

Rolf-Dieter Klein

# Das mc-Grafik-Terminal

## Teil 1: Hardware

Bisher war es nur wenigen möglich, von hochauflösender Grafik Gebrauch zu machen: Entweder ist sie umständlich zu bedienen oder zu teuer. Mit dem mc-Grafik-Terminal soll das anders werden. Das Terminal besteht aus einer Europakarte, die neben einem Z80-Prozessor auch einen Grafikprozessor beinhaltet.

Die Software zur Bedienung des Terminals ist knapp 8 KByte groß und ermöglicht neben einer Teilemulation des TVI-950 (wordstarfähige Teilmenge) die Emulation des T4014 (Tektronix-kompatibel) und einen eigenen Grafik-Modus mit vielen einfach zu bedienenden Befehlen der mc-Grafik-Sprache. Die Auflösung des Grafikbildschirms ist  $512 \times 256$  Punkte, die bei Auswahl eines anderen Grafikprozessors auch  $512 \times 512$  Punkte sein kann. Bei  $512 \times 256$  Punkten stehen vier Bildschirmenebenen zur Verfügung, die von der Software zum Scrollen gebraucht werden oder im Grafikmodus für bewegte Grafik verwendet werden können.

Bild 1 zeigt die Anschlüsse zur Außenwelt. Das Terminal besitzt zwei serielle

Schnittstellen, die eine ist gedacht zum Anschluß des Computers, die andere für den Anschluß einer Tastatur. Beide sind bidirektional, da die Tastatur auch programmiert werden soll. Am Ausgang steht ein BAS-Signal zur Verfügung, das direkt an einen Standard-Monitor angeschlossen werden kann. Ferner sind die Ausgänge HSYNC, VSYNC und Video für Spezialanwendungen und für die Farberweiterung vorhanden.

### Zwei Prozessoren

Bild 2 zeigt das Blockschaltbild der Platine. Ein Z80A übernimmt die Steuerung des Terminals. 8 KByte EPROM sind für die Steuersoftware vorhanden, die im nächsten Heft vorgestellt wird. 4 KByte

RAM werden für einen internen Bildwiederholtspeicher und einen Ringpuffer für die Zeichenebene gebraucht, wobei etwa  $\frac{1}{2}$  KByte frei bleibt für Anwendersoftware, die auch ladbar ist. Mit den neuen  $8 \text{ K} \times 8$ -Bit-RAMs, die leider noch sehr teuer sind, kann der Speicher auf 16 KBytes erweitert werden, und dann läßt sich dort auch ein programmierbarer Zeichengenerator unterbringen. Doch vorerst genügen die 4 KByte. Zwei STI-Bausteine (MK 3801), die eine serielle Schnittstelle, Timer, Interruptcontroller und Parallelports beinhalten, ermöglichen die Steuerung und Kommunikation. Als Grafikprozessor wurde der EF-9366 der Firma Thomson gewählt, da er der einzige ist, mit dem sich auf einer Europakarte ein komplettes Grafikterminal unterbringen läßt und der außerdem den schnellsten Vektorgenerator (1 Mio. Bildpunkte/s) besitzt. Dem Grafikprozessor steht ein vom Hauptspeicher getrennter Bildwiederholtspeicher von 64 KByte zur Verfügung. Beim EF-9366 ist er in vier Bildebenen von je  $512 \times 256$  Punkte aufgeteilt. Es wird immer nur eine Bildebene ausgelesen, aber in den anderen kann gleichzeitig geschrieben werden, wodurch ein unsichtbarer Bildaufbau möglich ist. Die Auswahl der Bildebenen erfolgt über Steuerbits der STI-A. Die STI-B kann auch entfallen, wenn keine Tastatur benötigt wird und das Terminal als Peripheriegerät betrieben wird: Die Software merkt dies automatisch und die Baudrate ist dann fest auf 9600 Baud eingestellt. Der EF-9365 erlaubt die Darstellung von  $512 \times 512$  Punkten, dieser

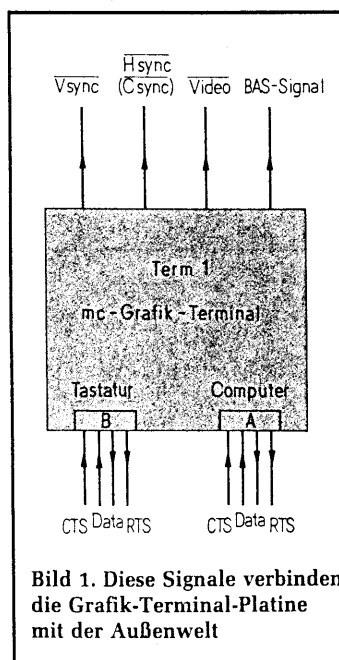


Bild 1. Diese Signale verbinden die Grafik-Terminal-Platine mit der Außenwelt

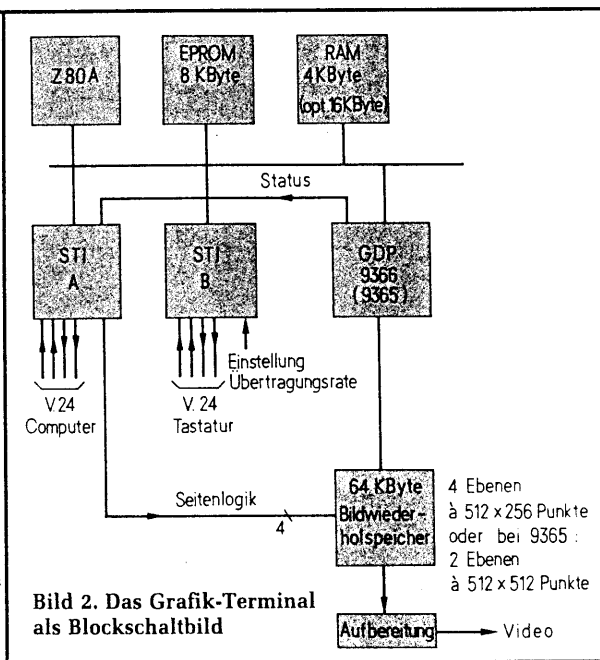


Bild 2. Das Grafik-Terminal als Blockschaltbild

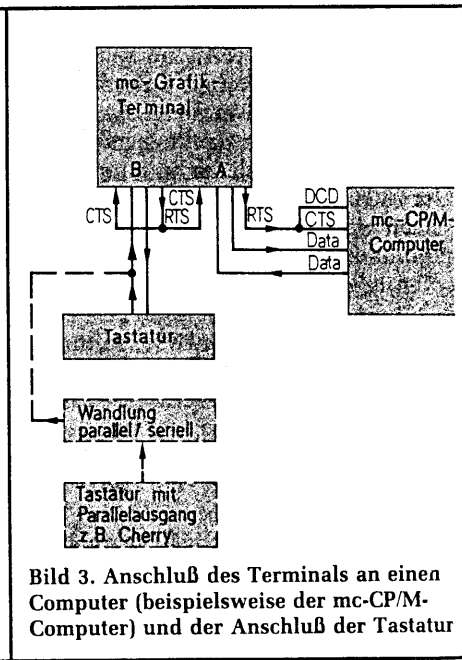


Bild 3. Anschluß des Terminals an einen Computer (beispielsweise der mc-CP/M-Computer) und der Anschluß der Tastatur

Grafikprozessor kann nach Umstellen einer Brücke verwendet werden. Die Software ist aber für den 9366 zugeschnitten, da sich dort wegen des fehlenden Zeilensprungverfahrens ein flimmerfreies Bild ergibt, was für die Textdarstellung (z. B. Wordstar) wichtig ist. Daher ist die Version mit dem EF-9366 die Standard-Version.

Bild 3 zeigt den Anschluß, beispielsweise an den mc-CP/M-Computer, für den das Terminal auch gedacht ist, wobei es natürlich auch mit jedem beliebigen Computer zusammenarbeiten kann.

### Verschiedene Konfigurationen

Das Terminal wird zum einen über die Datenleitungen mit dem Computer verbunden, zum anderen gibt es eine zusätzliche Leitung mit dem Namen RTS. Sie dient dazu, den Computer zu bremsen, wenn der Datenfluß zu hoch ist, da wegen der Grafikeigenschaften des Terminals ein ununterbrochener Datenstrom nicht immer möglich ist. Beim Anschluß an den mc-CP/M-Computer muß in der Monitorsoftware die SIO dafür programmiert werden, dies geschieht durch Ändern des Statuswortes auf Auto-enable (im Schreibregister 3 muß der Wert E1 anstelle des Wertes C1 programmiert werden). DCD und CTS geben die Übertragung nur dann frei, wenn die entsprechenden V.24-Leitungen auf +12 V liegen (am Chip 0 V). Die Baudrate ist normalerweise 9600 Baud, jedoch kann auch 28 800 als Spezialrate am Terminal eingestellt werden, nicht jedoch 19 200. Natürlich lassen sich alle darunterliegenden Normbaudraten einstellen.

Als Tastatur kann jede serielle Tastatur verwendet werden, z. B. die dafür entwickelte mc-Tastatur mit eigenem Z80, die demnächst vorgestellt wird. Soll eine parallele Tastatur verwendet werden,

so kann eine Zusatzschaltung verwendet werden [2].

Bild 4 zeigt eine andere Konfiguration. Das mc-Grafik-Terminal kann auch als Peripheriegerät eingesetzt werden. Es wird dann von CP/M aus wie ein Drucker gesehen bzw. wie ein Plotter. Bild 5 schließlich zeigt die Möglichkeit, das System auf Farbe zu erweitern. Dazu wird einfach eine in der Anzahl von Ebenen gewünschte Zahl von Platinen verwendet, die mittels einer einfachen Synchronisationsschaltung gekoppelt werden. In der Software gibt es dann Befehle, mit denen einzelne Ebenen angesprochen werden können, wenn nur eine serielle Schnittstelle zum Computer existiert. Im anderen Fall kann die Leistung dieser Anordnung voll ausgenutzt werden, da sich alle CPUs am Bildaufbau beteiligen können.

### Die Hardware

Nun zur Schaltung. Bild 6 zeigt die Schaltung des CPU-Teils, Bild 7 die Schaltung des Grafikprozessor-Teils. Der Z80 wird mit einem 3,6864-MHz-Takt betrieben, der gleichzeitig zur Baudratenerzeugung verwendet wird. Die Reset-Schaltung ist mit einem Timer 555 gelöst, um optimale Zuverlässigkeit des Systems zu erreichen. Die Speicherbereiche sind wie folgt aufgeteilt: EPROM von Adresse 0 bis 1FFFH mit einem 2764. Wenn nicht die mc-Software verwendet wird, kann auch ein 2732 verwendet werden, beispiels-

weise bei einer Anwendung als Single-Board-Computer mit Grafikausgabe. Als RAMs werden die Bausteine 6116 verwendet; hier ist darauf zu achten, daß es auf dem Markt ähnliche Bausteine 2 K x 8 gibt, die sich wegen unterschiedlicher Bedeutung von einzelnen Pins nicht einsetzen lassen. Die Zugriffszeit sollte besser als 250 ns sein.

Der Speicherbereich geht von 3800H bis 47FFH, wenn man beide Sockel zusammen nimmt. Da die Decodierschaltung nicht eindeutig decodiert, treten die Bereiche auch auf anderen Adressen auf. Bei der Verwendung von 8 K x 8 RAMs, ergibt sich ein Bereich von 2000H bis 5FFFH. Die Software unterstützt zur Zeit aber nur den 4-KByte-Speicherbereich. Bei der I/O-Decodierung wurden einige Standard-Bereiche verwendet, die auf den ersten Blick nicht ganz einsichtig sind. Der GDP liegt auf der Adresse 70H bis 7FH, da es für diesen Bereich schon einige Software gibt. Die STI A auf 80H bis 8FH und STI B von 90H bis 9FH. Dabei ist dieser Bereich so gewählt, daß er mit Adressen des mc-CP/M-Computers nicht kollidiert, da durch Verbinden der CPU-Anschlüsse mit dem ECB-Bus eine einfache Emulation möglich ist, wobei mit einer Zusatzschaltung nur der Adreßbereich 70H bis 9FH durchgelassen werden darf.

