



# HCOPY/MAUS

**Baugruppe für Hardcopy,  
Mausanschluß und Fadenkreuz**

**für den NDR - Computer**

**Stand: Oktober 1985**

Graf Elektronik Systeme GmbH  
Magnusstr. 13 · Postfach 1610  
8960 Kempten (Allgäu)  
Telefon: (08 31) 62 11  
Teletex: 831804 - GRAF  
Telex: 17 831 804 - GRAF

Filiale Hamburg  
Ehrenbergstraße 56  
2000 Hamburg 50  
Telefon: (0 40) 38 81 51  
Filiale München  
Georgenstraße 61  
8000 München 40  
Telefon: (0 89) 2 71 58 58



## Inhaltsverzeichnis

1. Einführung.....	1
2. Technische Daten.....	2
3. Prinzipielle Beschreibung.....	2
3.1 Funktionsweise der Maus.....	2
3.2 Funktionsweise der Hardcopy.....	3
3.3 Funktionsweise des Fadenkreuzes.....	4
4. Aufbauanleitung.....	4
4.1 Achtung MOS!.....	4
4.2 Stückliste.....	4
4.3 Bestückungsplan.....	6
4.4 Layout Bestückungsseite.....	7
4.5 Bestückungsanleitung.....	8
4.6 Einstellungen an der Baugruppe.....	9
5. Test der Baugruppe.....	10
5.1 Allgemeine Tests.....	10
5.2 Test des Fadenkreuzes und der Mausschnittstelle...11	11
5.3 Test der Hardcopy-Funktion.....	12
6. Fehlersuche.....	13
6.1 Sichtprüfung.....	13
6.2 Messungen.....	13
7. Schaltungsbeschreibung.....	15
7.1 Schaltplan.....	15
7.2 Beschreibung der Schaltung.....	16
8. Anwendungsbeispiele.....	18
9. Diverses.....	19
9.1 Testprogramm für Fadenkreuz und Maus (SBC2).....	19
9.2 Testprogramm für Hardcopy (SBC2).....	21
9.3 Testprogramm für Fadenkreuz und Maus (CPU Z80).....	24
9.4 Testprogramm für Hardcopy (CPU Z80).....	26
9.5 Testprogramm für Fadenkreuz und Maus (CPU 68008).....	29
9.6 Testprogramm für Hardcopy (CPU 68008).....	30
9.7 Programm für Hardcopy unter CP/M.....	32
10. Bauelemente.....	36

# 1. Einführung

Dank unserer langjährigen Erfahrung bei der Entwicklung von Baugruppen, unter Verwendung moderner Technologie, können wir mit der Hardcopy/Maus/Fadenkreuz-Platine (im folgenden abgekürzt als HCOPIY/MAUS bezeichnet) eine zuverlässige und ausgereifte Baugruppe für Ihren NDR-Klein-Computer anbieten. Die Verwendung von Platinen mit Bestückungsaufdruck, verbunden mit den übersichtlichen Bausätzen, garantiert eine nachbausichere Schaltung mit geringer Fehlerrate.

Die ausschließliche Verwendung von TTL-Bausteinen ermöglicht eine Benutzung der Baugruppe mit allen bisherigen CPU-Platinen (SBC2, CPUZ80 und CPU68K). Die Durchführung der grafischen Funktionen (Hardcopy und Fadenkreuz) erfolgt im Zusammenspiel mit der GDP64K-Platine und einer Ansteuerungsbaugruppe für einen Drucker (z.B. SER oder IOE+CENT). Abgesehen von den genannten Baugruppen genügt ein Minimalsystem zum Betrieb der Karte.

## Welche Aufgabe kommt der HCOPIY/MAUS-Baugruppe zu ?

Durch die Verwendung moderner und platzsparender Bausteine gelang es auf der Platine drei wichtige Funktionen unterzubringen:

- Erstellung einer Hardcopy,
- Ansteuerung einer Maus,
- Ausgabe eines flimmerfreien Fadenkreuzes und
- Anschluß eines A/D-Wandlers zur Digitalisierung von Bildern.

Unter einer Hardcopy versteht man die Ausgabe des aktuellen Bildschirminhalts auf einen Drucker. Bisher gab es keine Möglichkeit die durch die GDP-Baugruppe erzeugten Texte oder Grafiken auf einen Drucker auszugeben. Die Freude über gelungene Grafiken oder ähnliches blieb auf die kurze Betrachtung am Bildschirm beschränkt. Zusammen mit einem geeigneten Programm und einem grafikfähigen Drucker (z.B. EPSON RX80) erlaubt die HCOPIY/MAUS-Platine nun die Fixierung eines Bildes auf Papier.

Als Maus bezeichnet man ein kleines Kästchen, daß bei der Bewegung auf einer flachen Unterlage dem Computer Informationen über die Bewegungsrichtung und die zurückgelegte Entfernung liefert. Die Umsetzung der Bewegung kann rein mechanisch mit einer Rollkugel oder auf optischem Wege erfolgen. Optische Mäuse arbeiten zwar genauer und verschleißärmer, doch bildet der wesentlich höhere Preis einen unangenehmen Nachteil. "Intelligente" Mäuse liefern dem Computer die Bewegungsinformation fertig aufbereitet über eine serielle Schnittstelle. Dieser Komfort besitzt allerdings auch seinen Preis. Die HCOPIY/MAUS-Platine ermöglicht den Anschluß einer preisgünstigeren mechanischen Maus oder wahlweise eines noch günstigeren Trackballs. Eine einfache Maus oder ein Trackball besitzt 4 TTL-Ausgänge entsprechend den vier Bewegungsrichtungen. Anhand der Signale dieser Ausgänge ermittelt die HCOPIY/MAUS-Baugruppe, gesteuert durch das entsprechende Programm, die durchgeführte Bewegung.

Zur Erledigung grafischer Arbeiten benötigt man oft ein Fadenkreuz um beispielsweise eine Positionierung auf eine bestimmte Stelle vornehmen zu können. Das Fadenkreuz der HCOPIY/MAUS-Baugruppe arbeitet, im Gegensatz zum Fadenkreuz der GDP64K-Baugruppe, flimmerfrei und führt daher zu einer geringeren Ermüdung des Benutzers.

Über einen zusätzlichen Port besteht die Möglichkeit einen A/D-Wandler zur Digitalisierung von Video-Signalen anzuschließen.

# 2. Technische Daten

Betriebsspannung: + 5 Volt

Stromaufnahme: ca. 550 mA

Bus Format: NDR-Klein-Bus 36 polig

Größe der Leiterplatte: 100 x 105 x 1.5 mm

Anschluß der Maus: 9 pol. Cannon-Stecker (auf Seiten der Maus)

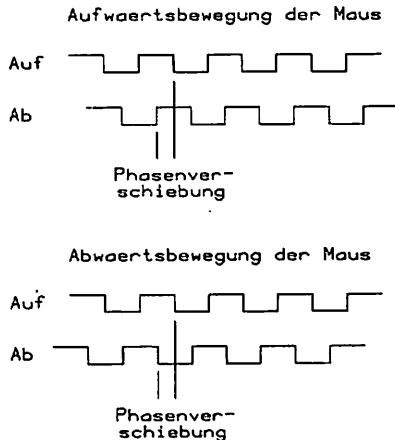
Anschluß an GDP64K-Platine: 2 x 7 pol. Stiftleiste

Anschluß an A/D-Wandler: 2 x 10 pol. Stiftleiste

# 3. Prinzipielle Beschreibung

## 3.1. Funktionsweise der Maus

Eine einfache Maus oder ein einfacher Trackball besitzt 4 TTL-Ausgänge, entsprechend den vier möglichen Bewegungsrichtungen (rechts, links, auf und ab). Bei einer Aufwärtsbewegung der Maus erscheinen dann Rechtecksignale an den beiden Ausgängen für auf und ab. Die Zahl der ausgesandten Impulse wächst proportional mit dem zurückgelegten Weg. Bei einer Abwärtsbewegung der Maus erscheinen dann beiden genannten Eingängen ebenfalls Rechteckimpulse. Wie läßt sich nun die Bewegungsrichtung ermitteln? Die beiden Signale einer Bewegungsrichtung (auf und ab bzw. links und rechts) weisen eine Phasenverschiebung zueinander auf. Anhand dieser Phasenverschiebung kann dann beispielsweise die Unterscheidung einer Auf- oder Abwärtsbewegung erfolgen. Bewegt sich die Maus nun nicht rein waage- oder senkrecht, so kann aus der Zahl der empfangenen Impulse in X- und Y-Richtung die Bewegungsrichtung ermittelt werden und somit eine Positionsbestimmung erfolgen.





### 3.2. Funktionsweise der Hardcopy

Der Grafikcontroller (EF9366) der GDP64K-Baugruppe verwaltet einen Speicher mit 64k Byte. Jedes der 524288 Bits entspricht einem der 512 x 256 Bildpunkten auf einer der 4 Bildseiten der GDP64K. Zur Ausgabe von Text oder Grafik setzt der Prozessor für jeden hell zu erscheinenden Bildpunkt das entsprechende Bit. Der Versuch eine Hardcopy durch Auslesen dieses Bildspeichers zu erstellen, gelingt leider nur bei einem Teil der verwendeten Grafikcontroller. Aus diesem Grund unterblieb die Implementierung einer Hardcopy während der Entwicklungsphase der jetzigen GDP64K-Platine. Um den Benutzern der GDP-Baugruppe trotzdem eine Möglichkeit der Hardcopy zu bieten, fand ein anderer Weg zur Erfassung des Bildinhaltes Verwendung.

In einem Fernseher oder Monitor entsteht das Bild durch ständiges Abfahren des Bildschirms mit einem feinem Elektronenstrahl innerhalb der Bildröhre. In einem handelsüblichen Monitor überstreicht der Elektronenstrahl den gesamten Bildschirm 50 Mal in der Sekunde. Bei jedem Durchlauf wandert der Elektronenstrahl dabei zeilenförmig von links nach rechts und von oben nach unten. Bei ausreichender Intensität des Strahls erscheint auf dem Bildschirm eine helle Spur. Zur Darstellung beispielsweise eines Buchstabens muß der Elektronenstrahl kleine helle Punkte in einer bestimmten geometrischen Anordnung liefern. Dazu muß der Strahl während seines Laufs über den Bildschirm in bestimmten Zeitabständen kurzzeitig (ca. 50 ns) "ein-" bzw. "ausgeschaltet" werden. Zur Steuerung des Ablaufs und der Intensität des Elektronenstrahls liefert der Grafikcontroller drei Signale. Diese Signale bezeichnet man mit horizontalem Synchronisationssignal (HSYNC), vertikalem Synchronisationssignal (VSYNC) und als Datensignal (VIDEO). Der Elektronenstrahl hinterläßt bei seiner Wanderung über den Bildschirm immer dann einen hellen Punkt, wenn auf der VIDEO-Leitung ein Low-Signal anliegt. Zur Erzeugung eines sinnvollen Bildes muß der Grafikprozessor wissen, wo sich der Elektronenstrahl augenblicklich befindet. Aus diesem Grund bedarf es einer Synchronisation zwischen Bildschirm und Prozessor. Sobald der Pegel auf der HSYNC-Leitung von Low auf High wechselt, stellt dies das Startsignal für die Ausgabe einer neuen Zeile dar. Nach ca. 63 us erreicht der Strahl das Ende der Zeile und verharrt dann einige Zeit bis zum erneuten Startsignal auf der HSYNC-Leitung. Durch Anlegen eines Low-Signals auf der VSYNC-Leitung zwingt der Prozessor den Strahl an die linke obere Ecke des Bildschirms, wo dieser nach einem Wechsel des VSYNC-Signals auf einen High-Pegel mit der Ausgabe der ersten Zeile beginnt. Die erwähnten drei Steuerleitungen reichen, zusammen mit dem Takt des Grafikcontrollers, zur Erzeugung des Bildes vollständig aus und enthalten gleichzeitig alle Informationen über das Bild selbst. Zur Erstellung einer Hardcopy bedarf es nur einer Auswertung dieser Signale in ihrer zeitlichen Abfolge. Anzumerken sei noch, daß den meisten Monitoren nicht die drei getrennten Signale zur Verfügung gestellt werden, sondern ein sogenanntes BAS-Signal. Dieses Signal stellt aber nur eine Vermischung der drei Einzelsignale dar.

### 3.3. Funktionsweise des Fadenkreuzes

Die Einblendung des Fadenkreuzes geschieht durch Beeinflussung des VIDEO-Signals der GDP-Baugruppe zu bestimmten Zeiten. Zur Ausgabe der vertikalen Linie des Fadenkreuzes verfolgt die HCOPIY-/MAUS-Baugruppe die augenblickliche Position des Elektronenstrahls innerhalb einer Zeile anhand des Taktes des Grafikcontrollers und setzt bei einer vorgegebenen Spalte das VIDEO-Signal auf Low. Durch Wiederholung dieses Vorgangs in allen auszugehenden Zeilen entsteht dann eine vertikale Linie. Die Bestimmung der gewünschten Zeile zur Ausgabe der waagerechten Linie des Fadenkreuzes erfolgt durch Mitzählen der horizontalen Synchronisationssignalen seit dem letzten Start an der linken oberen Ecke des Bildschirms, d.h. seit dem letzten Low-Pegel auf der VSYNC-Leitung. Solange der Elektronenstrahl die gewünschte Zeile durchläuft sorgt die Schaltung für einen Low-Pegel auf der VIDEO-Leitung und damit für eine sichtbare waagerechte Linie. Für den Betrieb eines Monitors mit einem BAS-Signal steht auf der Platine eine eigene Mischstufe zur Vereinigung der einzelnen Steuersignale zur Verfügung.

## 4. Aufbauanleitung

### 4.1. Achtung - MOS!

MOS-Bausteine sind hochempfindlich gegen elektrostatische Aufladung! Bewahren oder transportieren Sie MOS-Bausteine nur auf dem leitenden Schaumstoff (alle Pins müssen kurzgeschlossen sein)!

Tip: Bevor Sie einen Baustein berühren, sollten Sie ein geerdetes Teil (z.B. Heizungsrohr, Wasserleitung oder den Schutzkontakt der Steckdose) kurz berühren.

