

**DATA BECKERS**

# DEMO MAKER

Scanned  
by  
Jc8k

Taken from Amiga-Manuals-Website

**DATA BECKER**  
SOFTWARE



Florian Schroecker • Sven Sommer

**DATA BECKERs**

**DEMO  
MAKER**

**DATA BECKER**

**Copyright** © 1990 by DATA BECKER GmbH  
Merowingerstr. 30  
4000 Düsseldorf 1

1. Auflage 1990

**Umschlaggestaltung** Werner Leinhos

**Textverarbeitung  
und Gestaltung** Udo Bretschneider

**Text verarbeitet mit** Word 5.0, Microsoft

**Belichtung** Graf & Pflügge, Düsseldorf

**Druck und  
buchbinderische Verarbeitung** Mohndruck, Gütersloh

Alle Rechte vorbehalten. Kein Teil dieses Buches darf in irgendeiner Form (Druck, Fotokopie oder einem anderen Verfahren) ohne schriftliche Genehmigung der DATA BECKER GmbH reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

ISBN 3-89011-814-3



### **Wichtiger Hinweis**

Die in diesem Buch wiedergegebenen Verfahren und Programme werden ohne Rücksicht auf die Patentlage mitgeteilt. Sie sind ausschließlich für Amateur- und Lehrzwecke bestimmt und dürfen nicht gewerblich genutzt werden.

Alle technischen Angaben und Programme in diesem Buch wurden von dem Autor mit größter Sorgfalt erarbeitet bzw. zusammengestellt und unter Einschaltung wirksamer Kontrollmaßnahmen reproduziert. Trotzdem sind Fehler nicht ganz auszuschließen. DATA BECKER sieht sich deshalb gezwungen, darauf hinzuweisen, daß weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen, übernommen werden kann. Für die Mitteilung eventueller Fehler ist der Autor jederzeit dankbar.



# Inhaltsverzeichnis

<b>1. Einleitung</b> .....	<b>9</b>
<b>2. Installation</b> .....	<b>11</b>
2.1 Installieren auf einer Harddisk .....	11
2.2 Starten von einer Diskette .....	11
<b>3. Der Aufbau des Bildschirms</b> .....	<b>13</b>
<b>4. Die Bediener-Oberfläche</b> .....	<b>15</b>
4.1 Die Preferenzliste .....	16
4.2 Möglichkeiten der Preferenzliste .....	17
4.3 Editieren von Texten im Editor .....	23
4.3.1 Sonderzeichen im Texteditor .....	24
4.3.2 Sonderzeichen bei Sinusscroller .....	26
<b>5. Das Kernstück - der PATTERN-EDITOR</b> .....	<b>29</b>
5.1 Die Hauptpunkte des PATTERN-EDITORS .....	29
5.2 Die Teilstückbefehle .....	30
5.3 Ändern der Farbenlisten; .....	55
<b>6. Das Demo-Menü</b> .....	<b>57</b>
<b>7. Verwendung eigener Grafiken</b> .....	<b>59</b>
<b>8. Musikroutinen</b> .....	<b>61</b>

<b>9. Erstellen von eigenen Demos</b> .....	<b>63</b>
9.1 Vorbereitungen zum Demo .....	63
9.2 Der Pattern-Editor .....	66
<b>10. Eigene Vektorobjekte</b> .....	<b>73</b>
10.1 Eigene Vektorball-Objekte .....	77
10.2 Ändern bestehender Vektorobjekte .....	78
<b>11. Stichwortverzeichnis</b> .....	<b>85</b>

## 1. Einleitung

Demos! Sicher haben Sie auch schon einmal die phantastischen Demos bestaunt, die von diversen User-Gruppen in regelmäßigen Abständen herausgegeben werden. Und sicherlich haben Sie dabei auch schon einmal den Wunsch verspürt, selbst solche Demos zu erstellen, aber den Gedanken daran dann verwerfen müssen, da Ihnen die notwendige Programmiererfahrung fehlt. Mit dem vorliegenden Demomaker brauchen Sie sich keine Gedanken mehr um Programmierkenntnisse zu machen, Sie brauchen nicht einmal eine Programmiersprache zu kennen.

Nach nunmehr monatelanger intensiver Programmierarbeit liegt der Demomaker vor Ihnen. Mit diesem Programm haben Sie die Möglichkeit, Ihrer Phantasie und natürlich Ihrer Kreativität völlig freien Lauf zu lassen. Sie können die komplexesten Demos in kürzester Zeit erstellen. Schon nach wenigen Minuten werden Sie feststellen, daß Sie eine fast unbegrenzte Anzahl von Möglichkeiten haben, um selbst Demos zu erstellen.

Der Demomaker ist für den Amiga-Einsteiger genauso interessant wie für den langjährigen Programmier-Profi.

*Warnung:* Dieses Programm macht süchtig. Schon nach wenigen Minuten können Sie sich nicht mehr von Ihrem Amiga losreißen. Starten Sie das Programm keinesfalls, wenn Sie nur zwei bis drei Stunden Zeit haben, denn gerade dann werden Sie die ganze Nacht aufbleiben.

Sollten Sie mit den Musikprogrammen SoundTracker, NoiseTracker oder einem der anderen Programme bereits Erfahrung gesammelt haben, kommt Ihnen das jetzt zugute. Denn der Demomaker unterstützt die folgenden Musikformate:

bp soundmon  
 mark ii  
 sidmon  
 noisetracker  
 deltamusic  
 futurecomposer

Natürlich können bereits vorhandene Musikstücke direkt vom Demomaker geladen und abgespielt werden.

Damit Sie Ihre selbsterstellten Demos auch an Freunde und Bekannte weitergeben können, lassen sie sich als lauffähige Programme speichern. Dabei unterscheidet der Demomaker drei unterschiedliche Speicherarten. Erstens ist es möglich, eine vom CLI aus lauffähige Demo zu speichern, die aber durchaus wieder in den Demomaker eingeladen und so jederzeit wieder verändert oder überarbeitet werden kann. Die zweite Möglichkeit ist, eine Demo direkt als bootfähige Diskette zu speichern. Sie benötigen in diesem Fall keine Workbench-Diskette, um die Demo zu starten. Legen Sie einfach die Diskette in das Laufwerk DF0:, und booten Sie neu. Schon wird die Demo geladen. Als letzte Möglichkeit bietet Ihnen der Demomaker das Speichern eines vom CLI lauffähigen Programms, jedoch mit der Einschränkung, daß eine solche Demo nicht mehr mit dem Demomaker verändert oder überarbeitet werden kann. Sie sollten sich deshalb vorher genau überlegen, ob die Demo wirklich nicht mehr überarbeitet werden soll. Allerdings hat dies natürlich auch Vorteile, z.B. wenn Sie eine originelle Demo erstellt haben und nicht möchten, daß andere diese Demo wieder in den Demomaker laden können.



## 2. Installation

Falls der Demomaker auf einer Harddisk installiert werden soll, müssen alle Dateien und Verzeichnisse auf die entsprechende Festplatte kopiert werden - und zwar folgendermaßen:

### 2.1 Installieren auf einer Harddisk

1. Starten Sie Ihren Amiga mit Ihrer Startdiskette oder direkt von der Festplatte, wenn diese autobootfähig ist.
2. Öffnen Sie ein CLI/Shell-Fenster.
3. Legen Sie die Demomakerdiskette in DF0: ein.
4. Tippen Sie "Copy DF0: to Name\_der\_Festplatte: all
5. Der Demomaker ist nun auf Ihrer Festplatte installiert, und kann durch den CLI/Shell-Aufruf "Demomaker" gestartet werden.

### 2.2 Starten von einer Diskette

Wie 1. bis 3., dann mit "Loader" starten oder einfach die Original-Diskette einlegen und den Amiga einschalten - das Programm startet dann automatisch.







### 3. Der Aufbau des Bildschirms

Der Bildschirm, auf dem später Ihre Grafiken und Laufschriften zu sehen sind, ist in drei Teile gegliedert, die getrennt angesteuert werden. Die folgende Abbildung zeigt diese Bereiche im einzelnen:

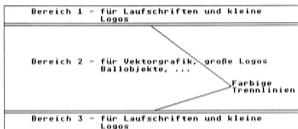


Abb. 1: Die verschiedenen Bereiche

- Bereich 1** Dieser Bereich stellt den oberen Bildschirmrand dar, der mit Laufschriften von 8x8, 16x16 und 32x32 Pixel versehen werden und dabei bis zu 16 Farben annehmen kann. Hier können auch kleinere Logos eingeblendet werden.
- Bereich 2** Hier liegt der Hauptbereich, in dem die wichtigsten und umfangreichsten Effekte wie Vektorgrafik, Sinus-Laufschriften, Bob-Effekte u.a. dargestellt werden können.
- Bereich 3** Dieser kann genau wie Bereich 1 eingesetzt werden. Natürlich ist es möglich, im ersten Bereich ein Logo einzublenden, während im dritten Bereich eine 32x32 Pixel-Laufschrift eingeblendet wird.

Der Bereich 2 kann oben und unten durch eine farbige Trennlinie optisch besser von den übrigen Bereichen getrennt werden.



## 4. Die Bediener-Oberfläche

Über die Oberfläche werden alle zum späteren Demo benötigten Schriften, Grafiken und Sounds geladen. Gleichzeitig dient die Oberfläche als Ausgangspunkt für die folgenden Möglichkeiten:

**TEST DEMO** Wenn Sie bereits eine Demo oder Teile davon erstellt haben, dient dieser Button zum Überprüfen des Ablaufs. Es werden alle Teilstücke in der von Ihnen festgelegten Reihenfolge unter Berücksichtigung der festgelegten Zeiteinteilung angezeigt.

**DEMO MENU** Hier lassen sich bereits erstellte Demos laden und speichern.

**PATTERN EDITOR** Das Herzstück des Demomakers. Hier wird die eigentliche Demo erstellt.

**MUSIC ON/OFF** Dieser Punkt dient zum Ein- bzw. Ausschalten der Musik im Editor, natürlich nur, wenn vorher ein entsprechendes Musikstück geladen wurde.

Zusätzlich gibt es noch eine Statuszeile, in der Sie jederzeit ablesen können, wieviel Speicher noch für eine Demo zur Verfügung steht. Beachten Sie dabei bitte, daß nur Chip-RAM angezeigt wird, denn der Grafik-Chip im Amiga kann nur Speicher ansprechen, der im sogenannten Chip-Memory liegt. Dies ist der Speicher bis 512 KByte bzw. 1 MByte bei einem Amiga mit BigAgnus oder bis 2 MByte bei einem Amiga 3000. Gleichzeitig können Sie auch noch ablesen, welche Musikroutine gerade verwendet wird. Dies ist besonders nützlich, da der Demomaker mehrere Formate unterstützt.

## 4.1 Die Preferenzliste

In der Preferenzliste stehen alle vom Demo benutzten externen Daten wie z.B. Textbildschirme, Musik, Farblisten, Vektorobjekte usw. Mittels der beiden Pfeilsymbole oder den Tasten **[←]** bzw. **[→]** ist es möglich, sich innerhalb dieser Liste zu bewegen. Sie können das leicht an dem blinkenden Farbbalken erkennen, der sich entsprechend mitbewegt. Um nun einen gewünschten Punkt in der Liste auszuwählen, klicken Sie einmal mit der linken Maustaste darauf. Der Punkt ist nun angewählt und kann sofort bearbeitet werden.

Da es beim Erstellen von Demos immer wieder erforderlich ist, neue Objekte o.ä. nachzuladen, wurden zur einfacheren Handhabung folgende Tastaturbelegungen vordefiniert:

Taste	Wirkung
<b>[F1]</b>	Sprung zum Auswahlfeld der Muskroutine
<b>[F2]</b>	Sprung zum Auswahlfeld der Grafiken
<b>[F3]</b>	Sprung zum Auswahlfeld der Texte
<b>[F4]</b>	Sprung zum Auswahlfeld der Textscreen 8
<b>[F5]</b>	Sprung zum Auswahlfeld der Textscreen 32
<b>[F6]</b>	Sprung zum Auswahlfeld der Objekte
<b>[F7]</b>	Sprung zum Auswahlfeld der Vectorballs-objekte
<b>[F8]</b>	Sprung zum Auswahlfeld der Farblisten
<b>[F9]</b>	Sprung zum Anfang der Liste
<b>[F10]</b>	Direktsprung

Der Farbbalken befindet sich z.B. gerade auf "LITTLELOGO". Wenn Sie nun **[F10]** drücken, wird die augenblickliche Position markiert. Bewegen Sie nun den Farbbalken auf eine andere Position, können Sie mit nochmaligem **[F10]** wieder an die vorherige Position, also auf "LITTLELOGO" springen.

Da aber nicht unbedingt jeder Anwender die vordefinierte Einstellung benutzt, gibt es zusätzlich die Möglichkeit, diese frei zu bestimmen. Dazu bewegen Sie den Farbbalken einfach auf die gewünschte



Position und drücken **Shift** und eine der Funktionstasten 1 bis 9. Die neue Position ist damit gespeichert und kann sofort über die entsprechenden Funktionstasten benutzt werden.

## 4.2 Möglichkeiten der Preferenzliste

Jeder der einzelnen Punkte in der Liste wird, je nach ausgewähltem Eintrag, über eine der folgenden Funktionen gesteuert:

EDIT  
CLEAR  
LOAD  
SAVE

Wenn Sie z.B. den Punkt "MUSICROUTINE" angewählt haben, können Sie mit "EDIT" eine der angezeigten Abspielroutinen laden. Die nun folgende Übersicht zeigt Ihnen, bei welchen Punkten der Liste welche Funktionen zur Verfügung stehen und welche Bedeutung diese Funktionen im einzelnen haben.

### MUSICROUTINE:

EDIT	Auswahl der Musikroutine
CLEAR	Löschen der Musikroutine

### MUSIC:

LOAD	Laden eines Musikstückes
CLEAR	Löschen des Musikstückes

### DEMOPATTERN:

EDIT	Gleiche Funktion wie PATTERN-EDITOR in der Kopfzeile
------	--

## Grafiken

**LITTLELOGO 0-7:**

- EDIT                    Editieren der Farben
- LOAD                    Laden eines Logos mit vier Bitplanes und den  
Maßen 320 Pixel Breite und 54 Pixel Höhe.



Beachten Sie bitte, daß hier und auch bei allen anderen Objekten die Anzahl der Bitplanes bzw. die Maße unbedingt eingehalten werden müssen. Farben, die Sie nicht benutzen, können Sie ja einfach schwarz einfärben. Weiterhin müssen solche Grafiken im IFF-Bildformat vorliegen. Speichern Sie daher die gewünschte Grafik immer als Pinsel ab, damit auch die richtige Größe beibehalten wird.