Der Decoder 74139 erhält nicht alle Adressen als Auswahl, weshalb die STI, der GDP und der Speicherbereich mehrmals im Adreßraum auftreten. Die V.24-Anpassung geschieht mit den

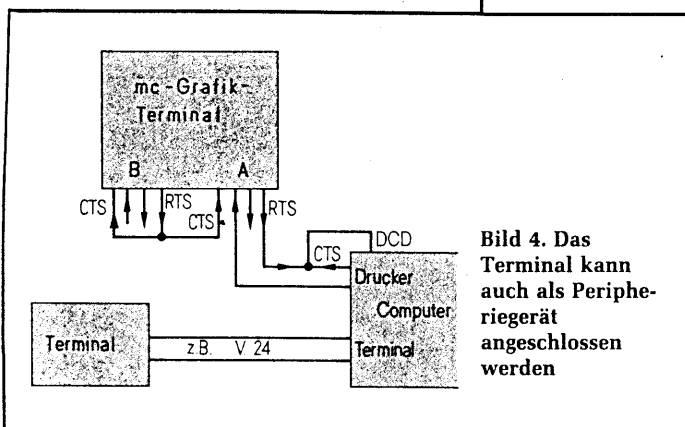


Bild 4. Das Terminal kann auch als Peripheriegerät angeschlossen werden

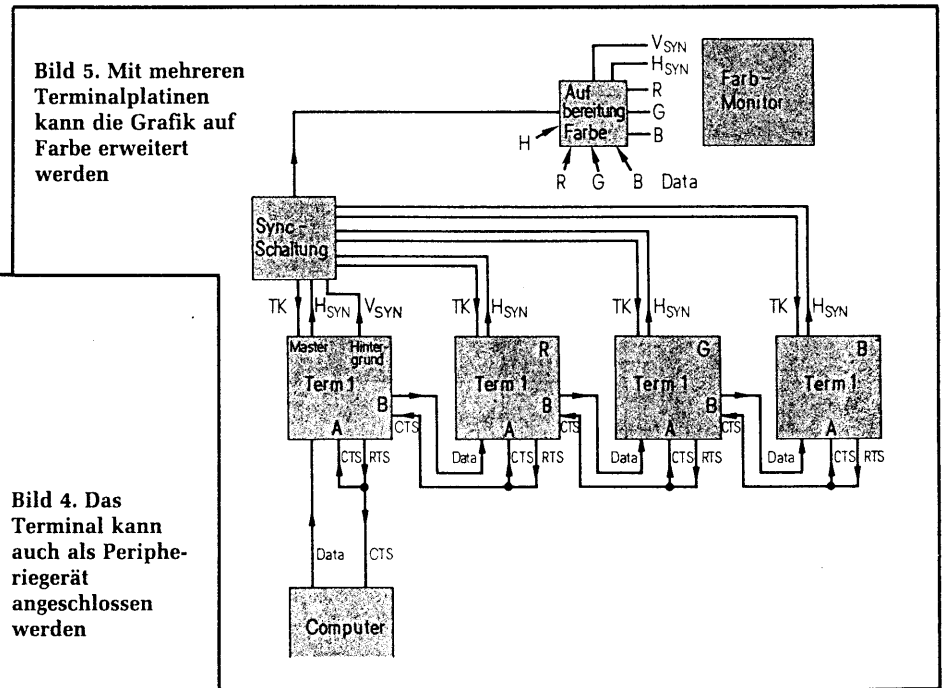


Bild 5. Mit mehreren Terminalplatinen kann die Grafik auf Farbe erweitert werden

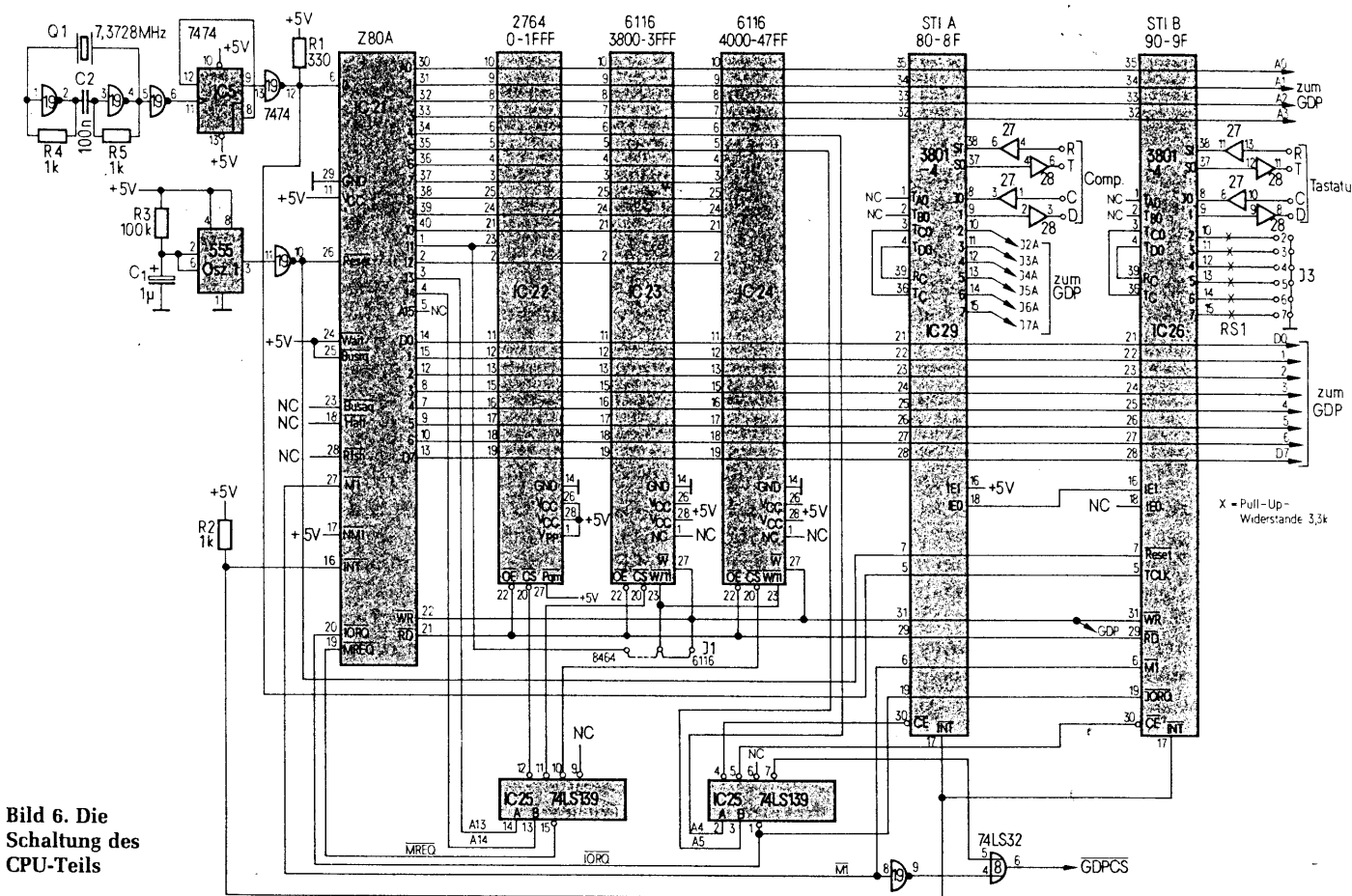


Bild 6. Die Schaltung des CPU-Teils

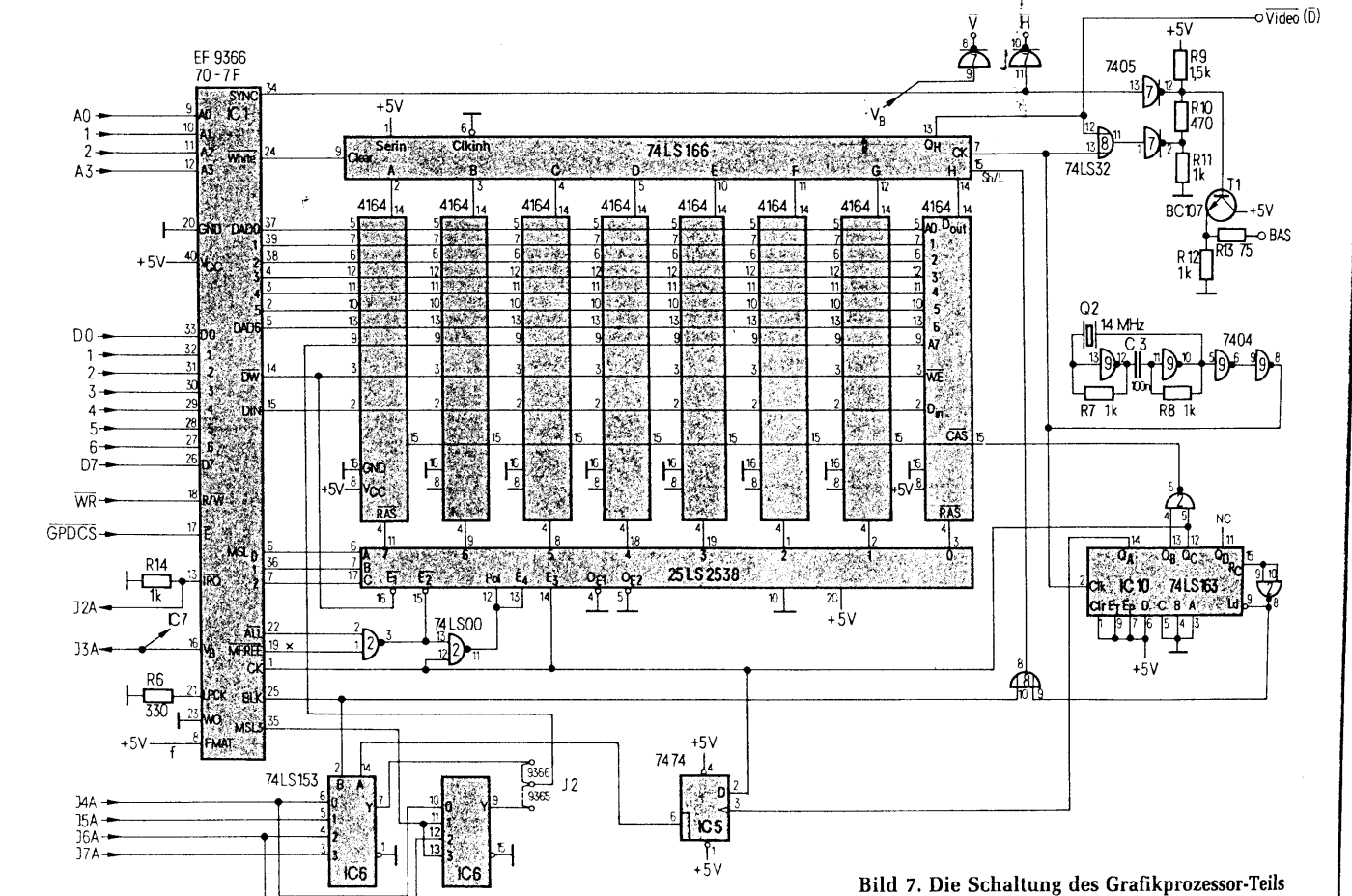
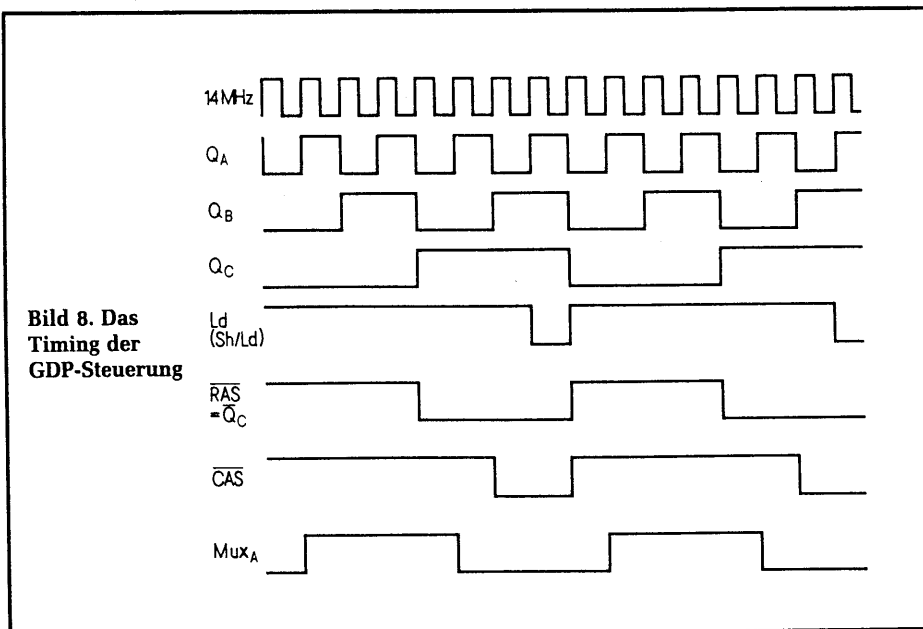


Bild 7. Die Schaltung des Grafikprozessor-Teils

Treibern 75188 und 75189, wobei die Steuerleitungen CTS und RTS per Software und über Ports realisiert werden.