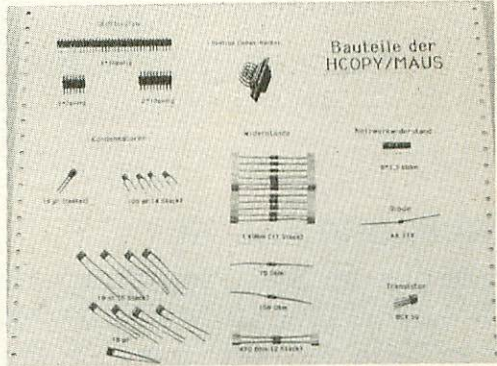
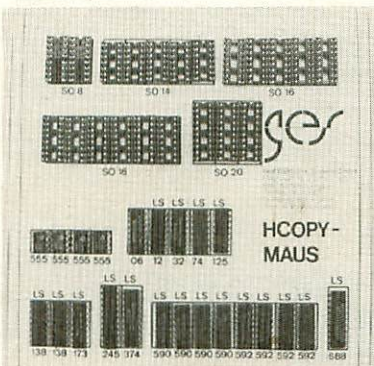
### 4.2. Stückliste

Prüfen Sie zunächst den Bausatz auf Vollständigkeit.

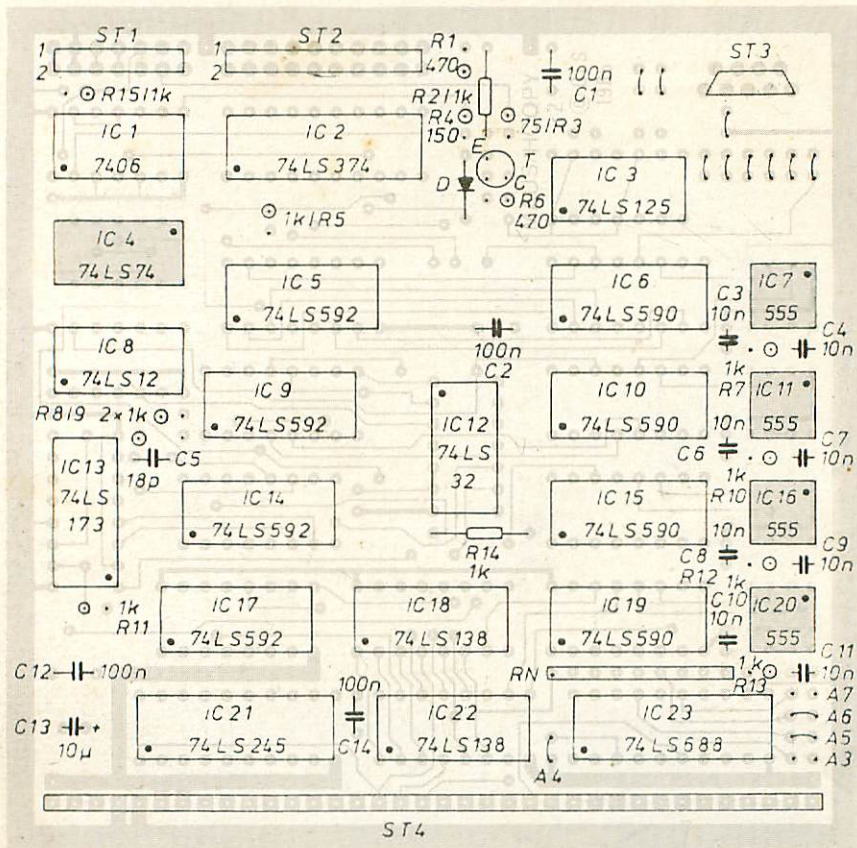
Anzahl	Bezeichnung	
1		Platine
11	R2,R5, R7 - R15	Widerstände 1 kOhm (braun, schwarz, rot)
1	R3	Widerstand 75 Ohm (violett, grün, schwarz)
2	R1,R6	Widerstand 470 Ohm (gelb, violett, braun)
1	R4	Widerstand 150 Ohm (braun, grün, braun)
1	RN	Widerstandnetzwerk 8 x 3.3 kOhm
4	C1,C2, C12,C14	Kondensator 100 nF
8	C3,C4, C6-C11	Kondensator 10 nF
1	C5	Kondensator 18 pF
1	C13	Kondensator 10 uF (Tantal)



1	ST1	Stiftleiste 2 x 7 pol. gew.
1	ST2	Stiftleiste 2 x 10 pol. gew.
1	ST3	Cannon-Buchse 9 pol.
1	ST4	Stiftleiste 1 x 36 pol. gew.
4		Socket 8 polig
5		Socket 14 polig
11		Socket 16 polig
3		Socket 20 polig
1	T	Transistor BCY 59
1	D	Diode AAl18 o.ä.
1	IC1	7406 ✓
1	IC2	74 LS 374 ✓
1	IC3	74 LS 125 ✓
1	IC4	74 LS 74
1	IC8	74 LS 12 ✓
4	IC5, IC9, IC14, IC17	74 LS 592
4	IC6, IC10, IC15, IC19	74 LS 590
4	IC7, IC11, IC16, IC20	NE555 ✓
1	IC12	74 LS 32 ✓
1	IC13	74 LS 173 X
2	IC18, IC22	74 LS 138 ✓
1	IC21	74 LS 245 ✓
1	IC23	74 LS 688 ✓



### 4.3. Bestückungsplan



### ACHTUNG FALLE!

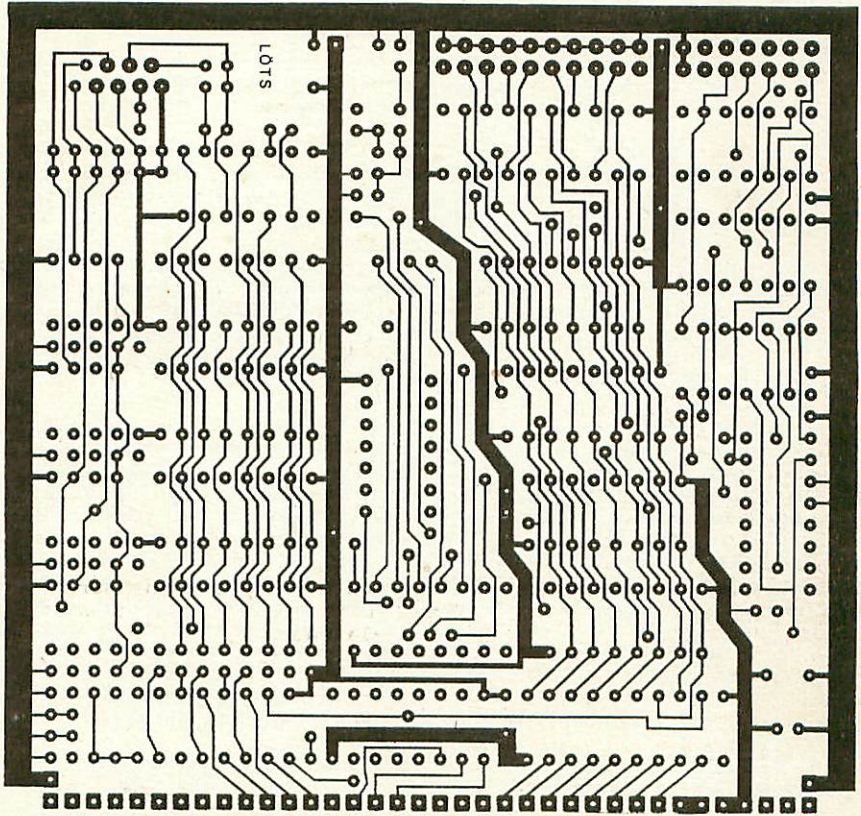
Die hier im Bestückungsplan hervorgehobenen ICs IC4, IC7, IC11, IC16 und IC20 werden entgegen der Richtung der restlichen ICs eingesetzt!

Achten Sie schon beim Einlöten der Sockel darauf, daß die Kerbe, die PIN 1 anzeigt, an der Stelle des Punktes im Bestückungsplan liegt. Prüfen Sie dann bitte vor der Inbetriebnahme nochmals die richtige Lage aller ICs!

Die eingezeichneten "Jumper"-stellungen sind bereits im Layout enthalten und müssen nicht durch Drähte hergestellt werden!



4.4. Layout Bestückungsseite



r2



#### 4.5. Bestückungsanleitung

Auf einer Seite der Platine steht der Hinweis "Löt's" (Lötseite); auf dieser Seite wird ausschließlich gelötet. Die Bauteile sind nur auf der anderen Seite aufzustecken.

Beim Einlöten der Bauelemente beginnt man am besten mit den IC-Sockeln. Dazu bestückt man die Platine zunächst mit allen Sockeln. Dabei muß darauf geachtet werden, daß die Sockeln richtig aufgesteckt werden. Sowohl im Bestückungsplan als auch beim Bestückungsaufdruck auf der Platine sind die Richtungen mit einer Kerbe gekennzeichnet. Diese muß mit der Richtung der Kerbe im IC-Sockel übereinstimmen.

Achtung ! Aus Layouttechnischen Gründen liegt keine einheitliche Orientierung der IC's vor !

Wo welche IC-Fassung hingehört, kann dem Bestückungsplan entnommen werden. Es sollten alle Fassungen auf einmal eingesteckt und dann die Platine zum Verlöten umgedreht werden; dabei ist es hilfreich, wenn man beim Umdrehen die Fassungen mit einem Stück festen Karton auf die Platine drückt. So wird erreicht, daß die Fassungen alle eben und gerade liegen. Beim Löten sollten zunächst nur zwei Pins (möglichst diagonal gegenüberliegend) einer jeden Fassung verlötet werden. Vor dem Anlöten der restlichen Pins sollte man sich durch einen Blick auf die Bestückungsseite von der richtigen Orientierung der Kerben und einer korrekten Auflage der Fassungen auf der Platine vergewissern.

Nach den Sockeln bestückt man am besten die drei abgewinkelten Steckerleisten. Dazu lötet man die Steckerleisten zuerst an jedem Ende an und richtet die Leiste dann parallel zur Leiterplatte. Dabei muß vor allem bei der 36 poligen Busleiste auf sauberes Anliegen der Steckerleiste in der Mitte der Platine geachtet werden. Die Steckerleisten wölben sich gerne in der Mitte von der Platine ab. Das Anlöten einiger Pins in der Mitte verhindert einen solchen "Bauch". Vor dem Einlöten der 9 poligen Cannon Buchse sollten Sie den Stecker und die Anschlußbelegung Ihrer Maus oder Ihres Trackballs konsultieren. Bei einem 9 poligen Stecker mit einer anderen Anschlußbelegung können Sie durch veränderte Brückung der voreingestellten Jumper einen Anschluß ermöglichen. Im Falle eines anderen Steckers sollte die Buchse unbestückt bleiben und der Anschluß durch Drähte erfolgen.

Nun wird der Netzwerkwiderstand RN eingelötet. Ein Netzwerkwiderstand hat an einem Ende einen kleinen weißen Punkt, der manchmal deutlich auf dem Widerstand aufgezeichnet ist, meistens befindet sich der Punkt jedoch relativ undeutlich direkt neben dem Aufdruck. Dieser Punkt markiert den gemeinsamen Anschluß aller Widerstände dieses Netzwerks. Die genaue Lage dieses Pins ist im Bestückungsplan angegeben.

Bei der Bestückung der Widerstände sind bis auf R2 und R14 alle Widerstände stehend einzulöten. Die Zuordnung der Widerstände im Bestückungsplan erfolgt mittels der aufgedruckten Farbringe auf den Widerständen. In der Stückliste befinden sich bei jedem Widerstandswert die Angaben über die zugehörigen Farbkombinationen.

Bei der Bestückung der Diode gilt es die richtige Lage des Bauelements zu beachten. An einem Ende der Diode befindet sich ein



kleiner schwarzer Ring. Die Diode ist so einzusetzen, daß das Dreieck des Schaltzeichens im Bestückungsaufdruck auf diesen Ring zeigt.

Der Elektrolyt-Kondensator C13 ist gepolt und darf auf keinen Fall falsch herum eingelötet werden. Der Pluspol ist mit einem "+" und einem schwarzen Strich gekennzeichnet. Im Bestückungsplan ist der Pluspol ebenfalls mit einem "+" markiert.

Die Kondensatoren C1 bis C12 und C14 sind ungepolt und können, ohne auf die richtige Polung zu achten, eingelötet werden.

Den Abschluß in der Bestückung bildet der Transistor. Bei einem der drei Anschlußdrähte befindet sich eine kleine "Nase" am Gehäuse. Der Transistor ist bei der Bestückung so zu drehen, daß dieser Draht in das mit "E" (Emitter) markierte Loch kommt. Auf Grund der Wärmeempfindlichkeit von Transistoren sollten Sie den Baustein mit etwas Abstand zur Leiterplatte einsetzen und beim Löten nicht zu lange auf den Lötstellen bleiben.

#### 4.6. Einstellungen an der Baugruppe

Für die HCOPIY/MAUS-Baugruppe blieb im Konzept für Ein-/Ausgabe-baugruppen des NDR-Klein-Computers der Adressbereich von 88H bis 8FH reserviert. Entsprechend dieser Adressvorgabe befinden sich auf der Lötseite der Platine Brücken bei den jeweiligen Adress-jumpers der Dekodierungslogik für die Baugruppenfreigabe. Daher muß eine Auftrennung der vorgegebenen Brücken und eine neue Brückung nur dann erfolgen, wenn die Baugruppe in einem anderen Adressbereich arbeiten soll. In diesem Falle läßt sich allerdings eine Anpassung der zugehörigen Software nicht umgehen.

Die voreingestellten Jumper bei der 9 poligen Anschlußbuchse für die Maus oder den Trackball führen zu einer Belegung entsprechend der unten abgebildeten Tabelle. Nur die Verwendung einer Maus mit abweichender Anschlußbelegung würde eine Veränderung der Jumper erfordern. Für den Betrieb der Baugruppe sind keine sonstigen Einstellungen nötig.