- CLEAR                    Löschen des Logos

**BIGLOGO 0-3:**

- EDIT                    Editieren der Farben
- LOAD                    Wie LITTLELOGO, jedoch müssen die Maße jetzt  
320 Pixel Breite und 158 Pixel Höhe haben.
- CLEAR                    Löschen des Logos

**FONT 32x32:**

- EDIT                    Editieren der Farben
- LOAD                    Laden eines Zeichensatzes mit vier Bitplanes und  
den Maßen: 320 Pixel Breite und 192 Pixel Höhe.
- CLEAR                    Löschen des Zeichensatzes

**FONT 16x16:**

- EDIT Editieren der Farben
- LOAD Laden eines Zeichensatzes mit drei Bitplanes und den Maßen 320 Pixel Breite und 48 Pixel Höhe.
- CLEAR Löschen des Zeichensatzes

**FONT 8x8:**

- EDIT Editieren der Farben
- LOAD Laden eines Zeichensatzes mit drei Bitplanes und den Maßen 320 Pixel Breite und 16 Pixel Höhe.
- CLEAR Löschen des Zeichensatzes

**FONT 16x16 1PL:**

- EDIT Editieren der Farben
- LOAD Laden eines Zeichensatzes mit einer Bitplane und den Maßen 320 Pixel Breite und 48 Pixel Höhe.
- CLEAR Löschen des Zeichensatzes

**VECTORCHARSET:**

- LOAD Laden eines Vektorzeichensatzes. Nur für Zoom-text
- CLEAR Löschen des Vektorzeichensatzes

**SPRITEOBJECT 0-3:**

- EDIT Editieren der Farben
- LOAD Laden eines Sprite-Objektes mit zwei Bitplanes und den Maßen 48 Pixel Breite und 9 Pixel Höhe.
- CLEAR Löschen des Sprite-Objektes

**STARBOBS 0-3:**

- EDIT            Editieren der Farben des Bobs
- LOAD            Laden eines Bob-Objektes mit zwei Bitplanes und den Maßen 128 Pixel Breite und 15 Pixel Höhe.
- CLEAR           Löschen des Bob-Objektes

**VECTORBALLSGFX:**

- EDIT            Editieren der Farben
- LOAD            Laden einer Vektorgrafik für den Effekt "VECTORBALLS" mit vier Bitplanes und den Maßen 320 Pixel Breite und 64 Pixel Höhe.
- CLEAR           Löschen der Vektorgrafik

**BOBEFFECTGFX:**

- EDIT            Editieren der Farben
- LOAD            Laden einer Bob-Grafik für den Effekt "BOBEFFECT" mit drei Bitplanes und den Maßen 160 Pixel Breite und 16 Pixel Höhe.
- CLEAR           Löschen der Bob-Grafik

**Scrolltexte**

**SCROLLTEXT UNTEN:**

- EDIT            Editieren des Textes. Je nach eingestelltem Zeichensatz können Sie hier mehr oder weniger Text eingeben.
- CLEAR           Löschen des Textes
- LOAD            Laden eines bereits gespeicherten Textes
- SAVE            Speichern eines Textes



**SCROLLTEXT OBEN:**

EDIT	Editieren des Textes. Je nach eingestelltem Zeichensatz können Sie hier mehr oder weniger Text eingeben.
CLEAR	Löschen des Textes
LOAD	Laden eines bereits gespeicherten Textes
SAVE	Speichern eines Textes

**SINUSSCROLLTEXT:**

EDIT	Editieren des Textes.
CLEAR	Löschen des Textes
LOAD	Laden eines bereits gespeicherten Textes
SAVE	Speichern eines Textes

**TEXTSCREEN16 0-9:**

EDIT	Editieren des Textes
CLEAR	Löschen des Textes
LOAD	Laden eines bereits gespeicherten Textes
SAVE	Speichern eines Textes

**TEXTSCREEN32 0-9:**

EDIT	Editieren des Textes
CLEAR	Löschen des Textes
LOAD	Laden eines bereits gespeicherten Textes
SAVE	Speichern eines Textes

**TEXTSCREEN8 0-9:**

EDIT	Editieren des Textes
CLEAR	Löschen des Textes

- LOAD           Laden eines bereits gespeicherten Textes  
 SAVE           Speichern eines Textes

## Vektorobjekte

### OBJECT 0-9:

- EDIT           Editieren der Objektfarben. Beachten Sie bitte, daß Vektorobjekte nur acht Farben haben können.  
 LOAD           Laden eines Vektorobjektes  
 CLEAR         Löschen des Vektorobjektes

### BALLSOBJECT 0-9:

- EDIT           Editieren der Objektfarben. Beachten Sie bitte, daß Ballobjekte nur 16 Farben haben können.  
 LOAD           Laden eines Ballobjektes  
 CLEAR         Löschen des Ballobjektes

## Farblisten

### SCRBACKGRCOLS 0-3

- EDIT           Editieren der Hintergrundfarben für Laufschriften  
 CLEAR         Löschen der Hintergrundfarbe  
 LOAD           Laden einer bereits gespeicherten Farbliste  
 SAVE           Speichern einer Farbliste

### BACKGRCOLS 0-7:

- EDIT           Editieren der Hintergrundfarben für Ablaufbefehle  
 CLEAR         Löschen der Hintergrundfarbe



LOAD	Laden einer bereits gespeicherten Farbliste
SAVE	Speichern einer Farbliste

**SCROLLBARCOLS:**

EDIT	Editieren der Farben für die Trennstriche
CLEAR	Löschen der Farben
LOAD	Laden einer bereits gespeicherten Farbliste
SAVE	Speichern einer Farbliste

**ZOOMTEXTCOLS 0-3:**

EDIT	Editieren der Farben für den Effekt "ZOOMTEXT"
CLEAR	Löschen der Farben
LOAD	Laden einer bereits gespeicherten Farbliste
SAVE	Speichern einer Farbliste

**VECTORBALLSCOLS:**

EDIT	Editieren der Farben für den Effekt "FULL-SCREEN" im Zusammenhang mit dem Effekt "VECTORBALLS"
CLEAR	Löschen der Farben
LOAD	Laden einer bereits gespeicherten Farbliste
SAVE	Speichern einer Farbliste

### 4.3 Editieren von Texten im Editor

Zum Editieren von Texten wird in einen separaten Texteditor umgeschaltet. Je nach ausgewählter Textscreen kann das Editorfenster größer oder kleiner ausfallen. Das liegt an der Größe des verwendeten Zeichensatzes. Bei einer Größe von 8x8 Pixeln können Sie viel Text eingeben, bei 32x32 Pixeln dagegen haben Sie nur recht wenig Platz,

um Text einzugeben. Die Menge des einzugebenden Textes ist schließlich durch den Bereich 2 (siehe auch Kapitel 3) begrenzt. Wenn Sie nun Text eingeben wollen, beachten Sie bitte die folgenden Hinweise:

1. Durch die Eingabe von `Return` wird der Text übernommen, und das Programm kehrt auf die Oberfläche zurück.
2. Sie können wie gewohnt den Eingabecursor mit den Pfeil-Tasten innerhalb des Texteditors bewegen.
3. Mit `Help` können Sie zwischen Überschreib- und Einfügemodus wechseln. Den aktuell eingestellten Modus können Sie in der Statuszeile am unteren Ende des Texteditors ablesen.
4. Verwenden Sie bitte nicht `Shift` beim Eingeben von normalen Texten, da diese Taste Sonderzeichen an die aktuelle Cursorposition schreibt, die nur für Laufschriften und kleine Logos in den Bereichen 1 und 3 (siehe auch Kapitel 3) Bedeutung haben.

### 4.3.1 Sonderzeichen im Texteditor

Beachten Sie bitte, daß die im folgenden erklärten Sonderzeichen nur Bedeutung haben, wenn Sie Text für die Bereiche 1 und 3 eingeben wollen. Alle in diesen Bereichen kleinen Buchstaben müssen im Texteditor mit `Shift` eingegeben werden. Sonderzeichen können Sie leicht an der grauen Farbe im Texteditor erkennen.

- sX** Speed - X kann dabei Werte zwischen 0 (stehend) und 8 (schnell) annehmen.
- h** Hold - Die Laufschrift wird einen Moment lang angehalten.
- oX** Choice - Wahl der Laufschrift, entweder X=0 (32x32 Pixel) oder X=1 (16x16 Pixel).



- IX YY** Logo - Einblenden eines "LITTLELOGO". Dabei kann X die Werte 0 bis 7 (Nummer des Logos) annehmen. Zusätzlich kann noch die Zeit in Sekunden angegeben werden, wie lange das Logo dargestellt werden soll. Dabei kann YY Werte von 00 bis 99 annehmen (dieser Parameter ist zwingend).
- mX** Mode - Mit diesem Parameter können Sie einer Laufschrift eine Spiegelung zuweisen. Beachten Sie bitte, daß dies nur bei 32x32-Pixel-Laufschriften möglich ist. X kann dabei die Werte 0 (keine Spiegelung) oder 1 (Spiegelung) annehmen.
- cX** Copperlist - Die Laufschrift kann durch Eingabe dieses Parameters mit einer Copperliste (Farbliste) hinterlegt werden. X gibt dabei die Nummer der Farbliste an und kann die Werte 0 (schwarz) und 1 bis 4 (Farblistennummer) annehmen.
- wX** Wobble - Mit diesem Parameter können Sie einer Laufschrift verschiedene Effekte zuweisen. Beachten Sie bitte, daß dies nur bei 32x32-Pixel-Laufschriften möglich ist. Dabei kann X folgende Werte annehmen:

0	Normal
1	Schrift kontinuierlich um 360° drehen
2	Schrift tanzt (wobbelt)
3	Schrift schrumpfen lassen
4	Zurück auf normale Größe
5	Schrift um 180° drehen
6	Effekt "5" umkehren

- jX** Jump - Mit diesem Parameter können Sie einer Laufschrift verschiedene Effekte zuweisen. Beachten Sie bitte, daß dies nur bei 16x16-Pixel-Laufschriften möglich ist. Dabei kann X die folgenden Werte annehmen:

0	Normal
1	Sprungliste eins
2	Sprungliste zwei
3	Sprungliste drei
4	Sprungliste vier

- b** Baseline - Dieser Parameter wählt den Modus für die Trennlinie zu Bereich 2. Dabei können folgende Zusatzparameter verwendet werden:

Trennlinie	b0 X (X=0-9 - Speed)
Scrolleisten	b1 von rechts nach links
Scrolleisten	b2 von links nach rechts
Scrolleisten	b3 von Mitte nach außen
Farbe	b4 XXX (XXX = \$000 - \$FFF)

Dazu ein Beispiel: "o0 s3 b4 F00 DIES IST EIN TEST". o0 schaltet die 32x32-Pixel-Laufschrift auf mittlere Geschwindigkeit (s3), und b4 F00 schaltet die Trennlinie auf rot.

### 4.3.2 Sonderzeichen bei Sinusscroller

Auch beim Editieren eines Sinusscrollers können Sonderzeichen verwendet werden. Auch hier gilt, daß alle kleingeschriebenen Buchstaben im Texteditor mit `Shift` eingegeben werden müssen. Sonderzeichen können Sie an der grauen Farbe im Texteditor erkennen.

#### Zeichen Bedeutung

- sX** Speed - Hiermit können Sie die Geschwindigkeit des Sinusscrollers festlegen. Dabei kann X die Werte 0 (stehend) bis 4 (schnell) annehmen.
- o** Off - Schaltet den Sinusscroller aus.



- h** Hold - Der Sinusscroller wird einen Moment lang angehalten.

Wenn z.B. mehrere Texte nacheinander im Sinusscroller angezeigt werden sollen, so trennt man die Texte mit dem Befehl "o" voneinander. Bei jedem erneuten Aufruf von "SINUSSCROLLER" wird bis zum nächsten "o" oder, wenn keins mehr folgt, zur Endmarkierung des Textes gesprungen.







## 5. Das Kernstück - der PATTERN-EDITOR

Der umfangreichste Teil und damit das Kernstück des Demomaker ist der PATTERN-EDITOR. Denn erst hier werden alle Effekte getestet, verändert, kombiniert und natürlich auch aufeinander abgestimmt - hier geben Sie Ihrer Demo Ihre ganz persönliche Note und den letzten Schliff, bestimmen die Effekte und geben vor, wie lange die einzelnen Effekte in der fertigen Demo auf den Betrachter wirken sollen.

### 5.1 Die Hauptpunkte des PATTERN-EDITORS

In der Menüleiste sehen Sie die einzelnen Effekte, die durch einen Mausklick aktiviert werden können. In der oberen rechten Ecke wird zur Kontrolle der aktuelle Effekt und die Teilstücknummer angezeigt. Bevor Sie nun endlich anfangen können, hier noch ein paar Informationen vorab.

- Test Demo** Wenn Sie bereits eine Demo oder Teile davon erstellt haben, dient dieser Button zum Überprüfen des Ablaufs. Es werden alle Teilstücke in der von Ihnen festgelegten Reihenfolge unter Berücksichtigung der festgelegten Zeiteinteilung angezeigt.
- Test Pattern** Das gerade erstellte Teilstück (Pattern) wird angezeigt. Die Anzeige dient hauptsächlich zur Kontrolle der eingegebenen Werte.
- Delete Pattern** Löscht das aktuelle Teilstück aus der Demo. Alle folgenden Teilstücke (wenn vorhanden) rücken um die Position auf.

- Insert Pattern** Fügt an die aktuelle Position ein neues Teilstück ein. Alle folgenden Teilstücke (wenn vorhanden) rücken um die Position nach.
- Quit** Hiermit verläßt man den PATTERN-EDITOR und kehrt auf die Hauptoberfläche zurück.
- Pfeil links** Es wird ein Teilstück zurückgeblättert. Die aktuelle Teilstücknummer können Sie oben rechts ablesen.
- Pfeil rechts** Es wird ein Teilstück vorgeblättert.



Auch jetzt ist es noch recht schwierig, etwas auf den Bildschirm zu bringen, da zu den einzelnen Effekten jeweils eine beachtliche Menge von Parametern eingegeben werden müssen. Also - nehmen Sie sich bitte die Zeit, und lesen Sie sich erst einmal alles in Ruhe durch. Sie werden sicher feststellen, daß es eine große Hilfe ist.