### Ein komplexer Grafikprozessor

Der Grafikprozessor erhält einen eigenen Taktoszillator von 14 MHz, der alle wichtigen Signale erzeugt. Über einen Teiler 74163 werden die verschiedenen Takte erzeugt. Bild 8 zeigt alle wichtigen Signale. Der Multiplexer 74153 dient der Seitenauswahl. Für den EF-9366 ist der linke Schaltungsteil in Bild 7 verantwortlich. Eine 4-Bit-Seitenadresse gelangt an den Multiplexer. Die Eingänge 0, 1 dienen der Auswahl einer Lese-Seite, die Eingänge 2 und 3 zur Auswahl für die Seite, auf die bei einem Schreibzugriff geschrieben wird. Der Multiplexer erhält dazu die Information, ob ein Lese- oder Schreibzugriff vorgenommen wird, über das Signal BLK. Der Ausgang DW, der ja direkt ein Schreiben vorgibt, ist leider zu zeitkritisch, um dafür Verwendung zu finden. BLK gibt an, ob ein Bild dargestellt wird oder nicht, da es das Bildfenster bestimmt. Wird geschrieben, so erfolgt es nicht während des Auslesens, sondern an einer unsichtbaren Stelle. Daher ist beim Schreiben das Signal auf High und beim Auslesen auf Low. Nun gibt es aber genau einen Fall, bei dem es nicht so ist, nämlich beim schnellen Bildschirmlöschern mit einem speziellen GDP-Befehl. Dort wird während des Auslesens geschrieben, und unsere Schaltung versagt, da



**Bild 8. Das Timing der GDP-Steuerung**

sie nicht eine unsichtbare Seite löschen kann. Dies wäre aber auch beim DW-Signal der Fall. Die mc-Terminal-Software berücksichtigt aber alle diese Fälle, und nur wenn direkt auf den GDP zugegriffen wird, was über einen Grafik-Befehl möglich ist, muß darauf geachtet werden, daß beim Bildlöschern zunächst der GDP auf den dunklen Schirm gestellt und dann der Befehl ausgeführt wird.

Der Baustein 25LS2538 übernimmt die Bausteinauswahl; über den Eingang POL können alle Ausgänge invertiert werden,

um z. B. alle Bildpunkte beim Auslesen selektieren zu können.

Die Ausgabe der Bildpunkte erfolgt über das Schieberegister 74166, das nur bei dem sichtbaren Teil (BLK-Signal) mit Daten geladen wird. Am Ausgang QH erscheint die serielle Bildinformation, wobei das Signal High ein dunkles Bildfeld bedeutet. Das Videosignal wird anschließend mit den Synchronsignalen gemischt und ein BAS-Signal erzeugt. Der Widerstand R11 kann entfallen, wenn eine höhere Ausgangsspannung beim BAS-Signal gewünscht wird.

**Tabelle: Liste der verwendeten Bauteile**

Pos.	Typ/Wert	Bemerkung	Pos.	Typ/Wert	Bemerkung
IC 1	EF-9366 (9365 opt.) 74LS00	GDP, Thomson	IC 25	74LS139	
IC 2	74LS00		IC 26, 29	MK3801	STI, Mostek
IC 3	AM 25LS2538	Decoder, AMD	IC 27	75189	V.24-Empfänger
IC 4	74LS166		IC 28	75188	V.24-Sender
IC 5	7474		Q 1	7,3728 MHz	Quarz
IC 6	74LS153		Q 2	14 MHz	Quarz
IC 7	7405		T 1	BC 107	
IC 8	74LS32		C 1	1 µF/10 V	Tantal
IC 9, 19	7404		C 2, 3	100 nF	
IC 10	74LS163		CV 1, 4	100 nF	
IC 11...18	4164	200 ns, NEC/Hitachi	CV 2, 3, 5	10 µF/10 V	Tantal
IC 20	555	Timer	R 1, 6	330 Ω/0,1 W	
IC 21	Z80A	CPU	R 2, 4, 5, 7, 8	1 kΩ/0,1 W	
IC 22	2764	EPROM mit Software	R 3	100 kΩ/0,1 W	
IC 23, 24	6116	CMOS-RAM	R 11, 12, 14	1 kΩ/0,1 W	
			R 9	1,5 kΩ/0,1 W	
			R 10	470 Ω/0,1 W	
			R 13	75 Ω/0,25 W	

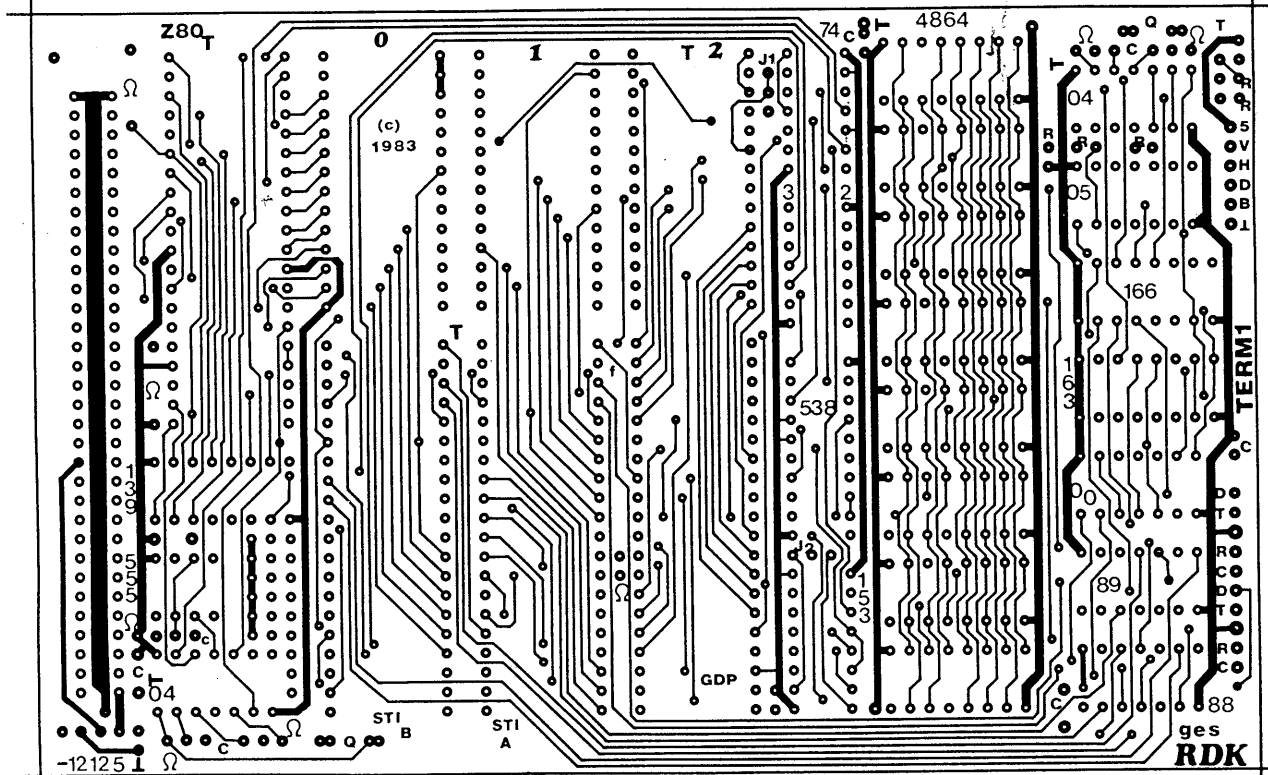
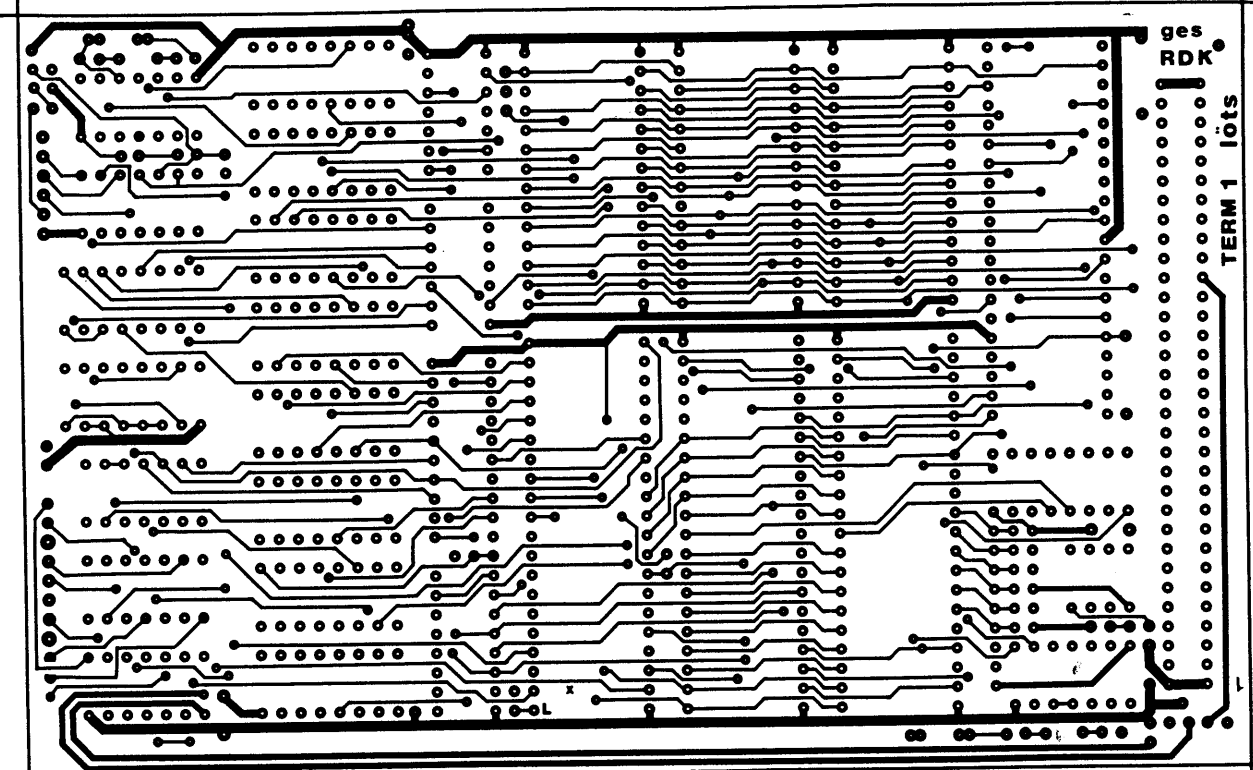


Bild 9. Die Schaltung befindet sich auf einer zweiseitigen Leiterplatte, die Lötseite oben und die Bestückungsseite unten

## Aufbau und Test

Bild 9 zeigt das Layout der zweiseitigen Platine; Bild 10 den Bestückungsplan. Die Platine besitzt auf der einen Seite eine VG-Leiste, von der nur die ECB-Versorgungsspannungen +5 V, +12 V und -12 V verwendet werden. Damit kann die Platine direkt in den mc-CP/M-Computer eingeschoben werden. Eine separate Kontaktleiste ermöglicht aber auch den Anschluß der Versorgungsspannungen über einen kleinen Stecker, wenn das Terminal als selbständiges Gerät aufgebaut wird.