Belegung der 9 pol. Buchse

Pin 1	Masse
Pin 2	+ 5 Volt
Pin 3	Masse
Pin 4	Bewegungsrichtung "links"
Pin 5	Bewegungsrichtung "rechts"
Pin 6	frei
Pin 7	event. Schalter der Maus
Pin 8	Bewegungsrichtung "ab"
Pin 9	Bewegungsrichtung "auf"

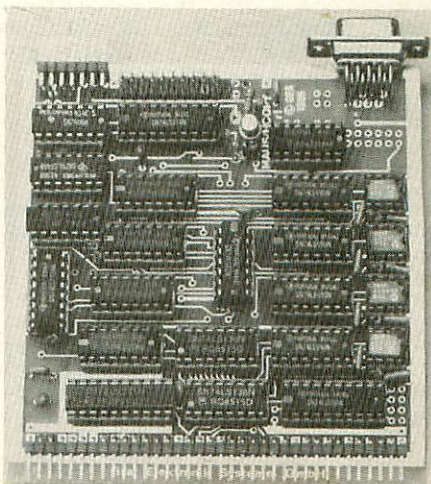


Abb.: Fertig aufgebaute Baugruppe



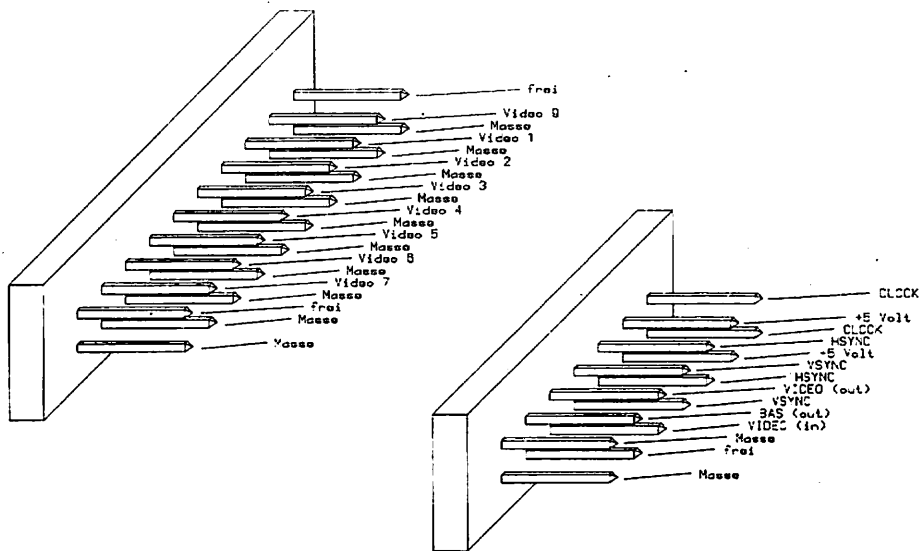
# 5. Test der Baugruppe

## 5.1. Allgemeine Tests

Die Platine ist bis jetzt erst mit den Sockeln und mit den passiven Bauelementen bestückt. Mit diesem Aufbau wird der erste Test durchgeführt. Dazu steckt man die Karte bei ausgeschalteter Stromversorgung in den Bus des funktionstüchtigen Rechners. Nach dem Anlegen der Spannung sollte das System ungestört in der gewohnten Art und Weise arbeiten. Sollte dies nicht der Fall sein, gilt es zunächst die Stromversorgung zu überprüfen. Bei einem Zusammenbruch der +5 Volt Versorgungsspannung liegt der Gedanke an einen Kurzschluß auf der HCOPY/MAUS-Baugruppe sehr nahe. Ein solcher Kurzschluß kann durch einen falsch gepolten Elektrolytkondensator, durch eine Ätzbrücke auf der Platine oder durch eine Lötzinnbrücke zustande kommen. Falls eine Sichtüberprüfung der Platine, mit einem Augenmerk auf die genannten Fälle, keine Abhilfe schafft oder das System trotz einwandfreier Stromversorgung nicht einwandfrei abläuft, sollten Sie Ihre Aufmerksamkeit dem Kapitel "Fehlersuche" zuwenden.

Nach dieser ersten Überprüfung können nun alle IC's eingesetzt werden. Beim Einsetzen der IC's gilt es die richtige Lage der Bausteine zu beachten. Die Markierung auf dem IC (Kerbe an einem Rand oder ein Punkt an einer Ecke) muß mit der Kerbe an der Fassung übereinstimmen. Bevor Sie die Baugruppe in die Busplatine einstecken, sollten Sie die richtige Position und Lage der Bausteine kontrollieren. Seitenverkehrt eingesetzte Bausteine geben bei angelegter Versorgungsspannung meistens in kürzester Zeit ihren Geist auf.

Für die Durchführung der weiteren Funktionstests gilt es nun die Anschlüsse zur Maus (Trackball) und der GDP64K-Baugruppe vorzunehmen. Die Anschlußbelegung der Maus befindet sich im Kapitel 4.4 - die Verbindung der Maus mit den Anschlüssen auf der Platine dürfte keine Schwierigkeit darstellen. Wie bereits zu Anfang des Handbuchs erwähnt, benötigt die Platine wichtige Steuersignale der Grafik-Platine (GDP64K). Wie aus der unten stehenden Zeichnung ersichtlich, weist die 2 x 7 polige Steckerleiste von rechts nach links die gleiche Reihenfolge der Signale auf wie die 7 polige Steckerleiste der GDP-Baugruppe. Bei der zweireihigen Steckerleiste dient die untere Reihe zur Erstellung der Verbindung mit der GDP-Platine und die obere Reihe zum einen zum Abgriff der übernommenen HSYNC- und VSYNC-Signale, sowie zum Abgriff des VIDEO- und BAS-Signals mit eingemischtem Fadenkreuz. Zur Verbindung der 5 Signale (CLOCK, Masse, HSYNC, VSYNC und VIDEO) bedient man sich am zweckmäßigsten eines kleinen 5 poligen Flachbandkabels mit den entsprechenden Buchsen. Dabei sollte die Länge des Kabels möglichst knapp bemessen werden, da bei der Übertragung von Signalen mit einer Frequenz von 14 MHz über längere Kabel die Signalgüte beträchtlich leidet und damit die Funktionssicherheit der Baugruppe beeinträchtigt. Beim Anschluß des Monitors kann der Benutzer dann zwischen einem BAS-Signal ohne Fadenkreuz (Abgriff an der GDP-Baugruppe) oder einem BAS-Signal mit Fadenkreuz (Abgriff an der HCOPY/MAUS-Baugruppe) wählen. Zur Überprüfung der Funktionstüchtigkeit der Platine stellt der Abgriff des BAS-Signals mit Fadenkreuz natürlich eine Voraussetzung dar.



## 5.2. Test des Fadenkreuzes und der Maus-Schnittstelle

Im folgenden gilt es die Baugruppe mit den bisher auf dem Markt erschienenen CPU-Baugruppen (SBC2, CPU Z80 und CPU 68k) zu testen. Falls ein Benutzer mehrere dieser Baugruppen besitzt, so genügt es selbstverständlich den Test der Baugruppe auf eine CPU zu beschränken.

Im Anhang befindet sich für die drei CPU's jeweils ein Programm zur Erprobung der Maus-Schnittstelle und des Fadenkreuzes. Alle drei Programme besitzen dabei die folgende Gemeinsamkeit:

- Nach dem Aufruf des Programms erscheint das Fadenkreuz in der Mitte des Bildschirms.
- Das Fadenkreuz folgt solange den Bewegungen der Maus, bis eine beliebige Taste gedrückt wird.

Sollte die Bewegung des Fadenkreuzes nicht mit der Bewegung der Maus übereinstimmen, so liegt dies entweder an vertauschten Anschlüssen oder an einer anderen Phasenverschiebung der Maus. Mit etwas Probieren läßt sich sicherlich die richtige Zuordnung der Signale erreichen.

### 5.2.1. Test mit der SBC2-Baugruppe

Im Anhang (Punkt 13.1.) finden Sie das Programm zum Test des Fadenkreuzes und der Maus mit der SBC2-Platine. Für die Durchführung des Tests genügt es die dezimalen Maschinenbefehle des abgedruckten Assembler-Listings unter Zuhilfenahme des Grundprogramms einzugeben. Bei fehlerfreier Eingabe sollte nach dem Start des Programms (Adresse 8800H) das Fadenkreuz erscheinen und allen Bewegungen der Maus folgen.



### 5.2.2. Test mit der CPU Z80-Baugruppe

Für den Test mit der Vollausbau CPU Z80 steht unter dem Punkt 13.3. im Anhang ein Programm zur Verfügung. Besitzer des M80-Macro-Assemblers und des CP/M-Betriebssystems können die Quelle des Programms eintippen und daraus ein lauffähiges Objekt (.COM-Datei) erstellen lassen. Für Anwender ohne Betriebssystem oder Assembler bietet sich die direkte Eingabe der sedezimalen Maschinencodes mittels des Grundprogramms an. Dabei sind die mit einem Apostroph markierten Adressen dem benutzen Speicherbereich anzupassen.

### 5.2.3. Test mit der CPU 68k-Baugruppe

Benutzern der 68008-CPU steht entweder im CP/M-Betriebssystem oder im Grundprogramm ein leistungsfähiger Assembler zur Verfügung. Daher wurde das Testprogramm für den 68008 (siehe Anhang Punkt 13.5.) auf diesen Assembler zugeschnitten.

### 5.3. Test der Hardcopy-Funktion

Falls der Test des Fadenkreuzes erfolgreich verlief, und nur dann, sollten Sie sich dem Test der Hardcopy zuwenden. Für jede der drei genannten CPU-Platinen befindet sich im Anhang ein Testprogramm zur Durchführung einer Hardcopy. Jedes dieser Testprogramme beruht auf der Verwendung eines grafikfähigen Matrixdruckers mit einem Befehlssatz entsprechend dem EPSON-Drucker RX80. Bei Verwendung eines anderen Druckers bedarf es einer Anpassung dieser Programme. Die Ausgabe der Daten für den Drucker erfolgt über eine IOE-Platine mit der CENT-Zusatzkarte.

#### 5.3.1. Test mit der SBC2-Baugruppe

Für den Test mit dem Single-Board-Computer gibt man von dem Programm, unter 13.2. im Anhang, nur die sedezimalen Maschinenbefehle mit Hilfe des Grundprogramms ab der Adresse 8800H ein. Der Aufruf des Programms sollte nicht durch das Grundprogramm erfolgen, da dieses beim Start des Programms den Bildschirm löscht und damit eine Hardcopy sinnlos wird. Am besten startet man ein kleines Programm mit Bildschirmausgabe und springt am Ende dieses Programms an die Adresse 8800H.

#### 5.3.2. Test mit der CPU Z80-Baugruppe

Besitzer des M80-Macro-Assemblers und des CP/M-Betriebssystems können die Quelle des Programm unter Punkt 13.4. des Anhangs eintippen und daraus ein lauffähiges Objekt (.COM-Datei) erstellen lassen. Für Anwender ohne Betriebssystem oder Assembler bietet sich die direkte Eingabe der sedezimalen Maschinencodes mittels des Grundprogramms an. Dabei sind die mit einem Apostroph markierten Adressen dem benutzen Speicherbereich anzupassen.

#### 5.3.3. Test mit der CPU68k-Baugruppe

Unter dem Punkt 13.6. des Anhangs befindet sich ein Testprogramm für den 68008-Prozessor. Die Umsetzung des Programms in die entsprechenden Maschinenbefehle kann mittels des Assemblers des Grundprogramms oder auf der CP/M System-Diskette erfolgen.



## 6. Fehlersuche

Sollte Ihre HCOPY/MAUS-Baugruppe bei den im Kapitel "Test der Baugruppe" beschriebenen Tests nicht funktionieren, so heißt es jetzt auf systematische Fehlersuche zu gehen. Wir wollen Ihnen nun ein paar Vorschläge machen, wie eine systematische Fehlersuche vor sich gehen kann:

- Sind die bisher verwendeten Baugruppen in Ordnung ?  
( Funktioniert das System ohne HCOPY/MAUS-Baugruppe ? )
- Sind die Jumpers richtig gesteckt ?

### 6.1. Sichtprüfung

1. Machen Sie zunächst eine Sichtprobe. Können Sie irgendwo auf der Platine unsaubere Lötstellen (zuviel Lötzinn, manchmal zieht das Lötzinn auch Fäden) erkennen, die eventuell einen Kurzschluß verursachen könnten? Dann müssen Sie diese Lötstellen nachlöten und jede unzulässige Verbindung beseitigen.

2. Haben Sie alle IC's richtig herum am richtigen Platz aufgesteckt? Manchmal können beim Einstecken der IC's einzelne Pins weggebogen sein. Da Sie dies durch reine Sichtkontrolle oft nicht erkennen können, sollten Sie jeden Baustein noch einmal herausziehen, kontrollieren und dann erneut einsetzen.

3. Haben Sie den gepolten Kondensator auch richtig eingesetzt?

4. Ist der Netzwerkwiderstand richtig eingelötet ?

5. Haben Sie auch keine Lötstelle vergessen ? (Sehen Sie lieber noch einmal nach.)

6. Sehen Sie irgendwo "kalte" Lötstellen ?  
Kalte Lötstellen erkennt man daran, daß sie nicht glänzen. Sie sind im Vergleich mit richtig gelöteten Lötstellen trübe.

7. Haben Sie auch nicht zu heiß gelötet ?  
Wenn der LötKolben zu heiß eingestellt ist und (oder) Sie zu lange auf der Lötstelle bleiben, dann kann es passieren, daß sich die Leiterbahnen von der Platine lösen, daß Durchkontaktierungen unterbrochen werden oder ,daß Bauteile durch zu heißes Löten zerstört werden.

Sollten Sie nach der Sichtprüfung noch keinen Fehler entdeckt haben, so wird es notwendig, daß man ein Meßgerät (Multimeter, Prüftstift, Oszilloskop etc.) zur Hand nimmt.

### 6.2. Messungen

Nehmen Sie alle IC's aus ihren Fassungen. Nehmen Sie sich die Layouts zur Hand und kontrollieren Sie alle Leiterbahnen mit einem Durchgangsprüfer oder einem Ohmmeter auf Durchgang. Bereits kontrollierte Bahnen können Sie auf dem Layout mit Bleistift markieren.

Wenn Sie alle Leiterbahnen kontrolliert und nichts gefunden haben, dann ist die Wahrscheinlichkeit groß, daß ein Bauteil defekt ist.

Wenn Sie einen Prüftstift oder ein Oszilloskop haben, dann können Sie jetzt überprüfen, ob Sie an den jeweiligen Ausgängen die richtigen Signale haben. Welche Signale wo anliegen müssen können Sie der Schaltungsbeschreibung und dem Schaltplan entnehmen.

Falls Sie keine Meßgeräte haben, dann müssen Sie alle Bausteine systematisch austauschen, bis sie das defekte Teil gefunden haben.