## 5.2 Die Teilstückbefehle

Bevor Sie nun die Bedeutung aller Teilstückbefehle kennenlernen, beachten Sie bitte, daß sich alle in Klammern stehenden Namen auf die Preferenzliste beziehen (siehe Kapitel 4.1). Ein Teilstück ist ein Platzhalter für einen bestimmten Effekt. Wenn nun ein solches Teilstück einen Befehl erhalten hat, wird der Befehl im unteren Bildschirmbereich angezeigt.

Hier sind, je nach Befehl, verschiedene Eingaben notwendig. Der rote Text stellt immer reine Kommentare dar, weißer Text kann mit der Maus angeklickt und per Tastatur editiert werden. Bei Zahlenkolonnen können nur die Zahlen- und die Cursortasten benutzt werden. Beachten Sie bitte, daß eine Eingabe innerhalb eines dieser Felder nicht mit `Return` bestätigt werden muß. Mit den Pfeiltasten können Sie den Cursor innerhalb des Eingabefeldes oder zum nächsten be-



wegen. Befindet sich vor einem Eingabefeld das Zeichen "\$", handelt es sich um eine hexadezimale Zahl. Dabei entspricht:

10 = 0A
11 = 0B
12 = 0C
13 = 0D
14 = 0E
15 = 0F

Die Zahlen 00 bis 09 in hexadezimaler Schreibweise haben den gleichen Wert wie die in der dezimalen Schreibweise. In mit Hochkommas angezeigten Bereichen kann man einen Text eingeben. Klicken Sie dazu innerhalb dieses Feldes mit der Maus. In den Moduszeilen kann der Modus durch Mausklick geändert werden. Bei manchen Feldern ist die Eingabe negativer Zahlen möglich, in diesem Fall kann man die Zahl mit  negieren und mit  wieder positiv machen. Beachten Sie dabei bitte, daß die Tasten  bzw.  auf dem Ziffernblock nicht akzeptiert werden.

## Aufstellung aller Befehle

### SCREEN MODE

---

Mit diesem Befehl lassen sich die beiden Scroll-Bereiche (Bereich 1 und 3) und die Musik ein- bzw. ausschalten. Wenn Sie diesen Befehl benutzen, gilt die Einstellung für alle folgenden Teilstücke, bis wieder SCREEN MODE verwendet wird. Zu Beginn einer jeden Demo sind die Musik und die Scroll-Bereiche automatisch eingeschaltet.

<b>SCROLLER1</b>	<b>ON</b>	Scroller oben an
	<b>OFF</b>	Scroller oben aus/ Trennlinie bleibt erhalten
<b>SCROLLER2</b>	<b>ON</b>	Scroller unten an
	<b>OFF</b>	Scroller unten aus/ Trennlinie bleibt erhalten

**MUSIC**      **ON**    Musik an  
                   **OFF**    Musik aus

**BIGLOGO** \_\_\_\_\_

Dieser Befehl zeigt ein großes Logo an.

**LOGONUMBER**      gibt Nummer des Logos an (BIGLOGO 0 bis BIGLOGO 3).

**TIME**                gibt die Dauer des Befehls in Sekunden an (0-99).

**STARS**              Scroll-Sterne werden wie folgt eingeblendet:

<b>NONE</b>	keine Scroll-Sterne
<b>SPR1 RIGHT</b>	Sprite1 von rechts
<b>SPR1 LEFT</b>	Sprite1 von links.

Gleiches gilt für SPR2-SPR4.

**BEGINNEFFECT**      Die folgenden Befehle stehen zur Auswahl:

**MIDDLE**              Logo erscheint in der Mitte

**DIRECTION1-8**      Logo erscheint wie folgt:

<b>1</b>	von rechts
<b>2</b>	aus der oberen rechten Ecke
<b>3</b>	von oben
<b>4</b>	aus der oberen linken Ecke
<b>5</b>	von links
<b>6</b>	aus der unteren rechten Ecke
<b>7</b>	von unten
<b>8</b>	aus der unteren rechten Ecke.



- FLOOD** Logo wird "eingefüllt"  
**ZOOM** Logo wird gezoomt  
**BUMPING 1-3** Logo fällt von oben herein, wobei 0=langsam und 3=schnell ist.  
**ENDEFFECT** Die folgenden Befehle stehen zur Auswahl:  
**MIDDLE** Logo wird in der Mitte ausgeblendet.  
**DIRECTION 1-8** Logo verschwindet wie folgt:

- |   |                          |
|---|--------------------------|
| 1 | nach links               |
| 2 | zur unteren linken Ecke  |
| 3 | nach unten               |
| 4 | zur unteren rechten Ecke |
| 5 | nach rechts              |
| 6 | zur oberen rechten Ecke  |
| 7 | nach oben                |
| 8 | zur oberen linken Ecke   |

- CLEAR 1-3** Siehe Abbildungen.



Abb. 2: Clear 1



Abb. 3: Clear 2

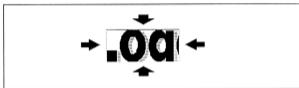


Abb. 4: Clear 3

In dem Feld "MODE" können zusätzlich folgende möglichen Befehle ausgewählt werden:

**NORMAL**                      nur die gerade beschriebenen Effekte

**BACKGRCOLS**

**BACKGRLIST**              Farbliste für Hintergrundfarben (BACKGRCOLS0 - 7)

**EFFECT1**

**BACKGRLIST**              Farbliste für Effekt (BACKGRCOLS 0 - 7)  
**DISTANCE**                gibt den Abstand für den Effekt an (z.B. 0007 oder 0158).

**LITTLELOGO**

Hier gelten die gleichen Möglichkeiten wie bei "BIGLOGO". Nur wird ein kleineres Logo angezeigt (LITTLELOGO 0 - 7).

**VECTORGFX**

Anzeigen eines Vektorobjektes.

<b>OBJECTNR</b>	Nummer des Objektes (OBJECT 0-9)
<b>TIME</b>	Zeit in Sekunden
<b>STARS</b>	Scroll-Sterne können wie folgt eingeblendet werden:

NONE	keine Scroll-Sterne
SPR1 RIGHT	Sprite1 von rechts
SPR1 LEFT	Sprite1 von links.

Gleiches gilt für SRP2-SPR4.

<b>ADD X</b>	Dieser Wert wird zu allen X-Koordinaten des Objektes addiert. Dieser Wert kann auch negativ sein. Es ist damit möglich, ein Objekt zu verschieben.
<b>ADD Y</b>	Wie "ADD X", nur, daß dieser Wert zu allen Y-Koordinaten des Objektes addiert wird.
<b>ADD Z</b>	Wie "ADD X", nur wird dieser Wert zu allen Z-Koordinaten des Objektes addiert wird.
<b>ANGLE X</b>	Startwinkel des Objektes in Grad. Gültige Werte sind: X, Y, Z von 0 bis 360.
<b>ANGLE Y</b>	Wie "ANGLE X"

<b>ANGLE Z</b>	Wie "ANGLE X"
<b>ROTATE</b>	X-Rotationswinkel für das Objekt, an der jeweiligen Achse in Grad. Gültige Werte sind: X, Y, Z von 0 bis 360
<b>ROTATE Y</b>	Wie "ROTATE X"
<b>ROTATE Z</b>	Wie "ROTATE X"

In dem Feld "MODE" können zusätzlich folgende Befehle ausgewählt werden:

<b>NORMAL</b>	Nur die gerade beschriebenen Effekte.
<b>MIRROR</b>	Spiegelung einschalten
<b>BACKGRLIST</b>	Farben der Spiegelung (BACKGRCOLS 0-7)
<b>VECTORSTARS</b>	Vektorstars einschalten
<b>ANGLE X</b>	Startwinkel X in Grad
<b>ANGLE Y</b>	Startwinkel Y in Grad
<b>ANGLE Z</b>	Startwinkel Z in Grad
<b>ROTATE X</b>	Drehwinkel X in Grad
<b>ROTATE Y</b>	Drehwinkel Y in Grad
<b>ROTATE Z</b>	Drehwinkel Z in Grad
<b>STARSPPEED</b>	Geschwindigkeit der Sterne
<b>STARBOBS</b>	Sternobjekt wählen (STARSOBJECT 0-3)
<b>PRIORITY</b>	Priorität der Vektorstars
<b>PIXELSTARS</b>	Normale Sterne einschalten
<b>COLOR 1</b>	Hexadezimale Farbe der Sterne
<b>COLOR 2</b>	Hexadezimale Farbe der Sterne
<b>COLOR 3</b>	Hexadezimale Farbe der Sterne
<b>ROTATE Z</b>	Drehwinkel Z der Sterne
<b>PRIORITY</b>	Sterne vor oder hinter dem Objekt ("IN FRONT"/"BEHIND")





<b>TEXTSCREEN16</b>	Textbildschirm mit (FONT 16x16)
<b>TEXTSCREEN</b>	Nummer des Textbildschirms (TEXTSCREEN16 0-9)
<b>PRIORITY</b>	Textbildschirm vor oder hinter dem Objekt ("IN FRONT"/"BEHIND")
<b>TEXTSCREEN8</b>	Textbildschirm mit (FONT 8x8)
<b>TEXTSCREEN</b>	Nummer des Textbildschirms (TEXTSCREEN 8 0-9)
<b>PRIORITY</b>	Textbildschirm vor oder hinter dem Objekt ("IN FRONT"/"BEHIND")

## VECTORBALLS

---

Anzeigen eines Vektorball-Objekts. Beachten Sie dabei bitte, daß das Objekt schon in "VECTORBALLSGFX" eingeladen sein muß.

<b>OBJECTNR</b>	Nummer des Objekts (BALLSOBJECT 0-9)
<b>TIME</b>	Zeit in Sekunden
<b>STARS</b>	Scroll-Sterne können wie folgt eingeblendet werden:

NONE	keine Scroll-Sterne
SPR1 RIGHT	Sprite1 von rechts
SPR1 LEFT	Sprite1 von links.
...	

Gleiches gilt für SPR2-SPR4.

<b>ADD X</b>	Dieser Wert wird zu allen X-Koordinaten des Objektes addiert. Dieser Wert kann auch negativ sein. Es ist damit möglich, ein Objekt zu verschieben.)
--------------	---

<b>ADD Y</b>	Wie "ADD X", nur, daß dieser Wert wird zu allen Y-Koordinaten des Objektes addiert wird.
<b>ADD Z</b>	Wie "ADD X", nur, daß dieser Wert wird zu allen Z-Koordinaten des Objektes addiert wird.
<b>ANGLE X</b>	Startwinkel des Objektes in Grad. Gültige Werte sind: X, Y, Z von 0 bis 360.
<b>ANGLE Y</b>	Wie "ANGLE X"
<b>ANGLE Z</b>	Wie "ANGLE X"
<b>ROTATE</b>	X-Rotationswinkel für das Objekt, an der jeweiligen Achse in Grad. Gültige Werte sind : X, Y, Z von 0 bis 360
<b>ROTATE Y</b>	Wie "ROTATE X"
<b>ROTATE Z</b>	Wie "ROTATE X"

In dem Feld "MODE" können zusätzlich folgende möglichen Befehle ausgewählt werden:

<b>NORMAL</b>	Nur die gerade beschriebenen Effekte
<b>MIRROR1</b>	Normaler Spiegel
<b>BACKGRLIST</b>	Farben für Spiegel (BACKGRCOLS 0-7)
<b>MIRROR2</b>	Spiegel mit Wasser-Effekt
<b>BACKGRLIST</b>	Farben für Spiegel (BACKGRCOLS 0-7)
<b>FULLSCREEN1</b>	Die untere Scroll-Leiste wird durch den Effekt "MIRROR1" ersetzt. Beachten Sie bitte, daß Sie jetzt keine Laufschrift im Bereich 3 mehr darstellen können.
<b>FULLSCREEN2</b>	Wie "FULLSCREEN2", jedoch wird der Effekt "MIRROR2" verwendet.

**TEXT16SPECIAL**

Textscreen mit FONT 16x16 anzeigen.

<b>TEXTSCREEN</b>	Nummer der Textscreens (TEXTSCREEN16 0-9)
<b>TIME</b>	Zeit in Sekunden.
<b>STARS</b>	Scroll-Sterne können wie folgt eingeblendet werden:

NONE	keine Scroll-Sterne
SPR1 RIGHT	Sprite1 von rechts
SPR1 LEFT	Sprite1 von links.
—	

Gleiches gilt für SPR2-SPR4. In dem Feld "MODE" können zusätzlich folgende möglichen Befehle ausgewählt werden:

<b>SCREEN ONLY</b>	Nur Textscreen
<b>VECTORSTARS</b>	Vektorstars einschalten
-ANGLE X	Startwinkel X in Grad
-ANGLE Y	Startwinkel Y in Grad
-ANGLE Z	Startwinkel Z in Grad
-ROTATE X	Drehwinkel X in Grad
-ROTATE Y	Drehwinkel Y in Grad
-ROTATE Z	Drehwinkel Z in Grad
-STARSPPEED	Geschwindigkeit der Sterne
-STARBOBS	Sternobjekt wählen (STARSOBJECT 0-3)
-PRIORITY	Priorität der Vektorstars
<b>PIXELSTARS</b>	normale Sterne einschalten
-COLOR 1	Hexadezimale Farbe der Sterne
-COLOR 2	Hexadezimale Farbe der Sterne
-COLOR 3	Hexadezimale Farbe der Sterne

- ROTATE Z Drehwinkel Z der Sterne
- PRIORITY Sterne vor oder hinter dem Objekt ("IN FRONT"/"BEHIND")

### TEXT8SPECIAL

Textscreen mit FONT 8x8 anzeigen.

- TEXTSCREEN Nummer der Textscreens (TEXTSCREEN 8 0-9)
- TIME Zeit in Sekunden
- STARS Scroll-Sterne können wie folgt eingeblendet werden:

NONE	keine Scroll-Sterne
SPR1 RIGHT	Sprite1 von rechts
SPR1 LEFT	Sprite1 von links.
...	

