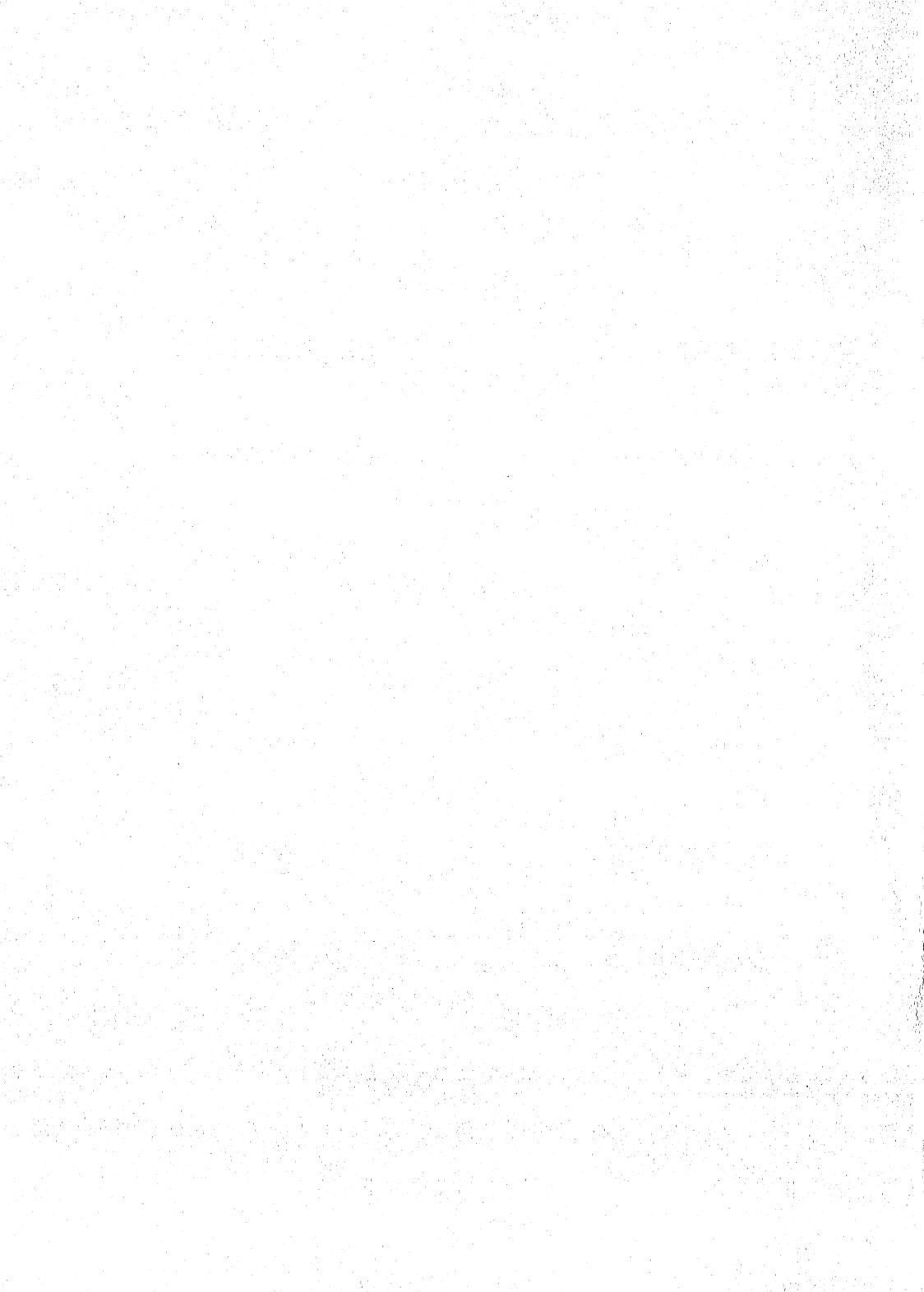


Die Fox-Bibel
zum
Printfox-Basar



Scanntronik
Mugrauer GmbH
Parkstr. 38 8011 Zorneding



Die Fox-Bibel

**Anleitungen, Tips und Tricks
zu
Printfox
Characterfox
Superscanner II
und natürlich zum
Printfox-Basar**

Von Hans Haberl

(C) 1986 by

 **Scanntronik**

**Hugrauer GmbH
Parkstr.38 8011 Zorneding**

PRINTFOX-BASAR

Inhaltsverzeichnis:

Einführung	3
1. Textlayout	5
1.1. Textposition und Zeilenlänge	7
1.2. Tabulatoren und Einrücken	13
1.3. Der IF-Befehl	16
1.4. Tips und Tricks	21
2. Textgestaltung	24
2.1. Initialen	24
2.2. Special Effects	28
3. Text und Grafik	31
3.1. Plazieren, Rahmen und Initialen	31
3.2. Große Grafiken	39
3.3. Weitere Techniken	44
4. Sonstiges	48
5. Literaturhinweise	51
Anhang: Verzeichnis aller Grafiken	52

Lieber User!

Mit dem Printfox-Basar liegt die zweite Erweiterung zum Printfox vor. Nach der Zeichensatz-Erweiterung "Characterfox" ist der Printfox-Basar eine Erweiterung der Grafikbibliothek. Besonders interessant ist dies für User, die noch keinen Superscanner besitzen. Vielleicht ist die Bildersammlung dann auch der Anreiz, sich dieses nützliche Gerät zuzulegen. Denn alle Bilder dieser Sammlung wurden natürlich mit dem Super-scanner digitalisiert.

Im Gegensatz zur Printfox-Programmdiskette sind die Grafiken diesmal als Gesamtbilder gespeichert. Viele der Grafiken sind größer als ein Bildschirm. Der Einbau solch großer Grafiken in das Layout ist zwar vielleicht etwas komplizierter, doch das Ergebnis ist dafür beeindruckend. Wenn Du eine Grafik aus Platzgründen doch mal kleiner brauchst, dann verkleinerst Du sie eben mit dem Befehl 'W' im Grafikeditor.

Der zweite Bestandteil des Printfox-Basar neben der Grafiksammlung ist dieses Handbuch, das Dir bei der Gestaltung Deiner Werke helfen soll. Bei den vielfältigen Möglichkeiten des Printfox ist schon ein Bißchen Übung nötig, um alles aus diesem Super-programm herauszuholen.

Die einzelnen Kapitel entsprechen den wesentlichen Themen bei der Erstellung der Text- und Grafik-Layouts, von der Textformatierung und der Textgestaltung bis zur Plazierung von Grafiken. Jedes Kapitel beginnt mit den einfachen Grundlagen des jeweiligen Themas, vertieft den Stoff an konkreten Beispielen und bringt zum Schluß fortgeschrittene Techniken, Tips und Tricks für den Profi.

Wer mit dem Printfox schon besser vertraut ist, den werden einige Kapitel wohl langweilen, da er das dort gesagte schon beherrscht. Doch nicht jeder ist ein Profi! Viele Anwender wissen noch gar nicht, was sie mit dem Printfox alles anstellen können, und für die sind die ausführlichen Darstellungen in diesem Buch gedacht. Freilich soll dies hier nicht bloß eine ausführlichere Bedienungsanleitung sein. Ganz im Gegenteil: Die Bedeutung der einzelnen Befehle soll hier nicht nochmals erläutert werden, dafür ist die Bedienungsanleitung da. Wenn Dir also ein Befehl nicht klar ist, dann sieh dort nach. Hier steht die richtige Anwendung der Möglichkeiten im Vordergrund.

Am Besten betrachtest Du dieses Handbuch als Lehrbuch und liest die Kapitel der Reihe nach durch. Nach der Lektüre wird Dir das Erstellen anspruchsvoller Layouts keine Probleme mehr bereiten.

1. Textlayout

Grundlage aller Layouts ist die passende Plazierung und Formatierung des Textes. Bei der Kombination mit Grafiken ist es üblich, immer zuerst die Grafiken anzuordnen und den Text dann in passender Weise in die freien Flächen "einzufüllen". Denn während die Grafiken in Form und Größe meist feststehen, ist der Text flexibel, kann wie eine Flüssigkeit die freien Flächen ausfüllen.