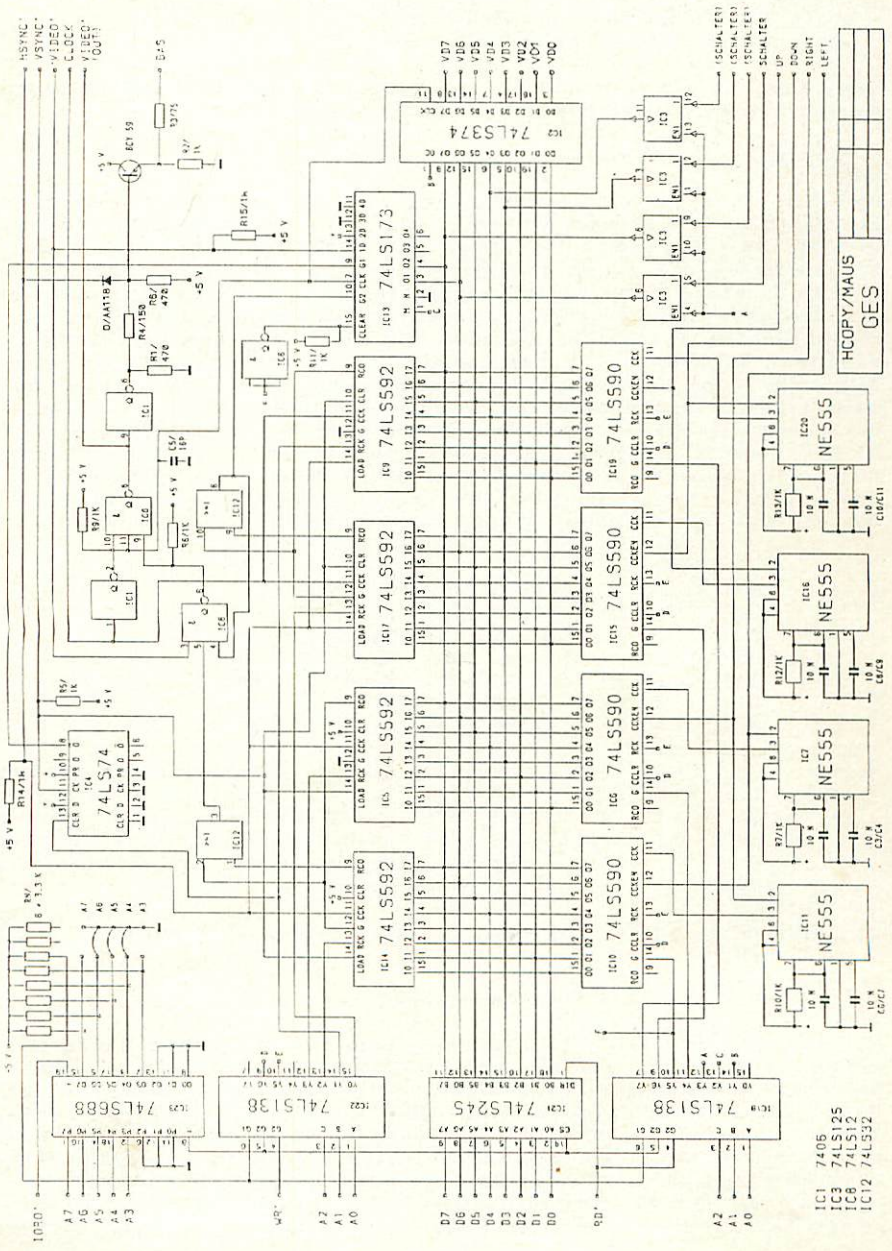
Verwenden Sie dazu eventuell eine zweite Baugruppe (die eines Freundes oder eines Bekannten).

Sollten Sie gar nicht zu Rande kommen, hilft Ihnen unser Pauschal-Reparatur-Service, dessen Bedingungen Sie der Preisliste entnehmen können.



# 7. Schaltungsbeschreibung

## 7.1. Schaltplan





## 7.2. Beschreibung der Schaltung

Die Schaltung der HCOPY/MAUS-Baugruppe läßt sich sinnvollerweise in drei logische Teile zerlegen:

- Adress- und Dekodierlogik,
- Logik zur Ansteuerung der Maus und
- Logik für Hardcopy und Fadenkreuz.

Zum besseren Verständnis der folgenden Schaltungsbeschreibung nehmen Sie am besten das Schaltbild zu Hilfe.

### 7.2.1. Adress- und Dekodierlogik

Der 8-Bit Komperator 74LS688 vergleicht die Signale der Adressleitungen A3 bis A7 vom Systembus mit der, durch Brücken voreingestellten, Adresse der Baugruppe. Bei Übereinstimmung der Adresssignale und einem gleichzeitigen Low-Pegel auf der IORQ-Leitung liefert der Vergleicherbaustein ein Low-Signal am Pin 8 zur Freigabe der Baugruppe. Dieses Freigabesignal führt zum bidirektionalen Bustreiber 74LS245 und zu den beiden Dekodierbausteinen 74LS138. Bei einem Lese- oder Schreibzugriff auf die Baugruppe wird in Abhängigkeit des RD- bzw. WR-Signals einer der beiden Dekodierbausteine freigegeben. Entsprechend der zu diesem Zeitpunkt anliegenden Signale auf den drei Adressleitungen A0 bis A2 führt einer der 8 Ausgänge des jeweiligen Dekodierbausteins einen Low-Pegel. Dieses Low-Signal dient dann zur Selektion einer der weiteren Funktionseinheiten der Baugruppe. Die folgende Tabelle zeigt die Adressen dieser Funktionseinheiten auf.

Read:	89H	IC 2	(74 LS 374)	8-Bit Port
	8AH	IC 13	(74 LS 173)	Ready-Flag und Daten-Bit
	8BH	IC 3	(74 LS 125)	Tasten der Maus
	8CH	IC 6	(74 LS 590)	Maus "auf"
	8DH	IC 10	(74 LS 590)	Maus "ab" und Clear IC 13
	8EH	IC 15	(74 LS 590)	Maus "rechts"
	8FH	IC 19	(74 LS 590)	Maus "links"
Write:	88H	IC 17	(74 LS 592)	Hi-Byte X-Position der Hardcopy
	89H	IC 9	(74 LS 592)	Lo-Byte X-Position der Hardcopy
	8AH	IC 14	(74 LS 592)	Hi-Byte Y-Position der Hardcopy
	8BH	IC 5	(74 LS 592)	Lo-Byte Y-Position der Hardcopy
	8DH	(alle 74 LS 590)		Speichern der Zählerstände
	8EH	(alle 74 LS 590)		Löschen der Zähler

Diese Daten gelten bei der voreingestellten Startadresse von 88H

### 7.2.2. Logik zur Aussteuerung der Maus

Wie bereits erwähnt, besitzt die Maus für die horizontale und vertikale Bewegung jeweils zwei Ausgänge, die zueinander phasenverschobene Rechtecksignale liefern. Die vier Ausgangssignale der Maus führen zum Triggereingang des Timerbausteins NE555. Entsprechend seiner externen Beschaltung arbeitet der Baustein als monostabiler Impulsgeber. Jede negative Flanke am Triggereingang bewirkt einen kurzen positiven Impuls am Ausgang. Diese Ausgangssignale dienen als Takt für die 8 Bit Zähler 74LS590. Da bei einer Aufwärtsbewegung der Maus Impulse sowohl am Ausgang für



Auf- als auch für Abwärtsbewegung erscheinen, gilt es anhand der Phasenverschiebung einen der beiden Zähler zu sperren. Durch Zuführung der paarweise vertauschten Rechtecksignale auf die Freigabeeingänge des Taktes wird entsprechend der Bewegung nur ein Zähler freigegeben. Zur Erfassung der Bewegung der Maus müssen die Inhalte der vier Zähler regelmäßig abgelesen werden. Da die Zählerbausteine mit Ausgaberegistern arbeiten, bedarf es vor dem Auslesen der Zählerinhalte eines Schreibzugriffs (beliebiges Datenwort) auf die Adresse 8DH, um eine Übernahme des aktuellen Zählerstandes in das Ausgaberegister zu veranlassen. Der Zugriff auf die Adresse 8DH bewirkt gleichzeitig eine Speicherung der Zählerstände bei allen vier Zählern. Ebenso bewirkt ein Schreibzugriff (beliebiges Datenwort) auf die Adresse 8EH ein gleichzeitiges Löschen aller vier Zähler. Über den Baustein 74LS125 besteht die Möglichkeit, den Zustand von maximal vier Tasten der Maus abzufragen. Bei der standardmäßigen Belegung führt allerdings nur ein Eingang zur 9 poligen Buchse zum Anschluß der Maus.

### 7.2.3. Logik für Hardcopy und Fadenkreuz

Über die Zähler IC9 und IC17 (74LS592) verfolgt die Baugruppe die augenblickliche Position des Elektronenstrahls innerhalb einer Zeile durch Aufsummierung der Taktimpulse der GDP-Platine. Ebenso berechnen die Zähler IC5 und IC14 (74LS592) die Nummer der augenblicklich durch den Elektronenstrahl beschriebenen Zeile anhand des horizontalen Synchronisationssignals. Nun gilt es aber nicht nur den Lauf des Elektronenstrahl zu verfolgen, sondern es besteht die Notwendigkeit beim Erreichen einer bestimmten Zeile oder Spalte ein Signal zu erhalten. Möchte man nun das Fadenkreuz beispielsweise in einer bestimmten Spalte einblenden, so müssen die Eingangsregister der Zähler IC9 und IC17 mit dem komplementierten Wert dieser Spalte geladen werden. Zu Beginn einer neuen Zeile (Low-Signal auf HSYNC-Leitung) übernehmen die Zähler die vorgegebenen Werte im Eingaberegister als neuen Zählerstand. Durch jeden Taktimpuls auf der Clock-Leitung erhöht sich der Wert des 16-Bit Zählers, bestehend aus IC9 und IC17, um Eins. Beim Erreichen des Zählerstands FFFFH führt der Übertragsausgang (Pin 9) der beiden Zähler einen Low-Impuls, und damit erscheint auch ein Low-Signal am Pin 6 des ODER-Gatters von IC12. Ein Low-Signal am Ausgang dieses Bausteins zeigt also das Erreichen einer bestimmten Spalte an. Ebenso zeigt ein Low-Signal am Pin 3 des gleichen Bausteins (IC12) das Durchlaufen einer bestimmten Zeile des Bildschirms an. Bei der Einblendung des Fadenkreuzes werden nun bei einem Low-Pegel auf Pin 3 oder Pin 6 das ursprüngliche VIDEO-Signal der GDP-Baugruppe zwangsweise auf Low gesetzt und damit eine sichtbare Linie erzeugt. Am Ausgang des Pin 6 des 74LS12 läßt sich dieses VIDEO-Signal mit Fadenkreuz abgreifen. Die weiteren Gatter bzw. Widerstände und der Transistor dienen nur zur Erzeugung eines BAS-Signals für den Monitor.

Bei der Erstellung einer Hardcopy werden von rechts nach links alle Spalten des Bildschirms nacheinander abgetastet. Bei der Abtastung einer Spalte besteht natürlich die Notwendigkeit mit dem obersten Punkt einer Spalte zu beginnen. Um diese Synchronisation zu erreichen, wird durch einen Schreibzugriff auf die Adresse 89H das Flip-Flop des 74LS74 gelöscht. Eine positive Flanke auf der VSYNC-Leitung setzt das Flip-Flop, und das daraus resultierende Low-Signal am Pin 8 dient als Startsignal für die Abtastung der Spalte. Von nun ab wird bei jedem Durchlauf des Elektronenstrahls durch eine vorgegebene Spalte der augenblickliche Wert auf der VIDEO-Leitung (Bildpunkt) im 74LS173 bzw. der

augenblickliche Wert der 8 Eingangsleitungen des 74LS374 gespeichert. Damit das System nun erkennen kann, wann ein neuer Bildpunkt vorliegt, werden durch einen Lesezugriff auf die Adresse 8DH alle Bits des IC13 gelöscht. Bei der Einspeicherung eines neuen Bildpunktes wird gleichzeitig ein Bit des 74LS173 auf High gesetzt und dient damit als Meldeflag für einen gespeicherten Bildpunkt. Durch aufeinander folgendes Einlesen der 256 Punkte einer Spalte entsteht das Bitmuster einer Spalte.

## **8. Anwendungsbeispiele**

Der Sinn einer Hardcopy-Schaltung liegt darin, jederzeit den augenblicklichen Bildschirminhalt auf Papier bringen zu können. Bei den bisher beschriebenen Routinen zur Erzeugung einer Hardcopy muß das entsprechende Maschinenprogramm in das jeweils ablaufende Programm eingebunden werden. Da man bei kommerziell verwerteten Programmen praktisch nie die Quellen erhält, stellt dies also eine nicht tragbare Lösung dar.

Im Anhang findet sich unter Punkt 13.7. ein Programm, das dieses Problem umgeht. Dazu wird das eigentliche Hardcopy-Programm in einem freien Speicherbereich des Monitors abgelegt. Gleichzeitig wird die Programmschnittstelle zur Tastatur derartig abgewandelt, daß eine gleichzeitige Betätigung der Control Taste und der Taste mit dem Klammeraffen (auf manchen Tastaturen entspricht dies dem Paragraph-Zeichen) die Auslösung einer Hardcopy bewirkt. Dadurch kann zu jedem Zeitpunkt, an dem ein ablaufendes Programm eine Eingabe zuläßt, eine Hardcopy veranlaßt werden.

Hinweis: Da im freien Pufferbereich verschiedene Routinen zur Unterstützung von anderen Baugruppen (z.B. SER) abgelegt werden, kann eine Überschneidung dieser Hilfsprogramme stattfinden. In diesem Falle bedarf es einer Neuübersetzung des Programms mit einer neuen Startadresse.