Alle Brücken sind auf der Platine schon voreingestellt und müssen nur bei Änderungen der Standard-Version verändert werden. J1 dient der Einstellung bei anderen RAMs (8 K × 8 Bit), J2 für den Einsatz des 9366 und J3 zur Einstellung der Baudrate, wobei die Übertragungsrate 9600 Baud beträgt, wenn alle Pins offen sind. RS 1 ist ein Widerstandsarray, wahlweise können aber auch Einzelwiderstände eingesetzt werden. Alle Kondensatoren mit der Bezeichnung CV.. sind Entstörkondensatoren und tauchen daher im Schaltplan nicht auf. Beim Aufbau sollten ausschließlich Sockel verwendet werden, was die Fehlersuche erheblich erleichtert. Nach dem Einschalten des Terminals muß der Bildschirm gelöscht sein und links oben ein Cursor blinken. Ist dies der Fall, so läuft der größte Teil der Platine. Wird der Bildschirm nicht gelöscht und erscheint kein Cursor, ist aber eine Umrandung beim Monitor sichtbar, so liegt ein Fehler im CPU-Teil, andernfalls umgekehrt. Es empfiehlt sich, das fertige EPROM im Handel (GES) zu kaufen und es nicht selbst abzutippen, obwohl im nächsten Teil das komplette Listing erscheint. Es würden sich dabei nur sehr viele Fehler einschleichen, gerade bei einem 8-KByte-Programm. Wird der mc-CP/M-Computer angeschlossen, so muß nach dem Reset die Meldung des Computers auf dem Bildschirm erscheinen. Zeigt der Bildschirm nur wirre Zeichen, so ist die Baudraten-Einstellung zu überprüfen. Bei der Eingabe von Zeichen über die Tastatur müssen diese am Computer ankommen, ist dies nicht der Fall, so ist die Spannung an den Leitungen CTS zu prüfen, die im Normalfall auf +12 V liegen muß. Verschwinden Zeichen bei der Ausgabe von längerem Text (mit Scrollen), so ist die CTS-Leitung am Eingang des Computers (und/oder DCD) zu prüfen, die die Ausgabe anhalten muß, wenn sie auf -12 V liegt. Der Puffer des Terminals beträgt 1024 Zeichen und die Ausgabe wird kurz vor Überlauf gesperrt.

Ansonsten sind die Oszillatoren genau zu testen und auch die Reset-Schaltung. Die verwendeten Bauteile sind in der Tabelle aufgeführt.

## Optionen und Zusatzschaltungen

Für den Anschluß von Tastaturen mit Parallelausgang kann eine Schaltung zur

Parallel/Seriell-Wandlung verwendet werden [2].

In Bild 11 ist die Synchronisationsschaltung abgebildet, wie sie für jede zusätzliche Farbebene gebraucht wird. Ein Terminal wird als Master erklärt, darauf werden alle anderen Platinen synchronisiert. Es wird das CSYNC-Signal als Ver-

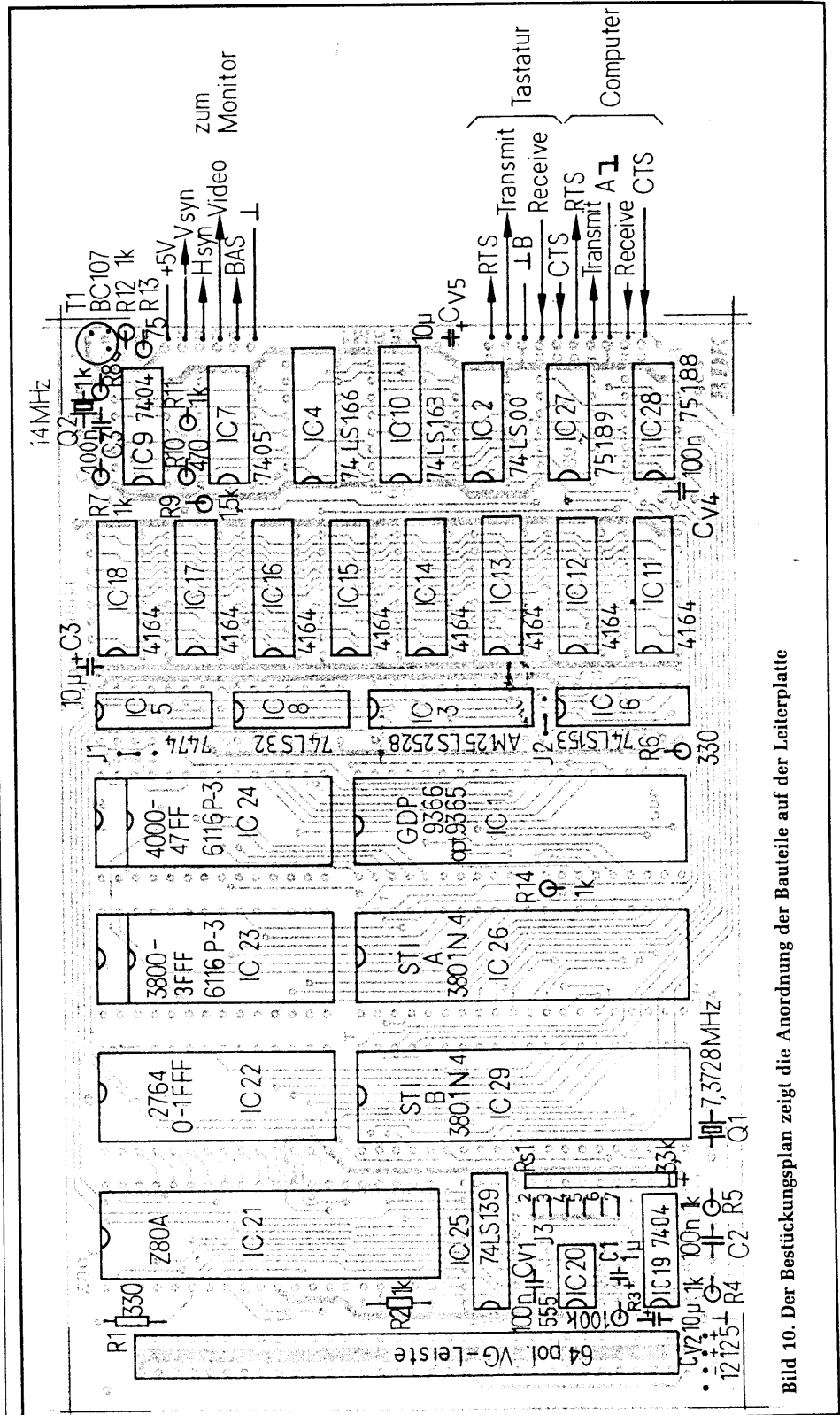


Bild 10. Der Bestückungsplan zeigt die Anordnung der Bauteile auf der Leiterplatte

gleich verwendet, da es HSYNC- und VSYNC-Informationen enthält. Das Signal ist in der Schaltung auch als HSYNC bezeichnet. Alle Slave-Platinen werden ohne Taktoszillator (14 MHz) aufgebaut, wobei es genügt, den 7404 (IC 9) herauszulassen. An Pin 8 des nun nicht mehr vorhandenen ICs wird dann ein neuer Takt angeschlossen, der von der Zusatzschaltung kommt. Stimmt CSYNC des Masters nicht mit CSYNC des Slave überein, so wird ein Taktpuls beim Slave ausgelassen, dadurch verschiebt sich das Timing. Nach einer Weile (einige Sekunden) sind beide CSYNC-Signale identisch und der Takt ist von da an gleich.

Dieses Verfahren hat sich gut bewährt und ist einfach zu realisieren. Beim Nachbau ist darauf zu achten, daß 14 MHz schon sehr kritisch sind und kurze Leitungen wie auch gute Entblockung Voraussetzung für das Funktionieren der Schaltung sind. Es empfiehlt sich auch, den Mastertakt nochmals getrennt auf der Zusatzplatine zu erzeugen und dem eigentlichen Master über ein Oder-Gatter zuzuführen, so daß alle Takte der Platinen genau synchron anfangen. Das ist aber nicht Voraussetzung der Schaltung, da Master und Slave nur auf einen Taktzyklus genau synchronisiert werden. Sind die Verzögerungszeiten aber gleich, so sind die beiden Teilbilder absolut deckungsgleich.

Am Schluß sei noch darauf hingewiesen, daß die Originalplatine (TERM 1) sowie das EPROM nur bei der Firma GES/Graf (Postfach 1610, 8960 Kempten) zu erhalten sind. Alle Original-Platinen werden mit Lötstopplack und alle Bauteile mit hochwertigen IC-Sockeln geliefert. (Fortsetzung folgt)

#### Literatur

- [1] Klein, Rolf-Dieter: CRT-Controller unterstützt Grafikfunktionen. ELEKTRONIK 1981, Heft 8, Seite 63.
- [2] Klein, Rolf-Dieter: Umwandlung. In diesem Heft.

## mc-Meinungsumfrage

Im Auftrag von mc führt derzeit die Dr.-Richter-Forschungsgruppe eine Umfrage durch. Zu diesem Zweck wurden Fragebogen an bestimmten ausgewählten Verkaufsstellen im Bundesgebiet dem mc-Heft Nr. 6/1983 beigelegt beziehungsweise an einige repräsentativ ausgewählte Abonnenten versandt.

Sollten Sie zu jenen Lesern gehören, die einen solchen Bogen erhalten, aber ihn noch nicht zurückgesandt haben, so ersuchen wir Sie hiermit nochmals, uns auf diesem Wege Ihre Meinung zukommen zu lassen. Denn Sie haben damit die Möglichkeit, Ihre

mc konstruktiv zu beeinflussen. Die Redaktion ist interessiert daran, Ihren Wünschen und Anregungen soweit wie möglich entgegenzukommen. Bitte haben Sie auch Verständnis dafür, daß der Fragebogen relativ umfangreich ist. Eine solche Erhebung macht man nicht alle Tage, und deshalb haben wir ziemlich alles hineingepackt, wovon wir glauben, daß Sie dazu eine Meinung haben könnten. Selbstverständlich ist der Fragebogen auch so ausgelegt, daß er leicht per Computer ausgewertet werden kann – oder hätten Sie von mc etwas anderes erwartet?!

## Nochmals: Datenbanken

Sowohl für das Genie-III als auch für den QX-10 werden mittlerweile leistungsfähige Datenbank-Programme angeboten. Das Genie-III-Programm „DIS“ (Daten-Informations-System) von Trommeschlager erlaubt es dem Anwender, alle Datenfelder wie Bezeichnung, Eingabelänge, Kommentare usw. selbst zu definieren und so seinen individuellen Bedürfnissen anzupassen. Auch Druckroutinen und Möglichkeiten zur mathematischen Verknüpfung zwischen einzelnen Datenfeldern sind integriert. Die Bedienung ist dialogorientiert in drei Menüebenen.

Das von Epson für den QX-10 lieferbare Datenbank-Programm heißt „EDDI“ (Epson's datenbank-orientiertes Datei-Informationssystem). Es ermöglicht das Anlegen von Datensätzen und Dateien, das spätere Einfügen und Ändern, das Sor-

tieren und auch das Rechnen innerhalb der Datenfelder. Es gibt sechs Feldtypen: Rechenfeld, Datumfeld, umgekehrtes Namensfeld, numerisches Feld, Zeichenfeld und druckaufbereitetes Feld. Letzteres dient zur formatierten Ausgabe von Zahlen. EDDI läßt auch das Erstellen von Auswahldateien zu, z. B. alle Adressen, die „Firma“ enthalten, gesondert zusammenzustellen. Datenschutz ist per Paßwort möglich.

## FTZ gibt nach

Das Fernmeldetechnische Zentralamt der Deutschen Bundespost wird, wie wir in Erfahrung bringen konnten, wegen der massiven Nachfrage von Industrie und Anwendern in Zukunft akustisch gekoppelte Telefon-Modems auch hierzulande außer im Originat- auch im Answer-Modus zulassen. Im Klartext heißt das: Zukünftige Modems dürfen nun auch in Deutschland einen Umschalter besitzen, mit dem man die Lage der Tonfrequenzpaare umschalten kann.

Dadurch wird es auch offiziell möglich, zwei Akustikkoppler miteinander in Verbindung treten zu lassen. Bisher war es Vorschrift, daß wenigstens auf einer Seite der Verbindung ein galvanisch gekoppeltes Modem installiert war.