Doch auch bei einer Seite ohne Grafiken bestimmt das Textlayout die Erscheinung, Übersichtlichkeit und nicht zuletzt die Lesbarkeit. So sind z.B. lange Zeilen, die über die volle Breite einer DIN A-4-Seite gehen, schlechter zu lesen als die kürzeren Zeilen bei zwei- oder dreispaltiger Formatierung, bei denen man nicht so leicht verrutscht.

Printfox bietet bei der Textformatierung eine praktisch unbegrenzte Flexibilität. Natürlich ist ein solch flexibles Programm nicht ganz so einfach zu bedienen wie eines, das dem Anwender die Formatierung bereits fest vorschreibt und keinerlei Variationen zuläßt. Insbesondere die Anwendung des 'IF'-Befehls ist recht anspruchsvoll, dafür ist dieser Befehl aber auch der Schlüssel zu wirklich professionellen Layouts. Doch bevor wir zu diesem mächtigen Befehl kommen, müssen die grundlegenden Befehle verstanden sein.

Doch vorher noch eine Bemerkung zur Vorgehensweise beim Durcharbeiten dieses Handbuches: Am Besten begreift man durch Probieren. Du solltest deshalb alle in diesem Buch besprochenen Methoden und Beispiele sofort selbst mit dem Printfox ausprobieren. Teilweise sind die dafür nötigen Files schon auf der Diskette Nummer 3 enthalten. Für die allgemeinen Formatier-Beispiele verwendest Du den "PROBETEXT.FT" auf der Diskette 3. Dies ist der Demo-Text von der Printfox-Diskette, jedoch wurden daraus alle Formatier-Befehle entfernt. Der Text ist damit eine Art "Modelliermasse", die Du für Deine Experimente in beliebige Formen gießen kannst. Auf den Inhalt des Textes kommt es dabei überhaupt nicht an. Auch in der professionellen Layout-Gestaltung ist es üblich, mit sogenannten Blind-Texten zu arbeiten, die nur eine Aufgabe haben: Sie sollen wie Text aussehen.

1.1. Textposition und Zeilenlänge

Wie Du aus der Bedienungsanleitung zum Printfox schon weißt, besteht die gesamte erfaßbare Seite aus 640 mal 800 Punkten. Der Nullpunkt ist dabei in der linken, oberen Ecke. Gibst Du keinerlei Befehle zur Plazierung, so beginnt der Text bei x=20 und y=20 mit einer Zeilenlänge von 600 Punkten. Es bleibt dann also oben, links und rechts ein Rand von 20 Punkten. Genausogut kannst Du aber die Formatzeile mit den entsprechenden Befehlen einbauen:

F x=20 y=20 l=600

Das große 'F' steht hier der Einfachheit halber für CTRL F. Probier' das nun mit dem "PROBETEXT.FT" im Printfox aus: Formatiere einmal den Text so, wie er auf der Diskette ist und ein zweites Mal, nachdem Du am Anfang des Textes die obige Formatzeile eingefügt hast. Es muß beide Male das gleiche Ergebnis herauskommen. Denke daran, daß Du Dir mittels 'W' im Grafikeditor jederzeit einen Gesamtüberblick über eine halbe Textseite verschaffen kannst. Gerade für die Beurteilung des Layouts ist dies sehr nützlich, den Text muß man dabei sowieso nicht lesen können.

Bei dieser Gelegenheit kannst Du auch gleich nachprüfen, ob die 20-Punkte-Ränder eingehalten werden. Dazu schaltest Du die Koordinatenanzeige ein und setzt den Cursor an die Textkanten links und oben. An der rechten Kante geht das beim Flattersatz natürlich schlecht. Du mußt deshalb mit Blocksatz oder rechtsbündig formatieren, d.h. Du setzt in die Formatzeile am Anfang noch den Befehl 's=1' oder 's=3'. Im "PROBETEXT.FT" wird mit 's=2' zeitweise auf

zentriert umgeschaltet, danach mit 's=0' wieder auf Flattersatz. Ersetze die '0' hier ebenfalls durch '1' oder '3'.