# 9. Diverses

## 9.1. Testprogramm für Fadenkreuz und Maus (SBC2)

```
0000'                                     .ZB0
                                           cseq

                                           ;
                                           ; *****
                                           ; Programm zum Test der Hardcopy/Maus-Baugruppe
                                           ; am NDR-Klein-Computer mit der CPU-Baugruppe SBC2.
                                           ;
                                           ; (C) G. Sternberg 1985                Stand 18.4.1985
                                           ; *****
                                           ;
                                           org 8800h

8800' 21 FF80          ld hl,0ff80h          ; Y-Startposition des Fadenkreuzes
8803' 22 8874'       ld (Y),hl
8806' 21 FE40          ld hl,0fe40h          ; X-Startposition des Fadenkreuzes
8809' 22 8876'       ld (X),hl
880C' D3 8D          Loop: out (8dh),a        ; Zaehler speichern
880E' D3 8E          out (8eh),a        ; Zaehler loeschen
8810' 2A 8874'       ld hl,(Y)          ; Adresse fuer X-Position
8813' DB 8C          in a,(8ch)         ; Impulse fuer Aufwaerts
8815' FE 00          cp 0                ; keine Aufwaertsbewegung ?
8817' CA 881F'       jp z,Label1
881A' 23            Label2: inc hl          ; Position veraendern
881B' 3D            dec a
881C' C2 881A'       jp nz,Label2
881F' DB 8D          Label1: in a,(8dh)     ; Impulse fuer Abwaerts
8821' FE 00          cp 0                ; keine Abwaertsbewegung ?
8823' CA 882B'       jp z,Label3
8826' 2B            Label4: dec hl          ; Position veraendern
8827' 3D            dec a
8828' C2 8826'       jp nz,Label4
882E' 22 8874'       Label3: ld (Y),hl
882E' 2A 8876'       ld hl,(X)          ; Adresse fuer Y-Position
8831' DB BF          in a,(8fh)         ; Impulse fuer Rechts
8833' FE 00          cp 0                ; keine Rechtsbewegung ?
8835' CA 883D'       jp z,Label5
8838' 23            Label6: inc hl          ; Position veraendern
8839' 3D            dec a
883A' C2 8838'       jp nz,Label6
883D' DB 8E          Label5: in a,(8eh)     ; Impulse fuer Links
883F' FE 00          cp 0                ; keine Linksbewegung ?
8841' CA 8849'       jp z,Label7
8844' 2B            Label8: dec hl          ; Position veraendern
8845' 3D            dec a
8846' C2 8844'       jp nz,Label8

8849' 22 8876'       Label7: ld (X),hl
884C' 3A 8876'       ld a,(X)
884F' D3 89          out (89h),a        ; Fadenkreuz X-Position (lo)
8851' 3A 8877'       ld a,(X+1)
8854' D3 88          out (88h),a        ; Fadenkreuz X-Position (hi)
8856' 3A 8874'       ld a,(Y)
8859' D3 8B          out (8bh),a        ; Fadenkreuz Y-Position (lo)
885B' 3A 8875'       ld a,(Y+1)
```

```

885E' D3 8A          out (8ah),a          ; Fadenkreuz Y-Position (hi)
8860' DB 68          in a,(68h)          ; Tastatur-Port
8862' E6 80          and 10000000b      ; Ready-Flag gesetzt ?
8864' C2 8800       jp nz,Loop
8867' 3E 00          ld a,0              ; Fadenkreuz ausschalten
8869' D3 BB          out (88h),a
886B' D3 B9          out (89h),a
886D' D3 BA          out (8ah),a
886F' D3 BB          out (8bh),a
8871' C3 0000       jp 0                ; Reset
8874' 0000          Y: dw 0                ; Y-Position des Fadenkreuzes
8876' 0000          X: dw 0                ; X-Position des Fadenkreuzes
end

```



## 9.2. Testprogramm für Hardcopy (SBC2)

```

                                .Z80
0000'                            cseg
                                ;
                                ; *****
                                ; Programm zum Test der Hardcopy/Maus-Baugruppe
                                ; am NDR-Klein-Computer mit der CPU-Baugruppe SBC2.
                                ;
                                ; (C) G. Sternberg 1985                Stand 18.4.1985
                                ; *****
                                ;
0048'                            centdaten equ 48h                ; Daten 0 .. 7
0049'                            centin   equ 49h                ; Bit 0 ist Busy-Status 1=Busy
0049'                            centstb  equ 49h                ; Bit 0 ist -Strobe 0=Puls

                                org 8800h

8800' 31 89D6'                    ld sp,Stack                ; Stackpointer laden
8803' 21 02D0'                    ld hl,720                  ; Nummr der letzten Bildspalte
8806' 22 88C0'                    ld (Row),hl                ; Nummer abspeichern
8809' 3E 01'                      ld a,1                    ; IDE+CENT initialisieren
880B' D3 49'                      out (centstb),a            ; Strobe Bit loeschen
880D' CD 882E'                    Call InitRX80           ; Initialisierung des Druckers
8810' 3E 43'                      ld a,67                  ; Zahl der Druckzeilen
                                ; (Punkte pro Zeile / 8)

8812' F5'                          Loop1: push af                ; Wiederholungsfaktor sichern
8813' CD 885F'                    Call GetLine          ; eine Druckzeile abtasten
8816' CD 8834'                    Call InitLine         ; Drucker fuer die Ausgabe
                                ; dieser Zeile initialisieren
8819' CD 8895'                    Call PrtLine          ; Zeilenpuffer ausgeben
881C' F1'                          pop af                ; Wiederholungsfaktor laden
881D' 3D'                          dec a                ; weitere Druckzeile ?
881E' C2 8812'                    jp nz,Loop1

8821' 3E 00'                      ld a,0                ; Fadenkreuz ausschalten
8823' D3 88'                      out (88h),a
8825' D3 89'                      out (89h),a
8827' D3 8A'                      out (8ah),a
8829' D3 8B'                      out (8bh),a

882B' C3 0000'                    jp 0                    ; Reset

                                ; Initialisierung des Druckers

882E' 21 8843'                    InitRX80: ld hl,InitTab        ; Adresse der Tabelle mit den
                                ; Daten zur Initialisierung
8831' C3 8837'                    jp Loop
8834' 21 884E'                    InitLine: ld hl,InitTab1      ; Initialisierung fuer eine Zeile
8837' 7E'                          Loop: ld a,(hl)          ; Byte laden
883B' CD 88AF'                    call Out             ; Byte ausgeben
883B' 23'                          inc hl              ; Adresse erhoehen
883C' 3E FF'                      ld a,0ffh           ; Akku loeschen
883E' BE'                          cp (hl)             ; Ende-Marke ?
883F' C2 8837'                    jp nz,Loop          ; naechstes Byte ausgeben
8842' C9'                          ret                    ; Ende der Initialisierung

```

```

8843' 1B 40      InitTab: db 1bh, 5      ; Drucker normieren
8845' 0D 0A 0A 0A db 0dh,0ah,0ah,0ah,0ah ; oberer Rand
8849' 0A
884A' 1B 33 18   db 1bh, '3 ,24   ; Zeilabstand 24/216 Zoll
884D' FF         db 0ffh      ; Offh als Ende-Marke
884E' 20 20 20 20 InitTab1: db '      ; linker Rand
8852' 20 20 20 20
8856' 20 20 20 20
885A' 1B 4B 00 01 db 1bh, 'K ',0,1   ; Bitmuster mit doppelter Dichte
885E' FF         db 0ffh      ; Offh als Ende-Marke

; 8 Spalten des Bildschirm abtasten

885F' 06 0B      GetLine: ld b,b      ; Zahl der Spalten pro Druckzeile
8861' 0E 00      Loop3:   ld c,0      ; Zahl der Punkte pro Spalte
8863' 3E FE      ld a,0feh   ; Zeilennummer laden
8865' D3 8B      out (0bh),a
8867' 3E FF      ld a,0ffh   ;
8869' D3 8A      out (8ah),a
886B' 2A 8BC0'   ld nl,(Row) ; Nummer der vorherigen Spalte
886E' 2B         dec hl      ; Spaltenzaehler erniedrigen
886F' 22 8BC0'   ld (Row),hl ; Spaltenzaehler wieder sichern
8872' 7C         ld a,h      ; Spaltennummer laden
8873' 2F         cpl          ; Byte komplementieren
8874' D3 8B      out (8Bh),a
8876' 7D         ld a,l
8877' 2F         cpl          ; Byte komplementieren
8878' 21 8BC2'   ld hl,Buffer ; Adresse des Zeilenpuffers
887B' D3 89      out (89h),a
887D' DB 8D      in a,(8dh)  ; Flag des SN 74173 loeschen

887F' DB 8A      Loop4:   in a,(8ah)  ; Warten bis Punkt gefunden
8881' 07         rca          ; Bit 7 ins CY-Flag
8882' D2 8B7F'   jp nc,Loop4 ; Punkt noch nicht gefunden
8885' 07         rca          ; Bit des Punkts in CY-Flag
8886' 7E         ld a,(hl)   ; bisheriges Byte laden
8887' 17         rla          ; Bit zum alten Byte hinzufuegen
8888' 77         ld (hl),a   ; Byte wieder abspeichern
8889' DB 8D      in a,(8dh)  ; Flag des SB 74173 loeschen
888B' 23         inc hl      ; naechste Stelle im Puffer
888C' 0D         dec c        ; weiterer Punkt ?
888D' C2 8B7F'   jp nz,Loop4

8890' 05         dec b        ; weitere Spalte ?
8891' C2 8B61'   jp nz,Loop3
8894' C9         ret

; Ausgabe des Zeilenpuffers

8895' 21 8BC2'   PrtLine: ld hl,Buffer ; Adresse des Zeilenpuffers
8898' 0E 00      ld c,0      ; Zahl der Speicherstellen
889A' 7E         Loop2:   ld a,(hl)  ; Speicherstelle laden
889B' 2F         cpl          ; Akku negieren
889C' CD 8BAF'   call Out   ; Byte ausgeben
889F' 23         inc hl      ; naechste Speicherstelle
88A0' 0D         dec c        ; Zahl der restl. Stellen
88A1' C2 8B9A'   jp nz,Loop2
88A4' 3E 0D      ld a,0dh   ; Carriage Return ausgeben
88A6' CD 8BAF'   call Out   ; Byte ausgeben
88A9' 3E 0A      ld a,0ah   ; Line Feed ausgeben
88AB' CD 8BAF'   call Out   ; Byte ausgeben

```



```

88AE' C9          ret

; Ausgabe eines Zeichens auf IDE+CENT

88AF' F5          Out:  push af          ; A-Register sichern
88B0' DB 49       Out1: in a,(centstb)   ; Busy-Flag
88B2' 0F          rrca          ; Bit 0
88B3' 3B FB       jr c,Out1     ; Drucker Ready ?
88B5' F1          pop af         ; Akku restaurieren
88B6' D3 48       out (centdaten),a ; Byte ausgeben
88B8' AF          xor a          ; Akku loeschen
88B9' D3 49       out (centstb),a  ; Strobe Pulse
88BE' 3E 01       ld a,1         ; Strobe Bit auf 1 setzen
88BD' D3 49       out (centstb),a  ; Pulse beenden
88BF' C9          ret

88C0' 0000       Row:   dw 0          ; Numer der naechsten Spalte
88C2'           Buffer: ds 256       ; Puffer fuer eine Zeile
89C2'           ds 20          ; Platz fuer Stack
89D6' 00          Stack: db 0

end

```

### 9.3. Testprogramm für Fadenkreuz und Maus (CPU Z80)

```

                                .Z80
0000'                                cseg

                                ;
                                ; *****
                                ; Programm zum Test der Hardcopy/Maus-Baugruppe
                                ; am NDR-Klein-Computer mit der Baugruppe CPU Z80.
                                ;
                                ; (C) G. Sternberg 1985                Stand 18.4.1985
                                ; *****
                                ;

0000' 21 FF80'                ld hl,0ff80h                ; Y-Startposition des Fadenkreuzes
0003' 22 0074'                ld (Y),hl
0006' 21 FE40'                ld hl,0fe40h                ; X-Startposition des Fadenkreuzes
0009' 22 0076'                ld (X),hl
000C' D3 8D'                Loop: out (8dh),a                ; Zaehler latchen
000E' D3 BE'                out (8eh),a                ; Zaehler loeschen
0010' 2A 0074'                ld hl,(Y)                ; Adresse fuer X-Position
0013' DB BC'                in a,(8ch)                ; Impulse fuer Aufwaerts
0015' FE 00'                cp 0                ; keine Aufwaertsbewegung ?
0017' CA 001F'                jp z,Label1
001A' 23'                Label2: inc hl                ; Position veraendern
001B' 3D'                dec a
001C' C2 001A'                jp nz,Label2
001F' DB 8D'                Label1: in a,(8dh)                ; Impulse fuer Abwaerts
0021' FE 00'                cp 0                ; keine Abwaertsbewegung ?
0023' CA 002B'                jp z,Label3
0026' 2B'                Label4: dec hl                ; Position veraendern
0027' 3D'                dec a
002B' C2 0026'                jp nz,Label4
002B' 22 0074'                Label3: ld (Y),hl
002E' 2A 0076'                ld hl,(X)                ; Adresse fuer Y-Position
0031' DB 8F'                in a,(8fh)                ; Impulse fuer Rechts
0033' FE 00'                cp 0                ; keine Rechtsbewegung ?
0035' CA 003D'                jp z,Label5
003B' 23'                Label6: inc hl                ; Position veraendern
0039' 3D'                dec a
003A' C2 003B'                jp nz,Label6
003D' DB 8E'                Label5: in a,(8eh)                ; Impulse fuer Links
003F' FE 00'                cp 0                ; keine Linksbewegung ?
0041' CA 0049'                jp z,Label7
0044' 2B'                Label8: dec hl                ; Position veraendern
0045' 3D'                dec a
0046' C2 0044'                jp nz,Label8

0049' 22 0076'                Label7: ld (X),hl
004C' 3A 0076'                ld a,(X)
004F' D3 89'                out (89h),a                ; Fadenkreuz X-Position (lo)
0051' 3A 0077'                ld a,(X+1)
0054' D3 88'                out (88h),a                ; Fadenkreuz X-Position (hi)
0056' 3A 0074'                ld a,(Y)
0059' D3 8B'                out (8bh),a                ; Fadenkreuz Y-Position (lo)
005B' 3A 0075'                ld a,(Y+1)
005E' D3 8A'                out (8ah),a                ; Fadenkreuz Y-Position (hi)

0060' DB 68'                in a,(68h)                ; Tastatur-Port
0062' E6 80'                and 10000000b            ; Ready-Flag gesetzt ?

