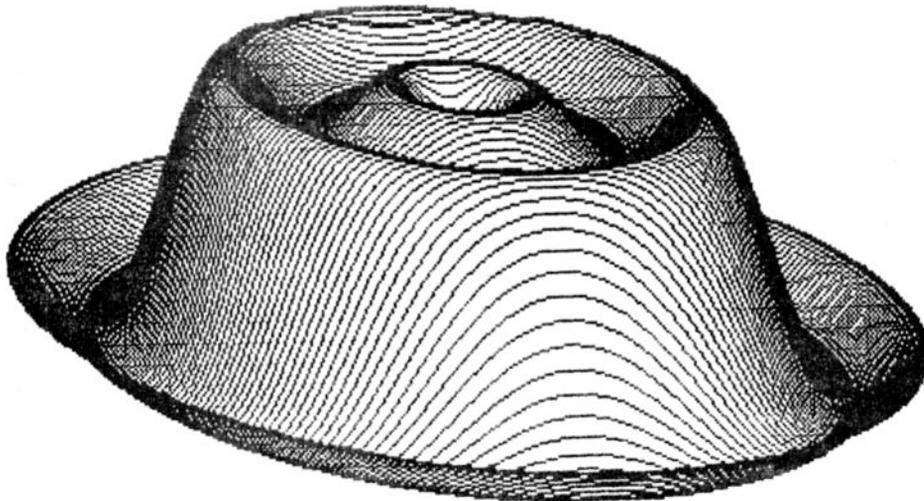


GENIE
TRS-80
USER CLUB
"BREMERHAVEN"

C L U B I N F O C L U B I N F O



4. JAHRGANG 2. OB. : OB. AUSGABE
AUGUST : SEPTEMBER 1986

REDAKTION: RALF FOLKERTS, NUTLHORNER STR. 9, 2875 BOOKHOLMBERG
PRINTED 1986 BY PETER SPIESS, RENNERTSHOFEN
AUFLAGE: 095 EXEMPLARE

INHALT CLUB INFO

IN DIESEM HEFT FINDET IHR
FOLGENDE ARTIKEL:

Seite.....Inhalt

- 1...Titelbild: Grafik von Peter Spieß
- 2...Inhaltsverzeichnis
- 3...Internes vom Betreuer
- 4 - 7...Real - Time Black - Box ? Von Arnulf Sopp
- 7...Mailbox Corner
- 8 - 9...Wenn die Uhr mal stört. Von Arnulf Sopp
- 10 - 11...Nachtrag zu "Sondertasten..". Von Arnulf Sopp
- 12 - 16...Neue Zeichensätze für das Genie IIIs.
.....Von Arnulf Sopp
- 17 - 21...DDE für das Genie IIIs. Von Arnulf Sopp
- 22 - 28...HRG Hardcopy für das Genie IIIs. Von Arnulf Sopp
- 29 - 34...Die HRG des Genie IIIs. Von Arnulf Sopp
- 35 - 36...Was haltet Ihr von Textsystemen ? Von Kajot
- 37 - 43...ARNULF SOPP. Von Kajot
- 44 - 46...Grafik CLS für das Genie IIIs. Von Arnulf Sopp
- 47 - 49...HRG Hardcopy. Von Ernst Sikora
- 50 - 53...HRG im Maßstab, Version 2. Von Karl Rubes
- 54 - 55...Ein neuer LIB - Befehl: "GO". Von Arnulf Sopp
- 56 - 58...Korrektur zu 'CALL um die Ecke'. Von Arnulf Sopp
- 59 - 60...Personelles im INFO

Meinen Dank an alle Autoren

INTERNES VON BETREUER

Bankverbindung des Clubs:

Post giroamt Hannover

BKz: 250 100 30

KTO: 5342 84 - 308

Hallo Leute,

diesmal gibt es ein 'Doppelheft'. Ich habe die August- und die September - Ausgaben zu einem Heft zusammengefaßt. Der Grund hierfür ist, daß Peter im September seinen Urlaub antritt. Außerdem möchte ich mich auch mal von den INFOS erholen.

Für diese Ausgabe habe ich fast meinen gesamten Vorrat an Artikeln 'verbraucht'. Ich habe hier nur noch einen Artikel über einen universellen Joystickport liegen, den ich jedoch erst noch einmal abtippen muß, da ich nur die schwache Kopie des handschriftlichen Originals habe.

Die Fragebogen - Aktion sehe ich als gestorben an. Ich habe jetzt 37 Fragebögen hier. Bei einer derart geringen Resonanz werde ich mir die Arbeit der Auswertung ersparen - vorerst jedenfalls. Für diejenigen, die mir die Fragebögen sofort zurückgeschickt haben, tut es mir natürlich leid; ich habe jedoch keine Lust, mir jetzt die Arbeit zu machen und die ganzen Bögen auszuwerten - wenn noch nicht einmal die Hälfte der Mitglieder mir diese ausgefüllt zurückschickt. Ich werde mir die eingegangenen jedoch durchlesen, und versuchen, auch evtl. vorhandene Kritik zu reagieren.

Das Preisblatt in diesem Heft hat mir Waldemar Grundmann gedruckt zugeschickt. Ich bitte um dessen Beachtung.

Im letzten INFO hat Peter im Streß leider ein paar Folien vertauscht, weshalb die Seiten nicht ganz so lagen, wie gewohnt.

Ach ja, die INFOS der letzten Ausgabe habe ich, mangels Buchbinderleim, mit weißem Holzleim geleimt - meiner Meinung nach waren diese Hefte besser zu trennen, als die 'Buchbindermäßigen'. Ferner trocknet der Holzleim schneller.

Ich weise noch einmal darauf hin, daß im September kein INFO erscheint und wünsche allen Urlaubern noch einen schönen selbigen.

Bis zum nächsten Heft im Oktober !!

Real Time Black Box?

Für das Genie III s gibt es eine Echtzeituhr (real time clock, RTC), auf der der Uhren- und Kalenderchip MSM 5832 steckt. Da TCS bzw. Phönix aus Gründen der Wirtschaftlichkeit vermutlich dieselbe Modellpolitik betreibt wie VW (Audi und VW unterscheiden sich hauptsächlich am Firmensignet), ist anzunehmen, daß noch weitere Genies diese Uhr haben. Es macht mich nicht besonders stolz, den UHR- oder DATUM-Befehl korrekt eintippen zu können. Vielmehr muß es möglich sein, ohne Unterstützung des DOS die Uhr in eigenen Programmen auszulesen.

Zu diesem Zweck fragte ich bei Phönix an, erhielt aber außer dem Versprechen, man wolle sich erkundigen, keine Reaktion. So blieb mal wieder nichts anderes übrig, als die Uhr-Routinen des Betriebssystems zu analysieren. Willkommene Unterstützung bot ein Auszug eines Artikels über eine andere Uhr mit dem MSM 5832, woraus ich die Bedeutung der internen Register (s. Tabelle am Ende dieses Textes) ersehen konnte. Die Angabe der Quelle muß ich leider schuldig bleiben, weil ich sie selbst nicht kenne.

Der MSM 5832 hat 13 Register, in denen die in der Tabelle genannten Zeitinformationen im BCD-Format vorliegen. Das bedeutet, daß alle Zeiteinheiten (außer dem einstelligen Wochentag) als zweistellige Dezimalzahlen schreib- bzw. lesbar sind. Für eine Digitaluhr mit LCD-Anzeige ist das zweifellos von Vorteil. In binär "denkenden" Computern wäre allerdings eine duale Codierung sinnvoller. Nun ja, ist nun mal so.

Um zum Stellen oder Auslesen der Uhr die entsprechenden Register zu erreichen, müssen dem Chip ihre 4 Bit breiten Adressen mitgeteilt werden. Zugleich muß eines von zwei weiteren Bits auf 1 stehen, um die Uhr entweder zum Lesen oder zum Schreiben vorzubereiten. Offenbar wird sie völlig deaktiviert (läuft aber intern weiter), wenn als Abschluß einer Schreib-/Leseoperation beide Bits rückgesetzt werden. Über diesen Punkt bin ich mir nicht ganz sicher. Jedenfalls wird in der Genie-Routine so verfahren.

Im Genie III s ist diese Ansprache der Zielregister folgendermaßen gelöst: Der Port 5B erhält einen Output mit dem Bitmuster `xxxx1yy0`. Die vier x-Bits beinhalten die Adresse des gewünschten Registers. Die Bits y unterscheiden zwischen Schreiben und Lesen. Beim Stellen (Schreiben) der Uhr lauten sie 01, beim Auslesen 10. So wird z. B. das Register 9 (Einerstelle der Monate) zum Schreiben mit dem Output `10011010 (=9A)` auf den Port 5B adressiert. 9C würde das Lesen des Monats-Einers ermöglichen.

Nachdem das Zielregister angesprochen ist, kann die Zeiteinheit eingegeben oder ausgelesen werden. Hierfür ist beim Genie III s der Port 5A zuständig. Hierbei ist zu beachten, daß nur die unteren 4 Bit von Belang sind. Außerdem beinhalten die Register 5 und 8 zusätzliche Informationen über Schaltjahre, 12h- bzw. 24h-Anzeige sowie die Tageshälfte für die AM/PM-Anzeige. Die letztlich relevanten Bits müssen deshalb vor der Anzeige maskiert werden. Umgekehrt müssen beim Stellen der Uhr diese Zusatzbits je nach dem gesetzt oder rückgesetzt sein.

Im Anschluß an diesen Text sind die beiden Routinen gelistet, die im Genie III s die Uhr stellen bzw. lesen (Labels `wrclock` und `rdclock`). Dabei handelt es sich um einen Auszug aus einem Listing des kompletten residenten Betriebssystems, das ich mir anfertigte, um die leider sehr bescheidene Dokumentation zu ergänzen. Daher fehlen einige Informationen, die mit LIST OFF ausgespart sind. Der Leser möge deshalb die jeweilige Ladeadresse der linken Spalte entnehmen. Labels, die nicht innerhalb des Listings definiert sind, ergeben sich aus der zweiten Spalte von links. Die Programmlogik dürfte sich aus den reichlichen Kommentaren ergeben.

Das Programm liegt "hinter" dem Bildschirm, ist also nicht ohne weiteres für Änderungen zugänglich. Zuvor muß auf den Port FA ein Byte mit gesetztem Bit 0 ausgegeben werden (also eine ungerade Zahl). Mit CALL 06A0 erledigt das auch das Betriebssystem. Nach der Manipulation wird mit CALL 06AB oder einem geradzahligem Output auf Port FA der Screen wieder zugeschaltet. Vorsicht mit diesem Port! Ein falscher Output kann die Arbeit von Stunden zunichte machen. Zuerst den Input lesen, dann das Bit 0 ändern, dann neu ausgeben!

Sagte ich weiter oben "Logik"? Der Algorithmus ist zwar wirklich ziemlich elegant und läßt sich kaum nennenswert verkürzen oder beschleunigen. Daß aber in Zeile 745 die Sekunden immer auf 0 gestellt werden, obgleich das DOS ohne weiteres die Eingabe von beispielsweise 13:54:32 zuläßt, ist nicht einzusehen. Es ist auch im Betriebssystem nicht vorgesehen, Schaltjahre zu berücksichtigen oder auch den Wochentag zu programmieren. Das ist peinlich, wenn CP/M ihn anzeigt - natürlich mit siebenfacher Chance den falschen. Mit dieser kleinen Routine läßt er sich ein- für allemal einstellen:

```

start   ORG      5200h
        CALL     06b5h      ;auf 1,78 MHz schalten
        LD      A,6ah      ;Register 6 schreiben
        OUT     (5bh),A    ;Befehl ausgeben
        LD      A,7        ;Beispiel hier: Sonntag
        OUT     (5ah),A    ;Ausgabe an die Uhr
        XOR     A          ;A ← '00, Uhr "stillegen"
        OUT     (5bh),A    ;Befehl ausgeben
        JP      06beh      ;7,2 MHz und Ende
        END     start     ;dort Einsprung

```

Die Triebfeder für mich, den Wecker auf die Couch zu legen, war zunächst reiner Wissensdurst. Der Leser möchte jedoch vielleicht etwas damit anfangen. Bei zeitkritischen Messungen kann beispielsweise das Register 0 via Polling einen Sekundentakt liefern, der von den Systeminterrupts unabhängig ist. Wenn für kaufmännische Anwendungen die Wochentage interessieren, die vom DOS aus nicht abgerufen werden können, so sind sie, wie ich hoffe, nach meinen Erläuterungen nun ebenfalls programmgesteuert verfügbar. Offen gestanden fällt mir im Moment nicht mehr dazu ein. Vielleicht lesen wir im nächsten Info einen interessanten Vorschlag von einem anderen Mitglied?

Tabelle zur Adressierung des MSM 5832:

Register	Datenbits				Bedeutung
	D3	D2	D1	D0	
0	x	x	x	x	Sekunden Einer
1		x	x	x	Sekunden Zehner
2	x	x	x	x	Minuten Einer
3		x	x	x	Minuten Zehner
4	x	x	x	x	Stunden Einer
5	a	b	x	x	Stunden Zehner *)
6		x	x	x	Wochentag (Montag = 0)
7	x	x	x	x	Datum Einer
8		c	x	x	Datum Zehner *)
9	x	x	x	x	Monat Einer
A				x	Monat Zehner
B	x	x	x	x	Jahr Einer
C	x	x	x	x	Jahr Zehner

- *) a: Register 5, Bit 3 gesetzt im 24h-Format
 b: Register 5, Bit 2 gesetzt ab Mittag (PM-Flag)
 c: Register 8, Bit 2 gesetzt in einem Schaltjahr

Arnulf Sopp

00718 ; Routinen zum Stellen und Lesen der Hardware-Uhr des G3s
 00719 ; (C) by TCS Computer GmbH

3E97	CDB506	00720	wrclock	CALL	lsp	; auf 1,78 MHz umschalten
3E9A	214140	00721		LD	HL,time	; Uhrzeit, Sekunden (RAM)
3E9D	0E02	00722		LD	C,02h	; 2 Durchl. (Zeit, Datum)
3E9F	3E23	00723		LD	A,23h	; = INC HL
3EA1	32BC3E	00724		LD	(incdec1),A	; dort patchen
3EA4	160A	00725		LD	D,0ah	; WR ab RTC-Reg. 0 (Sek.)
3EA6	0603	00726	wr3val	LD	B,03h	; Zähler drei Werte
3EAB	7E	00727	wr1val	LD	A,(HL)	; Sekunden usw. laden
3EA9	D9	00728		EXX		; Register retten
3EAA	6F	00729		LD	L,A	; HL ← Sekunden
3EAB	2600	00730		LD	H,00h	; als 16-Bit-Wort
3EAD	3E0A	00731		LD	A,0ah	; Divisor 10
3EAF	CDB44C	00732		CALL	diva	; Sekunden/10, Einer in A
3EB2	D9	00733		EXX		; Register tauschen
3EB3	CD193F	00734		CALL	wrrtc	; RTC-Register schreiben
3EB6	D9	00735		EXX		
3EB7	7D	00736		LD	A,L	; Sekunden, Zehner
3EB8	D9	00737		EXX		
3EB9	CD193F	00738		CALL	wrrtc	; RTC-Register schreiben
3EBC	23	00739	incdec1	INC	HL	; Uhrzeit Minuten usw.
3EBD	10E9	00740		DJNZ	wr1val	; bis ss/mm/hh fertig
3EBF	167A	00741		LD	D,7ah	; ab RTC-Reg. 7 (Datum)
3EC1	3E2B	00742		LD	A,2bh	; = DEC HL
3EC3	32BC3E	00743		LD	(incdec1),A	; dort patchen
3EC6	2E46	00744		LD	L,46h	; Datum Tag (RAM 4046)
3EC8	0D	00745		DEC	C	; von Zeit auf Datum
3EC9	20DB	00746		JR	NZ,wr3val	; falls noch nicht geschr.
3ECB	AF	00747		XOR	A	; A ← 0, 0 Sek. (Quatsch)
3ECC	160A	00748		LD	D,0ah	; RTC-Reg. 0 schreiben
3ECE	CD193F	00749		CALL	wrrtc	; RTC-Schreibroutine
3ED1	CDBE06	00750		CALL	hsp	; auf 7,2 MHz umschalten
3ED4	DDE5	00751	rdclock	PUSH	IX	; retten
3ED6	DD212C3F	00752		LD	IX,clktab-1	; Ziffernmasken-Tabelle
3EDA	2E44	00753		LD	L,44h	; Datum Jahr (RAM 4044)
3EDC	0E02	00754		LD	C,02h	; 2 Durchl. wie oben, RD
3EDE	3E23	00755		LD	A,23h	; = INC HL
3EE0	32F53E	00756		LD	(incdec2),A	; dort patchen
3EE3	16CC	00757		LD	D,0cch	; ab RTC-Reg. 12 (Jahr)
3EE5	0603	00758	rd3val	LD	B,03h	; Zähler 3 Werte
3EE7	CD0A3F	00759	rd1val	CALL	rdrtc	; RTC auslesen (Zehner)
3EEA	87	00760		ADD	A,A	; *2
3EEB	5F	00761		LD	E,A	; sichern
3EEC	87	00762		ADD	A,A	; *4
3EED	87	00763		ADD	A,A	; *8
3EEE	83	00764		ADD	A,E	; *10
3EEF	5F	00765		LD	E,A	; sichern
3EF0	CD0A3F	00766		CALL	rdrtc	; RTC auslesen (Einer)
3EF3	83	00767		ADD	A,E	; + Zehnerstelle
3EF4	77	00768		LD	(HL),A	; in den Puffer (RAM 404x)
3EF5	2B	00769	incdec2	DEC	HL	; 1 Stelle tiefer
3EF6	10EF	00770		DJNZ	rd1val	; bis 3 Werte fertig
3EF8	165C	00771		LD	D,5ch	; ab RTC-Reg. 5 (Stunden)
3EFA	3E2B	00772		LD	A,2bh	; = DEC HL
3EFC	32F53E	00773		LD	(incdec2),A	; alten Befehl restaur.
3EFF	2E43	00774		LD	L,43h	; Zeit Stunden (RAM 4043)
3F01	0D	00775		DEC	C	; von Datum auf Zeit
3F02	20E1	00776		JR	NZ,rd3val	; falls noch nicht gelesen
3F04	DDE1	00777		POP	IX	; restaurieren
3F06	AF	00778		XOR	A	; A ← 00, RTC rücksetzen
3F07	D35B	00779		OUT	(5bh),A	; auf RTC-Adresseport
3F09	C9	00780		RET		
3F0A	7A	00781	rdrtc	LD	A,D	; RTC-Adresse, RD/WR-Bit

3F0B	D35B	00782	OUT	(5bh),A	;Register adressieren
3F0D	1610	00783	LD	D,10h	;1 tiefer im ober. Nibble
3F0F	92	00784	SUB	D	
3F10	57	00785	LD	D,A	;D <- neues Register
3F11	DD23	00786	INC	IX	;nächste BCD-Ziffernmaske
3F13	DB5A	00787	IN	A,(5ah)	;RTC Register auslesen
3F15	DDA600	00788	AND	(IX+00h)	;relevante Bits maskieren
3F18	C9	00789	RET		
3F19	5F	00790 wrrtc	LD	E,A	;Zeit/Datum retten
3F1A	7A	00791	LD	A,D	;RTC-Adreßreg., RD/WR-Bit
3F1B	D35B	00792	OUT	(5bh),A	;RTC-Register adressieren
3F1D	1610	00793	LD	D,10h	;1 höher im oberen Nibble
3F1F	82	00794	ADD	A,D	
3F20	57	00795	LD	D,A	;D <- neues Adreßregister
3F21	E6F0	00796	AND	0f0h	;nur oberes Nibble
3F23	FE60	00797	CP	60h	;Std.-Zehner adressiert?
3F25	7B	00798	LD	A,E	;Zeit/Datum
3F26	2002	00799	JR	NZ,setrtc	;falls anderes Digit
3F28	CBDF	00800	SET	3,A	;24h-Anzeige-Bit setzen
3F2A	D35A	00801 setrtc	OUT	(5ah),A	;Zeit stellen
3F2C	C9	00802	RET		
		00803	; Tabelle von Bitmasken für die Zeiteinheiten:		
3F2D	0F	00804 clktab	DB	0fh,0fh	;Jahr Zehner, Einer
3F2F	01	00805	DB	01h,0fh	;Monat Zehner, Einer
3F31	03	00806	DB	03h,0fh	;Tag Zehner, Einer
3F33	03	00807	DB	03h,0fh	;Stunde Zehner, Einer
3F35	07	00808	DB	07h,0fh	;Minute Zehner, Einer
3F37	07	00809	DB	07h,0fh	;Sekunde Zehner, Einer

00000 Fehler

MAILBOX - CORNER

News von der C.I.A. Mailbox in HB und anderen

Telefon: 0421 / 59 21 64, 24h, 300/300, 8/2/no

*Arnulf Sopp könnt ihr unter dem Usernamen 'HACKTOR7'
in der C.I.A. Mailbox erreichen ?*

Wenn die Uhr mal stört

Der UHR-Befehl unseres DOS ist gewiß eine feine Sache, aber die angezeigte Zeit übertüncht natürlich alles, was in der rechten oberen Bildschirmecke steht. Meist stört das nicht besonders. Im Augenblick aber läuft auf meiner Maschine TSCRIPS, und da hätte ich rechts oben ganz gerne den Text anstatt der Zeit. Rechts unten in der Kommandozeile ist dafür Platz genug. Es sind eine Menge weiterer Fälle denkbar, wo das wünschenswert ist.

Hier möchte ich zwei Versionen eines Programms vorstellen, das die Uhrzeit am Ende des Bildschirms anzeigen läßt. Um es ein bißchen komfortabler zu machen, hat es noch drei weitere Fähigkeiten. Gleich zu Beginn prüft es, ob beim Aufruf mit dem Namen UHRUNTEN noch ein Parameter angehängt wurde, z. B. UHRUNTEN,N für "nein". Ist das der Fall, so wird der Patch rückgängig gemacht und die Anzeigestelle wieder an die obere Bildschirmecke gerückt. Dabei ist es übrigens gleichgültig, welches Zeichen man dem Aufruf nachstellt. Zweitens wird automatisch der DOS-Befehl UHR durchgeführt; die Anzeige, wo auch immer, erfolgt also in jedem Falle.

Und drittens wird ein Mangel behoben, wo die Designer von NEWDOS-80 offenbar gepennt haben: Die Uhrzeit wird nicht ganz rechts angezeigt, sondern drei Stellen in die Zeile eingerückt. Das ist zwar nicht viel, aber es ist nicht einzusehen, was das soll. Deshalb wird - oben oder unten - der äußerste rechte Rand zur Anzeige benutzt. Das bleibt auch so, wenn mit UHRUNTEN,N wieder auf Anzeige oben zurückgeschaltet wird. Mit einem einfachen Patch kann das übrigens zurechtgebogen werden, ohne UHRUNTEN zu fahren:

Für das Genie I und seine Geschwister:
in SYS0/SYS, rel. Sekt. 01, Byte 92 von 35 auf 38 zappen

Für das Genie III s:
in OVL4/SYS, rel. Sekt. 09, Byte 53 von 0B auf 08 zappen

Zunächst möchte ich die für beide Versionen gleichen Teile erläutern. Der CALL nach 4CD5h überprüft, ob nach dem Programmnamen noch ein Parameter folgt. Wenn das der Fall ist, kehrt 4CD5h mit der NZ-Bedingung zurück, andernfalls mit Z. Das Entsprechende kann nun ablaufen. Am Ende des Programms wird HL als Zeiger auf das 0Dh-Byte (das entspricht ENTER) geladen. Was das bedeutet, kommt gleich. Danach werden C mit 02h und A mit E5h geladen. E5h ist der Requestcode, der über RST 28h SYS3/SYS aufruft. Der Inhalt von C entscheidet mit darüber, welche der Routinen aus SYS3/SYS angesprungen wird. Für UHR muß es 02h sein. Beim Einsprung in die UHR-Routine wird zuerst ebenfalls 4CD5h geCALLt, um die Parameter zu checken. Wenn in (HL) (Zeiger auf das Ende des Aufrufbefehls) nur ENTER (0Dh) steht, bedeutet das dasselbe wie UHR,J. Deshalb wird mit RST 28h die Uhr nun auf jeden Fall angezeigt, wenn HL auf 0Dh steht.

Nun zur Version für das Genie III s. Ab 3594h steht die Routine zur Anzeige der Uhrzeit. Dort wird u. a. die je nach Bildschirmformat zutreffende Stelle errechnet. Dabei werden vom Beginn der zweiten Zeile 11 Bytes subtrahiert, um nicht ganz ans Ende der ersten zu gelangen. Dieser Subtrahend steht in 35A3h. Dort schreibt das Programm nun stattdessen 8 ein, damit das äußerste Ende benutzt wird. An der Stelle 359Dh steht der Befehl LD DE,(340Bh). In 340Bh ist die Zeilenlänge je nach Bildschirmformat abgelegt. Die Summe von Bildschirmstart und Zeilenlänge ergibt den Beginn der zweiten Zeile (s. o.).

Zur Anzeige rechts unten muß aber die Summe von Bildschirmanfang und seiner Länge errechnet werden, um das Bildschirmende zu erhalten. Die Länge steht in 340Bh. Beide Adressen unterscheiden sich nur im LSB, das entweder 0Bh (oben) oder 08h (unten) lautet. Daher genügt es, wenn der Akku in Zeile 7 mit 0Bh oder in Zeile 10 mit 08h geladen wird, je nach dem. Dieses LSB wird nun nach 359Fh gapatcht.

Beim Genie I läuft es etwas anders. An der Stelle 44A4h wird HL als Zeiger auf die (fast) rechte obere Bildschirmecke, nämlich 3C35h geladen. Wir nehmen lieber 3C38h, um ganz rechts zu landen. Rechts unten lautet diese Adresse 3FF8h. Je nach Wunsch wird nun eine dieser beiden Adressen

nach DE geladen. In der Folge (Zeilen 11-13) patcht das Programm die gewünschte Adresse nach 44A5h, wo der Operand des Ladebefehls steht.