Du wirst beim Nachmessen feststellen, daß der rechte Rand 21 Punkte breit ist, also einen Punkt breiter, als er rein rechnerisch sein sollte (nämlich die Gesamtbreite von 640 minus Zeilenlänge 600 minus linker Rand 20 sollte 20 ergeben). An den Stellen im Text, an denen der Zeichenabstand mittels 'h' auf 2 oder 3 Punkte vergrößert wurde, ist der Rand gar 22 oder 23 Punkte breit. Der gleiche Effekt ergibt sich bei Breitschrift (CTRL B), da hierbei automatisch auch der Zeichenabstand verdoppelt wird. Wie es zu diesem Effekt kommt und was man dagegen tun kann, erfährst Du in Kapitel 1.4.

Nun wollen wir mit den Befehlen 'x', 'y' und 'l' etwas spielen, um damit vertraut zu werden. Versuch' zum Beispiel, links und rechts einen Rand der Breite 100 hinzubekommen. Wenn Du dafür nur den 'x'-Befehl in 'x=100' und sonst nichts änderst, wird das der Printfox mit der Meldung 'Bereichsfehler' quittieren. Denn er versucht nun, neben einem 100 Punkte breiten Rand noch 600 Punkte lange Zeilen in den Grafikspeicher zu bekommen - was natürlich nicht geht, da der Speicher nur 640 Punkte breit ist. Du mußt also auch die Zeilenlänge ändern, und das möglichst so, daß auch auf der rechten Seite ein 100 Punkte breiter Rand bleibt. Dazu subtrahierst Du von den insgesamt zur Verfügung stehenden 640 Punkten die zwei Ränder zu je 100 Punkten und was übrig bleibt, nämlich 440, ist die Zeilenlänge. Also schreibst Du 'l=440' in die Formatzeile.

Und schon wieder gibt's einen Bereichsfehler! Formatiere die 2. Seitenhälfte und gehe, nachdem die Fehlermeldung erschienen ist, in den Grafikeditor, um die Ursache herauszubekommen: Diesmal liegt's nicht am rechten Rand, sondern am unteren: Dadurch, daß wir den Text in eine engere Form gepreßt haben, paßt er nicht mehr auf die Seite. Dies ist bei der Arbeit mit dem Printfox sicherlich die häufigste Ursache für einen Bereichsfehler. Um den Text in solchen Fällen dennoch auf die Seite zu bekommen, gibt's eine ganze Reihe von Maßnahmen, zum Beispiel: Verschmälern der Ränder, Verkleinern des Raumes für Grafiken, Verkleinerung des Zeilenabstandes oder Verwendung einer kleineren Schrift, beispielweise ZS2 statt ZS1. Natürlich gehen viele dieser Maßnahmen zu Lasten der Übersichtlichkeit und Lesbarkeit und sind folglich nur dann zu empfehlen, wenn's nicht zu weit fehlt. Durch Verkleinern des Zeilenabstandes von 'v=2' (Defaultwert) auf 'v=1' gewinnt man bei einer Zeichenhöhe von 9 Punkten immerhin 10% mehr Platz, wobei die Lesbarkeit kaum beeinträchtigt wird.

Nun wollen wir einen ersten Ansatz zur zweispaltigen Formatierung wagen. Dabei soll links, rechts und in der Mitte zwischen den beiden Spalten ein Rand von 20 Punkten bleiben. Das ergibt einen Gesamtbedarf von 60 Punkten für Ränder, subtrahiert von den 640 Punkten Papierbreite bleiben für den Text somit 580 Punkte. Da wir dies auf zwei Spalten aufteilen wollen, ergibt sich die Zeilenlänge zu 290 Punkten. Wir kommen zu der Formatzeile:

F x=20 y=20 l=290

Auch jetzt gibt's wieder einen Bereichsfehler, denn der Text paßt in einer Spalte natürlich nicht auf die Seite. Doch darum wollen wir uns zunächst nicht kümmern. Was uns nun bei der Betrachtung des Textes im Grafikeditor auffällt, ist die Tatsache, daß auch die Überschrift auf die linke Seite gerückt wurde. Das mag in vielen Fällen erwünscht sein, doch wir möchten die Überschrift in der Mitte über beiden Spalten haben. Dazu müssen wir, wie vorhin, die Zeilenlänge von 'l=600' belassen. Erst nach der Überschrift dürfen wir sie auf 290 verkleinern. Wir kommen somit wieder zu der Formatzeile:

F x=20 y=20 l=600

Nach der Überschrift, unmittelbar vor dem Beginn des Textes, fügen wir dann folgende Formatzeile ein:

F l=290

Bleibt immer noch der Bereichsfehler, dem wir nun zu Leibe rücken wollen. Die naheliegendste Lösung dazu: Wir schreiben einfach an der Stelle, an der die zweite Spalte beginnen soll, eine entsprechende Formatzeile in den Text. Dazu müssen wir diese Stelle erst finden: Formatiere die zweite Seitenhälfte und sieh Dir im Grafikeditor den Bildschirm 3 an, das ist die linke, untere Ecke der gesamten Seite. Wir wollen nun beispielsweise die letzte Zeile, also diejenige, die mit der Silbe "spaltig" beginnt, als erste Zeile der zweiten Spalte haben. Um dies zu erreichen, setzen wir ganz frech mitten in den Text, in diesem Fall sogar mitten in ein Wort, eine Formatzeile! Da Trennvorschläge nicht über eine Formatzeile hinweg funktionieren, mußt Du in diesem Fall das zerschnittene Wort "von

Hand" trennen. Die entsprechende Textsequenz mitsamt Formatzeile sähe also folgendermaßen aus:

```
... Zwei-
F x=330 y=42
spaltig ...
```

Du brauchst das "Zwei-" nicht mal mit einem RETURN abzuschließen. Wenn Du's nämlich tust, dann wird diese Zeile als Absatzende betrachtet und folglich nicht im Blocksatz gedruckt, auch wenn Du ihn mit 's=1' angewählt hast.

Ich möchte aber sofort darauf hinweisen, daß dies nicht die Methode ist, wie Du die zweispaltige Formatierung zukünftig realisieren sollst! Dieses Beispiel diente nur zur Übung der Textpositionierung und zur Demonstration der Vorteile des IF-Befehls, den ich Dir in einem späteren Kapitel erkläre. Jetzt mußtest Du die Stelle, an der die Formatzeile einzusetzen ist, erst durch Probieren herausknobeln - auch bei nachträglichen Textänderungen. Beim IF-Befehl ist dies nicht nötig!

Das obige Beispiel ist nicht nur umständlich, es verstößt auch gegen eine wichtige Regel: Der Text vor einer Formatzeile sollte grundätzlich mit RETURN abgeschlossen sein, andernfalls kann es durchaus in einigen Fällen Ärger geben. In diesem Fall jedoch funktioniert's und wir haben tatsächlich erreicht, was wir wollten: Einen hübsch zweispaltig formatierten Text!

Bleibt noch zu erklären, wie ich auf die Werte 'x=330' und 'y=42' kam. Der x-Wert ist einfach: Ihn erhalten wir, indem wir zum x-Wert der linken Spalte, nämlich 20, den Wert der Zeilenlänge, 290, und dazu noch die

Breite des Spalten-Zwischenraumes, 20, addieren. Komplizierter ist der y-Wert der rechten Spalte, der ja dem der linken Spalte entsprechen soll, denn die beiden Spalten sollen in gleicher Höhe beginnen. Wir wissen, daß die Überschrift bei 'y=20' beginnt. Bis zum Spaltenanfang kommen nun noch zwei Zeilen, nämlich die Überschrift und eine Leerzeile, dazu. Eine Zeile ist 11 Punkte hoch, das ergibt sich aus der Zeichenhöhe von 9 Punkten und dem Zeilenabstand 'v=2' (der ist voreingestellt). Somit fängt die linke Spalte bei $y=20+11+11=42$ an, wo wir nun auch die rechte Spalte beginnen lassen.