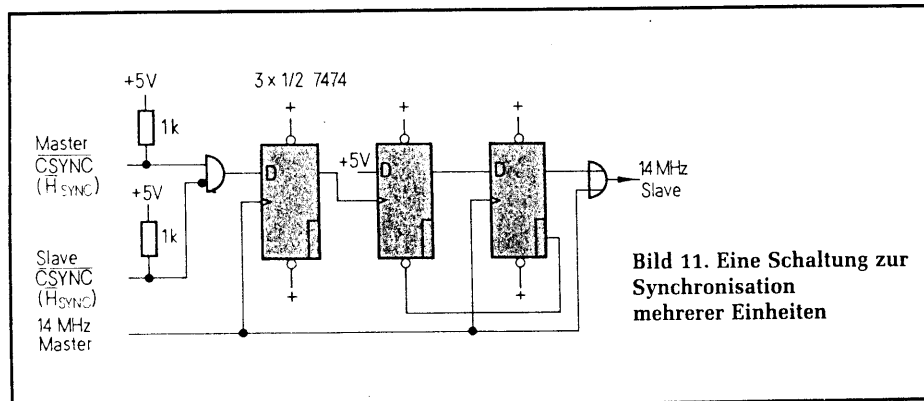


Bild 11. Eine Schaltung zur Synchronisation mehrerer Einheiten

Rolf-Dieter Klein

# Das mc-Grafik-Terminal

## Teil 2: Software

Nachdem im ersten Teil die Schaltung des Grafik-Terminals besprochen wurde, soll hier die Betriebssoftware vorgestellt werden. Es handelt sich dabei um ein Programm von 8 KByte, das Funktionen wie Terminalemulation (TVI 950), Tektronix 4010-Modus, einen Logo-Modus und die mc-Grafik-Sprache beinhaltet. Flächenfüll-Algorithmen sind darin genauso enthalten wie ein Kreisgenerator.

Das Terminal kann, wie schon im Hardware-Teil angedeutet, auf zwei verschiedene Arten eingesetzt werden. Einmal als zusätzliches Peripheriegerät, einmal als eigenständiges Terminal, wenn es mit einer Tastatur versehen wird. Beim Anschluß des Terminals ist darauf zu achten, daß die beiden Eingänge CTS des Kanals A und B richtig beschaltet sind. Sie müssen einen Ruhepegel von +12 V haben, denn sonst verschwindet der Cursor des Terminals nach Eingabe bestimmter Zeichen (z. B. BEL) und TERM1 wartet auf CTS als Freigabe. Umgekehrt muß zumindest der Ausgang D des Kanals A mit dem Computer verbunden sein, damit der Hostrechner angehalten werden kann, sobald zu viele Daten aufeinander folgen.

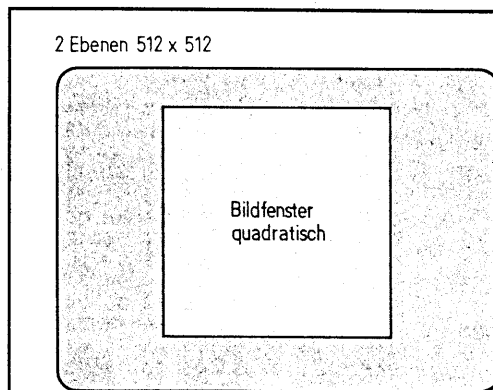
### Die Baudrate

Bild 1 zeigt eine Tabelle, in der die Jumperstellen eingetragen sind. Ohne jede Brücke sind 9600 Baud eingestellt, was auch die empfohlene Betriebsart darstellt. Die Jumper befinden sich direkt neben dem Widerstandsnetzwerk. Bit 7 der STIB ist dabei das höherwertige Bit der Baudrate, opt1 und opt2 (Bit 3 und Bit 2) sind nicht verwendet. Die Baudrate 19 200 Baud läßt sich wegen der verwendeten Quarzfrequenz mit der STI nicht einstellen, dafür stehen aber noch ein paar höhere Geschwindigkeiten (14 400, 28 800) zur Verfügung. Das Terminal kann insgesamt mit drei verschiedenen Grafik-Prozessoren ausgerüstet werden: mit dem EF 9365, dem

```

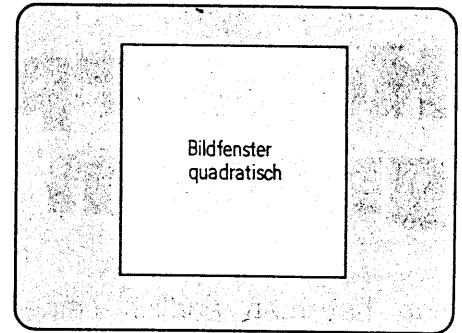
; stib sw-belegung
; 7   6   5   4   3   2   1   0
; baud baud baud baud opt1 opt2 dtr cts
;
; baud rate table
; 1111 = 9600 baud
; 1110 = 4800 baud
; 1101 = 2400 baud
; 1100 = 1200 baud
; 1011 = 600 baud
; 1010 = 300 baud
; 1001 = 110 baud
; 1000 = 75 baud
; 0111 = 50 baud
; 0110
; 0101
; 0100
; 0011
; 0010
; 0001 = 14400 baud
; 0000 = 28800 baud
    
```

**Bild 1.**  
Zur Baudrateneinstellung



**Bild 2.** Der EF 9365 erzeugt dieses Bildformat

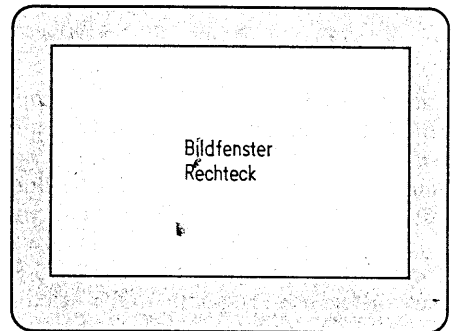
4 Ebenen 512 x 256



EF 9366  
Quarz: 14 MHz  
Jumper: default

**Bild 3.** Der EF 9366 und sein Bildformat

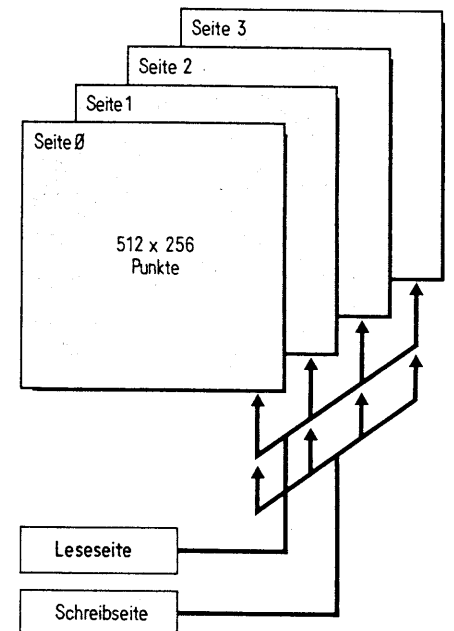
- a) 4 Ebenen 512 x 256
- b) 2 Ebenen 512 x 512



EF 9367  
Quarz: 12 MHz  
Jumper:

- a) x-trennen, f trennen, auf 0V-legen
- b) x-trennen, J2 auf 65 legen.
  - x = Pin 19 (im Layout bezeichnet, Lötseite)
  - f = Pin 8 (im Layout bezeichnet, Bestückungsseite)

**Bild 4.** Das Bildformat des EF 9367



**Bild 5.** Die Aufteilung in vier Bildebenen



EF 9366 und dem neueren EF 9367. Wir wollen standardmäßig den EF 9366 verwenden, der auch der preisgünstigste ist. Mit ihm erhält man eine Auflösung von 512 x 256 Pixel in je 4 Bildebenen. Bild 2 zeigt das Arbeitsergebnis des EF 9365, Bild 3 das des EF 9366 und Bild 4 das für den EF 9367. Der EF 9367 erzeugt ein Rechteckfenster, während die anderen Prozessoren quadratische Bildfenster besitzen. Beim EF 9367 tritt ein Vektor unter Umständen gespiegelt wieder ins Bild, wenn die Endpunkte außerhalb liegen, da in unserer Schaltung der X9-Ausgang nicht ausgewertet wird.

**Die Bilddarstellung des EF 9366**

Wir arbeiten also mit dem EF 9366 und erhalten eine Aufteilung nach Bild 5. Es zeigt vier Bildseiten mit je 512 x 256 Bildpunkten. Es kann immer nur eine Seite auf dem Bildschirm dargestellt werden, daher gibt es einen Lesezeiger. Geschrieben werden kann aber auch in eine unsichtbare Seite. Dafür ist der Schreibzeiger gedacht. Wie wir später noch genauer sehen, gibt es die Möglichkeit der quasimultanen Darstellung zweier Seiten, bei der die Seiten synchron (durch einen Grafik-Befehl veranlaßt) mit der Bildwechselfrequenz geschaltet werden. Es ergibt sich dann ein leicht flimmerndes Bild, ähnlich wie beim Zeilensprungverfahren. Dafür sind aber zwei Seiten „gleichzeitig“ sichtbar und es läßt sich dann mit Graubstufung arbeiten (wenn auf zwei Seiten das gleiche Bild ist, wirkt es heller als wenn nur eine Seite das Bild zeigt). Damit können auch Fadenkreuze und vieles mehr dargestellt werden.

**Die Textdarstellung**

Nach dem Einschalten der Term-Platine erscheint an der linken oberen Ecke im

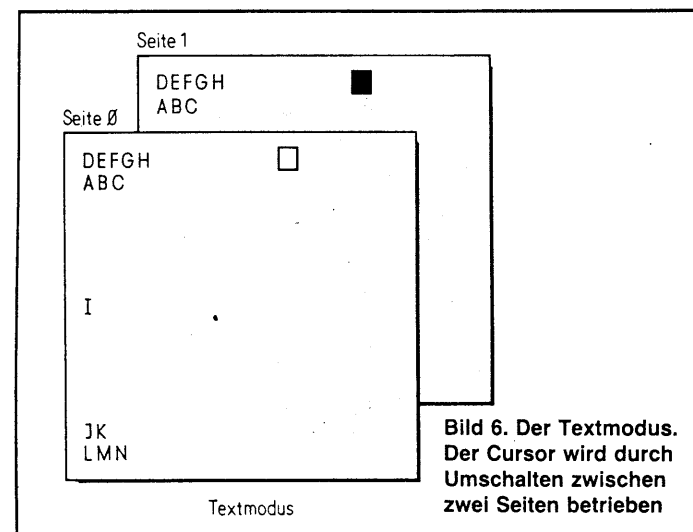
Bild ein blinkender Cursor. Die Platine ist nun bereit zur Texteingabe. Bevor wir auf die Befehle eingehen, zunächst etwas über die interne Darstellung von Texten. Wer schon einmal mit dem Grafik-Prozessor gearbeitet hat, wird wissen, daß er keine Befehle für die Verschiebung des Textfensters (Scrolling) besitzt. Daher muß das Scrolling, was auch für das Einfügen und Löschen von Zeilen gilt, per Software durch Neuschreiben des gesamten Bildes erfolgen. Die Besonderheit hierbei ist, daß jeder Punkt einzeln neu erzeugt werden muß. Mit einem konventionellen Verfahren ohne Grafik-Prozessor wäre das recht aussichtslos, denn dazu wäre eine beträchtliche Zeit notwendig. Hier ist der GDP von Thomson jedoch so schnell, daß ein Wiedereinschreiben eines Bildes im wesentlichen nur davon abhängt, wie schnell die einzelnen Zeichen angeliefert werden. Um das Löschen des alten Bildes und das Wiedereinschreiben ohne Flimmern zu realisieren, wurden zwei Bildebenen verwendet. Eine Bildebene ist sichtbar, während die andere für das neue Bild vorbereitet wird, dann wird umgeschaltet und die neue Ebene dargestellt. Bild 6 zeigt die Situation. Der Cursor entsteht dadurch, daß er auf der einen Bildebene geschrieben wird, auf der anderen nicht. Die Seiten werden nach dem Bildaufbau mit einer konstanten Frequenz hin- und hergeschaltet.

Um die Textausgabe optimal zu gestalten, ist ein zusätzlicher Bildwiederhol-speicher nötig, denn es ist nicht möglich, die Bildinformation, oder gar die Zeichencodes aus dem Bildspeicher wieder rückzulesen. Bild 7 zeigt die Speicherorganisation im CPU-Speicher. Ein Speicher mit 80 x 24 Zeichen beinhaltet die Zeicheninformationen. Dane-

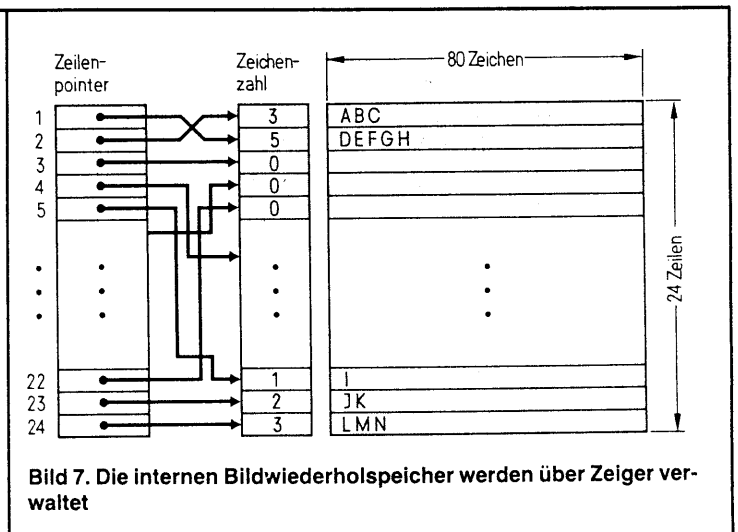
ben gibt es einen Speicher der Tiefe 24, in dem in einem Byte die jeweilige Anzahl der tatsächlich in dieser Zeile vorhandenen Zeichen abgelegt ist. Damit das Scrolling, Löschen und Einfügen von Zeilen möglichst schnell geht, gibt es einen weiteren Speicherraum von 24 Zeigern, die auf den Bildwiederhol-speicher gerichtet sind. Eine Zeile wird gelöscht, indem die Zeigerkette verändert wird. Nach Veränderung wird das Bild auf einer Seite gelöscht und neu aufgebaut. Das Löschen geschieht durch Einschreiben des alten Inhalts im Löschmodus um Zeit zu sparen, und außerdem ist es dadurch möglich, Texte und Grafik zu mischen und beim Scrolling die Grafik nicht zu zerstören, solange sie beim Scrollvorgang nicht von Zeichen durchkreuzt wird.

**Die Befehle der Grafik-Karte**

Nun zu den einzelnen Befehlen. Bild 8 zeigt eine Tabelle aller vorhandenen Befehle. Der Alpha-Modus ist der Grundmodus, der auch nach Reset vorliegt. Die Befehle wurden so gewählt, daß Wordstar mit der TVI-950-Installation direkt läuft. Es sind natürlich nicht alle TVI-950-Befehle realisiert, so fehlen zum Beispiel Invers, Underline und Blinkend. Alle diese Eigenschaften können aber über Grafik-Befehle viel effektiver dargestellt werden (und außerdem wäre dazu ein weiterer Attributspeicher notwendig gewesen). Hinzugefügt wurde die Umschaltung auf den deutschen Zeichensatz mit ESC z 1. Dann werden die Zeichen „ÄÖÜäöüß“ geplottet. Die Scrollgeschwindigkeit verringert sich dadurch leider, und zwar um so mehr, je mehr dieser Zeichen auf dem Bildschirm sind. Im Mittel scrollt das Terminal jedoch so schnell, daß es nicht möglich ist, die Zeichen zu lesen.



**Bild 6. Der Textmodus. Der Cursor wird durch Umschalten zwischen zwei Seiten betrieben**



**Bild 7. Die internen Bildwiederhol-speicher werden über Zeiger verwaltet**

## Tastatur-Eingabe STI-B

-----

0..7fh Werden an die STI-A geleitet

80h Lokal-Modus ein -- gleichzeitig als No-Scroll-Befehl verwendbar. Term1 läßt sich verlustfrei mit dem nächsten Befehl wieder starten

81h Lokal-Modus aus -- Scroll weiter Befehl

Im Lokal-Modus können alle Befehle über die Tastatur gegeben werden, die sonst vom Hostrechner kommen.

### Befehle

=====

#### ALPHA-MODUS

-----

Ist nach dem Einschalten aktiv

20h..7fh Codes der sichtbaren ASCII-Zeichen

07h CTRL G Bell wird an STI-B ausgegeben

08h CTRL H Backspace

09h CTRL I Cursor Right

0ah CTRL J Linefeed

0bh CTRL K Cursor Up

0ch CTRL L Cursor Right

0dh CTRL M Carriage Return

16h CTRL V Cursor Down

1ah CTRL Z Clear

1eh CTRL ^ Home

#### Escape Sequenzen

1bh 2eh x ESC . n Cursor Attribute setzen (n='0','1','2','3','4'). 1bh 2eh 31h entspricht ESC . 1 und läßt Cursor blinken

1bh 3dh y x ESC = r c Cursor setzen (y,x =20h..7fh)

1bh 3fh ESC ? Cursor Adresse abfragen Ergebnis wird über STI-A in dem Format wie oben (r c) ausgegeben

1bh 44h x ESC D c c='L' Lokal-Modus sonst Voll-Duplex

1bh 51h ESC O Zeichen bei Cursorposition einfügen

1bh 57h ESC W Zeichen bei Cursorposition löschen

1bh 45h ESC E Zeile bei Cursorposition einfügen

1bh 52h ESC R Zeile bei Cursorposition löschen

1bh 54h ESC T Zeile ab Cursorposition bis zum Ende löschen

1bh 74h ESC t Zeile ab Cursorposition bis zum Ende löschen

1bh 59h ESC Y Seite ab Cursorposition bis Ende löschen

1bh 79h ESC y Seite ab Cursorposition bis Ende löschen

1bh 7ah x ESC z n n='0' amerikanische Zeichensatz; n='1' deutscher Zeichensatz

#### Dopplescape Sequenzen

CR = 0dh Carriage return

1bh 1bh 54h 0dh ESC ESC T CR Tektronix-4010-Modus

1bh 1bh 54h x x x x x 0dh ESC ESC T dx dy sx sy f CR

Tektronix-4010-Modus. dx dy werden auf die Koordinaten für das Bildfenster addiert (Verschiebung des Bildfensters, Bereich -1024..+1024, die Werte dx,dy,sx,sy und f können dezimal gegeben werden). sx und sy sind Zahlen, die den Abbildungsmaßstab bestimmen. Das Bild wird in der jeweiligen Richtung um 2 hoch sx bzw. 2 hoch sy gestaucht. Wenn sx=1 und sy=2 bei einer Auflösung von 512\*256 Bildpunkten gewählt sind, dann ist das gesamte Tektronix Bildfenster mit 1024 x 1024 Bildpunkten sichtbar. Wird für f ein Wert angegeben (z.B. 1),dann ist ein Fadenkreuz an der aktuellen Cursorposition sichtbar. Wird f weggelassen ist kein Fadenkreuz vorhanden. Werden auch dx,dy,sx und sy weggelassen, wie im Kommando oben, so wird automatisch dx=0 dy=0 sx=1 sy=2 gesetzt

1bh 1bh 4ch 0dh ESC ESC L CR Logo-Modus. Um eine unverzerrte Darstellung bei einer Auflösung von 512 \* 256 Bildpunkten zu erreichen, wird die y-Koordinate jeweils durch 2 geteilt

1bh 1bh 4ch 30h 0dh ESC ESC L O CR Logo-Modus. Ohne Division für eine Auflösung von 512 \* 512 Bildpunkten ist das Bild unverzerrt

1bh 1bh 47h ESC ESC G Grafik-Modus

1bh 1bh 43h ESC ESC C Durchgangs-Modus. An der STI-A ankommende Zeichen werden an die STI-B weitergereicht. Abschalten nur mit ESC ESC R, das aber auch noch übertragen wird. Dieser Befehl wird zum Beispiel für eine Farberweiterung oder einen Druckeranschluß benötigt

1bh 1bh 50h ESC ESC P Parallel-Modus. Wie oben, jedoch mit Verarbeitung der Zeichen im Terminal

1bh 1bh 52h ESC ESC R Rücksetzen

#### GRAFIK-MODUS

-----

Das Zeichen n steht für die Eingabe eines numerischen Wertes. n kann eine vorzeichenbehaftete Dezimalzahl sein oder eine sedezimale Zahl: 0, -50, \$FF, -\$4034. Mehrere numerischen Werte werden durch Leerzeichen getrennt.

CR ist das Abschlußzeichen. Es kann bis auf wenige Ausnahmen auch das Zeichen Strichpunkt (;) verwendet werden.

A in den Modus zurückschalten, von dem aus die Grafik aufgerufen wurde

M n1 n2 CR Positionieren auf x=n1 y=n2

D n1 n2 CR Vektor nach x=n1, y=n2 zeichnen

m b1 b2 b3 b4 Wie M, jedoch mit binärer Übertragung und daher schnell. x = b1,b2 (b1 ist höherwertiges Byte, b2 ist niederwertiges Byte) y = b3,b4

d b1 b2 b3 b4 Wie D, jedoch binär. Format wie bei m

J n1 n2 CR Wie D, jedoch mit Angabe von relativen Koordinaten dx = n1 dy = n2

P n CR Seite asynchron anwählen. n = Nummer der Schreibseite \* 4 + Nummer der zu lesenden Seite z.B. Schreibseite =3, Leseseite = 1

Bild 8. Die Befehls-Liste von Term1 (Version 3.4)

n = 3\*4 + 1 = 13. Befehl:  
P 13 CR

S n CR Seite synchron anwählen.  
n wird wie oben bestimmt

X n CR Die Seiten 0,1,2,3 werden  
zyklisch angezeigt, n gibt die  
Sichtdauer einer Seite an.  
Die Anzeigedauer beträgt  
n \* 20ms  
n = 0 beendet den Wechsel

Y n CR Es werden jeweils nur zwei Seiten  
zyklisch angezeigt und zwar die  
Seiten 0 und 1, wenn eine dieser  
beiden Seiten als Leseseite definiert  
war, sonst 2 und 3

C Löschen der aktuellen Schreibseite

Z Löschen aller Seiten 0 bis 3.  
Seite 3 ist danach als Lese- und  
Schreibseite angewählt

G n1 n2 CR Befehl an GDP  
n1 = Nummer des GDP-Ports (0..15),  
n2 = Datenwert an diesen Port

B text CR Text (20h..7fh) an GDP-Port 0 senden.  
Der Text wird ab aktueller  
Koordinate des GDP ausgegeben

V binär 00h Binärdaten an den GDP-Port 0 senden.  
Das ASCII-Zeichen NUL beendet die  
Obertragung

O n1 n2 n3 n4 CR Ellipsenabschnitte zeichnen.  
Mit n1 wird die Länge der Halbachse  
in x-Richtung angegeben, mit  
n2 die Länge der Halbachse in  
y-Richtung.  
Mit n3 wird der Startwinkel  
bezüglich der x-Achse in Grad  
angegeben. Mit n4 der Endwinkel  
des Ellipsenabschnittes.  
Der Ellipsenabschnitt wird von  
der aktuellen x,y-Koordinate bis  
zum Erreichen des Endwinkels  
gezeichnet. Der Ellipsenmittelpunkt  
wird aus der Startwinkel- und Halbachsen-  
angabe vor Beginn des Zeichnens  
automatisch errechnet.

O n1 n2 n3 n4 1 CR Wie oben oben, jedoch der vom  
Kurvenstück und den Radien zum  
Mittelpunkt begrenzte Raum  
gefüllt (Torte)

R n1 n2 CR Rechteck ab aktueller x,y-Koordinate.  
n1=dx und n2=dy geben die Breite und  
die Höhe des Rechtecks an

L n1 n2 n3 n4 ... nn nm CR Polygon zeichnen, mit absoluten  
Koordinaten.  
x0=n1 y0=n2 gibt die Startposition an,  
alle weiteren Paare geben die  
Eckpunkte des Polygons an.  
Der letzte Eckpunkt wird wieder  
mit dem Startpunkt verbunden

L n1 n2 n3 n4 n5 n6 CR Dreieck gefüllt zeichnen  
x0 = n1 y0 = n2  
x1 = n3 y1 = n4  
x2 = n5 y2 = n6

F n1 n2 n3 CR Fadenkreuz zeichnen,  
an Position x=n1 y=n2,  
auf Seite n3 (0..3).  
Altes Fadenkreuz wird gelöscht.  
Die Schreib- und Leseseite  
bleiben erhalten

WA string CR Symbol für den Fadenkreuz-Befehl  
umdefinieren.  
string ist eine Zeichenkette  
mit Zeichen im ASCII-Bereich (30h,31h,40h..5fh).  
0 (30h) = Schreibstift hoch,  
1 (31h) = Schreibstift runter.  
Der Code für die Schreibstiftbewegung  
berechnet sich wie folgt:  
Richtung + (8 \* Länge) + 40h  
Es stehen die Richtungen von  
0 bis 7 (\* 45 Grad) zur Verfügung

WA CR Symbol für den Fadenkreuzbefehl  
auf das Symbol "Fadenkreuz"  
zurücksetzen

WB Fadenkreuzsymbol an aktueller  
x,y-Koordinate auf der  
aktuellen Schreibseite setzen

WC n1 n2 CR Fadenkreuzsymbol vergrößern  
und drehen.  
n1 = Vergrößerungsfaktor (1..255)  
n2 = Drehung (0..7)

WC CR Rückstellen auf n1 = 1,  
n2 = 0

WD n1 n2 n3 ... nn CR  
Download. Es können die Daten  
n2 bis nn ab Adresse n1  
in den Arbeitsspeicher der  
Term1 geladen werden

WE n1 CR Programm auf Adresse = n1  
starten

WF n1 CR Byte mit Adresse n1 aus dem  
Speicher der Term1 lesen.  
Ergebnis wird binär über den  
Port der STI-A übertragen

Dopplescape

1bh 1bh 41h ESC ESC A Alpha-Modus (oder Logo,T4010)

1bh 1bh 43h ESC ESC C Durchgangsmodus

1bh 1bh 50h ESC ESC P Parallel-Modus

1bh 1bh 52h ESC ESC R Rücksetzen

LOGO-MODUS  
-----

A Alpha-Modus

Z Lösche Bildschirm, alte  
Koordinaten bleiben

M n1 n2 n3 CR Turtle Positionieren für Initialisierung.  
x=n1 y=n2 Startwinkel=n3, Angabe in  
Grad (0..359)

F n1 CR Vorwärts (bei negativer Zahl Rückwärts).  
n1 ist die Schrittzahl

P n1 CR Drehung der Schildkröte von dem  
aktuellen Winkel aus um den Winkel  
n1 (in Grad, + und - möglich).