```



```

0064 C2 0000      jp nz,Loop
0067 3E 00      ld a,0          ; Fadenkreuz ausschalten
0069 D3 88      out (88h),a
006B D3 89      out (89h),a
006D D3 8A      out (8ah),a
006F D3 8B      out (8bh),a

0071 C3 0000      jp 0           ; Reset

0074 0000      Y: dw 0        ; Y-Position des Fadenkreuzes
0076 0000      X: dw 0        ; X-Position des Fadenkreuzes

end

```

#### 9.4. Testprogramm für Hardcopy (CPU Z80)

```

                                .Z80
0000'                          cseg
                                ;
                                ; *****
                                ; Programm zum Test der Hardcopy/Maus-Baugruppe
                                ; am NDR-Klein-Computer mit der Baugruppe CPU Z80.
                                ;
                                ; (C) G. Sternberg 1985                Stand 18.4.1985
                                ; *****
                                ;
0048'                          centdaten equ 48h          ; Daten 0 .. 7
0049'                          centin    equ 49h          ; Bit 0 ist Busy-Status 1=Busy
0049'                          centstb   equ 49h          ; Bit 0 ist -Strobe 0=Puls

0000' 31 01D6'                  ld sp,Stack              ; Stackpointer laden
0003' 21 02D0                    ld hl,720              ; Nummr der letzten Bildspalte
0006' 22 00C0'                  ld (Row),hl            ; Nummer abspeichern
0009' 3E 01                      ld a,l            ; IQE+CENT initialisieren
000B' D3 49                      out (centstb),a      ; Strobe Bit loeschen
000D' CD 002E'                  call InitRX80     ; Initialisierung des Druckers
0010' 3E 43                      ld a,67          ; Zahl der Druckzeilen
                                ; (Punkte pro Zeile / B)

0012' F5                          Loop1: push af          ; Wiederholungsfaktor sichern
0013' CD 005F'                  call GetLine     ; eine Druckzeile abtasten
0016' CD 0034'                  call InitLine    ; Drucker fue die Ausgabe
                                ; dieser Zeile initialisieren
0019' CD 0095'                  call PrtLine     ; Zeilenpuffer ausgeben
001C' F1                          pop af          ; Wiederholungsfaktor laden
001D' 3D                          dec a           ; weitere Druckzeile ?
001E' C2 0012'                  jp nz,Loop1

0021' 3E 00                      ld a,0          ; Fadenkreuz ausschalten
0023' D3 88                      out (88h),a
0025' D3 89                      out (89h),a
0027' D3 8A                      out (8ah),a
0029' D3 8B                      out (8bh),a

002B' C3 0000                    jp 0            ; Reset
                                ; Initialisierung des Druckers

002E' 21 0043'                  InitRX80: ld hl,InitTab ; Adresse der Tabelle mit den
                                ; Daten zur Initialisierung
0031' C3 0037'                  jp Loop
0034' 21 004E'                  InitLine: ld hl,InitTabl ; Initialisierung fuer eine Zeile
0037' 7E                          Loop:   ld a,(hl)          ; Byte laden
0038' CD 00AF'                  call Out        ; Byte ausgeben
003B' 23                          inc hl         ; Adresse erhoehen
003C' 3E FF                      ld a,0ffh     ; Akku loeschen
003E' BE                          cp (hl)        ; Ende-Marke ?
003F' C2 0037'                  jp nz,Loop     ; naechstes Byte ausgeben
0042' C9                          ret            ; Ende der Initialisierung

0043' 1B 40                      InitTab: db 1bh,5'      ; Drucker normieren
0045' 0D 0A 0A 0A                db 0dh,0ah,0ah,0ah,0ah ; oberer Rand

```



```

0049' 0A
004A' 1B 33 1B          db 1bh, '3', 24      ; 24/216 Zoll Zeilabstand
004D' FF              db 0ffh          ; 0ffh als Ende-Marke
004E' 20 20 20 20 InitTab1: db          ; linker Rand
0052' 20 20 20 20
0056' 20 20 20 20
005A' 1B 4B 00 01      db 1bh, 'k', 0, 1    ; Bitmuster mit doppelter Dichte
005E' FF              db 0ffh          ; 0ffh als Ende-Marke

; 8 Spalten des Bildschirm abtasten

005F' 06 0B          GetLine: id b, 8          ; Zahl der Spalten pro Druckzeile
0061' 0E 00          Loop3:  ld c, 0          ; Zahl der Punkte pro Spalte
0063' 3E FE              ld a, 0feh          ; Zeilennummer laden
0065' D3 8B              out (Bbh), a
0067' 3E FF              ld a, 0ffh
0069' D3 BA              out (Bah), a
006B' 2A 00C0'          ld hl, (Row)        ; Nummer der vorherigen Spalte
006E' 2B                dec hl              ; Spaltenzaehler erniedrigen
006F' 22 00C0'          ld (Row), hl        ; Spaltenzaehler wieder sichern
0072' 7C                ld a, h
0073' 2F                cpl                 ; Byte komplementieren
0074' D3 8B              out (88h), a
0076' 7D                ld a, l
0077' 2F                cpl                 ; Byte komplementieren
0078' 21 00C2'          ld hl, Buffer        ; Adresse des Zeilenpuffers
007B' D3 89              out (89h), a
007D' DB 8D              in a, (8dh)         ; Flag des SN 74173 loeschen

007F' DB 8A          Loop4:  in a, (8ah)          ; Warten bis Punkt gefunden
0081' 07              rlca                ; Bit 7 ins CY-Flag
0082' D2 007F'          jp nc, Loop4        ; Punkt noch nicht gefunden
0085' 07              rlca                ; Bit des Punkts in CY-Flag
0086' 7E              ld a, (hl)          ; bisheriges Byte laden
0087' 17              rla                 ; Bit zum alten Byte hinzufuegen
0088' 77              ld (hl), a          ; Byte wieder abspeichern
0089' DB 8D              in a, (8dh)         ; Flag des SB 74173 loeschen
008B' 23              inc hl              ; naechste Stelle im Puffer
008C' 0D              dec c                ; weiterer Punkt ?
008D' C2 007F'          jp nz, Loop4

0090' 05              dec b                ; weitere Spalte ?
0091' C2 0061'          jp nz, Loop3
0094' C9              ret

; Ausgabe des Zeilenpuffers

0095' 21 00C2'          PrtLine: ld hl, Buffer        ; Adresse des Zeilenpuffers
0098' 0E 00              ld c, 0            ; Zahl der Speicherstellen
009A' 7E          Loop2:  ld a, (hl)          ; Speicherstelle laden
009B' 2F              cpl                 ; Akku negieren
009C' CD 00AF'          call Out            ; Byte ausgeben
009F' 23              inc hl              ; naechste Speicherstelle
00A0' 0D              dec c                ; Zahl der restl. Stellen
00A1' C2 009A'          jp nz, Loop2
00A4' 3E 0D              ld a, 0dh          ; Carriage Return ausgeben
00A6' CD 00AF'          call Out            ; Byte ausgeben
00A9' 3E 0A              ld a, 0ah          ; Line Feed ausgeben
00AB' CD 00AF'          call Out            ; Byte ausgeben
00AE' C9              ret

```

; Ausgabe eines Zeichens auf IDE+CENT

```
00AF' FS      Out:      push af          ; A-Register sichern
00B0' DB 49   Out1:     in a,(centstb)   ; Busy-Flag
00B2' 0F                      rrca             ; Bit 0
00B3' 3B FB                      jr c,Out1       ; Drucker Ready ?
00B5' F1                      pop af          ; Akku restaurieren
00B6' D3 4B                      out (centdaten),a ; Byte ausgeben
00B8' AF                      xor a          ; Akku loeschen
00B9' D3 49                      out (centstb),a ; Strobe Pulse
00BB' 3E 01                      ld a,1        ; Strobe Bit auf 1 setzen
00BD' D3 49                      out (centstb),a ; Pulse beenden
00BF' C9                      ret

00E0' 02D0   Row:      dw 720          ; Numer der naechsten Spalte
00C2'       Buffer:    ds 256         ; Puffer fuer eine Zeile
01C2'       Stack:   ds 20          ; Platz fuer Stack
01D6' 00

end
```



9.5. Testprogramm für Fadenkreuz und Maus (CPU 68008)

```

;
; *****
; Programm zum Test der Maus-Schnittstelle und
; des Fadenkreuzes der HCDPY/MAUS-Platine mit
; der CPU 68008.
;
; Nach dem Aufruf des Programms erscheint das
; Fadenkreuz in der Mitte des Bildschirms und
; folgt allen Bewegungen der Maus. Die Betaetigung
; einer Taste fuehrt zur Beendigung des Programms.
;
; (C) G. Sternberg 1985          Stand 24.7.1985
; *****
start:
LoX   equ $ffffB9      ; Fadenkreuz X-Position (lo-Byte)
HiX   equ $ffffB8      ; Fadenkreuz X-Position (hi-Byte)
LoY   equ $ffffBb      ; Fadenkreuz Y-Position (lo-Byte)
HiY   equ $ffffBa      ; Fadenkreuz Y-Position (hi-Byte)
Up    equ $ffffBc      ; Zaehler fuer Aufwaertsimpulse
Down  equ $ffffBd      ; Zaehler fuer Abwaertsimpulse
Right equ $ffffBe      ; Zaehler fuer Rechtsbewegung
Left  equ $ffffBf      ; Zaehler fuer Linksbewegung
Latch equ $ffffBd      ; Adresse zum Latchen des Zaehlers
Clear equ $ffffBe      ; Adresse zum Loeschen des Zaehlers

START: move.w #$ffff-128,d0      ; Y-Position vorbelegen
       move.w #$ffff-256-191,d1 ; X-Position vorbelegen

Loop:  move.b #0,Latch      ; Zaehlerstand speichern
       move.b #0,Clear     ; alten Zaehlerstand loeschen

       move.l #0,d2        ; Register loeschen
       move.b right,d2     ; Rechtsbewegung
       sub d2,d1           ; X-Position aktualisieren
       move.b left,d2      ; Linksbewegung
       add d2,d1           ; X-Position aktualisieren
       move.b up,d2        ; Aufwaertsbewegung
       add d2,d0           ; Y-Position aktualisieren
       move.b down,d2      ; Abwaertsbewegung
       sub d2,d0           ; Y-Position aktualisieren

       move.b d0,LoY       ; Fadenkreuz Y-Position (lo-Byte)
       ror.w #8,d0         ; Register um 8 Bit verschieben
       move.b d0,HiY       ; Fadenkreuz Y-Position (hi-Byte)
       ror.w #8,d0         ; Register um 8 Bit verschieben
       move.b d1,LoX       ; Fadenkreuz X-Position (lo-Byte)
       ror.w #8,d1         ; Register um 8 Bit verschieben
       move.b d1,HiX       ; Fadenkreuz X-Position (hi-Byte)
       ror.w #8,d1         ; Register um 8 Bit verschieben
       bts #7,$ffff6B      ; Taste gedrueckt ?
       bne Loop

       move.b #0,LoX       ; Fadenkreuz ausblenden
       move.b #0,LoY
       move.b #0,HiX
       move.b #0,HiY
       rts
       end

```

## 9.6. Testprogramm für Hardcopy (CPU 68008)

```

;
; *****
; Programm zur Durchfuehrung einer Hardcopy
; mit der CPU 68008.
;
; (C) G. Sternberg 1985      Stand 24.7.1985
; *****
;
LoX   equ $ffff89      ; Fadenkreuz X-Position (lo-Byte)
HiX   equ $ffff88      ; Fadenkreuz X-Position (hi-Byte)
LoY   equ $ffff8b      ; Fadenkreuz Y-Position (lo-Byte)
HiY   equ $ffff8a      ; Fadenkreuz Y-Position (hi-Byte)
Ready equ $ffff8a      ; Flag fuer gefundene Spalte
Clear equ $ffff8d      ; Ready-Flag loeschen

START:— jsr InitRX80      ; Drucker initialisieren
         move.w #$fff-720,d2 ; Nummer der letzten Spalte
         move.b #67,d3      ; Zahl der Druckzeilen
                           ; (8 Punktspalten / Druckzeile)
Loop2:  jsr GetLine      ; eine Druckzeile abtasten
         jsr InitLine    ; Drucker fuer die Ausgabe
                           ; einer Zeile initialisieren
         jsr PrtLine     ; Druckzeile ausgeben
         subq.b #1,d3    ; alle Druckzeilen ausgegeben ?
         bne Loop2      ; naechste Druckspalte ausgeben

         move.b #0,LoX   ; Fadenkreuz ausblenden
         move.b #0,LoY
         move.b #0,HiX
         move.b #0,HiY
         rts

; Anfangs-Initialisierung des Druckers

InitRX80: lea InitTab,a0   ; Adresse der Tabelle mit den
                           ; Initialisierungsdaten

Loop:     move.b (a0)+,d0  ; Byte ins Register laden
         jsr $Lo         ; Byte ausgeben
         cmpi.b #$ff,(a0) ; $ff als Endemarke ?
         bne Loop      ; Endemarke gefunden ?
         rts

InitTab:  dc.b $1b,'9',40d,$0a,$0a,$0a,$0a,$1b,'3',24,$ff
         ds 0

; Initialisierung des Druckers fuer eine Zeile

InitLine: lea InitTab1,a0 ; Adresse der Tabelle mit den
                           ; Initialisierungsdaten

Loop1:   move.b (a0)+,d0  ; Byte ins Register laden
         jsr $LD         ; Byte ausgeben
         cmpi.b #$ff,(a0) ; $ff als Endemarke ?
         bne Loop1     ; Endemarke gefunden ?
         rts

InitTab1: dc.b          ;,$1b,'K',0,1,$ff
         ds 0

```



```

; 8 Spalten des Bildschirm abtasten

GetLine:  move.b #8,d4      ; Zahl der Spalten pro Druckzeile
Loop4:    move.w #256,d5   ; Zahl der Punkte pro Spalte
          lea Buffer,a0  ; Adresse des Puffers laden
          addq.w #1,d2   ; Spaltenzaehler erniedrigen
          move.b #ife,LoY ; Zeilennummer (lo-Byte) laden
          move.b #fff,HiY ; Zeilennummer (hi-Byte) laden
          ror.w #8,d2    ; Register um 8 Bit schieben
          move.b d2,HiX  ; Spaltennummer (hi-Byte) laden
          ror.w #8,d2    ; Register um 8 Bit schieben
          move.b d2,LoX  ; Spaltennummer (lo-Byte) laden
          move.b Clear,d6 ; Ready-flag loeschen

Loop5:    bst #7,Ready   ; Warten bis Punkt gefunden
          beq Loop5     ; Punkt noch nicht gefunden
          move.b Ready,d6 ; Ready-Flag + Bildpunkt laden
          lsl.b #2,d6    ; Bildpunkt in X-Flag schieben
          move.b (a0),d7 ; bisheriges Byte laden
          roxl.b #1,d7   ; bisheriges Byte rotieren
          move.b d7,(a0)+ ; bisheriges Byte speichern
          move.b Clear,d6 ; Ready-flag loeschen
          subq.w #1,d5   ; alle Punkte abgetastet
          bne Loop5

          subq.b #1,d4   ; alle Spalten abgetastet ?
          bne Loop4
          rts

; Ausgabe des Zeilenpuffers

PrtLine:  lea Buffer,a0  ; Adresse des Zeilenpuffers
          move.w #256,d1 ; Zahl der Speicherstellen

Loop3:    move.b (a0)+,d0 ; Speicherstelle laden
          not.b d0       ; Byte negieren
          jsr $Lo       ; Byte ausgeben
          subq.w #1,d1  ; alle Bytes ausgegeben
          bne Loop3    ; naechstes Byte ausgeben
          move.b #$0d,d0 ; Carriage Return ausgeben
          jsr $Lo       ; Byte ausgeben
          move.b #$0a,d0 ; Line Feed ausgeben
          jsr $Lo       ; Byte ausgeben
          rts

Buffer *  ds.b 256      ; Puffer fuer eine Druckzeile
end