Gleiches gilt für SPR2-SPR4. In dem Feld "MODE" können zusätzlich folgende möglichen Befehle ausgewählt werden:

- SCREEN ONLY Nur Textscreen
- VECTORSTARS Vektorstars einschalten
  - ANGLE X Startwinkel X in Grad
  - ANGLE Y Startwinkel Y in Grad
  - ANGLE Z Startwinkel Z in Grad
  - ROTATE X Drehwinkel X in Grad
  - ROTATE Y Drehwinkel Y in Grad
  - ROTATE Z Drehwinkel Z in Grad
  - STARSPEED Geschwindigkeit der Sterne
  - STARBOBS Sternobjekt wählen (STARSOBJECT 0-3)
  - PRIORITY Priorität der Vektorstars



<b>PIXELSTARS</b>	normale Sterne einschalten
-COLOR 1	Hexadezimale Farbe der Sterne
-COLOR 2	Hexadezimale Farbe der Sterne
-COLOR 3	Hexadezimale Farbe der Sterne
-ROTATE Z	Drehwinkel Z der Sterne
-PRIORITY	Sterne vor oder hinter dem Objekt ("IN FRONT"/"BEHIND")

## SINUSSCROLLER

Anzeigen eines Sinusscrollers mit FONT 16x16 1PL. Der benutzte Text ist SINUSSCROLLTEXT. Die Sinusbewegung entsteht aus der Überlagerung zweier Sinuskurven. Deshalb gibt es für jede Kurve jeweils einen Startwert, einen Wert, der zu jeder Spalte des Sinusscrollers addiert wird, und einen Wert, der jeden Bildschirmdurchlauf zum Startwert addiert. Es sind jeweils Werte von 0-\$200 möglich.

<b>START 1</b>	Startwert der ersten Sinuskurve (siehe Beispiel)
<b>START 2</b>	Startwert der zweiten Sinuskurve (siehe Beispiel)
<b>ADD 1</b>	Dieser Wert wird zur ersten Sinuskurve addiert. Dieser Wert kann auch negativ sein. Es ist damit möglich, ein Objekt zu verschieben.
<b>ADD 2</b>	Dieser Wert wird zu allen X-Koordinaten des Objektes addiert. Dieser Wert kann auch negativ sein.
<b>SPEED 1</b>	Geschwindigkeit der ersten Sinuskurve
<b>SPEED 2</b>	Geschwindigkeit der zweiten Sinuskurve

Einige Beispiele hierzu:

START 1/2	ADD 1/2	SPEED 1/2
000,000	002,002	-002,-002
000,000	002,001	-003,-003
000,000	002,002	-004,004
000,000	003,002	-004,002
000,000	000,003	000,-003
000,000	-001,003	000,-003
000,000	000,002	-003,-003
230,230	001,001	000,000
000,000	001,001	010,-010

**BACKGRLIST** Farben für den Hintergrund (BACKGRCOLS 0-7)

In dem Feld "MODE" können zusätzlich folgende möglichen Befehle ausgewählt werden:

**ONE COLOR** einfarbiger Sinusscroller  
**COLOR 1** hexadezimale Farbe

**COPPERLIST1** Farben des Sinusscrollers  
**COLORLIST** Farbliste (BACKGRCOLS 0-7)

**FLASHEN 1** Sinusscroller blinkt in den Farben des folgenden Parameters:  
**COLORLIST** Flashliste (BACKGRCOLS 0-7)

**3 COLORS** dreifarbiges Sinusscroller (3D-Effekt)  
**COLOR 1** hexadezimale Farbe 1  
**COLOR 2** hexadezimale Farbe 2  
**COLOR 3** hexadezimale Farbe 3



<b>COPPERLIST2</b>	Farben im Rand des Sinusscrollers (Outline-Effekt)
<b>COLORLIST</b>	Farbliste (BACKGRCOLS 0-7)
<b>COLOR 1</b>	hexadezimale Farbe 1
<b>COLOR 2</b>	hexadezimale Farbe 2

<b>FLASHEN 2</b>	Der Rand des Sinusscroller blinkt in den Farben des folgenden Parameters (Outline-Effekt):
<b>COLORLIST</b>	Flashliste (BACKGRCOLS 0-7)
<b>COLOR 1</b>	hexadezimale Farbe 1
<b>COLOR 2</b>	hexadezimale Farbe 2

## ZOOMTEXT

---

Eine Textzeile wird gezoomt. Beachten Sie bitte, daß der Zeichensatz "VECTORCHARSET" vorher geladen werden muß.

<b>COLORLIST</b>	Farben im Text (ZOOMTEXTCOLS 0-3)
<b>ZOOMTEXT</b>	Hier kann der Zoomtext eingegeben werden (10 Zeichen Text).
<b>COLOR1</b>	hexadezimale Randfarbe 1
<b>COLOR2</b>	hexadezimale Randfarbe 2

<b>ZOOMMODE</b>	Die folgenden Zoom-Möglichkeiten können genutzt werden:
0	Text wird von der Mitte zu den Rändern gezoomt
1	Wie 1, jedoch umgekehrt
2	Von den Rändern zur Mitte und zurück
3	Wie 2, jedoch umgekehrt

## TEXTSCR32

Ein Textscreen wird mit dem Zeichensatz FONT 32x32 angezeigt.

**TEXTSCREEN** Nummer des Textscreens (TEXTSCREEN32 0-9).

**TIME** gibt die Dauer des Befehls in Sekunden an (0-99)

**STARS** Scroll-Sterne können wie folgt eingeblendet werden:

<b>NONE</b>	keine Scroll-Sterne
<b>SPR1 RIGHT</b>	Sprite1 von rechts
<b>SPR1 LEFT</b>	Sprite1 von links
<b>SPR2 RIGHT</b>	Sprite2 von rechts
<b>SPR2 LEFT</b>	Sprite2 von links
<b>SPR3 RIGHT</b>	Sprite3 von rechts
<b>SPR3 LEFT</b>	Sprite3 von links
<b>SPR4 RIGHT</b>	Sprite4 von rechts
<b>SPR4 LEFT</b>	Sprite4 von links

**BEGINEFFECT** die folgenden Befehle stehen zur Auswahl:

**MIDDLE** Text erscheint in der Mitte

**DIRECTION 1-8** Text erscheint wie folgt:

- |   |                          |
|---|--------------------------|
| 1 | nach links               |
| 2 | zur unteren linken Ecke  |
| 3 | nach unten               |
| 4 | zur unteren rechten Ecke |
| 5 | nach rechts              |
| 6 | zur oberen rechten Ecke  |
| 7 | nach oben                |
| 8 | zur oberen linken Ecke   |





<b>FLOOD</b>	Text wird "eingefüllt"
<b>ZOOM</b>	Text wird gezoomt
<b>BUMPING 1-3</b>	Text fällt von oben herein, wobei 0=langsam und 3=schnell ist.
<b>ENDEFFECT</b>	die folgenden Befehle stehen zur Auswahl:
<b>MIDDLE</b>	Text wird in der Mitte ausgeblendet
<b>DIRECTION 1-8</b>	Text verschwindet wie folgt:

1	nach links
2	zur unteren linken Ecke
3	nach unten
4	zur unteren rechten Ecke
5	nach rechts
6	zur oberen rechten Ecke
7	nach oben
8	zur oberen linken Ecke

**CLEAR 1-3**     Siehe Abbildungen

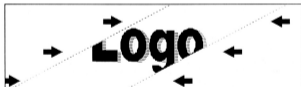


Abb. 5: Clear 1

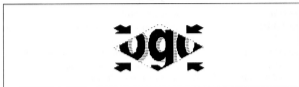


Abb. 6: Clear 2

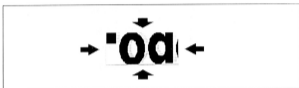


Abb. 7: Clear 3

In dem Feld "MODE" können zusätzlich folgende möglichen Befehle ausgewählt werden:

**NORMAL**

Nur die gerade beschriebenen Effekte.

**BACKGRCOLS****BACKGRLIST**

Farbliste für Hintergrundfarben (BACKGRCOLS 0 - 7)

**EFFECT1****BACKGRLIST**

Farbliste für Effekt (BACKGRCOLS 0 - 7)

**DISTANCE**

Gibt den Abstand für den Effekt an. (z.B. 0007 oder 0158)

**TEXTSCR16**

---

Ein Textscreen wird mit dem Zeichensatz FONT 16x16 angezeigt.

**TEXTSCREEN** Nummer des Textscreens (TEXTSCREEN16 0-9). Alles folgende entspricht dem Befehl TEXTSCR32!

**TEXTSCR8**

---

Ein Textscreen wird mit dem Zeichensatz FONT 8x8 angezeigt.

**TEXTSCREEN** Nummer des Textscreens (TEXTSCREEN8 0-9). Alles folgende entspricht dem Befehl TEXTSCR32!

**TEXTLINE32**

---

Eine Textzeile wird mit dem Zeichensatz FONT 32x32 angezeigt.

**LINENUMBER** Nummer der Bildschirmzeile (0-4).

**LINETEXT** Hier kann der Text eingegeben werden (10 Zeichen Text). Alles übrige entspricht dem Befehl TEXTSCR32!

**TEXTLINE16**

---

Eine Textzeile wird mit dem Zeichensatz FONT 16x16 angezeigt.

**LINENUMBER** Nummer der Bildschirmzeile (0-8)

**LINETEXT** Hier kann der Text eingegeben werden (20 Zeichen Text). Alles übrige entspricht dem Befehl TEXTSCR32!

## TEXTLINE8

---

Eine Textzeile wird mit dem Zeichensatz FONT 8x8 angezeigt.

**LINENUMBER** Nummer der Bildschirmzeile (0-18)

**LINETEXT** Hier kann der Text eingegeben werden (40 Zeichen Text). Alles übrige entspricht dem Befehl TEXTSCR32!

## PLASMA

---

Der Plasmaeffekt ist eigentlich nur ein Sinusscrolling von Farblisten. Wird kein Text eingegeben, siehe auch Beispiel in Kapitel 9.1, kann man sehr schöne Farbverläufe erzielen.

Wird zu diesem Effekt ein Textscreen eingeblendet, so erscheint der Text in den zu scrollenden Farben des Plasmaeffektes. Dieser ist aber nur dann zu sehen, wenn nicht die Farbe schwarz angezeigt wird.

**TIME** Zeit in Sekunden

**COLORLIST** Farbliste, die für den Effekt verwendet wird

Der Plasmaeffekt entsteht aus der Überlagerung zweier Sinuskurven sowohl in X- als auch in Y-Richtung.

<b>X-ADD1</b>	<b>Addwert 1 für X</b>
X-ADD2	Addwert 2 für X
X-SPEED 1	Geschwindigkeit 1 für X
X-SPEED 2	Geschwindigkeit 2 für X
Y-ADD1	Addwert 1 für Y
Y-ADD2	Addwert 2 für Y
Y-SPEED 1	Geschwindigkeit 1 für Y
Y-SPEED 2	Geschwindigkeit 2 für Y



Einige Beispiele:

X-ADD 1/2	X-SPEED 1/2	Y-ADD 1/2	Y-SPEED 1/2
004,004	004, 004	004,004	004, 004
008,010	004, 006	-004,008	008,-008
008,010	014, 006	-004,008	028,-008
004,004	008, 008	000,000	000, 000
004,004	008, 008	020,020	010, 010
008,008	-004, 008	050,020	-008,-008
004,004	-008,-008	256,010	000, 004
004,004	-008,-008	020,010	008,-004
004,004	256, 004	004,014	256, 004

In dem Feld "MODE" können zusätzlich folgende Befehle ausgewählt werden:

<b>NORMAL</b>	Nur Plasma
<b>TEXTSCREEN16</b>	Textbildschirm mit FONT 16x16 1PL
<b>TEXTSCREEN</b>	Textscreennummer (TEXTSCREEN16 0-9)
<b>LINEEFFECT</b>	
<b>TIME</b>	Zeit in Sekunden.
<b>STARS</b>	Scroll-Sterne können wie folgt eingeblendet werden:
<b>CLEARNR</b>	gibt an, wieviele Linieneffekte erhalten bleiben (0-15)

Der Linieneffekt besteht aus der Überlagerung dreier Sinuskurven sowohl in X- als auch in Y-Richtung. Für die "Start"-, "Add"- und "Speed"-Werte sind Zahlen von 0 bis \$200 möglich.

Wenn für X-START der Wert 0 und für Y-START der Wert \$80 eingesetzt wird, erhöht man Sinus in X-Richtung und Cosinus in Y-Richtung. In diesem Fall entstehen dann Kreise. Wenn der Linieneffekt

zusammenhängend erscheinen soll, müssen sowohl die "XADD"-Werte und "YADD"-Werte durch \$8 teilbar sein. (d.h. Werte wie

-\$200, ..., -\$10, -\$8, 0, \$8, \$10, \$18, \$20, \$28, \$30, ..., \$200)

<b>XSTART 1</b>	Startwert für X1
<b>XSTART 2</b>	Startwert für X2
<b>XSTART 3</b>	Startwert für X3
<b>XADD 1</b>	Dieser Wert wird zu allen X-Koordinaten des Objektes addiert. Dieser Wert kann auch negativ sein. Es ist damit möglich, ein Objekt zu verschieben.
<b>XADD 2</b>	Wie "XADD 1"
<b>XADD 3</b>	Wie "XADD 1"
<b>XSPEED 1</b>	Geschwindigkeit. Dieser Wert kann auch negativ sein.
<b>XSPEED 2</b>	Wie "XSPEED 1"
<b>XSPEED 3</b>	Wie "XSPEED 1"
<b>YSTART 1</b>	Wie "XSPEED 1"
<b>YSTART 2</b>	Wie "XSPEED 1"
<b>YSTART 3</b>	Wie "XSPEED 1"
<b>YADD 1</b>	Dieser Wert wird zu allen Y-Koordinaten des Objektes addiert. Dieser Wert kann auch negativ sein. Es ist damit möglich, ein Objekt zu verschieben.
<b>YADD 2</b>	Wie "YADD 1"
<b>YADD 3</b>	Wie "YADD1"
<b>YSPEED 1</b>	Geschwindigkeit. Dieser Wert kann auch negativ sein.



**YSPEED 2**                    Wie "YSPEED 1"

**YSPEED 3**                    Wie "YSPEED 1"

Dazu wieder einige Beispiele:

Effect	XSTART 1-3	XADD 1-3	XSPEED 1-3
0	000,000,000	048,100,048	005,002,003
1	000,000,000	048,080,048	005,000,000
2	000,000,000	008,008,018	005,005,005
3	000,000,000	008,008,088	003,003,003

Effect	YSTART 1-3	YADD 1-3	YSPEED 1-3
0	080,080,080	048,100,048	005,002,003
1	080,080,080	048,080,048	005,000,000
2	080,080,080	008,008,018	005,005,005
3	080,080,080	008,008,088	003,003,003

**LINECOLOR**                hexadezimale Farbe der Linien

In dem Feld "MODE" können zusätzlich folgende Befehle ausgewählt werden:

**NORMAL**

**TXSCREEN16**                Textbildschirm mit (FONT 16x16)

**TEXTSCREEN**                Nummer des Textbildschirms  
(TEXTSCREEN16 0-9)

**TXSCREEN8**                 Textbildschirm mit (FONT 8x8)

**TEXTSCREEN**                Nummer der Textbildschirms  
(TEXTSCREEN8 0-9)

## BOBEFFECT

Es werden die Bobs aus BOBEFFECTGFX benutzt.