D Pen Down

U Pen Up

TO Turtle unsichtbar

T1 Turtle sichtbar

1bh 1bh 41h ESC ESC A Alpha-Modus

1bh 1bh 47h ESC ESC G Grafik-Modus aufrufen

1bh 1bh 43h ESC ESC C Durchgangsmodus

1bh 1bh 50h ESC ESC P Parallel-Modus

1bh 1bh 52h ESC ESC R Rücksetzen

TEKTRONIX-4010-MODUS  
-----

Es sind implementiert:  
Alpha-, Grafik-, Incremental-, Pointplot-Modus

Nicht implementiert:  
Gin-Modus nicht implementiert  
(Definitionen siehe Tektronix 4010-Manual)

T4010-Alpha-Modus:

20h bis 7fh Textzeichen im  
Tektronix-Alpha-Modus

08h	Backspace	1bh 0ch	Clear und in Alpha-Modus zurück
0bh	Cursor Up	Pointplot-Modus:	wie beim Grafik-Modus, nur daß keine Vektoren sondern nur die Endpunkte gezeichnet werden
0ah	Linefeed	Grafik-Modus:	
0dh	Carriage Return	20h bis 7fh	Vektorkoordinaten
1ch	Pointplot-Modus	Die Koordinaten x und y werden in niederwertige (Low) und höherwertige (High) Teile zerlegt.	
1dh	Grafik-Modus	Lowx = 40h..5fh	die unteren 5 Bit der Koordinate werden verwendet
1eh	Incremental-Modus	Lowy = 60h..7fh	die oberen 5 Bit der Koordinate werden verwendet
1bh 05h	Übertragen der Cursorposition im 1024 * 1024 Raster	Highx, Highy = 20h..3fh	
1bh 0ch	Clear All	HIY LOY HIX LOX	Übertragungsformat: Es werden maximal vier Koordinatenteile übertragen.
1bh 1ah	Gin-Modus -- Nicht implementiert bleibt in Alpha-Modus	HIY LOX	Sofern sich die Koordinatenteile nur teilweise ändern, können auch bleibende Koordinatenstücke nach dem links stehenden Schema fortgelassen werden
1bh 1ch	Spezial Pointplot-Modus -- hier wie Pointplot-Modus	LOY HIX LOX	
Doppel-ESC-Sequenzen nur vom T4010-Alpha-Modus aus wirksam			
1bh 1bh 41h	ESC ESC A	Alpha-Modus	
1bh 1bh 47h	ESC ESC G	Grafik-Modus aufrufen	zusätzlich bei T4014 (Auflösung 4096 x 4086)
1bh 1bh 43h	ESC ESC C	Durchgangsmodus	HIY EXTRA LOY HIX LOX Extra wird ignoriert
1bh 1bh 50h	ESC ESC P	Parallel-Modus	
1bh 1bh 52h	ESC ESC R	Rücksetzen	
Incremental-Modus:			
Zeichen	Richtungsvektor.		
A E D F B J H I	Jeweils in 45 Grad Schritten gegen den Uhrzeiger (A= 0 Grad ... I = 315 Grad)		
P	Pen Down	07h	Erster Vektor wird doch geschrieben
20h	Pen Up	0dh	Alpha-Modus
0dh	in Alpha-Modus zurück	1ch	Pointplot-Modus
1fh	in Alpha-Modus zurück	1dh	Vektor ungeschrieben
		1eh	Incremental-Modus
		1fh	Alpha-Modus
		1bh 05h	Grafik-Cursor-Koordinaten Ausgabe
		1bh 0ch	Clear und dann Alpha-Modus
		1bh 1ah	Gin-Modus -- hier Alpha-Modus
		1bh 1ch	Spezial-Pointplot -- hier Pointplot

Mit einer Doppel-Escape-Sequenz kann man in einen der anderen Modi kommen. In den Grafik-Modus kommt man mit ESC ESC G. Dort gibt es eine Vielzahl von Befehlen. Mit ESC ESC A kommt man wieder in den vorherigen Modus zurück. M und m (binär) positioniert den Schreibzeiger, mit D oder d (bei Binärdaten) wird ein Vektor gezeichnet. Die Parameter werden dezimal oder hexadezimal angegeben. Sie können auch ein Vorzeichen enthalten. Beispiele:

M100 150cr. Es wird die Position x = 100 und y = 150 eingestellt (cr = Wagenrücklauf)

Mit D0 10cr wird ein Vektor zur Position x = 0 und y = 10 gezeichnet.

Mit M\$FF \$80cr wird die Position x = 255 und y = 128 eingestellt. Die weiteren Befehle sind in Bild 8 gezeigt.

### Logo-Modus

Mit dem Grafik-Terminal läßt sich „Turtle-Grafik“ realisieren. Dort gibt es

einen Cursor, der eine Richtung anzeigt. Dieser Cursor (Turtle) kann in Ein-Grad-Schritten gedreht werden. Und mit einem Schreite-Befehl kann dann entlang der eingestellten Richtung ein Vektor gezeichnet werden. Intern besitzen die Koordinaten vier Schutz-Bits, um bei kleinen Distanzen Schritte mit ausreichender Genauigkeit in der richtigen Richtung zu ermöglichen. Eine Schleife mit der Sequenz F 1 P 1 ergibt, 360mal ausgeführt, einen Kreis.

Die „Turtle“ wird durch Quasisimultandarstellung automatisch in einer zweiten Ebene eingeblendet, sie läßt sich aber auch abschalten, wenn der Befehl T0 gegeben wird. Mit Dopplescape-Sequenzen können andere Modi aufgerufen werden. Dabei kann von dort wieder in den Logo-Modus zurückgesprungen werden, so daß zum Beispiel Grafik-Befehle mit Logo-Befehlen gemischt werden können.

### Tektronix emulieren

Insbesondere für Standard-Pakete ist der Tektronix-Modus gedacht (z. B. Plot 10). Nicht implementiert ist dabei der Gin-Modus, bei dem ein Fadenkreuz auf dem Bildschirm sichtbar ist und in dem digitalisiert werden kann.

Nun nochmal zum Grafik-Modus, im folgenden mc-Grafik genannt, zurück. Dort gibt es die Möglichkeit ein Fadenkreuz einzublenden. Bild 9 zeigt ein Beispiel. Auf Ebene 0 sei ein Vektor gezeichnet. Durch den Befehl Y1 werden die beiden Seiten 0 und 1 synchron umgeschaltet. Die Zahl 1 gibt die Umschaltrate an (1: alle 20ms einmal). Der Befehl F blendet ein Fadenkreuz ein. Die Form des Fadenkreuzes ist voreingestellt, jedoch kann sie auch umprogrammiert werden. Dazu wird der Befehl WA verwendet. Es muß nach WA in Textform eine Codesequenz angegeben werden, die eine Figur inkremental beschreibt. Bild 10 zeigt die Codierungsvorschrift. Bild 11 zeigt ein Beispiel. Die Pfeile zeigen die Reihenfol-

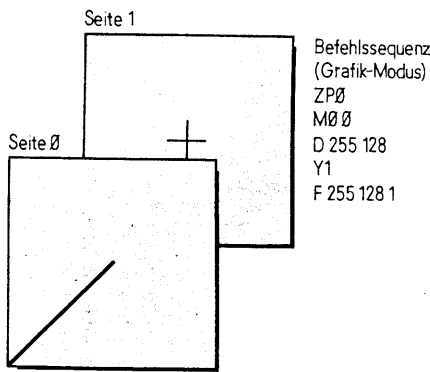
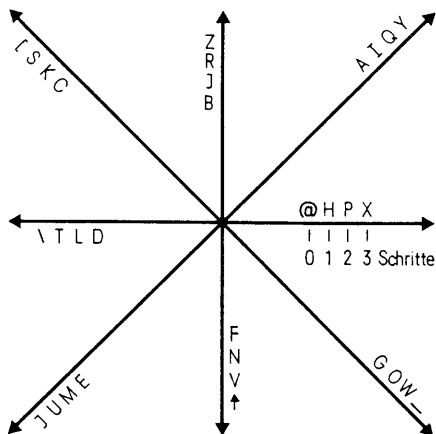


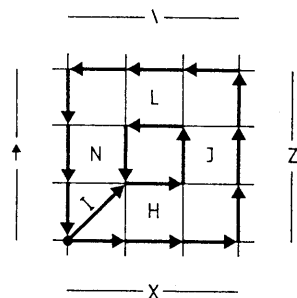
Bild 9. Der Fadenkreuzmodus

Befehlssequenz  
(Grafik-Modus)  
ZPØ  
MØØ  
D 255 128  
Y1  
F 255 128 1



0 = up (pen)  
1 = down (pen)

Bild 10. Die Codierung von Figuren



"XZ\↑ØI1HJLN"

Bild 11. Ein Beispiel zu einer Figurcodierung

```
'10 CLEAR 2000
20 REM Beispiele fuer TERM 1
30 REM Rolf-Dieter Klein 830508
40 REM TERMINAL-TVI-950 Funktionen
50 PRINT CHR$(26); 'Clear Screen
60 FOR X=0 TO 79 'Alle Spalten
70 Y = SIN(X/79*2*PI)*11+11
80 Y = INT(Y)
90 PRINT CHR$(27); "="; CHR$(32+Y); CHR$(32+X); "*"
100 NEXT X
110 REM Wegschieben durch Scrollen
120 FORI=1TO40:PRINT:NEXT I
130 REM
140 REM verschiedene Zeichensaetze
150 PRINT CHR$(27); "=", CHR$(32+5); CHR$(32)
160 PRINT CHR$(27); "z0" 'amerikan Zeichen
```

Bild 12. Ein Programmbeispiel für den Textmodus

```
170 PRINT
180 FORI=32 TO 127:PRINT CHR$(I);:NEXT I
190 PRINT
200 PRINT CHR$(27); "z1" 'deutsche Zeichen
210 FORI=32 TO 127:PRINT CHR$(I);:NEXT I
220 PRINT
230 FORI=1TO1000:NEXT I
240 REM Einfuegen, Loeschen etc.
250 FOR J=1 TO 2
260 PRINT CHR$(26); CHR$(27); "="
270 PRINT "Testsatz fuer Einfuegen etc."
280 PRINT CHR$(11); ' cursor up
290 FORI=1TO10:PRINT CHR$(27); "E";:NEXT I
300 FORI=1TO10:PRINT CHR$(27); "R";:NEXT I
310 FORI=1TO10:PRINT CHR$(27); "Q";:NEXT I
320 FORI=1TO10:PRINT CHR$(27); "W";:NEXT I
330 PRINT 'cr ausgeben, wegen autocr von Basic
340 NEXT J
```

```
10 REM Graphic Modus
20 REM Rolf-Dieter Klein 830508
30 PRINT CHR$(27); CHR$(27); "GZPO" 'Graphic Modus
40 PRINT "MO 0;RS11 255" 'Rechteck-Rahmen zeichnen
50 PRINT "M50 200;G3 $22;BGraphic-Befehle"
60 PRINT "M200 50;O100 50 0 0 1"
70 PRINT "L300 0 511 20 400 200"
80 REM Symbol definieren
90 PRINT "WA"; "HHHHIJKLLMMNO"
100 PRINT "Y1" 'zwei Seiten quasismultan
110 PRINT "WC2 0" 'VERGROESSERUNG 2 DREHUNG 0
120 FOR X=0 TO 511
130 PRINT "F"; X; INT(SIN(X/100*2*PI)*120+127); 1
140 NEXT X
150 PRINT "A" 'in Alpha zurueck
999 PRINT "A"
```

Bild 13. Ein Programmbeispiel für den Grafik-Modus

```
10 REM Kleine LOGO-Demonstration
20 REM Rolf-Dieter Klein
30 PRINT CHR$(27); CHR$(27); "L" 'std mode
40 PRINT "M255 255 90" 'umgerechnet auf 512x256
50 FOR I=0 TO 360/10
60 GOSUB 130 'Muster 1
70 PRINT "P-10"
80 NEXT I
90 PRINT "TO" 'Turtle off
100 FOR I=1 TO 1000:NEXT I
110 PRINT "A"
120 END
130 REM Unterprogramm Figur
140 FOR N=1 TO 4
150 PRINT "F150;P"; 360/4
160 NEXT N
170 RETURN
```

Bild 15. Logo-Demo

```
10 REM Statuszeile ueber Graphic Modus
20 REM Rolf-Dieter Klein 830508
30 PRINT CHR$(27); CHR$(27); "G" 'ohne Loeschen
40 PRINT "PO;MO 0;RS11 15" ' mit Rahmen
50 PRINT "PS;MO 0;RS11 15" ' beide Seiten
60 PRINT "PO;G3 $22;M40 -1;Bstatus Zeile doppelt hoch"
70 PRINT "PS;M40 -1;Bstatus Zeile hoch"
80 PRINT "A" 'wieder zurueck in Textmodus
90 LIST
```

Bild 14. Einblendung der Statuszeile

```
10 REM Tektronix Demo
20 REM Rolf-Dieter Klein 830508
30 PRINT CHR$(27); CHR$(27); "TO 0 1 2 1"
40 FOR X=0 TO 1023 STEP 4
50 Y=(SIN(X/1023*2*PI)*500+512)
55 Y=INT(Y)
60 GOSUB 100 'Tektronix Ausgabe
70 NEXT X
80 FORI=1TO1000:NEXT I
90 PRINT CHR$(27); CHR$(27); "A":END
100 REM x,y trennen und Ausgeben
105 PRINT CHR$(28); CHR$(7); 'Pointplot mode
110 PRINT CHR$(INT(Y/32)+32); CHR$(Y MOD 32 + 96);
999 RETURN
CHR$(INT(X/32)+32); CHR$(X MOD 32 + 64)
```

Bild 16. Der Tektronix-Modus



```

; initialisieren der Peripheriebausteine
;
siolist::
defb 2*18 ;Anzahl der Eintraege
;STI-A
defb stia+12,10001000b ;/16 1 Stop 8 Bit
defb stia+13,00000001b ;receive enable
defb stia+14,00000101b ;transmitter enable
defb stia+8,1 ;pointer to timer d
defb stia+0,3 ;/3
defb stia+8,2 ;pointer to timer c
defb stia+0,3 ;/3 9600 baud
defb stia+8,7 ;pointer to control c,d
defb stia+0,10011001b ;/4 /4
defb stia+8,6 ;pointer
defb stia+0,11110010b, ;I/O definieren Bit 1 =
dtr Ausgang
defb stia+1,0 ;page 0 dtr high
defb stia+8,4
defb stia+0,00000000b ;disable rest
defb stia+8,5 ;iera
defb stia+0,00010000b ;bit 4=1 enable receive
interrupt
defb stia+7,00010000b ;imra freigeben
defb stia+8,01000110b ;Vektor xx40

stillist:: ;STI-B
defb 2*16 ;adjust for:

defb stib+12,10001000b ;/16 1 Stop 8 Bit
defb stib+13,00000001b ;receive enable
defb stib+14,00000101b ;transmitter enable
defb stib+8,1 ;pointer to timer d

defb stib+0,3 ;/3
defb stib+8,2 ;pointer to timer c
defb stib+0,3 ;/3 9600 baud
defb stib+8,7 ;pointer to control c,d
defb stib+0,10011001b ;/4 /4
defb stib+8,6 ;pointer
defb stib+0,00000010b ;I/O definieren Bit 1 =
dtr Ausgang
defb stib+1,00000000b ;dtr high Freigabe
defb stib+8,4
defb stib+0,00000000b ;disable rest
defb stib+8,5 ;iera
defb stib+0,00000000b ;disable
;
;
; STI-B Bits 7 6 5 4 3 2 1 0
;
; sw sw sw sw sw sw dtr cts
;
; Baudraten Zuordnung
bdtab::
defb 10011001b,1 ;/4 /1 28800 baud
defb 10011001b,2 ;/4 /2 14400 baud
defb 10011001b,3 ;/4 /3 9600 baud
defb 10011001b,3 ;/4 /3 9600 baud
defb 10011001b,3 ;/4 /3 9600 baud
defb 10011001b,3 ;/4 /3 9600 baud
defb 10101010b,230 ;/10 /230 50 baud
defb 10101010b,154 ;/10 /154 75 baud
defb 10101010b,105 ;/10 /105 110 baud
defb 10011001b,96 ;/4 /96 300 baud
defb 10011001b,48 ;/4 /48 600 baud
defb 10011001b,24 ;/4 /24 1200 baud
defb 10011001b,12 ;/4 /12 2400 baud
defb 10011001b,6 ;/4 /6 4800 baud
defb 10011001b,3 ;/4 /3 9600 baud

```

Bild 18. Die STI-Initialisierungstabelle

ge, in welcher das Bild gezeichnet werden soll. Darunter ist die Codierung gezeigt. Mit WAXZ... wird der Code übertragen. Das Zeichen cr (Wagenrücklauf) dient der Trennung. Mit dem Befehl WCn r kann die Vergrößerung und Drehung dieses Symbols bestimmt werden. Der Befehl F schließlich macht es sichtbar.

### Ein paar Programmbeispiele

Bild 12 zeigt ein Programm in Basic, das ein paar Terminalfunktionen testet. Bild 13 zeigt ein Beispiel für den Grafik-Modus. Bild 14 zeigt die Möglichkeit, eine Statuszeile in den Alpha-Modus einzublenden. Diese Textzeile bleibt so lange erhalten, bis der Bildschirm (z. B. mit CTRL-Z) gelöscht wird. Scrolling stört sie nicht. Als Statuszeilenhöhe stehen zwei Textzeilen (25 und 26) zur Verfügung. Die Statuszeile wird mit Hilfe des Grafik-Befehls eingblendet, weshalb es auch möglich ist, Zeichnungen darin unterzubringen.

Bild 15 zeigt, wie man im Logo-Modus arbeitet. Hierbei ist übrigens zu beachten, daß automatisch eine Umsetzung in ein quadratisches Raster erfolgt (512 x 256 Bildpunkte ergeben nur ein Rechteck). Der Logomodus kann auch mit ei-

nem Zusatzparameter aufgerufen werden, so daß 512 x 512 Bildpunkte verwendet werden, um den Einsatz des EF 9365 zu ermöglichen. Bild 16 schließlich zeigt den Aufruf des Tektronix-Modus. Es ist möglich, einen Bildausschnitt zu wählen und eine Vergrößerung (eigentlich Verkleinerung) anzugeben. Damit lassen sich Bilder mit 1024 x 1024 Punkten auf einem Schirmbild unterbringen und auch Ausschnitte zeigen. Der letzte Parameter gibt an, ob ein Cursor (hier als Fadenkreuz) verwendet werden soll. Wenn ja, so wird durch das schnelle Hin- und Herschalten eine quasisimultane Darstellung erreicht. Zu beachten ist, daß bei allen diesen Darstellungen der EF 9366 verwendet werden sollte. Falls nicht, so muß über den Grafik-Modus der Befehl X2 gegeben werden, um nur alle 40 ms umzuschalten. Der EF 9365 arbeitet selbst schon mit dem Zeilensprungverfahren und ist dementsprechend langsamer. Ferner ist damit unbedingt ein lang nachleuchtender Schirm zu verwenden.

### Das Assemblerprogramm

Bild 17 zeigt Ausschnitte aus dem Assemblerlisting. Für eigene Programme, die ab Adresse 4700h bis 47FFh abgelegt

werden können, gibt es eine Sprungtabelle, um nützliche Unterprogramme verwenden zu können.

Bild 18 zeigt den Ausschnitt der Programmierstabelle der STIs. Bild 19 zeigt die Belegung des RAM-Speichers. Mit dem Auslesebefehl WF können damit zum Beispiel der Inhalt des Bildwiederholerspeichers oder andere Parameter ausgelesen werden.

### Die Arbeit mit dem Terminal

Für das Arbeiten mit diesem umfangreichen Software-Paket sei geraten, einmal jeden Befehl getrennt in einem kleinen Programm zu verwenden und zu testen. Wichtig ist ferner, daß manche Basic-Versionen automatisch CRLF nach einer bestimmten Anzahl von Zeichen einfügen und daher Vorsicht mit allen Binärbefehlen geboten ist (V-Befehl, m oder d oder Tektronix-Modus). Der mc-Grafik-Modus ignoriert zusätzliche CRLFs, außer bei den Binärbefehlen, der Tektronix-Modus nicht.

(Der Bausatz und das PROM mit Handbuch sind von der Firma GES, Graf Elektronik Systeme, Kempten, unter der Bezeichnung TERM1 lieferbar. Hexdump und EPROM vom Franzis-Software-Service, Tel. 0 89/51 17-3 31.)

3800	0000	crpos::	defw 0	; Intern	3860	00	turdo::	defb 0	; Turtle schreibt
3802	0000	newx::	defw 0	; Fuer drawto	3861	00	teilflag::	defb 0	; 1=Teilen durch 2
3804	0000	newy::	defw 0						
3806	0000	oldx::	defw 0						
3808	0000	oldy::	defw 0						
380A	0000	dx::	defw 0						
380C	0000	dy::	defw 0						
380E	00	sign::	defb 0	; Vorzeichenmerker					
380F	00	stibthere::	defb 0	; 0= STI-B nicht da					
3810	00	vzsto::	defb 0	; Merker fuer sin,cos					
3811	0000	phianf::	defw 0						
3813	0000	phiend::	defw 0						
3815	0000	ha::	defw 0						
3817	0000	hb::	defw 0						
3819	0000	x0::	defw 0	; Fuer l-Befehl					
381B	0000	y0::	defw 0						
381D	0000	x1::	defw 0						
381F	0000	y1::	defw 0						
3821	0000	x2::	defw 0						
3823	0000	y2::	defw 0						
3825	0000	xkoor::	defw 0						
3827	0000	ykoor::	defw 0						
3829	0000	newxx::	defw 0						
382B	0000	newyy::	defw 0						
382D	0000	phi::	defw 0						
382F	0000	sitemp::	defw 0						
3831	0000	dxx::	defw 0						
3833	0000	dyy::	defw 0						
3835	0000	ssto::	defw 0						
3837	0000	yaddsto::	defw 0						
3839	0000	xdx::	defw 0	; fuer Fuell-Befehle					
383B	0000	ydy::	defw 0						
383D	00	cirfuel::	defb 0	; Merker 0=nicht fuellen					
383E	0000	xalt::	defw 0	; Tektronix Merker					
3840	0000	yalt::	defw 0						
3842	00	tdark::	defb 0						
3843	0000	offx::	defw 0						
3845	0000	offy::	defw 0						
3847	00	scalx::	defb 0						
3848	00	scalx::	defb 0						
3849	0000	xgra::	defw 0	; aktuelle Position					
384B	0000	ygra::	defw 0						
384D	00	f1on::	defb 0						
384E	0000	f1x::	defw 0						
3850	0000	f1y::	defw 0	; Position Fadenkreuz					
3852	00	pflag::	defb 0	; Point Modus					
3853	00	pmode::	defb 0						
3854	0000	turx::	defw 0	; Logo-Modus					
3856	0000	tury::	defw 0	; Position					
3858	0000	turphi::	defw 0						
385A	0000	turfx::	defw 0	; alte Position					
385C	0000	turyf::	defw 0						
385E	0000	tur1phi::	defw 0						

Bild 19. Die Belegung des RAM