```

9.7. Programm für Hardcopy unter CP/M 2.2

```

                                .Z80
0000'                          cseg

                                ;
                                ; *****
                                ; Programm zum Betrieb der Hardcopy/Maus-Baugruppe
                                ; unter dem Betriebssystem CP/M am mc-CP/M oder
                                ; NDR-Klein-Computer.
                                ; Durch den Aufruf des Programms wird die eigentliche
                                ; Hardcopy-Routine in den freien Speicherbereich
                                ; hinter dem Monitor kopiert. Die Aktivierung der
                                ; Routine erfolgt durch die Eingabe eines bestimmten
                                ; Zeichens mittels Tastatur.
                                ;
                                ; (C) G. Sternberg 1985                      Stand 18.4.1985
                                ;
                                ; Das Programm ist zur Veröffentlichung in der
                                ; Zeitschrift mc bestimmt. Alle Rechte bezüglich
                                ; einer Veröffentlichung liegen beim Franzis
                                ; Verlag München
                                ; *****
                                ;
0000' 11 001A'                ld de,Text          ; Statuszeile ausgeben
0003' 0E 09                   ld c,09h           ; Kennung fuer "Print String"
0005' CD 0005                 call 0005h
0008' 21 007B'                ld hl,lo          ; Startadresse
000B' 11 FB00                 ld de,0FB00h      ; Zieladresse
000E' 01 01E1                 ld bc,hi-lo       ; Zahl der zu kopierenden Bytes
0011' ED B0                   idir             ; Programm kopieren
0013' 21 FB00                 ld hl,Check       ; Adresse der Check-Routine
0016' 22 EA0A                 ld (0ea0ah),hl    ; Consol-Input umlenken (60 k CP/M)
0019' C9                      ret

001A' 1B 1B 47                Text:        db 1bh,1bh,'G'
001D' 50 30 0D 4C             db 'P0',0dh,'L0 0 511 0 511 12 0 12',0dh
0021' 30 20 30 20
0025' 35 31 31 20
0029' 30 20 35 31
002D' 31 20 31 32
0031' 20 30 20 31
0035' 32 0D
0037' 4D 34 33 30             db 'M430 2',0dh,'BHardcopy ^5',0dh
003B' 20 32 0D 42
003F' 48 61 72 64
0043' 63 6F 70 79
0047' 20 5E 40 0D
004B' 50 35 0D 4C             db 'P5',0dh,'L0 0 511 0 511 12 0 12',0dh
004F' 30 20 30 20
0053' 35 31 31 20
0057' 30 20 35 31
005B' 31 20 31 32
005F' 20 30 20 31
0063' 32 0D
0065' 4D 34 33 30             db 'M430 2',0dh,'BHardcopy ^5',0dh
0069' 20 32 0D 42
006D' 48 61 72 64
0071' 63 6F 70 79
0075' 20 5E 40 0D

```

```

0079 41 24 db 'AF'

007B io:

.Phase 0f800h

F800 CD F003 Check: call 0f003h ; eingegebenes Zeichen laden
F803 FE 00 cp 0 ; Control 0 als Startzeichen
F805 CA F80B jp z,Hcopy ; Hardcopy ausfuehren
F808 E6 7F and 7fh ; Bit 7 loeschen
F80A C9 ret

F80E 21 0000 HCopy: ld hl,0 ; H+L-Register loeschen
F80E 39 add hl,sp ; Stackpointer nach H+L kopieren
F80F 22 F8C8 ld (OldStack),hl ; alten Stackpointer retten
F812 31 F9E0 ld sp,Stack ; Stackpointer laden
F815 21 02D0 ld hl,720 ; Nummer der ersten Spalte
F818 22 F8CA ld (Row),hl ; Zahl abspeichern
F81B CD F840 Call InitRX80 ; Initialisierung des Druckers
F81E 3E 43 ld a,67 ; Zahl der Druckzeilen
; (Punkte pro Zeile / 8)

F820 F5 Loop1: push af ; Wiederholungsfaktor sichern
F821 CD F873 Call GetLine ; eine Druckzeile abtasten
F824 CD F846 Call InitLine ; Drucker fuer die Ausgabe
; dieser Zeile initialisieren
F827 CD F8A9 Call PrtLine ; Zeilenpuffer ausgeben
F82A F1 pop af ; Wiederholungsfaktor laden
F82B 3D dec a ; weitere Druckzeile ?
F82C D2 F820 jp nz,Loop1

F82F 3E 00 ld a,0 ; Fadenkreuz ausschalten
F831 D3 88 out (88h),a
F833 D3 89 out (89h),a
F835 D3 8A out (8Ah),a
F837 D3 8B out (8bh),a
F839 2A F8C8 ld hl,(OldStack) ; H+L mit altem Stackpointer laden
F83C F9 ld sp,hl ; Stackpointer restaurieren
F83D C3 F800 jp Check ; Eingabe des naechsten Zeichens

; *****
; Initialisierung des Druckers
; *****
;

F840 21 F857 InitRX80: ld hl,InitTab ; Adresse der Tabelle mit den
; Daten zur Initialisierung

F843 C3 F849 jp Loop
F846 21 F862 InitLine: ld hl,InitTab1 ; Initialisierung fuer eine Zeile
F849 4E Loop: ld c,(hl) ; Byte laden
F84A E5 push hl ; Register sichern
F84B CD F00F call 0f00fh ; Byte ausgeben
F84E E1 pop hl ; Register restaurieren
F84F 23 inc hl ; Adresse erhoehen
F850 3E FF ld a,0ffh ; Akku loeschen
F852 BE cp (hl) ; Ende-Marke ?
F853 C2 F849 jp nz,Loop ; naechstes Byte ausgeben
F856 C9 ret ; Ende der Initialisierung

F857 1B 40 InitTab: db 1bh,5' ; Drucker normieren
F859 0D 0A 0A 0A db 0dh,0ah,0ah,0ah,0ah ; oberer Rand
F85D 0A

```



```

F85E 1B 33 18          db 1bh,'3',24      ; Zeilabstand auf 24/216 Zeil
F861 FF                db 0ffh              ; 0ffh als Ende-Marke
F862 20 20 20 20      InitTab1: db                ; linker Rand
F866 20 20 20 20
F86A 20 20 20 20
F86E 1B 4B 00 01      db 1bh,'K',0,1     ; Bitmuster mit doppelter Dichte
F872 FF                db 0ffh              ; 0ffh als Ende-Marke

; *****
; B Spalten des Bildschirms abtasten
; *****
;

F873 06 0B          GetLine:  ld b,9          ; Zahl der Spalten pro Druckzeile
F875 0E 00          Loop3:   ld c,0          ; Zahl der Punkte pro Spalte
F877 3E FE                ld a,0feh      ; Zeilennummer laden
F879 D3 8B          out (8bh),a
F87B 3E FF                ld a,0ffh
F87D D3 8A          out (8ah),a
F87F 2A FB8A        ld hl,(Row)    ; Nummer der vorherigen Spalte
F882 2B                dec hl        ; Spaltenzaehler erniedrigen
F883 22 FB8A        ld (Row),hl   ; Spaltenzaehler wieder sichern
F886 7C                ld a,h        ; Spaltennummer laden
F887 2F                cpl          ; Byte komplementieren
F88B D3 8B          out (88h),a
F88A 7D                ld a,l
F88B 2F                cpl          ; Byte komplementieren
F88C 21 FBCC        ld hl,Buffer  ; Adresse des Zeilenpuffers
F88F D3 89          out (89h),a
F891 DB 8D          in a,(8dh)    ; Flag des SN 74173 loeschen

F893 DB 8A          Loop4:  in a,(8ah)    ; warten bis Punkt gefunden
F895 07                rlc         ; Bit 7 ins CY-Flag
F896 D2 F893        jp nc,Loop4   ; Punkt noch nicht gefunden
F899 07                rlc         ; Bit des Punkts in CY-Flag
F89A 7E                ld a,(hl)    ; bisheriges Byte laden
F89B 17                rla         ; Bit zum alten Byte hinzufuegen
F89C 77                ld (hl),a    ; Byte wieder abspeichern
F89D DB 8D          in a,(8dh)    ; Flag des SB 74173 loeschen
F89F 2B                inc hl       ; naechste Stelle im Puffer
F8A0 0D                dec c        ; weiterer Punkt in dieser Spalte ?
F8A1 C2 F893        jp nz,Loop4

F8A4 05                dec b        ; weitere Spalte ?
F8A5 C2 F875        jp nz,Loop3
F8AB C9                ret

; *****
; Ausgabe des Zeilenpuffers
; *****
;

F8A9 21 FBCC        PrtLine: ld hl,Buffer  ; Adresse des Zeilenpuffers
F8AC 0E 00          ld c,0        ; Zahl der Speicherstellen
F8AE 7E                Loop2:  ld a,(hl)    ; Speicherstelle laden
F8AF 2F                cpl          ; Akku negieren
F8B0 E5                push hl     ; Register sichern
F8B1 C5                push bc
F8B2 4F                ld c,a
F8B3 CD F00F        call 0f00fh  ; Byte ausgeben
F8B6 C1                pop bc     ; Register restaurieren
F8B7 E1                pop hl
F8BB 23                inc hl     ; Adresse erhoehen

```

```

FBB9 0D                dec c                ; Zahl der restl. Stellen
F8BA C2 FBAE          jp nz,Loop2
F8BD 0E 0D            ld c,0dh            ; Carriage Return ausgeben
F8BF CD F00F          call 0f00fh         ; Byte ausgeben
F8C2 0E 0A            ld c,0ah            ; Line Feed ausgeben
F8C4 CD F00F          call 0f00fh         ; Byte ausgeben
F8C7 C9                ret

F8C8 0000             OldStack: dw 0      ; Wert des Stackpointers
F8CA 02D0             Row:      dw 720   ; Nummer der naechsten Spalte
F8CC                 Buffer:    ds 256  ; Puffer fuer eine Zeile
F9CC                 ds 20   ; Platz fuer Stack
F9E0 00              Stack:   db 0

```

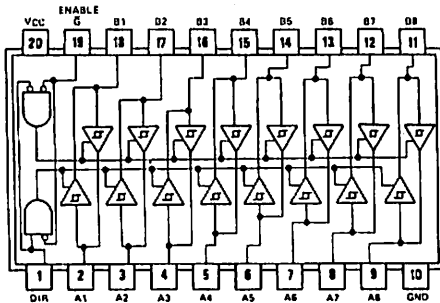
.Dephase

025C' hi:

end

# 10. Bauelemente

74 LS 245



8 Bit Bustreiber mit Tri-State

Es handelt sich hier um ein reines Tri-State-Element, d.h. es hat drei Funktionsmöglichkeiten. Es kann Daten von der A-Seite zur B-Seite oder umgekehrt durchschalten, oder aber es sperrt in beiden Richtungen. Den sperrenden Zustand nennt man auch "hochohmig". Der Zustand des Bus-Treibers wird durch die Signale auf den Eingängen DIR und G festgelegt.

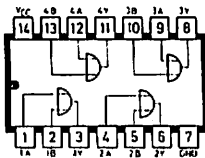
FUNCTION TABLE

ENABLE G	DIRECTION CONTROL DIR	OPERATION
L	L	B data to A bus
L	H	A data to B bus
H	X	Isolation

H = high level, L = low level, X = irrelevant

74 LS 32

4 OR-Gatter



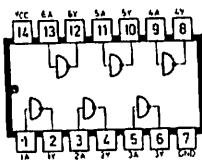
PositiveLogik Y = A · B

Dieser TTL-Baustein enthält 4 OR-Gatter. Jeweils zwei Eingänge werden gemäß der ODER-Funktion verknüpft. Das Ergebniss liefert der jeweilige Y-Ausgang.

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

7406

6 Invert-Gatter mit offenem Kollektor



PositiveLogik Y = A

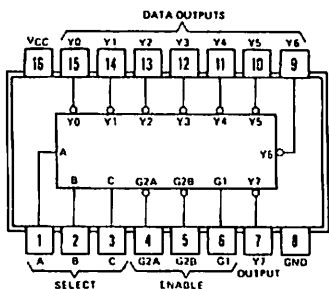
Der Inverter (auch NICHT-Glied genannt) kehrt die Signale einfach um; L wird zu H und H wird zu L. NICHT-Glieder werden z.B. dazu verwendet, um von positiver zu negativer Logik zu wechseln.

Die Ausgänge des vorliegenden Bausteins stellen sogenannte "Open Collector"-Ausgänge dar. Diese Art von Ausgang kann einen höheren Ausgangsstrom liefern, und gleichzeitig lassen sich mehrere Ausgänge dieses Typs einfach zusammenschalten ("wired Or").



74 LS 138

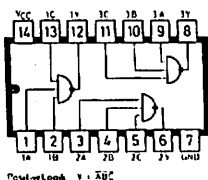
3-Bit Dekocder-Baustein (3 zu 8)



Dieser Baustein liefert in Abhängigkeit der Selektions- und Freigabeeingänge ein low-Signal auf einem der 8 Ausgangsleitungen. Die Eingänge A, B und C enthalten binär die Nummer des gewünschten Ausgangs. Der entsprechende Ausgang führt aber nur dann ein low-Signal (alle anderen Ausgänge liegen auf high), wenn der Selektionseingang G1 ein high-Signal hat und die Selektionseingänge G2A und G2B auf low-Pegel liegen. Falls die Freigabeeingänge diese Bedingung nicht erfüllen, so führen alle Ausgänge ein high-Signal.

74 LS 12

3 NAND-Gatter (3 Eingänge) mit offenem Kollektor

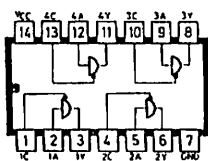


Dieser TTL-Baustein enthält 3 NAND-Gatter. Jeweils drei Eingänge werden gemäß der NAND-Funktion verknüpft. Das Ergebnis liefert der jeweilige Y-Ausgang.