Etliche Clubmitglieder haben die 80-Zeichen-Karte von RB-Elektronik. Es wäre interessant, im nächsten Info eine Version zu finden, die mit dieser Karte läuft. Da es beim Komtek, dem Genie III (ohne "s") und ähnlichen Computern auch Unterschiede geben dürfte, sind wir auf weitere Versionen gespannt.

```

00001 ;                UHRUNTEN/CMD
00002 ; Programm zur Anzeige der Uhrzeit in der unteren Ecke
00003 ; Version für Genie I, TRS-80 usw.:
00004
5200      00005      ORG      5200h
5200 CDD54C 00006 start CALL   4cd5h      ;folgt ein Parameter?
5203 11F83F 00007      LD     DE,3ff8h    ;Anz. am Bildschirmende
5206 2803   00008      JR     Z,unten    ;falls kein Parameter
5208 11383C 00009      LD     DE,3c38h    ;rechte obere Ecke
520B 21A544 00010 unten LD     HL,44a5h   ;Adresse für Ladebefehl
520E 73     00011      LD     (HL),E    ;Adreß-LSB des Ladebef.
520F 23     00012      INC    HL          ;auf MSB erhöhen
5210 72     00013      LD     (HL),D    ;MSB patchen
5211 211A52 00014      LD     HL,cr      ;ENTER für UHR-Befehl
5214 0E02   00015      LD     C,02h     ;Zeiger auf UHR in SYS3
5216 3EE5   00016      LD     A,0e5h    ;Requestcode für SYS3
5218 EF     00017      RST    28h      ;aufrufen (= UHR<ENTER>)
5219 C9     00018      RET                    ;zur. ins Betriebssystem
521A 0D     00019 cr  DB     0dh      ;ENTER für UHR-Befehl
5200      00020      END     start

```

00000 Fehler

```

00001 ;                UHRUNTEN/CMD
00002 ; Programm zur Anzeige der Uhrzeit in der unteren Ecke
00003 ; Version für das Genie III s:
00004
5200      00005      ORG      5200h
5200 CDD54C 00006 start CALL   4cd5h      ;folgt ein Parameter?
5203 3E08   00007      LD     A,08h     ;für 3408h und 8 Stellen
5205 32A335 00008      LD     (35a3h),A  ;8 Stellen vor Zeilenende
5208 2802   00009      JR     Z,unten    ;falls kein Parameter
520A 3E0B   00010      LD     A,0bh     ;DE aus 430Bh laden
520C 329F35 00011 unten LD     (359fh),A  ;Adresse für Ladebefehl
520F 211852 00012      LD     HL,cr      ;ENTER für UHR-Befehl
5212 0E02   00013      LD     C,02h     ;Zeiger auf UHR in SYS3
5214 3EE5   00014      LD     A,0e5h    ;Requestcode für SYS3
5216 EF     00015      RST    28h      ;aufrufen (= UHR<ENTER>)
5217 C9     00016      RET                    ;zur. ins Betriebssystem
5218 0D     00017 cr  DB     0dh      ;ENTER für UHR-Befehl
5200      00018      END     start

```

00000 Fehler

Arnulf Sopp

Nachtrag zu "Sondertasten . . ."

In meinem Beitrag "Sondertasten diverser Genies" fragte ich, ob jemand schon herausgefunden habe, wie die Doppelnull erzeugt wird. Wieso habe ich mir diese Frage nicht selber gestellt? Inzwischen habe ich es nämlich raus:

Auf der nächsten Seite ist oben ein Listing abgedruckt. Der obere Teil ist die Routine, die die Doppelnull auf den Bildschirm bringt. Da wird zunächst die Anzahl der zusätzlich zu printenden Zeichen, hier 1, nach 4019h geladen. Dort ist normalerweise für die F-Tasten die Länge ihrer Strings abgelegt. Die Adresse des einen zusätzlichen Zeichens, 3E54h, wird nach 401Ah übertragen. Dort ist sonst der Beginn eines F-Strings gespeichert. Bei 3E47h geht es dann weiter, wo alles angezeigt wird.

Jede dieser Informationen kann geändert werden: Die um 1 verminderte Anzahl der Zeichen darf beliebig sein. Sie wird entsprechend nach 3E4Fh gepatcht. Das erste anzuzeigende Zeichen in 3E54h darf auch beliebig lauten. Schließlich kann man in 3E56/57h die Adresse des zweiten und der folgenden Zeichen ebenfalls nach Gusto ändern. Auf diese Weise ist es möglich, die 00-Taste wie eine zusätzliche F-Taste zu gebrauchen. Ihr String kann beliebig lang sein (solange er kürzer als 256 Bytes ist). Sie ist auch abschaltbar, wenn in 3E54h z. B. eine logische 00 oder ein anderer unwirksamer Code steht. In diesem Falle kann die 00-Taste z. B. während einer INT-Bearbeitung als Steuertaste für jedwede Funktion dienen, ohne daß sie außerhalb der INT-Routine stören würde.

Das Erkennen und Anzeigen der PRINT-Taste ist im unteren Teil des Listings wiedergegeben. Wie man sie abschalten oder anders belegen kann, habe ich bereits im o. g. Artikel beschrieben. Zusätzlich hier ein Tip, wie ihr ebenfalls ein beliebig langer Code zugeordnet werden kann: Die Anzahl der Zeichen ist in 3E6Eh abgelegt. Sie darf geändert werden. Die Adresse des Strings auch, die in 3E70h steht. Man definiert einen eigenen String und patcht seine Adresse dorthin.

Die gelistete Routine steht "neben" dem Bildwiederholungspeicher in der Nachbarbank. Den Änderungen muß deshalb ein entsprechendes Banking vorausgehen: Beim Genie III s muß beim Systembyte 1 (Port FAh) dazu das Bit 0 gesetzt werden. Hierzu genügt ein CALL 06A0h. Jetzt sind die Tastatur und der Bildschirm ausgeblendet, stattdessen ist das RAM zugänglich. Nach der Manipulation wird das Bit 0 wieder zurückgesetzt, wofür ein CALL 06ABh langt. Es empfiehlt sich, zuvor die Interrupts mit DI zu sperren, weil evtl. Tastatur- oder Bildschirmzugriffe (Uhr usw.) sonst das Nachbar-RAM adressieren.

In den beiden Sektordumps unter dem Listing ist alles unterstrichen, was hier zur Änderung vorgeschlagen wird. Das sind die Sektoren 04 und 05 von QVL4/SYS für das Genie III s, gebankte G-DOS-Version. Bei den anderen Overlays für die anderen Genies wird man die entsprechenden Codes leicht finden können, weil in ihrer Umgebung so ziemlich dasselbe stehen dürfte.

Arnulf Sopp

3E4E	<u>3E01</u>	00224	LD	A,01h	;1 zusätzliches Zeichen
3E50	321940	00225	LD	(fcount),A	;als Zeichenzähler ableg.
3E53	<u>3E30</u>	00226	LD	A,'0'	;zweite Null bei 00-Taste
3E54		00227	key00 EQU	\$-1	;zu verdoppelndes Zeichen
3E55	<u>21543E</u>	00228	LD	HL,key00	;Adr. des Doppelzeichens
3E58	221A40	00229	LD	(faddr),HL	;als Stringadresse ableg.
3E5B	18EA	00230	JR	m3e47	;dort weiter
		00231			
3E66	<u>FE3F</u>	00238	CP	'?'	;PRINT-Taste gedrückt?
3E68	C0	00239	RET	NZ	;falls nein
3E69	CD7E3D	00240	CALL	m3d7e	;Cursor aus?
3E6C	C8	00241	RET	Z	;falls ja
3E6D	<u>3E06</u>	00242	LD	A,06h	;6 Zeichen von PRINT
3E6F	<u>21753E</u>	00243	LD	HL,kyPRINT	;Belegg. der PRINT-Taste
3E72	<u>C3763D</u>	00244	JP	m3d76	;dort weiter
3E75	<u>50</u>	00245	kyPRINT DM	'PRINT '	;Belegung PRINT-Taste

00000 Fehler

DRV	00	003C	7E23	0528	0485	6F18	F732	1940	221A	<B#.(.o..2.5".
1	10	40AF	C94F	3A22	40B7	79C8	3A6E	43CB	4779	\$..0:"\$.y.:nC.Gy
1H	20	C9CD	993D	ED4B	4845	CD21	01C3	4A45	21B5	...=.KHE.!..JE!
	30	3DCD	2101	5FAE	73A3	200C	7AC6	0857	2379	=.!..e...z...W#y
DRS	40	C620	4F20	ECC9	E1C3	5E45	0000	0000	280E	..0.....^E....(.
864	50	2100	002B	22BC	3D7C	B53E	00C2	8845	2100	!..+".=6.>...E!.
360H60	0022	BC3D	C355	4521	0000	22BC	3D32	4F45	".=.UE!..."=20E	
	70	AFC9	219F	377A	2334	3528	05ED	A120	F77E	..!.7z#45(.B
	80	FE0B	2806	FE41	DBFE	5FD0	4FCD	003D	E642	..(..A...D...=.B
	90	79C8	E61F	FE0B	C001	4038	CD21	01E6	083E	y.....58.!...>
	A0	0BC8	3E5B	C9D6	3838	79C8	3DC8	2118	0136	..>A..88y.=.!...6
	B0	C9E5	CD24	3EE1	36CD	C9FE	0730	1207	4FCD	...\$>.6....0..0.
FRS	C0	0102	3C4F	003D	0F79	3001	3C21	7B3E	CD46	..<0.=.y0.<!ä>.F
4	D0	047A	C9C6	79FE	88DB	E67F	FE12	3007	C628	.z..y.... ..0..(
4H	E0	CD4B	047A	C920	0F3E	<u>0132</u>	1940	<u>3E30</u>	<u>2154</u>	.K.z...>.2.5>0!T
	F0	<u>3E22</u>	1A40	18EA	D613	C821	883E	CD35	3EFE	>".6.....!..>.5>.

DRV	00	<u>3FC0</u>	CD7E	3DC8	<u>3E06</u>	<u>2175</u>	<u>3EC3</u>	763D	<u>5052</u>	2..B=.>.!u>.v=PR
1	10	<u>494E</u>	<u>5420</u>	0000	1B1B	0A1A	0B1B	2B2B	0000	INT.....++..
1H	20	0000	2C2D	2E3F	AF32	2104	CD00	3DC3	0E04	...-.?.2!...=...
	30	CDB5	0621	4140	0E02	3E23	32BC	3E16	0A06	...!A5..>#2.>...
DRS	40	037E	D96F	2600	3E0A	CDB4	4CD9	CD19	3FD9	.B.o&.>...L...?.
865	50	7DD9	CD19	3F23	10E9	167A	3E2B	32BC	3E2E	ü...?#...z>+2.>.
361H60	460D	20DB	AF16	0ACD	193F	CDBE	06DD	E5DD	F.....?.....	
	70	212C	3F2E	440E	023E	2332	F53E	16CC	0603	!..?.D..>#2.>....
	80	CDOA	3F87	5F87	8783	5FCD	0A3F	8377	2310	..?._..._...?.w#.
	90	EF16	5C3E	2B32	F53E	2E43	0D20	E1DD	E1AF	..8>+2.>.C.....
	A0	D35B	C97A	D35B	1610	9257	DD23	DB5A	DDA6	.A.z.A...W.#.Z..
	B0	00C9	5F7A	D35B	1610	8257	E6F0	FE60	7B20	.._z.A...W... 'ä.
FRS	C0	02CB	DFD3	0110	3C50	SAC9	0F0F	010F	030F<PZ.....
5	D0	030F	070F	070F	010F	0645	00F5	3A05	45EEE...E.
5H	E0	2032	7304	F100	0001	0C5E	04ED	5B41	36F5	.2s.....^...AA6.
	F0	19F1	C342	3401	2780	04E5	ED5B	4136	B7ED	...B4.'.....AA6..

Neue Zeichensätze für das Genie IIIs

Eine sehr erfreuliche Eigenschaft des G3s ist es, daß sein Zeichensatz soft vorliegt und von der Systemdiskette (oder sonstwoher) geladen wird. Zur Auswahl stehen ab Werk vier verschiedene: STD (Standard deutsch), HIRES (verlängert, um beim Bildschirmformat 32 X 64 Zeichen normal auszusehen), ASCII (amerikanisch) und Tandy (so ähnlich). Die beiden Letztgenannten haben für die Praxis hierzulande keine nennenswerte Bedeutung. Also ärgerte ich mich nur darüber, daß STD und HIRES ästhetisch und praktisch ihre Nachteile hatten, die jetzt behoben sind:

Die Codes 01-1A waren nur eine Wiederholung des Großbuchstaben-Alphabets 41-5A. Nun sind sie unterstrichen. Man kann damit nicht nur unterstrichene Texte ausgeben, sondern auch etwa unter CP/M einen Code wie CTRL-C als zweifelsfrei unterscheidbares Zeichen ausgeben lassen.

Überflüssigerweise gab es zwei Cursorzeichen (1F und 5F), obwohl beim G3s der Cursor vom Video-Controller hard erzeugt wird. Sie flogen raus und sind jetzt das Grad-Zeichen (1F) und der griechische Buchstabe x (5F).

Die übrigen Änderungen, von denen es noch ungefähr zwanzig gibt, haben nur rein geschmackliche Bedeutung. So sind z. B. die unmögliche Form der großen Umlaute, des p und des q mit viel zu kleinen Bäuchen usw. gestreamlined. Im Zeichensatz HIRES, der noch phantasieloser als STD war, sind die Modifikationen etwas zahlreicher.

In den Dumps im Anschluß an diesen Artikel sind die neuen Zeichensätze dargestellt (zunächst STD, dann HIRES). In der ersten Spalte steht der ASCII-Code, danach die einzelnen Bytes für die Matrixzeilen des Zeichens, am Ende ist das Zeichen selbst mit Hilfe der Drucker-HRG ausgeprintet. Es liegt auf der Seite, weil mein Drucker nun mal nur neun Nadeln hat. Zwischen je 16 Zeilen ist eine Leerzeile, um die Grenzen der Disksectoren zu markieren.

Die neuen Zeichen unterscheiden sich von den alten mitunter nur durch ein Bit. Man muß deshalb schon sehr genau hinsehen, um die Unterschiede festzustellen. Wer die Modifikationen durchführen will, hat beim Abtippen reichlich zu tun. Deshalb möchte ich anbieten, den Interessenten die fertigen Zeichensätze auf Diskette zuzuschicken. Sendet mir hierzu bitte einen selbstadressierten Freiumschlag mit einer Leerdiskette (oder formatiert a la G-DOS 2.4).

Arnulf Sood

00 :	00	3C	42	99	A5	85	85	85	85	A5	99	42	3C	00	00	00	Q
01 :	00	08	08	14	14	22	22	3E	22	22	22	22	22	00	FF	00	R
02 :	00	1E	24	24	24	24	1C	24	24	24	24	24	1E	00	FF	00	S
03 :	00	1C	22	02	02	02	02	02	02	02	02	02	1C	00	FF	00	T
04 :	00	1E	24	24	24	24	24	24	24	24	24	24	1E	00	FF	00	U
05 :	00	3E	02	02	02	02	0E	02	02	02	02	02	3E	00	FF	00	V
06 :	00	3E	02	02	02	02	0E	02	02	02	02	02	02	00	FF	00	W
07 :	00	1C	22	22	02	02	32	22	22	22	22	22	1C	00	FF	00	X
08 :	00	22	22	22	22	22	3E	22	22	22	22	22	22	00	FF	00	Y
09 :	00	3E	08	08	08	08	08	08	08	08	08	08	3E	00	FF	00	Z
0A :	00	3E	20	20	20	20	20	20	20	20	20	22	1C	00	FF	00	[
0B :	00	22	22	12	12	0A	06	06	0A	12	12	22	22	00	FF	00]
0C :	00	02	02	02	02	02	02	02	02	02	02	02	3E	00	FF	00	^
0D :	00	22	36	36	2A	2A	22	22	22	22	22	22	22	00	FF	00	_
0E :	00	22	26	26	2A	2A	2A	2A	32	32	22	22	22	00	FF	00	`
0F :	00	1C	22	22	22	22	22	22	22	22	22	22	1C	00	FF	00	!
10 :	00	1E	22	22	22	22	22	1E	02	02	02	02	02	00	FF	00	"
11 :	00	1C	22	22	22	22	22	22	2A	1A	12	2C	00	FF	00	#	
12 :	00	1E	22	22	22	22	1E	06	0A	0A	12	22	22	00	FF	00	\$
13 :	00	1C	22	22	02	02	1C	20	20	20	22	22	1C	00	FF	00	%
14 :	00	3E	08	08	08	08	08	08	08	08	08	08	08	00	FF	00	&
15 :	00	22	22	22	22	22	22	22	22	22	22	22	1C	00	FF	00	'
16 :	00	22	22	22	22	22	22	22	22	14	14	08	08	00	FF	00	(
17 :	00	22	22	22	22	22	22	2A	2A	36	36	22	22	00	FF	00)
18 :	00	22	22	22	14	14	08	08	14	14	22	22	22	00	FF	00	!
19 :	00	22	22	22	22	14	14	08	08	08	08	08	08	00	FF	00	"
1A :	00	3E	20	20	10	10	08	08	04	04	02	02	3E	00	FF	00	#
1B :	00	08	08	1C	1C	2A	2A	08	08	08	08	08	08	00	00	00	\$
1C :	00	08	08	08	08	08	08	2A	2A	1C	1C	08	08	00	00	00	%
1D :	00	00	08	08	04	04	3E	04	04	08	08	00	00	00	00	00	'
1E :	00	00	08	08	10	10	3E	10	10	08	08	00	00	00	00	00	(
1F :	00	18	24	24	24	18	00	00	00	00	00	00	00	00	00	00)
20 :	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	!
21 :	00	08	08	08	08	08	08	08	08	00	00	08	08	00	00	00	"
22 :	00	14	14	14	14	00	00	00	00	00	00	00	00	00	00	00	#
23 :	00	14	14	14	3E	14	14	14	14	3E	14	14	14	00	00	00	\$
24 :	00	08	3C	0A	0A	0A	1C	28	28	28	1E	08	08	00	00	00	%
25 :	00	04	2A	24	10	10	08	08	04	04	12	2A	10	00	00	00	'
26 :	00	0C	12	12	12	12	0C	0C	4A	52	32	52	4C	00	00	00	(
27 :	00	18	18	18	08	04	00	00	00	00	00	00	00	00	00	00)
28 :	00	40	20	20	10	10	10	10	10	10	20	20	40	00	00	00	!
29 :	00	02	04	04	08	08	08	08	08	08	04	04	02	00	00	00	"
2A :	00	00	00	08	2A	2A	1C	1C	2A	2A	08	00	00	00	00	00	#
2B :	00	00	08	08	08	08	7F	08	08	08	08	00	00	00	00	00	\$
2C :	00	00	00	00	00	00	00	00	00	00	18	18	08	04	00	00	%
2D :	00	00	00	00	00	00	7E	00	00	00	00	00	00	00	00	00	'
2E :	00	00	00	00	00	00	00	00	00	00	00	0C	0C	00	00	00	(
2F :	00	20	20	10	10	08	08	08	04	04	02	02	00	00	00	00)
30 :	00	1C	22	32	32	2A	2A	2A	2A	26	26	22	1C	00	00	00	!
31 :	00	08	0C	0C	0A	0A	08	08	08	08	08	08	3E	00	00	00	"
32 :	00	1C	22	22	10	10	08	08	04	04	02	02	3E	00	00	00	#
33 :	00	1C	22	22	20	10	0C	10	20	20	22	22	1C	00	00	00	\$
34 :	00	10	18	18	14	14	12	12	7E	10	10	10	10	00	00	00	%
35 :	00	3E	02	02	02	02	0E	10	20	20	22	22	1C	00	00	00	'
36 :	00	18	04	04	02	02	1E	22	22	22	22	22	1C	00	00	00	(
37 :	00	3E	20	20	10	10	10	08	08	08	04	04	04	00	00	00)
38 :	00	1C	22	22	22	14	1C	22	22	22	22	22	1C	00	00	00	!
39 :	00	1C	22	22	22	22	3C	20	20	20	20	20	1C	00	00	00	"
3A :	00	00	00	00	0C	0C	00	00	00	00	00	0C	0C	00	00	00	#
3B :	00	00	00	00	18	18	00	00	00	00	18	18	08	04	00	00	\$
3C :	00	10	08	08	04	04	02	02	04	04	08	08	10	00	00	00	%
3D :	00	00	00	00	7E	00	00	00	7E	00	00	00	00	00	00	00	'
3E :	00	08	10	10	20	20	40	40	20	20	10	10	08	00	00	00	(
3F :	00	1C	22	22	20	10	10	08	08	00	00	08	08	00	00	00)

DDE für das Genie III s

Seit ein paar Tagen steht in meiner Gummizelle ein neuer Komposter namens G3s. Sein Betriebssystem, G-DOS 2.4, steckt zwar so manches andere in die Tasche, aber gewisse H-DOS-Features vermisse ich doch sehr. Bei der ewigen Zapperei ist es natürlich der Library-Befehl DDE in SYS15/SYS, den ich am meisten brauche. Und so ist denn logischerweise SYS15/SYS das erste File, das in Richtung H-DOS gestreamlined wurde:

Im Gegensatz zur Version für das G1/2 lädt diese SYS-Datei in den DOS-Overlay-Bereich. Man kann sie deshalb sogar unter Mini-DOS benutzen. Das Argument, man solle den Disketten-Editor lieber als CMD-File fahren, zieht deshalb beim G3 nicht mehr.

Beim Einsprung wird zunächst geprüft, ob der Speicherbereich ab 3000h benutzt werden darf oder gerade von einem anderen Modul des Betriebssystems belegt ist. Wenn er nicht frei ist, wird automatisch die Datei DDE52/SYS aufgerufen, die mit dem SYS15/SYS für das G1/2 praktisch identisch ist. Klarer Fall: In DDE52/SYS wurde sofort SYS15/SYS aus H-DOS mit geringfügigen Änderungen einkopiert. Diese Änderungen werden hier nicht näher benannt, denn sie erscheinen auch weiter unten im modifizierten SYS15/SYS für das G3s.

Wenn in 3000h ff. aber Platz ist, verlagert sich der Hauptteil der Datei dorthin. Das Bildschirmformat wird auf 16 X 64 Zeichen eingestellt und das Programm an seinem neuen Platz schließlich angesprungen.

Ein paar Programmschritte zuvor hakt die erste Änderung ein: Im M-Modus, wenn also ein Sektor modifiziert werden soll, erscheint ein Cursor. Das ist in der Originalversion ein Graphikblock, der das betreffende Zeichen natürlich überdeckt. Schade drum! Das G3s kann Zeichen invers darstellen. Also ruft die neue Version von DDE zunächst in SYS22/SYS den Befehl Z,YR auf, der von Graphik auf inverse Zeichen ab 80h umstellt. Das sind im anhängenden Programmlisting die Zeilen 58-64. Der Z-Befehl braucht dabei nicht zu erscheinen, da das Programm ökonomischer direkt mit RST 28h arbeitet. Nur das Argument YR, gefolgt von CR (ENTER), erscheint explizit an der Stelle invpar (Zeile 479, Parameter für Inversdarstellung). Übrigens sind diejenigen Teile des Listings, die hier nicht von Belang sind, mit LIST OFF ausgespart. Die Auslassungen sind durch Striche zwischen den Zeilennummern gekennzeichnet.

Die Änderung in den Zeilen 169-171 und 178-180 bezieht sich direkt auf die Form des Cursors: Das gerade angezeigte Zeichen wird jetzt nicht mehr vom Graphikblock mit dem Code 8Fh zugeschaufelt, sondern stattdessen wird sein Bit 7 invertiert, so daß das inverse Zeichen jetzt im Akku steht. Es wird statt seiner in den Bildschirm geladen.

Im gleichen Zusammenhang sind die Zeilen 380 und 381 zu sehen; der Code, der aus dem Sektorpuffer geholt wird (DE als Zeiger), wird direkt und ohne Änderungen in den Bildschirm geladen (HL als Zeiger). Da alle 256 Codes mit dem G3s anzeigbar sind, ist es nicht mehr erforderlich, einzelne von ihnen durch Punkte zu ersetzen.

Bis hierher beziehen sich die Modifikationen nur auf das optische Erscheinungsbild, sind also nur ebenso sinnvoll oder sinnlos wie ein auf Diskette sortiertes Directory (das beim nächsten Dateieintrag eh' wieder durcheinandergerät). Eine wirklich sinnvolle, geradezu notwendige Änderung ist in den Zeilen 300-328 programmiert:

Beim Umschalten mit CLEAR vom linken Hex- ins rechte ASCII-Feld geht im M-Modus der Cursor in der Originalversion immer in die jeweilige linke obere Ecke. Modifiziert man also gerade ziemlich weit unten im Sektor und will beispielsweise zwischen zwei Wörtern ein Steuerzeichen eingeben, das nicht über die Tastatur erreichbar ist, dann darf man zunächst mit den

Pfeiltasten Schiffeversenken spielen. In der hier vorgestellten Version geht der Cursor im Nachbarfeld immer genau an dasselbe Byte, so daß man sofort weitertippen kann. Und das geht so:

Beim Ansprung dieses Programmsegments steht die Cursoradresse gerade in HL. Der Anfang der aktuellen Bildschirmzeile wird durch UND-Verknüpfung des Cursor-LSB mit C0h gewonnen und vorübergehend in L verstaute. Jede Bildschirmzeile bezieht sich auf 16 Bytes des Sektorpuffers. Dessen Zeiger-LSB ist E. Mit 0Fh verknüpft, ergibt es die aktuelle Stelle in einem jeden Block von je 16 Bytes. Sie wird in B kühl und trocken aufbewahrt. Jetzt sind die passende Videozeile und das laufende Byte in ihr dingfest gemacht. Ein schlichtes LD A,(HL) o. ä. hätte leider nicht genügt, denn wäre der Cursor gerade im Hexfeld gewesen, hätte diese Operation nur ein Nibble erbracht, obendrein in ASCII-Darstellung.

Jetzt wird in Zeile 306 das LSB der aktuellen Videoadresse aus ihrem Puffer geladen. Nun geschieht der eigentliche Wechsel von ASCII nach Hex bzw. umgekehrt. Weshalb er ausgerechnet durch die Exklusiv-ODER-Verknüpfung mit 38h vollzogen wird, wäre einen eigenen Artikel über das Liebesleben der Bits wert. Freuen wir uns einfach, daß mir dieser kurze Weg eingefallen ist und reden wir nicht weiter drüber! Das neue Video-LSB wird anschließend in den Puffer geschrieben.

Für das Weitere muß das Programm nun wissen, ob das gerade ASCII oder Hex wurde. Wenn im LSB der Cursoradresse das Bit 4 oder 5 gesetzt war (= 30h), dann war es wohl das ASCII-Feld, denn im rechten Viertel jeder Bildschirmzeile sind diese Bits immer 1. Das LSB des Zeilenanfangs wird aus dem Datenfriedhof L in den Akku zurückgeladen. Falls ASCII, erfolgt nun der Sprung zum Label ascii (Zeile 325). Auch was dort geschieht, ist wieder binäre Akrobatik, die ich hier nicht erläutern will. Letztenendes kommt das neue Bildschirm-LSB in den halben Zeiger L, und die Arbeit ist getan.

Ein bißchen schwieriger ist es, wenn soeben ins Hexfeld umgeschaltet wurde. Die erste Hex-Ziffer erscheint an der achten Stelle, weil am Zeilenanfang zuerst der Kopf mit Sektor- und Zeilennummer steht. Vorab kommt dieses vorläufige Cursor-LSB nach L. Weiter oben war die Stelle in einem laufenden 16er-Block des Sektorpuffers nach B gerettet worden. Wenn es die nullte Stelle war, ist es damit erledigt. Andernfalls wird zunächst die halbe Videostellenzahl ermittelt, um sie später als Anzahl der trennenden Blanks hinzuaddieren zu können. Danach werden für jedes Byte im Puffer zwei Stellen addiert, denn es werden ja je zwei Nibbles angezeigt. Die Summe der Nibbles und Blanks ist schließlich das LSB der tatsächlichen Bildschirmadresse.

Uff! In einer letzten mörderischen Anstrengung habe ich schließlich noch in den Zeilen 411 und 412 klargemacht, daß dies eine neue Version von DDE ist, und wer da wann mitgemischt hat.

Abschließend ein Wort zum Listing. Da sich die Versionen von SYS15/SYS für G1/2 und G3/3s unterscheiden, mußte zunächst eine Disassembly erstellt und näher untersucht werden. Die große Mehrzahl der absoluten Adressen (bei CALLs, JPs, Speicherladebefehlen usw.) bezog sich darin auf den Bereich ab 3000h nach der Relokation des Hauptprogramms. Da aber die Modifikationen nahezu alle Adressen veränderten, mußte jede dieser Adressen auf die Ladeadresse vor dem Programmstart bezogen werden. Deshalb wurde das Label offs (Offset zur Arbeitsadresse) eingeführt. Als Subtrahend eingesetzt, korrigierte es alle Verschiebungen durch die Manipulationen.

Der Quelltext wurde mit ZEUS/CMD erstellt. Dieser Assembler erlaubt eine sehr bequeme Programmierung. Es dürfte für den EDTASM-gewohnten Assembler-Crack kein Problem sein, die Programmlogik zu verstehen.

Das komplette Listing des Programms SYS15/SYS wäre für unsere Postille nicht nur zu lang, sondern es würde sich garantiert keiner der Mühe unterziehen, es abzutippen. Recht so! Wenigstens ein Hexdump der ersten vier Sektoren ist beigefügt, damit der geneigte Leser und Selbstmodifizierer an der Umgebung erkennt, was sich wo gerade abspielt. Im Dump sind die Änderungen gegenüber Version 2.1 unterstrichen. Da Teile gelöscht, andere Teile hinzugefügt sind, finden sich die unterstrichenen Segmente nicht unbedingt an derselben Stelle wie im TCS-Original.

Wer für sein G3s die Version 2.2 haben will, mag mir einen frankierten und adressierten Umschlag mit einer Leerdiskette zuschicken (bitte mit der üblichen G-DOS-Formatierung, sonst Zettelchen nicht vergessen!). Ohne das G3 (ohne s) zu kennen, vermute ich, daß das Programm auch auf dieser Maschine ohne größere Änderungen laufen müßte.

Arnulf Sopp

ACHTUNG !
Einsendeschluß des Preisausschreibens ist der
30. 09. 1986

4D48	E5	00058	m4d48	PUSH	HL	;Befehlszeiger retten
4D49	21DF51	00059		LD	HL,invpar	;YR für Z-Befehl
4D4C	3EF8	00060		LD	A,0f8h	;für SYS22/SYS
4D4E	0E01	00061		LD	C,01h	;für Z,YR
4D50	CD0244	00062		CALL	4402h	;dort RST 28h
4D53	E1	00063		POP	HL	;Befehlsz. restaurieren
4D54	11D732	00064		LD	DE,m32d6-offs	;weiter wie gehabt
4E31	3E80	00169		LD	A,80h	;Bit 7 = 1
4E33	AE	00170		XOR	(HL)	;Zeichen invertieren
4E34	77	00171		LD	(HL),A	;und neu laden
4E3D	3E80	00178		LD	A,80h	;Bit 7 = 1
4E3F	AE	00179		XOR	(HL)	;Zeichen invertieren
4E40	77	00180		LD	(HL),A	;und neu laden
4F00	7D	00300	m31a8	LD	A,L	;LSB der Cursoradresse
4F01	E6C0	00301		AND	0c0h	;auf Zeilenanfang
4F03	6F	00302		LD	L,A	;zwischenspeichern
4F04	7B	00303		LD	A,E	;LSB Pufferstelle
4F05	E60F	00304		AND	0fh	;Stelle im 16er-Block
4F07	47	00305		LD	B,A	;zwischenspeichern
4F08	3ABD32	00306		LD	A,(m32bc-offs)	;Video-LSB
4F0B	EE38	00307		XOR	38h	;ASCII-Feld <-> Hexfeld
4F0D	32BD32	00308		LD	(m32bc-offs),A	;Video-LSB neu ablegen
4F10	E630	00309		AND	30h	;jetzt ASCII-Feld?
4F12	7D	00310		LD	A,L	;LSB d. Cursoradr. zurück
4F13	2011	00311		JR	NZ,ascii	;falls jetzt ASCII-Feld
4F15	F608	00312		OR	08h	;Hexfeld, auf B. Stelle
4F17	6F	00313		LD	L,A	;Hexfeld-Zeilenanfang
4F18	AF	00314		XOR	A	;A <- 00
4F19	B0	00315		OR	B	;0. Stelle im 16er-Block
4F1A	C8	00316		RET	Z	;erledigt, falls ja
4F1B	48	00317		LD	C,B	;C <- Stellenzahl
4F1C	CB39	00318		SRL	C	;die Hälfte = Blanks
4F1E	7D	00319		LD	A,L	;Zeilenanfang
4F1F	C602	00320	hex	ADD	A,02h	;je 2 Stellen/Byte
4F21	10FC	00321		DJNZ	hex	;fortgesetzt addieren
4F23	81	00322		ADD	A,C	;+ Anzahl der Blanks
4F24	6F	00323		LD	L,A	;neue Cursoradresse
4F25	C9	00324		RET		;erledigt
4F26	F630	00325	ascii	OR	30h	;auf letzte 16 Stellen
4F28	B0	00326		OR	B	;+ Stelle im 16er-Block
4F29	6F	00327		LD	L,A	;neue Cursoradresse
4F2A	C9	00328		RET		;erledigt
4F87	1A	00380	m4f58	LD	A,(DE)	;ASCII-Code laden
4F88	77	00381		LD	(HL),A	;unverändert anzeigen
4FBA	1C	00410	m324d	DB	1ch,1fh	
4FBC	47	00411		DM	'Genie-DOS Diskdaten-Editor Vers. 2.2'	
4FE0	0A	00412		DM	0ah,0ah,'(C) 1986 TCS/HACKTORY',0dh	
51D7	0000	00478	m51e0	DW	0000h,0000h,0000h,0000h	
51DF	59	00479	invpar	DM	'YR',0dh	
1D48		00480	offs	EQU	m4d48-3000h	

00000 Fehler

```

000000: 0102 004D 3A07 43E6 0FFE 023B 1BE5 214B AB5M:GC 0 BBA !H
000010: 4D11 0030 0100 04ED B021 0030 E33A 1419 MQ50A5D !50 :TY
000020: FE40 3E10 C233 00C9 3A6A 43CB 7F3E 38C0 5>P 35 :jC >8
000030: E501 0A00 213E 4D11 D751 D5ED B0D1 E1C3 AJ5!>MQ 0
000040: 3344 4444 4535 322F 5359 530D E521 DF51 3DDDE52/SYSM ! 0
000050: 3EF8 0E01 CD02 44E1 11D7 32CD 1C44 2827 > NA BD 0 2 5D('
000060: 2172 32CD 6744 21B0 32CD 6744 061B 21BE !r2 gD! 2 gDFX!
000070: 32CD 4000 D87E FE23 281A FE2A 2830 11D7 2 55 B #(Z *(0G
000080: 32CD 1C44 C251 3206 0021 0033 CD24 44C4 2 5D Q2F5!53 #D
000090: 5A32 1807 237E E60F 32DD 32ED 4BE1 3211 Z2XG#B 02 2 K 2Q
0000A0: D732 CD42 44CD 3644 C45A 32CD 4544 C00A 2 BD 6D Z2 ED J
0000B0: 32CD 4B32 FE01 28A8 FE3B 2836 FE2B 2815 2 K2 A( ;(6 +(U
0000C0: FE3D 281E FE2D 281F CBAF FE53 2839 FE4D =(^( -( _ S(9 M
0000D0: CAD7 3018 D9ED 4BE3 323A DF32 B720 C00B 0X K 2: 2 K
0000E0: 18BD 0100 0018 B8ED 4BE1 327B B12B BF0B X A55X K 2x ( K
0000F0: 18AD ED4B E132 032A E332 3ADF 32B7 2001 X K 2C* 2: 2 A

000100: 2BED 4230 0102 004E 9A18 A701 0000 CD4B + BOAB5N X A55 K
000110: 32FE 2138 E6C5 CDF7 31C1 38F2 6069 2929 2 !B 1 B 'i))
000120: 2929 854F 4418 E711 0033 263C 3ABD 326F )) ODX Q53&<: 2o
000130: C00A 327C E603 C63C 673E 80AE 773A BD32 J26 C <g> w: 2
000140: D608 2006 233E 80AE 772B 01E0 30C5 CD4B H F#> w+A 0 K
000150: 32FE 0928 46FE 0B28 64FE 0828 6CFE 0ACA 2 I(F K(d H(1 J
000160: A031 FE40 CAAC 31FE 1FCA BB31 FE0D CAE3 1 5 1 1 M
000170: 31FE 01CA F331 473A BD32 D608 7820 1ACD 1 A 1G: 2 Hx Z
000180: F731 38CA 7017 1717 174F CD4B 32CD F731 18 pWWWWD K2 1
000190: 38F8 2370 2B81 1218 0277 123A BD32 D608 B #p+ RXBwR: 2 H
0001A0: 2010 7B0F 7DCE 026F 1C7B E60F C001 1800 Pädü Boää 0 AX5
0001B0: 09C9 231C 7BE6 0FC0 0130 0009 C901 COFF I #ää 0 A05I A
0001C0: 097B D610 5F00 263F C93A BD32 D608 2013 Iä P_ &? : 2 H S
0001D0: 7B0F 7D3F DE02 6F1D 7BE6 0FEE 0FC0 01E8 ä0ü? Boüä 0 0 A
0001E0: FF09 C92B 1D7B E60F EEOF C001 DOFF 09C9 I +Üä 0 0 A I
0001F0: 0140 0009 7BC6 105F D026 3CC9 D53A BD32 A55Iä P_ &< : 2

000200: E620 121C 20FC D1C9 0102 004F 7DE6 C06F R6 AB50ü o
000210: 7BE6 0F47 3ABD 32EE 3832 BD32 E630 7D20 ä 0G: 2 B2 2 0ü
000220: 11F6 086F AF80 C848 CB39 7DC6 0210 FC81 0 Ho H 9ü BF
000230: 6FC9 F630 B06F C9C1 11D7 32CD 3944 C45A o 0 o 0 2 9D Z
000240: 32CD 4544 C362 30C1 C34F 3047 D630 DBFE 2 ED b0 00G 0
000250: 0A38 09CB AFFE 11DB D607 FE10 3FC9 D921 JBI 0 G P? !
000260: 003C 1100 33D5 0608 3AE2 32CD 6840 3AE1 5<Q53 FH: 2 h5:
000270: 32CD 6840 7BCD 6840 3E3A 7723 3E20 7723 2 h5ä h5>:w#> w#
000280: 1ACD 6840 1C1A CD68 401C 10F0 3E20 7723 Z h56Z h56F > w#
000290: D106 101A 7723 1C28 0410 F818 C8D9 C9D5 FPZw#6(DP X
0002A0: CD49 00D1 C921 B932 CD67 44C3 1A30 F6C0 15 ! 2 gD Z0
0002B0: FEC6 2808 3333 CD09 44C3 1A30 3AD7 32F6 (H33 ID Z0: 2
0002C0: 0132 D732 AFC9 1C1F 4765 6E69 652D 444F A2 2 6_Genie-DO
0002D0: 5320 4469 736B 6461 7465 6E2D 4564 6974 S Diskdaten-Edit
0002E0: 6F72 2056 6572 732E 2032 2E32 0A0A 2843 or Vers. 2.2JJ(C
0002F0: 2920 3139 3836 2054 4353 2F48 4143 4B54 ) 1986 TCS/HACKT

000300: 4F52 590D 0A44 6174 6569 3F20 0102 0050 DRYMJDatei? AB5P
000310: 0307 1B1B 0330 3031 3233 3435 3637 3839 CGAÄC00123456789
000320: 4142 4344 4546 3031 3233 3435 3637 0DB2 ABCDEF01234567M
000330: 2800 0033 0000 0000 0000 00FF FFFF FFFF (5535555555
000340: FFFF FFFF FFFF FFFF FFFF FFFF FFFF FF00 5
000350: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
000360: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
000370: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
000380: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
000390: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
0003A0: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
0003B0: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
0003C0: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
0003D0: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
0003E0: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555
0003F0: 0000 0000 0000 0000 0000 0000 0000 0000 5555555555555555

```

HRG-Hardcopy für das G I

Mit der JKL-Erweiterung des H-DOS für das Genie I wurde der Bildschirm in seinen richtigen Proportionen wiedergegeben. Bei diesem Programm war das nicht möglich. Die vier verschiedenen Bildschirmformate des II hätten jedes eine eigene Programmvariante erfordert, welchen Aufwand aber der Erfolg nicht mehr rechtfertigen würde. Hier wurde ein Kompromiß gewählt: Eine anderthalbfache Dehnung in der Breite läßt die Formate 16X64, 24X64 und 25X80 Zeichen breiter als der Bildschirm erscheinen, während sich beim Format 32X64 ein leichtes Hochformat ergibt. Die Vereinhalfachung wird dadurch erreicht, daß abwechselnd ein Graphikcode einmal, der nächste zweimal ausgegeben wird usw..

Die Hardcopy-Routine arbeitet mit der hochauflösenden Graphik doppelter Dichte des Gemini-10X. Für andere Drucker müssen entsprechende andere SteuerCodes ausgegeben werden. Um dem Leser dies zu ermöglichen, werden die druckerspezifischen Teile des Programms im folgenden näher erläutert. Das soll auch das Hauptanliegen dieses Artikels sein, denn zweieinhalb Seiten Listing eines immerhin recht anspruchsvollen Programms einigermaßen erschöpfend zu erklären, würde den Rahmen dieses Beitrags sprengen.

Da der Drucker nur neun Nadeln hat, von denen im Bit-Image-Modus nur acht gleichzeitig angesprochen werden können, werden pro Bildschirmzeile auf den Drucker zwei Halbzeilen ausgegeben. Die obere umfaßt immer acht senkrecht übereinanderstehende Bildpunkte, die untere den Rest der Bildschirmzeile, der je nach Format unterschiedlich hoch sein kann. Dieses Verfahren habe ich im Hinblick auf eine spätere Einbeziehung der ASCII-Anteile des Bildschirms gewählt (s. u.), die ja bildschirmzeilenorientiert sind. Für die Graphik allein wäre es auch simpler gegangen.

Dieses Programm druckt nur die HRG des IIs aus. Die ASCII-Anteile bzw. die Blockgraphik, Inverssschrift usw. bleiben unberücksichtigt. Da die Zeichensätze von der Diskette geladen werden, müßte ansonsten die Routine den Benutzer zunächst fragen, welcher denn gerade angezeigt wird, der müßte in einen Puffer geschrieben, seine Alternativzeichen oberhalb 80h errechnet und ebenfalls gepuffert werden. Nein, danke! Sobald die Garantie meiner neuen Denkprothese abgelaufen ist, wird das Zeichensatz-RAM für die CPU lesbar gelötet, so daß die Zeichen einfach mit der HRG auf dem Drucker gemischt werden können.

Da das Programm 244 Bytes lang ist, paßt es bequem in den Sektorpuffer. Daher die Ladeadresse 4204h. Um für Pufferbereiche, den Stack usw. keinen zusätzlichen Platz zu benötigen, werden zu diesem Zweck die Programmteile am Anfang überschrieben, denn sie dienen nur zu Initialisierungen und sind anschließend überflüssig. Auf diese Weise kann das Programm jederzeit und mit jedem Betriebssystem laufen, denn es belegt keinen Raum im Anwenderspeicher. Da es seinen eigenen Stack unterhält, läuft es auch unter BASIC. Das wäre sonst nicht möglich, weil der BASIC-Stack im Adreßbereich der Graphik liegt.

Zu den druckerspezifischen Eigenschaften des Programms: Es wird davon ausgegangen, daß das höchstwertige Bit des Druckkopfes oben liegt. Für Drucker, bei denen das umgekehrt ist (im Handbuch nachschlagen!), sind folgende Änderungen erforderlich: In Zeile 129 muß der Befehl RL (HL) in RR (HL) umdefiniert werden. Die AND-Maske, die für die zweite Halbzeile überflüssige Punkte löscht, wird dann auch in Zeile 37 nicht mehr mit RRA, sondern mit RLA ermittelt. Ohne das jetzt selbst nachprüfen zu können, scheint mir, daß dieser Punkt damit erledigt ist.

Weitere Druckerspezifika finden sich in der Tabelle am Ende des Listings. Bei Label lpinit wird der Drucker vorsichtshalber in den Einschaltzustand versetzt, weil evtl. zuvor schon eine bestimmte Randeinstellung o. ä. vorgegeben war. Beim Gemini geht das mit ESC '9'. Das

nachfolgende ODH-Byte ist ein gewöhnlicher Wagenrücklauf (der in der neuesten Version herausgenommen ist), das abschließende FFh gibt lediglich dem Programm bekannt, daß der auszudruckende String hier zuende ist. Mit ESC 'M' 9 wird der linke Rand auf die 9. Stelle festgelegt. Bei Epson-Typen geschieht das mit ESC '1' 9.

Beim Label Ininit erscheint zunächst wieder ein Wagenrücklauf, dann die Einstellung des Zeilenabstandes mit ESC '3' n auf n/144 Zoll. Das Nullbyte ist nur ein Platzhalter für n, das jeweils neu definiert und dort eingeschrieben wird (Label inby144). Beim Gemini ist der senkrechte Abstand zweier Punkte 1/72, also 2/144 Zoll. Deshalb wird für eine Spalte mit allen 8 Dots das Register C in Zeile 63 mit 10h (16d) geladen, um $16/144 = 8/72$ Zoll = 8 Punkte Zeilenabstand zu definieren. In der zweiten Halbzeile ist der Abstand je nach Bildschirmformat variabel. Er wird in den Zeilen 30-33 errechnet. Anpassungen für andere Drucker müssen hier ansetzen.

Die HRG doppelter Dichte des Druckers wird in Zeile 158 mit ESC 'L' n1 n2 initialisiert. Dabei bedeuten n1 und n2 in der Reihenfolge LSB-MSB die Anzahl der zu druckenden Dotspalten pro Zeile. Das MSB ist für jedes Format dasselbe (03h), aber das LSB (Label nrcols) variiert. Anpassungen an andere Matrixdrucker sind an dieser Stelle wohl am schwierigsten. Leser, die das Schicksal mit einem ITOH oder NEC geschlagen hat, beneide ich nicht, so gut diese Geräte in anderer Hinsicht auch sind.

Am Ende des Listings ist ein Bildschirm Ausdruck des Files RDLDEMO wiedergegeben. Es ist eine Phase vor der Anzeige einer neuen Graphik, daher das unmotivierete "und" am Fuße. Die übrigen Ausdrücke sind weniger "schön", aber aussagekräftiger. Sie entstanden der Reihe nach mit den Bildschirmformaten 16X64, 24X64, 25X80 und 32X64 Zeichen. Man erkennt, daß nicht bei allen Formaten alle Scan-Zeilen in der Graphik angezeigt werden. Das ist nur bei 32X64 Zeichen der Fall. Hier gehen die Schrägstriche ohne Unterbrechung durch. In den übrigen Formaten werden mehr oder weniger auffällige "Treppen" gebildet. Es ist auch zu sehen, daß ein Graphikpunkt abwechselnd einmal bzw. zweimal ausgedruckt wird (unterschiedlich dick erscheinende Punkte). Jedenfalls hoffe ich, daß der Leser bei seiner Kopie einer Kopie einer Kopie ... es noch erkennen kann.

Arnulf Sopp

Infotip

Der Drucker NEC 8823 B-C druckt lt. Betriebsanleitung mit einer Geschwindigkeit von 100 Zeichen / Sekunde. Die Mechanik ist baugleich mit dem ITOH 8510 und trägt eine Druckgeschwindigkeit bis 160 Zeichen / Sekunde. Durch einfachen Austausch des Quarzes kann somit der Ausdruck beschleunigt werden. Der Originalquarz hat einen Wert von 6,144 MHz; der neue Quarz 7,3728 MHz oder 8,000 MHz. Ich hab's probiert; es funktioniert einwandfrei !
Peter Spieß

```

00001 :*****
00002 ;      Hardcopy der hochauflösenden Graphik
00003 ;      des Genie I
00004 ;      zentriert für alle Bildschirmformate
00005 ;
00006 ;      (C) 1986 by The HACKTORY Arnulf Sopp
00007 :*****
00008
4204   00009      ORG      4204h      ;im Sektorpuffer
00010
00011 ;Drucker initialisieren, Bildschirmformat feststellen,
00012 ;Programm entsprechend modifizieren, Graphik zuschalten
4204   21F042 00013 start LD      HL,lpinit      ;Codes zur Druckerinit.
4207   CD8F42 00014 CALL   prtstr      ;auf den Drucker ausgeben
420A   3AF137 00015 LD      A,(37f1h)   ;Zeichen pro Zeile
420D   325F42 00016 LD      (chrpl1),A  ;als Zähler ablegen
4210   327842 00017 LD      (chrpl2),A  ;und als Summand
4213   CB67    00018 BIT     4,A         ;80 Zeichen?
4215   2007    00019 JR      NZ,no64     ;falls ja
4217   AF      00020 XOR     A           ;AK-00 = LSB von 64*8*1.5
4218   32ED42 00021 LD      (nrcols),A  ;LSB Anzahl Dotspalten
421B   CD8F42 00022 CALL   prtstr      ;Randeinst. bei 64 Zeich.
421E   3AF637 00023 no64 LD      A,(37f6h)  ;Zeilen pro Seite
4221   21F837 00024 LD      HL,37f8h    ;Bit 0 = Interlace-Modus
4224   CB46    00025 BIT     0,(HL)     ;32 Zeilen pro Seite?
4226   2801    00026 JR      Z,skipilc  ;falls nein
4228   07      00027 RLCA           ;Zeilen/Seite verdoppeln
4229   325542 00028 skipilc LD      (linppag),A ;Zeilenzähler dort ableg.
422C   23      00029 INC     HL          ;HL ← 37F9h
422D   7E      00030 LD      A,(HL)     ;Scan-Zeilen pro Zeichen
422E   D607    00031 SUB     07h        ;+ 1 - 8 = 2. Halbzeile
4230   47      00032 LD      B,A        ;Dotzeilen der 2. Halbz.
4231   07      00033 RLCA           ;verdoppeln
4232   327342 00034 LD      (linpchr),A ;für Druckerzeilenabstand
4235   AF      00035 XOR     A           ;Akkus löschen
4236   37      00036 masklop SCF          ;Cy auf 1 setzen
4237   1F      00037 codbuf RRA           ;Akkus: ab Bit 7 auf 1
4238   10FC    00038 DJNZ   masklop     ;AND-Maske errechnen
423A   327142 00039 LD      (andmask),A ;AND-Operanden anpassen
423D   3AFC37 00040 LD      A,(37fch)  ;rel. Anfang Bildschirm
4240   CBFF    00041 SET     7,A        ;auf Graphik umrechnen
4242   325842 00042 LD      (stgra),A  ;Anfang Graphikspeicher
4245   ED738942 00043 LD      (spbuff),SP ;Stackpointer retten
4249   F3      00044 DI           ;keine Störungen
424A   313642 00045 LD      SP,masklop ;eigener Stack
424D   DBFA    00046 IN      A,(0fah) ;Systembyte 1 einlesen
424F   F5      00047 PUSH   AF         ;alten Zustand retten
4250   CBDF    00048 SET     3,A        ;Graphik für CPU
4252   D3FA    00049 OUT     (0fah),A ;zugänglich machen
00050
00051 ;äußere Schleife: 16, 24, 25 oder 32 Zeilen ausdrucken
4254   0600    00052 LD      B,00h      ;Zähler für n Zeilen
4255           00053 linppag EQU   $-1      ;je nach Bildschirmformat
4256   210000 00054 LD      HL,0000h  ;Adr. der 1. Graphikzeile
4258           00055 stgra EQU   $-1      ;MSB je nach Bildschirmf.
4259   110008 00056 LD      DE,0800h  ;Summand 1 Dotsl. (2 kB)
425C   C5      00057 pagloop PUSH   BC       ;Zeilenzähler retten
425D   E5      00058 PUSH   HL         ;Graphikzeiger retten
425E   0600    00059 LD      B,00h      ;Zeichen pro Zeile
425F           00060 chrpl1 EQU   $-1      ;je nach Bildschirmformat
4260   C5      00061 PUSH   BC       ;Zeichenzähler retten
4261   3EFF    00062 LD      A,0ffh    ;AND-Maske: alle Dots
4263   0E10    00063 LD      C,10h     ;Zeilenabst. Dots*2/144
4265   CDA342 00064 CALL   oneline   ;1. Halbzeile ausgeben
4268   C1      00065 POP    BC        ;Zeichenzähler
4269   E1      00066 POP    HL         ;Graphikzeiger