Natürlich wäre man auch durch simples Nachmessen im Grafikeditor auf diesen Wert gekommen. Beide Methoden sind jedoch umständlich und daher nicht praktikabel. Möchtest Du die Überschrift durch einen größeren Zeichensatz hervorheben, verschiebt sich die Oberkante der linken Spalte und das ganze Theater geht wieder von vorne los.

Deshalb ist es sinnvoller, bereits die linke Spalte bei einem festen y-Wert beginnen zu lassen. Erweitere folglich die Formatzeile nach der Überschrift auf folgende Form:

F y=42 l=290

Zwischen dieser Formatzeile und dem folgenden Text darf nun keine Leerzeile mehr sein, sonst würde der Text erst in der zweiten Zeile der Spalte beginnen. Dieser y-Befehl ändert nun nichts am Aussehen des Textes, doch wir ersparen uns die Rumrechnerei beim Festlegen der rechten Spalte - wir übernehmen jetzt einfach den y-Wert - und beim Ändern der Überschrift gibt's keine ungewollten Verschiebungen.

Natürlich müßten wir nicht den Wert 42 nehmen, der sich hier nur durch die Herleitung ergab. Tatsächlich ist dieser Wert für Perfektionisten nicht befriedigend. Warum, verbischt nochmal? Das erkläre ich in Kapitel 1.4., bei den Tips für Profis.

1.2. Tabulatoren und Einrücken

Mit den Befehlen 'x', 'y', 'l' und 'i' wird der äußere Rahmen, in den der Text eingepaßt werden soll, festgelegt. Doch auch innerhalb dieses Rahmens sind noch Formatierungen möglich: Einrückungen und Tabulatoren. Die Steuerzeichen CTRL T und CTRL I rücken eine oder mehrere Zeilen innerhalb des äußeren Textrahmens bis zu bestimmten Positionen nach rechts. Diese Positionen müssen mit dem Formatbefehl 't' in einer Formatzeile angegeben sein.

Bei der Angabe der Positionen ist zu beachten, daß sie immer relativ zum Zeilenanfang gelten. Mit dem Befehl 't=20' wird der Text um 20 Punkte gegenüber dem Zeilenanfang verschoben und nicht etwa auf x-Koordinate 20. Dadurch ist gewährleistet, daß auch bei nachträglicher Änderung des äußeren Rahmens, zum Beispiel einer neuen Plazierung eines Textabschnitts, die Einrückungen und Tabulatoren unbeeinflußt bleiben.

Was ist nun der Unterschied zwischen einem Tabulator (CTRL T) und einer Einrückung (CTRL I)?

Ein Tabulator bewirkt nur eine einmalige Verschiebung des Textes: Der unmittelbar auf das Steuerzeichen CTRL T folgende Text wird

bis zu einer bestimmten Position geschoben. Diese Position muß in einer Formatzeile im Befehl 't' angegeben sein. Maximal 15 Positionen können für eine Zeile definiert werden. Nehmen wir ein Beispiel mit 4 Positionen. Der Formatbefehl dafür könnte so aussehen:

t=50,100,150,200

Das erste CTRL T in einer Zeile verschiebt nun den Text bis Position 50, das zweite CTRL T in der selben Zeile bis Position 100, das dritte bis 150 und das vierte bis 200. Folgen weitere CTRL T's in der selben Zeile, so werden sie ignoriert, da für sie keine Positionen mehr definiert sind. Ebenso wird ein CTRL T ignoriert, wenn die entsprechende Position bereits überschritten ist. Beispiel: Beim ersten CTRL T, das den Text auf Position 50 schieben sollte, reicht der Text schon bis Position 60.

Eine Einrückung, die im Text durch CTRL I markiert wird, hat zunächst viele Gemeinsamkeiten mit einem Tabulator: Auch CTRL I verschiebt den nachfolgenden Text bis zu einer Position, die ebenfalls in einer Formatzeile mittels 't'-Befehl definiert werden muß. Während jedoch CTRL T nur ein einziges Mal verschiebt, wirkt CTRL I auch auf die folgenden Zeilen. Alle folgenden Zeilen werden also bis zur selben Position eingerückt, erst durch ein Absatzende (RETURN) wird die Einrückung aufgehoben. Es können auch hier mehrere CTRL I's in einem Absatz hintereinanderstehen. Zwei CTRL I's zum Beispiel bewirken, daß die folgenden Zeilen bis zur zweiten in 't' definierten Position eingerückt werden. Auch beim Überschreiten einer angegebenen Position gilt für eine Einrückung das selbe wie beim

Tabulator: Reicht der Text bereits bis Position 60, wenn ihn ein CTRL I nach Position 50 schieben soll, so wird dieses CTRL I zunächst ignoriert. Die folgenden Zeilen jedoch werden bis Position 50 eingerückt. Diese Eigenschaft erlaubt es, aus einem Absatz die erste Zeile herausstehen zu lassen.

Ein Beispiel für diesen Trick sowie eine Reihe weiterer Anwendungen von Tabulatoren und Einrückungen findest Du im File "TABDEMO.FT" auf Disk 3. Formatiere es und sieh' Dir das Ergebnis an.

Wenn Du dieses Demo genau analysierst, weißt Du eigentlich alles über Tabulatoren und Einrücken. Eine Anmerkung noch zu den Tabulatoren: Im Demo sind die Zeilen der Tabelle alle mit RETURN abgeschlossen. Das ist nötig, damit die Tabulatoren jeder Zeile wieder ab eins durchgezählt werden. Würdest Du die RETURN's weglassen, würden die Tabulatoren ab 5 weitergezählt und folglich ignoriert, da keine Positionen über 5 definiert sind. Du kannst das leicht ausprobieren, indem Du mittels Replace-Befehl alle RETURNs in der Tabelle rausnimmst: Cursor an den Tabellenanfang stellen, 'C= R' eingeben, bei 'Suchen' gibst Du 'CTRL RETURN' ein, bei 'Neu' gar nichts. Die Frage 'Ersetzen/Überspringen' beantwortest Du mit 'SHIFT RETURN' (falls Dir diese Eingaben alle recht fremd vorkommen, dann lies in der Bedienungsanleitung das Kapitel zum Such- und Ersetzbefehl). Nun formatierst Du die mißhandelte Tabelle und versuchst, das merkwürdige Ergebnis zu erklären!

1.3. Der IF-Befehl

Sicherlich hast Du schon einmal in Basic programmiert. Dann wird Dir auch der Befehl 'IF' bekannt sein. Abhängig von einer Bedingung verzweigt dieser Befehl im Programm-lauf und ermöglicht somit die Reaktion auf bestimmte Situationen. Der IF-Befehl lässt das Programm intelligent erscheinen, weil es mit seiner Hilfe selbst Entscheidungen treffen kann.

Auch der Printfox besitzt einen solchen IF-Befehl: Er heißt hier 'i' und kann in Formatzeilen eingebaut werden. Ein einfaches Beispiel für eine Anwendung dieses Befehls im Printfox ist bereits in der Bedienungs-anleitung ausführlich dargelegt: Der mehrspaltige Druck. Ich will dieses Beispiel hier wieder aufgreifen, da wir in Kapitel 1.1. bereits versucht haben, einen Text zweispaltig zu formatieren.

Dort haben wir den Spaltenumbruch von Hand vorgenommen, indem wir an die richtige Stelle eine Formatzeile einsetzen, die den Text an den Anfang der zweiten Spalte plazierte. Das mühsame Heraussuchen dieser Stelle soll nun der Printfox übernehmen. Wir teilen ihm mit, daß er die zweite Spalte beginnen soll, wenn in der ersten Spalte der untere Rand erreicht ist. Die Bedingung lautet also "unterer Rand erreicht" und die Aktion, die dann ausgeführt werden soll, ist das Setzen neuer Werte für x und y. Die Bedingung müssen wir noch präzisieren: Unterer Rand erreicht, d.h. wenn eine bestimmte y-Koordinate erreicht oder überschritten ist. Diese y-Koordinate übergeben wir mit dem 'i'-Befehl.