**BOBNUMBER** gibt an, welcher 16x16-Bob aus BOBEFFECTGFX genommen wird (0-9)

**BOBANIM** falls 0, keine Animation, sonst gibt der Wert an, wieviele folgende Bobs nach BOBNUMBER angezeigt werden sollen.

Wenn der Bob-Effekt zusammenhängend sein soll, müssen sowohl die XADD-Werte und YADD-Werte durch \$8 teilbar sein, d.h. Werte wie: -\$200,....,-\$20,-\$10,0,\$10,\$20,\$30,\$40,\$50,\$60,....,\$200)

START-, ADD- und SPEED-Werte wie bei LINEEFFECT. Auch hierzu wieder einige Beispiele:

Effect	XSTART 1-3	XADD 1-3	XSPEED 1-3
0	000,000,000	-020,010,010	005,002,003
1	000,000,000	020,010,010	-008,002,003
2	000,000,000	020,008,008	008,-005,004
3	000,000,000	100,100,010	006,005,-004
4	000,000,000	110,110,010	006,005,-004
5	000,000,000	010,010,010	003,003,003
6	000,000,000	020,010,010	005,005,005

Effect	YSTART 1-3	YADD 1-3	YSPEED 1-3
0	080,080,080	-020,010,010	005,002,003
1	080,080,080	020,010,010	-008,002,003
2	080,080,080	020,008,008	-008,005,004
3	080,080,080	100,100,010	006,005,-004
4	080,080,080	110,110,010	006,005,004
5	080,080,080	010,010,010	003,003,003
6	080,080,080	030,010,020	005,005,005





In dem Feld "MODE" können zusätzlich folgende Befehle ausgewählt werden:

#### NORMAL

**TXSCREEN16**           Textbildschirm mit (FONT 16x16)  
**TEXTSCREEN**        Nummer des Textbildschirms  
 (TEXTSCREEN16 0-9)

**TXSCREEN8**           Textbildschirm mit (FONT 8x8)  
**TEXTSCREEN**        Nummer der Textbildschirms  
 (TEXTSCREEN8 0-9)

#### PIXELEFFECT

Wenn der Pixeleffekt zusammenhängend sein soll, müssen sowohl die XADD-Werte als auch die YADD-Werte durch \$4 teilbar sein, d.h. Werte wie: -\$200,....,-\$8,-\$4,0,\$4,\$8,\$c,\$10,\$14,\$18,....,\$200)

START-, ADD- und SPEED-Werte wie bei LINEEFFECT. Beispiele:

Effect	XSTART 1-3	XADD 1-3	XSPEED 1-3
0	000,000,000	104,004,004	-004,003,002
1	000,000,000	018,004,004	004,002,002

Effect	YSTART 1-3	YADD 1-3	YSPEED 1-3
0	080,080,080	104,004,004	-004,002,002
1	080,080,080	018,004,004	004,002,002

**LINECOLOR**           hexadezimale Farbe der Punkte

In dem Feld "MODE" können zusätzlich folgende Befehle ausgewählt werden:

## NORMAL

**TEXTSCREEN16**      Textbildschirm mit (FONT 16x16)  
**TEXTSCREEN**        Nummer des Textbildschirms  
 (TEXTSCREEN16 0-9)

**TEXTSCREEN8**        Textbildschirm mit (FONT 8x8)  
**TEXTSCREEN**        Nummer des Textbildschirms  
 (TEXTSCREEN 8 0-9)

## Erläuterungen

### *Hexadezimale Farbeingabe*

Eine Farbe setzt sich aus den RGB-Komponenten zusammen, d.h. man kann den Rot-, Grün- und Blauanteil einzeln festlegen. Jede Komponente kann einen Wert von 0 bis 15 annehmen, der hexadezimal eingegeben wird. Also kann je Komponente ein Wert von 0 bis F eingegeben werden (0 bis 9 und A bis F). Hierzu einige Beispiele:

33f	helles Blau
fff	weiß
888	grau
f0f	rosa
c08	lila
a00	ziegelrot
eca	beige
080	laubgrün
f09	pink
f9f	hellrosa
ff0	vollgelb
00f	vollblau
0f0	vollgrün
f00	vollrot
666	steingrau
ddd	silber
f00	gold
683	froschgrün
247	königsblau



## 5.3 Ändern der Farbenlisten

Sollten Sie schon einmal mit dem Farbrequester von DPaint gearbeitet haben, wird Ihnen die Erstellung eigener Farblisten nicht schwerfallen. Klicken Sie eine der Farblisten im Hauptmenü an, und wählen Sie "EDIT" in der Menüleiste aus.

Sie sehen nun einen Screen, der sich aus drei Bereichen zusammensetzt:

1. Der Funktionsteil, in dem Sie alle Eingaben für Farbverläufe machen.
2. Der Farbteil, in dem Sie den Farbverlauf sehen.
3. Das Spektrum. Dies befindet sich unterhalb der blinkenden Linie. Hier kann man die gesamte Reichweite des Farbverlaufs sehen.

Um nun einen eigenen Verlauf zu erstellen, gehen Sie folgendermaßen vor:

Die beiden blinkenden eckigen Klammern stellen den jeweils gültigen Ausgangspunkt dar. Bewegen Sie diesen nun mit der Maus auf eine beliebige Position innerhalb des Farbteils. Als nächstes wählen Sie mit Hilfe der RGB-Regler eine Farbe aus. Stellen Sie nun die beiden blinkenden Klammern auf eine neue Position. Wenn Sie nun im Funktionsteil "VERLAUF" anklicken, sehen Sie, daß sich der Mauszeiger geändert hat. Ein kleines "TO" weist Sie darauf hin, daß Sie jetzt die Zielfarbe für den Verlauf auswählen sollen. Klicken Sie dazu eine beliebige Farbe innerhalb des Farbteils an.

Mit den Funktionen "Pfeil rauf" und "Pfeil runter" können Sie einen weiteren Teil des Spektrums anwählen. Die restlichen Funktionen haben folgende Bedeutung:

<b>RESET</b>	Originalfarben wiederherstellen
<b>CLEAR</b>	Alle Farben löschen
<b>COPY</b>	Aktuelle Farbe auf Zielfarbe kopieren
<b>OK</b>	Farben übernehmen, Rückkehr ins Hauptmenü



## 6. Das Demo-Menü

Damit Sie Ihre selbsterstellten Demos auch an Freunde und Bekannte weitergeben können, lassen sich diese als lauffähige Programme und natürlich auch als Bootintro speichern. (Sie können diese Diskette direkt booten.) Im einzelnen gibt es folgende Unterscheidungen:

### **SAVE FINAL**

Speichern eines vom CLI aus lauffähigen Demos. Es wird nach Pfad und Name der Demo gefragt, z.B. "DF0:TESTDEMO". Wenn ein fertig gespeichertes Demo vom CLI aus gestartet werden soll, so sollte man den CLI-Befehl RUNDEMO benutzen, der sich auf der Diskette befindet.

Der Aufruf wird dann z.B. so gemacht: RUNDEMO DF0:TESTDEMO. Wenn nicht genug Speicher vorhanden ist oder das Demo unter dem angegebenen Namen nicht existiert, bricht der RUNDEMO-Befehl automatisch ab.

### **SAVE SHORT**

Wie SAVE FINAL, jedoch mit dem Unterschied, daß das abgespeicherte Demo wieder in den Demomaker eingeladen werden und so wieder verändert werden kann. Das gespeicherte Demo ist aber dennoch vom CLI aus lauffähig.

### **LOAD SHORT**

Laden eines mit SAVE SHORT gespeicherten Files zur Weiterverarbeitung mit dem Demomaker. Beachten Sie bitte, daß dabei alle bereits im Demomaker bearbeiteten Daten gelöscht werden.

**SAVE BOOTBLOCK** Ein lauffähiges Demo wird als Bootintro auf eine Diskette geschrieben. Die Diskette sollte vorher neu formatiert werden, da sonst ein Datenverlust möglich ist!

Es wird nach dem Pfad gefragt: DF0-DF3. Wenn eine Diskette mit SAVE BOOTBLOCK behandelt wurde, kann sie danach als ganz normale Diskette verwendet werden, wobei natürlich weniger Platz ist, da das Demo einige der vorderen Spuren belegt.

**FORMAT** Eine Diskette wird neu formatiert, alle Daten gehen dabei verloren. Es wird nach dem Pfad und dem Namen gefragt, z.B. "DF0:DEMO-DISK"

**EXIT** Zurück ins Hauptmenü

## 7. Verwendung eigener Grafiken

Eigene Grafiken können einfach eingeladen werden. Die einzige Voraussetzung ist, daß die Bilder im IFF-Grafikformat vorliegen und als Pinsel gespeichert wurden. Dabei ist auch auf die korrekte Anzahl von Bitplanes und die Größe zu achten. Die Grafiken, die sich auf der Demomaker-Diskette befinden, können jederzeit in ein beliebiges Malprogramm eingeladen und nach Belieben verändert werden.

Dazu ein Beispiel:

Sie haben ein eigenes BIGLOGO mit DPaint erstellt und möchten dieses nun im Demomaker verwenden. Um nun auch die gewünschte Größe von 320x158 Pixel zu erhalten, schalten Sie erst einmal im ganz rechten Menü von DPaint die Funktion "Koordinaten" ein.

Wählen Sie als nächstes die Pinselgreifen-Funktion in der Seitenleiste aus. Bewegen Sie den Mauszeiger auf die linke obere Ecke der auszuscheidenden Grafik. Drücken Sie die linke Maustaste und halten sie diese gedrückt. Bewegen Sie nun die Maus auf die untere rechte Ecke der Grafik.

In der Titelleiste von DPaint können Sie nun die aktuelle Größe ablesen. Lassen Sie die linke Maustaste aber erst los, wenn die exakte Größe von 320x158 Pixel erreicht ist, da der Demomaker die Grafik sonst nicht akzeptiert.

Jetzt brauchen Sie die Grafik nur noch als Pinsel zu speichern. Dies ist besonders wichtig, da sonst die komplette Bildgröße, also 320x256 Pixel, gespeichert wird.

Hier eine Aufstellung aller verwendeten Formate:



Beachten Sie bitte, daß auch bei allen anderen Objekten die Anzahl der BitPlanes bzw. die Maße unbedingt eingehalten werden müssen. Farben, die Sie nicht benutzen, können Sie ja einfach schwarz einfärben. Weiterhin müssen solche Grafiken im IFF-Bildformat vorliegen. Speichern Sie daher die gewünschte Grafik *immer* als Pinsel ab.

LITTLELOGO	vier Bitplanes, 320 Pixel Breite und 54 Pixel Höhe
BIGLOGO	vier Bitplanes, 320 Pixel Breite und 158 Pixel Höhe
FONT 32x32	vier Bitplanes, 320 Pixel Breite und 192 Pixel Höhe
FONT 16x16	drei Bitplanes, 320 Pixel Breite und 48 Pixel Höhe
FONT 8x8	drei Bitplanes, 320 Pixel Breite und 16 Pixel Höhe
FONT 16x16 1PL	eine Bitplane, 320 Pixel Breite und 48 Pixel Höhe
SPRITEOBJECT	zwei Bitplanes, 48 Pixel Breite und 9 Pixel Höhe
STARBOBS	zwei Bitplanes, 128 Pixel Breite und 15 Pixel Höhe
VEKTORBALLSGFX	vier Bitplanes, 320 Pixel Breite und 64 Pixel Höhe
BOBEFFECTGFX	drei Bitplanes, 160 Pixel Breite und 16 Pixel Höhe.





## 8. Musikroutinen

Sollten Sie mit den Musikprogrammen SoundTracker, NoiseTracker oder einem anderen der unten aufgeführten Programme bereits Erfahrung gesammelt haben, kommt Ihnen das jetzt zugute. Denn der Demomaker kann bereits vorhandene Musikstücke direkt laden und abspielen.

Auf der Ihnen vorliegenden Diskette befinden sich bereits Abspielroutinen für die folgenden Musikprogramme:

bp soundmon  
mark ii  
sidmon  
noisetracker  
deltamusic  
futurecomposer



Bitte beachten Sie, daß - bevor Sie Musikstücke laden - Sie *vorher* die richtige Abspielroutine eingestellt haben. Wenn Musikdaten eingeladen werden und es sich um eine zur eingestellten Musikroutine nicht kompatible Musik handelt, kann der Demomaker dies nicht erkennen, und es kann zu einem Absturz kommen.





## 9. Erstellen von eigenen Demos

Auf den nun folgenden Seiten lernen Sie den Umgang mit dem Demomaker kennen und natürlich auch die Erstellung von Demos.

### Beispiel 1:

Sollten Sie den Demomaker noch nicht geladen haben, holen Sie dies jetzt bitte nach. Anschließend befinden sich im Hauptmenü, in dem alle in der späteren Demo verwendeten Objekte, Schriften usw. geladen werden.

Das Laden von Objekten wird nun einmal gezeigt, und im weiteren Verlauf als bekannt vorausgesetzt. Weiterhin wird vorausgesetzt, daß sich die Demomaker-Diskette in DF0: befindet.

### 9.1 Vorbereitungen zum Demo

Klicken Sie mit der Maus (also einmal mit der linken Maustaste) in der Preferenzliste auf "LITTLELOGO 0". Wählen Sie in der Menüzeile den Punkt "laden" an. In der folgenden Menüzeile klicken Sie bitte auf "DF0:". Es wird nun automatisch in das richtige Verzeichnis gewechselt. Klicken Sie hier bitte "LITTLELOGO 1" an. Abschließend wählen Sie in der Menüzeile "LOAD" an. Sie kehren zurück in das Hauptmenü und sehen in der Zeile "LITTLELOGO 0", das soeben geladene Logo.

<b>LITTLELOGO1</b>	Laden Sie "LITTLELOGO 3 RSI".
<b>BIGLOGO 0</b>	Laden Sie "BIGLOGO DATA 2"
<b>FONT 32x32</b>	Laden Sie "PIX32.MAGIC"
<b>FONT 16x16</b>	Laden Sie "PIX16.DANGER"
<b>FONT 16x16 1PL</b>	Laden Sie "PIX16.1PL.NORMALFONT"
<b>VECTORCHARSET</b>	Laden Sie "VECTORFONT1"

<b>SPRITEOBJECT 0</b>	Laden Sie "SPRITE.POINTS"
<b>SPRITEOBJECT 1</b>	Laden Sie "SPRITE.STAR"
<b>STARBOBS 0</b>	Laden Sie "STARBOB.BOX2"
<b>STARBOBS 1</b>	Laden Sie "STARBOB.PACMAN"
<b>VECTORBALLSGFX</b>	Laden Sie "VECTORBALLS"
<b>BOBEFFECTSGFX</b>	Laden Sie "BOBEFFECTSBOB4"

Geben Sie nun bei "SCROLLTEXT OBEN" folgenden Text ein. Beachten Sie dabei bitte, daß Buchstaben, die klein geschrieben sind, in Verbindung mit **Shift** eingegeben werden müssen.

Klicken Sie auf "SCROLLTEXT OBEN". Wählen Sie jetzt in der Menüleiste "EDIT" an. Sie befinden sich nun im Texteditor. Schreiben Sie den Text ohne Zeilenschaltung ab und am Ende bestätigen Sie mit **Return**.

```
m1c0b3 c0 s5w1GUTEN TAG! hW0h.....WIR, DIE FA. DATA BECKER AUS
DOESSELDORF, MOECHTEN IHNEN AN DIESER STELLE ETWAS VOELLIG NEUES
VORSTELLEN.....DATA
BECKER.....SEIGT!.....hh.....DEN ULTIMATIVEN
DEMOMAKER.....WIERSTELT VON.....10 05 11 05 10 05 11 05 10 99
```

**Erklärung:** m1 schaltet für die Laufschrift die Bodenspiegelung ein. Man hat dann den Eindruck, als würde die Laufschrift etwas über einem Spiegel schweben. c0 bewirkt, daß der Spiegel schwarz ist. b3 schaltet die farbige Trennlinie oben ein. Daneben werden die Farben von der Mitte zu den Rändern gecyled. o0 wählt den 32x32 Pixel-Font. s5 legt die Geschwindigkeit für die Laufschrift fest. w1 schließlich sorgt dafür, daß der Schriftzug "GUTEN TAG!" während dieser von rechts hereingescrollt wird, um 360° gedreht wird. Hinter dem "GUTEN TAG!" folgt ein h, das den Schriftzug einen Moment lang anhält. Danach wird die Drehung wieder ausgeschaltet und noch einen Moment gewartet. Jetzt folgt der eigentliche Text der Laufschrift.



Am Ende stehen noch die Befehle "l0 05", "l1 05", l0 05", "l1 05" und "l0 99", die dafür sorgen, daß die Logos 0 und 1 in abwechselnder Reihenfolge jeweils fünf Sekunden angezeigt werden. Als letztes wird "LITTLELOGO 1" schließlich für 99 Sekunden angezeigt.

Geben Sie nun den folgenden Text bei "SCROLLTEXT UNTEN" ein. Beachten Sie hierbei auch, daß Buchstaben, die klein geschrieben sind, in Verbindung mit **[Shift]** eingegeben werden müssen.

```
b3 o1 s3 DATA BECKER.....l0
05.....DATA BECKER.....l1 99
```

**Erklärung:** b3 schaltet auch hier die farbige Trennlinie ein. 01 wählt den 16x16 Pixel-Font und s3 gibt die Scrollgeschwindigkeit an. l1 05 läßt das "LITTLELOGO 0" für fünf Sekunden einblenden. l1 99 blendet schließlich das "LITTLELOGO 1" für 99 Sekunden ein.

Geben Sie nun bitte den Sinusscrolltext ein.

MIT DIESEM DEMOMAKER HABEN SIE DIE MOEGlichkeit, AUF EINFACHSTE ART UND WEISE IHRE GANZ PERSOENLICHEN DEMOS ZU ERSTELLEN.....

Wählen Sie jetzt bitte "TEXTSCREEN16 0" an und geben Sie

```
DER
DATA BECKER
DEMOMAKER
```

ein. Zentrieren Sie die Eingabe, damit es später in der Demo noch besser aussieht. Drücken Sie einmal **[Help]**, um in den Einfügemodus zu gelangen.

Bevor Sie nun endlich anfangen können, Ihre erste eigene Demo zu erstellen, laden Sie bitte noch die folgenden Objekte:

<b>OBJECT 0</b>	ANALIZER1.VECA
<b>OBJECT 1</b>	SPACESHIP1.NORA
<b>BALLSOBJECT 0</b>	FEDER4.BALA
<b>BALLSOBJECT 1</b>	ROEHRE.BALA
<b>BALLSOBJECT 2</b>	STARFIGHT.BALA

## 9.2 Der Pattern-Editor

Um nun das eigentliche Demo editieren zu können, wählen Sie bitte in der Menüleiste "PATTERN-EDITOR" an. Sie sehen nun eine neue Menüleiste, die in einzelne Funktionen gegliedert ist. In der oberen rechten Ecke kann man ablesen, in welchem Teilstück man sich zur Zeit befindet. Zur einfacheren Verständigung werde ich diese Zeile im folgenden "Infozeile" nennen.

Beginnen Sie nun die Demo, indem Sie in der Menüleiste "SCREENMODE" auswählen. Klicken Sie jede der drei Zeilen solange an, bis in allen "ON" steht. Schließlich soll ja auch der Scrolltext sofort eingeblendet werden. Klicken Sie nun "VECTORBALLS" in der Menüleiste an. Sie sehen schon, daß ein neues Teilstück eingeblendet wurde und in der Infozeile "Pattern 02 VECTORBALLS" steht.

Übertragen Sie nun die Werte aus den folgenden Tabellen in den Editor. Beachten Sie dabei bitte, daß Sie häufig eine neue Funktion in der Menüleiste auswählen müssen.

```
VECTORBALLS - PATTERN 1
OBJECTNR: 0000
TIME:      0005
STARS:     SPRI RIGHT

ADD X: 0110    ANGLE X: 0030    ROTATE X: 0001
ADD Y: 0050    ANGLE Y: 0020    ROTATE Y: 0005
ADD Z: 0060    ANGLE Z: 0030    ROTATE Z: 0003

MODE: FULLSCREEN2
```



## VECTORBALLS - PATTERN 2

OBJECTNR: 0001

TIME: 0005

STARS: SPRI RIGHT

ADD X: 0110

ANGLE X: 0030

ROTATE X: 0001

ADD Y: 0050

ANGLE Y 0020

ROTATE Y 0005

ADD Z: 0060

ANGLE Z 0030

ROTATE Z 0003

MODE: FULLSCREEN2

## BOBEFFECT - PATTERN 3

TIME: 0006

STARS: SPRI RIGHT

BOBNUMBER: 0000

BOBANIM: 0003

XSTART1: \$0112

XADD1: \$0012

XSPEED1: \$0002

XSTART2: \$0114

XADD2: \$0014

XSPEED2: \$0004

XSTART3: \$0116

XADD3: \$0016

XSPEED3: \$0006

YSTART1: \$0118

YADD1: \$0018

YSPEED1: \$0008

YSTART2: \$0120

YADD2: \$0020

YSPEED2: \$0006

YSTART3: \$0122

YADD3: \$0022

YSPEED3: \$0004

MODE: NORMAL

## PIXELEFFECT - PATTERN 4

TIME: 0015

STARS: NONE

XSTART1: \$0100

XADD1: \$0094

XSPEED1: -\$0006

XSTART2: \$0000

XADD2: \$0004

XSPEED2: \$0005

XSTART3: \$0000

XADD3: \$0004

XSPEED3: \$0004

YSTART1: \$0090

YADD1: \$0094

YSPEED1: \$0006

YSTART2: \$0090

YADD2: \$0004

YSPEED2: \$0005

YSTART3: \$0090

YADD3: \$0004

YSPEED3: \$0004

PIXELCOLOR: \$00FF

MODE: TEXTSCREEN16

TEXTSCREEN: 0000

## LINEEFFECT - PATTERN 5

TIME: 0006

STARS: SPRI RIGHT

XSTART1: \$0020

XADD1: \$0030

XSPEED1: -\$0006

XSTART2: \$0020

XADD2: \$0020

XSPEED2: \$0005

XSTART3: \$0080

XADD3: \$0010

XSPEED3: \$0010

```

YSTART1: $0080      YADD1: $0030      YSPEED1: $0009
YSTART2: $0080      YADD2: $0020      YSPEED2: $0008
YSTART3: $0080      YADD3: $0010      YSPEED3: -$0010

```

```

PIXELCOLOR: $00FF
MODE:      TEXTSCREEN16
TEXTSCREEN: 0000

```

```

VECTORGFX - PATTERN 6
OBJECTNR: 0000
TIME:     0020
STARS:    SPRL RIGHT

```

```

ADD X: 0050      ANGLE X: 0000      ROTATE X: 0001
ADD Y: 0000      ANGLE Y: 0000      ROTATE Y: 0001
ADD Z: 0000      ANGLE Z: 0000      ROTATE Z: 0001

```

```
MODE: VECTORSTARS
```

```

ANGLE X: 0000      ANGLE Y: 0000      ANGLE Z: 0000
ROTATE X: 0001      ROTATE Y: 0002      ROTATE Z: 0000

```

```

STARSPD: 0006
STARBOB: 0000
PRIORITY: IN FRONT

```

```

VECTORGFX - PATTERN 7
OBJECTNR: 0001
TIME:     0015
STARS:    NONE

```

```

ADD X: -0050      ANGLE X: 0090      ROTATE X: -0001
ADD Y: 0010      ANGLE Y: 0045      ROTATE Y: 0000
ADD Z: 0020      ANGLE Z: 0000      ROTATE Z: 0002

```

```
MODE: VECTORSTARS
```

```

ANGLE X: 0000      ANGLE Y: 0000      ANGLE Z: 0000
ROTATE X: 0001      ROTATE Y: 0002      ROTATE Z: 0000

```

```

STARSPD: 0006
STARBOB: 0002
PRIORITY: IN FRONT

```

```

SIMUSSCROLLER - PATTERN 8
STARS: SPRL RIGHT

```

```
START1: 0001      ADD1: -0001      SPEED1: 0002
```





```
START2: 0002      ADD2: 0002      SPEED2: -0001

BACKGRLIST: 0007
MODE: COPPERLIST2:
COLORLIST: 0000      COLOR1: $0886      COLOR2: $0FFF

SCREENMODE - PATTERN 9
SCROLLER1: OFF
SCROLLER2: OFF
MUSIC:  ON

BIGLOGO - PATTERN 10
LOGNUMBER: 0000
TIME: 0006
STARS:  SPRI RIGHT

BEGINEFFECT: BUMPING3
ENDEFFECT: FLOOD
MODE:  SINUS BARS
BACKGRLIST: 0004
BARSLIST: 0003
BARS ADD: 0020
BARS SPEED: 0005
BARS NUMBER: 0008
BARMODE:  ALL BARS

SCREENMODE - PATTERN 11
SCROLLER1: ON
SCROLLER2: ON
MUSIC:  ON

ZOOMTEXT - PATTERN 12
COLORLIST: 0000
ZOOMTEXT: 'TCC-DESIGN'

COLOR1: $0FFF      COLOR2: $0F00      ZOOMMODE 0003

ZOOMTEXT - PATTERN 13
COLORLIST: 0000
ZOOMTEXT: 'RED-SECTOR'

COLOR1: $0FFF      COLOR2: $0F00      ZOOMMODE 0003

VECTORBALLS - PATTERN 14
OBJECTNR: 0002
TIME: 0010
STARS:  SPRI RIGHT
```

```
ADD X: 0110      ANGLE X: 0010      ROTATE X: 0001
ADD Y: 0050      ANGLE Y 0010      ROTATE Y 0001
ADD Z: 0060      ANGLE Z  0010      ROTATE Z  0001
```

```
MODE: FULLSCREEN2
```

```
PLASMA - PATTERN 15
TIME: 10
COLORLIST: 0000
```

```
X-ADD1: 0011  X-SPEED1: 0012  Y-ADD1: 0013  Y-SPEED1: 0002
X-ADD2: 0010  X-SPEED2: 0014  Y-ADD2: 0015  Y-SPEED2: 0003
```

Klicken Sie in der Menüleiste auf das Symbol "Pfeil rechts". Wählen Sie nun bei "EXIT DEMO" "REPEAT DEMO". Anschließend klicken Sie in der Menüzelle auf "QUIT", um wieder ins Hauptmenü zu gelangen.

Bevor Sie die Demo starten, sollten Sie sie erst einmal auf Diskette/Festplatte speichern. Klicken Sie dazu "DEMO-MENU" in der Menüleiste an. Wählen Sie nun "SAVE SHORT" und geben den Dateinamen mit Pfad an:

```
DF0:DEMO1
```

Sollten Sie gerade keine formatierte Diskette haben, wählen Sie bitte erst einmal den Punkt "FORMAT" aus.

#### FORMAT

Eine Diskette wird neu formatiert, alle Daten gehen dabei verloren. Es wird nach dem Pfad und dem Namen gefragt. (z.B. "DF0:DEMO-DISK")

Der Demomaker ermöglicht Ihnen folgende Speicherarten:

#### SAVE FINAL

Speichern eines vom CLI aus lauffähigen Demos. Es wird nach Pfad und Name der Demo gefragt, z.B. "DF0:TESTDEMO". Wenn eine fertig gespeicherte Demo vom CLI aus gestartet werden soll, sollte man den CLI-Befehl



RUNDEMO benutzen, der sich auf der Diskette befindet. Der Aufruf wird dann z.B. so gemacht: RUNDEMO DF0:TESTDEMO. Wenn nicht genug Speicher vorhanden ist oder die Demo unter dem angegebenen Namen nicht existiert, bricht der RUNDEMO-Befehl automatisch ab.

**SAVE SHORT**

Wie "SAVE FINAL", jedoch mit dem Unterschied, daß die abgespeicherte Demo wieder in den Demomaker eingeladen und so jederzeit wieder verändert werden kann. Die gespeicherte Demo ist aber dennoch vom CLI aus lauffähig.

**SAVE BOOTBLOCK**

Eine lauffähige Demo wird als Bootintro auf eine Diskette geschrieben, die Diskette sollte vorher neu formatiert werden, da sonst ein Datenverlust möglich ist! Es wird nach dem Pfad gefragt. (DF0-DF3). Wenn eine Diskette mit "SAVE BOOTBLOCK" behandelt wurde, kann sie danach als ganz normale Diskette verwendet werden, wobei natürlich weniger Platz ist, da die Demo einige der vorderen Spuren belegt.





## 10. Eigene Vektorobjekte

Die Objekte können mittels eines Assemblers erstellt werden. Hierzu befinden sich einige Beispiel-Sourcecodes auf der Diskette.

### Der Aufbau eines Objekt-Sourcecodes:

Beachten Sie bitte, daß die Sourcecodes auf der Diskette im Seka-Format vorliegen und auch alle folgenden Beispiele sich darauf beziehen. Durch geringfügige Änderungen können die Sourcecodes/Beispiele auch an andere Assembler angepaßt werden.)

Um einen Objekt-Sourcecode als Objekt zu speichern, muß der Sourcecode assembliert werden und als Datenfile auf der Diskette abgespeichert werden.

Verwenden Sie dazu die folgenden Optionen:

Im Seka:    a / w / Objektname / start / end

### Der Object-Header:

#### Object:

dc.l testpoints-object	Offset zu Punkttabelle
dc.l testflaechen-object	Offset zu Flaechentabelle
dc.w 600	Zoom
dc.w 0,38,\$c,\$44f,\$aaa,\$aaa,\$aaa,\$f00	Objectfarben
dc.w 0	objektart(0-normal/1-complex/2-vectorlines)
dc.b 0	blanklimit(dient zur Synchronisation)
dc.b 0	outline?(0-off/1-on) nur bei complex

dc.l visit-Object	offset zu extrasinit(0-kein Init)
dc.l vds-Object	offset zu extrasjump(0-kein Extra)
	Zoom gibt die Größe des Objekts an

Die folgenden Farben geben die acht Farben des Objektes an. Die Farbe 0 sollte dabei immer 0 sein. Objektart gibt die Art des Objektes an.

Folgende Objekte sind möglich:

- 0 Gefülltes konvexes Objekt (keine sich überschneidenden Flächen)
- 1 Gefülltes konkaves Objekt (Flächen werden sortiert)
- 2 Vektorlinien-Objekt (Nur Farbe eins und Farbe zwei werden benutzt)

Blanklimit kann zur Synchronisation des Objektes benutzt werden. Blanklimit > 0 gibt die Zahl an, wie viele Vertikal-Blanks der Aufbau des Objektes mindestens benötigt. Z.B. würde 2 bedeuten, daß das Objekt in 25 Hertz angezeigt wird. Die Formel hierfür lautet:

$$50/\text{Blanklimit}=\text{Hertz}$$

Falls die Objektart 1 ist, kann mit Outline gewählt werden, ob die Flächen des Objektes extra umrandet werden sollen. Dies gibt einen Outline-Effekt.

Extrasinit und Extrasjump geben den Offset zu einer Init und einer Extra-Routine an. Wenn 0 angegeben wird, dann werden keine Routinen angesprungen.

Extrasinit ist dafür vorgesehen, daß eventuelle Pointer oder ähnliches für die Extra-Routine initialisiert werden. Dabei ist wichtig, zu beachten, daß die Routinen PC relativ programmiert werden (Also keine festen Adressen verwenden!).



Extrasinit wird nur angesprungen, falls das Objekt angezeigt wird.

Extrasjump wird nach jedem Aufbau des Objektes angesprungen. Das Demo liest dann die Farben und die Punkte/Flächen des Objektes nach jedem Aufbau neu aus, so daß mit dieser Routine die Animation des Objektes, das Verändern der Farben oder ähnliches möglich wird.

Um Animationen zu erleichtern, bietet die Demo eine Art Library, in der nützliche Routinen vorhanden sind. Wenn nun Extrasinit oder Extrasjump angesprungen wird, wird in A0 der Zeiger auf die Librarytabelle übergeben. Dieser Zeiger zeigt auf folgende Tabelle:

#### OFFSET

- 0 Rotate (d0, d1 - Koordinaten/d6 - Winkel (0-720 gerade Werte))  
Rotieren der Koordinaten um den Nullpunkt.
- 4 Transformadds (a2- 1.Koordinatenliste/a3- 2.Koordinatenliste  
d1- Anzahl der Punkte /d5- Anzahl der Schritte) Berechnen der  
Transformadds (Sollte im Init angesprungen werden, muß aber  
angesprungen werden, bevor transformiert wird).
- 8 Transformit (a6- Zieladresse für Koordinaten/d0- Punktzahl)  
Transformierung der mit Transformadds festgelegten Koordinaten.  
Die Routine sollte so oft angesprungen werden, wie Schritte festgelegt wurden.
- 12 Transformitback (wie Transformit). Wenn ein Objekt mit  
Transformit fertig transformiert worden ist, kann es mit  
Transformitback zurück transformiert werden. Verwendung  
wie Transformit.
- 16 Zeiger auf Equalizertabelle. Man kann diesen Zeiger auslesen  
und erhält die folgende Tabelle, falls eine Musikroutine  
eingeladen ist. Ist keine Musikroutine vorhanden, steht hier eine 0  
(Langwort). Durch diese Tabelle ist es möglich, eigene Equalizer  
zu programmieren.

*Equaliertab:*

dc.w 0	Sample für Stimme 1
dc.w 0	Note für Stimme 1
dc.w 0	Lautstaerke für Stimme 1
dc.w 0,0,0	Stimme 2
dc.w 0,0,0	Stimme 3
dc.w 0,0,0	Stimme 4

Testpoints enthält die Koordinaten der einzelnen Punkte:

*Testpoints:*

dc.w 2	Anzahl der Punkte
dc.w 50,-50,0	XYZ-Koordinaten für 1.Punkt
dc.w 50,50,0	XYZ-Koordinaten für 2.Punkt
dc.w -50,50,0	XYZ-Koordinaten für 3.Punkt
dc.w -50,-50,0	XYZ-Koordinaten für 4.Punkt

Je nach Objektart gibt Testflächen die Flächen (0-1) oder die Linien (2) des Objekts an (Objektart 0 oder 1).

*Testflächen:*

dc.w 0	Anzahl der Flächen-1
dc.b 0	Flächenart für Fläche 1
dc.b 4	Anzahl der Punkte für Fläche 1
dc.b 0	Farbe2 für Fläche 1
dc.b 1	Farbe1 für Fläche 1
dc.w 0,1,2,3,0	Punkte für Fläche 1 (letzter Punkt=1 Punkt. - Punkt)





*nächste Fläche...*

Flächenart = 0 Normale Fläche mit Farbe 1

Flächenart = \$10 Immer sichtbare Fläche mit Farbe 1

Flächenart = \$80 Beidseitige Fläche mit Farbe 1 für vorne und Farbe 2 für hinten.

Farbe = 1 bis 7 für Farbe

Bei Objektart = Complex kann die Fläche durch Farbe=1 bis 7 + \$10 als durchsichtig deklariert werden.

(Objektart 2)

*Testflächen:*

dc.w 4 Anzahl der Linien

dc.w 0,1 Punkte für Linie 1

dc.w 1,2 Punkte für Linie 2

dc.w 2,3 ...

dc.w 3,0 ...

## 10.1 Eigene Vektorball-Objekte

Auch für Vektorball-Objekte besteht aber die Möglichkeit, Objekte mittels eines Assemblers zu erzeugen

*Objektheader:*

**ballsob1:**

dc.l ballpoints-ballsob1 Offset zu Punkttabelle

dc.l ballart-ballsob1 Offset zu Ballarten

dc.w 500 Zoom

dc.w 0,\$500,\$710,\$930,\$a50,\$d90,\$fd0,\$000,\$5,\$6,  
\$118,\$33a,\$55b,\$88c,\$ccd

dc.w \$fff 16 Objektfarben

dc.l vinit-ballsobl offset zu extrasinit(0-kein Init)

dc.l vdo-ballsobl offset zu extrasjump(0-kein Extra)

Extrasinit und Extrasjump wie bei Vektorobjekten.

### Ballpoints:

dc.w 3 Ballanzahl

dc.w 0,0,0 x/y/z Koordinaten für ball1

dc.w 0,0,0 x/y/z Koordinaten für ball2

dc.w 0,0,0 x/y/z Koordinaten für ball3

### Ballarten:

dc.w 0 ballart für ball1

dc.w 10 ballart für ball2

dc.w 40 ballart für ball3

Die Ballarten bestimmen, welche Kugel aus dem Vectorballs-Screen herauskopiert werden sollen.

Ballart = 0 bis 19 1. Zeile des Screens 16x16 Kugeln

Ballart = 20 bis 39 2. Zeile des Screens 16x16 Kugeln

Ballart = 40 bis 49 3. Zeile des Screens 32x32 Kugeln

## 10.2 Ändern bestehender Vektorobjekte

In diesem Kapitel befassen wir uns mit dem Ändern von bestehenden Vektorobjekten durch Ändern des Source-Codes.

Im Sources-Verzeichnis auf der Diskette befinden sich einige Quell-Codes, die Sie Ihren eigenen Wünschen entsprechend abändern



können. Diese Source-Codes wurden mit dem Seka-Assembler erstellt, können aber mit ein paar Tricks auch von anderen Assemblern (z.B. Devpack) benutzt werden. Um einen Source mit dem Devpack zu verwenden, müssen Sie folgende Änderungen vornehmen:

1. Fügen Sie vor jeder DC-Anweisung ein Leerzeichen ein.
2. Ersetzen Sie, falls im Source vorhanden, den Befehl BLK durch den DCB-Befehl.

Nun können Sie den Source assemblieren. Dieser Source kann aber noch nicht ohne weiteres benutzt werden, da der Devpack einen ausführbaren Code erzeugt. Veranlassen Sie den Devpack deshalb dazu, mit Hilfe der "OPTL+"-Anweisung einen linkbaren Code zu erzeugen. Von diesem Code müssen nun die ersten zehn sowie die letzten drei Langwörter entfernt werden. Danach haben Sie den richtigen Code.

Kommen wir aber nun zu dem Aufbau eines Vektorobjektes. Dies geschieht anhand der Vectorballs1.S-Datei.

Sehen wir uns diese Datei einmal genauer an:

```
start:
```

Dieses Label definiert den Programm-Anfang für den Seka.

```
dc.l    *BALL*
```

Das erste Langwort der Datei enthält die ASCII-Kennung "BALL", damit der Demomaker die Datei als Vectorballs-Objektdatei identifizieren kann.

```
ballsobl:
```

Das "ballsobl"-Label kennzeichnet den Start der Ball-Objektdatei. Dieses Label wird benötigt, um Offset-Berechnungen durchzuführen.

```
dc.l ballpoints-ballsobl
```

Das zweite Langwort der Datei enthält den Offset auf die Punktstruktur, in der die Koordinaten jedes Balls sowie die Anzahl der Bälle des Vektorobjektes festgelegt sind.

```
dc.l ballart-ballsobl
```

Das dritte Langwort enthält den Offset auf die Ballartstruktur. In dieser Struktur wird das Aussehen jedes Balls festgelegt.

```
dc.w 650
```

Das darauffolgende Wort enthält nun den Wert für den Zoom-Mode. Je größer dieser Wert ist, desto kleiner wird das Vektorobjekt später dargestellt.

```
dc.w 0,$500,$710,$930,$a50,$d90,$fd0,$000,$5,$6,$118,$33a,$55b,$88c,$cod
```

```
dc.w $fff
```

Die 16 darauf folgenden Worte enthalten die Farb-Palette für das Vektorobjekt. Diese Farbpalette ist identisch mit der, die Sie im Edit-Mode des Demomakers sehen.

```
dc.l 0
```

Das darauffolgende Langwort kann einen Zeiger auf eine Initialisierungs-Routine enthalten. In diesem Fall wurde aber darauf verzichtet.



```
dc.l 0
```

Danach folgt ein Zeiger auf eine eventuell vorhandene Zusatzroutine, die ausgeführt werden soll. Im Source-Code wurde darauf verzichtet.

```
ballart:
```

Das "ballart"-Label kennzeichnet den Start der Ballartdaten. Dieses Label wird benötigt, um Offsetberechnungen durchzuführen.

```
blk.w 24,6
```

Die nun folgenden Worte reservieren für 24 Bälle das Aussehen des siebten Balles. Das Aussehen ist identisch mit dem Aussehen des siebten Balles der ersten Zeile der Vectorballs-Datei. Beachten Sie, daß der Demomaker das Aussehen intern zwischen 0 und 48 festlegen kann. Das heißt, der erste Ball der Vectorballs-Datei hat nicht die Kennung eins, sondern die Kennung null.

```
blk.w 24,26
```

Die nächsten 24 Bälle bekommen das Aussehen des siebenundzwanzigsten Balles der Vectorballs-Datei. Gerechnet wird hierbei von links nach rechts und von oben nach unten.

```
ballpoints:
```

Das "ballpoints"-Label kennzeichnet den Start der Ballpunktstruktur. Dieses Label wird benötigt, um Offsetberechnungen durchzuführen.

```
dc.w 46
```

Dieses Wort enthält die Anzahl der in dem Vektorobjekt benutzten Bälle.

Nun folgen noch die Koordinaten jedes einzelnen Balls. Zur besseren Übersicht wurden die Koordinaten für jeden Ball in eine separate Zeile geschrieben. Das erste Wort enthält die X-, das zweite die Y- und das dritte die Z-Koordinate des Balls.

```
DC.W 000, 000, 000
DC.W 015, -015, 000
DC.W 045, -045, 000
DC.W 030, -030, 000
DC.W 060, -060, 000
DC.W -015, 015, 000
DC.W -030, 030, 000
DC.W -045, 045, 000
DC.W -060, 060, 000
DC.W -075, 045, 000
DC.W -090, 030, 000
DC.W -105, 015, 000
DC.W 075, -060, 015
DC.W 060, -045, 015
DC.W 045, -030, 015
DC.W 030, -015, 015
DC.W 015, 000, 015
DC.W 000, 015, 015
DC.W -015, 030, 015
DC.W -030, 045, 015
DC.W -045, 060, 015
DC.W -060, 045, 015
DC.W -075, 030, 015
DC.W -090, 015, 015
DC.W -075, 015, 030
DC.W -060, 030, 030
DC.W -045, 045, 030
DC.W -030, 060, 030
DC.W -015, 045, 030
DC.W 000, 030, 030
DC.W 015, 015, 030
DC.W 030, 000, 030
DC.W 045, -015, 030
DC.W 060, -030, 030
DC.W 075, -045, 030
DC.W 090, -060, 030
DC.W -060, 015, 045
DC.W -045, 030, 045
DC.W -030, 045, 045
DC.W -015, 060, 045
DC.W 000, 045, 045
DC.W 015, 030, 045
```



```
DC.W 030, 015, 045  
DC.W 045, 000, 045  
DC.W 060, -015, 045  
DC.W 075, -030, 045  
DC.W 090, -045, 045  
DC.W 105, -060, 045
```

```
end:
```

Das End-Label markiert das Ende der Datei.







## 11. Stichwortverzeichnis

3 COLORS .....	42
ADD 1 .....	41
ADD 2 .....	41
ADD X .....	35
ADD Y .....	35
ADD Z .....	35
Ändern der Farbenlisten .....	55
ANGLE X .....	35f, 39
ANGLE Y .....	35f, 39
ANGLE Z .....	36, 39
BACKGRCOLS .....	34
BACKGRCOLS 0-7 .....	22
BACKGRLIST .....	34, 36, 38, 42
BALLSOBJECT 0-9 .....	22
Baseline .....	26
BEGINEFFECT .....	32
BIGLOGO .....	18, 32
Bildschirmbereich .....	30
BOBANIM .....	52
BOBEFFECT .....	52
BOBEFFECTGFX .....	20
BOBNUMBER .....	52
BUMPING .....	33
Choice .....	24
CLEAR 1-3 .....	33
CLEARNR .....	49
COLOR 1 .....	36, 39
COLOR 2 .....	36, 39
COLOR 3 .....	36, 39

Copperlist .....	25
COPPERLIST1 .....	42
COPPERLIST2 .....	43
Delete Pattern .....	29
DISTANCE .....	34, 46
EDIT .....	17
Editieren .....	23
EFFECT1 .....	34
Eigene Demos .....	63
Eigene Grafiken .....	59
Eigene Vektorobjekte .....	73
Eingabefeld .....	31
ENDEFFECT .....	33
EXIT .....	58
Farblisten .....	22
Farbverlauf .....	55
FLASHEN 1 .....	42
FLASHEN 2 .....	43
FLOOD .....	33
FONT 16x16 .....	19
FONT 16x16 IPL .....	19
FONT 32x32 .....	18
FONT 8x8 .....	19
FORMAT .....	58
FULLSCREEN1 .....	38
FULLSCREEN2 .....	38
Funktionstasten .....	17
Grafiken .....	18
Hexadezimale Farbeingabe .....	54
Hold .....	24, 27
Insert Pattern .....	30



Jump .....	25
Kommentare .....	30
LINEEFFECT .....	49
LITTLELOGO .....	18, 35
LOAD SHORT .....	57
Logo .....	25
LOGONUMBER .....	32
MIDDLE .....	44
MIRROR .....	36
MIRROR1 .....	38
MIRROR2 .....	38
Mode .....	25
MUSIC .....	32
NONE .....	37, 39
NORMAL .....	34, 36, 38
OBJECT 0-9 .....	22
OBJECTNR .....	35, 37
Objekt-Sourcecodes .....	73
Off .....	26
ONE COLOR .....	42
Pfeil links .....	30
Pfeil rechts .....	30
Pinzel .....	60
PIXELEFFECT .....	53
PIXELSTARS .....	36, 39, 41
PLASMA .....	48
Präferenzliste .....	16
PRIORITY .....	36f, 39, 40
Quit .....	30

ROTATE .....	36
ROTATE X .....	36, 39
ROTATE Y .....	36, 39
ROTATE Z .....	36, 39f
SAVE BOOTBLOCK .....	58
SAVE FINAL .....	57
SAVE SHORT .....	57
SCRBACKGRCOLS 0-3 .....	22
SCREEN MODE .....	31
SCREEN ONLY .....	39f
SCROLLBARCOLS .....	23
SCROLLER1 .....	31
SCROLLER2 .....	31
SCROLLTEXT OBEN .....	21
SCROLLTEXT UNTEN .....	20
Scrolltexte .....	20
SINUSSCROLLER .....	26, 41
SINUSSCROLLTEXT .....	21
Sonderzeichen .....	24, 26
Speed .....	24, 26
SPEED 1 .....	41
SPEED 2 .....	41
SPR1 LEFT .....	37
SPR1 RIGHT .....	37
SPRITEOBJECT .....	19
STARBOBS .....	20, 36, 39
STARS .....	32, 35, 40
STARSPPEED .....	36, 39
START 1 .....	41
START 2 .....	41
Tastaturbelegungen .....	16
Teilstück .....	30
Test Demo .....	29
Test Pattern .....	29
TEXT16SPECIAL .....	39



TEXT8SPECIAL .....	40
TEXTLINE16 .....	47
TEXTLINE32 .....	47
TEXTLINE8 .....	48
TEXTSCR16 .....	47
TEXTSCR32 .....	44
TEXTSCR8 .....	47
TEXTSCREEN .....	37, 39f
TEXTSCREEN16 0-9 .....	21
TEXTSCREEN32 0-9 .....	21
TEXTSCREEN8 0-9 .....	21
TIME .....	32, 35, 40
Trennlinie .....	14
TXTSCREEN16 .....	37
TXTSCREEN8 .....	37
Überschreib- und Einfügemodus .....	24
VECTORBALLS .....	37
VECTORBALLSCOLS .....	23
VECTORBALLSGFX .....	20
VECTORCHARSET .....	19
VECTORGFX, .....	35
VECTORSTARS .....	36, 39f
Vektorobjekte .....	22
Wobble .....	25
ZOOM .....	33
ZOOMMODE .....	43
ZOOMTEXT .....	43
ZOOMTEXTCOLS 0-3 .....	23

# In einem Band: alle Details zu Ihrem Amiga.

Amiga Intern – von der ersten bis zur letzten Seite bietet Ihnen dieses einzigartige Nachschlagewerk alle harten Facts zu Ihrem Rechner. Angefangen



von einer detaillierten Beschreibung des 68000-Prozessors, der CIA, der Custom-Chips und der Schnittstellen über die Hardware-Programmierung bis hin zu einer leichtverständlichen Dokumentation aller Library-Funktionen – zu allen bisher ausgelieferten Kickstart-Versionen. Aus dem Inhalt: die Strukturen von EXEC, I/O-Handhabung, Verwaltung der Resources, Erstellen eigener Devices, EXEC-Base, reseedete

Programme, DOS-Funktionen, interne DOS-Bibliothek, Aufbau einer Diskette, Autoboot mit der ROMboot.library, Programmierung eigener Handler, Ein- und Ausgabe über die verschiedenen Amiga-Devices, Standard-Austausch-Formate und IFF-Komprimierungsverfahren, alle Amiga-Libraries mit den dazugehörigen Strukturen, Basis- und Grundstrukturen, Preferences als Datenstruktur, Datenübermittlung von Workbench und CLI, Konventionen im Programmierstil...

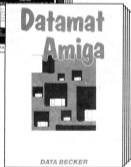
**Bleek/Dittrich/Gelfand/Jenrich/Schemmel/Schulz**

**Amiga Intern**

**Hardcover, 1095 Seiten, DM 98,-**

**ISBN 3-89011-398-2**

# TEXTOMAT und DATAMAT : Das Power-Pack Amiga.



Für alle Amiga-Fans halten wir etwas ganz Besonderes parat: das Power-Pack Amiga! Es umfaßt unsere bewährten Programme TEXTOMAT und DATAMAT für den Amiga. TEXTOMAT ist die ideale Textverarbeitung für Anfänger und Textprofis. Das Programm bietet Ihnen z. B. die Möglichkeit, sämtliche Befehle über Menüleisten oder Kurzbefehle anzu-steuern, Bildschirmausschnitte aus anderen Programmen einzulesen und zu ver-

arbeiten, eine schnelle Direktformatierung, die Darstellung der Schriftattribute und Grafikeinbindung direkt auf dem Bildschirm, eine automatische Silbentrennung sowie fertige Druckanpassungen für alle gängigen Drucker. DATAMAT ist das bewährte Dateiprogramm, das Ihnen die Pflege und Verwaltung Ihrer Daten so einfach wie möglich macht. Zu den Leistungsmerkmalen von

DATAMAT zählen eine Dateigröße von max. 2 Milliarden Zeichen, maximal 8 gleichzeitig zu öffnende Dateien, eine maximale Feldgröße von 32.000 Zeichen, maximal 80 Indexfelder mit wählbarer Genauigkeit, frei gestaltbare Bildschirmmasken, ein integrierter Druckmasken- und Listeneditor u.v.a.m.

**TEXTOMAT Amiga**  
unverb. Preisempf. DM 99,-  
ISBN 3-89011-580-2

**DATAMAT Amiga**  
unverb. Preisempf. DM 99,-  
ISBN 3-89011-581-0

# Die ganze, far- benfrohe Palette von DPaint III.

DPaint III gehört wohl zu den außergewöhnlichsten Grafikprogrammen, die derzeit für den Amiga verfügbar sind. Bereits mit den einfachen, „normalen“ Funktionen erzielt man erstaunliche Ergebnisse. Mit dem großen DPaint-III-Buch jedoch wird's erst so richtig professionell. Neben einer ausführlichen, leichtverständlichen Beschreibung der DPaint-Grundfunktionen zeigt dieses Buch vor allem, was DPaint tatsächlich leisten kann: perspektivische Zeichnungen, verbogene Brushes, Animation in 3D, Erstellen von Videos... Dazu alles Wichtige über den Datenaustausch mit anderen Programmen. Natürlich verraten Ihnen die Autoren auch ihre zahlreichen Tips und Tricks, mit denen sie noch mehr aus diesem Programm herausholen. Ein hilfreicher Anhang, der alle Funktionen übersichtlich darstellt, rundet das Ganze ab und macht diesen Band zu einem Standardwerk, in dem auch der Profi immer wieder nachschlagen wird. Das große DPaint-III-Buch – für alle, die nach Höherem streben.



Das große DPaint-III-Buch

Langlotz/Vignjevic  
Das große DPaint-III-Buch  
393 Seiten, DM 39,-  
ISBN 3-89011-369-9



# Die 90 besten, preiswertesten Programme!

Tolle Amiga-Software „für einen Apfel und ein Ei“? Im Public-Domain-, Shareware- und Freeware-Markt gibt es das. Wer sagt einem aber, wie man

die Rosinen aus dem ständig wachsenden Angebot herauspicks? Wo erfährt man, was sich hinter Namen wie „StealMemBoof“ oder „A68K“ verbirgt? Alle Fragen haben jetzt ein Ende: Die besten Public-Domain- und Shareware-Programme werden ausführlich im gleichnamigen Buch vorgestellt. Sauber sortiert nach Programmen „zur täglichen Arbeit“, Programmierhilfen, Utilities und neuen CU-Befehlen,

DOS-Shells, Spielen, Animationen, Programmiersprachen und speziellen Anwendungen. Genau 90 Super-Programme von „Atari Meets Amiga“ bis „Xplor“ sind es, die hier detailliert beschrieben werden. Quälen Sie sich also nicht länger mit englischen Hilfstexten oder einer ungewohnten Bedienung – „Die besten Public-Domain- & Shareware-Programme“ hilft Ihnen nicht nur bei der Auswahl, sondern auch beim Einsatz der preisgünstigen Software.



**Rührkh/Sanio/Tornsdorf**  
**Die besten Public-Domain- & Shareware-Programme**  
357 Seiten, DM 39,-  
ISBN 3-89011-368-0

# So kommt der „Kleine“ ganz groß raus.

Zum Spielen alleine ist er zu schade: Lernen Sie Ihren Amiga 500 von seiner professionellen Seite kennen. Das große Amiga-500-Buch vermittelt



Ihnen nicht nur den Umgang mit CU und Workbench, sondern enthält auch ausführliche Kapitel zur Systemprogrammierung, zum professionellen Einsatz von Software und zu "Umbauten", die Sie selbst ausführen können. Ob Sie Einsteiger oder Profi sind: In diesem Buch finden Sie alle wichtigen Informationen rund um den Amiga 500 - etwa zum Virenschutz, zum Soundsampling, zu

Hardware-Erweiterungen (z.B. neuer Prozessor, Harddisk, PC-Karte, Profifgehäuse), zu Kickstart 1.3 und zu vielen Themen mehr. Außerdem gibt es natürlich jede Menge Tips und Tricks.

**Bleek/Langlotz**  
**Das große Amiga-500-Buch**  
Hardcover, 527 Seiten, DM 49,-  
ISBN: 3-89011-279-X

# Kompetentes Detailwissen zum Amiga 2000.

Von den glorreichen Drei (Rügheimer/Spanik/Amiga) wurden die Amiga-Anwender schon immer verwöhnt: Ihre Fachbücher strotzen vor fundiertem

Know-how und sind dennoch dank einer lockeren Schreibeweise überaus unterhaltsam. Warum sollte es also beim großen Amiga-2000-Buch anders sein? Mit viel Liebe zum Detail und der gewohnten leichtverständlichen Sprache beschreiben Sie alles, was den Amiga 2000 so interessant macht: die möglichen Speichererweiterungen, Einbau und Einrichtung einer PC-/Amiga-Harddisk, Arbei-



ten mit einer PC-/AT-Karte, Kickstart im RAM, jede Menge zur Janus.Library 2.0, Software-Installationstips, der Umgang mit dem AmigaDOS und und und. Selbstverständlich auch mit einer detaillierten Einführung zum Amiga 2000 und einigen wichtigen Software-Tips. Das große Amiga-2000-Buch – ein großartiges Buch zu einem großartigen Rechner.

**Rügheimer/Spanik**  
**Das große Amiga-2000-Buch**  
**Hardcover, 756 Seiten, DM 59,-**  
**ISBN 3-89011-199-8**

# Alles korrekt? Fragen Sie den Rechtschreibprofi.

DATA BECKERs Rechtschreibprofi ist das schnelle Prüfprogramm für alle, die Wert auf einwandfreie Texte legen. Ganz gleich, welche Textverarbeitung Sie besitzen: Lassen Sie Ihre Berichte, Artikel,

Protokolle, wissenschaftliche Texte und Diplom- oder Studienarbeiten bequem vom Computer „gegenlesen“. Die komfortable Oberfläche erlaubt auch Computer-Neulingen die rasche Nutzung des Programms. Für die Richtigkeit des Prüfungsvorgangs garantiert ein bekannter Name: Langenscheidt-Mitarbeiter haben die rund 106.000 Wörter und Flexionen des Hauptwörterbuchs zusam-

mengestellt. Da DATA BECKERs Rechtschreibprofi sich am Wortstamm orientiert, sind damit auch alle möglichen Kombinationen erfasst: Insgesamt werden also 2,5 Millionen Wörter überprüft. Natürlich haben Sie außerdem die Möglichkeit, weitere Fachausdrücke und fremdsprachliche Begriffe im Benutzerlexikon abzulegen.



**DATA BECKERs Rechtschreibprofi Amiga**  
ISBN 3-89011-585-3  
DM 99,- (unverb. Preisempfehlung)

**DATA BECKER**

DATA BECKER<sub>s</sub>  
DEMDMAKER AMIGA

Software

Bestell-Nr.  
440814/1

Serien-Nr.  
13182

---

Jede von uns nicht autorisierte Weitergabe des Programms  
(Verleih, Tausch, Verkauf) ist untersagt.  
© Copyright DATA BECKER