```

```

426A E5      00067      PUSH      HL      ;wieder retten
426B 1640    00068      LD        D,40h   ;Smd. f. 16 kB (9. Dotz.)
426D 19      00069      ADD      HL,DE    ;untere Halbzeile
426E 1608    00070      LD        D,08h   ;Summand für 2 kB
4270 3E00    00071      LD        A,00h   ;AND-Maske in den Akku
4271        00072 andmask EQU      $-1      ;je nach Bildschirmformat
4272 0E00    00073      LD        C,00h   ;Druckzeilenabstand
4273        00074 linpchr EQU     $-1      ;je nach Bildschirmformat
4274 CDA342   00075      CALL     oneline   ;2. Halbzeile ausgeben
4277 0E00    00076      LD        C,00h   ;Zeichen pro Zeile
4278        00077 chrpl2 EQU     $-1      ;je nach Bildschirmformat
4279 E1        00078      POP      HL      ;Graphikzeiger
427A 09      00079      ADD      HL,BC    ;nächste Graphikzeile
427B C1        00080      POP      BC      ;Zeilenzähler restaur.
427C 3A4038  00081      LD        A,(3840h) ;Tastatur abfragen
427F CB57    00082      BIT      2,A      ;BREAK gedrückt?
4281 2002    00083      JR        NZ,exit  ;falls ja
4283 10D7    00084      DJNZ     pagloop  ;bis Hardcopy fertig
          00085
          00086 ;alten Zustand wiederherstellen, zurück ins Betriebssystem.
4285 F1        00087 exit   POP      AF      ;altes Systembyte 1
4286 D3FA    00088      OUT     (0fah),A  ;restaurieren
4288 310000  00089      LD        SP,0000h ;Stackpointer restaur.
4289        00090 spbuff EQU     $-2      ;(bis jetzt hier gepuff.)
428B FB        00091      EI        ;INTs wieder zulassen
428C 21F042  00092      LD        HL,1pirfit ;Drucker neu initialis.
          00093
          00094 ;einen String ab (HL) auf den Drucker ausgeben
428F 7E        00095 prtstr LD        A,(HL)   ;Druckzeichen laden
4290 23      00096      INC     HL        ;nächste St. des Strings
4291 FEFF    00097      CP      0ffh     ;String zuende?
4293 C8        00098      RET     Z        ;zurück, falls ja
4294 CD9942  00099      CALL    print    ;nein, ausdrucken
4297 18F6    00100      JR        prtstr  ;nächstes Byte
          00101
          00102 ;ein Byte auf den Drucker ausgeben
4299 F5        00103 print  PUSH     AF      ;retten, wird verändert
429A CDD105  00104 wait  CALL     05d1h   ;Drucker bereit?
429D 20FB    00105      JR        NZ,wait ;falls noch nicht
429F F1        00106      POP     AF      ;Code restaurieren
42A0 D3FD    00107      OUT     (0fdh),A ;auf Drucker ausgeben
42A2 C9        00108      RET
          00109
          00110 ;eine Druckzeile initialisieren
42A3 E5        00111 oneline PUSH     HL      ;Graphikzeiger retten
42A4 32CD42  00112      LD        (andoprd),A ;AND-Maske setzen
42A7 21EA42  00113      LD        HL,inby144 ;Zeilenabstand in Dots
42AA 71        00114      LD        (HL),C   ;im Codestring anpassen
42AB 21E742  00115      LD        HL,lninit ;Codes für Zeileninitial.
42AE CD8F42  00116      CALL     prtstr   ;einer Druckzeile ausgeb.
42B1 E1        00117      POP     HL      ;Graphikzeiger restaur.
          00118
          00119 ;16-, 24-, 25- o. 32mal 8 Dotspalten auf den Dr. ausgeben
42B2 E5        00120 onechr  PUSH     HL      ;Graphikzeiger retten
42B3 C5        00121      PUSH     BC      ;Zeichenzähler retten
42B4 42      00122      LD        B,D     ;8 Dotzeilen (D = 08)
42B5 C5        00123 onecpl1 PUSH     BC      ;Dotzeilen-Zähler retten
42B6 E5        00124      PUSH     HL      ;Graphikzeiger retten
42B7 7E        00125      LD        A,(HL)  ;Graphikbyte in den Akku
42B8 42      00126      LD        B,D     ;Zähler für 8 Spalten
42B9 213742  00127      LD        HL,codbuf ;Zeiger auf Graphikpuffer
42BC 07        00128 onecpl2 RLCA    ;Cy ← je das oberste Bit
42BD CB16    00129      RL        (HL)   ;in den Puffer rotieren
42BF 23      00130      INC     HL        ;nächste Pufferstelle
42C0 10FA    00131      DJNZ     onecpl2 ;bis 1 Graphikbyte fertig
42C2 E1        00132      POP     HL      ;Graphikzeiger

```

```

4203 01      00133      POP      BC      ;Zeilenzähler
4204 19      00134      ADD      HL,DE   ;nächste Graphikzeile
4205 10EE    00135      DJNZ    oneclp1 ;bis alle Dotzeilen fert.
4207 213E42 00136      LD      HL,codbuf+7 ;Ende des Graphikpuffers
420A 42      00137      LD      B,D     ;Zähler 8 Pufferstellen
420B 7E      00138      oneclp3 LD      A,(HL)  ;1 Pufferbyte in den Akku
420C E600    00139      AND     00h     ;je nach Halbzeile mask.
420D        00140      andoprnd EQU    #-1    ;(Msk. je nach Bildschf.)
420E CD9942 00141      CALL   print    ;Graphikbyte ausdrucken
42D1 1803    00142      JR     onecol   ;Akku abwechselnd
42D2        00143      jroffs EQU    #-1 ;einmal oder zweimal
42D3 CD9942 00144      twocol CALL   print    ;auf Drucker ausgeben
42D6 3AD242 00145      onecol LD      A,(jroffs) ;JR-Distanz undefinieren
42D9 EE03    00146      XOR    onecol-twocol ;neuen Offset
42DB 32D242 00147      LD     (jroffs),A ;im Distanzbyte ablegen
42DE 2B      00148      DEC    HL      ;nächste Pufferadresse
42DF 10EA    00149      DJNZ   oneclp3 ;bis Puffer ausgedruckt
42E1 01      00150      POP    BC      ;Zeichenzähler restaur.
42E2 E1      00151      POP    HL      ;Graphikzeiger
42E3 23      00152      INC    HL      ;nächste Graphikadresse
42E4 10CC    00153      DJNZ   onechr  ;bis 1 Halbzeile fertig
42E6 09      00154      RET
00155
00156 ;ESC-Sequenzen für die HRG und andere Drucker-Features
42E7 0D      00157      lninit DB    0dh,1bh,'3' ;CR, Einst. Zeilenabst.
42EA 00      00158      inby144 DB 00h,1bh,'L' ;und der Drucker-HRG
42ED C0      00159      nrcols DB    0c0h,03h,0ffh ;Anzahl der Dotspalten
42F0 1B      00160      lpinit DB    1bh,'8',0dh,0ffh ;Drucker neu initialis.
42F4 1B      00161      DB     1bh,'M',09h,0ffh ;li. Rand auf 9. Stelle
00162
4204        00163      END     start   ;dort Einsprung

```

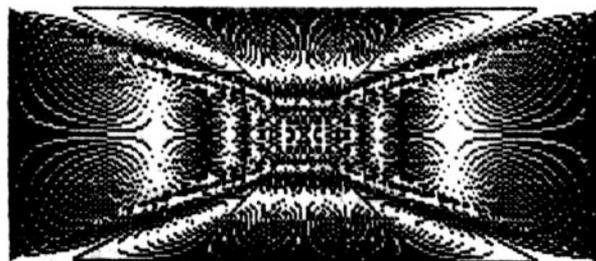
00000 Fehler

```

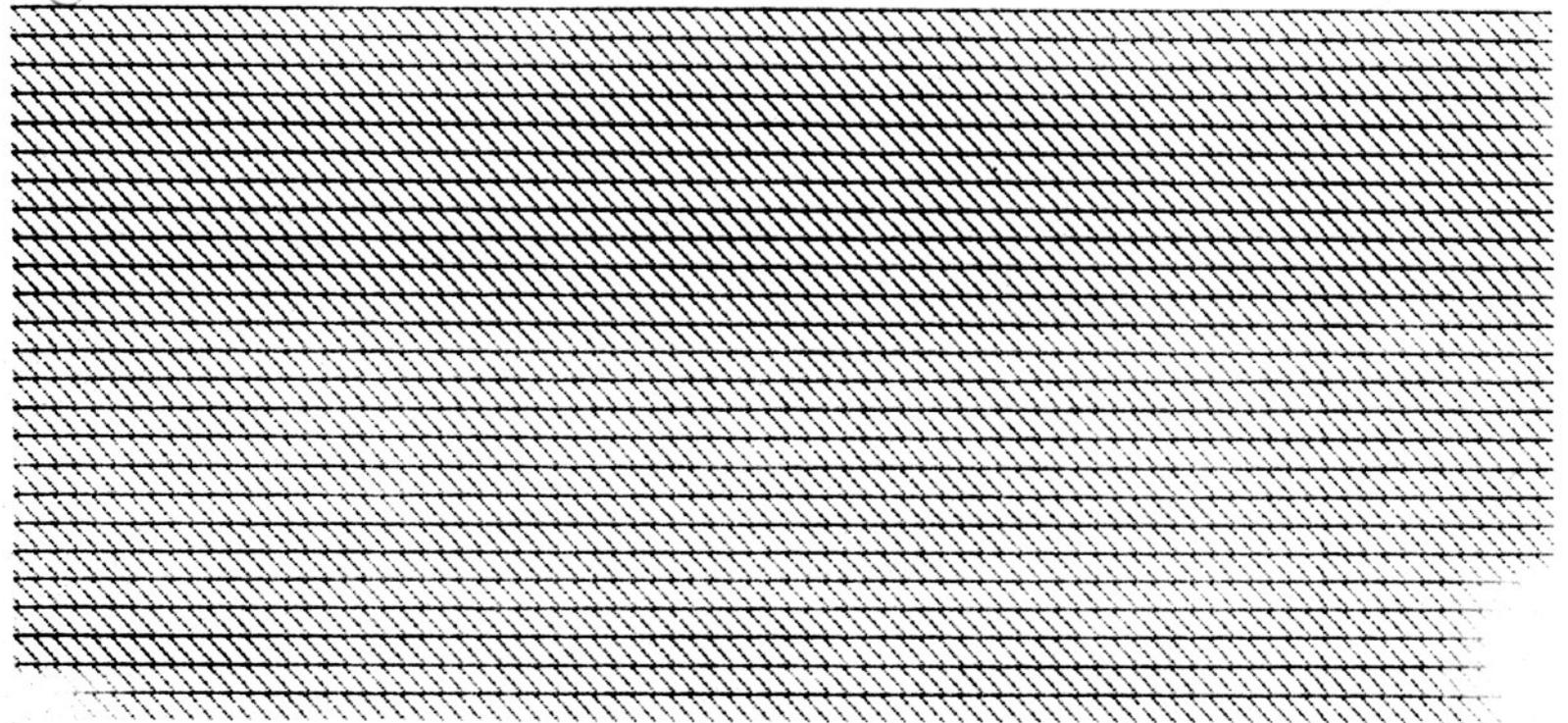
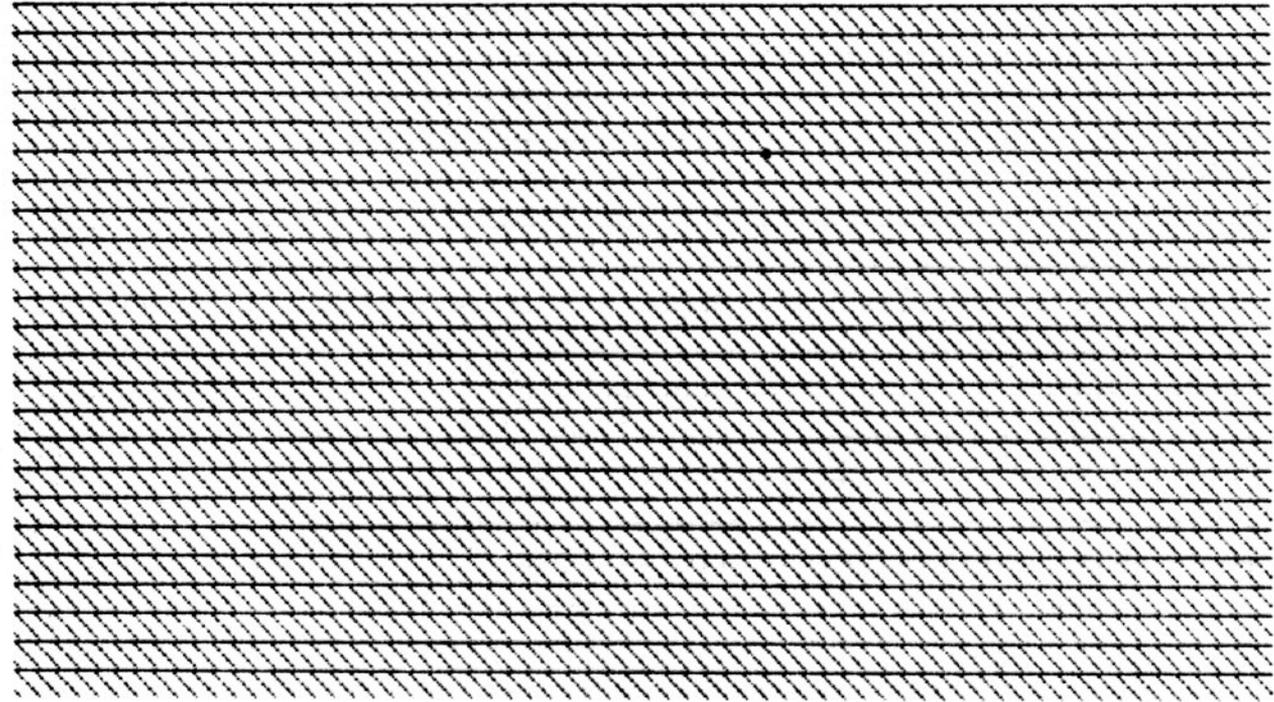
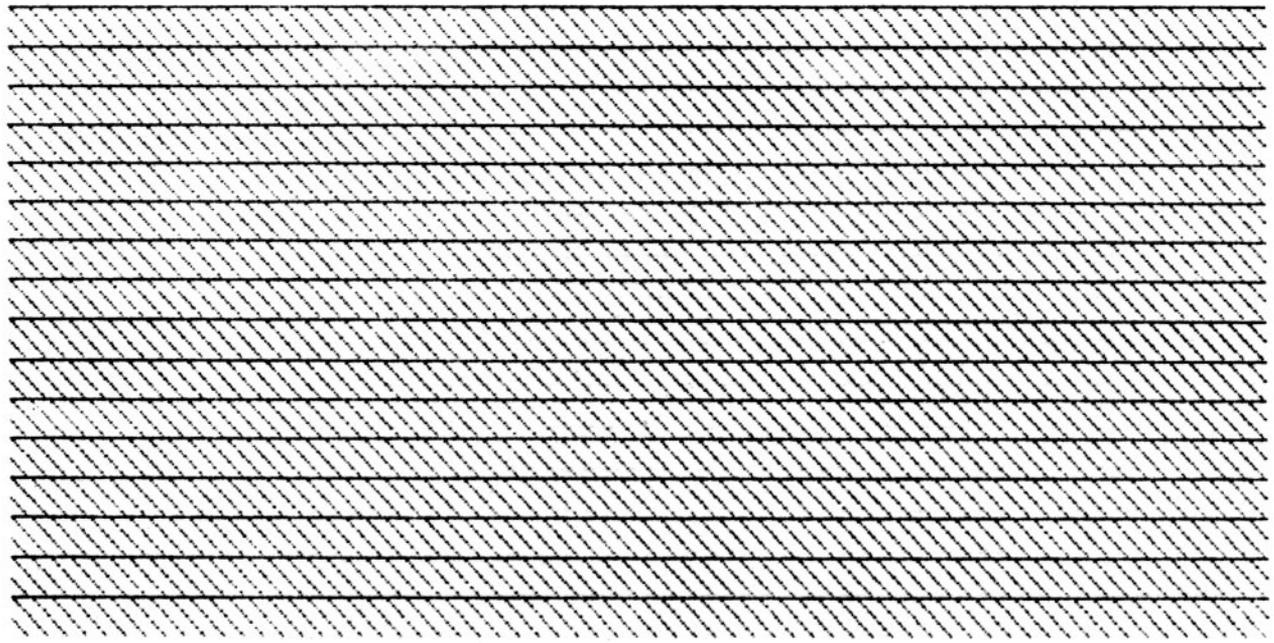
andmask 4271      andoprnd 42CD      chrpl1 425F      chrpl2 4278      codbuf 4237
exit     4285      inby144 42EA      jroffs 42D2      linpchr 4273      linppag 4255
lninit  42E7      lpinit  42F0      masklop 4236     no64    421E      nrcols 42ED
onechr  42B2      oneclp1 42B5      oneclp2 42BC     oneclp3 42CB     onecol 42D6
oneline 42A3      pagloop 425C     print   4299     prtstr  428F     skipilc 4229
spbuff  4289      start   4204     stgra   4258     twocol 42D3     wait   429A

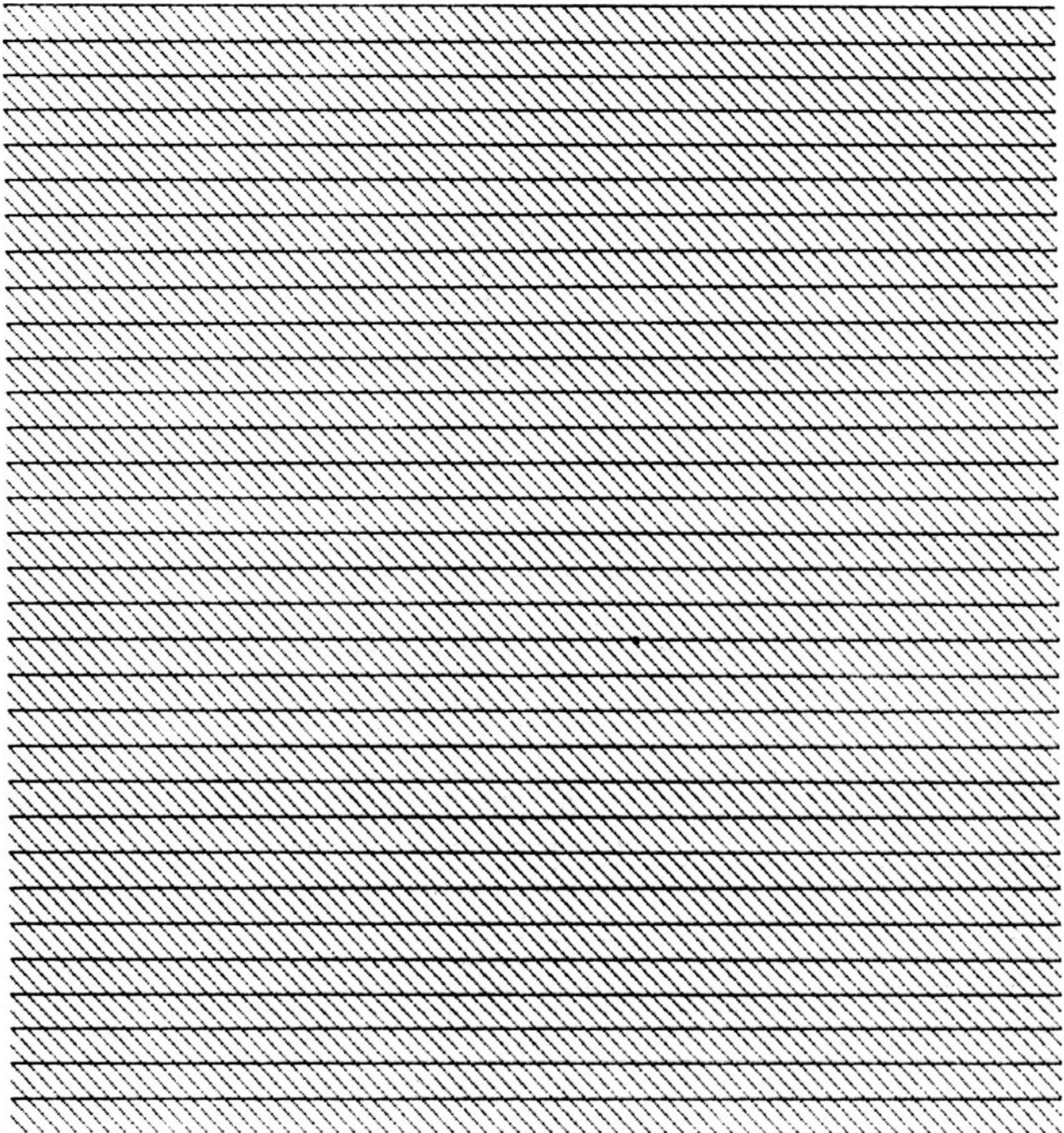
```

TCS
präsentiert den



Genie II S
und





Die HRG des Genie III s

Bis ich mich endlich entschlossen hatte, bei meinem alten Genie I die HRG 1b einzubauen, vergingen seit dem Kauf des Computers über zwei Jahre. "Spielkram!" war mein Argument dagegen. Aber die zusätzlichen 12 kB Speicher waren schließlich doch zu verlockend. Beim G3s ist von Hause aus eine Baugruppe für hochauflösende Graphik drin. Sie umfaßt sogar zweimal 32, also zusammen 64 kB. So kann sich der geneigte Leser lebhaft meinen Ärger vorstellen, als in der Dokumentation (wie bei Trommeschläger üblich nicht ausreichend) keine brauchbare Erklärung darüber zu finden war, wie man diesen Speicher anspricht.

Es wird ein sehr interessantes Graphik-BASIC auf der G-DOS-Diskette mitgeliefert. Aber es braucht den Speicher für die absonderlichsten Luxusroutinen auch dann, wenn man bloß eben mal irgendwo einen Punkt hinsetzen möchte. Und letztenendes ist es natürlich nur unter BASIC nutzbar; für den Assembler-Crack ein gewichtiger Grund dagegen. Also war der Mann mal wieder selbst:

Aus der technischen Beschreibung geht (zumindest nach mehrmaligem Lesen und Probieren) hervor, daß der Graphikspeicher von 8000-FFFF geht. Er kann anstelle des Hauptspeichers durch einen Output auf den Port FA mit gesetztem Bit 3 der CPU zugänglich gemacht werden. Da dieser Port noch weitere Funktionen des Systems steuert, muß er zunächst gelesen und an diesem Input nur bitweise gefummelt werden, damit der bisherige Betriebszustand nicht in fataler Weise verändert wird. Jetzt kann in 8000-FFFF beliebig geschrieben und gelesen werden. Dieser Adreßraum ist jetzt nach dem Umschalten über Port FA der Graphikspeicher.

Der logische Aufbau der HRG des G3s ist der HRG 1b ganz ähnlich. Auch hier entspricht ein bestimmtes Byte des Speichers der ganzen Breite einer Punktzeile einer normalen Anzeigestelle des Bildschirms. Deshalb sind auch hier die LSB der Graphik- und Bildschirmadressen gleich. Die Reihenfolge der Bits (Graphikpunkte) ist ebenfalls dieselbe: Bit 0 wird ganz links angezeigt, Bit 7 rechts (die HRG 1b zeigt nur bis Bit 5 an). Wegen der großen Ähnlichkeit mit der HRG 1b, die mir sehr vertraut ist, erschien deshalb die Aufgabe lösbar, der HRG des G3s auf die Schliche zu kommen.

Aber da ist eine Merkwürdigkeit, die vorab geklärt werden mußte: Jeder Anzeigemodus (25 X 80, 16 X 64, 24 X 64 und 32 X 64 Zeichen) benutzt einen unterschiedlich großen Bildschirmspeicher. Außerdem werden unterschiedlich tiefe Punktspalten pro Bildschirmbyte angezeigt. Nur im Modus 32 X 64 kommen 32 X 64 X 16 X 8, also 512 X 512 = 262144 Punkte, das sind 32 kB, zur Anzeige. Was geschieht in den anderen Modi mit dem übrigbleibenden Graphikspeicher? Ein Testprogramm, das im Anschluß an diesen Artikel aufgelistet ist, brachte Aufschluß.

Das Programm hat seinen Zweck erfüllt. Es mündete in die beiden Tabellen hinter dem Listing. Da es deshalb niemand abzutippen braucht, seien hier nur diejenigen Teile erläutert, die die Graphik ansprechen, denn diese Teile kann der Leser in seine eigenen Programme einbauen:

Im DOS liegt der Stack ziemlich tief unten. Um aber das Programm notfalls auch fahren zu können, wenn der Stack im Himem liegt, war es notwendig, den Stackpointer zu retten und neu zu laden. Nach dem Umschalten würde er zwar normal arbeiten, wäre aber in der Graphik sichtbar.

Der nächste Schritt (Unterprogramm change) sollte an wichtigen Stellen des Bildschirm-RAMs Marken setzen, so daß die jeweilige Video-Adresse ablesbar ist. Es sind die Anfangs- und Endadressen des Bildschirms bei jedem der vier möglichen Bildschirmformate. Da im unteren Adreßraum der großen Formate gleichzeitig die Tastatur memory mapped ist,

mußte sie zunächst über das Bit 4 des Ports FA weggebankt und anschließend wieder enabled werden. Der einleitende CALL nach 01C9h löscht den Bildschirm und aktiviert gleichzeitig Formatänderungen (s. u.).

Das Löschen des Graphikspeichers ist einfacher als mit der HRG 1b: Da die CPU direkten Zugriff hat, braucht er nur mit LDIR ausgefüllt zu werden. Die anschließend folgende Abfrage der P-Tasten, mit denen das Programm zwischen den Bildschirmformaten hin- und herspringen kann, soll hier nicht weiter erklärt werden. Wie aber die Formate gewechselt werden, ist wieder wichtig für den Graphik-Programmierer:

Ab 3760h stehen vier Tabellen zu je 16 Bytes, die die Bildschirmparameter enthalten. Die Parameter für das gerade benutzte Format stehen ab 37F0h. Deshalb werden beim Umschalten des Formats die neuen Parameter, wieder mit LDIR, dorthin geladen. Das alleine genügt allerdings nicht. Aktiv werden sie erst, wenn gewisse Control-Codes ausgegeben werden. Das ist z. B. auch beim Löschen des Bildschirms der Fall, deshalb wird nach dem Formatwechsel erneut change aufgerufen.

Das eigentliche Arbeitsprogramm steht ab loop. Es sei nur so viel dazu gesagt, daß ein Graphik-Cursor mit den Pfeiltasten über den Bildschirm gejagt werden kann. Mit der Blank-Taste kann er verlangsamt werden, um einzelne Bildschirmstellen gezielt ansteuern zu können. Die jeweilige Adresse des Cursors im Graphikspeicher wird ständig angezeigt. Wenn dieser Cursor nun eine der im UP change gesetzten Marken durchläuft, kann man sich die Bildschirmadresse der Marke und die Graphikadresse des Cursors notieren und erfährt so den Zusammenhang zwischen beiden.

Für den Computergraphiker wird es nun ab der Stelle exit wieder interessant: Der alte Stackpointer wird wieder aus dem Puffer geladen. Die Bits 1 und 3 des Systembytes 1 (Port FA) werden wieder auf 0 gesetzt, wodurch die Graphik vom Bildschirm verschwindet und der Hauptspeicher anstelle des Graphikspeichers in die obere Adreßhälfte eingeblendet wird. Die Interrupts waren die ganze Zeit über ausgeschaltet, denn bei zeitweise ausgebankter Tastatur und manipuliertem Stack ist Vorsicht angebracht. Mit EI werden sie wieder enabled. Mit RET geht es schließlich zurück ins Betriebssystem.

Die beiden Tabellen sind folgendermaßen zu interpretieren: Wie auch bei der HRG 1b, so kann man sich für die jeweils oberste Punktzeile eines Zeichens den ganzen Bildschirm als eine ununterbrochene Perlenkette vorstellen. Wenn die oberste Zeile durchlaufen ist (also rechts unten im Bildschirm!), dann dann geht es in der zweiten Punktzeile (oben links!) weiter. Beim Anzeigeformat 32 X 64 geschieht das lückenlos.

Die anderen Formate zeigen aber Unterbrechungen beim Zeilenwechsel. Besonders die Graphik auf der letzten Seite dieses Artikels dürfte das anschaulich machen. Aus beiden Tabellen geht auch hervor, daß pro Bildschirmstelle bei den kleineren Formaten auch nicht alle Punktzeilen zur Anzeige kommen. Die Punktzeilen sind von 0 - F durchnummeriert.

Zur praktischen Anwendung: Die Tabellen zeigen, welche Bereiche des Graphikspeichers überhaupt im Bildschirm sichtbar werden können, je nach Format. Die Abszisse eines Graphikbytes errechnet sich nun ebenso wie die eines Buchstabens im normalen Bildschirm. Um aus diesem Beitrag kein Buch zu machen, verweise ich hierzu auf die Vielzahl von Artikeln zur HRG 1b, die an dieser Stelle erschienen sind.

Die Ordinate, also die Lage eines Bytes in senkrechter Richtung, ist vom darüber-/darunterliegenden Byte immer genau 2 kB im Graphikspeicher entfernt, solange es sich um dieselbe Videozeile handelt. Beim Sprung in die nächste Videozeile (nicht zu Verwechseln mit den Punktzeilen eines Zeichens!) ist einfach zu einer Adresse die Zeilenlänge hinzuzuaddieren, also 80 oder 64.

Jedes Graphikbyte hat selbstverständlich ebenfalls 8 Bits. Jedes Bit repräsentiert einen Punkt in der Waagerechten. Wie auch bei der HRG 1b, so kann man auch beim G3s durch Setzen, Löschen oder Prüfen einzelner Bits die BASIC-Entsprechungen SET, RESET und POINT durchführen. Auch zu diesem Punkt mag es genügen, auf die bisher erschienen Artikel zur HRG 1b zu verweisen. Dabei muß lediglich bedacht werden, daß alle acht Bits beim G3s zur Anzeige kommen.

Arnulf Sopp

5300		00001	ORG	5300h	;darunter Platz für Stack
5300		00002	stack EQU	\$;hier fängt er an
0002		00003	spbuf DS	2	;Puffer für Stackpointer
5302	F3	00004	start DI		;nichts riskieren
5303	ED730053	00005	LD	(spbuf),SP	;Stackpointer retten
5307	310053	00006	LD	SP,stack	;eigener Stack f. Progr.
530A	CD8553	00007	CALL	change	;Bildschirm markieren
530D	210080	00008	LD	HL,8000h	;Beginn des Graphiksp.
5310	110180	00009	LD	DE,8001h	;eine Stelle weiter
5313	01FF7F	00010	LD	BC,7fffh	;Zähler zum Löschen
5316	75	00011	LD	(HL),L	;1. Graphikstelle löschen
5317	EDB0	00012	LDIR		;alle löschen
5319	11023C	00013	LD	DE,3c02h	;Bildschirmbereich
531C	3A803B	00014	loop LD	A,(3880h)	;P-Tasten abfragen
531F	E6F0	00015	AND	0f0h	;eine gedrückt?
5321	281A	00016	JR	Z,ctrlkey	;falls nein
5323	E5	00017	PUSH	HL	;Graphikzeiger retten
5324	D5	00018	PUSH	DE	;dto. Bildschirmzeiger
5325	210637	00019	LD	HL,3706h	; (Adr. Video-Par.-tab.)
5328	11F037	00020	LD	DE,37f0h	;dto. aktive Parameter
532B	0E20	00021	LD	C,20h	;16 Parameter
532D	CDA353	00022	CALL	pkey	;Parameter-LSB ermitteln
5330	7D	00023	LD	A,L	;Parameter-LSB
5331	07	00024	RLCA		;um 4 Bits rot. (*16)
5332	07	00025	RLCA		
5333	07	00026	RLCA		
5334	07	00027	RLCA		
5335	6F	00028	LD	L,A	;jetzt kompl. Adr. in HL
5336	EDB0	00029	LDIR		;neue Parameter setzen
5338	D1	00030	POP	DE	;Register restaurieren
5339	E1	00031	POP	HL	
533A	CD8553	00032	CALL	change	;Änderungen aktivieren
533D	3A403B	00033	ctrlkey LD	A,(3840h)	;Tastatur auslesen
5340	E6FC	00034	AND	0fch	;Pfeile, Blank oder BRK?
5342	28DB	00035	JR	Z,loop	;falls nichts dergleichen
5344	07	00036	RLCA		;Blank?
5345	063F	00037	LD	B,03fh	;für Verzögerung
5347	3802	00038	JR	C,goloop	;falls Blank
5349	0607	00039	LD	B,07h	;sonst weniger Verzög.
534B	0EFF	00040	goloop LD	C,0fffh	;Zähler BC komplettieren
534D	D5	00041	PUSH	DE	;retten
534E	110004	00042	LD	DE,0400h	;in d. Gr.-splt. darunter
5351	3600	00043	LD	(HL),00h	;alten Strich löschen
5353	07	00044	RLCA		;Rechtspfeil?
5354	DCAD53	00045	CALL	C,right	;falls ja
5357	07	00046	RLCA		;Linkspfeil?
5358	DCAF53	00047	CALL	C,left	;falls ja
535B	07	00048	RLCA		;Abwärtspfeil
535C	DCB153	00049	CALL	C,down	;falls ja
535F	07	00050	RLCA		;Aufwärtspfeil?
5360	DCB353	00051	CALL	C,up	;falls ja

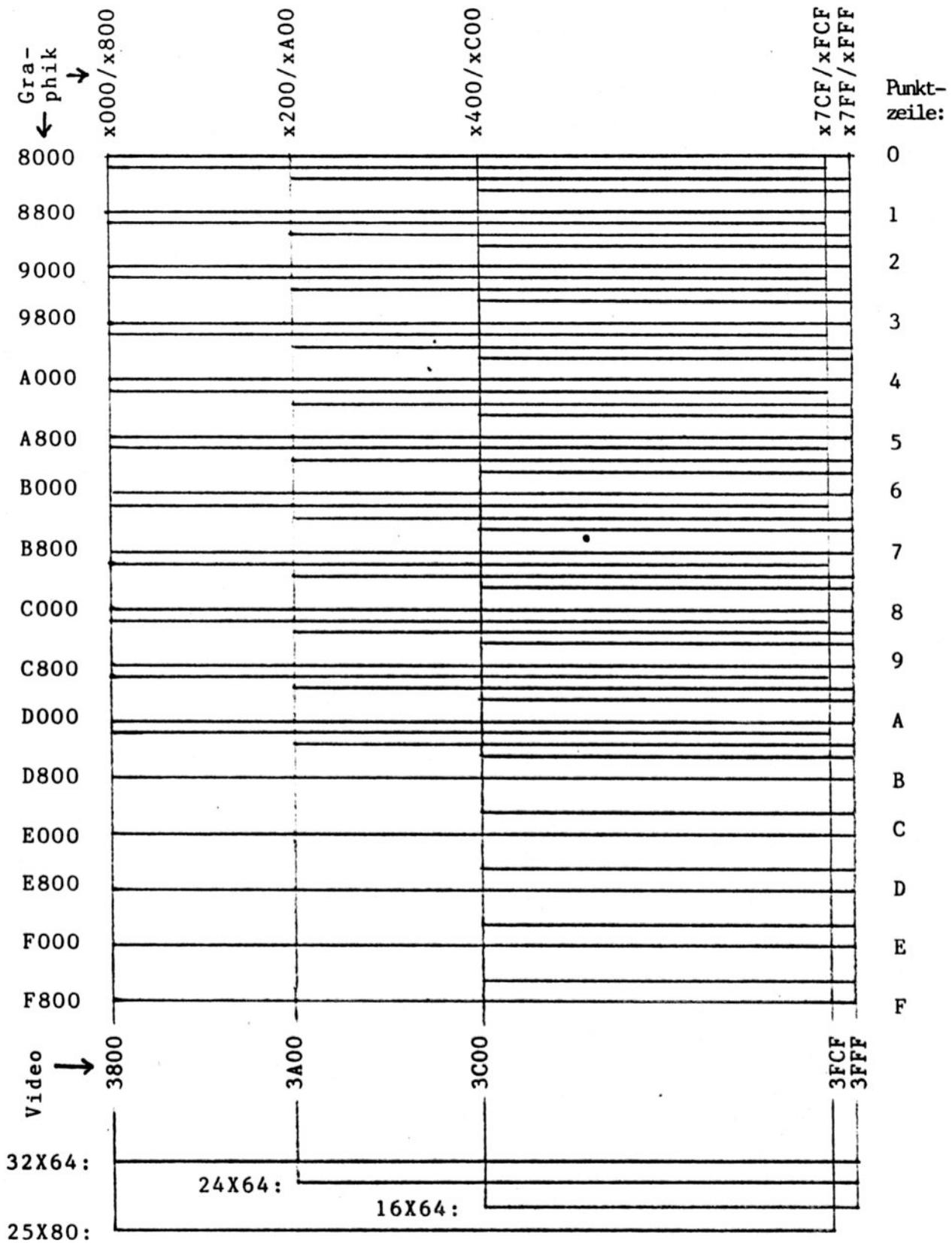
5363	D1	00052	POP	DE	;restaurieren	
5364	07	00053	RLCA		;BREAK	
5365	3812	00054	JR	C,exit	;falls ja	
5367	CBFC	00055	SET	7,H	;nur 8000 - FFFF zulassen	
5369	36FF	00056	LD	(HL),Offh	;dort neuen Strich laden	
536B	EB	00057	EX	DE,HL	;austauschen	
536C	E5	00058	PUSH	HL	;wird verändert	
536D	CD6340	00059	CALL	4063h	;Stelle in Hex anzeigen	
5370	E1	00060	POP	HL	;restaurieren	
5371	EB	00061	EX	DE,HL	;zurücktauschen	
5372	0B	00062	delay	DEC	BC	;Verzögerungszähler dekr.
5373	78	00063	LD	A,B	;nachsehen, ob	
5374	B1	00064	OR	C	;Zähler abgelaufen	
5375	20FB	00065	JR	NZ,delay	;falls noch nicht	
5377	18A3	00066	JR	loop	;o. k., von vorne	
5379	ED7B0053	00067	exit	LD	SP,(spbuf)	;Stackpointer restaur.
537D	DBFA	00068	IN	A,(Ofah)	;Systembyte 1 lesen	
537F	E6F5	00069	AND	0f5h	;Bits 7 und 5 wieder aus	
5381	D3FA	00070	OUT	(Ofah),A	;Graphik ausblenden	
5383	FB	00071	EI		;keine Gefahr mehr	
5384	C9	00072	RET		;ins Betriebssystem	
5385	CDC901	00073	change	CALL	01c9h	;Änderungen aktivieren
5388	DBFA	00074	IN	A,(Ofah)	;Systembyte 1 lesen	
538A	F61A	00075	OR	1ah	;Bits 1, 3 und 4 setzen	
538C	D3FA	00076	OUT	(Ofah),A	;Graphik fitmachen	
538E	3EBF	00077	LD	A,0bfh	;Grobgraphikblock	
5390	32003B	00078	LD	(3800h),A	;Anfang Großbildschirm	
5393	32003C	00079	LD	(3c00h),A	;Anfang Kleinbildschirm	
5396	32CF3F	00080	LD	(3fcfh),A	;Ende 80-Zeichen-Bildsch.	
5399	32FF3F	00081	LD	(3fffh),A	;Ende sonstiger Bildsch.	
539C	DBFA	00082	IN	A,(Ofah)	;Port wieder lesen	
539E	CBA7	00083	RES	4,A	;Tastatur wieder aktiv.	
53A0	D3FA	00084	OUT	(Ofah),A	;mit Systembyte 1	
53A2	C9	00085	RET		;oben weiter	
53A3	07	00086	pkey	RLCA		;P1-Taste? (80 X 25 Z.)
53A4	DB	00087	RET	C	;falls ja	
53A5	2C	00088	INC	L	;Adr.-LSB d. Tab. erhöhen	
53A6	07	00089	RLCA		;P2-Taste? (64 X 16 Z.)	
53A7	DB	00090	RET	C	;usw.	
53A8	2C	00091	INC	L		
53A9	07	00092	RLCA		;P3-Taste? (64 X 24 Z.)	
53AA	DB	00093	RET	C		
53AB	2C	00094	INC	L		
53AC	C9	00095	RET		;P4: 64 X 32 Z.	
53AD	23	00096	right	INC	HL	;nächste Graphikstelle
53AE	C9	00097	RET		;oben weiter	
53AF	2B	00098	left	DEC	HL	;vorige Graphikstelle
53B0	C9	00099	RET			
53B1	19	00100	down	ADD	HL,DE	;tiefer in selber Spalte
53B2	C9	00101	RET			
53B3	B7	00102	up	OR	A	;Cy löschen
53B4	ED52	00103	SBC	HL,DE		;höher in selber Spalte
53B6	C9	00104	RET		;erledigt	
5302		00105	END	start		;dort Einsprung

00000 Fehler

change	5385	ctrlkey	533D	delay	5372	down	53B1	exit	5379
goloop	534B	left	53AF	loop	531C	pkey	53A3	right	53AD
spbuf	5300	stack	5300	start	5302	up	53B3		

Tabelle der Graphik- und Bildschirmadressen des Genie III s:

Bildschirm- darstellung:	Bildsch.- adressen:	Zeile:	Graphik- adressen:
25 X 80 Zeichen:	3800 - 3FCF	0	8000 - 87CF
		1	8800 - 8FCF
Beispiel für einige Zeilen:		2	9000 - 97CF
3800 - 384F <-	8000 - 804F	3	9800 - 9FCF
3850 - 389F <-	8050 - 809F	4	A000 - A7CF
.		5	AB00 - AF CF
usw.		6	B000 - B7CF
.		7	BB00 - BF CF
3F30 - 3F7F <-	D730 - D77F	8	C000 - C7CF
3F80 - 3FCF <-	D780 - D7CF	9	CB00 - CF CF
		A	D000 - D7CF
16 X 64 Zeichen:	3C00 - 3FFF	0	8400 - 87FF
		1	8C00 - 8FFF
Beispiel für einige Zeilen:		2	9400 - 97FF
3C00 - 3C3F <-	8400 - 843F	3	9C00 - 9FFF
3C40 - 3C7F <-	8440 - 847F	4	A400 - A7FF
.		5	AC00 - AFFF
usw.		6	B400 - B7FF
.		7	BC00 - BFFF
3F80 - 3FBF <-	F780 - F7BF	8	C400 - C7FF
3FC0 - 3FFF <-	F7C0 - F7FF	9	CC00 - CFFF
		A	D400 - D7FF
		B	DC00 - DFFF
		C	E400 - E7FF
		D	EC00 - EFFF
		E	F400 - F7FF
24 X 64 Zeichen:	3A00 - 3FFF	0	8200 - 87FF
		1	8A00 - 8FFF
Beispiel für einige Zeilen:		2	9200 - 97FF
3A00 - 3A3F <-	8200 - 823F	3	9A00 - 9FFF
3A40 - 3A7F <-	8240 - 827F	4	A200 - A7FF
.		5	AA00 - AFFF
usw.		6	B200 - B7FF
.		7	BA00 - BFFF
3F80 - 3FBF <-	CF80 - CF BF	8	C200 - C7FF
3FC0 - 3FFF <-	CF C0 - CFFF	9	CA00 - CFFF
		A	D200 - D7FF
32 X 64 Zeichen:	3800 - 3FFF	0	8000 - 87FF
		1	8800 - 8FFF
Beispiel für einige Zeilen:		2	9000 - 97FF
3800 - 383F <-	8000 - 803F	3	9800 - 9FFF
3840 - 387F <-	8040 - 807F	4	A000 - A7FF
.		5	AB00 - AFFF
usw.		6	B000 - B7FF
.		7	BB00 - BFFF
3F80 - 3FBF <-	FF80 - FF BF	8	C000 - C7FF
3FC0 - 3FFF <-	FF C0 - FFFF	9	CB00 - CFFF
		A	D000 - D7FF
		B	DB00 - DFFF
		C	E000 - E7FF
		D	EB00 - EFFF
		E	F000 - F7FF
		F	FB00 - FFFF



Korrespondierende Graphik- und Bildschirmadressen des Genie II bei unterschiedlichen Bildschirmformaten

**Was haltet Ihr von Textsystemen?*

Im Laufe der letzten 5 Jahre habe ich mich mit 6 (sechs!) verschiedenen Textsystem für den TRS 80 abgegeben und glaube, ein vergleichendes Urteil abgeben zu dürfen. Ich könnte mir denken, daß dies den einen oder anderen interessiert - vor allem jene, die vor der Entscheidung stehen, sich eins anzuschaffen.

Tue er es, nachdem er dies hier gelesen; aber bitte: ohne meine Gewähr! Er erschieße mich, jedoch ohne Gewehr!

In der Tat ist die Beurteilung der div. Textsysteme offensichtlich reine Geschmackssache - und natürlich auch Übungssache. Wenn ich die mir bekannten (und ausprobierten) Systeme:

- 1) GENIETEXT (3.0)
- 2) SCRIPSIT bzw. SUPER-SCRIPSIT
- 3) TSCRIPS
- 4) NEWSSCRIPT
- 5) LAZYWRITER
- 6) DOTWRITER
- 7) WORDSTAR f. NEWDOS80 ???

vergleichend beurteilen sollte, sähe das so aus:

Nr.	Vorteile	Nachteile
1)	Bester Bedienungskomfort: Die Kombinationen der außen liegenden Tasten gestatten "blindes" Schreiben und Formatieren; kinderleichtes Umbrechen von Zeilen sowie ganzer Textblöcke; Umlaute u. ß <u>direkt</u> abrufbar; Pixel-Grafik sowie eigene Zeichensätze möglich; sortiertes Inhaltsverz.; klares Deutsch; bequemer Probeausdruck	unzuverlässiges "standing" (steigt unversehens aus); keine Anpassung der Zeilenlänge bei Drucker-Steuerzeichen innerhalb einer Zeile
2)	an TRS80 angepaßt (Tandy)	veraltet; undiskutabel
3)	sehr flexibel, hervorragend an EPSON angepaßt, bequeme Einbindung von Schrift- und eigenen Zeichensätzen; Pixelgrafik mögl. (auch HRG?)	nicht so ergonomisch wie GENIETEXT; komplizierte Textformatierung; Umlaute und ß nur mit Vortaste erreichbar
4)	ungewöhnlich umfangreicher und rel.leicht zu merkender Befehlsschatz, daher äußerst flexibel und anpassungsfähig	Erfordert ständig Steuerzeichen zur Formatierung (verwandt mit DOTWRITER); echter Probeausdruck nicht möglich (nur mitsamt Steuerz)
5)	Für eine Hervorhebung positiver Eigenschaften ist meine Erfahrung zu gering. Es soll angeblich "lazy", d.h. lässig! also "easy" zu handhaben sein!	Wegen der rel. undurchsichtigen Befehlsstrukturen (?) hab ich es mehrmals wieder aufgeben, hierin einzusteigen. (Wahrscheinlich ist meine rechte Gehirnhälfte schuld!)
6)	Geniale Verwendung zahlreicher lieferbarer oder selbst zu entwerfender Zeichensätze in allen Größen; gegenüber NEWSSCRIPT und TSCRIPS reduzierter Formatierungsaufwand Auch Pixel-u.HR-Grafik mögl.	Außerst langsam! Sehr komplizierte Handhabung b.Mischung v.Zeichensätzen; Fehlermeldg. nicht aufschlußreich genug; Bearbeitung daher sehr zeitaufwendig; (und papier=) aufwendig; Probeausdruck umständlich.

5
 0
 1
 0
 0
 1
 0

So kommt es, daß jeder dasjenige System als das "beste" bezeichnet, in das er am besten eingearbeitet ist. Schwierig wird es erst dann, wenn man - wie ich - mehrere Systeme erprobt hat und sich nun von Fall zu Fall entscheiden muß, welches man jeweils verwenden will...!

Diesen Text hier z.B. habe ich spaßeshalber seit ca. 2 Jahren mal wieder mit GENIETEXT geschrieben - und ich erfreute mich gerade erneut an seiner patenten, sehr handlichen Bedienung --- und ärgerte mich prompt erneut über das saumäßige Standvermögen im Speicher: dreimal passierte es, daß der Text von selbst vom Bildschirm verschwand oder der Drucker streikte und die Diskette plötzlich wieder zu "booten" begann, mit dem Effekt, daß ich einen großen Teil nochmal eingeben mußte!! Mit einem so unzuverlässigen System kann Herr Zender, der sein Textsystem für ernsthafte Anwendungen in Büros propagiert, keinen Blumentopf gewinnen. Nach diesem "Erfolg" müßte sein "Unternehmen" inzwischen eigentlich pleite sein! Besonders groß ist mein Ärger über ihn gerade deshalb, weil sein Textsystem sich so fantastisch anließ und besser als alles andere zu sein schien.-

7x (!)

Nun schwanke ich ständig zwischen TSCRIPS (V.5.2) und NEWSCRIPT, die ich beide teuer erwarb (denn anständige Clubkollegen kopieren nichts, und mit unanständigen verkehre ich nie - oder selten...), und möchte keins von beiden missen: Das eine seiner bequem zu handhabenden (handzuhabenden?) Zeichensätze wegen, das andere seiner enormen Vielseitigkeit und seines logischen Befehlsaufbaus halber...

Auch mit LAZYWRITER habe ich es in diesen Tagen wieder einmal versucht, aufgestachelt durch Ralf's Lobeshymne (s.letztes INFO). Pustekuchen: ich hatte plötzlich 12 statt 10 Finger, und trotzdem fehlten mir immer wieder zwei! Wenn Du denkst, der Aufwärtspfeil läßt den Cursor (dtsh.: Läufer) nach oben laufen, so denkst du nur, Du denkst, gedacht zu haben... plötzlich bist Du im Editiermodus! - Das Unterstreichen (mit GENIETEXT ein Kinderspiel) ist hier eine reine Volksbelustigung; Du denkst, Du sitzt mit verbundenen Augen im Autoscooter - nur ist das nicht lustig für den eiligen Schreiber; usw. usf. usw. Wenn man das mit "gewöhnungsbedürftig" entschuldigen soll, muß ich sagen, daß das ein Beweis für Mangel an Komfort ist.

So bleibt's einstweilen beim Pendeln. Mit dem Effekt, daß ich mich von Zeit zu Zeit verbiestere, weil mir die Befehle des einen und anderen Editors durcheinander geraten; dann aber: "Von Zeit zu Zeit seh' ich den Alten gern..." dachte sich schon Mephisto, Club-Mitglied Nr.00h, und hütete sich, mit IHM zu brechen; so halte auch ich es (obschon ich außer dem "M" nur wenig mit ihm teile) und sehe den Alten - verzeiht: den alten Editor immer wieder gern und hüte mich, die Scheibe des GENIES zu brechen, obgleich sie es verdient hätte...

Wer Weiß Wieder Was Wertvolleres?

Wer nennt mir ein System, das alle Vorzüge (siehe oben) vereint und keinen Nachzug, sprich: Nachteil (siehe oben) aufzwingt? Vielleicht eins, das einen Aufzug nachzwingt, wenn man vor Erschöpfung im (Wein-)Keller gelandet ist und wieder hoch will? **Natürlich niemand!**

"Du bist zu kritisch!" - spricht Mephisto.
"Jawohl, Herr" - sagt KaJott - "das ist so!!"



Assembler -
Routiniers
Nutzen
Und
Lernbegierige
Freaks

Suchen
Oft
Prima
Programme -

oder, wenn Euch Chinesisch (von oben nach unten) vorkommt wie
Schisenich (von rechts nach links) und zu muhsam ist, dann eben
einfach:

"Assembler-Routiniers nutzen -
Und Lernbegierige Freaks
Suchen Oft: Prima Programme!"

Damit Ihr nun nie mehr soviel suchen musst: Hier ein erschöpfendes
Verzeichnis der Produkte aus der Werkstatt unseres Chef-Denkens. Ihm
seien diese Seiten gewidmet. Ein bisschen Dank hat er schon verdient,
denke ich...

Ihm wuensche ich - sicher im Namen aller Z80-Fans - noch unzählige
* Stern * -Stunden am "Heckbrett"
im neuen Lebensjahr!

PS: WER DAS DATENBAUKYSTER "SUPER" BESITZT, KÖNN DIE DATEI [FILENAME "SOPPLI"] UEBERSPIELT BEKOMMEN UND SIE DANN SELBST LFD. ERGÄNZEN.

Thema: "ASSEMBLER / DISASSEMBLER"

- 1 Z80-Assembler fuer Autodidakten
Thema: Einige Grundbegriffe (Bsp.-Programmen).- Leider fand sich fuer Arnulf kein Mitstreiter. So blieb es bei diesem Anlauf!
Heft 10/84 Seite 26

- 2 Vom Umgang mit "Fremdsprachen"
Thema: Einfuehrung in den Umgang mit dem EDITOR/ASSEMBLER von TANDY (EDTASM)
Heft 9/84 Seite 11

- 3 Dein A S S E M B L E R, das unbekannte Wesen
Thema: Verbiegen eines DCB-Vektors
(z.B. Druckerausgabe verhindern).
Heft 3/85 Seite 15

- 4 Eine eigenartige Adresse: \$
Thema: Das Symbol, das die Adresse des eigenen Opcodes meint.
Heft 7/85 Seite 20

- 5 Der Maschinenbefehl SLL s
Thema: Unterschied zwischen Links- und Rechts-SHIFT: eine Warnung!
Heft 7/85 Seite 6

- 6 Springen - aber wie?
Thema: Der Unterschied zwischen JP und JR.
Heft 6/85 Seite 28

- 7 DSMBLR/CMD - ein wenig komfortabler
Thema: Dieser DISASSEMBLER erstellt Source-Codes, die EDTASM weiter verarbeiten kann.
Heft 10/84 Seite 10

- 8 Hex - wozu ?
Thema: Sinn und Struktur sedezimaler Darstellung
Heft 3/85 Seite 15

- 9 Nochmals: HEX-Darstellung.
Thema: Ausgabe einer Variablen in sedezimaler Darstellung.
Heft 3/84 Seite 19

- 10 BASIC selbst erweitern
Thema: DISK-BASIC-Befehle fuer eigene Zwecke verbiegen.
(Mit einer Tabelle der BASIC-Vektoren.)
Heft 9/85 Seite 17

- 11 PUT TO adresse - ein neuer BASIC-Befehl
Thema: Implementierung eines neuen BASIC-Befehls. BASIC wird verschoben ohne Zerstoerung residenter Programme.
Heft 4/84 Seite 10

- 12 Kaffeekochen ab sofort gestattet
Thema: Implementierung eines akustischen Signals, das bei Auftritt eines Fehlers ertoeut.
Heft 10/84 Seite 28

Thema: BANKING

- 13 BANK SELECTION mit dem GENIE I
Thema: Beschreibung des EG64 MBA (Trommeschlaeger) sowie ein
Assembler-Listing: "BASIC mitten im Interpreter" dazu.
Heft 5/84 Seite 10
- 14 Memory Banking - umsonst !
Thema: Ueber die Moeglichkeiten, fuer das GENIE 64 KB zu
erschliessen (lt. c't 8/84). - Mit Spooler.
Heft 11/84 Seite 5
- 15 Memory Banking - umsonst ! (Nachtrag)
Thema: (siehe den Hauptartikel zu diesem Thema)
Heft 11/84 Seite 12
- 16 Nochmals: MEMORY BANKING
Thema: Vergleich des EG64 von Schmidtke und von Trommeschlaeger.
Heft 5/84 Seite 26
- 17 Ein Spooler fuer den EG 64 MBA
Thema: 3-Tastenbefehl <567> ruft Druckerspooles auf. Doppel-Job
(Befehlseingabe/Drucken gleichzeitig). Besser als HRG-Spooler
Heft 9/84 Seite 20
- 18 Noch'n Zap !
Thema: EG64 MBA-Anwendung ohne AUTO-Verschleiss
(mit Nachtrag in 8/84, S.5)
Heft 6/84 Seite 23
- 19 Banking und kein Ende!
Thema: Initialisierung des MBA gleich beim Booten.
Heft 10/84 Seite 21
- 20 BEL ohne Nachladen gleich bei BOOT
Thema: - und den ASCII-Code 07 nutzbar machen...
Heft 11/84 Seite 21

Thema: BILDSCHIRM

- 21 VIDHEX - Hexanzeige ds. Bildschirms mit der HRG
Thema: Verwandlung von Sonderzeichen in Hex-Zahlen mittels der HRG
Heft 8/85 Seite 6
- 22 Unbenutzte Sonderzeichen
Thema: Was tut sich beim Einschalten auf dem Bildschirm ?
Bemerkungen und Fragen hierzu.
Heft 5/84 Seite 21
- 23 Und noch einmal: DIE SONDERZEICHEN DES GENIE I
Thema: Das Programm zeigt die Sonderzeichen ASCII 0-31 mit PRIN1
auf dem Bildschirm an.
Heft 3/85 Seite 24
- 24 Ein Treiber fuer die Sonderzeichen
Thema: Siehe auch den Beitrag aus Heft 3/85, Seite 24.
Heft 4/85 Seite 3

Beitraege von A.SOPP im INFO des Bremerhavener GENIE/TRS80-Clubs

25 RESET - ein Maedchen fuer alles
Thema: Bildschirmausdruck mit der RESET-Taste

Heft 4/85 Seite 28

Thema: BILDUNG

26 Deutsch mit und trotz Computer
Thema: Nachhilfe nicht nur in Deutsch sondern auch in der
 Aussprache und Deklination von Fremdwoertern u.a.

Heft 5/85 Seite 17

Thema: DOS

27 H-DOS 2.3
Thema: G-DOS-2.x-Modifikation
 Kurzanleitung.

Heft 11/84 Seite 35

28 Praedikant "wertvoll": Grosser, "Das DOS(-)Buch
Thema: Buchbesprechung.

Heft 11/85 Seite 11

29 Neuer DOS-Befehl: OUT port#,xx,yy,...
Thema: Genaue Beschreibung der Vorgehensweise bei dessen
 Implementierung in SYS 29.

Heft 11/84 Seite 8

30 Die Library-Befehle des G-DOS 2.x
Thema: Nach dem Beitrag in 9/84 nun eine lueckenlose Tabelle der
 LIB-Befehle und der SYS-Files, bei denen eingesprungen wird.

Heft 11/84 Seite 13

Thema: DRUCKER

31 Die Systemoptik ausgetrickst!
Thema: Formatierung - z.B. von Listings - durch nachtraeglichen
 Einbau von Steuerzeichen in den Code.

Heft 2/84 Seite 20

32 HEX-Output auf den Drucker
Thema: Jetzt kann auch der GEMINI die ASCII-Codes einer Zeichen=
 vorgabe sedezimal ausdrucken!

Heft 9/85 Seite 5

33 JKL - aber druckerschonend!
Thema: Statt Punkte werden Leerzeichen als Ersatz fuer nicht-druck=
 bare Zeichen "gedruckt". (2 HEX-Dumps versehentl. auf S.24!)

Heft 4/86 Seite 22

34 Und es geht doch: LPRINT CHR\$(10)
Thema: Verbesserter Druckertreiber, der auch CHR\$(0...etc.) erlaubt

Heft 11/84 Seite 20

35 LPRINTCHR\$(irgendwas)
Thema: Weiteres zum Thema "CHR\$(0, 10, 11, 12) ausdrucken"
 (siehe auch 11/84, Seite 20)

Heft 12/84 Seite 3

36 Ein Hoch auf die Mitdenker!
Thema: Listings, Sector-Dumps, Hardcopies u.a. mitten im Text
 ausdrucken.

Heft 6/86 Seite 20

Thema: HARDWARE

37 Hard entwanzt: Die HRG 1b
Thema: Entflechtung des unteren I/O-Adressraums der Ports 0-127.
Heft 12/85 Seite 12

38 HARDWARE-Sonderheft
Thema: Die Software zum "Versechzehnfacher" von Helmut Bernhardt.
(Sonderheft 1985)
Heft Seite 0

39 Tune-up des GENIE mit "Speed-Up"
Thema: Einbau- und Erfahrungsbericht.

Heft 9/84 Seite 25

Thema: GRAFIK

40 Bildschirm-Hardcopy bei Grafik
Thema: Pixel-Wiedergabe mit 3x7 Matrix-Punkten und 7/72" Zeilen-
abstand.
Heft 6/84 Seite 9

41 Die HRG 1b und BASIC netto
Thema: Ein BASIC- (!) -Programm als HRG-Treiber fuer die drei
wichtigsten Befehle SET/RESET/POINT.
Heft 5/85 Seite 2

42 Minimaltreiber fuer die HRG 1b
Thema: Ein HRG-Treiber in Maschinensprache - nur 130 Bytes lang!
(Fuer SET / RESET / POINT.)
Heft 6/85 Seite 9

43 HRG - aber fix!
Thema: Ein Assembler-Listing, mit dem die Kombination von HRG mit
ASCII u.Pixel-Grafik gespeichert und geladen werden kann.
Heft 2/85 Seite 23

44 Den HRG-Speicher loeschen
Thema: Alltagspraxis zu den bisherigen theoretischen Ausfuehrungen
ueber die Programmierung der HRG 1b.
Heft 1/85 Seite 20

45 Die Schraube soll gleich rotieren
Thema: Schnellerer Aufbau der Schrauben-Grafik mithilfe eines
Maschinen-Programmes.
Heft 4/84 Seite 5

46 Schon wieder 'n Zap
Thema: Grafik-Ausdruck mit JKL - RAM-schonend!
Heft 8/84 Seite 10

Thema: INTERRUPT

47 Der I80 und seine Interrupts
Thema: Erlaeuterungen zu den verschiedenen Interrupts.
Heft 5/84 Seite 6

48 Noch etwas ueber INTERRUPTS
Thema: Fortsetzung der Erlaeuterungen der Interrupts.
Heft 5/84 Seite 24

49 Schon wieder INTERRUPTS
Thema: Fortsetzung der Beitrage in Heft 5/84.

Heft 8/84 Seite 16

50 Endlich geknackt: Der IM 2
Thema: Weitere Erlaeuterungen zum Thema "INTERRUPTs"
nach Heft 5 & 8/84.

Heft 11/85 Seite 3

51 DI ? - na und ?
Thema: Eine Anwendung des EG64 MBA:
Wieder-Einschalten der Interrupts mit RESET.

Heft 5/85 Seite 12

Thema: MATHEMATIK

52 Lissajoussche Figuren
Thema: Graf.Darstellung der Ueberlagerung von Sinus=
schwingungen (PIXEL- und HR-Grafik)

Heft 6/85 Seite 2

Thema: MEMORY SIZE

53 BASIC frei im RAM verlagern
Thema: Mal andersherum: BASIC ungefaehrlich machen und oberhalb des
zu schuetzendes Programmes hinlegen!

Heft 3/84 Seite 3

54 Die MEMORY SIZE automatisch
Thema: Das Maschinenprogramm, das selbsttaetig die Obergrenze fuer
BASIC setzt und sich damit vor ihm schuetzt.

Heft 3/84 Seite 14

Thema: SYSTEM-FILES

55 Spielereien mit Disk-Namen
Thema: Eingaben werden realisiert, die sonst im Namen- und
Datum-Feld nicht moeglich sind.

Heft 3/84 Seite 7

56 SYS-Files und wie man sie macht
Thema: Die Struktur von SYS-Files wird erkluert und wie man sie
implementieren kann.

Heft 9/84 Seite 3

57 Mehrere SYS-Files gleichzeitig
Thema: Wie eine Systemdatei jederzeit "im Hintergrund" abrufbereit
bleibt, ohne andere SYS-Files zu stoeren.

Heft 1/85 Seite 10

58 Der SYS3-Zap fuer G-DOS und NEWDOS80
Thema: Modifizierte Version des Zaps, der die Grafik-Hardcopy
in beiden Systemen unterstuetzt.

Heft 8/84 Seite 12

59 Die Library vergroessern
Thema: Richtige "Ellbogentechnik" schafft Platz
auch in der kleinsten Huette...

Heft 2/85 Seite 6

60 Was tun bei voller Library ?
Thema: Zusaetzhliche DOS-Befehle koennen trotzdem implementiert
werden mithilfe einer Zweit-Library.

Heft 4/86 Seite 11

Beitraege von A.SOPP im INFO des Bremerhavener GENIE/TRS80-Clubs

61 Neuer Dreitastenbefehl " , . / "
Thema: - ruft SYS26/SYS auf; hiermit laesst sich weiterwursteln...
(Entschuldige bitte die banale Diktion, lieber Arnulf!)
Heft 8/84 Seite 23

62 Die Records handhaben
Thema: Record-Struktur und Record-Codes am Beispiel "GDOS/SYS"
erklaert.
Heft 9/84 Seite 17

63 Mehr ueber DDE
Thema: Nuetzliche Aenderungen im SYS15/SYS.
Heft 11/84 Seite 30

64 Mini-RAM-Floppy im "sicheren Plaetzchen"
Thema: Maschinenprogramme bzw. Systemerweiterungen im Bereich
3900-3BFFh unterbringen (s.H.Bernhardts Schaltg.in c't 5/85)
Heft 10/85 Seite 7

65 Hardcopy unter DEBUG
Thema: Sperre f.Bildschirm Ausdruck m.JKL bei DEBUG ausgetrickst.
(gemeinsam mit Wolfgang Frey)
Heft 11/85 Seite 14

Thema: TASTATUR

66 Entprellung der Tastatur
Thema: Wessen Tastatur es noch oetig hat, kann sich hiermit helfen,
um Hazards zu verhindern.
Heft 6/84 Seite 3

67 Die Tastatur aufmoebeln
Thema: Zweite Tastatur ermoeöglichen und Keyboard komplettieren
(Hardware-Tips).
Heft 5/84 Seite 3

68 Funktionstasten fuer das GENIE I/II
Thema: Hardware-Tips fuer eine perfekte Tastatur
z.B. Umlauttasten, Kombitasten etc.
Heft 4/85 Seite 11

Thema: TOKEN

69 Ein mysterioeser BASIC-Befehl: IsA
Thema: Aufbau der Tokens-Tabelle nebst einer Bemerkung, die in
1/85, S.8 allerdings korrigiert wird.
Heft 12/84 Seite 15

70 Vom Token zum Befehlswort
Thema: Mehr ueber die Organisation von Tokens.

Heft 1/85 Seite 8

VERSCHIEDENES

71 BEL-Code fuer das GENIE I/II
Thema: Wie bringt man auch GENIE I u. II zum "BEL"len (Klingeln) ?
Heft 11/84 Seite 5

72 Sondertasten diverser GENIEs
Thema: Aenderungsvorschlaege fuer die Sondertasten im Ziffernblock
(Fortsetzung nach Seite 23 versehentlich auf Seite 25!)
Heft 4/86 Seite 23

Graphik-CLS für das Genie IIIs

Schon im alten H-DOS für das Genie I wurde der Library-Befehl CLS so verbogen, daß auch die HRG 1b damit gelöscht werden konnte. Das ist jetzt auch in H-DOS 2.4 für das Genie IIIs der Fall. Da die beiden Graphik-Systeme jedoch nicht gleich sind, mußte die Bearbeitungsroutine völlig neu geschrieben werden. Sie residiert zusammen mit der SYS-Routine (auch in diesem Info abgedruckt?) im letzten Sektor von SYS4/SYS, wo noch genügend Platz frei ist.

Beim Einsprung wird zunächst geprüft, ob der Requestcode im Akku 86h oder A6h für den Befehl SYS bzw. CLS beträgt. Ist das nicht der Fall, muß es sich um eine Anforderung für eine andere Routine in SYS4/SYS handeln. In diesem Falle wird nach 4D00h weitergesprungen, dem alten Entrypoint von SYS4. Handelt es sich um den umgemodelten Befehl CLS, dann war A6h im Akku, und es geht los:

Mit einem CALL nach 4CD5h wird überprüft, ob nach CLS noch weitere Parameter folgen. Falls nein, war es der alte Befehl, wie er schon immer möglich war. Dann wird HL als Zeiger auf die Bildschirmsteuerzeichen 1Ch und 1Fh geladen und diese ausgegeben. Das war dann schon alles. Folgt aber etwas nach, dann wird geprüft, welche(r) von drei möglichen Parametern mit eingegeben wurde(n).

Möglich sind V (Video löschen), 0 (Graphikseite 0 löschen) und 1 (Graphikseite 1 löschen). Wenn man gleich alles leeren will, kann man auch CLS,0,1,V eingeben, wobei die Reihenfolge und Anzahl der Parameter egal ist.

Im Kern passiert folgendes: Durch einen Output auf den Port FAh (Systembyte 1) mit gesetztem Bit 3 wird ab 8000h das Anwender-RAM gegen den Graphikspeicher ausgetauscht. Da er somit für die CPU direkt zugänglich ist, kann er mit dem einfachen LDIR-Konstrukt ab Zeile 37 im Listing gemächt werden. Bei der HRG 1b war das komplizierter.

Die Anwahl der gewünschten Graphikseite geschieht kurz davor: Wenn das Bit 5 des Systembytes 0 (Port F9h) gesetzt ist, wird die Graphikseite 1 zum Schreiben und Lesen freigegeben. Bei rückgesetztem Bit 5 ist es die Seite 0. Um im Kernprogramm auf beide Fälle vorbereitet zu sein, wird in Zeile 15 (Label clspar) zunächst das Register C mit 00h geladen. Bei Seite 1 wird in C das Bit 5 (s. o.) gesetzt, bei Seite 0 wird dieser Befehl übersprungen. Wenn nun in Zeile 30 der Akku-Inhalt (alter Zustand von Systembyte 0) mit C oderiert wird, ist die entsprechende Seite selektiert.

Wenn sie schließlich gelöscht ist, wird beim Label getpar (Zeile 47) nach weiteren Parametern geforscht. Gibt es keine mehr, geht's zurück ins Betriebssystem. Andernfalls geht das alte Spiel bei clspar von vorne los.

Um das Löschen der Graphik auch unter BASIC zu ermöglichen, muß dafür Sorge getragen werden, daß der Stack nichts zu tun hat, während auf dem Port FAh das Bit 5 gesetzt ist. BASIC unterhält den Stack ziemlich hoch im Speicher, also ausgerechnet dort, wo vorübergehend die Graphik eingeblendet ist. Würden nun in Zeile 34 die Registerpaare HL und DE mit PUSH gerettet werden, dann überschreibe der Löschvorgang den Stack mit Nullen. Beim anschließenden POP stünde demnach 0000h in beiden Registerpaaren. Der Befehl EXX tauscht stattdessen die CPU-Register gegen den alternativen Satz aus, so daß hier keine Gefahr droht.

Die Sektordumps, in denen wie üblich die geänderten Codes unterstrichen sind, zeigen der Reihe nach folgendes:

1. Sektor 02 von SYS1/SYS, Änderung am Befehl CLS
2. Sektor 04 von SYS4/SYS, Programm und geänderter Einsprung

Zusätzlich kann die ehemalige CLS-Routine in SYS1/SYS jetzt überschrieben werden. Sie ist jedoch nur sechs Bytes lang, so daß kaum etwas machbar ist. Aber die Codes 1Ch-1Fh-03h im letzten Sektor von SYS1/SYS sind nun frei und können für eine etwas längere, vielleicht lustigere DOS-Ready-Meldung ausgenutzt werden.

Arnulf Sopp

H-DOS-Diskdaten-Editor Vers. 2.4
(C) 1986 by TCS / The HACKTORY

Datei: SYS1/SYS

```

drv: 0          frs: 0002h          drs: 0598h
000200: D54C 20A9 CB59 2902 0102 004F E3E5 79E6 L Y( 0 y
000210: 0728 0EE5 21BC 5123 2323 3D20 FACD 2A4F ( ! Q###= *D
000220: E179 01D3 49C5 CB7F C806 0021 0042 CB77 y I ! B w
000230: CA24 44C3 2044 D5C5 011C 091A FE3A 280A $D D : (
000240: FE2F 3806 281B 0D13 10F1 2323 E5EB 0600 /B ( ##
000250: 0954 5D2B 1313 13ED B8E1 0E03 EDB8 3E2F TU+ >/
000260: 12C1 D1C9 4149 4B80 5300 4150 5045 4E44 AIK S APPEND
000270: C068 0041 5454 5249 4285 E988 4155 544F h ATTRIB AUTO
000280: 80DC 0042 3286 EB00 424C 81E5 0042 4F4F B2 BL BOO
000290: 548A EB10 4252 4541 4B85 E500 434C 5380 T BREAK CLS
0002A0: A600 434F 4E54 C5EB 0043 4F50 59C0 4800 CONT COPY H
0002B0: 4352 4541 5445 82F0 4044 4154 554D 8BE9 CREATE $DATUM
0002C0: 0044 4445 81F1 0044 4952 802A 0044 4953 DDE DIR * DIS
0002D0: 4E83 FF00 444F C3EB 8A44 5282 FE00 4455 K DO DR DU
0002E0: 4D50 87E9 C845 87F0 0046 4F52 4D88 FE00 MP E FORM
0002F0: 4652 4545 804A 0046 2380 FB00 4849 4D45 FREE J F# HIME

```

H-DOS-Diskdaten-Editor Vers. 2.4
(C) 1986 by TCS / The HACKTORY

Datei: SYS4/SYS

```

drv: 0          frs: 0004h          drs: 05C7h
000400: E544 6973 6B65 7474 65EE 4174 7472 6962 Diskette Attrib
000410: 01EA 0051 75F4 7363 686C 6563 6874 65F2 Qu schlechte
000420: 4B65 696E E518 1818 6665 686C 65F2 6265 Kein fehle be
000430: E975 6E7A 756C 6165 7373 6967 E508 F3FE unzuLaessig
000440: 8628 56FE A6C2 004D CDD5 4C20 0921 3F51 (V M L !?Q
000450: C367 441C 1F03 0E00 7ED6 3028 103D 280B gD B O( =(
000460: FE25 2031 E5CD 3951 E118 23CB E9DB F957 % 1 9Q # W
000470: B1D3 F9DB FA5F D9CB DFD3 FA21 0080 1101 !
000480: 8001 FF7F 75ED B0D9 7AD3 F97B D3FA 23CD u z ä #
000490: D54C C818 C13E AF18 225E 2356 23CD D54C L > "#V# L
0004A0: 3819 E506 03D5 21AB 515E 2356 D723 E34F 8 ! Q^#V # 0
0004B0: DF79 E328 0410 F23E 34E1 E1C2 0944 EF49 y ( >4 D I
0004C0: 443D 494F A548 435D 0000 0000 0000 0000 D=ID HCU
0004D0: 0000 0000 0000 0000 0000 0000 0000 0000
0004E0: 0000 0000 0000 0000 0000 0000 0000 0000
0004F0: 0000 0000 0000 0000 0000 0000 0202 2B51 +0

```

00001 ;Systemerweiterg. am Ende von SYS4/SYS für den SYS-Befehl
 00002 ;und die Varianten des CLS-Befehls

```

00003
512B      00004      ORG      512bh      ;ab hier Platz in SYS4
512B FE86      00005 sys4    CP      86h      ;SYS-Befehl?
512D 2856      00006      JR      Z,sys    ;falls ja
512F FEA6      00007      CP      0A6h     ;CLS-Befehl?
5131 C2004D     00008      JF      NZ,4D00h ;falls nein
00009
5134 CDD54C     00010      CALL   4cd5h     ;Trennzeichen erkennen
5137 2009      00011      JR      NZ,clspar ;falls Parameter folgen
5139 213F51     00012 clsv    LD      HL,cls    ;nein, nur ASCII-Bildsch.
513C C36744     00013      JF      4467h     ;diesen löschen
513F 1C        00014 cls    DB      1ch,1fh,03h ;Codes für CLS
5142 0E00      00015 clspar LD      C,00h     ;Anfangswert OR-Operand
5144 7E        00016      LD      A,(HL)    ;was folgt nach CLS?
5145 D630      00017      SUB    '0'        ;Graphikseite 0 löschen?
5147 2810      00018      JR      Z,clsgra0 ;falls ja
5149 3D        00019      DEC    A          ;Graphikseite 1 löschen?
514A 280B      00020      JR      Z,clsgral ;falls ja
514C FE25      00021      CP      '%?'      ;ASCII-B.? (ehem. "V")
514E 2031      00022      JR      NZ,error1 ;falls nichts von alledem
5150 E5        00023      PUSH   HL         ;retten, wird verändert
5151 CD3951     00024      CALL   clsv       ;ASCII-Bildschirm löschen
5154 E1        00025      POP    HL         ;restaurieren
5155 1823      00026      JR      getpar    ;nächster Parameter
5157 CBE9      00027 clsgral SET    5,C        ;Sysbyte 0, Bit 5: Gr. 1
5159 DBF9      00028 clsgra0 IN    A,(0f9h)   ;Systembyte 0 lesen
515B 57        00029      LD      D,A       ;altes Systembyte retten
515C B1        00030      OR     C          ;Graphikseite selektieren
515D D3F9      00031      OUT    (0f9h),A  ;Systembyte neu ausgeben
515F DBFA      00032      IN    A,(0fah)   ;Systembyte 1
5161 5F        00033      LD      E,A       ;retten
5162 D9        00034      EXX                    ;Reg. ohne Stack retten
5163 CBDF      00035      SET    3,A       ;Graphikseite freigeben
5165 D3FA      00036      OUT    (0fah),A  ;Systembyte neu ausgeben
5167 2100B0    00037      LD      HL,8000h  ;Anfang der Graphik
516A 110180    00038      LD      DE,8001h  ;nächste Stelle
516D 01FF7F    00039      LD      BC,7fffh  ;Länge des Gr.-speichers
5170 75        00040      LD      (HL),L    ;Stelle ausnullen
5171 EDB0      00041      LDIR                    ;dto. kpl. Graphikspeich.
5173 D9        00042      EXX                    ;Register restaurieren
5174 7A        00043      LD      A,D       ;altes Systembyte 0
5175 D3F9      00044      OUT    (0f9h),A  ;restaurieren
5177 7B        00045      LD      A,E       ;dto. Systembyte 1
5178 D3FA      00046      OUT    (0fah),A  ;
517A 23        00047 getpar  INC    HL         ;weiter im Befehl
517B CDD54C     00048      CALL   4cd5h     ;Trennzeichen erkennen
517E C8        00049      RET    Z          ;falls Befehl zuende
517F 18C1      00050      JR      clspar    ;sonst weitere Parameter
5181 3EAF      00051 error1 LD    A,0afh     ;"falsche Parameter"
5183 1822      00052      JR      error2    ;anzeigen und zurück
00053
00054 ;Hier folgt die Bearbeitungsroutine des
00055 ;Library-Befehls SYS (mit LIST OFF ausgespart).
00056
51A7 C20944     00084 error2 JP    NZ,4409h ;Fehleranzeige und zurück
00091
512B      00092      END    sys4      ;dort neuer Einsprung

```

00000 Fehler

E. Sikora, Von-Hessen-Str. 18 5040 Brühl

EINGEGANGEN 19. Juli 1986

An den

Genie/TRS-80 User Club

c/o Ralf Folkerts
Nutzhorner Str. 9

2875 Bockholzberg-Ganderkeese 2

M-Nr.: 850630

Betr.: HRG-Hardcopy im Info 6/86

Lieber Ralf!

A. Von Arnulf Sopp habe ich einen Brief betreffend meinen Beitrag im letzten Info bekommen, wo er mich darauf aufmerksam macht, daß ich noch folgendes vergessen habe mitzuteilen:

1. Der abgebildete HEXDUMP ist ein Sektordump mit den zugehörigen Record-Codes, die beim Eintippen natürlich weglassen werden müssen.
2. Vor dem Eintippen muß zuvor ein Dummy-File von einem Sektor Länge eröffnet werden (z.B. mit DUMP), in das dann die Codes eingetippt werden können.

Für diejenigen, die nicht zurecht kommen lege ich noch ein Listing des Source-Files bei.

B. Eine Information für 80-Zeichen-Karten-Besitzer mit einem EPSON-Drucker:

Die Firma Computerstudio Euskirchen, Nordstr. 45, 5350 EUSKIRCHEN vertreibt EPROMS für EPSON-Drucker mit dem IBM-Zeichensatz, der dem der 80-Zeichen-Karte fast entspricht. Dieser Zeichensatz wird anstelle des Alternativ-Zeichensatzes dargestellt.

C. Für die Schmunzelecke:

Ein Indianerstamm wollte wissen wie streng der nächste Winter werde. Der Mediziner zerteilte die Leber einer Krähe und sah, der Winter werde streng.

Fleißig sammelte der Stamm Holz, doch Ende Oktober war es immer noch warm. Sie zweifelten nun an der Voraussage-Methode und gingen zum GROSSEN RECHNER.

Der COMPUTER kam auch zu dem Ergebnis: der Winter wird streng. Auf die Frage warum, antwortete der COMPUTER: "Seit einem Monat sammeln die Indianer Holz".

Mit freundlichen Grüßen

Ernst

09/86 → 47

MARKUSI 2/01

```

00100 ORG 0FD00H
00110 STRAM DEFS 2
00120 SPALT DEFS 1
00130 FLAG DEFS 1
00140 Z78 DEFS 1
00150 ADRAM DEFS 2
00160 ADBUF DEFS 2
00170 IZ12 DEFS 1
00180 ADRU EQU 14312
00200 BUFFER DEFS 512
00210 PUSH IX
00220 PUSH HL
00230 PUSH DE
00240 PUSH BC
00250 PUSH AF
00260 LD A, (FLAG)
00270 CP 1
00280 JR Z, NR60
00290 LD HL, (STRAM)
00300 LD (ADRAM), HL
00310 LD A, 12
00320 LD (IZ12), A
00330 LD A, 1
00340 LD (FLAG), A
00350 NR60 CALL L858
00360 CALL KZDRU
00370 POP AF
00380 POP BC
00390 POP DE
00400 POP HL
00410 POP IX
00420 RET
00500 ;SUB L8/S8 LESEN UND SPEICHERN VON 8 GRAFIKZEILEN
00510 L858 LD A, 8
00520 LD (Z78), A
00530 LD HL, BUFFER
00540 LD (ADBUF), HL
00550 NR13 CALL L151
00560 LD A, (Z78)
00570 DEC A
00580 CP 0
00590 JR Z, NR20
00600 LD (Z78), A
00610 CALL ADR
00620 LD DE, (ADBUF)
00630 INC DE
00640 LD (ADBUF), DE
00650 JR NR13
00660 NR20 CALL ADR
00670 RET
00700 ;SUB ADR ZUR ADRESSENBERECHNUNG
00710 ADR LD A, (IZ12)
00720 DEC A
00730 CP 0
00740 JR Z, NR12
00745 LD (IZ12), A
00750 LD DE, 1024
00760 LD HL, (ADRAM)
00770 ADD HL, DE
00780 LD (ADRAM), HL
00790 RET
00800 NR12 LD A, 12
00810 LD (IZ12), A
00820 LD DE, 64
00830 LD HL, (STRAM)
00840 ADD HL, DE
00850 LD (STRAM), HL
00860 LD (ADRAM), HL
00870 RET

```

```

00910 L151 LD A, (SPALT)
00920 LD B, A
00930 LD HL, (ADRAM)
00940 LD DE, (ADBUF)
00950 NR3 LD C, 2
00960 OUT (C), L
00970 INC C
00980 OUT (C), H
00990 INC C
01000 IN A, (C)
01010 LD (DE), A
01020 DJNZ NR1
01030 RET
01040 NR1 INC HL
01050 PUSH HL
01060 LD HL, 8
01070 ADD HL, DE
01080 PUSH HL
01090 POP DE
01100 POP HL
01110 JR NR3
01200 ;SUB KZDRU 6 BYTE 8 ANZAHL SPALTEN FUER DRUCKER
01230 KZDRU LD HL, BUFFER
01240 LD (ADBUF), HL
01250 LD A, (SPALT)
01260 * PUSH AF
01270 NR40 CALL BYTE6
01280 POP AF
01290 DEC A
01300 CP 0
01310 RET Z
01320 PUSH AF
01330 JR NR40
01400 ;SUB BYTE6 ERZEUGEN VON 6 DRUCKERBYTES JE 8 ZEILEN HOCH
01410 BYTE6 LD C, 6
01420 NR31 CALL ROTGR
01430 DEC C
01440 PUSH AF
01450 LD A, C
01460 CP 0
01470 JR Z, NR35
01480 POP AF
01490 JR NR31
01500 NR35 POP AF
01510 LD DE, 8
01520 LD HL, (ADBUF)
01530 ADD HL, DE
01540 LD (ADBUF), HL
01550 RET
01600 ;SUB ROTGR ERZEUGEN EINER DRUCKERSPALTE 8 BIT HOCH
01610 ROTGR LD B, 8
01620 LD HL, (ADBUF)
01640 OR A
01645 LD A, 8
01650 NR30 RRC (HL)
01660 RLA
01670 INC HL
01680 DJNZ NR30
01682 PUSH AF
01684 NR100 LD A, (ADRU)
01686 CP 63
01688 JR NZ, NR100
01690 POP AF
01692 LD (ADRU), A
01720 RET
01750 END

```

HARDCOPY/SRC

```

00100 ORG 0F800H
00110 STRAM DEFS 2
00120 SPALT DEFS 1
00130 FLAG DEFS 1
00140 Z78 DEFS 1
00150 ADRAM DEFS 2
00160 ADBUF DEFS 2
00170 IZ12 DEFS 1
00180 ADRU EQU 14312
00200 BUFFER DEFS 260
00210 PUSH IX
00220 PUSH HL
00230 PUSH DE
00240 PUSH BC
00250 PUSH AF
00260 LD A, (FLAG)
00270 CP 1
00280 JR Z, NR60
00290 LD HL, (STRAM)
00300 LD (ADRAM), HL
00310 LD A, 12
00320 LD (IZ12), A
00330 LD A, 1
00340 LD (FLAG), A
00350 NR60 CALL L858
00360 CALL KZDRU
00370 POP AF
00380 POP BC
00390 POP DE
00400 POP HL
00410 POP IX
00420 RET
00500 ;SUB L8/S8 LESEN UND SPEICHERN VON 8 GRAFIKZEILEN
00510 L858 LD A, 4
00520 LD (Z78), A
00530 LD HL, BUFFER
00540 LD (ADBUF), HL
00550 NR13 CALL L1S1
00560 LD A, (Z78)
00570 DEC A
00580 CP 0
00590 JR Z, NR20
00600 LD (Z78), A
00610 CALL ADR
00620 LD DE, (ADBUF)
00630 INC DE
00640 LD (ADBUF), DE
00650 JR NR13
00660 NR20 CALL ADR
00670 RET
00700 ;SUB ADR ZUR ADRESSENBERECHNUNG
00710 ADR LD A, (IZ12)
00720 DEC A
00730 CP 0
00740 JR Z, NR12
00745 LD (IZ12), A
00750 LD DE, 1024
00760 LD HL, (ADRAM)
00770 ADD HL, DE
00780 LD (ADRAM), HL
00790 RET
00800 NR12 LD A, 12
00810 LD (IZ12), A
00820 LD DE, 64
00830 LD HL, (STRAM)
00840 ADD HL, DE

```

```

00900 ;SUB L1/S1 LESEN UND SPEICHERN EINER GRAFIKZEILE
00910 L1S1 LD A, (SPALT)
00920 LD B, A
00930 LD HL, (ADRAM)
00940 LD DE, (ADBUF)
00950 NR3 LD C, 2
00960 OUT (C), L
00970 INC C
00980 OUT (C), H
00990 INC C
01000 IN A, (C)
01010 LD (DE), A
01020 DJNZ NR1
01030 RET
01040 NR1 INC HL
01050 PUSH HL
01060 LD HL, 4
01070 ADD HL, DE
01080 PUSH HL
01090 POP DE
01100 POP HL
01110 JR NR3
01200 ;SUB KZDRU 6 BYTE 8 ANZAHL SPALTEN FUER DRUCKER
01230 KZDRU LD HL, BUFFER
01240 LD (ADBUF), HL
01250 LD A, (SPALT)
01260 PUSH AF
01270 NR40 CALL BYTE6
01280 POP AF
01290 DEC A
01300 CP 0
01310 RET Z
01320 PUSH AF
01330 JR NR40
01400 ;SUB BYTE6 ERZEUGEN VON 6 DRUCKERBYTES JE 8 ZEILEN HOCH
01410 BYTE6 LD C, 6
01420 NR31 CALL ROTGR
01430 DEC C
01440 PUSH AF
01450 LD A, C
01460 CP 0
01470 JR Z, NR35
01480 POP AF
01490 JR NR31
01500 NR35 POP AF
01510 LD DE, 4
01520 LD HL, (ADBUF)
01530 ADD HL, DE
01540 LD (ADBUF), HL
01550 RET
01600 ;SUB ROTGR ERZEUGEN EINER DRUCKERSPALTE 8 BIT HOCH
01610 ROTGR LD B, 4
01620 LD HL, (ADBUF)
01640 OR A
01645 LD A, 0
01650 NR30 RRC (HL)
01660 RLA
01662 RLA
01670 INC HL
01680 DJNZ NR30
01682 PUSH AF
01684 NR100 LD A, (ADRU)
01686 CP 63
01688 JR NZ, NR100
01690 POP AF
01692 LD (ADRU), A
01720 RET
01750 END

```

HRG-Hardcopy im Maßstab Version 2

Ich habe das Programm von Ernst Sikora im INFO Nr.6 teilweise umgeschrieben und die normale #LPRINT-Routine im HRG-1b damit ersetzt. Es ist eine gute Idee, jede zweite "Dot-Zeile" durch ein Blank zu ersetzen! Damit hat man am FX-80 (RX-80) eine eine maßstabgerechte Kopie vom Bildschirm.

Zum Assemblerlisting: Die Startadresse der #LPRINT-Routine ist bei meinem HRG-1b FE4B Hex, die des Buffers im HRG ist F10D Hex. Ob es da eine Übereinstimmung in allen HRG-1b gibt, wage ich zu bezweifeln. Deshalb ist beim Linken der neuen Routine Vorsicht geboten! Wer da keine Erfahrungen hat, kann mir sein HRG-1b-Programm auf Diskette im Format: 40 Trk. SS/SD/DD mit entsprechendem Rückporto schicken.

Man kann auch das Programm als HARD/CMD benutzen wenn man die Zeilen 00730-00740 abändert.

```
FEC8 23      00730      INC      HL
FEC9 C31E1D  00740      JP       1D1EH
```

In:

```
FEC8 C9      00730      RET
FEC9 C31E1D  00740      kann bleiben oder 3xNOP
```

Der Buffer in F10D Hex kann bleiben oder irgendwo sein wo er nicht stört.

Karl Rubes (meine Anschrift ist beim Ralf Folkerts).

*e-Uhr im Video-Genie
(Die weinstehe Uhr, c'1/86)*
Tip für alle Video-Genie-1/II-
Benutzer: Man unterbreche die
Verbindung von Pin 6
IC 74LS30 zu Pin 21 der
EPROM-Fassung auf der Pla-
tine und verbinde dann Pin 5 des
LS30 mit Pin 6 des gleichen ICs.
Diese Arbeiten lassen sich auch
nachträglich am fertigen Bau-
stein auf der Lötseite der Platine
ausführen. Damit hat man er-
reicht, daß das PAL in einem
anderen Bereich angesprochen
wird, der frei ist von sonstigen
rechnerinternen Belangen. Im
Video-Genie I liegt die Uhr
jetzt also im Bereich
3000H...33FFH, die EPROM-
Bytes beträgt also jetzt 400H-
software) (wichtig für die Uhren-
BASIC lesen will, kann nach
etwas Umformen will, kann nach
druckte Programm das abge-
verwenden, sollte jedoch vor
Lesen der Register durchaus
58321 die Interrupts des RTC
ten (mit CMD 'T').
Rainer Knoch, 6450 Hanau I*

c'1 1986, Heft 9

```

00010 ;*****
00020 ;***           H R G - 1 b           ***
00030 ;*** Neue #LPRINT-Routine für FX-80 Drucker. ***
00040 ;***           Ausdruck im Bildschirmformat. ***
00050 ;*** übernommen: Ernst Sikora (Info Nr.6) ***
00060 ;*** Angepaßt: Karl Rubes           ***
00070 ;*****
00080 ;

```

```

FE4B 00090          ORG      0FE4BH          ;BEGINN: #LPRINT (HR6)
00100 ;
F10D 00110 BUFFER EQU      0F10DH          ;256 BYTE BUFFER (HR6)
FE4B 0000 00120 STADR  DW      0000H          ;STARTADRESSE
FE4D 40 00130 SPALTE DB      64             ;64 CHAR./ZEILE
FE4E 30 00140 ZEILE  DB      48             ;ZAEHLER 48...1 (16*3=48)
FE4F 00 00150 DOT4   DB      0              ;ZAEHLER: 4...0
FE50 00 00160 DOT12  DB      0              ;ZAEHLER: 12...0
FE51 0000 00170 AKTADR DW      0000H          ;AKT.ADRESSE
FE53 0000 00180 ZEIGER DW      0000H          ;ZEIGER BUFFER
00190 ;-----
FE55 E5 00200 LPRINT  PUSH   HL
FE56 3E1B 00210          LD      A,1BH          ;RESET PRINTER
FE58 CD81FF 00220          CALL   PRINT
FE5B 3E40 00230          LD      A,⊙
FE5D CD81FF 00240          CALL   PRINT
00250 ;-----
FE60 3E1B 00260          LD      A,1BH          ;LINKER RAND = 8
FE62 CD81FF 00270          CALL   PRINT
FE65 3E6C 00280          LD      A,6CH
FE67 CD81FF 00290          CALL   PRINT
FE6A 3E08 00300          LD      A,8
FE6C CD81FF 00310          CALL   PRINT
00320 ;-----
FE6F 3E1B 00330          LD      A,1BH          ;LINEFEED = 8 DOT
FE71 CD81FF 00340          CALL   PRINT
FE74 3E41 00350          LD      A,'A'
FE76 CD81FF 00360          CALL   PRINT
FE79 3E08 00370          LD      A,8
FE7B CD81FF 00380          CALL   PRINT
00390 ;-----
FE7E 210000 00400          LD      HL,0000H          ;ADRESSEN RESTAURIEREN
FE81 224BFE 00410          LD      (STADR),HL
FE84 2251FE 00420          LD      (AKTADR),HL
FE87 3E0C 00430          LD      A,12
FE89 3250FE 00440          LD      (DOT12),A
FE8C 3E30 00450          LD      A,48          ;16*3=48
FE8E 324EFE 00460          LD      (ZEILE),A          ;SAVE
00470 ;-----
FE91 3E1B 00480 NEXTZ  LD      A,1BH          ;480 DOT-DICHTE
FE93 CD81FF 00490          CALL   PRINT
FE96 3E4B 00500          LD      A,'K'
FE98 CD81FF 00510          CALL   PRINT
FE9B 3E80 00520          LD      A,128          ;384-256=128
FE9D CD81FF 00530          CALL   PRINT
FEA0 3E01 00540          LD      A,1          ;INT(384/256)=1
FEA2 CD81FF 00550          CALL   PRINT
00560 ;-----
FEA5 CDCCFE 00570          CALL   MAL4
FEA8 CD42FF 00580          CALL   SAVBUF
FEAB 3E0D 00590          LD      A,0DH
FEAD CD81FF 00600          CALL   PRINT
FEB0 3A4EFE 00610          LD      A,(ZEILE)
FEB3 FE01 00620          CP      1

```

FEB5	2806	00630	JR	Z, FERTIG	
FEB7	3D	00640	DEC	A	
FEB8	324EFE	00650	LD	(ZEILE), A	
FEBB	18D4	00660	JR	NEXTZ	
		00670	;-----		
FEBD	3E1B	00680	FERTIG	LD	A, 1BH ; FERTIG, RESET PRINTER
FEBF	CD81FF	00690	CALL	PRINT	
FEC2	3E40	00700	LD	A, 0	
FEC4	CD81FF	00710	CALL	PRINT	
FEC7	E1	00720	POP	HL	
FEC8	23	00730	INC	HL	; NEXT TOKEN!
FEC9	C31E1D	00740	JP	1D1EH	; SYNTAX/AUSFUEHREN
		00750	;-----		
FECC	3E04	00760	MAL4	LD	A, 4
FECE	324FFE	00770		LD	(DOT4), A
FED1	210DF1	00780		LD	HL, BUFFER
FED4	2253FE	00790		LD	(ZEIGER), HL
FED7	CD20FF	00800	LOOP4	CALL	READ
FEDA	3A4FFE	00810		LD	A, (DDT4)
FEDD	3D	00820		DEC	A
FEDE	FE00	00830		CP	0
FEE0	2811	00840	JR	Z, HFESF	
FEE2	324FFE	00850		LD	(DOT4), A
FEE5	CD7FE	00860	CALL	MAL12	
FEE8	ED5B53FE	00870		LD	DE, (ZEIGER)
FEEC	13	00880	INC	DE	
FEED	ED5353FE	00890		LD	(ZEIGER), DE
FEF1	18E4	00900	JR	LOOP4	
		00910	;-----		
FEF3	CD7FE	00920	HFESF	CALL	MAL12
FEF6	C9	00930		RET	
		00940	;-----		
FEF7	3A50FE	00950	MAL12	LD	A, (DOT12)
FEFA	3D	00960		DEC	A
FEFB	FE00	00970		CP	0
FEFD	280E	00980	JR	Z, NEXT12	
FEFF	3250FE	00990		LD	(DOT12), A
FF02	110004	01000		LD	DE, 0400H
FF05	2A51FE	01010		LD	HL, (AKTADR)
FF08	19	01020	ADD	HL, DE	
FF09	2251FE	01030		LD	(AKTADR), HL
FF0C	C9	01040		RET	
		01050	;-----		
FF0D	3E0C	01060	NEXT12	LD	A, 12
FF0F	3250FE	01070		LD	(DOT12), A
FF12	114000	01080		LD	DE, 0040H
FF15	2A4BFE	01090		LD	HL, (STADR)
FF18	19	01100	ADD	HL, DE	
FF19	224BFE	01110		LD	(STADR), HL
FF1C	2251FE	01120		LD	(AKTADR), HL
FF1F	C9	01130		RET	
		01140	;-----		
FF20	3A4DFE	01150	READ	LD	A, (SPALTE)
FF23	47	01160		LD	B, A
FF24	2A51FE	01170		LD	HL, (AKTADR)
FF27	ED5B53FE	01180		LD	DE, (ZEIGER)
FF2B	0E02	01190	LOOPR	LD	C, 2
FF2D	ED69	01200		OUT	(C), L
FF2F	0C	01210	INC	C	
FF30	ED61	01220		OUT	(C), H
FF32	0C	01230	INC	C	
FF33	ED78	01240	IN	A, (C)	

FF35	12	01250		LD	(DE),A
FF36	23	01260		INC	HL
FF37	E5	01270		PUSH	HL
FF38	210400	01280		LD	HL,0004H
FF3B	19	01290		ADD	HL,DE
FF3C	E5	01300		PUSH	HL
FF3D	D1	01310		POP	DE
FF3E	E1	01320		POP	HL
FF3F	10EA	01330		DJNZ	LOOPR
FF41	C9	01340		RET	
		01350	;-----		
FF42	210DF1	01360	SAVBUF	LD	HL,BUFFER
FF45	2253FE	01370		LD	(ZEIGER),HL
FF48	3A4DFE	01380		LD	A,(SPALTE)
FF4B	F5	01390		PUSH	AF
FF4C	CD57FF	01400	LOOPSP	CALL	MAL6
FF4F	F1	01410		POP	AF
FF50	3D	01420		DEC	A
FF51	FE00	01430		CP	0
FF53	C8	01440		RET	Z
FF54	F5	01450		PUSH	AF
FF55	18F5	01460		JR	LOOPSP
		01470	;-----		
FF57	0E06	01480	MAL6	LD	C,6
FF59	CD72FF	01490	LOOP6	CALL	LEERZ
FF5C	0D	01500		DEC	C
FF5D	F5	01510		PUSH	AF
FF5E	79	01520		LD	A,C
FF5F	FE00	01530		CP	0
FF61	2803	01540		JR	Z,HFED4
FF63	F1	01550		POP	AF
FF64	18F3	01560		JR	LOOP6
		01570	;-----		
FF66	F1	01580	HFED4	POP	AF
FF67	110400	01590		LD	DE,0004H
FF6A	2A53FE	01600		LD	HL,(ZEIGER)
FF6D	19	01610		ADD	HL,DE
FF6E	2253FE	01620		LD	(ZEIGER),HL
FF71	C9	01630		RET	
		01640	;-----		
FF72	0604	01650	LEERZ	LD	B,4
FF74	2A53FE	01660		LD	HL,(ZEIGER)
FF77	B7	01670		OR	A
FF78	3E00	01680		LD	A,0
FF7A	CB0E	01690	LOOPSH	RRC	(HL)
FF7C	17	01700		RLA	
FF7D	17	01710		RLA	
FF7E	23	01720		INC	HL
FF7F	10F9	01730		DJNZ	LOOPSH
FF81	F5	01740	PRINT	PUSH	AF
FF82	3AE837	01750	BUSY	LD	A,(37EBH)
FF85	FE3F	01760		CP	3FH
FF87	20F9	01770		JR	NZ,BUSY
FF89	F1	01780		POP	AF
FF8A	32E837	01790		LD	(37EBH),A
FF8D	C9	01800		RET	
		01810	;-----		
FE55		01820		END	LPRINT

00000 Total errors

Ein neuer Lib-Befehl: GO<,adresse>

Es ist nicht selten sinnvoll, an eine beliebige Stelle im Speicher springen zu können. Alle halbwegs ernstzunehmenden Monitoren bieten diese Möglichkeit an, auch unser Debugger. Dieser ist jedoch in der Anwendung sehr eingeschränkt: Er läuft nicht ohne Risiko unter Mini-DOS und kann nur sehr umständlich in JOB-Dateien eingebaut oder auf F-Tasten programmiert werden. Selbst wenn man es tut, ist jemand an der Tastatur kaum entbehrlich. Schließlich findet er die Einsprungsadresse einer CMD-Datei nicht selbsttätig, was das hier vorgestellte Programm kann. Es ist oben drein phosphatfrei und grundwasserneutral.

Auf jeden Fall sollte es in eine SYS-Datei gepatcht werden, um alle Möglichkeiten ausnutzen zu können. In diesem Fall wurde SYS9/SYS von G-DOS 2.4 benutzt, wo noch genügend Platz ist. Die neue Entry-Adresse ist im Hex-Dump auf der nächsten Seite doppelt unterstrichen. Wer in seinem SYS9 diesen Platz nicht mehr hat oder im schraffierten Teil, besonders an der einfach unterstrichenen Stelle (Ladecodes) Unterschiede feststellt, müßte u. U. auf ein anderes SYS-File ausweichen. Wie dann der Requestcode für diesen Befehl anzupassen ist, wurde im Info schon verschiedentlich beschrieben. Ebenfalls ist bekannt, wie das neue Befehlswort GO in die Library zu implementieren ist.

Die Befehlssyntax geht aus der Überschrift hervor. Wenn nur GO ohne nachfolgende Adresse eingegeben wird, holt das Programm das Sprungziel aus den Speicherstellen 4403/04h. Dort wird die Entry eines CMD-Files beim Laden abgelegt, auch dann, wenn nur LOAD,filename eingegeben wurde. Ein wenig Vorsicht ist hier geboten. Hartmut Grosser stellt in seinem "DOS-Buch" fest, daß dies der einzige Verwendungszweck für 4403/04h ist und schließt daraus messerscharf, daß man dort im DOS-CALL-Status Parameter an ein Benutzerprogramm übergeben darf. Nach dieser ziemlich überflüssigen Anregung (es gibt noch mehr solche Freistellen) könnte es nun sein, daß dort in der Tat zu irgendeinem Zeitpunkt nicht mehr die Einsprungsadresse eines Programms steht, weil ein nicht selbst geschriebenes Programm dort herumwühlt. Zumindest unmittelbar nach dem Laden kann man sich jedoch darauf verlassen, daß GO ohne Angabe einer Adresse das Programm zuverlässig anspringt.

Wird eine Adresse durch Komma oder Blank getrennt mit angegeben, wird an diese Adresse gesprungen. Hat man also mehrere Programme im Speicher, so kann jedes von ihnen aufgerufen werden. Ohne Angabe der Adresse wird jedoch nur das zuletzt geladene gefunden. Auf diese Weise ist es z. B. auch möglich, ROM- oder DOS-Routinen aufzurufen, die normalerweise aus der Kommandoebene nicht zugänglich sind. Sie werden dabei wie Unterprogramme behandelt, müssen also mit RET abschließen. Tun sie das nicht, dann kann man u. U. in Level 2, im Debugger oder im Nirwana landen, je nach dem.

Das Programm selbst (Listing umseitig) ist mit den Kommentaren wohl hinreichend erläutert. Es sei nur hinzugefügt, daß die optionale Adresse immer vierstellig in Hex eingegeben werden muß. Der Hexkennner "h" o. dergl. erübrigt sich, denn es werden bei einer Adresse nur die ersten vier Stellen gelesen. Was nachfolgt, wird einfach ignoriert.

Arnulf Sopp

```

5147          00001      ORG      5147h      :ab hier Platz in SYS9
          00002
5147 FE6B      00003 sys9   CP       6bh          ;Lib-Befehl GO?
5149 C2004D    00004      JP       NZ,4d00h   ;sonst dort weiter
514C CDD54C    00005      CALL    4cd5h      ;folgt eine Sprungadr.?
514F 2004      00006      JR       NZ,param   ;falls ja
5151 2A0344    00007      LD       HL,(4403h) ;nein, aktuelle Startadr.
5154 E9          00008      JP       (HL)       ;Programm anspringen
          00009
5155 2B          00010 param DEC     HL          ;wegen RST 10 Zeiger zur.
5156 CD6C51    00011      CALL    read       ;Ziffer einlesen
5159 57          00012      LD       D,A        ;1. Hälfte des MSB
515A CD6C51    00013      CALL    read       ;nächste Ziffer
515D 79          00014      LD       A,C        ;unteres Nibble
515E B2          00015      OR      D           ;komplettes MSB
515F 57          00016      LD       D,A        ;laden
5160 CD6C51    00017      CALL    read       ;nächste Ziffer
5163 5F          00018      LD       E,A        ;1. Hälfte des LSB
5164 CD6C51    00019      CALL    read       ;nächste Ziffer
5167 79          00020      LD       A,C        ;unteres Nibble
5168 B3          00021      OR      E           ;komplettes LSB
5169 6F          00022      LD       L,A        ;L <- Adreß-LSB
516A 62          00023      LD       H,D        ;HL <- komplette Adresse
516B E9          00024      JP       (HL)       ;Adresse anspringen
          00025
516C D7          00026 read  RST     10h       ;Ziffer aus dem String
516D 380A      00027      JR      C,ciphok   ;falls gültige Deziffer
516F FE47      00028      CP      'G'        ;ungültige Ziffer?
5171 300E      00029      JR      NC,error   ;falls ja
5173 FE41      00030      CP      'A'        ;gültige Ziffer?
5175 380A      00031      JR      C,error    ;falls nein
5177 D607      00032      SUB     07h        ;an Deziffern angleichen
5179 E60F      00033 ciphok AND     0fh        ;ASCII -> binär
517B 4F          00034      LD      C,A        ;Akku retten
517C 07          00035      RLCA          ;unteres -> oberes Nibble
517D 07          00036      RLCA
517E 07          00037      RLCA
517F 07          00038      RLCA
5180 C9          00039      RET
          00040
5181 3E2F      00041 error LD      A,2fh     ;Code "falsche Parameter"
5183 B7          00042      OR      A          ;NZ-Bedingung
5184 C30944     00043      JP      4409h     ;traus mit Fehler
          00044
5147          00045      END     sys9

```

00000 Fehler

```

DRV 00 0608 1A96 1223 20DA 01EA FB50 10FB D1E1 .....#.....P....
0 10 C1F6 35C9 2752 554E 204F 4E4C 5927 2041 ..S.'RUN.ONLY'.A
OH 20 4242 5255 4348 0344 6F73 6665 686C 6572 BBRUCH.Dosfehler
30 2066 6174 616C 0321 2120 2052 4553 4554 .fatal!!!.RESET
DRS 40 206D 6974 2054 6173 7465 2052 210D B7C2 .mit.Taste.R!...
148950 994E 78E6 50FE 40CA 2D40 C9FE 6BC2 004D .Nk.P.s.-s.k..M
5D1H60 CDD5 4C20 042A 0344 E92B CD6C 5157 CD6C ..L.*.D.+1QW.1
70 5179 B257 CD6C 515F CD6C 5179 B36F 62E9 Gy.W.1Q_1Qy.ob.
80 D738 0AFE 4730 0EFE 4138 0AD6 07E6 0F4F .B..GO..AB.....Q
90 0707 0707 C93E 2FB7 C309 44b0 0000 0000 .....>/...D.....
A0 0000 0000 0000 0000 0000 0000 0000 0000 .....
B0 0000 0000 0000 0000 0000 0000 0000 0000 .....
FRS C0 0000 0000 0000 0000 0000 0000 0000 0000 .....
4 D0 0000 0000 0000 0000 0000 0000 0000 0000 .....
4H E0 0000 0000 0000 00CD 6000 C300 00F5 D3FA .....
F0 AFC3 0000 0202 4751 0000 0000 0000 0000 .....GQ.....

```

Korrektur zu "CALL um die Ecke"

Man sollte wirklich auch den selbstverständlichsten Kleinkram dreimal checken und ausprobieren, bevor man unschuldige Leser damit frustet. In meinem Programm, das in dem oben zitierten Artikel vorgestellt wird, steckt ein fataler Fehler. Wenn ein Unterprogramm in einer benachbarten Bank aufgerufen wird, muß unbedingt dafür Sorge getragen werden, daß sich der Stack im Common-Bereich befindet. Andernfalls würde nämlich das RET am Ende der Routine die Rückkehradresse in der eigenen Bank suchen - und fatalerweise sogar finden! Es fragt sich bloß, was für eine absonderliche Adresse das ist.

In der hier korrigierten Form läuft das Programm aber einwandfrei, denn der Stack wird nach Common verlagert. Der erste Teil, der unverändert geblieben ist, ist weggelassen. Da sich nun aber die Adressen hinter der Korrektur verschieben, ist der ganze Rest gelistet. Die veränderten Adressen für die in der Tabelle genannten Umschaltroutinen lauten jetzt so:

bank0 (die Bank 0 selektieren) 35D7h
ram (das Boot-EEPROM ausschalten) 35E8h
rom (das Boot-EEPROM selektieren) 35EFh

```
00017 ;verschiedene Speicherbereiche zugänglich machen
00018 bank1 PUSH AF ;retten
00019 IN A,(0f9h) ;Systembyte 0
00020 SET 6,A ;Bank 1 selektieren
00021 LD SP,0000h ;Stack restaurieren
00022 spbuf EQU #-2 ;Adreß-Operand variabel
00023 JR exit9 ;dort weiter
00024
00025 bank0 PUSH AF ;retten
00026 LD (spbuf),SP ;Stackpointer retten
00027 LD SP,37d0h ;eigener Stack im Common
00028 IN A,(0f9h) ;Systembyte 0
00029 AND 3fh ;Bank 0 selektieren
00030 DI ;INTs sperren
00031 exit9 OUT (0f9h),A ;Systembyte neu schreiben
00032 POP AF ;restaurieren
00033 RET
00034
00035 ram PUSH AF ;retten
00036 IN A,(0fah) ;Systembyte 1
00037 SET 2,A ;Boot-EEPROM ausschalten
00038 JR exita ;dort weiter
00039
00040 rom PUSH AF ;retten
00041 IN A,(0fah) ;Systembyte 1
00042 RES 2,A ;Boot-EEPROM selektieren
00043 DI ;INTs sperren
00044 exita OUT (0fah),A ;Systembyte neu schreiben
00045 POP AF ;restaurieren
00046 RET
```

Arnulf Sopp

CALL um die Ecke

Was ich hier vorstellen möchte, ist in dieser Form nur auf dem Genie III s oder vergleichbaren Computern lauffähig. Das Strickmuster, nach dem es aufgebaut ist, eignet sich jedoch für alle in unserem Club vertretenen Maschinen, überhaupt für alle Z80-Rechner. Es geht darum, Speicherbereiche zu erreichen, die ohne Umwege nicht verfügbar sind. Dazu gehören beispielsweise die Boot-EPRoMs der neueren Genies, aber auch Programme, die bei einfacheren Computern etwa mit Hilfe eines Bankers im Adreßbereich des ROMs, der Tastatur oder des Bildschirms laufen. Die hierfür notwendigen Umschaltsequenzen kann man sich schenken.

Vorbild waren dafür das Level-4-ROM von TCS (Genie I/II) und CP/M. Im Genie-Sonder-ROM kann ein String auf den Bildschirm ausgegeben werden, der sich unmittelbar hinter dem Ausgabebefehl befindet. So braucht man keinen Zeiger zu laden. CP/M ruft alles Mögliche mit einem CALL nach 0005h auf, je nach dem, was in den Registern übergeben wird. Ähnlich ist es hier:

```
CALL      call0      ;s. Listing
DEFW     4080h      ;Unterprogramm-Adresse
```

Mit dieser Befehlssequenz wird ein Unterprogramm an der Adresse 4080h in der normalerweise nicht zugänglichen Bank 0 des G3s aufgerufen. Um das tun zu können, müßte man sonst zuerst die Bank 0 selektieren, dann die Unterprogrammadresse anspringen, zuletzt wieder in die Arbeitsbank (normalerweise 1) zurückkehren. Aber bereits beim Umschalten auf Bank 0 würde sich ein solches Programm selber in die ewigen Jagdgründe bancken, es wäre bei RET nicht mehr auffindbar.

Die hier vorgestellte Methode arbeitet im Common-Bereich, der unabhängig von der eingestellten Bank immer selektiert ist. Von hier aus kann die Umschaltung risikolos vorgenommen werden. Die häufiger gebrauchten Umschaltroutinen liegen bereits als Unterprogramme vor, andere wurden (ab Label bank1) hier eigens angefügt. So ist es mit dem G3s nunmehr möglich, Unterprogramme in allen Speicherbereichen aufzurufen. Folgende Routinen besorgen die Selektion:

```
RAM "neben" den memory-mapped Devices: 06A0h
dieses RAM wieder ausblenden:          06ABh
RAM in Bank 0:                          35D4h (Label bank0)
wieder Bank 1 selektieren:              35CDh (Label bank1)
Boot-EPRoM selektieren:                 35E5h (Label rom)
Boot-EPRoM wieder ausknipsen:          35DEh (Label ram)
ein Unterprogramm in Bank 0 aufrufen:    35BBh (Label call0)
(Restoration des alten Zustandes nicht erforderlich)
```

Das EPROM oder der Speicher im memory-mapped-Adreßbereich sind mit einem gewöhnlichen CALL an die genannten Adressen erreichbar. Nur bei Bank 0 geht das aus den oben skizzierten Gründen nicht, daher ist die Routine call0 implementiert. Wenn sie mit einem CALL aufgerufen wird, steht die RET-Adresse im Stack. Es ist die Adresse unmittelbar hinter dem CALL. Nichts spricht dagegen, dort sonstwas unterzubringen, in unserem Fall die Adresse des gewünschten Unterprogramms. Hiervon geht die Routine call0 aus. Deshalb erhöht sie die RET-Adresse auf dem Stack um zwei Stellen, um die CALL-Adresse zu überspringen. Bei der Rückkehr aus dem Unterprogramm wird nun wieder sinnvoller Code vorgefunden.

Alle genannten Routinen, auch call0 und die zitierten System-Unterprogramme können gefahrlos aufgerufen werden, auch unter BASIC mitUSR. Der User sollte jedoch ziemlich genau wissen, was er da tut. So knipst er sich beispielsweise mit dem Programm an 06A0h (s. o.) die Tastatur und den Bildschirm aus. Da hilft nur noch abschalten. Daher muß der Anwender in seinen Programmen immer für eine Restoration des alten Zustands sor-

gen. Die zuständigen Routinen sind oben benannt. Auch der Interrupt-Status muß bedacht werden. Alle aufgezählten Routinen, die einen besonderen Speicherbereich selektieren, schalten die Interrupts ab. Unter denen, die den Normalzustand restaurieren, läßt nur 06ABh (s. o.) sie anschließend wieder zu. Der User muß deshalb selbst für den gewünschten INT-Status sorgen (unter BASIC mit CMD"T" und CMD"R").

Für andere Computer sind diese Adressen ungültig. Der Leser muß sich jeweils darüber informieren, wie es bei seiner eigenen Maschine aussieht. Dazu gehört natürlich auch die Suche nach einem freien Speicherplatz, an dem neu hinzukommende Routinen untergebracht werden können. Für die weitaus meisten von euch dürfte es der Adreßraum ab 3000h sein, der beim Genie I/II mit dem EG 64 MBA nutzbar gemacht werden kann.

Arnulf Sopp

```

35BB      00001      ORG      35bbh      ;hier freier Speicher
          00002
          00003 ;ein Unterprogramm in Bank 0 anspringen
35BB E3      00004 call0  EX      (SP),HL      ;HL retten, Stack laden
35BC D5      00005      PUSH     DE      ;retten
35BD 5E      00006      LD      E,(HL)   ;E <- (Stack)
35BE 23      00007      INC     HL      ;auf Stack + 1
35BF 56      00008      LD      D,(HL)   ;D <- (Stack + 1)
35C0 23      00009      INC     HL      ;auf RET-Adresse
35C1 ED53CB35 00010      LD      (calladr),DE ;CALL-Adresse patchen
35C5 D1      00011      POP     DE      ;DE restaurieren
35C6 E3      00012      EX      (SP),HL ;dto. HL, RET-Adr. -> St.
35C7 CDD435 00013      CALL   bank0    ;Bank 0 selektieren
35CA CD0000 00014      CALL   0000h    ;dort Unterprg. aufrufen
35CB      00015 calladr EQU    $-2      ;(Adresse variabel)
          00016
          00017 ;verschiedene Speicherbereiche zugänglich machen
35CD F5      00018 bank1  PUSH   AF      ;retten
35CE DBF9    00019      IN     A,(0f9h) ;Systembyte 0
35D0 CBF7    00020      SET   6,A      ;Bank 1 selektieren
35D2 1806    00021      JR    exit9    ;dort weiter
          00022
35D4 F5      00023 bank0  PUSH   AF      ;retten
35D5 DBF9    00024      IN     A,(0f9h) ;Systembyte 0
35D7 E63F    00025      AND   3fh     ;Bank 0 selektieren
35D9 F3      00026      DI     ;INTs sperren
35DA D3F9    00027 exit9  OUT    (0f9h),A ;Systembyte neu schreiben
35DC F1      00028      POP   AF      ;restaurieren
35DD C9      00029      RET
          00030
35DE F5      00031 ram    PUSH   AF      ;retten
35DF DBFA    00032      IN     A,(0fah) ;Systembyte 1
35E1 CBD7    00033      SET   2,A     ;Boot-EPROM ausschalten
35E3 1806    00034      JR    exita    ;dort weiter
          00035
35E5 F5      00036 rom    PUSH   AF      ;retten
35E6 DBFA    00037      IN     A,(0fah) ;Systembyte 1
35E8 CB97    00038      RES   2,A     ;Boot-EPROM selektieren
35EA F3      00039      DI     ;INTs sperren
35EB D3FA    00040 exita  OUT    (0fah),A ;Systembyte neu schreiben
35ED F1      00041      POP   AF      ;restaurieren
35EE C9      00042      RET
          00043
0000      00044      END

```

00000 Fehler

MNUM	NACHNAME	VORNAME	STRASSE	L	PLZ	ORT	USER-NAME	TELVOR	TELRUF
850264	Albertz	Dieter	Feldstrasse 2	D	2246	HENNSTEDT		04836	1414
830611	Bach	Siggi	Oster 17 A	D	4715	ASCHEBERG		02593	60210
850768	Berner	Horst	Hohewegstrasse 14	D	2940	WILHELMSHAVEN			
851077	Bernhardt	Helmut	Hafenstrasse 7	D	2305	HEIKENDORF		0431	241907
860612	Bertschi	Rudolf	Drusbergstraße 17	C	8053	ZÜRICH - SCHWEIZ -			
840120	Blaschek	Manfred	Inzersdorfer Str. 111/8/9	A	1100	WIEN - AUSTRIA -		0222	6400483
850643	Bochtler	Peter	Lautengasse 19	D	7900	ULM		0731	64406
840738	Bornschlegel	Hans	Königshofstraße 13	D	8605	HALLSTADT		0951	73831
840413	Brake	Thilo	Kastanienweg 26	D	2850	BREMERHAVEN		0471	64717
860207	Braun	Harald	Postfach 8011	D	2300	KIEL 17			
841117	Böckling	Ulrich	Am Sonnenhang 11	D	5414	VALLENDAR		0261	69522
840646	Dürhammer	Ulf	Eckenstrasse 8	D	4784	RÜTHEN 13		02954	786
850870	Engel	Jürgen	Lohstrasse 79	D	4300	ESSEN 11		0201	696902
840127	Fischbeck	Uwe	Friederikenstrasse 17	D	2940	WILHELMSHAVEN		04421	34282
840110	Folkerts	Ralf	Nutzhorner Straße 9	D	2875	BOOKHOLZBERG	RAFO	04223	2632
830816	Frey	Wolfgang	Berner Strasse 30	D	2000	HAMBURG 73			
851005	Geißler	Hans - Peter	Hochemmericherstr. 16	D	4100	DUISBURG 14		02135	52265
841005	Gerblinger	Dieter	Alpenkorpsstrasse 23	D	8102	MITTENWALD		08823	2017
860611	Goltz	Dietmar	Spardorferstraße 38	D	8520	ERLANGEN		09131	22169
830507	Grajewski	Werner	Zedernweg 29	D	4220	DINSLAKEN		02134	54573
830815	Grundmann	Waldemar	Beverbäkstraße 46	D	2900	OLDENBURG		0441	36218
841057	Hanss	Dirk	Blumenstrasse 1	D	7900	ULM		0731	23193
850161	Hemme	Wilhelm	Grunauer Strasse 168	D	8858	NEUBURG / DO.		08431	9468
850767	Hinze	Rolf	Bauxhof 4	D	5140	ERKELENZ		02431	4821
851182	Honcamp	Jochen	Schanzenweg 7	D	5900	SIEGEN		0271	76319
840544	Hose	Rüdiger	Wodanstrasse 7	D	8500	NÜRNBERG 40		0911	460012
850162	Jahn	Markus	Chr. v. Schmid Str. 28	D	8880	DILLINGEN/DONAU		09071	2453
850871	Jaschke	Siegfried	Karl - Hofbauer Str. 19	D	8481	ALTENSTADT/MN			
830419	Karnatz	Michael	Schweriner Ring 23	D	2940	WILHELMSHAVEN		04421	53936
860310	Kellermann	Reinhold	Haidfeldstraße 24	A	4020	LINZ / ÖSTERREICH		0732	831463
840234	Klein	Gerhard	Kempener Strasse 68	D	5000	KÖLN 60		0221	734809
831023	Kröher	Paul	Karpfenweg 6	D	2970	EMDEN		04921	27707
860209	Krüger	Karl - Herbert	Bruchweg 65	D	4920	LEMGO		05261	13686
840336	Kummerow	Jens	Hauptstrasse 4	D	2407	BAD SCHWARTAU		0451	21429
840748	Küster	Heinz - Gerd	Schwachtenbergweg 2	D	5603	WÜLFRAH	HEINZ GERD	02058	3037
851002	Langguth	Hans - Otto	Merkensstrasse 17	D	5000	KÖLN 30		0221	556643
851002	Liebig	Erich	Stollgasse 1 B	A	1070	WIEN		0222	9348433
851079	Linnenschmidt	Georg	Belfortstrasse 14	D	2800	BREMEN 1		0421	4988209
831224	Linneweber	Manfred	Auf der Brigg 15	D	2850	BREMERHAVEN		0471	25453
850625	Loose	Gerhard	Viefhaushof 42	D	4300	ESSEN 13	TRS 80	0201	212608
851076	Lutz	Eckhard	Walter-Delius-Str. 37 A	D	2850	BREMERHAVEN		0471	21270
850765	Mahlert	Herbert	Baumschulstrasse 7	D	4100	DUISBURG 14		02135	81462
860613	Mattern	Peter	Wesermünder Straße 100	D	2740	BREMERVÄRDE		04761	6579
830508	May	Holger	Marienstrasse 9	D	5768	SÜNDERN 2		02935	1668
840126	Meier	Hans-Christian	Raabeustrasse 42	D	2940	WILHELMSHAVEN		04421	64577
850766	Misioch	Waldemar	Adenauerring 25	D	8505	RÜTHENBACH		0911	506051
850163	Mühlenbein	Klaus - Jürgen	Am Mönchgarten 28	D	6940	WEINHEIM - LÜTZ.		06201	55052
851181	Nepp	Alfred	KobliceKgasse 4/10/1	A	1110	WIEN		0222	7642334
850614	Obermann	Hartmut	Schwalbacherstrasse 6	D	6209	HEIDENROD / KEMEL		06124	3913
840954	Offermann	Hartmut	Im Südkamp 2	D	5130	GEILENKIRCHEN 6		02462	3967
840339	Omasreiter	Irmgard	Am Römerfeld 39	D	8858	NEUBURG / DO.		08431	46519
860106	Posing	Carlo	Rue de Bastogne 184	L	9011	ETTELBRUCK		00352	81584
841056	Punzet	Alfred	Rosenweg 8	D	6120	MICHELSTADT		06061	3527
840129	Reichelsdorfer	Wolfgang	Herrenberg 25	D	8870	GÜNZBURG / REISEN		08221	32414

Mitgliedsliste des Genie / TRS-80 User Clubs 'Bremerhaven'. Stand: siehe Druckdatum. Anzahl Mitglieder: 086.

MNUM	NACHNAME	VORNAME	STRASSE	L	PLZ	ORT	USER-NAME	TELVOR	TELRUF
851232	Rubes	Karl	Nur über Betreuer erreichbar		0000	BAR		0711	594247
860614	Rychlik	Andreas	Königsberger Allee 120	D	4100	DUISBURG 1		0203	331383
850751	Sandkühler	Peter	Ortbergstrasse 10	D	4250	BOTTROP			
850203	Sanz	Alfonso	Santa Virgilia 16	E	2803	3 MADRID - SPANIEN -		7642373	
850521	Scharnhölz	Theodor	Postfach 1109	D	4534	RECKE 1		05453	1830
830509	Scheidt	Uwe von	Ströcker 45 C	D	2850	BREMERHAVEN		0471	85418
851075	Schiegl	Margit	Josef-Dabsch Str. 10/5/15	A	2102	BISAMBERG		02244	4395
850160	Schloeske	Holger	Dr. Fabri Strasse 19	D	8859	BURGHEIM / STRASS		08432	1847
850745	Schmid	Alexander	St. Cajetan Str. 38 / VII	D	8000	MÜNCHEN 80		089	495326
830302	Schmidt	Horst	Körnerstraße 7	D	2850	BREMERHAVEN		0471	414611
840235	Schmitz	Paul - Jürgen	Lübecker Straße 6	D	6236	ESCHBORN			
850633	Schopen	Peter	Rosstraße 10	D	4000	DÜSSELDORF 30		0211	581518
841158	Schäfer	Walter	Rathausstrasse 4	D	8160	MIESBACH		08025	1631
841259	Seelmann-Eggeber	Jörg	Henri Spaak Strasse 96	D	5305	ALFTER		0228	643853
850972	Seibold	Joachim	Eichenweg 41	D	7121	LÜCHGAU		07143	23595
850630	Sikora	Ernst	Von Hessen Strasse 18	D	5040	BRÜHL		02232	227
840131	Sopp	Annulf	Wakenitzstrasse 8	D	2400	LÜBECK	HACKTORY	0451	791
830401	Spieß	Peter	Trugenhofener Straße 27	D	8859	RENNERTSHOFEN 1		08434	454
840340	Stark	Othmar	Schillerstrasse 112	A	2340	MÖDLING - AUSTRIA -		02236	811805
860715	Stephan	Hans - Martin	Am Glasesch 9 A	D	4506	HAGEN A.T.W.		05401	99585
840128	Thalmeier	Gregor	Postfach 1140	D	8011	KIRCHSEEN		08091	9085
860208	Thum	Helmut	Unterstr. 2	D	5483	BAD NEUENAHN-ADRHEI.			
830306	Thönnißen	Heinrich	Steinhäuser Straße 17	D	2800	BREMEN 1		0421	14927
840749	Topp	Gerhard	Heininger Weg 1	D	3342	WERLABURGDORF		05335	240
860816	Vanselow	Peter	Sagekühle 26	D	2150	BUXTEHUDE		04161	52721
850869	Wagner	Günther	Gartenstrasse 4	D	8201	NEUBEUERN		08035	3361
860105	Wala	Erwin	Sulz 191/7	A	2392	WIENERWALD -AUSTRIA-		02238	83702
850418	Weidmann	Josef	Augsburger Strasse 32	D	8858	ND - FELDKIRCHEN		08431	8471
850973	Weikamp	Horst	Fontane Strasse 77	D	4290	BOCHOLT		02871	12835
850604	Weiss	Dieter	Buerglestrasse 3	D	7209	MEHINGEN		07426	7194
840750	Wittmann	Reinhard	Klausenbrunnenweg 32	D	8852	RAIN / LECH		09002	2381
840852	Wolf	Klaus	Niddastrasse 15	D	6457	MAINTAL 1		06181	493450

Im September
haben

Oktober
Geburtstag:

Horst Schmidt
Heinrich Thönnißen
Werner Grajeweski
Holger May
Annulf Sopp
Heinz - Gerd Küster
Hartmut Offermann
Peter Schopen
Jürgen Engel

Waldemar Grundmann
Wolfgang Frey
Othmar Stark
Thilo Brake
Wilhelm Hemme
Theodor Scharnhölz
Horst Weikamp
Karl Rubes

Meinen herzlichsten Glückwunsch.

H A R D W A R E - H a r d w a r e

=====

Guten Tag. Wir sind offizielle Genie und Tandy Händler, daneben vertreiben wir die Produkte der Firma Plantron. Außer Verkauf übernimmt unser Haus auch Wartung, Reperatur, Einweisung, Installation für die von uns gelieferten Geräte. Mit unserem **Service** sind Sie immer eine Nase voraus.

C O M P U T E R

=====

Nachfolgend eine Übersicht über unser Lieferprogramm und unsere Preise. Technische Details entnehmen Sie bitte den Prospekten. Weitere Informationen liefern wir Ihnen auf Anfrage gerne.

Teil 1: 8 - BIT Rechner

- | | |
|---|-------------|
| 1. Genie III s mit Monitor, 256 KRAM, 7.2 MHz, 1.4 MBYTE Disk | DM 5550.00 |
| Ein sehr schnelle Gerät der Profiklasse. Prädestiniert als Geschäftsgerät. | |
| 2. Tandy 4 unser Standard 8 BIT System mit zwei Disk und Monitor | DM 1755.00 |
| 3. Tandy 4p unser transportabler Künstler. Ausstattung wie vor | DM 1755.00 |
| Beide Tandy Rechner sind äußerst günstige Geräte für den Einstieg. Im Preis enthalten ist eine Dateiverwaltung. Das Gerät ist aufrüstbar auf 1 MByte und 8 MHz sowie High Resolution Grafik und damit zu den Geräten der Spitzenklasse zu zählen. | |
| 4. Tandy 100 und Tandy 200 die transportablen Handheldcomputer | auf Anfrage |

Teil 2: MSDOS Rechner

- | | |
|---|-------------|
| 0. Genie 16 Low Cost , 1 Disk, 256 KRAM, Grafikkarte | DM 1595.00 |
| 1. Genie 16 C 16-BIT Rechner, IBM kompatibel, 640 KB
2 Disk a 360 KBYTE, Farbgrafikkarte(200x640), MS/DOS 2.11 | DM 3975.00 |
| 2. Genie 16 XC wie vor, 1 Disk, 20 MByte Festplatte | DM 5950.00 |
| 3. Genie 16 TC mit 1 MB-RAM on Board.4,7/8 MHz umschaltbar | DM 4485.00 |
| 4. Genie 16 TXC Gerät wie vor, mit 20 MByte HD, 1 Laufwerk | DM 6455.00 |
| 5. TANDY 1000, 128 K RAM, 1 Disk, Monochrom Monitor 12", MSDOS 2.11
GW-Basic, DeskMate | DM 3485.00 |
| 6. Tandy 1000, wie vor, 384 KRAM, 2 Disk | DM 4255.00 |
| 7. Tandy 1000 HD, 384 KRAM, 1Disk, 10MB HD, sonst wie vor | DM 6455.00 |
| 8. Tandy 1000 HD, 640 KRAM, 20 MB HD, soonst wie vor | DM 7455.00 |
| 9. Tandy 2000 , 256 K, 2 Disk a 800 KB | auf Anfrage |

- | | |
|---|-------------|
| 10. Tandy 2000 HD, 256 K, 1 Disk, 10 MB HD | auf Anfrage |
| 11. PT 16-LC, 256 KRAM, Colour Grafikkarte, 2Disk | DM 1998.00 |
| 12. PT-16 XT,Turbo 4,7/8 MHz, 256 KRAM, Multi I/O, 2 Disk | DM 2498.00 |
| 13. PT-16 XT/20, wie vor, 1 Disk, 20 MB HD | DM 4955.00 |

A T - Rechner

- | | |
|--|-------------|
| 1. Genie 286 AT voll AT kompatibel, mit 20 MB Harddisk
MS-DOS 3.1, 1MBRAM,Uhr, Better Basic | DM 9955.00 |
| 2. wie vor mit 30 MB HD | DM 10955.00 |
| 3. wie vor mit 40 MB HD | DM 11955.00 |
| Genie Terminal mindestens 2 Rechner, bis zu 10 Rechner im Netz | |
| 4. Tandy 3000, 512 KBRAM, 2 1.2MB Disk, Monitor, DeskMate | DM 6995.00 |
| 5. Tandy 3000 HD, 512 KB RAM, 1 Disk, 20 MB HD, Monitor | DM 8755.00 |
| 6. Tandy 3000 , wie 4.) mit Farbmonitor | DM 8755.00 |
| 7. Tandy 3000 HD, wie 5.) mit Farbmonitor | DM 10485.00 |
| 8. PT-16 AT, 640 KRAM, 1 Disk 1.2 MB | DM 5975.00 |
| 9. PT-16 AT/20, wie vor, mit 20 MB HD | DM 7955.00 |

Zu allen Rechnern bieten wir umfangreiches Zubehör an Soft- und Hardware. Zum Beispiel Drucker von Oki Data, Sakata, Citizen, Brother, Honeywell etc. Zum Beispiel Streamer für Festplatten. Zum Beispiel Monitore. Zum Beispiel Erweiterungskaten für alle PC's. Oder auch Papier, Farbbänder, Plotter, sowie Programme aller namhafter Software Häuser. Da die Preise sich oft schnell ändern, lohnt sich immer ein Anruf. Wir informieren Sie dann über den aktuellen Stand der Preise. Wir können in der Regel kurzfristig liefern. Alle Preise enthalten 14 % MWST