Die Ausgänge des vorliegenden Bausteins stellen sogenannte "Open Collector"-Ausgänge dar. Diese Art von Ausgang kann einen höheren Ausgangstrom liefern, und gleichzeitig lassen sich mehrere Ausgänge dieses Typs einfach zusammenschalten ("wired Or").

74 LS 125

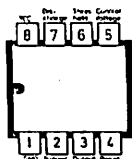
4 Treiber mit 3-State Ausgang



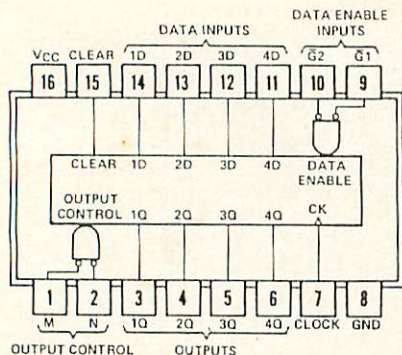
Bei einem Low-Signal auf dem Control-Eingang (C-Eingang) besitzt der Ausgang den gleichen Logikzustand wie der zugehörige Eingang. Bei einem High-Signal auf dem Control-Eingang geht der Ausgang in einen hochohmigen Zustand über. In diesem Zustand gleicht der Ausgangs-Pin einem Anschluß, der im Inneren des Bausteins keinen Anschluß hätte.

NE 555

Timer



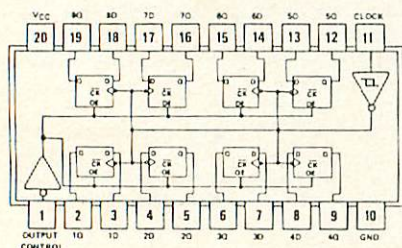
Entsprechend seiner externen Beschaltung kann der Baustein u.a. sowohl als monostabiler als auch als astabiler Zeitgeber wirken.



Bei einer positiven Flanke am Clock-Eingang und einem Low-Signal an den beiden Freigabeeingängen (G1 und G2) übernimmt das Register den augenblicklichen Zustand an den 4 Eingängen. Bei einem Low-Signal an den beiden Freigabeeingängen (M und N) erscheint der gespeicherte Zustand an den Ausgängen. Ansonsten besitzen die Ausgänge einen hochohmigen Zustand. Ein High-Signal am Clear-Eingang setzt alle 4 Bits des Registers auf Low.

74 LS 374

8-Bit D-Register mit 3-State-Ausgängen



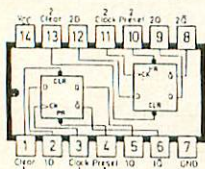
Dieser Baustein verhält sich praktisch wie der 74 LS 173, besitzt aber keinen Clear-Eingang.

FUNCTION TABLE

OUTPUT CONTROL	CLOCK	D	OUTPUT
L	↑	H	H
L	↑	L	L
L	L	X	Q <sub>0</sub>
H	X	X	Z

74 LS 74

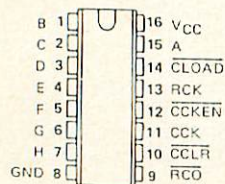
2 D-Flip-Flops mit Clear und Preset



Bei einer positiven Flanke am Clock-Eingang übernimmt das Flip-Flop den Zustand des D-Eingangs und zeigt diesen am Q-Ausgang an. Bei einem Low-Signal am Clear-Eingang geht der Ausgang (Q) auf Low; bei einem Low-Signal am Preset-Eingang nimmt der Ausgang ein High-Signal an.

74 LS 592

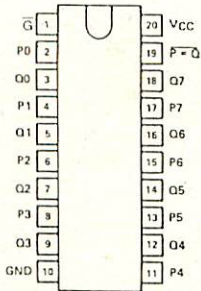
8-Bit Zähler mit Eingaberegister



Bei einem Low-Signal am Freigabeeingang CCKEN und einer positiven Flanke am CCK-Eingang taktet der Zähler um Eins weiter. Bei Erreichen des Zählerstandes FFH liefert der RCO-Ausgang ein Low-Signal. Ein Low-Signal am CCLR-Eingang löscht den Zähler. Eine positive Flanke am RCK-Eingang bewirkt die Übernahme der Daten an den Eingängen A-H in das Eingaberegister und ein Low-Signal am CLOAD-Eingang eine Übernahme des Eingaberegisters als neuen Zählerstand.

## 74 LS 688

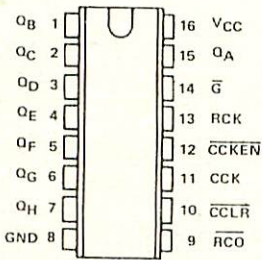
## 8-Bit Vergleicher



Bei Übereinstimmung der 8 Signale der P-Eingänge mit den 8 Signalen der Q-Eingängen und einem Low-Signal am G-Eingang liefert der Baustein ein Low-Signal am P=Q-Ausgang.

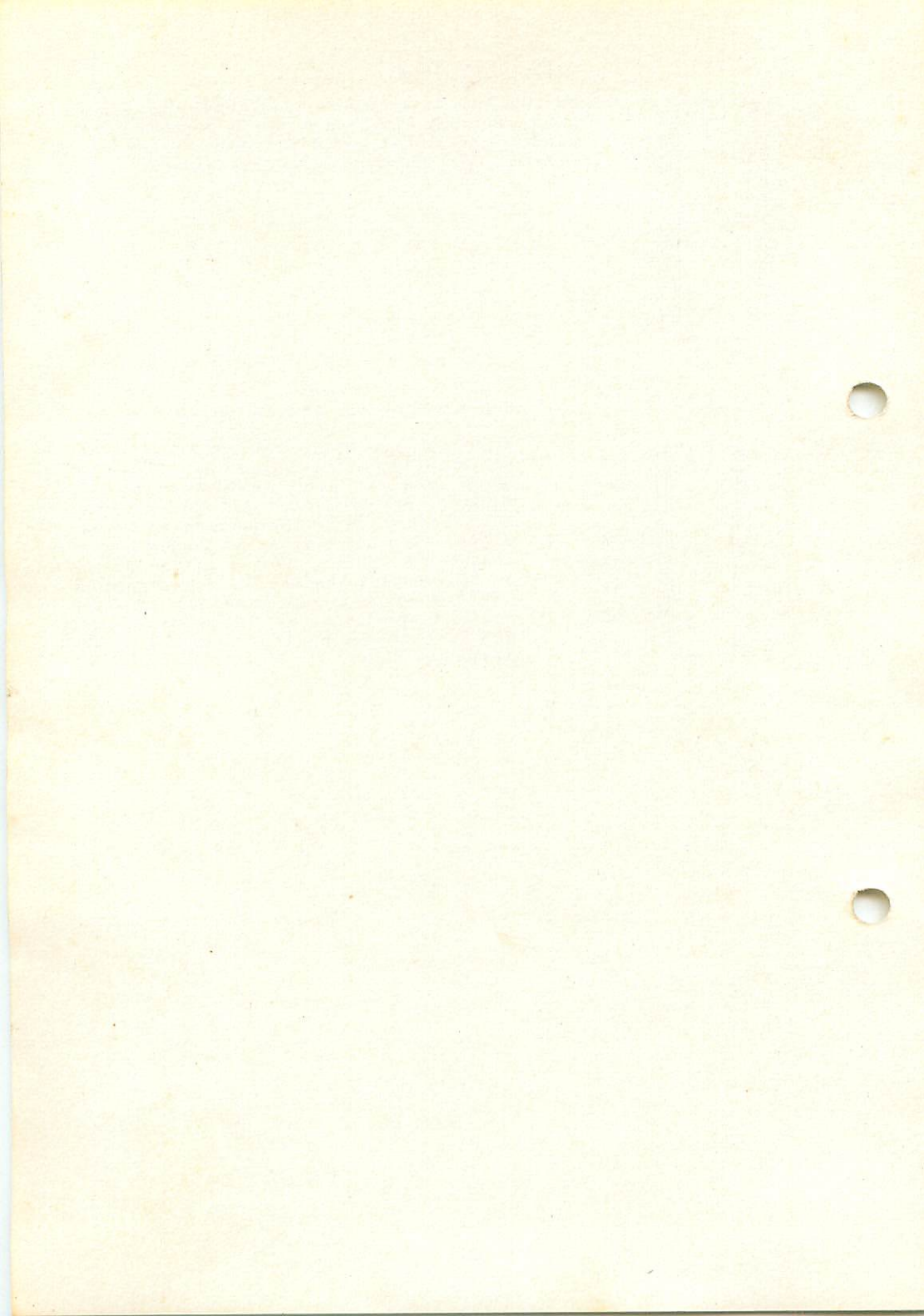
## 74 LS 590

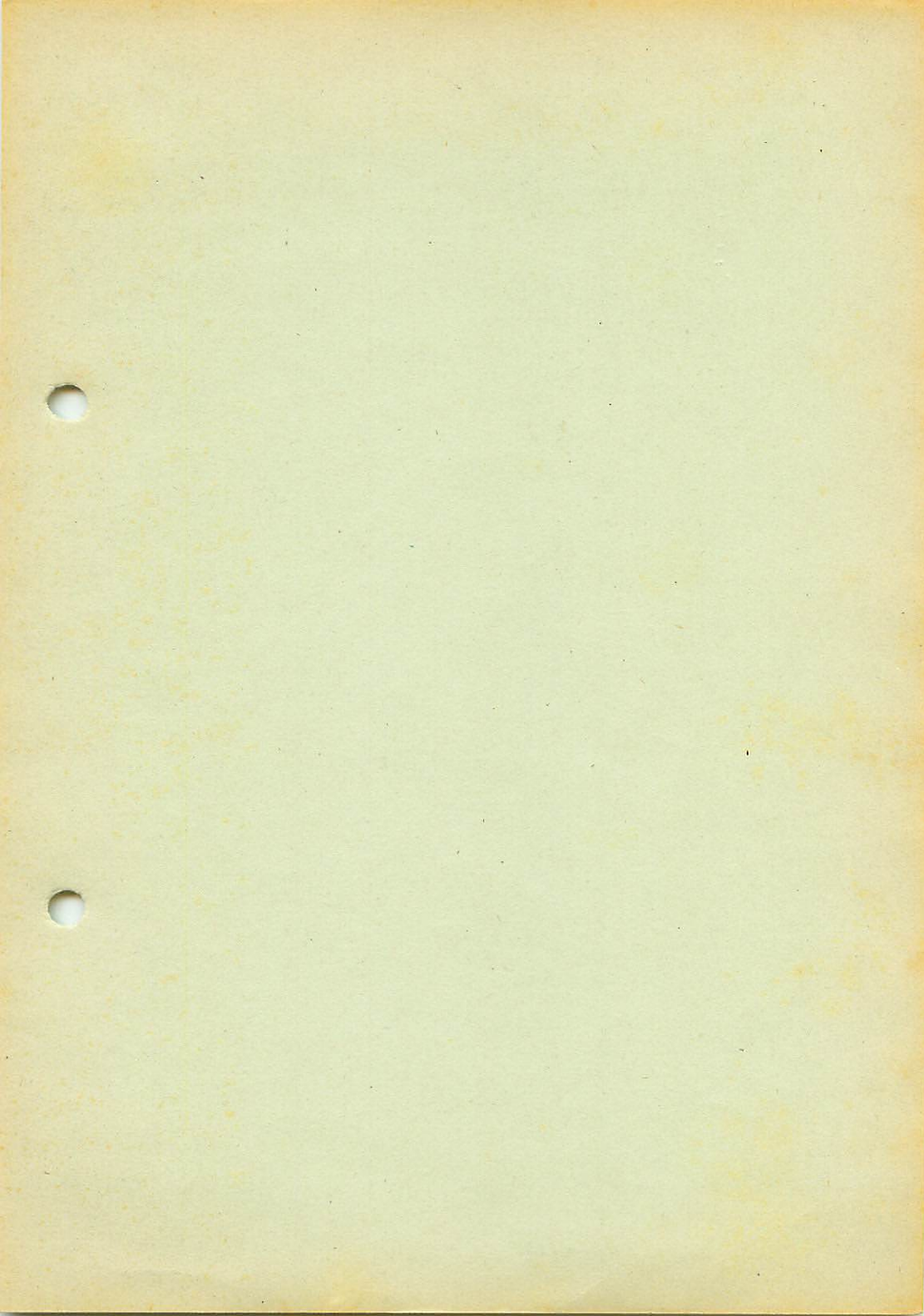
## 3-Bit Zähler mit Ausgaberegister



Eine positive Flanke am CCK-Eingang, bei gleichzeitigem Low-Signal am CCKEN-Eingang, bewirkt eine Weitertaktung des Zählers. Bei Erreichen des Zählerstandes FFH liefert der RCO-Ausgang ein Low-Signal. Ein Low-Signal am CCLR-Eingang löscht den Zähler. Eine positive Flanke am RCK-Eingang bewirkt die Übernahme des Zählerstandes in das Ausgangsregister, und ein Low-Signal am G-Eingang bewirkt die Ausgabe des Registerinhalts an den Ausgängen A-H.







**Graf Elektronik Systeme GmbH**  
Magnusstr. 13 · Postfach 1610  
8960 Kempten (Allgäu)  
Telefon: (08 31) 62 11  
Teletex: 831804 = GRAF  
Telex: 17 831 804 = GRAF

**Filiale Hamburg**  
Ehrenbergstraße 56  
2000 Hamburg 50  
Telefon: (0 40) 38 81 51  
**Filiale München**  
Georgenstraße 61  
8000 München 40  
Telefon: (0 89) 2 71 58 58

ger



Fehlerberichtigungen  
HCOPY/MAUS-Handbuch

- Seite 6 Achtung Falle  
Auch die IC12 und IC13 sind verdreht
  
- Seite 7 Falsche Überschrift:  
Layout Bestückungsseite  
  
richtig wäre: Layout Lötseite  
(Die Bestückungsseite ist dem  
Bestückungsplan unterlegt)
  
- Seite 15 Schaltplan ist die Ausgabe 1
  
- Seite 29 Testprogramm 68008  
  
In Zeile 25 (erste Zeile = nur;)  
fehlt Start:  
falsch                    move.w #\$ffff-128,do.  
richtig Start: move.w #\$ffff-128,do.
  
- Seite 30 In Zeile 16 fehlt Start:  
falsch                    jsr InitRX80.....  
richtig Start: jsr InitRX80.....
  
- Seite 31 Doppelpunkt fehlt  
falsch                    Buffer  
richtig                    Buffer:

