E9 0B 22 29 0C ED 5B 1E 5E
0DA8 0C 21 E8 0B 0E 10 06 08 01
0DB0 36 7C 23 1A 05 07 16 20 BE
0DB8 30 02 16 3A 72 23 72 23 71
0DC0 10 F3 36 7C 2B CD 49 0E D1
0DC8 01 13 0D 20 E1 C9 2A 29 E3
0DD0 0C 7D E6 3F FE 29 C9 0E 89
0DD8 20 18 02 0E 3A D7 EF D8 05
0DE0 71 23 71 23 CD 0D 0E C9 C6
0DE8 ED 5B 1E 0C 21 EA 08 0E 88
0DF0 10 06 08 05 7E C6 00 CB CF
0DF8 12 23 23 10 F7 7A CD 49 F4
0E00 0E D1 12 13 0D 20 EA C9 F2
0E08 11 FF FF 18 03 11 01 00 52
0E10 2A 29 0C 7D 19 E6 3F FE 36
0E18 29 38 0E 18 08 11 C0 FF 88
0E20 18 03 11 40 00 2A 29 0C F9
0E28 19 7D E6 3F 11 30 00 FE 30
0E30 1C 29 0F FE 39 28 08 11 0C
0E38 00 FF FE 0B 28 04 FE 27 6F
0E40 20 01 19 07 08 22 29 0C BE
0E48 C9 11 30 00 19 CB 94 CB A3
0E50 DC C9 11 08 0E 12 00 0E 57
0E58 13 1D 0E 14 22 0E 20 D7 DF
0E60 0D 2E DB 0D 44 36 0D 43 5B
0E68 A7 0E 50 74 0E 4E 80 0E 09
  
```

```

ZERP Z80 Assembler - Source Listing

      0010 ;Dieses Programm erzeugt
      0020 ;die BASIC-Blockgrafik
      0030 ;und ist voll verschiebbar.
      0040 ;
21AC 9C00      0050 GRAM      EQU #9C00
21AC 005B      0060 MRET      EQU #5B
      0070 ;
0D00          0080          ORG #D00
0D00 21009C    0090 GRA      LD HL,GRAM
0D03 1EC0     0100          LD E,#C0
0D05 53       0110 GR3     LD D,E
0D06 0E03    0120          LD C,3
0D08 0605    0130 GR0     LD B,5
0D0A AF      0140          XOR A
0D0B CB42    0150          BIT 0,D
0D0D 2802    0160          JR Z,GR1
0D0F C6F0    0170          ADD A,#F0
0D11 CB5A    0180 GR1     BIT 3,D
0D13 2802    0190          JR Z,GR2
0D15 C60F    0200          ADD A,#F
0D17 77      0210 GR2     LD (HL),A
0D18 23      0220          INC HL
0D19 10FC    0230          DJNZ GR2
0D1B CB1A    0240          RR D
0D1D 8D      0250          DEC C
0D1E 20E8    0260          JR NZ,GR0
0D20 77      0270          LD (HL),A
0D21 23      0280          INC HL
0D22 1C      0290          INC E
0D23 20E0    0300          JR NZ,GR3
0D25 DF5B    0310          SCAL MRET
      0320 ;
  
```

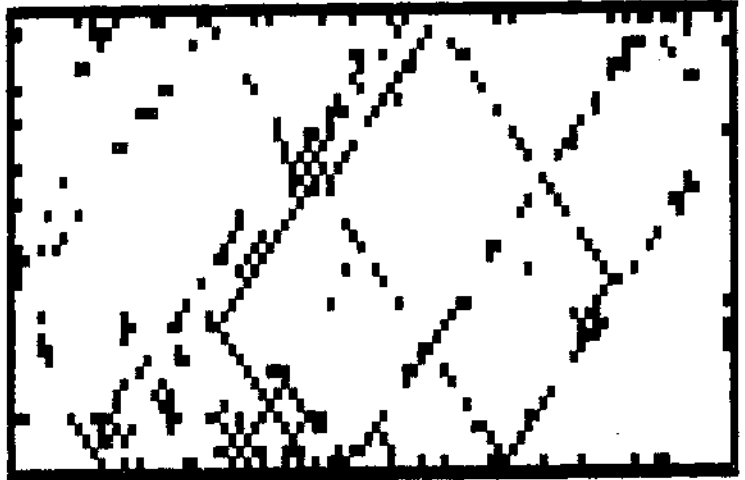
BASIC - Programm zur Demonstration
der Blockgrafik erzeugt bewegte
Zufallsmuster

```

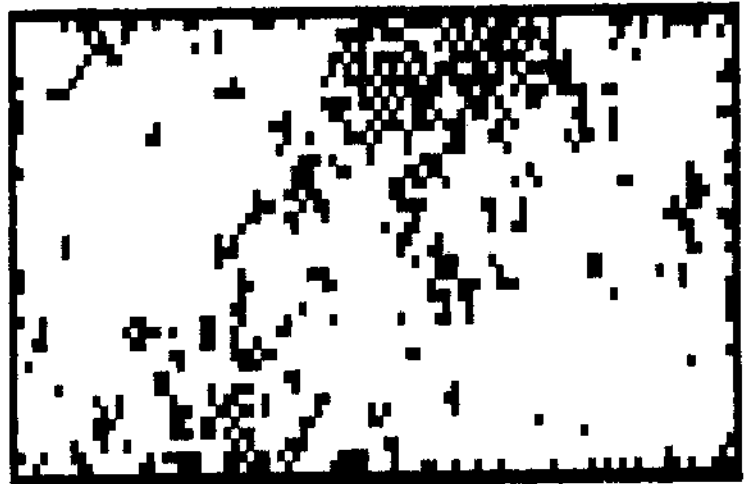
10 CLS
20 FOR J=0 TO 47:SET(47-J,3):SET(48+J,3)
30 SET(47-J,44):SET(48+J,44):NEXT
40 FOR J=3 TO 44:SET(0,J):SET(95,J):NEXT
50 P1=21:P2=30:P3=1:P4=1
60 Q1=50:Q2=20:Q3=-1:Q4=1
70 M1=33:M2=35:M3=1:M4=-1
80 N1=70:N2=26:N3=-1:N4=-1
90 O1=60:O2=17:O3=-1:O4=1
100 R1=61:R2=17:R3=-1:R4=1
110 FOR K=1 TO 3:P3=P3*(1-2*POINT(P1+P3,P2))
120 P4=P4*(1-2*POINT(P1,P2+P4))
130 P1=P1+P3:P2=P2+P4:SET(P1,P2)
140 IF 9*RND(1)>7 THEN M3=-M3
150 M3=M3*(1-2*POINT(M1+M3,M2))
160 M4=M4*(1-2*POINT(M1,M2+M4))
170 RESET(M1,M2):M1=M1+M3:M2=M2+M4:SET(M1,M2)
180 N3=N3*(1-2*POINT(N1+N3,N2))
190 N4=N4*(1-2*POINT(N1,N2+N4))
200 RESET(N1,N2):N1=N1+N3:N2=N2+N4:SET(N1,N2)
210 Z7=Z6:Z6=Z5:Z5=Z4:Z4=Z3:Z3=Z2:Z2=P1
220 X7=X6:X6=X5:X5=X4:X4=X3:X3=X2:X2=P2
230 O3=O3*(1-2*POINT(O1+O3,O2))
240 O4=O4*(1-2*POINT(O1,O2+O4))
250 RESET(O1,O2):O1=O1+O3:O2=O2+O4:SET(O1,O2)
260 R3=R3*(1-2*POINT(R1+R3,R2))
270 R4=R4*(1-2*POINT(R1,R2+R4))
280 RESET(R1,R2):R1=R1+R3:R2=R2+R4:SET(R1,R2)
290 I=INT(94*RND(1)+1):J=INT(4+40*RND(1))
300 IF POINT(I+1,J)+POINT(I-1,J)+POINT(I,J+1)+PO
INT(I,J-1)>0 THEN SET(I,J)
310 RESET(94*RND(1)+1,4+40*RND(1))
320 RESET(Z7,X7):NEXT
330 Q3=Q3*(1-2*POINT(Q1+Q3,Q2))
340 Q4=Q4*(1-2*POINT(Q1,Q2+Q4))
350 Q1=Q1+Q3:Q2=Q2+Q4:SET(Q1,Q2)
360 N=N+1:SCREEN 15,1
370 PRINT "ZYKLUS";N
380 GOTO 110

```

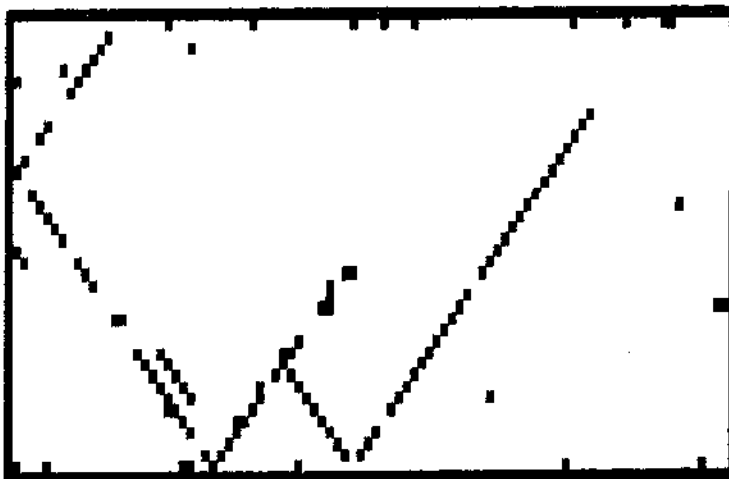
ZYKLUS 502



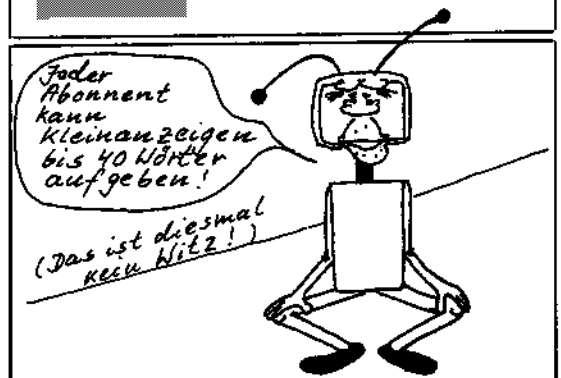
ZYKLUS 1362



ZYKLUS 150

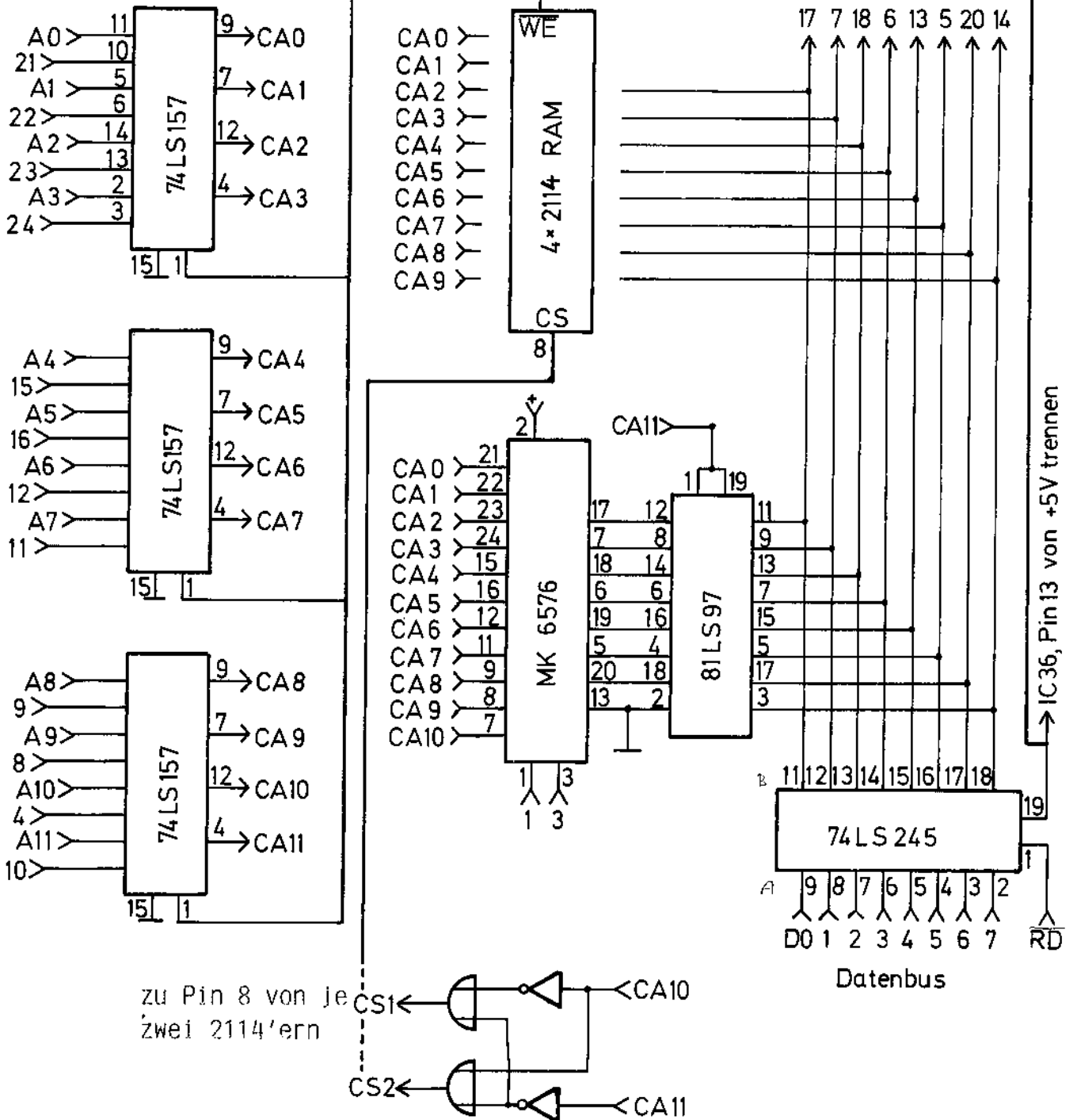


VERKAUFE ZEAP 1 Orig. Cassette mit
Handbuch DM 20,-
NASCOM 1 Zeichen-ROM MCM6576 DM 10,-
NASCOM 2 Grafik-ROM DM 20,-
Texteditor (4K m,Umlauten für
NASSYS 3 DM 40,-
Michael Bach



Anschl. 1-24 vom 24-pol. Stecker zum NASCOM 1 Zeichengener.-Sockel

IC 22 u. IC 36 auf 32K-RAM-Platine



Schnelle Tastatur-Abfrage

von Josef Zeller

1) Problemstellung

Die Tastaturabfrage und die Ermittlung des Zeichens ist beim Nascom System der CPU uebertragen. Damit wird die CPU mit "Verwaltungsaufgaben belastet, die viel Zeit benoetigen. Diese Zeitbelastung fuer die CPU faellt natuerlich nicht weiter auf, wenn der Rechner darauf wartet, bis der Benutzer eine Taste drueckt. Bei Programmen, die waehrend der Laufzeit die Tastatur abfragen, um z.B. das Auflisten zu stoppen oder das Programm abzubrechen, nimmt die Zeit fuer die Tastaturabfrage einen relativ grossen Teil der Programmlaufzeit in Anspruch. Im folgenden soll eine Moeglichkeit gezeigt werden, wie die Abfrage, ob eine Taste gedruickt ist, rapide verkuerzt werden kann.

2) Tastaturhardware - Prinzip

Die Tasten beim Nascom 1 sind in einer 8mal6 Matrix (8 Spalten und 6 Zeilen) angeordnet. Am Anfang der Tastaturroutine setzt die CPU ueber "KBD Counter Reset" die Zaehler auf der Tastaturplatine zurueck. Danach werden ueber "KBD Counter Clock" die 8 Spalten der Matrix jeweils einmal durch die CPU gepulst, d.h. die Leitungen (D0-D7 siehe Originalschaltbild) werden jeweils einmal auf low gesetzt. Nachdem eine Spalte getaktet wurde, fragt die CPU den Port 00 ab. In diesem Port ist der Zustand der 6 Spalten angegeben. Wird z.B. die Taste "B" gedruickt, so wird beim Takten von Spalte D1 die Zeile S3 auf low gesetzt. Ist eines der Bits 0-5 auf low, erkennt die CPU, dass eine Taste gedruickt wurde, und ermittelt daraus das ASCII-Zeichen. Nur um zu ermitteln, ob ueberhaupt eine Taste gedruickt wurde, benoetigt die CPU ca. 940 Taktzyklen.

3) Ergebnis

Es wurde also eine Moeglichkeit gesucht, diese Abfrage erheblich zu verkuerzen. Die dazu benoetigte Hardwareaenderung ist minimal. Das Programm benoetigt nur noch 124 Taktzyklen und ist damit ueber 7 mal schneller als das Original. Der Assembler uebersetzt z.B. eine Quelle mit 4,5 Bytes Quelltext statt in 32 sec. in 18 sec.

4) Hardware

Benoetigt wird dazu ein CTC, durch den die Spalten gepulst werden, und ein OR-Gatter,

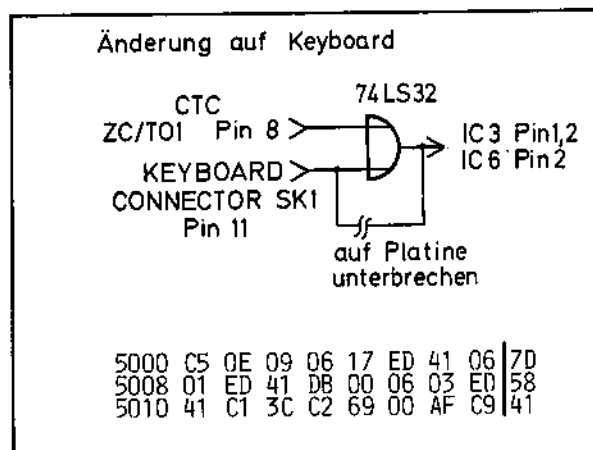
damit entweder der CTC oder die CPU den "KBD Counter Clock" takten kann.

5) Programm

Das Programm EXTKBD wird anstatt der Nascom Monitorroutine aufgerufen. Der CTC wird initialisiert und gibt alle 16 Takte einen Impuls aus. Die CPU fragt den Zustand von Port 00 ab und kehrt, wenn keine Taste gedruickt wurde, ins Aufruferprogramm zurueck. Wurde eine Taste gedruickt, wird die Monitorroutine KBD aufgerufen und das Zeichen ermittelt.

6) Weiterentwicklung

Eine interruptgesteuerte KBD-Routine ist bei mir ebenfalls im Einsatz. Sie ist vom Programmaufwand etwas umfangreicher. Leider ist das Programm noch nicht so ausgereift, um es zu veroeffentlichen. Sobald es aber zuverlässig funktioniert, wird es nachgeliefert.



```

0002 CTCCH1 EQU #09 ;I/O PORT CTC CHANNEL 1
0003 KBD EQU #0069 ;NASCOM-UNTERPROGRAMM
0004 ;DAS DIE TASTATUR NACH ZEICHEN ABFRAGT
0005 ;
0006 ;
0007 ;
0008 ORG #5000
0009 ;
0010 ;
0011 ;
0012 ; - EXTKBD - NEUE KEYBOARD ROUTINE
0013 ;ANSTATT VON NASCOM KBD ANSPRINGEN
0014 EXTKBD PUSH BC ;RETTE REGISTER BC
0015 LD C,CTCCH1 ;LADE I/O ADRESSE
0016 LD B,#17 ;CHANNEL CONTROL REGISTER
0017 OUT (C),B ;AUSGEBEN
0018 LD B,#01 ;TIME CONSTANT REGISTER
0019 OUT (C),B ;AUSGEBEN
0020 IN A,(00) ;ABFRAGE OB TASTE GEDRUECKT
0021 LD B,03 ;RESET CTC ZUM
0022 OUT (C),B ;CTC CHANNEL 1 AUSGEBEN
0023 POP BC ;HOLE BC AUS STACK
0024 INC A ;WURDE KEINE TASTE GEDRUECKT
0025 ;STEHT IN REGISTER A NUN 00
0026 JP NZ,KBD ;TASTE WURDE GEDRUECKT
0027 ;IN KBD WIRD ZEICHEN ERMITTELT UND
0028 ;AUFRUFERPROGRAMM UEBERGEHEN.
0029 XOR A ;KEINE TASTE GEDRUECKT.RUECK-
0030 RET ;KEHR INS AUFRUFERPROGRAMM
  
```


HAUSHALTS- BUCHFÜHRUNG

von Klaus Mombaur

Start: Ohne Drucker 1000
Mit Drucker 2000

1000 - 177F Grd Programm
2000 - 27FF Druckerprogr
2800 - 28FF U - Programme
3000 - 302F Kontenstände
3030 - 303F Monat, Jahr
3040 - 4FFF Daten

```

1000 3E LD A,0C
1002 F7 ROUT
1003 EF PRS "Haushalts
buchführung

1032 00
1033 00
1034 3E LD A,8A
1036 04 LD B,30
1038 F7 ROUT
1039 05 DEC B
103A 20 JR NZ,FC<1038>
103C 00 NOP
103D EF PRS
103E - 1065 (z.B. Postscheckkto mit Nr)
1046 dann:" . . (P)
104E 00
104F 00

1070 EF PRS
1071 - 1098 (z.B. Girokonto mit Nr)
1099 dann:" . . (G)
10A1 00
10A2 00
10A3 EF PRS
10A4 - 10CB (z.B. Sparkonto mit Nr)
10CC dann:" . . (S)
10D4 00
10D5 00
10D6 EF PRS
10D7 - 10FE (z.B. Sparvertrag mit Nr)
10FF dann:" . . (U)
1107 00
1108 00
1109 EF PRS
110A - 1131 (z.B. Darlehenskonto mit Nr)
1132 dann:" . . (D)
113A 00
113B 00
113C EF PRS
113D - 1164 (z.B. Haushaltsbarkasse)
1165 dann:" . . (H)
1166 00
116E 00
116F EF PRS "Welches Konto (Kennbuch-
1170 staben) bitte ?
1197 00
1198 DF BLINK
1199 70
119A FE CP 50 =P?
119C CA JP Z,1340 >PS-Konto
119F FE CP 47 =B?
11A1 CA JP Z,1440 >Girokonto
11A4 FE CP 53 =S?
11A6 CA JP Z,1540 >Sparkonto
11A9 FE CP 56 =U?
11AB CA JP Z,1640 >Sparv.konto
11AE FE CP 44 =D?
11B0 CA JP Z,16E0 >Darl.konto
11B3 FE CP 48 =H?
11B5 28 JR Z,02<11B9> >Haushaltskasse
11B7 18 JR DF<1198> Wdh

=====
Haushaltskasse
=====

11B9 3E LD A,0C
11BB F7 ROUT
11BC CD CALL 29C0 >Zus.Zeile
11BF CD CALL 2800 >Angebot
11C2 FE CP 6E =n?
11C4 CA JP Z,12E0 >Neuer Monat
11C7 FE CP 41 =A?

```

```

11C9 CA JP Z,1260 >Ausgänge
11CC FE CP 45 =E?
11CE CA JP Z,1200 >Eingänge
=K
Kontostand ausg:
CLS

11D1 3E LD A,0C
11D3 F7 ROUT
11D4 Z1 LD HL,3028 Anf. Ktospeicher
11D7 CD CALL 28E0 >Kontenstand
11DA CD CALL 2BE0 >Führ 00 weg
11DD C3 JP 2920 >Ende
11FE Speicherzelle

Eingänge:
CLS

1200 3E LD A,0C
1202 F7 ROUT
1203 CD CALL 2A00 >Eing Monat
1206 Z1 LD HL,3040 Spei Anf.
1209 11 LD DE,2A00 /Adr für
120C ED53 LD (2B41),DE /UP "Listen"
1210 CD CALL 2AE1 >Listen
Neue Eingänge:
Cursor

1213 DD21 LD IX,0B4A
1217 DD22 LD (0C29),IX
1218 EF PRS "Neue Eingänge ? (J/N)

1232 00
1233 E5 PUSH HL
1234 DF BLINK
1235 78
1236 E1 POP HL
1237 E5 PUSH HL
1238 FE CP 4A
123A 28 JR Z,03<123F>
123C C3 JP 11D1 =N:Kto Stand
123F CD CALL 2A30 >Betrag speicher
1242 FD21 LD IY,302F Spei Ende H.kass
1244 0E LD C,00 C löschen
1248 00 NOP
1249 00 NOP
124A CD CALL 2A65 >Addieren
124D E1 POP HL
124E CD CALL 2AB0 >Text speichern
1251 C3 JP 1200 >Eingänge

Ausgänge:
CLS

1260 3E LD A,0C
1262 F7 ROUT
1263 CD CALL 2B50 >Ausg Monat
1266 Z1 LD HL,3200 Spei Anf.
1269 11 LD DE,2B50 /Adr für
126C ED53 LD (2B41),DE /UP "Listen"
1270 CD CALL 2AE1 >Listen
Neue Ausgänge:
Cursor

1273 DD21 LD IX,0B4A
1277 DD22 LD (0C29),IX
1278 EF PRS "Neue Ausgänge ? (J/N)

1292 00
1293 E5 PUSH HL
1294 DF BLINK
1295 78
1296 E1 POP HL
1297 E5 PUSH HL
1298 FE CP 4A
129A 28 JR Z,03<129F>
129C C3 JP 11D1 =N:Kto Stand
129F CD CALL 2A30 >Betrag speicher
12A2 FD21 LD IY,302F Spei Ende H.kass
12A4 0E LD C,00 C löschen
12A8 DD21 LD IX,2A79 /vor >Addieren
12AC DD36 LD (IX+00),90 /dies in
12B0 DD36 LD (IX+02),91 /Subtrahieren
12B4 DD36 LD (IX+0B),90 /wandeln
12B8 CD CALL 2A65 >Addieren
12BB DD21 LD IX,2A79 /Rückwandeln
12BF DD36 LD (IX+00),88 /in
12C3 DD36 LD (IX+02),91 /Addieren
12C7 DD36 LD (IX+0B),10 /
12C8 E1 POP HL
12CC CD CALL 2AB0 >Text speichern
12CF C3 JP 1260 >Ausgänge

Neuer Monat:
Zähler 14 Zchn
Spei Anf. Monat
CLS

"Geben Sie mir Monatsname und
Jahreszahl dann *

```

1312	0D			1401	CD	CALL	2A30		
1313	00			1404	FD21	LD	IY,3007		
1314	DF	INLIN		1408	0E	LD	C,00		
1315	63		DE ist Zeilenbeg	140A	0D21	LD	IX,2A79		
1316	EB	EX	DE,HL	140E	0D36	LD	(IX+00),90		
1317	7E	LD	A,(HL)	1412	0D36	LD	(IX+02),91		
1318	FE	CP	2A	1416	0D36	LD	(IX+0B),90		
131A	28	JR	Z,06<1322>	141A	CD	CALL	2A65		
131C	EDA0	LDI		141D	0D21	LD	IX,2A79		
131E	20	JR	NZ,F7<1317>	1421	0D36	LD	(IX+00),88		
1320	18	JR	07<1329>	1425	0D36	LD	(IX+02),91		
1322	3E	LD	A,00	1429	0D36	LD	(IX+0B),10		
1324	12	LD	(DE),A	142D	E1	POP	HL		
1325	13	INC	DE	142E	CD	CALL	2A80		
1326	0D	DEC	C	1431	C3	JP	13C2		
1327	20	JR	NZ,F9<1322>	1434	3E	LD	A,00		
1329	3E	LD	A,00	1436	32	LD	(4000),A	Neuer Monat:	
132B	32	LD	(3040),A	1439	32	LD	(4200),A	nur 1.Speizelle	
132E	32	LD	(3200),A	143C	C3	JP	1000	auf 00=löschen	
1331	C3	JP	1000	143F	00	NOP			
			>Anfang						
			=====						
			PS-Konto						=====
			=====**						=====
1340	3E	LD	A,00	1440	3E	LD	A,00		
1342	F7	ROUT		1442	F7	ROUT			
1343	CD	CALL	29C0	1443	CD	CALL	29C0	siehe	
1344	CD	CALL	2800	1446	CD	CALL	2800	PS-Konto	
1349	FE	CP	6E	1449	FE	CP	6E		
134B	CA	JP	Z,1434	144B	CA	JP	Z,1534		
134E	FE	CP	41	144E	FE	CP	41		
1350	CA	JP	Z,13C2	1450	CA	JP	Z,14C2		
1353	FE	CP	45	1453	FE	CP	45		
1355	CA	JP	Z,1370	1455	CA	JP	Z,1470		
			Kontostand ausg	1458	3E	LD	A,00	Kontostand ausg:	
1358	3E	LD	A,00	145A	F7	ROUT			
135A	F7	ROUT		145B	21	LD	HL,3008		
135B	21	LD	HL,3000	145E	CD	CALL	28E0		
135E	CD	CALL	28E0	1461	CD	CALL	28E0		
1361	CD	CALL	28E0	1464	C3	JP	2920		
1364	C3	JP	2920					Eingänge:	
			Eingänge:	1470	3E	LD	A,00		
1370	3E	LD	A,00	1472	F7	ROUT			
1372	F7	ROUT		1473	CD	CALL	2A00		
1373	CD	CALL	2A00	1476	21	LD	HL,4600		
1376	21	LD	HL,4000	1479	11	LD	DE,2A00		
1379	11	LD	DE,2A00	147C	ED53	LD	(2B41),DE		
137C	ED53	LD	(2B41),DE	1480	CD	CALL	2AE1		
1380	CD	CALL	2AE1	1483	DD21	LD	IX,0B4A	Neue Eingänge:	
1383	DD21	LD	IX,0B4A	1487	DD22	LD	(0C29),IX		
1387	DD22	LD	(0C29),IX	148B	EF	PRS			
138B	EF	PRS		14A2	00				
13A2	00			14A3	E5	PUSH	HL		
13A3	E5	PUSH	HL	14A4	DF	BLINK			
13A4	DF	BLINK		14A5	7B				
13A5	7B			14A6	E1	POP	HL		
13A6	E1	POP	HL	14A7	E5	PUSH	HL		
13A7	E5	PUSH	HL	14A8	FE	CP	4A		
13A8	FE	CP	4A	14AA	28	JR	Z,03<14AF>		
13AA	28	JR	Z,03<14AF>	14AC	C3	JP	1458		
13AC	C3	JP	1358	14AF	CD	CALL	2A30		
13AF	CD	CALL	2A30	14B2	FD21	LD	IY,300F		
13B2	FD21	LD	IY,3007	14B6	0E	LD	C,00		
13B4	0E	LD	C,00	14B8	CD	CALL	2A65		
13B6	CD	CALL	2A65	14B8	E1	POP	HL		
13B8	E1	POP	HL	14BC	CD	CALL	2AB0		
13BC	CD	CALL	2AB0	14BF	C3	JP	1470	Wdh Giro-Eing	
13BF	C3	JP	1370					Ausgänge:	
			Wdh PS-Eingänge	14C2	3E	LD	A,00		
			Ausgänge:	14C4	F7	ROUT			
13C2	3E	LD	A,00	14C5	CD	CALL	2B50		
13C4	F7	ROUT		14C8	21	LD	HL,4800		
13C5	CD	CALL	2B50	14CB	11	LD	DE,2B50		
13C8	21	LD	HL,4200	14CE	ED53	LD	(2B41),DE		
13CB	11	LD	DE,2B50	14D2	CD	CALL	2AE1		
13CE	ED53	LD	(2B41),DE	14D5	DD21	LD	IX,0B4A	Neue Ausgänge:	
13D2	CD	CALL	2AE1	14D9	DD22	LD	(0C29),IX		
13D5	DD21	LD	IX,0B4A	14DD	EF	PRS			
13D9	DD22	LD	(0C29),IX	14F4	00				
13D0	EF	PRS		14F5	E5	PUSH	HL		
13F4	00			14F6	DF	BLINK			
13F5	E5	PUSH	HL	14F7	7B				
13F6	DF	BLINK		14F8	E1	POP	HL		
13F7	7B			14F9	E5	PUSH	HL		
13F8	E1	POP	HL	14FA	FE	CP	4A		
13F9	E5	PUSH	HL	14FC	28	JR	Z,03<1501>		
13FA	FE	CP	4A	14FE	C3	JP	1458		
13FC	28	JR	Z,03<1401>	1501	CD	CALL	2A30		
13FE	C3	JP	1358	1504	FD21	LD	IY,300F		

1508	0E	LD	C,00		160E	DD36	LD	(IX+00),90	
150A	DD21	LD	IX,2A79		1612	DD36	LD	(IX+02),91	
150E	DD36	LD	(IX+00),90		1616	DD36	LD	(IX+0B),90	
1512	DD36	LD	(IX+02),91		161A	CD	CALL	2A65	
1516	DD36	LD	(IX+0B),90		161D	DD21	LD	IX,2A79	
151A	CD	CALL	2A65		1621	DD36	LD	(IX+00),88	
151D	DD21	LD	IX,2A79		1625	DD36	LD	(IX+02),81	
1521	DD36	LD	(IX+00),88		1629	DD36	LD	(IX+0B),10	
1525	DD36	LD	(IX+02),81		162D	E1	POP	HL	
1529	DD36	LD	(IX+0B),10		162E	CD	CALL	2AB0	
152D	E1	POP	HL		1631	C3	JP	15C2	
152E	CD	CALL	2AB0		1634	3E	LD	A,00	Neuer Monat:
1531	C3	JP	14C2		1636	32	LD	(4C00),A	
1534	3E	LD	A,00	Neuer Monat:	1639	32	LD	(4E00),A	
1536	32	LD	(4600),A		163C	C3	JP	1000	
1539	32	LD	(4800),A		163F	00			
153C	C3	JP	1000						=====
153F	00	NOP							Sparvertrag
				=====					=====
			Sparkonto	1640	3E	LD	A,0C		ab hier nur
				1642	F7	ROUT			Beträge zum
1540	3E	LD	A,0C	1643	00	NOP			Kontostand
1542	F7	ROUT		1644	00	NOP			add/subt.
1543	CD	CALL	29C0	1645	00	NOP			Kein Text
1546	CD	CALL	2800	1646	CD	CALL	2800		Kein Neuer Monat
1549	FE	CP	6E	1649	00	NOP			
154B	CA	JP	Z,1634	164A	00	NOP			
154E	FE	CP	41	164B	00	NOP			
1550	CA	JP	Z,15C2	164C	00	NOP			
1553	FE	CP	45	164D	00	NOP			Sonst wie
1555	CA	JP	Z,1570	164E	FE	CP	41		PS-Konto
1558	3E	LD	A,0C	1650	CA	JP	Z,1691		
155A	F7	ROUT		1653	FE	CP	45		
155B	Z1	LD	HL,3010	1655	CA	JP	Z,1670		
155E	CD	CALL	28E0	1658	3E	LD	A,0C	Kontostand ausg:	
1561	CD	CALL	28E0	165A	F7	ROUT			
1564	C3	JP	2920	165B	Z1	LD	HL,3018		
			Eingänge:	165E	CD	CALL	28E0		
				1661	CD	CALL	28E0		
				1664	C3	JP	2920		
1570	3E	LD	A,0C	1670	EF	PRs			Eingänge:
1572	F7	ROUT		1671	0C				CLS
1573	CD	CALL	2A00			"Betrag: ?			ZV
1576	Z1	LD	HL,4C00	167A	0D				DE ist Zeilenbeg
1579	11	LD	DE,2A00	167B	00				Zbeg+1 für UP
157C	ED53	LD	(2B41),DE	167C	DF	INLIN			
1580	CD	CALL	2AE1	167D	63				
1583	DD21	LD	IX,0B4A	167E	13	INC	DE		
1587	DD22	LD	(0C29),IX	167F	1A	LD	A,(DE)		
158B	EF	PRs		1680	FE	CP	Z0		
15A2	00			1682	Z0	JR	NZ,FA<167E>		
15A3	E5	PUSH	HL	1684	EB	EX	DE,HL		
15A4	DF	BLINK		1685	FD21	LD	IY,301F	Ende Spei Sparv.	
15A5	7B			1689	0E	LD	C,00	C löschen	
15A6	E1	POP	HL	168B	CD	CALL	2A65	>Addieren	
15A7	E5	PUSH	HL	168E	C3	JP	1658	>Ktostd ausg	
15A8	FE	CP	4A	1691	EF	PRs		Ausgänge:	
15AA	Z0	JR	Z,03<15AF>	1692	0C			CLS	
15AC	C3	JP	1558			"Betrag: ?			ZV
15AF	CD	CALL	2A30	169B	0D				Zbeg+1 für UP
15B2	FD21	LD	IY,3017	169C	00				
15B6	0E	LD	C,00	169D	DF	INLIN			
15B8	CD	CALL	2A65	169E	63				
15BB	E1	POP	HL	169F	13	INC	DE		
15BC	CD	CALL	2AB0	16A0	1A	LD	A,(DE)		
15BF	C3	JP	1570	16A1	FE	CP	Z0		
15C2	3E	LD	A,0C	16A3	Z0	JR	NZ,F9<169E>		
15C4	F7	ROUT		16A5	EB	EX	DE,HL		
15C5	CD	CALL	2B50	16A6	FD21	LD	IY,301F		
15C8	Z1	LD	HL,4E00	16AA	0E	LD	C,00		
15CB	11	LD	DE,2B50	16AC	DD21	LD	IX,2A79		
15CE	ED53	LD	(2B41),DE	16B0	DD36	LD	(IX+00),90		
15D2	CD	CALL	2AE1	16B4	DD36	LD	(IX+02),91		
15D5	DD21	LD	IX,0B4A	16B8	DD36	LD	(IX+0B),90		
15D9	DD22	LD	(0C29),IX	16BC	CD	CALL	2A65		
15DD	EF	PRs		16BF	DD21	LD	IX,2A79		
15F4	00			16C3	DD36	LD	(IX+00),88		
15F4	00			16C7	DD36	LD	(IX+02),81		
15F5	E5	PUSH	HL	16CB	DD36	LD	(IX+0B),10		
15F6	DF	BLINK		16CF	C3	JP	1658		
15F7	7B								=====
15F8	E1	POP	HL						Darl.konto
15F9	E5	PUSH	HL						=====
15FA	FE	CP	4A	16E0	3E	LD	A,0C		wie
15FC	Z0	JR	Z,03<1601>	16E2	F7	ROUT			Sparvertrag
15FE	C3	JP	1558	16E3	00	NOP			
1601	CD	CALL	2A30	16E4	00	NOP			
1604	FD21	LD	IY,3017						
1608	0E	LD	C,00						
160A	DD21	LD	IX,2A79						

```

14E5 00 NOP
14E6 CD CALL 2800
14E9 00 NOP
14EA 00 NOP
14EB 00 NOP
14EC 00 NOP
14ED 00 NOP
14EE FE CP 41
14F0 CA JP Z,1731
14F3 FE CP 45
14F5 CA JP Z,1710
14F8 3E LD A,0C Kontostand ausg
14FA F7 ROUT
14FB 21 LD HL,3020
14FE CD CALL 28E0
1701 CD CALL 28E0
1704 C3 JP 2920

1710 EF PRS
1711 0C CLS
"Betrag: ?"
171A 00 ZV
171B 00
171C DF INLIN
171D 63
171E 13 INC 0E
171F 1A LD A,(0E)
1720 FE CP 20
1722 20 JR NZ,F9<171D>
1724 EB EX DE,HL
1725 FD21 LD IY,3027
1729 0E LD C,00
172B CD CALL 2A65
172E C3 JP 16F8
1731 EF PRS
1732 0C
173B 0D
173C 00
173D DF INLIN
173E 63
173F 13 INC DE
1740 1A LD A,(DE)
1741 FE CP 20
1743 20 JR NZ,F9<173E>
1745 EB EX DE,HL
1746 FD21 LD IY,3027
174A 0E LD C,00
174C DD21 LD IX,2A79
1750 DD36 LD (IX+00),90
1754 DD36 LD (IX+02),91
1758 DD36 LD (IX+08),90
175C CD CALL 2A65
175F DD21 LD IX,2A79
1763 DD36 LD (IX+00),88
1767 DD36 LD (IX+02),81
176B DD36 LD (IX+08),10
176F C3 JP 16F8

Eingänge:
2070 EF PRS
"Mit Ueberschrift (U)
Wieviel Zeilen? (10)"
209B 00
209C DF BLINK
209D 7B
209E F7 ROUT
209F 11 LD DE,0080
20A2 FE CP 55 =U?
20A4 20 JR NZ,10<20B6>
20A6 DF BLINK =U:
20A7 7B A hat Zeil.zahl
20A8 F7 ROUT
20A9 11 LD DE,0080
20AC FE CP 30 <30?
20AE 30 JR NC,02<20B2>
20B0 3E LD A,3A A hat Standard
20B2 C6 ADD A,02 +2 mit Uebers.
20B4 18 JR 06<20BC>
20B6 FE CP 30 <30?
20B8 30 JR NC,02<20BC>
20BA 3E LD A,3A A hat Standard
20BC D6 SUB 30 ASCII in Hex
20BE 47 LD B,A B hat Zeil.zahl
20BF 21 LD HL,0080A
20C2 19 ADD HL,DE ? HL hat 1.Druze
20C3 7E LD A,(HL) /BS ausdrucken
20C4 FE CP 00 /
20C6 20 JR NZ,0D<20D5> /
20C8 3E LD A,0F /
20CA 0D ADC A,L /
20CB LD L,A /
20CC 30 JR NC,04<20D2> /
20CE 3E LD A,01 /
20D0 04 ADD A,H /
20D1 67 LD H,A /
20D2 3E LD A,0D /Drubefehl
20D4 05 DEC B /
20D5 CD CALL 2050 />U- Druck
20D8 23 INC HL /
20D9 78 LD A,B /
20DA FE CP 00 /
20DC 20 JR NZ,E5<20C3> /
20DE C9 RET /

Unterprogramme
2000 3E LD A,0F
2002 D3 OUT (06),A Drucker
2004 3E LD A,CF initialisieren
2006 D3 OUT (07),A
2008 3E LD A,F5
200A D3 OUT (07),A
200C 3E LD A,FF
200E D3 OUT (05),A
2010 CB9F RES 3,A
2012 D3 OUT (05),A
2014 CBDF SET 3,A
2016 D3 OUT (05),A
2018 EF PRS
2019 0C CLS
"Druckart, etc eingeben
Ueberschrift, dann '1'"
203E 00
203F 00 ZV
2040 DF BLINK
2041 7B
2042 F7 ROUT
2043 FE CP 5D =J?
2045 28 JR Z,05<204C>
2047 CD CALL 2050 >U- Druck
204A 18 JR F4<2040>
204C C3 JP 1000 >Hptprogramm

2050 D3 OUT (04),A
2052 DB IN A,(05)
2054 CB9F RES 1,A
2056 D3 OUT (05),A
2058 CB9F SET 1,A
205A D3 OUT (05),A
205C DB IN A,(05)
205E CB67 BIT 4,A
2060 20 JR NZ,FA<205C>
2062 C9 RET

2070 EF PRS
"Mit Ueberschrift (U)
Wieviel Zeilen? (10)"
209B 00
209C DF BLINK
209D 7B
209E F7 ROUT
209F 11 LD DE,0080
20A2 FE CP 55 =U?
20A4 20 JR NZ,10<20B6>
20A6 DF BLINK =U:
20A7 7B A hat Zeil.zahl
20A8 F7 ROUT
20A9 11 LD DE,0080
20AC FE CP 30 <30?
20AE 30 JR NC,02<20B2>
20B0 3E LD A,3A A hat Standard
20B2 C6 ADD A,02 +2 mit Uebers.
20B4 18 JR 06<20BC>
20B6 FE CP 30 <30?
20B8 30 JR NC,02<20BC>
20BA 3E LD A,3A A hat Standard
20BC D6 SUB 30 ASCII in Hex
20BE 47 LD B,A B hat Zeil.zahl
20BF 21 LD HL,0080A
20C2 19 ADD HL,DE ? HL hat 1.Druze
20C3 7E LD A,(HL) /BS ausdrucken
20C4 FE CP 00 /
20C6 20 JR NZ,0D<20D5> /
20C8 3E LD A,0F /
20CA 0D ADC A,L /
20CB LD L,A /
20CC 30 JR NC,04<20D2> /
20CE 3E LD A,01 /
20D0 04 ADD A,H /
20D1 67 LD H,A /
20D2 3E LD A,0D /Drubefehl
20D4 05 DEC B /
20D5 CD CALL 2050 />U- Druck
20D8 23 INC HL /
20D9 78 LD A,B /
20DA FE CP 00 /
20DC 20 JR NZ,E5<20C3> /
20DE C9 RET /

U- BS ausdrucken
(bei Ktostand)
20E0 06 LD B,01 nur 2.BS Zeile
20E2 21 LD HL,004A
20E5 CD CALL 20C3
20E8 C3 JP 2018 >U- BS ausdr
>Beginn

U- Angebot
2800 EF PRS
2801 0D ZV
2802 0D ZV
"Wollen Sie Eingänge buchen
(pruefen)...(E)"
2833 00
2834 EF PRS
2835 0D ZV
"Wollen Sie Ausgänge buchen
(pruefen)...(A)"
2866 00
2867 EF PRS
2868 0D ZV
"Wollen Sie den letzten
Kontostand...(K)"
2899 00

```

289A	EF	PRS			299F	DF	BLINK		
289B	0D		ZU	"Geben Sie einen Kennbuchstaben...?"	29A0	7B			
					29A1	FE	CP 44	=D?	
					29A3	CC	CALL 2,2070	>BS-ausdrucken	
28CA	00				29A4	E1	POP HL		
28CB	DF	BLINK			29A7	EF	PRS		CLS
28CC	7B				29A8	0C			
28CD	FE	CP 6E	=n?		29A9	00			
28CE	C8	RET Z			29AA	C9	RET		
28CF	C8	CP 45	=E?						
28D0	FE	CP 41	=A?						U- Zus Zeile
28D2	C8	RET Z							
28D3	FE	CP 41	=A?						
28D5	C8	RET Z			29C0	EF	PRS	"Wollen Sie einen neuen Monat beginnen . . . (n)"	
28D6	FE	CP 4B	=K?						
28D8	C8	RET Z							
28D9	3E	LD A,0C	CLS		29F1	00			
28DB	F7	ROUT			29F2	C9	RET		
28DC	C3	JP 2800	Wdh						
									U- Eing Monat
					2A00	01	LD BC,000E	Monat aus	
					2A03	21	LD HL,3030	Monatsspei	
					2A06	11	LD DE,2A1F	holen	
8E0	EF	PRS	ZU	"Der Kontostand betrægt: DM"	2A09	EDB0	LDIR		
28E1	0D				2A0B	EF	PRS	"Eingænge im Monat (max 14 Zchn)"	
2900	00				2A1F				
2901	C3	JP 28C0	>U- Pruefen		2A2D	00			
2904	00	NOP			2A2E	C9	RET		
2905	06	LD B,06							
2907	7E	LD A,(HL)							
2908	F7	ROUT	DM anzeigen						U- Betrag speich
2909	00	NOP							
290A	23	INC HL			2A30	EF	PRS		
290B	05	DEC B			2A31	0C	"Betrag ?"		CLS
290C	20	JR NZ,F9<2907>							
290E	3E	LD A,2E			2A3A	0D			ZU
2910	F7	ROUT	Pfg anzeigen		2A3B	00			
2911	06	LD B,02			2A3C	DF	INLIN		DE ist Zeianfg
2913	7E	LD A,(HL)			2A3D	63			HL ist letzte 00
2914	F7	ROUT			2A3E	D5	PUSH DE		
2915	00	NOP			2A3F	0E	LD C,00	Pruefen wieviel	
2916	23	INC HL			2A41	1A	LD A,(DE)	DM - Stellen	
2917	05	DEC B			2A42	FE	CP 2E	=,?	
2918	20	JR NZ,F9<2913>			2A44	28	JR Z,04<2A4A>		
291A	C9	RET			2A46	13	INC DE		
					2A47	0C	INC C		
			U- Ende		2A48	18	JR F7<2A41>		
					2A4A	01	POP DE		
					2A4B	3E	LD A,06	& Stellen	
2920	EF	PRS	ZU		2A4D	91	SUB C	abzgl C mit	
2921	0D		ZU		2A4E	4F	LD C,A	Space fuellen	
2922	0D		ZU	"A = Zurueck zum Angebot"	2A4F	3E	LD A,20		
2923	0D			"D = Kto - Stand drucken"	2A51	77	LD (HL),A		
				"E = Programm beenden"	2A52	23	INC HL		
293A	0D				2A53	0D	DEC C		
					2A54	20	JR NZ,F9<2A4F>		
2950	0D				2A56	05	PUSH DE		
					2A57	EB	EX DE,HL		
2965	0D				2A58	EDA0	LDI		
2966	,?				2A5A	7E	LD A,(HL)		
2967	20				2A5B	FE	CP 20		
2968	00				2A5D	20	JR NZ,F9<2A58>		
2969	DF	BLINK			2A5F	3E	LD A,26	& als Trennung	
296A	7B				2A61	12	LD (DE),A	DM von Text	
296B	FE	CP 44	=D?		2A62	13	INC DE		
296D	CC	CALL 2,20E0	>Drucken		2A63	D1	POP DE		
2970	FE	CP 45	=E?		2A64	C9	RET		
2972	20	JR Z,0A<297E>	>Ende						
2974	FE	CP 41	=A?						
2976	20	JR Z,0B<2983>							U- Addieren
2978	3E	LD A,0C	CLS						
297A	F7	ROUT			2A65	DD21	LD IX,0009	Zae 8+i Stellen	
297B	18	JR A3<2920>	Wdh		2A67	2B	DEC HL	HL ist BS Pos+1	
297D	00	NOP			2A6A	7E	LD A,(HL)		
297E	EF	PRS	Ende:		2A6B	FE	CP 00	Statt 00 bei	
297F	0C		CLS		2A6D	20	JR NZ,02<2A71>	BS Rand 30	
2980	00	NOP			2A6F	3E	LD A,30		
2981	DF	MRET			2A71	06	SUB 30	ASCII in Dez	
2982	5B				2A73	47	LD B,A	B ist neuer Wert	
2983	C3	JP 1000	>Angebot		2A74	FD7E	LD A,(IY+00)		
					2A77	06	SUB 30	alt + neu	
			Dru in -Listen-		2A79	88	ADC A,B	add / (subtr)	
					2A7A	27	DAA		
					2A7B	81	ADD A,C	bei Uebertr +/-1	
2990	DD21	LD IX,0B4A	Cursor		2A7C	0E	LD C,00	Uebertr loeschen	
2994	DD22	LD (0C29),IX			2A7E	27	DAA		
2998	EF	PRS			2A7F	FE	CP 10	wenn >9 C=1	
2999	"D ?"				2A81	38	JR C,04<2A87>	und -10 bei add	
299D	00				2A83	D6	SUB 10	-90 bei sub	
299E	E5	PUSH HL							

2A85	0E	LD	C,01		2B27	DD21	LD	IX,0B4A	
2A87	C6	ADD	A,30	Dez in ASCII	2B28	DD22	LD	(0C29),IX	
2A89	FD77	LD	(IY+00),A	speichern	2B2F	EF	PRS	"0 ?	
2A8C	FD2B	DEC	IY	nächster Wert	2B30				
2A8E	2B	DEC	HL		2B34	00			
2A8F	7E	LD	A,(HL)		2B35	E5	PUSH	HL	
2A90	FE	CP	2E	"." überspringen	2B36	DF	BLINK		
2A92	20	JR	NZ,01<2A95>		2B37	7B			
2A94	2B	DEC	HL		2B38	FE	CP	44	=D?
2A95	DD2B	DEC	IX	Zae -1	2B3A	CC	CALL	Z,2070	>BS- ausdrucken
2A97	DD7E	LD	A,(IX+00)	Zae = 0 ?	2B3D	3E	LD	A,0C	CLS
2A9A	FE	CP	00		2B3F	F7	RQUT		
2A9C	20	JR	NZ,CC<2A6A>	Wdh	2B40	CD	CALL	2A00	>Eing/Ausg Monat
2A9E	00	NOP			2B43	E1	POP	HL	(u.Hptpro laden
2A9F	00	NOP			2B44	18	JR	9B<2AE1>	Wdh
2AA0	00	NOP							=====
2AA1	C9	RET							Ausg Monat
				=====					=====
				U- Text speichern	2B50	01	LD	BC,000E	Monat aus
				=====	2B53	21	LD	HL,3030	Monatsspei
2AB0	7E	LD	A,(HL)	HL auf Platz	2B56	11	LD	DE,2B6F	holen
2AB1	23	INC	HL	nach &	2B59	EDB0	LDIR		
2AB2	FE	CP	26		2B5B	EF	PRS		
2AB4	20	JR	NZ,FA<2AB0>						
2AB6	EF	PRS							
2AB7			"Text ?		2B7D	00			
2ABD	0D			ZV	2B7E	C9	RET		
2ABE	00								=====
2ABF	DF	INLIN							U- Negativ
2AC0	63								=====
2AC1	EB	EX	DE,HL		2B80	0E	LD	C,01	IX ist BS-Zeile
2AC2	EDA0	LDI		Kopieren bis	2B82	3E	LD	A,07	IY ist Speibeg
2AC4	7E	LD	A,(HL)	2 x Space	2B84	FD23	INC	IY	
2AC5	FE	CP	20		2B86	30	DEC	A	Spei auf
2AC7	20	JR	NZ,F9<2AC2>		2B87	20	JR	NZ,FB<2B64>	Pfg
2AC9	23	INC	HL		2B89	16	LD	D,02	2 Schleifen
2ACA	7E	LD	A,(HL)		2B8B	06	LD	B,02	2 Stellen
2ACB	FE	CP	20		2B8D	FD6E	LD	L,(IY+00)	
2ACD	20	JR	NZ,09<2AD0>		2B90	3E	LD	A,39	Differenz zu
2ACF	3E	LD	A,2A		2B92	95	SUB	L	100000.00
2AD1	12	LD	(DE),A		2B93	01	ADD	A,C	bilden
2AD2	3E	LD	A,00	* und 00	2B94	27	DAA		
2AD4	13	INC	DE	ans Ende	2B95	0E	LD	C,00	C für Uebertrag
2AD5	12	LD	(DE),A		2B97	FE	CP	10	
2AD6	C9	RET			2B99	20	JR	NZ,04<2B9F>	Wenn 10, dann
2AD7	2B	DEC	HL		2B9B	06	SUB	10	Uebertr und
2AD8	2B	DEC	HL		2B9D	0E	LD	C,01	ASCII Korrektur
2AD9	18	JR	E7<2AC2>		2B9F	C6	ADD	A,30	
				=====	2BA1	DD77	LD	(IX+00),A	Ziffer auf BS
				U- Listen	2BA4	FD2B	DEC	IY	
				=====	2BA6	DD2B	DEC	IX	
2AE1	11	LD	DE,088A		2BA8	05	DEC	B	
2AE4	01	LD	BC,000A	Zeilenzaehler	2BA9	20	JR	NZ,E2<2B8D>	
2AE7	ED53	LD	(11FE),DE		2BAB	3E	LD	A,2E	3.Stelle= "."
2AEB	7E	LD	A,(HL)	(HL) ist	2BAD	DD77	LD	(IX+00),A	(2."." wird von
2AEC	FE	CP	00	jew Speianfg	2BAE	DD2B	DEC	IX	U- Nullen gel)
2AEE	CA	JP	Z,2990	>Dru in Listen	2BB2	06	LD	B,05	5 Stellen
2AF1	FE	CP	2A	Ausgabe durch	2BB4	15	DEC	D	
2AF3	28	JR	Z,1B<2B18>	Kopieren aus	2BB5	20	JR	NZ,D6<2B6D>	
2AF5	FE	CP	26	jew Speicher	2BB7	3E	LD	A,2D	"-" nach DM
2AF7	20	JR	NZ,0F<2B00>	bis * oder 00	2BB9	DD77	LD	(IX+0B),A	
2AF9	ED5B	LD	DE,(11FE)		2BBC	C9	RET		
2AFD	7B	LD	A,E	DE +11 wenn &					=====
2AFE	CE	ADC	A,0B						U- Prüfen
2B00	5F	LD	E,A						=====
2B01	30	JR	NC,04<2B07>		2B00	E5	PUSH	HL	(HL) ist MS0
2B03	7A	LD	A,D		2B01	7E	LD	A,(HL)	
2B04	C6	ADD	A,01		2B02	FE	CP	39	<39 entspricht
2B06	57	LD	D,A		2B04	30	JR	NC,04<2BCA>	<899999.99
2B07	23	INC	HL	BC nur	2B06	E1	POP	HL	ok
2B08	03	INC	BC	pro Zeile dec	2B07	C3	JP	2904	
2B09	EDA0	LDI			2BCA	DD21	LD	IX,0B6E	sonst wandeln
2B0B	E2	JP	P0,2B27	wenn 10 Zeilen	2BCE	FDE3	EX	(SP),IY	in Ausg negativ
2B0E	18	JR	DB<2AEB>	Wdh	2BD0	E1	POP	HL	Stack korr
2B10	ED5B	LD	DE,(11FE)		2BD1	C3	JP	2B90	>U- Negativ
2B14	7B	LD	A,E	neue Zeile					=====
2B15	CE	ADC	A,40	wenn *					U- Nullen
2B17	5F	LD	E,A	DE +40					=====
2B18	30	JR	NC,04<2B1E>	mit Carry	2BE0	11	LD	DE,0866	Führende Nullen
2B1A	7A	LD	A,D		2BE3	0E	LD	C,05	weg
2B1B	C6	ADD	A,01		2BE5	1A	LD	A,(DE)	
2B1D	57	LD	D,A		2BE6	FE	CP	31	
2B1E	23	INC	HL		2BE8	D0	RET	NC	
2B1F	0B	DEC	BC		2BE9	3E	LD	A,20	
2B20	3E	LD	A,00		2BEB	12	LD	(DE),A	
2B22	B9	CP	C		2BEC	13	INC	DE	
2B23	20	JR	Z,02<2B27>		2BED	0D	DEC	C	
2B25	18	JR	C0<2AE7>	Wdh	2BEE	20	JR	NZ,F5<2BE5>	
					2BF8	C9	RET		

Speicherbereich

```

3000 - 3007 Ktostand PS           Je: HT ZT T
3008 - 300F Ktostand Bk           H 2 E
3010 - 3017 Ktostand Spar        1/10 1/100
3018 - 301F Ktostand Sparv
3020 - 3027 Ktostand Darl
3028 - 302F Ktostand HH
3030 - 303F Monatsname           Ende: 00
3040 - 31FF HH   Eingänge         Betrag stets
3200 - 3FFF HH   Ausgänge         8 Stellen "/"
4000 - 41FF PS   Eingänge         Text <35 Zchn
4200 - 45FF PS   Ausgänge         dann *
4600 - 47FF Bk   Eingänge
4800 - 48FF Bk   Ausgänge
4C00 - 4DFF Spar Eingänge
4E00 - 4FFF Spar Ausgänge
    
```

Bemerkungen:

PS, Bk, Spar u. HH können Beträge u. Text speichern, Sparvertrag u. Darl. nur Beträge. Bei Eingabe "Neuer Monat" werden Beträge und Text gelöscht, Kontostand bleibt erhalten. Eingabe: Betrag stets mit "Punkt" und Pfg

Bsp.: 1 Pfennig: 0.01
 Betrag ohne führende Blanks
 Einzelbetrag bis 99 999.99 DM
 Ktostände: bis +999 999.99 DM
 bis - 99 999.99 DM

Keine Sicherung gegen Speicherüberlauf!
 Programm -Drucken- (2000 - 20E8) nach eigenen Wünschen und je nach Drucker ändern!

Haushaltsbuchführung:

```

1000 3E 0C F7 EF 20 20 20 48 IE8
1008 20 61 20 75 20 73 20 68 I49
1010 20 61 20 8C 20 74 20 73 I54
1018 20 62 20 75 20 63 20 68 I4A
1020 20 66 20 75 20 65 20 68 I56
1028 20 72 20 75 20 6E 20 67 I74
1030 20 20 0D 00 3E 8A 86 30 I8E
1038 F7 05 20 FC 00 EF 50 6F I8E
1048 73 74 73 63 68 65 63 68 IAB
1048 6B 6F 6E 74 6F 20 20 20 IE3
1050 20 20 20 20 20 20 20 20 I60
1058 20 20 20 20 20 20 20 20 I68
1060 20 20 20 20 20 20 20 2E I7E
1068 20 2E 20 28 50 29 0D 00 I94
1070 EF 47 69 72 6F 68 6F 6E I48
1078 74 6F 20 20 20 20 20 20 I2B
1080 20 20 20 20 20 20 20 20 I90
1088 20 20 20 20 20 20 20 20 I98
1090 20 20 20 20 20 20 20 20 IAA
1098 20 20 2E 20 2E 20 28 47 IFA
10A0 29 0D 00 EF 50 6F 73 74 I7E
10A8 73 70 61 72 62 75 63 68 I10
10B0 20 20 20 20 20 20 20 20 I00
10B8 20 20 20 20 20 20 20 20 I08
10C0 20 20 20 20 20 20 20 20 I08
10C8 20 20 20 20 20 2E 20 2E IFA
10D0 20 28 53 29 0D 00 EF 53 IFA
10D8 70 61 72 76 65 72 74 72 I5E
10E0 61 67 20 28 20 20 20 20 I78
10E8 20 20 20 20 20 20 20 20 IFA
10F0 20 20 20 20 20 20 20 20 I80
10FB 20 20 20 20 20 20 20 20 I88
1100 2E 20 2E 20 28 56 29 0D I61
1108 00 EF 44 61 72 6C 65 68 I58
1110 65 6E 73 76 65 72 74 72 I9A
1118 61 67 20 20 20 20 20 20 I81
1120 20 20 20 20 20 20 20 20 I31
1128 20 20 20 20 20 20 20 20 I39
1130 20 20 2E 20 2E 20 28 I65
1138 44 29 0D 00 EF 48 61 75 I00
1140 73 69 61 6C 74 73 6B 6F IBA
1148 6E 74 6F 20 20 20 20 20 I4A
1150 20 20 20 20 20 20 20 20 I61
1158 20 20 20 20 20 20 20 20 I69
1160 20 20 20 20 20 20 2E 20 I7F
1168 2E 20 28 48 29 0D 00 EF I5C
1170 57 65 6C 63 68 65 73 20 I6C
1178 4B 6F 6E 74 6F 20 28 4B I27
1180 65 6E 6E 62 75 63 68 73 IE7
    
```

```

1328 F9 3E 00 32 40 30 32 00 I46
1330 32 03 00 10 6E 20 75 6D I88
1338 20 65 74 77 61 73 20 47 IF6
1340 3E 0C F7 CD 00 29 CD 00 I17
1348 28 FE 6E CA 34 14 FE 41 I40
1350 CA C2 13 FE 45 CA 70 13 I92
1358 3E 0C F7 21 00 30 CD 00 IAA
1360 28 CD 00 26 C3 20 29 E7 I66
1368 3A 95 41 31 2C 31 35 33 I81
1370 3E 0C F7 CD 00 2A 21 00 I0C
1378 40 11 00 2A ED 53 41 2B I82
1380 CD E1 2A DD 21 4A 08 DD I9B
1388 22 29 0C EF 4E 65 75 65 I6E
1390 20 45 69 6E 67 61 65 6E I7A
1398 67 65 20 3F 20 28 4A 2F I97
13A0 4E 29 00 85 DF 7B E1 E5 I2F
13A8 FE 4A 28 03 C3 58 13 CD I29
13B0 30 2A FD 21 07 30 0E 00 I80
13B8 CD 65 2A E1 CD 00 2A C3 I72
13C0 70 13 3E 0C F7 CD 50 20 I0F
13C8 21 00 42 11 50 2B ED 53 I0A
13D0 41 2B CD E1 2A DD 21 4A I6F
13D8 08 DD 22 29 0C EF 4E 65 I0C
13E0 75 65 20 41 75 73 67 61 I0E
13E8 65 6E 67 65 20 3F 20 28 I41
13F0 4A 2F 4E 29 00 85 DF 7B I32
13F8 E1 E5 FE 4A 28 03 C3 58 I5F
1400 13 CD 30 2A FD 21 07 30 IAA
1408 0E 00 DD 21 79 2A DD 36 I0E
1410 00 90 DD 36 02 91 DD 36 I6D
1418 0B 90 CD 45 2A DD 21 79 I9A
1420 2A DD 36 00 88 DD 36 02 I0E
1428 81 DD 36 0B 10 E1 CD 00 I49
1430 2A C3 C2 13 3E 0C F7 CD I76
1438 40 32 00 42 C3 00 10 00 I03
1440 3E 0C F7 CD 00 29 CD 00 I18
1448 28 FE 6E CA 34 15 FE 41 I42
1450 CA C2 14 FE 45 CA 70 14 I95
1458 3E 0C F7 21 00 30 CD 00 I83
1460 28 CD 00 2B C3 20 29 E7 I67
1468 3A 95 41 31 2C 31 35 33 I82
1470 3E 0C F7 CD 00 2A 21 00 I0D
1478 46 11 00 2A ED 53 41 2B I89
1480 CD E1 2A DD 21 4A 08 DD I9C
1488 22 29 0C EF 4E 65 75 65 I6F
1490 20 45 69 6E 67 61 65 6E I7B
1498 67 65 20 3F 20 28 4A 2F I98
14A0 4E 29 00 85 DF 7B E1 E5 I30
14A8 FE 4A 28 03 C3 58 14 CD I28
14B0 30 2A FD 21 07 30 0E 00 I89
14B8 CD 65 2A E1 CD 00 2A C3 I73
14C0 70 14 3E 0C F7 CD 50 20 I0F
14C8 21 00 40 11 50 2B ED 53 I11
14D0 41 2B CD E1 2A DD 21 4A I70
14D8 08 DD 22 29 0C EF 4E 65 I0D
14E0 75 65 20 41 75 73 67 61 I0F
14E8 65 6E 67 65 20 3F 20 28 I42
14F0 4A 2F 4E 29 00 85 DF 7B I33
14F8 E1 E5 FE 4A 28 03 C3 58 I60
1500 14 CD 30 2A FD 21 07 30 IAD
1508 0E 00 DD 21 79 2A DD 36 I0F
1510 00 90 DD 36 02 91 DD 36 I6E
1518 0B 90 CD 45 2A DD 21 79 I9B
1520 2A DD 36 00 88 DD 36 02 I0F
1528 81 DD 36 0B 10 E1 CD 00 I4A
1530 2A C3 C2 14 3E 0C F7 CD I78
1538 46 32 00 48 C3 00 10 00 I00
1540 3E 0C F7 CD 00 29 CD 00 I19
1548 28 FE 6E CA 34 16 FE 41 I44
1550 CA C2 15 FE 45 CA 70 15 I98
1558 3E 0C F7 21 10 30 CD 00 I8C
1560 28 CD 00 2B C3 20 29 E7 I68
1568 3A 95 41 31 2C 31 35 33 I83
1570 3E 0C F7 CD 00 2A 21 00 I0E
1578 40 11 00 2A ED 53 41 2B I80
1580 CD E1 2A DD 21 4A 08 DD I9D
1588 22 29 0C EF 4E 65 75 65 I70
1590 20 45 69 6E 67 61 65 6E I7C
1598 67 65 20 3F 20 28 4A 2F I99
15A0 4E 29 00 85 DF 7B E1 E5 I31
15A8 FE 4A 28 03 C3 58 15 CD I2D
15B0 30 2A FD 21 17 30 0E 00 I92
15B8 CD 65 2A E1 CD 00 2A C3 I74
15C0 70 15 3E 0C F7 CD 50 20 I03
15C8 21 00 4E 11 50 2B ED 53 I18
15D0 41 2B CD E1 2A DD 21 4A I71
15D8 08 DD 22 29 0C EF 4E 65 I0E
15E0 75 65 20 41 75 73 67 61 I0E
    
```

15E8 65 4E 67 65 20 3F 20 28 143
15F0 4A 2F 4E 29 00 E5 DF 7B 134
15F8 E1 E5 FE 4A 28 83 C3 58 161
1600 15 CD 30 2A FD 21 17 30 1B7
1608 0E 00 DD 21 79 2A DD 36 1E8
1610 00 90 DD 36 02 91 DD 36 16F
1618 0B 90 CD 65 2A DD 21 79 19C
1620 2A DD 36 00 88 DD 36 02 110
1628 81 DD 36 06 10 E1 CD 80 14B
1630 2A C3 C2 15 3E 00 32 00 17A
1638 4C 32 00 4E C3 00 10 00 1ED
1640 3E 00 F7 00 00 00 CD 00 164
1648 28 00 00 00 00 00 FE 41 1C5
1650 CA 91 16 FE 45 CA 70 16 16A
1658 0E 0C F7 21 18 30 CD E0 1C5
1660 28 CD E0 2B C3 20 29 E7 169
1668 3A 95 41 31 2C 31 35 33 184
1670 EF 0C 42 65 74 72 61 67 1D6
1678 20 3F 0D 00 DF 63 13 1A 169
1680 FE 20 20 FA EB FD 21 1F 1F6
1688 30 0E 00 CD 65 2A C3 58 153
1690 16 EF 0C 42 65 74 72 61 1A5
1698 67 20 3F 0D 00 DF 63 13 1D6
16A0 1A FE 20 20 F9 EB FD 21 110
16A8 1F 30 0E 00 DD 21 79 2A 18C
16B0 DD 36 00 90 DD 36 02 91 18F
16B8 DD 36 0B 90 CD 65 2A DD 185
16C0 21 79 2A DD 36 00 88 DD 112
16C8 36 02 81 DD 36 0B 10 C3 188
16D0 58 16 CD E0 2A DD 21 4A 173
16D8 0B DD 22 29 0C EF 4E 65 1CF
16E0 3E 0C F7 00 00 00 CD 00 184
16E8 28 00 00 00 00 00 FE 41 165
16F0 CA 31 17 FE 45 CA 10 17 14C
16F8 3E 0C F7 21 20 30 CD E0 16D
1700 28 CD E0 2B C3 20 29 E7 18A
1708 3A 95 41 31 2C 31 35 33 125
1710 EF 0C 42 65 74 72 61 67 177
1718 20 3F 0D 00 DF 63 13 1A 18A
1720 FE 20 20 F9 EB FD 21 27 19E
1728 30 0E 00 CD 65 2A C3 F8 194
1730 16 EF 0C 42 65 74 72 61 146
1738 67 20 3F 0D 00 DF 63 13 177
1740 1A FE 20 20 F9 EB FD 21 1B1
1748 27 30 0E 00 DD 21 79 2A 165
1750 DD 36 00 90 DD 36 02 91 1B0
1758 DD 36 0B 90 CD 65 2A DD 154
1760 21 79 2A DD 36 00 88 DD 1B3
1768 36 02 81 DD 36 0B 10 C3 129
1770 F8 16 00 20 2E 20 2E 20 151
1778 2E 20 2E 20 2E 20 2E 20 1C7

2000 3E 0F D3 86 3E CF D3 07 12D
2008 3E F5 D3 07 3E FF D3 05 14A
2010 CB 9F D3 05 CB DF D3 05 1F4
2018 EF 0C 44 72 75 63 68 61 18D
2020 72 74 2C 20 55 65 62 65 1F3
2028 72 73 63 68 72 69 66 74 1AD
2030 20 65 74 63 2E 20 64 61 18F
2038 6E 6E 20 22 50 22 00 00 102
2040 DF 7B F7 FE 5D 28 05 CD 106
2048 50 20 18 F4 C3 00 10 00 1B7
2050 D3 04 D8 05 CB 8F D3 05 159
2058 CB CF D3 05 D8 05 CB 67 1FC
2060 20 FA C9 30 38 2C 33 32 154
2068 30 30 3A 41 31 B4 89 28 129
2070 EF 20 4D 69 74 20 55 65 1A3
2078 62 65 72 73 63 68 72 69 1EA
2080 66 74 20 28 55 29 20 57 1B7
2088 69 65 76 69 65 6C 20 5A 1A8
2090 65 69 6C 65 6E 20 28 31 136
2098 30 29 20 00 DF 7B F7 11 193
20A0 90 00 FE 55 20 10 DF 7B 11D
20A8 F7 11 00 00 FE 30 30 02 138
20B0 3E 3A C6 02 18 06 FE 30 15C
20B8 30 02 3E 3A D6 30 47 21 1F8
20C0 0A 08 19 7E FE 00 20 8D 1B4
20C8 3E 0F 8D 6F 30 04 3E 01 1A4
20D0 84 67 3E 0D 05 CD 30 20 168
20D8 23 78 FE 00 20 E5 C9 65 1C4
20E0 06 01 21 4A 08 CD C3 20 12A
20E8 C3 18 20 20 53 46 24 00 1E8

2800 EF 0D 0D 57 6F 6C 6C 45 134
2808 6E 20 53 69 65 20 45 69 1AD
2810 6E 67 61 65 6E 67 65 20 12D
2818 62 75 63 68 65 66 20 28 1FD
2820 70 72 75 65 66 65 6E 29 164
2828 20 2E 20 2E 20 2E 20 2E 188
2830 28 45 29 00 EF 0D 57 6F 180
2838 6C 6C 65 6E 20 53 69 65 14C
2840 20 41 75 73 67 61 65 6E 14C
2848 67 65 20 62 75 63 68 65 163
2850 6E 20 28 70 72 75 65 66 150
2858 65 6E 29 20 2E 20 2E 20 138
2860 2E 20 2E 28 41 29 00 EF 185
2868 0D 57 6F 6C 6C 65 6E 20 12E
2870 53 69 65 20 64 65 6E 20 130
2878 6C 65 74 7A 74 65 6E 20 1C6
2880 4B 6F 6E 74 6F 73 74 61 1FB
2888 6E 64 20 2E 20 2E 20 2E 16C
2890 20 2E 20 2E 20 2E 20 4E 115
2898 29 00 EF 0D 47 65 62 65 158
28A0 6E 20 59 69 65 20 65 69 165
28A8 6E 65 6E 20 4B 65 6E 18D
28B0 62 75 63 68 73 74 61 62 124
28B8 65 6E 20 2E 20 2E 20 2E 19D
28C0 20 2E 20 2E 20 2E 20 3F 131
28C8 20 20 00 DF 7B FE 6E C8 18E
28D0 FE 45 C8 FE 41 C8 FE 4B 153
28D8 C8 3E 0C F7 C3 00 28 00 1F4
28E0 EF 0D 44 65 72 20 48 6F 1F9
28E8 6E 74 6F 73 74 61 6E 64 178
28F0 20 62 65 74 72 61 65 67 112
28F8 74 3A 20 44 4D 20 00 00 19F
2900 00 C3 C0 2B 00 06 06 7E 161
2908 F7 06 28 05 20 00 F9 3E 1D5
2910 F7 06 82 7E F7 00 23 05 1D5
2918 20 F9 C9 B4 30 A9 00 29 1D9
2920 EF 0D 0D 0D 41 20 30 20 110
2928 5A 75 72 2E 20 7A 75 6D 13C
2930 20 41 6E 67 65 62 6F 74 139
2938 20 28 0D 44 20 3D 20 4B 16A
2940 74 6F 2D 53 74 61 6E 64 173
2948 20 64 72 75 63 68 65 6E 17D
2950 0D 45 20 3D 20 50 72 6F 179
2958 67 72 61 6D 6D 20 62 65 17C
2960 65 6E 64 65 6E 0D 3F 20 1FF
2968 0D DF 7B FE 44 CC E0 20 1F9
2970 FE 45 28 0A FE 41 28 0B 180
2978 3E 0C F7 18 A3 00 EF 0C 198
2980 00 DF 5B C3 00 10 00 00 1B6
2988 00 00 00 00 00 00 00 00 181
2990 DD 21 4A 0B 0D 22 29 0C 140
2998 EF 44 28 3F 20 00 E5 DF 137
29A0 7B FE 44 CC 70 20 E1 EF 182
29A8 0C 00 C9 00 00 0F C9 C3 141
29B0 20 29 EF 0C 00 00 08 C3 11A
29B8 00 10 00 00 00 00 00 00 1F1
29C0 EF 57 6F 6C 6C 65 6E 20 169
29C8 53 69 65 20 65 69 6E 65 1D3
29D0 6E 20 6E 65 75 65 6E 20 1C2
29D8 4D 6F 6E 61 74 20 62 65 1E7
29E0 67 69 6E 6E 65 6E 20 2E 1D6
29E8 20 2E 20 2E 20 2E 20 6E 191
29F0 29 00 C9 8E 00 14 2A 57 12E
29F8 13 8A 20 50 48 B5 36 34 195
2A00 01 0E 00 21 30 30 11 1F 1EA
2A08 2A ED 80 EF 45 69 6E 67 168
2A10 61 65 6E 67 65 20 69 6D 138
2A18 20 4D 6F 6E 61 74 20 4D 1CE
2A20 61 65 72 7A 20 31 39 38 1BE
2A28 32 00 00 00 00 00 C9 00 14D
2A30 EF 0C 42 65 74 72 61 67 1AA
2A38 20 3F 0D 00 DF 63 05 0E 1F3
2A40 00 1A FE 2E 28 04 13 0E 1FB
2A48 18 F7 01 3E 04 91 4F 3E 1B4
2A50 20 77 23 0D 20 F9 05 EB 11A
2A58 ED A0 7E FE 20 20 F9 3E 102
2A60 24 12 13 01 C9 0D 21 09 174
2A68 00 2B 7E FE 00 20 02 3E 199
2A70 30 8D 30 47 FD 7E 00 D6 168
2A78 30 88 27 81 0E 00 27 FE 135
2A80 10 38 84 D6 10 0E 01 C6 1B1
2A88 30 FD 77 00 FD 2B 28 7E 127
2A90 FE 2E 20 01 28 DD 2B DD 117
2A98 7E 00 FE 00 20 CC 00 00 12A
2AA0 00 C9 C3 80 2B 00 2B 00 1AC
2AA8 00 00 00 00 00 00 00 00 1D2
2AB0 7E 23 FE 26 20 FA EF 54 1FC
2AB8 65 78 74 20 3F 0D 00 DF 17E

GEWINNER DES PREISAUSSCHREIBENS

Die Überschrift ist irreführend; wir haben noch keinen Gewinner für unser Grafikpreisausschreiben. Die Programme der Teilnehmer sind in diesem Heft abgedruckt (Herrn Mombaus Seeschlacht aus Heft 1-82 gehört auch dazu). Sie werden selbst feststellen, daß die Wahl sehr schwer fällt. Wir haben zwar unsere Favoriten, aber wollen uns doch nicht vorschnell entscheiden. Deshalb die Bitte; Helfen Sie uns bei der Wahl, und schicken Sie uns den Namen Ihres Favoriten. (auch z.B. wenn Sie eine Kleinanzeige aufgeben etc.). Im nächsten Heft veröffentlichten wir dann den endgültigen Gewinner. Er bedankt sich bei Ihnen für die "richtige" Wahl.

SPIELAUTOMAT

von Wolfgang Schröder

Das Programm wird bei 1000H gestartet und laeuft unter NASSYS mit einem Grafik-ROM (kann man aber auch ohne laufen lassen, sieht dann nur etwas anders aus).

Nach dem Start des Programmes erscheint oben auf dem Bildschirm der Guthabenzaehler, darunter die 3 'Rollen' und ganz unten die Anzeige fuer die Sonderspiele.

Um die Rollen zum Laufen zu bringen, geben Sie Ihren Einsatz in Pfennigen ein, also 500 fuer 5DM; anschliessend druecken Sie die CH-Taste (17H). Nun erscheint oben in der Anzeige Ihr Guthaben.

Pro Spiel wird dieser Betrag dann um 30Pf vermindert; sollte waehrend eines Spieles der Zaehler weniger als 30Pf anzeigen, so bleibt das Spiel stehen, und Sie muessen erst wieder erhoehen, wie oben beschrieben. Sie koennen nun, nachdem sich die Rollen drehen, diese beeinflussen, indem Sie eine beliebige Taste druecken, wenn unten am Bildschirmrand der Text START oder STOP erscheint. Die linke Walze kann dabei wieder gestartet und die mittlere und rechte angehalten werden.

Nachdem alle 3 Rollen stehen, wird ueberprueft, ob Sie etwas gewonnen haben, und Ihr Gewinn wird zu Ihrem Guthaben dazuaddiert.

Am Ende eines jeden Durchlaufes erscheint dann unten E/R; jetzt koennen Sie entweder Ihr Guthaben wie oben bechrieben erhoehen oder es sich 'auszahlen' lassen. Wenn Sie sich das Guthaben auszahlen lassen, so kehren Sie wieder zum Anfang der Programmes zurueck, andernfalls geht's gleich weiter.

Die einzige Moeglichkeit, aus dem Programm herauszukommen, ist die etwas unfeine Art des Drueckens der Reset-Taste.

Nun der Gewinnplan:

Befinden sich auf der rechten und linken Walze jeweils die gleichen Zahlen und in der Mitte entweder auch diese Zahl oder ein Joker, so wird Ihnen dieser Betrag zum Guthaben dazuaddiert.

Ein Joker allein in der Mitte gibt 30Pf.

Haben Sie auf jeder Rolle mindestens einen Joker, dann erhalten Sie 5 Sonderspiele und 3DM.

Fuenf Joker ergeben 10 Sonderspiele plus 3DM.

Bei den Sonderspielen wird Jeder Gewinn auf 3DM aufgerundet, und auch ein schraffiertes Feld ('*' in der Mitte) bringt 3DM ein.

Die Sonderspiele werden unten extra angezeigt, wobei folgende Besonderheiten gelten:

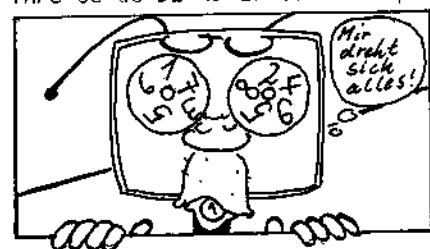
- a) Sonderspiel 1+10: Joker in der Mitte erhoehrt auf 20 Sonderspiele.
- b) Sonderspiel 5+6: Joker in der Mitte erhoehrt auf 10 Sonderspiele.
- c) Sonderspiel 14+17: wie a)
- d) Sonderspiel 15+16: Joker in der Mitte ergibt 100 Sonderspiele.
- e) Sonderspiel 3+13: Hier bleibt der Zaehler so lange stehen, bis Sie ein '*' in der Mitte haben.

Sollte Ihr Guthaben ueber 90DM anwachsen, so erhalten Sie 70DM 'ausgezahlt' und der Rest verbleibt im Guthabenzaehler.

1000	C3	3C	11	20	B6	20	20	38	6E
1008	30	32	31	30	32	35	30	32	A4
1010	31	30	20	36	30	20	38	30	8F
1018	20	36	30	32	30	30	31	35	A6
1020	30	20	B6	20	31	35	30	20	0C
1028	36	30	20	38	30	31	38	30	BF
1030	31	30	30	20	B6	20	20	36	1D
1038	30	32	35	30	20	38	30	20	B7
1040	36	30	20	B6	20	31	35	30	42
1048	20	38	30	32	31	30	31	38	DC
1050	30	31	32	30	20	38	30	32	DD
1058	31	30	32	30	30	31	30	30	EC
1060	20	36	30	20	B7	20	20	38	45
1068	30	20	36	30	20	38	30	32	E8
1070	30	30	31	30	30	31	35	30	07
1078	20	36	30	31	38	30	20	B7	7E
1080	20	31	35	30	32	31	30	20	F9
1088	38	30	31	32	30	32	35	30	2A
1090	20	36	30	31	30	30	31	35	1D
1098	30	32	31	30	32	35	30	32	34
10A0	30	30	20	36	30	20	38	30	1E
10A8	20	B5	20	20	36	30	20	38	8B
10B0	30	31	38	30	20	36	30	32	41
10B8	31	30	20	38	30	31	32	30	44
10C0	20	B5	20	31	35	30	32	35	C2
10C8	30	20	36	30	20	B5	20	20	A3
10D0	36	30	20	38	30	20	36	30	54
10D8	20	B5	20	31	30	30	32	31	D1
10E0	30	32	30	30	20	38	30	31	6B
10E8	38	30	32	31	30	20	38	30	7B
10F0	31	32	30	2D	11	20	38	30	59
10F8	14	D2	B0	13	C4	A8	14	32	63
1100	35	30	00	00	47	00	20	00	DD
1108	00	01	00	00	00	00	4C	0A	70
1110	00	30	30	20	44	4D	20	30	82

1118 30 20 50 66 00 00 30 30 8F	12C8 00 3A 05 11 FE 00 C2 77 61	1478 14 C3 13 15 3A 1C 11 C6 B8
1120 30 30 20 2A 20 2A 20 2A 6F	12D0 13 3A 10 11 FE 00 C2 B5 C5	1480 03 27 32 1C 11 C3 13 15 08
1128 20 2A 20 2A 20 2A 20 2A 61	12D8 14 CD 08 19 3A 05 11 FE 3A	1488 2A 0E 11 22 29 0C 3E 20 9A
1130 20 2A 20 2A 20 2A 20 2A 69	12E0 00 C2 5E 13 CD B4 16 3A F6	1490 F7 21 E1 09 22 29 0C 3E 3B
1138 20 2A 20 2A 3E 0C F7 3E 5C	12E8 08 11 21 07 11 B6 C2 FE C2	1498 31 F7 3A 1C 11 C6 03 27 2B
1140 30 32 1E 11 32 1F 11 32 76	12F0 12 3A E2 08 FE B7 C2 13 C2	14A0 32 1C 11 CD AD 18 DF 5D E1
1148 20 11 32 21 11 3E 00 32 5E	12F8 15 3E 03 32 08 11 3A 08 ED	14A8 3E FF 32 10 11 3E 00 32 BC
1150 1C 11 32 1D 11 32 05 11 36	1300 11 CB 27 C8 27 CB 27 CB C5	14B0 05 11 C3 13 15 3A 22 09 2A
1158 32 10 11 ED 5F E6 0D 06 C1	1308 27 21 1D 11 86 27 32 1D 8D	14B8 FE 2A CA D2 14 3A E2 08 C8
1160 01 47 3E 00 06 03 10 FC CC	1310 11 2B 3A 07 11 8E 27 32 98	14C0 FE B7 CA D2 14 CD B4 16 D0
1168 21 03 10 6F E5 DD E1 21 E0	1318 1C 11 DF 5D CD AD 18 C3 E9	14C8 21 08 11 3A 07 11 B6 CA E8
1170 22 11 22 F3 10 3E 00 32 49	1320 13 15 3A 1C 11 FE 00 C2 82	14D0 E0 14 3A 1C 11 C6 03 27 2F
1178 0A 11 32 0B 11 32 0C 11 41	1328 4B 13 3A 1D 11 FE 30 F2 21	14D8 32 1C 11 CD AD 18 DF 5D 19
1180 2A 33 10 7D 32 06 11 11 D5	1330 4B 13 21 60 0B 22 29 0C 84	14E0 21 E1 09 22 29 0C 3E 30 C4
1188 E0 0B 21 9A 11 01 08 00 59	1338 EF 45 2F 52 00 CF FE 45 12	14E8 F7 3A 10 11 FE FF CA F8 0D
1190 ED 80 3E 07 32 0D 11 C3 96	1340 CA 3C 15 FE 52 CA 4D 15 EA	14F0 14 3F DE 01 27 C3 FA 14 2E
1198 A2 11 44 4D 20 20 20 20 6D	1348 C3 3D 13 CD E5 18 CD 3A 3F	14F8 3E 99 32 10 11 CB 3F CB 0B
11A0 50 66 21 10 0A 22 29 0C F9	1350 16 CD AD 18 DF 5D 3E 07 8C	1500 3F CB 3F CB 3F F6 30 F7 85
11A8 3E B7 F7 EF 31 20 20 20 25	1358 32 0D 11 C3 1C 12 3A 05 EB	1508 3A 10 11 E6 0F F6 30 F7 8A
11B0 32 20 20 7F 33 20 20 20 45	1360 11 47 21 4C 0A 11 04 00 57	1510 C3 13 15 21 60 0B 22 29 E7
11B8 34 20 20 B7 35 20 20 B7 20	1368 19 10 FD 22 0E 11 22 29 2D	1518 0C EF 45 2F 52 00 11 00 FF
11C0 36 20 20 20 37 20 20 20 FE	1370 0C 3E 80 F7 C3 4D 14 3A A2	1520 18 DF 62 DA 2F 15 1B 7A 41
11C8 38 20 20 20 39 20 B7 31 B2	1378 E2 08 FE B7 C2 AD 13 3A E6	1528 B3 C2 21 15 C3 3F 15 FE FD
11D0 30 00 21 D0 0A 22 29 0C 63	1380 05 11 FE 01 CA 16 14 FE 9A	1530 45 CA 3C 15 FE 52 CA 4D 0C
11D8 EF 31 31 20 20 31 32 20 FD	1388 06 CA 33 14 FE 05 CA 33 B2	1538 15 C3 21 15 CD 44 18 21 A5
11E0 7F 31 33 20 B7 31 34 20 30	1390 14 FE 0A CA 16 14 FE 0E BF	1540 60 0B 22 29 0C EF 20 20 46
11E8 B7 31 35 20 B7 31 36 20 74	1398 CA 16 14 FE 0F CA 88 14 12	1548 20 00 C3 22 13 3E 0C F7 B6
11F0 B7 31 37 20 20 31 38 20 E9	13A0 FE 10 CA 88 14 FE 11 CA 00	1550 21 0B 08 22 29 0C 06 2B 21
11F8 20 31 39 20 20 32 30 00 35	13A8 16 14 C3 CF 13 3A 05 11 DA	1558 3E 3D F7 10 FD 21 8B 0B A3
1200 21 E1 09 22 29 0C EF 30 93	13B0 FE 03 CA 5C 14 FE 0D CA D3	1560 22 29 0C 06 2B F7 10 FD 01
1208 30 30 00 CD A0 17 CD 44 0F	13B8 5C 14 3A 22 09 FE 2A CA 92	1568 21 90 08 22 29 0C EF 2A A6
1210 18 CD 7C 17 CD E5 18 DF 43	13C0 CF 13 CD B4 16 21 08 11 86	1570 2A 2A 2A 2A 2A 20 20 42 D9
1218 5D CD AD 18 DF 5D CD F9 1B	13C8 3A 07 11 B6 CA DB 13 3A D5	1578 41 4E 4B 48 41 55 53 20 B8
1220 18 21 0D 11 CB 86 21 60 5B	13D0 1C 11 C6 03 27 32 1C 11 5F	1580 42 55 44 4E 41 53 45 20 B7
1228 0B 22 29 0C EF 53 54 41 73	13D8 CD AD 18 DF 5D 2A 0E 11 02	1588 20 2A 2A 2A 2A 2A 00 B9
1230 52 54 00 11 FF 02 DF 62 3B	13E0 22 29 0C 3E 20 F7 2B 2B F5	1590 21 4D 09 22 29 0C EF 5A BC
1238 DA 47 12 CD A0 17 1B 7A 96	13E8 2B 2B 3A 05 11 FE 0B C2 6C	1598 61 68 6C 65 6E 20 53 69 91
1240 B3 C2 36 12 C3 4C 12 21 51	13F0 F5 13 21 75 0A 22 0E 11 EC	15A0 65 20 67 65 67 65 6E 20 60
1248 0D 11 CB C6 21 60 0B 22 B7	13F8 22 29 0C 3E 80 F7 3A 05 56	15A8 64 69 65 73 65 6E 20 53 A8
1250 29 0C EF 20 20 20 20 20 26	1400 11 3D 32 05 11 FE 00 C2 6A	15B0 63 68 65 63 6B 00 21 D2 B6
1258 00 CD F9 18 21 60 0B 22 F6	1408 13 15 2A 0E 11 22 29 0C E4	15B8 09 22 29 0C EF 2D 2D 2D A3
1260 29 0C EF 53 54 4F 50 00 DC	1410 3E 20 F7 C3 13 15 2A 0E 9C	15C0 2D 2D 20 20 20 20 20 20 EF
1268 ED 5F F6 F0 5F 16 02 3E 61	1418 11 22 29 0C 3E 20 F7 21 0A	15C8 20 44 4D 20 2D 2D 2D 2D 62
1270 00 DF 62 DA 7F 12 CD A0 9B	1420 34 0B 22 0E 11 22 29 0C 0B	15D0 2D 00 21 8C 0A 22 29 0C 20
1278 17 1B 7A 83 C2 6F 12 21 4D	1428 3E 80 F7 3E 14 32 05 11 8B	15D8 EF 44 6F 65 73 65 6E 62 9C
1280 0D 11 CB 96 21 60 0B 22 BF	1430 C3 4D 14 2A 0E 11 22 29 FC	15E0 75 65 74 74 65 6C 2C 20 D4
1288 29 0C EF 20 20 20 20 00 3E	1438 0C 3E 20 F7 21 74 0A 22 6E	15E8 33 31 2E 20 46 65 62 72 2E
1290 CD F9 18 21 0D 11 CB 86 10	1440 0E 11 22 29 0C 3E 80 F7 7F	15F0 2E 20 31 39 38 30 00 21 46
1298 CD F9 18 21 60 0B 22 29 5F	1448 3E 0A 32 05 11 3A 1C 11 53	15F8 29 0B 22 29 0C EF 58 78 57
12A0 0C EF 53 54 4F 50 00 ED E0	1450 C6 03 27 32 1C 11 CD AD 2D	1600 78 78 78 78 78 00 3A 20
12A8 5F F6 F0 5F 16 02 DF 62 B7	1458 18 C3 13 15 3A 22 09 FE D2	1608 1C 11 F5 CB 3F CB 3F CB 1F
12B0 DA BC 12 CD A0 17 1B 7A 83	1460 2A CA CF 13 3A E2 08 FE 6C	1610 3F CB 3F F6 30 21 D8 09 97
12B8 B3 C2 AE 12 21 60 0B 22 AD	1468 B7 CA 7C 14 CD B4 16 21 45	1618 22 29 0C F7 F1 E6 0F F6 58
12C0 29 0C EF 20 20 20 20 20 96	1470 08 11 3A 07 11 B6 C2 7C E3	1620 30 F7 3E 2C F7 3A 1D 11 26

1628	CB 3F CB 3F CB 3F CB 3F 66	17D8	29 0C 2A F3 10 7E F7 21 E7	1988	11 01 77 FB 11 01 77 FB A9
1630	F6 30 F7 3E 30 F7 CF C3 5A	17E0	F3 10 34 21 E1 08 22 29 83	1990	11 01 77 FB 11 01 77 FB B1
1638	3C 11 3A 1C 11 DE 90 27 97	17E8	0C DD 7E 60 F7 DD 7E 61 79	1998	11 01 77 FB 11 00 00 00 46
1640	D8 06 99 0E FF 21 56 0B 5C	17F0	F7 DD 7E 62 F7 21 0D 11 F1	19A0	00 01 77 E6 8F 01 77 FB 19
1648	22 29 0C EF 20 20 20 20 24	17F8	CB 56 CA 2A 18 21 6B 08 D0	19A8	11 01 77 FB 11 01 77 FB C9
1650	20 20 20 20 20 20 20 20 66	1800	22 29 0C DD E5 01 90 00 C2	19B0	11 01 77 FB 11 01 77 FB D1
1658	20 20 20 20 20 20 20 20 6E	1808	DD 09 DD 7E 00 F7 DD 7E B3	19B8	11 01 77 FB 11 01 77 FB D9
1660	00 0D FF C2 61 16 0E FF C8	1810	01 F7 DD 7E 02 F7 21 AB 40	19C0	11 01 77 FB 11 01 77 FB E1
1668	21 56 0B 22 29 0C EF 41 87	1818	09 22 29 0C DD 7E 30 F7 12	19C8	11 00 00 00 00 01 77 FB 65
1670	55 53 5A 41 48 4C 55 4E 00	1820	DD 7E 31 F7 DD 7E 32 F7 3F	19D0	11 01 77 FB 11 01 77 FB F1
1678	47 20 20 37 30 2E 2D 20 F7	1828	DD E1 DD 23 DD 23 DD 23 FE	19D8	11 01 77 FB 11 01 77 FB F9
1680	44 40 00 0D FF C2 83 16 8E	1830	DD E5 E1 3E 30 BD C2 43 1B	19E0	11 01 77 FB 11 01 77 FB 01
1688	10 B9 CF AF 3A 1C 11 DE 2A	1838	18 DD 21 03 10 21 22 11 CD	19E8	11 01 77 FB 11 01 77 FB 09
1690	70 27 32 1C 11 21 56 0B 1E	1840	22 F3 10 C9 AF 32 FF 10 36	19F0	11 01 77 FB 11 00 00 00 9E
1698	22 29 0C EF 20 20 20 20 74	1848	32 00 11 32 01 11 32 02 1B	19F8	00 01 77 FB 11 01 77 FB 08
16A0	20 20 20 20 20 20 20 20 B6	1850	11 CF FE 17 CA 71 18 4F FF	1A00	11 01 77 FB 11 01 77 FB 22
16A8	20 20 20 20 20 20 20 20 BE	1858	06 04 FD 21 02 11 FD 7E 26	1A08	11 01 77 FB 11 01 77 FB 2A
16B0	20 20 00 C9 3E 00 32 07 46	1860	00 FD 77 01 FD 2B 10 F6 1B	1A10	11 01 77 FB 11 01 77 FB 32
16B8	11 32 08 11 32 0D 11 3A B4	1868	79 D6 30 FD 77 01 C3 51 88	1A18	11 01 77 FB 11 01 77 FB 3A
16C0	57 08 32 FF 10 3A 58 08 10	1870	18 FD 21 02 11 FD 7E 00 4C	1A20	11 00 00 00 00 01 77 FB BE
16C8	32 00 11 3A 59 08 32 01 EF	1878	CB 27 CB 27 CB 27 CB 27 58	1A28	11 01 77 FB 11 01 77 FB 4A
16D0	11 CD 3D 17 3A 0D 11 FE 6E	1880	FD 2B FD B6 00 21 1C 11 C1	1A30	11 01 77 FB 11 01 77 FB 52
16D8	00 CA F8 16 3A FF 10 FE 0D	1888	86 27 32 1C 11 FD 2B FD D1	1A38	11 01 77 FB 11 01 77 FB 5A
16E0	20 CA E9 16 D6 30 32 07 1E	1890	7E 00 CB 27 CB 27 CB 27 FC	1A40	11 01 77 FB 11 01 77 FB 62
16E8	11 3A 00 11 D6 30 32 08 9A	1898	CB 27 FD 2B FD B6 00 23 A0	1A48	11 01 77 FB 11 00 4C 2F 72
16F0	11 21 0D 11 CB CE CB 86 40	18A0	86 27 32 1D 11 3A 1C 11 2C	1A50	00 01 0B 00 00 00 00 00 76
16F8	3A 97 09 32 FF 10 3A 98 FB	18A8	CE 00 32 1C 11 3A 1C 11 54	1A58	00 00 60 84 00 00 80 00 D6
1700	09 32 00 11 3A 99 09 32 71	18B0	F5 E6 F0 CB 3F CB 3F CB 72	1A60	00 00 C0 00 00 00 80 00 BA
1708	01 11 CD 3D 17 3A 0D 11 AA	18B8	3F CB 3F F6 30 32 1E 11 A0	1A68	00 00 A0 00 00 00 C0 00 E2
1710	FE 00 C8 FE 02 C8 3A 07 F6	18C0	F1 E6 0F F6 30 32 1F 11 46	1A70	00 00 E0 00 00 00 80 00 EA
1718	11 21 FF 10 BE CA 21 17 30	18C8	3A 1D 11 F5 CB 3F CB 3F 51	1A78	00 00 90 00 00 00 A0 00 C2
1720	F0 23 3A 08 11 BE F0 3A 85	18D0	CB 3F CB 3F F6 30 32 20 74	1A80	00 57 43 00 01 10 00 00 45
1728	FF 10 FE 20 CA 34 17 D6 57	18D8	11 F1 E6 0F F6 30 32 21 60	1A88	00 00 00 00 00 00 81 00 23
1730	30 32 07 11 3A 00 11 D6 E2	18E0	11 CD 7C 17 C9 3A 1D 11 9A	1A90	00 00 82 00 00 40 82 00 EE
1738	30 32 08 11 C9 21 FF 10 C3	18E8	D6 30 27 32 1D 11 D0 3A 97	1A98	00 00 83 00 00 20 83 00 D8
1740	3A E2 08 FE B7 CA 53 17 64	18F0	1C 11 DE 00 27 32 1C 11 99	1AA0	00 40 83 00 00 60 83 00 60
1748	3A E1 08 BE C0 23 3A E2 3F	18F8	C9 ED 5F 5F 16 01 CD A0 08	1AA8	00 00 84 00 00 60 83 00 29
1750	08 BE C0 21 FF 10 3A 6B C2	1900	17 1B 7A B3 C2 FE 18 C9 19	1AB0	00 40 83 00 00 20 83 00 30
1758	08 BE C2 6B 17 23 3A 6C 42	1908	21 0D 11 36 00 3A E2 08 BA	1AB8	00 00 83 00 00 40 82 00 17
1760	08 BE C2 6B 17 21 0D 11 C0	1910	FE B7 C0 3A 58 08 FE B6 EC	1AC0	00 00 82 00 00 00 81 55 32
1768	CB C6 C9 21 FF 10 3A AB EE	1918	C2 1D 19 CB C6 3A 98 09 95	1AC8	53 2F 00 01 0B 00 00 00 70
1770	09 BE C0 23 3A AC 09 BE DE	1920	FE B6 C2 27 19 CB CE 7E 06	1AD0	00 00 00 00 00 00 00 00 EA
1778	00 C3 65 17 3A 1E 11 32 29	1928	FE 00 C8 3A 6C 08 FE B5 68	1AD8	00 00 00 00 00 00 00 00 F2
1780	11 11 3A 1F 11 32 12 11 78	1930	C2 35 19 CB D6 3A AC 09 E9	1AE0	00 00 00 00 00 00 00 00 FA
1788	3A 20 11 32 17 11 3A 21 BF	1938	FE B5 C2 3F 19 CB DE 7E 45	1AE8	00 00 00 00 00 00 00 00 02
1790	11 32 18 11 11 DD 0B 21 2D	1940	FE 0F C2 4B 19 3E 0A 32 06	1AF0	00 00 00 00 00 00 00 00 0A
1798	11 11 01 0A 00 ED B0 C9 42	1948	05 11 C9 FE 03 C8 E6 0C FB	1AF8	00 00 52 45 2F 00 01 0B E4
17A0	21 0D 11 CB 46 CA CC 17 B4	1950	C8 3E 05 32 05 11 C9 0C 91		
17A8	21 57 08 22 29 0C DD 7E F1	1958	C8 3E 0E 0F 23 7C B5 28 10		
17B0	00 F7 DD 7E 01 F7 DD 7E 6C	1960	02 2B E9 3E 04 C3 30 1B DF		
17B8	02 F7 21 97 09 22 29 0C E0	1968	7B 32 09 0F C9 7B 32 08 C4		
17C0	DD 7E 30 F7 DD 7E 31 F7 DC	1970	11 00 00 00 00 01 77 E8 FA		
17C8	DD 7E 32 F7 21 0D 11 CB 6D	1978	8F 01 77 FB 11 01 77 FB 17		
17D0	4E CA F5 17 21 22 09 22 79	1980	11 01 77 FB 11 01 77 FB 1A1		



BOWLING

von Klaus Stehlik

```
2 REM      Nach einem Programm aus
3 REM      ' The First Book of 80-US '
4 REM      Umgeschrieben und modifiziert von
5 REM      C l a u s   S T E H L I K
7 DIM A(50):CLS
15 GOSUB 5000:GOSUB 3000:GOSUB 3030
17 FOR I=1 TO 50:A(I)=0:NEXT
20 FORX=0T095:SET(X,21):SET(X,43):SET(X,13):NEX
T
25 FOR Y=14 TO 20:SET(0,Y):SET(95,Y):NEXT
30 FOR X=14 TO 86 STEP8:FOR Y=14 TO 20
35 SET (X,Y):NEXT Y:NEXT X
40 SCREEN 10,4:PRINT"1  2  3  4  5  6  7
  8";
45 PRINT"  9  10";
50 SCREEN 2,7:PRINTA$,:GOSUB 3000
60 I=0:L=0:F=0
70 B=0
75 IF F<10 THEN F=F+1
100 RESTORE
110 READ X,Y
120 IF X<0 GOTO 150
130 SET(X,Y)
140 GOTO 110
150 D=2:X=2:Y=24
155 I=I+1
160 C=USR(0):IF C=32 THEN 200
170 IF Y+D<24 OR Y+D>40 THEN D=-D
180 RESET(X,Y):Y=Y+D:SET(X,Y)
190 GOTO 160
200 D=2
205 IF X+D>95 THEN 320
210 IFPOINT(X+D,Y-1)=1ORPOINT(X+D,Y)=1ORPOINT(X
+D,Y+1)=1 THEN 300
220 RESET (X,Y)
230 X=X+D:IF X>95 GOTO 320
235 IF G=H AND X>50 THEN Y=INT(Y-.7499+RND(1))
240 IF G=H OR X<50 GOTO 250
242 IFINT(RND(1)*10+1)=5THENY=INT(Y+RND(1)):GOT
0250
245 IFINT(RND(1)*10+1)=2THENY=INT(Y-RND(1)):GOT
0250
247 IF INT(RND(1)*19+1)=9 THEN Y=INT(Y)
250 IF Y<23 THEN Y=23
255 IF Y>41 THEN Y=41
260 SET(X,Y)
270 GOTO 205
300 RESET (X,Y)
304 Y=INT(Y)
305 X=X+D:R=0
307 IFPOINT(X-1,Y-1)ORPOINT(X-1,Y)ORPOINT(X-1,Y
+1)=1THENX=X-1
310 FORK=XT095STEP9:R=R+3:FORJ=Y-RTOY+R
312 IFJ<41ANDJ>23THENRESET(K,J):RESET(K+1,J)
315 NEXT J:NEXT K
320 B=B+1
330 GOSUB 1000
340 IF A(I)=10 AND B=1 THEN A(I+25)=2:GOTO 400
345 IF A(I)=10 AND B=2 THEN A(I+25)=1
350 IF B=2 THEN A(I)=A(I)-A(I-1):GOTO 400
360 IF I<20 GOTO 400
390 GOTO 150
400 GOSUB 2000:A(0)=0:FOR J=1 TO I
401 IF J<20 GOTO 412
402 IF A(J+25)=1 THEN A(0)=A(0)+A(J+1)
403 IF A(J+25)<=2 GOTO 410
405 A(0)=A(0)+A(J+2)
407 IF A(J+27)-2THENA(0)=A(0)+A(J+4):GOTO 410
408 A(0)=A(0)+A(J+3)
410 IF J<21 THEN A(0)=A(0)+A(J)
412 NEXT J
415 IF A(I+25)=2 THEN I=I+1
417 IF F=10 THEN SCREEN44,7:PRINTA(0):GOTO425
420 SCREEN(INT(I/2)-1)*4+8,7:PRINTA(0);
425 IF I=20 AND A(I+24)=2 GOTO 70
426 IF I=21 AND A(I+23)=2 GOTO 70
427 IF I=22 AND A(I+24)=2 GOTO 70
430 IF I<19 AND A(I+25)=0 GOTO 440
432 IF I<21 GOTO 70
440 SCREEN 47,15
450 IF A(0)>150 THENPRINT"  Nicht schlecht ":G
0T0470
460 PRINT
470 INPUT"  Noch ein Spiel ":R$
475 CLS:IF LEFT$(R$,1)<="J" THEN END
490 A1=A1+A(0)
500 A2=A2+1
510 PRINT"  "A$;" ,Dein Durchschnitt ist ";A1/A
2
520 GOTO 17
1000 A(I)=0:RESTORE
1010 READ X,Y,W,Z
1020 IF X=-1 THEN RETURN
1030 IF POINT(X,Y)=1 GOTO 1010
1040 A(I)=A(I)+1:GOTO 1010
2000 IFA(I+25)=2THENSREEN(F-1)*4+10,6:PRINT"X"
;:RETURN
2010 IFA(I+25)=1THENSREEN(F-1)*4+10,6:PRINT"/"
;:RETURN
2020 SREEN(F-1)*4+10,6:PRINT"-":RETURN
3000 T$="B O W L I N G"
3010 FOR K=1 TO LEN(T$)
3020 POKE 3036+K,ASC(MID$(T$,K,1)):NEXT:RETURN
3030 PRINT:PRINT
3040 PRINT" Willkommen beim B o w l i n g.":PRI
NT
3050 INPUT" Wie sind Deine Initialen (max.5)":A
$
```

```

3060 H=1:S=2
3070 PRINT" Willst Du einen Kurvenball (K)
3080 INPUT" oder Geradenball (G) werfen ";G$
3090 IF LEFT$(G$,1)="K" THEN G=H:GOTO 3110
3100 G=S
3110 PRINT:PRINT" Wenn das Spiel beginnt,bewegt
sich die
3120 PRINT" Kugel an der Bahn auf und ab.Um die
Ku-
3130 PRINT" gel zu werfen,druecke die 'Space'-T
aste.
3140 PRINT:PRINT
3150 PRINT" Druecke eine Taste um zu beginnen."
3160 C=USR(0):IF C<0 THEN 3160
3170 CLS:RETURN
4000 DATA 65,32,66,32,74,35,75,35
4010 DATA 74,29,75,29,83,27,84,27,83,32,84,32
4020 DATA 83,37,84,37,92,24,93,24,92,29,93,29
4030 DATA 92,35,93,35,92,40,93,40,-1,-1,-1,-1
5000 RESTORE 5010
5010 DATA 25055,1080,-53,536,-20665
5020 DATA 2602,-5664,0
5030 DOKE 4100,3328:FOR I9=3328 TO 3342 STEP2
5040 READ J9:DOKE I9,J9:NEXT
5050 RETURN

```

* oder 3370 je nach BASIC Version Red.

SPACEPOTATOES

von Claus Stehlik

```

5 REM      Frei nach einem Programm aus
6 REM      80 Microcomputing 8/81
7 REM      Umgeschrieben,modifiziert etc.von
8 REM      C l a u s   S T E H L I K
10 GOSUB 1010:GOSUB 2000
30 DIM MV$(4),Z9(14,2),T$(20):GOSUB 940
40 CLS:SCREEN15,5:INPUT" Difficulty (1-10) ";DI
50 Y0=1:Y9=13:X0=1:X9=20:RF=0:OD=0:OF=28
60 MV$(1)="Y":MV$(2)="N":MW$(3)="G":MV$(4)="K"
65 T$(2)="Time":T$(4)="Hits":T$(6)="Miss"
66 T$(8)="Fuel":T$(10)="Damage"
70 C$=""
80 GOSUB 190
90 GOSUB 720
100 A=USR(0):IF A<=0 THEN M$=" ":GOTO 103
101 M$=CHR$(A)
103 IF M$=" " THEN M$=C$
105 IF M$<>" " THEN F9=F9-G1
106 IF M$<>"F" THEN C$=M$
110 IF M$="F" THEN GOSUB 320:GOTO 170
120 IF M$="Y" THEN Y3=1:GOSUB 350:GOTO 170
130 IF M$="N" THEN Y3=-1:GOSUB 350:GOTO 170
140 IF M$="G" THEN X3=1:GOSUB 350:GOTO 170
150 IF M$="J" THEN X3=-1:GOSUB 350

```

```

170 GOSUB 460
180 GOTO 100
190 FOR I=2058 TO 3001:POKEI,128:NEXT
195 SCREEN1,1
200 FOR I=1T014:PRINT"                ":NEXT
210 FOR I=2066T02194STEP64:POKEI,148:NEXT
220 FOR I=2834T02962STEP64:POKEI,148:NEXT
230 SCREEN 1,8
240 PRINT" _____"
250 FORI=2T014STEP2:SCREEN26,I:PRINTT$(I):NEXT
260 SCREEN21,9:FORI=1T027:PRINTCHR$(210);:NEXT
265 PRINT
270 SCREEN21,11:FORI=1T027:PRINTCHR$(210);:NEXT
280 T9=500:F9=27:D9=0:H9=0:M9=0
285 C9=INT(3*RND(1)*D1)
290 F=0:MF=0:SF=0:A9=140:G1=.05:LF=0
300 GOSUB 500:RETURN
310 P1=2058+Y1*64+X1:RETURN
320 REM ** Fire
330 GOSUB 910:IFX1=8 AND Y1=7 THEN GOSUB 670
335 IF X1<=14 AND Y1<=7 THEN GOSUB 790
340 RETURN
350 IF SF=0 THEN POKE P1,B9
360 IF X1+X3 <=2 OR X1+X3 >= 22 THEN 370
365 X1=X1+X3
370 IF Y1+Y3 <=2 OR Y1+Y3 >= 16 THEN 380
375 Y1=Y1+Y3
380 X3=0:Y3=0
390 IF((Y1>Y9ORY1<Y0)OR(X1>X9ORX1<X0))THENSF=1
395 IF SF=1 THEN RETURN
400 SF=0
410 GOSUB 310
420 B9=PEEK(P1)
430 IF B9=140 THEN A9=136
440 POKE P1,A9
450 RETURN
460 IFHF=1 THEN IF C9<0 THEN GOSUB 720:GOTO500
465 IF HF=1 THEN IF C9<0 THEN LF=2:GOTO 500
470 IF MF=1 THEN TF=2:GOSUB 840:GOTO 500
480 IF SF=1 THEN TF=1:GOSUB 840:GOTO 500
490 REM ** Update Screen
500 A9=140:T9=T9-1:IF T9<=0 THEN LF=1
510 IF D9<=28 THEN LF=1
520 IF F9<=0 THEN LF=1
530 IF LF<=0 THEN 890
540 SCREEN 32,2:PRINT T9
550 SCREEN 32,4:PRINT H9
560 SCREEN 32,6:PRINT M9
570 IF INT(OF)=INT(F9)THEN 590
580 SCREEN(INT(20+OF)),9:PRINTCHR$(128):OF=F9
590 IF INT(OD)=INT(D9)THEN 610
600 SCREEN(INT(20+D9)),11:PRINTCHR$(128);:OD=D9

```

```

610 GOSUB 860:HF=0:MF=0
620 IF RF=0 THEN 650
630 IF D9<22 OR F9<5 OR T9<75 THEN RF=1
640 RETURN
650 IF RF=1 THEN SCREEN5,1:PRINT" = RED ="
660 IF RF<=1 THEN SCREEN5,1:PRINT"      "
665 RF=1:RETURN
670 REM ** Hit
680 H9=H9+1
690 C9=C9-1
700 FORI=128TO191:POKE2514,I:NEXTI:POKE2514,32
710 HF=1:RETURN
720 REM ** Place craft
730 X1=INT(RND(1)*16+4):Y1=INT(RND(1)*10+2)
740 GOSUB 310
750 B9=PEEK(P1)
755 IF P1<30000 OR P1<2058 THEN STOP
760 POKE P1,A9
770 C$=MV$(INT(RND(1)*4))
780 RETURN
790 REM ** Miss
800 M9=M9+1
810 MF=1
820 RETURN
830 REM ** Hit back
840 D9=D9+(INT(RND(1)*3)+TF)/5:SF=0
850 RETURN
860 IF INT(RND(1)*10)<=1 THEN RETURN
870 C$=MV$(INT(RND(1)*4))
880 RETURN
890 SCREEN 26,13:IF LF=1 THEN PRINT" You loose"
895 IF LF=1 THEN 900
897 PRINT" You win ";
900 SCREEN 38,13:INPUT"Again";A$
902 IF LEFT$(A$,1)="J"ORLEFT$(A$,1)="Y"THEN 40
905 CLS:END
910 FOR I=1TO14:POKEZ9(I,1),Z9(I,2):NEXTI
915 POKE 2514,246
920 FOR I=1TO14:POKEZ9(I,1),32:NEXT I
925 POKE 2514,32
930 RETURN
940 REM ** Fire trail
950 L=2954:R=2970
960 FOR I=1 TO 14 STEP 2
970 Z9(I,1)=L:Z9(I+1,1)=R
980 Z9(I,2)=207:Z9(I+1,2)=249
990 L=L-63:R=R-65:NEXT I
1000 RETURN
1010 DATA 25055,1080,-53,536,-20665
1020 DATA 2602,-5664,0
1030 DOKE 4100,3328:FOR I9=3328 TO 3342 STEP 2
1040 READ J9:DOKE I9,J9:NEXT
1050 RETURN
2000 CLS:T$="S P A C E   P O T A T O E S"

```

```

2010 FOR K=1 TO LEN(T$)
2020 POKE 3027+K,ASC(MID$(T$,K,1)):NEXT
2030 PRINT
2040 PRINT" Deine Aufgabe ist es,moeglichst alle
2050 PRINT" feindlichen Raumschiffe zu zerstoeren.
2060 PRINT" Mit Hilfe der Tasten um H (Y,J,N,G)
2070 PRINT" steuerst Du Dein eigenes Schiff (hinauf,
2080 PRINT" rechts,hinunter,links).Wenn ein gegnerisches
2090 PRINT" Raumschiff im Fadenkreuz ist,
2100 PRINT" kannst Du Deine Laser mit 'F' abfeuern.
2110 PRINT
2120 PRINT"                Achtung !
2130 PRINT" Wenn der Feind ausser Sicht ist,beginnt
2140 PRINT" er zurueckzuschieszen !
2150 FOR I=2 TO14:SCREEN 45,I:PRINT"u":NEXT
2160 A=USR(0):IF A<0 THEN 2160
2170 RETURN
Ok
Achtung: Zeile 1020 Data für die versch. BASIC Versionen 3370 oder 2602 Red.

```

Großschrift von Jürgen Loh

Das Programm laeuft auf NASCOM 1 mit NAS-SYS und "Simple Grafik". Es handelt sich um eine Laufschrift... das ist eigentlich nichts so tolles. Eine normale Schrift ist es auch nicht, es werden naemlich Zeichen verwendet, die 5 normale Zeichen gross sind! Davon passen dann auch nur 2 Zeilen untereinander auf den Bildschirm. Als Anwendung dieser "Riesenschrift" koennte man sich zum Beispiel Werbezwecke denken. Aber auch so ist sie recht schoen anzusehen.

Nun zum Programm. Es teilt sich in 3 verschiedene Teile auf:

- 1. Das Programm zur Darstellung der Buchstaben und Zeichen auf dem Bildschirm,
 - 2. Eine Tabelle, in der die Zeichen festgelegt sind, und
 - 3. Ein Hilfsprogramm zum Ausprobieren.
- Der Zeichensatz erstreckt sich von 20H bis 5FH. Zusaetzlich werden noch die Codes 0CH (Bildschirm loeschen) und 0DH (neue Zeile)

akzeptiert. ODH wird dabei zum Umschalten zwischen den beiden Zeilen benutzt (Ausprobieren!). Alle anderen Codes werden ignoriert.

Der erste Programmteil besitzt 2 Einsprungsadressen. Ein Unterprogrammaufruf (CALL) bei 0C80H gibt das Zeichen, das im Accumulator steht, aus. Es entspricht in seiner Funktion der ROUT - Routine des Monitors, nur eben fuer die Riesenbuchstaben. Ein CALL bei 0CDAH wirkt genauso wie die PRS - Routine (RST 28H), d.h. hinter dem CALL muss die auszugebende Zeichenkette, abgeschlossen mit 00H, folgen.

Der dritte Programmteil hat ebenfalls zwei Einsprungsadressen. Mit E 0EB8 vom Monitor aus kann man den ersten Programmteil starten. Es erwartet eine Eingabe von der Tastatur und gibt dann das Zeichen in Riesenschrift auf dem Bildschirm aus. Mit der Taste GRAPH - NEW LINE und einer anschliessend gedruckten Zahlentaste kann man den Abstand zwischen den einzelnen Zeichen veraendern. Mit ESC (SHIFT - NEW LINE) kann man das Programm verlassen. E 0ED8 vom Monitor aus gibt eine Laufschrift aus. Ruecksprung in den Monitor ist nur mit RESET moeglich.

Aenderungen fuer T2/T4

In 0C81H muss 1E, in 0C84 muss CD 3B 01 und in 0C8F muss 1F. Die Demonstrationsprogramme ab 0EB8 muessen neu geschrieben werden.

Aenderungen fuer NASCOM 2

Das Programm verwendet 0AH (Reverses Blank)* als Grafikzeichen NA 2 - Benutzer muessen das in FFH umaendern, und zwar an der Adresse 0CD0.

Das Programm belegt im RAM noch die Adressen 0D35H bis 0D38H. In 0D35H und 0D36H ist die Adresse des Bildschirms gespeichert, an der das naechste Zeichen ausgegeben wird. Sie wird nach jedem NEW LINE oder CLEAR SCREEN automatisch neu gesetzt. In 0D37H steht der Wert der Verzoegerung, die waehrend der Ausgabe eines Zeichens eingefuegt wird, und in 0D38H steht die Anzahl der Zwischenraeume zwischen zwei Zeichen. Die letzten beide Speicherstellen muessen, bevor der erste Programmteil aufgerufen wird, auf den gewuenschten Wert + 1 gesetzt werden. Beispiel: die Befehlsfolge LD HL,0802 ; LD (0D37),HL setzt die Verzoegerung auf 07 und die Zwischenraumbreite auf 01.

Zum Schluss noch die Adressen, die beim Ver-

schieben des Programms geaendert werden muessen:

0C8B 0C93 0CA5 0CB2 0CB8 0CC5 0CD5 0CE4 0CEE 0D06 0D14

0CA5, 0CB8 und 0CE4 sind CALLs, in 0D14 steht der Tabellenanfang (0D39).

In allen anderen Adressen wird auf die 4 RAM - Zellen zugegriffen.

* Anm. d. Red.: Das funktioniert nur, wenn der Hardware eine Invertierung der Zeichen durch Bit 7 vorgesehen ist (wie in älteren Ausgaben des Journals mehrfach beschrieben).

0C80	FE	0C	20	0A	F7	00	00	21	D8
0C88	79	08	22	35	0D	C9	FE	0D	4D
0C90	20	12	2A	35	0D	7D	FE	39	EE
0C98	20	05	21	79	08	18	EB	21	8F
0CA0	39	0A	18	E6	CD	0F	0D	D8	AE
0CA8	7E	47	E6	FE	FE	FE	78	20	F1
0CB0	0B	3A	38	0D	4F	0D	C8	CD	37
0CB8	E9	0C	18	F9	23	E5	4F	06	27
0CC0	07	11	40	00	2A	35	0D	CB	5B
0CC8	19	38	04	36	20	18	02	36	CF
0CD0	FF	19	10	F3	CD	E9	0C	E1	9A
0CD8	18	CE	E3	7E	23	B7	20	02	27
0CE0	E3	C9	E5	CD	80	0C	E1	18	CF
0CE8	F2	06	07	3E	20	2A	35	0D	BD
0CF0	11	D2	FF	19	C5	54	5D	1B	88
0CF8	01	2F	00	ED	B0	12	11	11	05
0D00	00	19	C1	10	EF	3A	37	0D	64
0D08	47	AF	05	C8	FF	18	F8	0E	F8
0D10	20	B9	D8	21	39	0D	91	4F	15
0D18	06	FF	7E	FE	FF	20	0D	23	F5
0D20	04	79	B8	20	F5	7E	FE	FF	F2
0D28	28	06	AF	C9	FE	FE	20	02	F9
0D30	37	C9	23	18	E5	39	0A	01	A1
0D38	02	FF	00	00	00	00	FF	17	5C
0D40	17	FF	03	03	00	03	03	FF	6E
0D48	0A	1F	0A	1F	0A	FF	02	17	C9
0D50	15	1F	15	1D	08	FF	11	09	E4
0D58	04	12	11	FF	1C	10	1C	00	D3
0D60	10	FF	05	03	FF	0E	11	11	B3
0D68	FF	11	11	0E	FF	11	0A	04	C2
0D70	1F	04	0A	11	FF	04	04	0E	DD
0D78	0E	04	04	FF	28	18	FF	04	DD
0D80	04	04	04	04	04	FF	10	10	CC
0D88	FF	10	08	04	02	01	FF	1F	D1
0D90	1F	11	11	1F	1F	FF	02	1F	3C
0D98	1F	FF	13	19	15	13	FF	11	27
0DA0	15	1F	1F	FF	0C	0A	09	1F	3D
0DA8	08	FF	17	17	15	1D	09	FF	24
0DB0	1F	1F	15	15	1D	1D	FF	01	5F
0DB8	11	19	0D	07	03	FF	0A	1F	2E

```

ODC0 15 15 1F 0A FF 17 17 15 62
ODC8 15 1F 1F FF 0A 0A FF 2A 64
ODD0 1A FF 04 0A 11 FF 0A 0A 28
ODD8 0A 0A 0A FF 11 0A 04 FF 20
ODE0 01 15 15 02 FF 1F 11 15 5E
ODE8 15 13 FF 1E 1F 05 05 1F 82
ODF0 1E FF 1F 1F 15 1F 0E FF 99
ODF8 0E 1F 11 11 FF 1F 1F 11 A2
OE00 11 1F 0E FF 1F 1F 15 15 B3
OE08 11 FF 1F 1F 05 05 01 FF 6E
OE10 1F 1F 11 15 1D 1D FF 1F DA
OE18 1F 04 04 1F 1F FF 1F 1F C8
OE20 FF 19 11 1F 1F FF 1F 1F D2
OE28 04 0E 1B 11 FF 1F 1F 10 C1
OE30 10 FF 1F 1F 02 04 02 1F B2
OE38 1F FF 1F 1F 02 04 08 1F CF
OE40 1F FF 0E 1F 11 11 1F 0E E8
OE48 FF 1F 1F 05 05 07 07 FF AA
OE50 0E 1F 11 19 1F 0E FF 1F 00
OE58 1F 05 0D 1D 17 FF 02 17 E3
OE60 15 15 15 1D 08 FF 01 01 D3
OE68 1F 1F 01 01 FF 0F 1F 10 F3
OE70 10 1F 0F FF 07 0F 18 10 F9
OE78 18 0F 07 FF 1F 1F 08 04 FD
OE80 08 1F 1F FF 11 1B 0E 04 11
OE88 0E 1B 11 FF 03 07 1C 1C 11
OE90 07 03 FF 19 15 13 FF 1F 06
OE98 11 11 11 FF 01 02 04 08 E7
OEA0 10 FF 11 11 11 1F FF 04 12
OEA8 02 1F 1F 02 04 FF 10 10 1B
OEB0 10 10 10 10 FE 00 00 00 FC
OEB8 21 01 02 22 37 0D 3E 0C 9A
OEC0 CD 80 0C CF FE 1B 20 02 31
OEC8 DF 5B FE 8D 20 F2 CF E6 62
OED0 0F 3C 32 38 0D 18 EC 00 A4
OED8 21 06 02 22 37 0D 3E 0C BF
OEE0 CD 80 0C CD DA 0C 20 44 5E
OEE8 45 4D 4F 4E 53 54 52 41 5F
OEF0 54 49 4F 4E 20 45 49 4E 34
OEF8 45 52 20 4C 41 55 46 53 38
OF00 43 48 52 49 46 54 20 49 38
OF08 4E 20 52 49 45 53 45 4E 4B
OF10 42 55 43 48 53 54 41 42 6B
OF18 45 4E 20 20 20 2A 2A 2B 99
OF20 2B 2A 2A 0D 00 18 BC 00 8F

```

Doppelwurm

von Michael Bach

Für folgendes Programm ist die NASCOM 2 Grafik erforderlich. Die Steuerung des Spiels erfolgt über die Tastatur (die entsprechenden 2*4 Tasten werden im Spiel angegeben) oder besser über 2 Knüppel an Port A (Neutralstellung=high). Zur Spielwiederholung kann die Eingabe durch 2 mal Leertaste verkürzt werden. Interessant sind die Unterprogramme "SET, RESET und POINT", die wir vom BASIC kennen; allerdings hier als Maschinenprogramme viel schneller in der Ausführung.

Ein Listing der Graphic - Routinen wurde bereits in Heft 2/82 veröffentlicht.

```

OC80 CD AA 0E DD 21 18 0C DD 10
OC88 36 02 12 EF 57 69 65 76 68
OC90 69 65 6C 65 20 53 70 69 87
OC98 65 6C 65 72 20 28 31 2C F1
OCA0 32 29 3F 0D 00 DF 7B F7 A4
OCA8 DD CB 00 86 FE 31 28 08 41
OCB0 FE 32 20 D7 DD CB 00 C6 51
OCB8 DF 6A EF 53 74 65 75 65 02
OCC0 72 75 6E 67 20 7D 62 65 EC
OCC8 72 20 4B 6E 7D 70 70 65 E1
OCD0 6C 20 61 6E 20 50 6F 72 88
OCD8 74 20 30 20 28 4B 29 0D 71
OCE0 6F 64 65 72 20 54 61 73 DE
OCE8 74 61 74 75 72 3A 20 57 D5
OCF0 41 5A 53 20 40 3B 2F 3A EE
OCF8 20 28 54 29 0D 00 DF 7B 30
OD00 DD CB 00 8E FE 4B 28 08 BC
OD08 FE 54 20 AC DD CB 00 CE A9
OD10 CD AA 0E FD 21 00 00 DD 9D
OD18 36 01 10 DF 6A EF 47 65 50
OD20 73 63 68 77 69 6E 64 69 86
OD28 67 6B 65 69 74 20 28 31 C2
OD30 2D 39 29 3F 20 0D 00 DF 17
OD38 7B F7 FE 20 28 12 FE 31 3E
OD40 38 D9 FE 3A 30 D5 D6 30 A1
OD48 47 3E 0A 90 17 DD 77 02 E1
OD50 CD AA 0E 21 00 00 06 2C 35
OD58 CD 09 0F 24 10 FA 06 5F DD
OD60 CD 09 0F 2C 10 FA 06 2C BA
OD68 CD 09 0F 25 10 FA 06 5F EE
OD70 CD 09 0F 2D 10 FA 3E CF A6
OD78 D3 06 3E FF D3 06 21 4A DF
OD80 17 11 00 02 D9 21 14 15 DA
OD88 11 00 01 FD 23 CD 72 0E 14
OD90 DD 35 01 20 07 CD 1F 0F D2
OD98 DD 36 01 10 CD 1F 0E CD 90
ODA0 59 0E CD 39 0F 38 23 CD 51
ODA8 09 0F DD CB 00 46 28 DB BE
ODB0 D9 D7 6C 1F 1F 1F 1F CD 22

```



```

QDB8 59 0E CD 39 0F 38 06 CD 4C
QDC0 09 0F D9 18 C6 EB 3E 4C 11
QDC8 18 0F EB 21 D3 0B 22 29 31
QDD0 0C DD CB 00 46 28 1A 3E 57
QDD8 52 21 CC 0B 22 29 0C F7 7D
QDE0 EF 20 68 61 74 20 67 65 25
QDE8 77 6F 6E 6E 65 6E 2E 20 D8
QDF0 00 EF 4E 6F 63 68 20 65 F9
QDF8 69 6E 20 53 70 69 65 6C F9
QEE0 63 68 65 6E 3F 00 CD CB 83
QEE8 0E F7 F5 DF 6A F1 FE 4E 96
QE10 28 0B FE 4A CA 10 0D FE 7E
QE18 00 99 79 A9 D4 DF 5B E5 A4
QE20 DD 46 02 FF DD CB 00 4E 48
QE28 28 21 C5 D5 DF 61 D1 30 5A
QE30 19 21 F3 0E 0E 00 06 08 95
QE38 CB 21 BE 20 02 CB C1 23 C1
QE40 10 F6 79 B3 5F 79 D9 B3 E4
QE48 5F D9 C1 10 D6 DB 04 2F 43
QE50 DD CB 00 4E 28 01 7B E1 D9
QE58 C9 E6 0F 20 01 7A 57 1E 34
QE60 00 1F 30 01 2C 1F 30 01 3A
QE68 2D 1F 30 01 25 1F 30 01 68
QE70 24 C9 05 E5 21 F4 0B 22 67
QE78 29 0C FD E5 E1 06 00 11 95
QE80 E8 03 D7 12 11 64 00 D7 AE
QE88 0D 11 0A 0D D7 08 11 01 AF
QE90 0D D7 03 E1 D1 C9 3E 2F 6D
QE98 3C R7 ED 42 30 FA 19 CB D6
QEA0 40 20 03 FE 30 C8 F7 06 04
QEA8 FF C9 3E 0C F7 21 D4 0B BF
QEB0 22 29 0C EF 2A 2A 2A 20 A2
QEB8 44 4F 50 50 45 4C 57 55 36
QEC0 52 4D 20 2A 2A 2A 0D 20 38
QEC8 20 00 C9 2A 29 0C 4E 36 A2
QED0 5F EB CD 1F 0F EB D7 0C F1
QED8 71 D8 EB CD 09 0F EB D7 C1
QEE0 03 30 E8 C9 E5 2A 32 0C 1F
QEE8 DF 62 38 05 2B 7C B5 20 F0
QEF0 F7 E1 C9 2E 50 4C 3B 5A FE
QEF8 57 41 53 20 4D 2E 42 61 2F
QF00 63 68 31 39 2E 31 2E 38 09
QF08 32 F5 C5 D5 E5 D7 39 47 14
QF10 7E E6 C0 FE C0 78 20 01 9A
QF18 87 F7 E1 F1 C1 F1 C9 F5 E7
QF20 C5 D5 E5 D7 23 47 7E 4F BC
QF28 E6 C0 FE C0 78 20 05 2F 67
QF30 F6 C0 A1 77 E1 D1 C1 F1 71
QF38 C9 C5 D5 E5 D7 0A 47 7E 35
QF40 2F A0 E1 D1 C1 C0 37 C9 51
QF48 7C 0E 03 16 FF 14 91 30 CE
QF50 FC 81 47 7D 0E 00 CB 3F B8
QF58 30 01 0C 5F D7 13 3E 01 2C
QF60 04 87 10 FD CB 41 28 04 3F

```

```

OF68 87 87 18 02 CB 3F F6 C0 5F
OF70 C9 26 00 7A E6 0F 6F C5 11
OF78 06 06 29 10 FD C1 16 00 A0
OF80 7B FE 30 38 02 1E 2F 19 D8
OF88 11 0A 08 19 C9 FF FF FF 99

```

KLEINE BASIC-HILFEN

von Wolfgang v. Jan

Das folgende Programm dient z.B. bei speicherresidenten Dateiprogrammen dazu, die Aufnahme weiterer Werte nicht erst dann beendet zu bekommen, wenn nach vielen Eingaben der NASCOM einen OS-ERROR ausgibt. Es sorgt dafür, daß, als Unterprogramm eingesetzt, z.B. nach jeder Eingabe rechts oben auf dem Bildschirm der noch freie Speicherplatz für numerische wie Stringvariable angezeigt wird.

```

10 REM Ausdruck Speicher frei oben rechts
20 REM in Titelzeile 27.2.82 v.Jan
30 REM ██████████, ██████████ Langenhagen
40 S1$=STR$(FRE(1))
50 S2$=STR$(FRE(A$))
60 S3$=" "+S1$+" $ "+S2$
70 S2=LEN(S3$)
80 FOR S1=1 TO S2
90 POKE(3065-S2+S1),ASC(MID$(S3$,S1,1))
100 NEXT

```

Soll eine Bildschirmausgabe innerhalb eines Programmes häufiger mittig (d.h. nach rechts und links gleicher Abstand) auf dem Bildschirm erfolgen, kann folgendes Unterprogramm eingesetzt werden. Die Zeilen 60 bis 80 dienen dazu, die Ausgaben alternativ zu festen Zeilenpositionen auszugeben.

```

10 REM Bildschirm-Formatierung mittig
20 REM lt. ELCOMP 4/81 S. 39 v.Jan
30 INPUT AA$:GOSUB 30:REM 30: Testzeile
40 DEF FNA(V)=((48-LEN(AA$))/2):REM Mitte
50 PRINT TAB(FNA(V));AA$:RETURN
60 PRINTTAB(3);AB$:RETURN:REM UPRO TAB 3
70 PRINTTAB(5);AB$:RETURN:REM UPRO TAB 5
80 PRINTTAB(10);AB$:RETURN:REM UPRO TAB 10
90 REM ENDE Bildschirmformatierung

```

3-D-IRRGARTEN

von Rüdiger Maurer

```
10 CLS; REM 3D-IRRGARTEN v. R.Maurer
20 DOKE3200,25311;DOKE3202,312;DOKE3204,18351
30 DOKE3206,10927;DOKE3208,-8182;POKE3210,233
40 CLEAR
50 GOSUB1310
60 DIMM1(13,25)
70 DEFFNR(K)=INT(K*RND(1))+1
80 GOSUB1040;GOSUB1190
90 GOT01010
100 FORI=0T090;SET(I,0);NEXTI;REM LOC 0
110 FORI=0T038;SET(5,I);NEXTI
120 FORI=0T038;SET(85,I);NEXTI
130 IFJ<=3GOTO160
140 SET(0,40);SET(1,40);SET(2,39);SET(3,39)
150 SET(4,38);SET(5,38);RETURN
160 FORI=0T05;SET(I,38);NEXTI;RETURN
170 IFJ<=3GOTO200
180 SET(85,38);SET(86,38);SET(87,39);SET(88,39)
190 SET(89,40);SET(90,40);RETURN
200 FORI=85T092;SET(I,38);NEXTI;RETURN
210 FORI=0T034;SET(12,I);NEXTI;REM LOC 1
220 FORI=0T034;SET(78,I);NEXTI
230 IFJ<=3GOTO260
240 SET(6,37);SET(7,37);SET(8,36);SET(9,36)
250 SET(10,35);SET(11,35);RETURN;REM MAUER
260 FORI=5T012;SET(I,34);NEXTI;RETURN
270 IFJ<=3GOTO300
280 SET(79,35);SET(80,35);SET(81,36);SET(82,36)
290 SET(83,37);SET(84,37);RETURN
300 FORI=78T085;SET(I,34);NEXTI;RETURN
310 FORI=0T030;SET(19,I);NEXTI;REM LOC 2
320 FORI=0T030;SET(71,I);NEXTI
330 IFJ<=3GOTO360
340 SET(13,33);SET(14,33);SET(15,32);SET(16,32)
350 SET(17,31);SET(18,31);RETURN
360 FORI=12T019;SET(I,30);NEXTI;RETURN
370 IFJ<=3GOTO400
380 SET(72,31);SET(73,31);SET(74,32);SET(75,32)
390 SET(76,33);SET(77,33);RETURN
400 FORI=71T078;SET(I,30);NEXTI;RETURN
410 FORI=0T026;SET(26,I);NEXTI;REM LOC 3
420 FORI=0T026;SET(64,I);NEXTI
430 IFJ<=3GOTO460
440 SET(20,29);SET(21,29);SET(22,28);SET(23,28)
450 SET(24,27);SET(25,27);RETURN
460 FORI=19T026;SET(I,26);NEXTI;RETURN
470 IFJ<=3GOTO500
480 SET(65,27);SET(66,27);SET(67,28);SET(68,28)
490 SET(69,29);SET(70,29);RETURN
500 FORI=64T071;SET(I,26);NEXTI;RETURN
510 FORI=0T022;SET(33,I);NEXTI;REM LOC 4
520 FORI=0T022;SET(57,I);NEXTI
530 IFJ<=3GOTO560
540 SET(27,25);SET(28,25);SET(29,24);SET(30,24)
550 SET(31,23);SET(32,23);RETURN
560 FORI=26T033;SET(I,22);NEXTI;RETURN
570 IFJ<=3GOTO600
580 SET(58,23);SET(59,23);SET(60,24);SET(61,24)
590 SET(62,25);SET(63,25);RETURN
600 FORI=58T064;SET(I,22);NEXTI;RETURN
610 FORI=0T018;SET(40,I);NEXTI;REM LOC 5
620 FORI=0T018;SET(50,I);NEXTI
630 IFJ<=3GOTO660
640 SET(34,21);SET(35,21);SET(36,20);SET(37,20)
650 SET(38,19);SET(39,19);RETURN
660 FORI=33T040;SET(I,18);NEXTI;RETURN
670 IFJ<=3GOTO700
680 SET(51,19);SET(52,19);SET(53,20);SET(54,20)
690 SET(55,21);SET(56,21);RETURN
700 FORI=51T056;SET(I,18);NEXTI;RETURN
710 FORI=5T085;SET(I,38);NEXTI;RETURN
720 FORI=12T078;SET(I,34);NEXTI;RETURN
730 FORI=19T071;SET(I,30);NEXTI;RETURN
740 FORI=26T064;SET(I,26);NEXTI;RETURN
750 FORI=33T056;SET(I,22);NEXTI;RETURN
760 RETURN
770 SCREEN21,6;PRINT"START";RETURN
780 SCREEN20,6;PRINT"AUSGANG";RETURN
790 CT=0
800 J=M1(AI+DI*CT-DJ,AJ+DJ*CT+DI)
810 ONCT+1GOSUB100,210,310,410,510,610
820 J=M1(AI+DI*CT+DJ,AJ+DJ*CT-DI)
830 ONCT+1GOSUB170,270,370,470,570,670
840 CT=CT+1;J=M1(AI+DI*CT,AJ+DJ*CT)
850 IFJ=1ORJ=2GOTO870
860 IFJ=0ORJ=3ORJ=4GOTO880
870 IFCT<6GOTO800
880 ONCTGOSUB710,720,730,740,750,760
890 IFJ=0THENIFCT<6THENGOTO770
900 IFJ=4THENIFCT<6THENGOTO780
910 RETURN
920 DOKE 4100,3200;TA=USR(0);IFTA=0THEN920
930 IFTA=49THENJ=DJ;DJ=DI;DI=-J;CLS;GOTO1000
940 IFTA=51THENJ=DJ;DJ=-DI;DI=J;CLS;GOTO1000
950 IFTA=50THENAI=AI+DI;AJ=AJ+DJ;GOTO980
960 IFTA=83THEN;GOSUB1190
970 GOSUB790;GOTO920
980 IFM1(AI,AJ)<=3ANDM1(AI,AJ)<=0GOTO1000
990 AI=AI-DI;AJ=AJ-DJ;GOTO920
1000 CLS;IFM1(AI,AJ)=4GOTO1020
1010 GOSUB790;GOTO920
1020 GOT01550;REM PICTURE
1030 RETURN
1040 CM=MI*MJ/4;FORI=0TOMI;FORJ=0TOMJ
```

```

1050 M1(I,J)=3:NEXTJ:NEXTI
1060 DI=-1;DJ=0;AI=MI;AJ=FNR((MJ/2)-1)*2:CT=1
1070 M1(AI,AJ)=0;AI=AI+DI;AJ=AJ+DJ;SI=AI;SJ=AJ
1080 M1(AI,AJ)=1
1090 DI=FNR(3)-2;DJ=FNR(3)-2
1100 IFABS(DI)=ABS(DJ)GOTO1090
1110 AI=AI+DI;AJ=AJ+DJ
1120 IFAI=0ORAI=MIORAJ=0ORAJ=MJGOTO1170
1130 IFM1(AI,AJ)=1GOTO1090
1140 FORI=-1TO+1STEP+2;FORJ=-1TO+1STEP+2
1150 IFM1(AI+I,AJ+J)=1ANDM1(AI,AJ+J)=1ANDM1(AI+
I,AJ)=1GOTO1180
1160 NEXTJ:NEXTI:CT=CT+1;GOTO1080
1170 IFCT=CMTHENM1(AI,AJ)=4;AI=SI;AJ=SJ:RETURN
1180 AI=AI-DI;AJ=AJ-DJ;GOTO1090
1190 CLS
1200 FORI=0TOMI;FORJ=0TOMJ
1210 IFM1(I,J)=1ORM1(I,J)=2THENK=32
1220 IFM1(I,J)=3THENK=255
1230 IFM1(I,J)=0THENK=83
1240 IFM1(I,J)=4THENK=65
1250 IFAI=IANDAJ=JTHENK=7
1260 SCREEN J+10,I+1;PRINTCHR$(K)
1270 NEXTJ:NEXTI
1280 FORI=1TO5000:NEXTI
1290 CLS
1300 RETURN
1310 PRINT"Willkommen zum 3D-Irrgarten!"
1320 PRINT:PRINT"Sie sehen gleich eine kurze Ze
it einen"
1330 PRINT"Irrgarten aus der Vogelperspektive u
nd"
1340 PRINT"kurz danach befinden Sie sich mitten
"
1350 PRINT"im Irrgarten."
1360 PRINT:PRINT"Sie koennen sich im Irrgarten
frei"
1370 PRINT"bewegen: Linksdrehung = 1"
1380 PRINT"          Rechtsdrehung = 3"
1390 PRINT"          Geradeaus    = 2"
1400 PRINT"Wenn Sie gar nicht mehr weiter wisse
n,"
1410 PRINT"dann koennen Sie sich mit 'S' den zu
r"
1420 PRINT"Zeit gueltigen Standort anschauen."
1430 PRINT"Welchen Schwierigkeitsgrad (1-7) ?";
1440 DOKE4100,3200:TA=USR(0):IFTA=0THEN1440
1450 PRINTTA-48;
1460 IFTA=49ORTA=55THENCLS:PRINT:GOTO1430
1470 IFTA=49THENMI=7;MJ=7
1480 IFTA=50THENMI=8;MJ=10
1490 IFTA=51THENMI=9;MJ=13
1500 IFTA=52THENMI=10;MJ=16
1510 IFTA=53THENMI=11;MJ=19

```

```

1520 IFTA=54THENMI=12;MJ=22
1530 IFTA=55THENMI=13;MJ=25
1540 RETURN
1550 CLS:SCREEN15,2:PRINT"H U R R A ! ! ! ! !"
1560 SCREEN15,4:PRINT"D U B I S T "
1570 SCREEN15,6:PRINT"D R A U S S E N"
1580 SCREEN15,10:INPUT"Nach ein Spiel (J/N)";A$
1590 IFA$="J"GOTO10
1600 SCREEN15,12:PRINT"Na dann bis demnaechst!"
1610 END

```

Ok

BANDLAUF- ZÄHLER

von Erich Mehnert

Haben Sie schon 'mal auf einer C60 Cassette ein bestimmtes Programm gesucht? Wer des dauernden Vor- und Zurückspulens müde ist, sollte es einmal mit folgender Methode probieren: Mit Hilfe des kleinen Programms wird die Rückseite mit fortlaufenden Zahlen geladen, die anzeigen, welches Programm sich gerade auf der anderen Cassettenseite befindet. Man stellt den Recorder auf Aufnahme und startet bei C80. Nun kann man am linken oberen Bildrand eine laufende Nummer beobachten, die beim späteren Abspielen der Cassette wieder zu sehen ist. Drehen Sie vor dem Abspeichern eines Programmes die Casstte um und notieren Sie die entsprechende Zahl: das Programm ist jederzeit wieder leicht auffindbar.

```

OC80 3A 00 0E 00 CE 01 00 27 CA
OC88 00 32 20 0E 3A 01 0E 00 3D
OC90 CE 00 00 27 00 32 21 0E F2
OC98 3A 02 0E 00 CE 00 00 27 E3
OCA0 00 32 22 0E 3A 03 0E 00 59
OCA8 CE 00 00 27 00 32 23 0E 0C
OCB0 3A 04 0E 00 C2 00 00 27 F1
OCB8 00 32 24 0E 3A 05 0E 00 75

```

Weiter auf Seite 45

ERWEITERUNG der NASSYS 1 BEFEHLE

von Otto Föbel

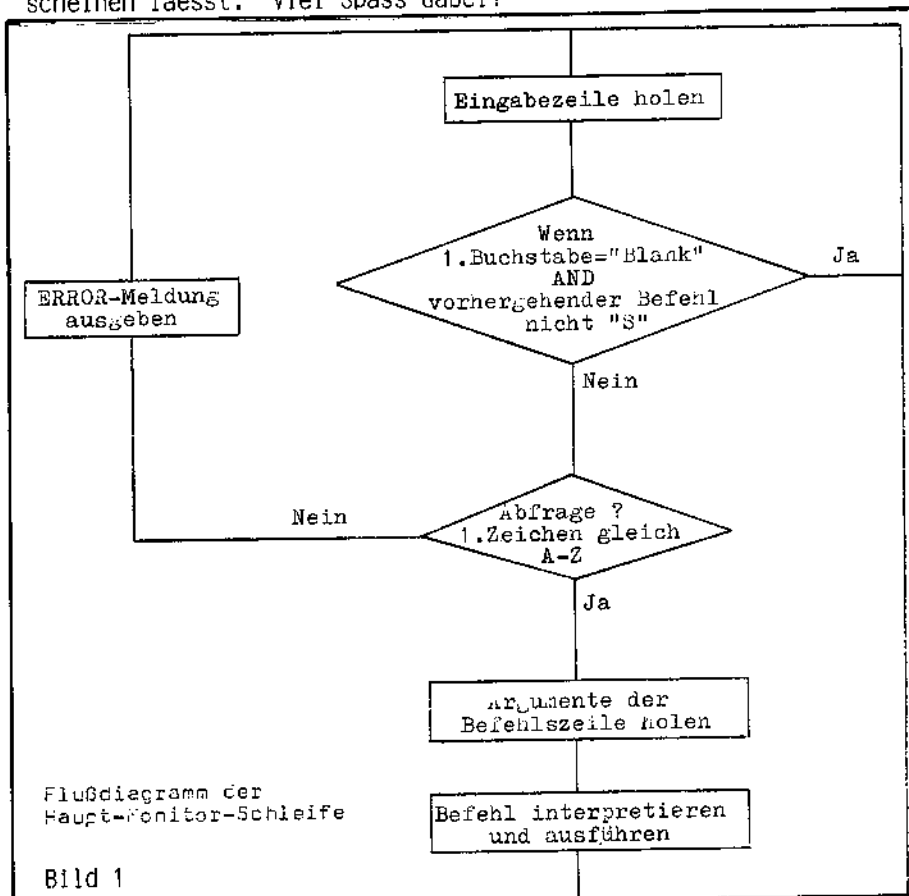
Während man fuer den Aufruf des Basic-Interpreters die Moeglichkeit besitzt, dazu einen speziellen kurzen Befehl in NASSYS zu verwenden (J oder Z), gibt es so eine Moeglichkeit fuer die anderen Systemprogramme nicht. Nasoen, Debugger, Zeap und Disassembler muessen alle mit dem Execute-Befehl gestartet werden, wobei jeweils die richtige Adresse fuer Warm-, bzw. Kaltstart bekannt sein muss. Leicht unuebersichtlich kann es werden, wenn man zusaetzliche Programme im Eprom parat hat, wie z.B. Treiberprogramme fuer Drucker, EPROMMER-Software oder sonstige Hilfsprogramme. Eleganter waere es dann doch, wenn man diese Programme ebenfalls mit besonders dafuer vorgesehenen Befehlen aufrufen koennte (z.B. Kleinbuchstaben). Mein Loesungsvorschlag dazu erfordert das Umprogrammieren von 4 Bytes in NASSYS 1 und ein kleines Programm, das dann zusaetzlich immer im Speicher vorhanden sein muss. Die Aenderung in NASSYS wuerde ich fuer Interessierte kostenlos machen (bitte Rueckporto nicht vergessen). Vorher moechte ich aber noch darauf hinweisen, dass Herr Maurer im Heft 11/12 eine aehnliche Moeglichkeit angekuendigt hat. (Siehe Heft 2/82 Red.) Wahrscheinlich kommt er ohne ein Umprogrammieren des Monitors aus, dafuer erscheint mir aber der benoetigte Speicherplatz von 2k als sehr gross und, ohne meine Loesung anpreisen zu wollen, glaube ich, dass sie flexibler ist, da sich jeder ohne Muehe seine eigenen Befehle definieren und anhaengen kann. Aber jetzt zu den Aenderungen im Monitor. Bei Adresse 03C9 bis 03F2 liegt in Nassys die Haupt-Monitor-Schleife, die jeweils eine Eingabezeile uebernimmt und dann den gueltigen Befehl ausfuehrt, bzw. bei ungueltigem Befehl eine ERROR-Meldung ausgibt. (Siehe Flussdiagramm Bild 1.) Diese Schleife wird einfach durch einen JP-Befehl zum eigenen Programm, das die zusaetzlichen Befehle interpretiert, geoeffnet. Vorher muss noch die Kontrolle auf gueltige Befehle (A-Z) durch die Aenderung eines Bytes erweitert werden (A-Z und a-z). Insgesamt muessten dafuer im Monitor diese vier Speicherplaetze dann folgendermassen aussehen:

03DD; 7B und ab 03EE: C3 xx xx ,wobei xx xx

die Anfangsadresse des angehaengten Interpreterprogrammes ist. Jetzt zu diesem Programm, das fast genauso einfach und kurz aussieht. Als erstes wird noch einmal ueberprüft, ob der Befehlsbuchstabe (ASCII Code), der in den Akku geholt wurde, ein grosser Buchstabe ist. Wenn ja, kann gleich wieder zurueck in die Monitorschleife gesprungen werden. Nachdem aber durch das Einfuegen des JMP-Befehles dort 2 Befehle ueberschrieben wurden, muessen diese zwei Befehle erst noch hier durchgefuehrt werden. (Siehe ab Label "GROSS" im Programmlisting Bild 2). Wurde ein kleiner Buchstabe im Akku vorgefunden, so wird das Programm bei "NEU" fortgesetzt, und dort kann man jetzt seine eigenen Befehle definieren, indem einfach mit einem CP-Befehl nachgeschaut wird, ob gerade dieser Buchstabe im Akku vorliegt. Ist dies der Fall, wird mit JR Z,xx dort hingespungen, wo festgelegt ist, was auf Grund dieses Befehls gemacht werden soll. Dadurch kann nacheinander auf die verschiedenen Buchstaben abgefragt werden. Am Ende dieser CP-Abfragen steht der Sprung zurueck in den Monitor, wo die ERROR-Meldung ausgegeben wird. Dadurch werden alle Buchstaben, die nicht explizit als neue Befehle durch die CP-Abfragen ausgesondert wurden, als ungueltig erkannt. Als einfaches Beispiel ist gezeigt, wie mit dem Buchstaben "o" das Programmierprogramm fuer EPROMs aufgerufen wird und mit dem Buchstaben "z" der ZEAP Assembler. Bei letzterem wurde vereinbart, dass der blanke Buchstabe "z" einen Kaltstart bedeutet; wird nach dem "z" noch eine beliebige Hexzahl mit eingetippt, so soll das als Warmstart interpretiert werden. Uebrigens: die Moeglichkeit, mit einem einfachen JMP-Befehl in die Systemprogramme zu springen, ist erlaubt. Soll naemlich z.B. von ZEAP zurueck in den Monitor gesprungen werden, so geschieht dies nicht mit einem RET-Befehl, denn dann wuerde hoechstens der Stack durcheinanderkommen und der Monitor trotzdem nicht gefunden. Statt dessen wird dort - und das sollte man auch fuer die eigenen Programme beachten - mit den Befehlen RST 0 (entspricht Reset) oder RST 20 zum Monitor zurueckgekehrt! Zum Schluss noch ein Tip, um die Programmierphantasie anzuregen. Ein Befehl "h" (help) koennte definiert werden, der eine Tabelle der neuen Befehle und ihrer Bedeutung auf dem Bildschirm er-

scheinen laesst. Viel Spass dabei!

Fortsetzung von Seite 43



Flussdiagramm der Haupt-Monitor-Schleife

Bild 1

Bild 2: Interpreterprogramm

```

BEFEHL EQU 0000A
ERROR EQU 000EA
ARGV EQU 0000B
;
Adr xx xx: BEGIN LD A,(BEFEHL)
                CP "a"
                JR C,GROSS
;
                CP "p"
                JR Z,PROMER
;
                CP "z"
                JR Z,ZEAP
                .
                .
                .
                JP ERROR
;
GROSS RST 18H
      DEFB 6CH
      RST 18H
      DEFB 5CH
      JP 03F2H
;
;
;
PROMER JP 0000H
;
ZEAP LD A,(ARGV)
     CP 1
     JP Z,DOUSH
     JP 0000H
     .
     .
  
```

```

0000 CE 00 00 27 00 32 25 0E 26
0008 21 0A 08 22 29 0C 3A 05 9D
00D0 0E DF 68 21 0C 08 22 29 B1
00D8 0C 3A 04 0E DF 68 21 0E B2
00E0 08 22 29 0C 3A 03 0E DF 75
00E8 68 21 10 08 22 29 0C 3A 26
00F0 02 0E DF 68 21 12 08 22 B0
00F8 29 0C 3A 01 0E DF 68 21 EA
0D00 14 08 22 29 0C 3A 00 0E C8
0D08 DF 68 DF 62 FE 2A CA 00 8F
0D10 00 FE 40 CA 20 0D 00 00 52
0D18 00 00 00 00 00 00 00 25
0D20 2A 20 0E 22 00 0E 2A 22 01
0D28 0E 22 02 0E 2A 24 0E 22 F3
0D30 04 0E 3E 20 00 DF 6F 3A 35
0D38 10 08 DF 6F 3A 11 08 DF DD
0D40 6F 3A 12 08 DF 6F 3A 13 AB
0D48 08 DF 6F 3A 14 08 DF 6F 4F
0D50 3A 15 08 DF 6F 3E 0D DF 2C
0D58 6F C3 80 0C 00 00 00 23
0D60 00 00 00 22 1E 0D 21 00 DB
0D68 10 22 2D 0D 22 48 0D CD 25
0D70 1D 0D C9 21 62 34 06 3A 67
0D78 0D 21 28 10 22 2D 0D 22 69
0D80 48 0D CD 1D 29 C8 20 6A 47
0D88 10 22 1E 0D 21 50 10 22 95
0D90 2D 0D 22 48 0D CD 1D 0D 45
0D98 C9 21 96 10 22 1E 0D 21 A3
0DA0 78 10 22 2D 0D 22 48 0D 08
0DA8 CD 1D 0D C9 21 BE 10 22 86
0DB0 1E 0D 21 A0 10 22 2D 0D 15
0DB8 22 48 0D CD 1D 0D C9 3A 36
0DC0 26 14 32 AC 10 C9 2A 00 A8
0DC8 0D 11 02 00 19 22 00 0D FD
0DD0 2A C3 0D 11 28 00 19 22 4B
0DD8 C3 0D C9 3E 28 32 CA 0D ED
0DE0 3E 02 32 D4 0D C9 3E 02 49
0DE8 32 CA 0D 3E 28 32 D4 0D 77
0DF0 C9 21 4C 11 06 C4 09 05 1C
0DF8 1E 14 22 C3 0D CD DB 0D DE
0E00 00 00 00 00 00 00 00 0E
0E08 00 00 00 00 00 00 00 16
0E10 00 00 00 00 00 00 00 1E
0E18 00 00 00 00 00 00 00 26
0E20 00 00 00 00 00 00 00 2E
0E28 00 00 00 00 00 00 00 36
0E30 00 00 00 00 00 00 00 3E
0E38 00 00 00 00 00 00 00 46
0E40 01 00 00 00 00 00 00 4F
0E48 00 00 00 00 00 00 00 56
0E50 00 00 00 00 00 00 00 5E
0E58 00 00 00 00 00 00 00 66
  
```

Flipper

von Peter Brendel

```
Ø CLS:U=12;REM * PETER BRENDL *
1 DOKE32ØØ,25311;REM * * *
2 DOKE32Ø2,312;REM * * MANNHEIM *
3 DOKE32Ø4,18351
4 DOKE32Ø6,1Ø927
5 DOKE32Ø8,-8182 (oder -8179 je nach BASIC-
6 POKE321Ø,233 Version Red.)
7 DOKE41ØØ,32ØØ
9 L=1;W2=2169;W1=2617;REM * TAUSEND *
1Ø Q(1)=Ø:Q(2)=Ø:Q(3)=Ø:N=Ø:U=15;GOSUB11Ø
15 REM * L= SPIELER N=ANZAHL U=ANZEIGEFELD
16 REM *Q(1)= 1,SPIELER (ERGEBNIS)
17 REM *Q(2)= 2,SPIELER (ERGEBNIS)
18 REM P=POSITION D=DISTANZ E=ERGEBNISANZEIGE
19 REM * TITEL "SPIELER" *
2Ø E=2166
3Ø POKEE,216;POKEE+1,227;POKEE+2,198
31 E=E+128
35 POKEE,246;POKEE+1,23Ø;POKEE+2,226
36 E=E+128
4Ø POKEE,237;POKEE+1,237;POKEE+2,237
41 E=E+128
45 POKEE,24Ø;POKEE+1,198;POKEE+2,198
46 E=E+128
5Ø POKEE,24Ø;POKEE+1,246;POKEE+2,246
51 E=E+128
55 POKEE,24Ø;POKEE+1,198;POKEE+2,198
56 E=E+128
6Ø POKEE,226;POKEE+1,198;POKEE+2,226
1ØØ GOTO175
11Ø FORI=1TOU
115 REM * BILDAUFBAU *
12Ø READB,C,D,F,G,H
13Ø FORJ=1TOG
14Ø FORA=BTØCSTEPD:POKEA,F
15Ø NEXTA;B=B+H:C=C+H:NEXTJ:NEXTI
16Ø P=2441;FORI=1TO4;POKEP,32:P=P+64;NEXT
17Ø RESTORE
172 RETURN
175 S=USR(Ø)
18Ø IFS=49THENGOSUB975
181 IFS=5ØTHENGOSUB975
19Ø IFPEEK(2168)=-255THEN175
194 IFN=3*MTHEN76Ø
195 IFL=1 THENPOKE21Ø4-N/M,239;FORI=1TO5Ø;NEXT
196 IFL=1 THENPOKE21Ø4-N/M,253
197 IFL=MTHENL=1
198 IFL=2THENPOKE2553-N/M,239;FORI=1TO5Ø;NEXT
199 IFL=2THENPOKE2553-N/M,253
2ØØ IFS=-32THEN175
21Ø N=N+1
22Ø REM * MERKER *
23Ø IFL=1 THENNR=2165
235 IFL=2 THENNR=2613
25Ø L=L-1
28Ø FORP=2898TO2928
29Ø POKEP,185;FORJ=1TO22;NEXT:POKEP,32
3ØØ NEXT:D=-64
31Ø POKEP,185
32Ø K=PEEK(P-64);IFK=-32THENGOSUB39Ø
33Ø K=PEEK(P+64);IFK=-32THENGOSUB4ØØ
34Ø K=PEEK(P+D);IFK=-32THENGOSUB39Ø
35Ø K=PEEK(P+1);IFK=-32THENGOSUB39Ø
36Ø S=USR(Ø);IFS=ØTHENGOSUB55Ø
365 Z=INT(RND(1)*4)
37Ø POKEP,32:P=P+D
38Ø GOTO31Ø
385 REM * ART DES HINDERNIS U.ABLENKUNG *
39Ø IFK=247THEND=64-Z;RETURN
4ØØ IFK=251THEND=-64-Z;RETURN
42Ø IFK=133THENGOSUB79Ø;GOTO5ØØ
43Ø IFK=134THENGOSUB81Ø;GOTO5ØØ
44Ø IFK=255THENGOSUB83Ø;RETURN
46Ø IFK=135THEND=-64;RETURN
47Ø IFK=21ØTHEND=-62-Z;RETURN
48Ø IFK=2Ø1THEND=65-Z;RETURN
485 IFK=187THEND=-5+Z;RETURN
49Ø IFK=186THENGOSUB87Ø;RETURN
495 REM * KUGEL IM AUS ? *
5ØØ IFPEEK(2441)=185THEN7Ø3
51Ø IFPEEK(25Ø5)=185THEN7Ø3
52Ø IFPEEK(2569)=185THEN7Ø3
53Ø IFPEEK(2377)=185THEN7Ø3
54Ø RETURN
55Ø IFS=19THENPOKE2443,137;REM * FLIPPER *
56Ø IFS=2ØTHENPOKE25Ø7,139
57Ø IFS=19THENPOKE2442,32
58Ø IFS=2ØTHENPOKE25Ø6,32
59Ø IFPEEK(2444)=185THEND=72;REM POSIT.D,KUGEL
6ØØ IFPEEK(2445)=185THEND=72;REM * VOR DEN
61Ø IFPEEK(2443)=185THEND=3;REM * FLIPPERN
62Ø IFPEEK(25Ø7)=185THEND=3
63Ø POKE2443,32;POKE25Ø7,32
64Ø IFPEEK(25Ø6)=185THEND=2
65Ø IFPEEK(25Ø8)=185THEND=-58
66Ø IFPEEK(25Ø9)=185THEND=-56
67Ø IFPEEK(2572)=185THEND=65
68Ø IFPEEK(2442)=185THEND=2
69Ø Q(L)=Q(L)+5
7ØØ POKE2442,136;POKE25Ø6,138
7Ø1 RETURN
7Ø3 FORI=22Ø8TO2218;REM * PUNKTE *
7Ø4 IFPEEK(I+64)=32THENQ(L)=Q(L)+4Ø
7Ø5 IFPEEK(I)=187THENQ(L)=Q(L)+1Ø
```

```

706 NEXT
707 FORI=2848T02858
708 IFPEEK(I-64)=32THENQ(L)=Q(L)+40
709 IFPEEK(I)=187THENQ(L)=Q(L)+10
710 NEXT
711 IFQ(3)≠1000THENGOSUB920
712 IFQ(2)≠1000THENGOSUB922
715 IFQ(L)≠1000THENQ(L)=Q(L)-1000
720 GOSUB925
745 U=14:GOSUB110:GOTO175:REM* TARGETS LOESCHEN
760 POKE2999,230:POKE2998,230:POKE2997,231
770 S=USR(0):IFS≠32THEN760
780 GOTO9
790 D=INT(RND(1)*4+62):IFD≠64THEND=63
795 Q(L)=Q(L)+1
800 RETURN
810 D=INT(RND(1)*4-68):IFD≠-64THEND=-65
815 Q(L)=Q(L)+1
820 RETURN
830 Y=RND(1)
840 IFY<.5THEND=61+INT(RND(1)*4)
850 IFY>.5THEND=-67-INT(RND(1)*4)
855 IFINT(Y*10)=5THEND=-2
860 RETURN
870 IFP≠2226THEN879:REM * LINKE GASSE
871 IFP≠2830THEN900:REM * RECHTE GASSE
872 D=2
875 RETURN
879 D=-INT(RND(1)*4):FORI=2144T02154
880 IFPEEK(I)=185THENPOKEI+64,187
893 IFPEEK(I)=185THEND=-8
895 IFPEEK(I)≠185THEND=-1
896 NEXT
897 GOTO875
900 D=-INT(RND(1)*4):FORI=2912T02922
901 IFPEEK(I)=185THENPOKEI-64,187
903 IFPEEK(I)=185THEND=-8
904 IFPEEK(I)≠185THEND=-1
905 NEXT
906 GOTO875
910 POKER,219:POKER+1,219:POKER+2,219:GOTO940
911 POKER,240:POKER+1,204:POKER+2,227:GOTO940
912 POKER,216:POKER+1,203:POKER+2,195:GOTO940
913 POKER,219:POKER+1,32:POKER+2,210:GOTO940
914 POKER,216:POKER+1,195:POKER+2,198:GOTO940
915 POKER,208:POKER+1,240:POKER+2,238:GOTO940
916 POKER,237:POKER+1,217:POKER+2,195:GOTO940
917 POKER,208:POKER+1,204:POKER+2,194:GOTO940
918 POKER,217:POKER+1,195:POKER+2,194:GOTO940
919 REM * UNTERPROGRAMME : AUSWERTUNG *
920 Q$="000":POKEW1,249:GOSUB927:R=2613
921 W1=W1+64:RETURN
922 Q$="000":POKEW2,249:GOSUB927:R=2165
923 W2=W2+64:RETURN

```

```

925 Q$=STR$(Q(L))
927 FORI=2TOLEN(Q$)
930 A$=MID$(Q$,I,1):A=VAL(A$)
933 REM * ZAHLENGRAFIK *
935 ONAGOTO910,911,912,913,914,915,916,917,918
937 POKER,213:POKER+1,210:POKER+2,234
940 R=R+128
943 NEXTI
945 RETURN
975 M=S-48
980 GOSUB110:GOTO190
986 K=255
1100 DATA2065,2099,1,247,1,0
1110 DATA2962,2995,1,251,1,0
1130 DATA2055,2378,63,133,10,1
1140 DATA2562,2961,65,134,10,1
1170 DATA2208,2858,640,186,11,1
1180 DATA2784,2794,1,210,1,0:REM210
1190 DATA2272,2282,1,201,1,0:REM201
1200 DATA2710,2713,1,133,1,0
1210 DATA2646,2649,1,135,1,0
1220 DATA2474,2478,1,135,1,0
1230 DATA2538,2542,1,133,1,0
1235 DATA2930,2930,1,135,1,0
1240 DATA2442,2442,1,136,1,0
1250 DATA2506,2506,1,138,1,0
1260 DATA2099,2995,64,255,7,1
1280 REM ANFANG,ENDE,STEP,DATUM,ANZAHL,INCREM,
OK

```

nascom- Journal INTERN

Liebe Leser,
diesmal das Editorial an ungewöhnlicher
Stelle und aus "Raumnot" auf engstem
Platz. Deshalb auch sehr kurz und bündig.

Beim Schaltbild zu J.C.Lotters Grafik
ging leider die Pin-Belegung der 2114
verloren. Sie wird nachgeliefert.
Die Seite für Einsteiger fiel diesmal ins
Wasser, da dem Autor für diese Ausgabe
eigentlich alles schiefging, was nur
schiefgehen konnte. Herr X wird sie eben-
falls nachliefern.
Die Rundläufe vom letzten Heft sind noch
nicht herausgegangen, wir wollten erst
das Erscheinen dieses Heftes abwarten,
das ja (hoffentlich!) knapp auf das letz-
te folgt. Einsendetermin für die Teilnah-
me bis zehn Tage nach Erscheinen dieses
Heftes. Folgende Rundläufe werden angebo-
ten: Yatzi, Format, Schröder-Spiele,
Grafik-Wettbewerb und Datenverwaltung.
Und nun viel Spaß mit dem neuen Journal
Ihr Günter Böhm

SUPER-ADVENTURE

von Wolfgang Schröder

Dies ist eine erweiterte Fassung des Spieles QUEST aus dem NASCOM Journal 11/12 81. Die Vorlage lieferte hier das ADVENTURE-Programm der DECUS-Gruppe. Das Programm benoetigt 32K Speicher fuer das Basic-Programm plus den Speicherbereich von 0CA0H bis 0FA0H fuer die Hoehlenkonfiguration.

ACHTUNG: Die Zeilennummerierung darf unter keinen Umstaenden geaendert werden! Das Programm traegt die benoetigten Zeilennummern in den Zeilen 1 und 2 selbstaendig ein. Dieses Verfahren eines berechneten Sprunges wurde bereits im NASCOM Journal... beschrieben und macht hier die sonst verwendeten ON, ..GOTO Spruenge ueberfluessig.

Bevor ich nun das Spiel erkläre, noch eine Bitte: Trotz ca. einwoechigem Test des Programmes kann es bei einem derart umfangreichen Programm noch vorkommen, dass Fehler unentdeckt geblieben sind. Sollten Sie also einen Fehler entdecken, so teilen Sie ihn mir bitte mit, damit dann eine Korrektur erfolgen kann.

Nach so vielen Vorworten nun die 'Bedienungs-Anleitung'. Das Programm kennt folgende 3 Arten von Woertern:

- 1.) Die 4 Himmelsrichtungen plus OBEN/UNTEN
- 2.) Aktionswoerter
- 3.) Gegenstaende

zu 1.) Um von einem Ort zu einem anderen zu gelangen, geben Sie eine der Richtungen ein, die unter 'Wege fuehren nach:' aufgefuehrt sind. Haben Sie sich aus Versehen (?!) einmal in die falsche Richtung bewegt, so kommen Sie durch die Eingabe von 'ZURUECK' in den meisten Faellen wieder an Ihren vorherigen Standort zurueck.

zu 2.) Die Aktionswoerter dienen dazu, Gegenstaende zu manipulieren. Eine Liste aller verfuegbaren Woerter erhalten Sie bei der Eingabe von 'HILFE'. Wollen Sie z.B. einen Gegenstand aufheben, so geben Sie bitte zuerst 'AUFHEBEN' ein. In den meisten Faellen weiss das Programm dann alleine, was Sie manipulieren wollen, ansonsten wird explizit nach dem Gegenstand gefragt. Es muss hier gesagt werden: in bestimmten Faellen wird nicht gleich der Gegenstand manipuliert, den

Sie eigentlich wollten, aber beharren Sie trotzdem hartnaeckig auf die 'Erfuellung' Ihres Wunsches (in diese Lage kommen sowieso nur erfahrene Hoehlenforscher!).

Unter die Aktionswoerter fallen auch bestimmte 'magische' Woerter; aber auf deren Bedeutung und Wirkung sollen Sie selber kommen.

zu 3.) Innerhalb und ausserhalb der Hoehle sind eine Menge Gegenstaende versteckt. Diese werden in 2 Gruppen eingeteilt: a) Hilfsobjekte und b) Schaetze.

Diese Hilfsobjekte brauchen Sie, um die verschiedenen Huerden, die in der Hoehle versteckt sind, zu ueberwinden (so z.B. einen Drachen); die Schaetze bringen Ihnen die ersehnten Punkte ein. Die volle Punktzahl gibt es aber nur, wenn Sie die Schaetze wohlbehalten in einem Haus abgeliefert haben; aber allein ihre Entdeckung gibt schon Punkte.

Aber passen Sie auf, manche Objekte haben auch unerwuenschte Nebeneffekte (die ich Ihnen hier natuerlich nicht verrate). Ausserdem ist Ihre Tragfaehigkeit auf 7 Gegenstaende beschaenkt.

Das Ende des Spieles kann auf zweierlei Arten erreicht werden: a) durch Eingabe des Aktionswortes 'AUFGEBEN', b) durch das Finden aller versteckten Schaetze.

Am Anfang werden Sie meistens die Moeglichkeit a) waehlen muessen (ich brauchte am Originalprogramm der DECUS-Gruppe ca. 1/2 Jahr um alles zu finden), Ihnen wird dann Ihre Punktzahl mit einer Bewertung ausgedruckt.

Sollten Sie nach reichlichem Training aber alles gefunden haben, dann geschieht etwas Interessantes, und Sie erhalten einen extra Bonus angerechnet.

Damit Sie an manchen Stellen nicht ewig haengen bleiben, gibt Ihnen das Programm manchmal nuetzliche Tips. Befolgen Sie diese, sonst wird's nie was mit dem Grossmeister aller Abenteurer.

Einen Tip gebe ich Ihnen noch zum Schluss: nehmen Sie die Axt mit, wenn Sie sie finden!

Dieses "Mammutspiel" ist, wie bereits im letzten Heft angekündigt, auf Cassette unter dem Stichwort "Schröder-Soiele" von der Redaktion als Rundlauf erhältlich. Die Cassette enthält NASCOM 1 und Kansas City Format.

Türme von Hanoi

von Eberhard Horch

```
100 REM *** INITIALIZE
110 DIM T(7,3)
120 E=0
130 FOR D=1 TO 7
140 FOR N=1 TO 3
150 T(D,N)=0
160 NEXT N
170 NEXT D
180 CLS:PRINT TAB(12);"Die Türme von Hanoi"
185 PRINTTAB(12);"_____"
190 PRINT:PRINT:PRINT"Eberhard Horch,Hannover":PRINT
200 PRINT"Sie müssen die Scheiben vom linken zum"
205 PRINT"rechten Turm bewegen,nur eine auf einmal"
210 PRINT"und keine Größere auf eine Kleinere.":PRINT
215 INPUT"Wieviel Scheiben sollen bewegt werden?";S
220 PRINT
230 M=0
240 FOR Q=1 TO 7
250 IF Q=S THEN 350
260 NEXT Q
270 E=E+1
280 IF E=2 THEN 310
290 PRINT" Tut mir leid,aber das geht nicht":GOTO215
310 PRINT"Gut,wenn Du das Spiel nicht richtig"
320 PRINT"machst hören wir eben auf!So long":STOP
340 REM *** STORE DISK FROM SMALL TO LARGE
350 PRINT"3 ist die kleinste Scheibe,dann kommt die"
360 PRINT"5,danach die 7 usw. bis 15.Spielst Du mit"
365 PRINT"2 Scheiben,ist der Codename 13 und 15.Mit"
370 PRINT"3 sind es die 11,13 und 15 usw.Die Nadeln"
375 PRINT"zählen von links nach rechts 1 bis 3.Wir"
380 PRINT"beginnen mit den Scheiben auf Nadel 1 und"
385 PRINT"versuchen sie auf Nadel 3 zu bringen,"
390 PRINT:PRINT"Viel Glück !":INPUT XY
400 Y=7:D=15
420 FOR X=S TO 1 STEP -1
430 T(Y,1)=D:D=D-2:Y=Y-1
460 NEXT X
470 GOSUB 1230
480 PRINT"Welche Scheibe wollen Sie bewegen?";
500 INPUT D
508 LET B=(D-3)*(D-5)*(D-7)*(D-9)
509 LET L=(D-11)*(D-13)*(D-15)
510 IF B*L=0 THEN 580
520 PRINT"Illegale Eingabe...nur3,5,7,9,11,13,15!!"
530 E=E+1:IF E=1 THEN 560
550 GOTO 500
560 PRINT"Halt,Sie verschwenden meine Zeit !"
580 REM *** CHECK IF DISK BELOW ANOTHER
590 FOR R=1 TO 7
600 FOR C=1 TO 3
610 IF T(R,C)=D THEN 640
620 NEXT C:NEXT R
640 FOR Q=R TO 1 STEP -1
645 IF T(Q,C)=0 THEN 660
650 IF T(Q,C)≠D THEN 680
660 NEXT Q
670 GOTO 700
680 PRINT"Diese Scheibe ist unter einer anderen!"
690 GOTO 480
700 E=0
705 INPUT"Auf welche Nadel soll die Scheibe";N
730 IF(N-1)*(N-2)*(N-3)=0 THEN 800
735 E=E+1
740 IF E=1 THEN 780
750 PRINT" Ich will den Fehler noch einmal vergeben,"
760 PRINT"aber denk daran,Ich erlaube nur einmal"
765 PRINT"einen Fehler.":GOTO 705
780 PRINT" Ich habe Sie gewarnt,aber Sie wollen ja"
790 PRINT"nicht hören! Bye bye !!":STOP
800 FOR R= 1 TO 7
810 IF T(R,N)≠0 THEN 840
820 NEXT R
830 GOTO 880
835 REM *** CHECK IF DISK ON A LARGER ONE
840 IF D<T(R,N) THEN 880
850 PRINT"Sie können keine größere Scheibe auf"
855 PRINT"eine Kleinere legen!"
860 PRINT"Sie kann zerbrechen !":PRINT:GOTO 480
880 FOR V=1 TO 7:FOR W=1 TO 3
900 IF T(V,W)=D THEN 930
910 NEXT W:NEXT V
925 REM *** LOCATE EMPTY SPACE ON NEEDLE N
930 FOR U=1 TO 7
940 IF T(U,N)≠0 THEN 970
950 NEXT U
960 U=7:GOTO 980
965 REM *** MOVE DISK,SET OLD LOCATION TO 0
970 U=U-1
980 T(U,N)=T(V,W):T(V,W)=0
995 REM *** PRINT OUT CURRENT STATUS
1000 GOSUB 1230
1018 REM *** CHECK IF DONE
1020 M=M+1
1030 FOR R=1 TO 7:FOR C=1 TO 3
1050 IF T(R,C)≠0 THEN 1090
1060 NEXTC:NEXT R
1080 GOTO 1120
1090 IF M= 128 THEN 480
1100 PRINT"Schade,aber nach 128 Zügen ist Ende,"
1110 PRINT"mehr sind nicht zugelassen!":STOP
1120 IFM<=21S-1 THEN 1140
1130 PRINT"Gratuliere!!"
1140 PRINT"Sie haben es in";M;"Zügen geschafft."
```

```

1150 PRINT:PRINT"Nach ein Versuch?(Ja,Nein)";
:INPUTA$
1160 IF A$="NEIN"THEN1390
1170 IF A$="JA" THEN 1200
1180 PRINT:PRINT"Ja oder Nein bitte";:INPUTA$
:GOTO1160
1190 IF M=128 THEN 480
1230 REM *** PRINT SUBROUTINE
1240 FOR K=1 TO 7
1250 Z=7
1260 FOR J=1 TO 3
1270 IF T(K,J)=0 THEN 1330
1280 PRINTTAB(Z-INT(T(K,J)/2));
1290 FOR V=1 TO T(K,J)
1300 PRINT"SIN";
1310 NEXT V
1320 GOTO 1340
1330 PRINTTAB(Z);"DEF";
1340 Z=Z+15
1350 NEXT J
1360 PRINT
1370 NEXT K
1380 RETURN
1390 PRINT"Danke für das Spiel!":PRINT:END

```

PROGRAMM "PROMILLE"

Das Programm "Promille" ist ein Würfelspiel für 1 bis 3 Spieler. Zuerst zeichnet der Computer für jeden Spieler ein Spielquadrat auf den Schirm, das aus neun einzelnen Feldern besteht. In diese neun Felder trägt jeder Spieler seine Würfelergebnisse ein, wobei er aussuchen kann, ob die Würfe mit 1, mit 10 oder mit 100 multipliziert werden. Wie bei einer dreistelligen Zahl werden die "Hunderter"-Würfe ganz links, die "Zehner" Würfe in der Mitte und die Einer ganz rechts eingetragen.

Es muß also jeder Spieler drei seiner neun Würfe mit 100, drei mit 10 und drei mit 1 multiplizieren lassen, sodaß er dann drei dreistellige Zahlen in seinem Spielfeld eingetragen hat. Diese drei Zahlen werden zusammengesetzt.

Das Ziel des Spiels ist es, möglichst nahe an die Zahl 1000 heranzukommen. Jede Differenz zu 1000 zählt als Schlechtpunkt (deshalb der Name "Promille"), es hat also sowohl ein Spieler mit 99 als auch ein Spieler mit 1010 Punkten als Ergebnis 10 Schlechtpunkte.

Jeweils unter dem Spielfeld wird die momentane Punktezahl aufsummiert; nach Ende einer Promillerrunde erscheint eine Zeile darunter die Anzahl der Schlechtpunkte für dieses Spiel und zwei Zeilen darunter die Summe aller Schlechtpunkte seit Spielbeginn.

P.S. Die Subroutinen zum Zeichnen der Spielfelder sind sehr abhängig von den Leerzeichen in den PRINT-Statements. Deshalb habe ich die Blanks mit "Ø" dargestellt.

KONVERTIERUNG AUF DEUTSCH

Zeile 10: Text "Wieviele Spieler (1-3)"
 Zeile 150: Print-Statement: PRINT "Spieler";I;";";R;"Punkte"
 Zeile 160: Text: "Welchen Multiplikator (H, Z oder E)"
 Zeile 170: IF W\$="H" OR W\$="Z" OR W\$="E" THEN 190
 Zeile 191: IF W\$="Z"
 Zeile 290: Text: "Noch ein Spiel"

PROGRAMM "PROMILLE" BY CH.PETER

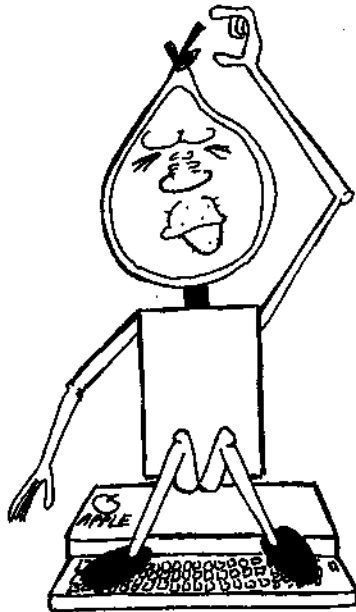
JUNE 9, 1980

```

10 CLS:INPUT"How many players (1-3)";Z:Z=INT(Z)
15 IF Z<1 OR Z>3 GOTO 10
17 DIM A(Z,3,3),S(2*Z)
20 CLS:FOR H=1TO3:GOSUB 100:NEXT H
30 GOSUB 130:PRINT
35 FOR J=1TO9
40 FOR I=1TOZ
50 R=INT(10*RND(2))
60 IF R<1 OR R>6 GOTO 50
80 GOTO 150
100 FOR I=1TOZ:PRINT "ØØ";:FOR J=1TO3:PRINT "Ø-ØØ";:NEXT J
104 PRINT"Ø";:NEXT I:PRINT
110 FOR I=1TOZ:PRINT CHR$(8);:FOR J=1TO4:PRINT "Ø|Ø";:NEXT J,I
115 IF I=4 THEN PRINT CHR$(8);:RETURN
120 PRINT:RETURN
130 FOR I=1TOZ:PRINT "ØØ";:FOR J=1TO3:PRINT "Ø-ØØ";:NEXT J
135 PRINT"Ø";:NEXT I
140 RETURN
150 SCREEN 1,10:PRINT"Player";I;";";R;"Points"
160 INPUT"Which multiplier (H,T or O)";W$
170 IF W$="H" OR W$="T" OR W$="O" THEN 190
180 SCREEN 1,11:PRINT SPC(96):GOTO 150
190 IF W$="H" THEN W=I*15-12:V=3:GOTO 200
191 IF W$="T" THEN W=I*15-8:V=2:GOTO 200
192 W=I*15-4:V=1
200 IF A(I,V,1)=0 THEN A(I,V,1)=R:Y=2:GOTO 240
210 IF A(I,V,2)=0 THEN A(I,V,2)=R:Y=4:GOTO 240
220 IF A(I,V,3)=0 THEN A(I,V,3)=R:Y=6:GOTO 240
230 SCREEN 1,10:PRINT SPC(96):GOTO 150
240 SCREEN W,Y:PRINT R:S(I)=S(I)+R*10*(V-1)
250 SCREEN I*15-6,8:PRINT S(I)
260 NEXT I,J
270 FOR I=1TOZ
280 SCREEN I*15-6,9:S(I)=ABS(INT(1000-S(I)+.5))
282 S(Z+1)=S(Z+1)+S(I):PRINT S(I):PRINT SPC(80)
285 SCREEN I*15-6,13:PRINT S(Z+1):NEXT I:PRINT SPC(48)
290 INPUT"Another game";A$
300 IF LEFT$(A$,1)="N" THEN END
310 FOR I=1TOZ:FOR J=1TO3:FOR H=1TO3:S(I)=0
330 A(I,J,H)=0:NEXT H,J,I:GOTO 20

```

NASCOMPL



Hallo liebe Leser,

zu Pfingsten war es, als die Apostel begannen, in verschiedenen Sprachen zu reden. Was würde sich eher anbieten, als zu diesem Zeitpunkt über die Sprachenvielfalt (oder besser Sprachverwirrung) in unserer Microcomputer-Welt zu sprechen?

Wer hat nicht schon bedauert, nicht über LISP oder FORTH, über FORTRAN oder PASCAL bescheid zu wissen? Auch für die Profis ergeben sich Schwierigkeiten, weil einfach ein allgemeingültiger Standard fehlt.

Doch nun ist Schluß mit dem Wirrwarr, vorbei mit der Vielzahl unverständlicher Dialekte. MK-Systemtechnik hat nun endlich eine Sprache entwickelt, die schon dabei ist, sich international durchzusetzen: BIRNE, die Sprache die Vorteile aller existierender Sprachen in sich vereinigt. BIRNE ermöglicht die Formulierung der komplexesten Probleme auf einfachste Weise. Mit mnemonischen Ausdrücken, die der deutschen Sprache entlehnt sind, haben es natürlich wir Deutschsprachigen besonders leicht, in die neue Entwicklung einzusteigen. So bedeutet der Befehl ERBSE 80 : "Nimm den Erstbesten von 160."

Nutzen Sie die Möglichkeit, BIRNE jetzt zum Einführungspreis bei MKS zu bestellen! Ein Bleistift und ein Radiergummi gehören zum Lieferumfang. Das alleine ist schon ein Grund, schnell zuzugreifen. In diesem Sinne Ihr NASCOMPL

IMPRESSUM

REDAKTION: Günter Böhm, Günter Kreidl
Wolfgang Mayer-Gürr, Josef Zeller

RESSORTS:

MASCHINENPROGRAMME:

Günter Böhm, [REDACTED]

[REDACTED] Karlsruhe, [REDACTED] Tel. [REDACTED]

Günter Kreidl, [REDACTED], [REDACTED] Straelen
Tel. [REDACTED]

BASIC und FLOPPY:

Wolfgang Mayer-Gürr, [REDACTED]

[REDACTED], [REDACTED] Recklinghausen
Tel. [REDACTED]

HARDWARE:

Josef Zeller, [REDACTED], [REDACTED] Neu-Ulm

VERLAG: NASCOM JOURNAL, c/o MK-Systemtechnik

Pater-Mayer-Str.6,

6728 Germersheim

Tel. 07274/2756

Telex 453500 mkxd

VERTRIEB: Direktvertrieb durch den Verlag
Erscheinungsweise: monatlich

Bezugspreis: Im In- und Ausland 48,- für ein Jahresabonnement. Abonnements können aus technischen Gründen immer nur für die Dauer eines Kalenderjahres, d.h. vom 1.1. bis 31.12. laufen. Bei Bestellung nach dem 1.1. werden die fehlenden Hefte mit der ersten Lieferung bis zum Bestellzeitpunkt automatisch mitgeliefert. Bei nicht fristgerechter Kündigung verlängert sich das Abonnement automatisch um ein Jahr.

Bezugsmöglichkeiten: Durch Bestellung bei MK Systemtechnik.

Bankverbindungen: Alle Zahlungen für das NASCOM JOURNAL unter Angabe der Rechnungsnummer an MK - Systemtechnik, Germersheim.

Zahlung: Nach Eingang Ihrer Bestellung erhalten Sie von uns die ausstehenden Hefte bis zur aktuellen Ausgabe sowie eine Rechnung. Bitte, zahlen Sie dann den Rechnungsbetrag.

Bitte keine Vorauszahlungen!

Bitte, Anfragen wegen Abonnements oder Lieferung nicht an die Redaktion sondern nur an den Verlag. Die Autoren tragen die Verantwortung für ihre Beiträge selbst. Für Fehler in Text, Bildern und sonstigen Angaben kann keine Haftung übernommen werden.

NASCOM

Die Alternative!

Kein »langweiliger Computer«

NASCOM 1 und NASCOM 2 sind Computer für Selbsterbauer, Tüftler, erfolgreiche Do-it-yourself-Freunde. NASCOM-Computer werden niemals langweilig! Die Systeme 1 und 2 sind keine fertigen »Kästen« ohne Erweiterungsmöglichkeit. Der hochwertige Platineinsatz Computer und Keyboard kann so aufgebaut, erweitert und »verpackt« werden, wie Sie es wünschen.

Für Vollpreis gibt die NASCOMs auch als Bausatz. Aber aufgepaßt: Das ist eine Sache nur für wirkliche Könnler! Und damit es auch nach dem Aufbauen nicht langweilig wird, gibt es das monatlich erscheinende NASCOM-JOURNAL. Eine Zeitschrift speziell für NASCOM-Freaks vollgestopft mit Hardware- und Software-Ideen, Kleinanzeigen, den neuesten Infos, und, und, und. . .

Die NASCOMs sind keine »Spielcomputer«. Mehr als 60% aller NASCOM-Systeme werden als sogenannte

»OEM-Baugruppen« von professionellen Anwendern in eigene Systeme eingebaut. Ingenieurbüros verwenden den NASCOM als Entwicklungssystem. Die Anwendungsmöglichkeiten sind mehr durch Ihre Phantasie begrenzt. Ein NASCOM-System kann fast alle gängigen Probleme lösen.

Mit NASCOM wachsen!

NASCOM-Systeme sind aufwärtskompatibel. Das kleinste, preisgünstigste NASCOM 1-System kann bis auf NASCOM 3-Level mit Floppy-Laufwerken und CP/M* ausgebaut werden. Bildschirm-Aussteuerung, Tastatur Betriebssystem und Systemsoftware sind durchweg kompatibel. Ohne faule Kompromisse! Mit NASCOM-Systemen gehen Sie kein Risiko ein. Ihr NASCOM wächst mit!



NASCOM 1

Spezifikationen:

- QWERTY-Tastatur, aufgebaut mit hochwertigen Magnetasten
- NAS-SYS Betriebssystem (2k Byte)
- 16 I/O-Leitungen
- Video (BAS) und TV-Ausgang
- 1k RAM, ausbaubar auf 192k RAM
- Display 64 Zeichen in 16 Zeilen

ab DM 935,-



NASCOM 2

Spezifikationen:

- Wie NASCOM 1, jedoch zusätzl.:
- 8k Mikrosoft-BASIC u. 8k Stat. RAM
 - Z80A-Mikroprozessor, 4 MHz
 - Erweiterte Tastatur 57 Tasten
 - Integrierte Bus-Pufferung
 - Bis 192k Byte RAM
 - Grafik-Möglichkeiten: 48 x 96 Punkte
 - Serielle Schnittstelle; Baudrate wählbar, R5232C/20mA
 - 16 parallele Ein/Ausgabeleitungen (Z80APIO)

ab DM 1950,-

NASCOM 3 — der Profi



Spezifikationen: Wie NASCOM 2, jedoch zusätzl.:

- 0.35 Megabyte pro 5,25-Zoll Laufwerk
- Betriebssystem CP/M* 2.2 oder NAS-DOS
- Bildschirmausgabe erweiterbar auf 80 x 25 Zeichen

ab DM 2735,-

Die dritte NASCOM-Generation

NASCOM 1 und 2 haben OEM-Board, Schulungscomputer, Kompaktrechner etc. ca. 20 000 mal ihren Partner gefunden. Der NASCOM 3 möchte Ihr persönlicher Computer werden! Er möchte Ihnen helfen, sich selbst fortzubilden, im Beruf weiter zu kommen, auch mal in die Computertechnik »rein zu riechen«. Ingenieurbüros und Softwareingenieuren dient der NASCOM 3 als preisgünstiges Entwicklungssystem.

Universelle Betriebssoftware

Der NASCOM 3 kennt zwei Betriebssysteme: Das CP/M* (Version 2.2) — inzwischen Standard — und sein eigenes NAS-DOS. Die 5-Zoll Floppys bieten eine Speicherkapazität von 0.35 Megabyte pro Laufwerk (single sided, double density, double tracked). Damit wird das Spektrum universeller CP/M*-Software verfügbar!

**Wir informieren Sie unverbindlich:
Fordern Sie Ihr NASCOM-INFO-PAKET an! ****

Unsere Händler:

Heinz Vogel Verlag GmbH & Co.
Lehrmittelenzentrum, Herr Seloff
Innsbrucker Straße 96
2800 Bremen-Findorff
☎ (04 21) 35 10 69

ELPHOTEC GmbH
Große Bleiche 1
6500 Mainz
☎ (061 31) 1 89 56

MK-SYSTEMTECHNIK
Michael von Keitz
Pflaßberg 4
5650 Solingen
☎ (0 21 22) 4 72 67

MK-SYSTEMTECHNIK
Kriegsstraße 164
7500 Karlsruhe
☎ (07 21) 2 92 43

Radio Zinburg
Herr Zinburg, Jr.
Röhstraße 10
5760 Arnsberg
☎ (0 29 32) 3 15 10

Graf Elektronik Systeme GmbH
Postfach 1610
8791 Kempten
☎ (08 31) 6 19 30

Autorisierter Distributor:



MK-SYSTEMTECHNIK
Peter-Mayer-Straße 6
6728 Gernersheim
☎ (0 72 74) 20 93
Telex 453500 mks d

CP/M* ist ein eingetragenes Warenzeichen der DIGITAL RESEARCH

** NASCOM-INFO-PAKET gegen DM 2,- in Briefmarken (wird bei Kauf angerechnet)