Nun zur Praxis: Lade wieder den "PROBETEXT.FT", füge am Anfang folgende Formatzeile ein:

F x=20 y=20 l=600 s=1

Und nach der Überschrift, unmittelbar vor dem Text folgende Formatzeile:

F l=290 y=42 i=780 y=42 x=330

Bis zum 'i'-Befehl ist uns das alles schon bekannt. Nun zum 'i': Alle Befehle, die in der selben Formatzeile hinter diesem Befehl stehen, werden erst ausgeführt, wenn die aktuelle y-Koordinate den im IF-Befehl angegebenen Wert erreicht oder überschritten hat. In unserem Fall kommt somit die erste Zeile, die unterhalb von y=780 gedruckt würde, in die zweite Spalte. Überzeuge Dich davon, indem Du den Text nun formatierst!

Hast Du alles richtig gemacht, so ist das Ergebnis genau das selbe wie in Kapitel 1.1. Allerdings mußten wir diesmal die Stelle für den Spaltenumbruch nicht umständlich durch Probeformatieren suchen. Das hat diesmal der Printfox für uns erledigt. Auch bei nachträglichen Änderungen, durch die der Text länger oder kürzer wird und sich der Umbruch damit verschiebt, brauchen wir uns nicht mehr darum zu kümmern. Probier's aus, indem Du in die erste Formatzeile den Befehl 't=20' einfügst und damit die CTRL I's aktivierst, die bisher nutzlos im Text herumlungerten. Durch die Einrückung verschiebt sich der Text nach unten, die Umbruchstelle ändert sich.

Nun bleibt noch zu klären, warum ich 780 als y-Wert beim IF-Befehl angab. Schließlich könnte man die Unterkante der Seite doch auch mit 'i=799' angeben. Probier's aus!

Ergebnis: Die letzte Zeile ist nur noch teilweise vorhanden. Der Grund: Für die Positionierung ist immer die Oberkante der Zeilen relevant! Mit 'y=20' beispielsweise wird die Zeile so gedruckt, daß ihre Oberkante auf y-Koordinate 20 zu liegen kommt. Ebenso beim IF-Befehl: Erst wenn die Oberkante einer Zeile den IF-Wert erreicht oder überschritten hat, tritt der Befehl in Aktion. Die abgeschnittene Zeile jedoch begann bei y=797, also noch vor 'i=799'.

Daraus folgt: Soll mit einem IF-Befehl eine Abfrage formuliert werden, bei der die Zeilenunterkante interessiert, so muß man erst auf die Oberkante zurückrechnen. Die Oberkante erhält man, indem man von der Unterkante die Zeichenhöhe subtrahiert. Die Unterkante ist hier 799, die Zeichenhöhe 9. Folglich muß der IF-Befehl in unserem Fall 'i=790' lauten, wenn wir sicherstellen wollen, daß die Zeile nicht abgeschnitten wird. Probier's aus! Du wirst feststellen, daß nun eine Zeile später umgebrochen wird als mit 'i=780'. 780 ist nur ein Standardwert von mir, den ich fast immer verwende. Mit diesem Wert ist immer noch etwas Luft drin, die ich vielleicht brauche, wenn's mit dem Platz knapp wird.

Und jetzt dreispaltig! Dazu teilen wir zuerst die 640 Punkte auf: 3*200 für die Spalten und 4*10 für die zwei Ränder und zwei Zwischenräume (Merke: Man vermeide schwierige Kopfrechnungen, das gibt Kopfschmerzen). Die erste Formatzeile bleibt unverändert, die zweite sieht so aus:

```
F x=10 y=42 l=200 i=780 y=42 x=220 i=780
y=42 x=430
```

Auch hier werden die Befehle hinter dem ersten 'i' erst abgearbeitet, wenn die y-Koordinate den Wert 780 überschritten hat. Einer dieser Befehle setzt den y-Wert wieder auf 42. Der zweite IF-Befehl wird somit erst ausgeführt, wenn der y-Wert zum zweiten Mal die 780 überschritten hat.

Nun könnte jemand auf die Idee kommen, die IF-Befehle auf verschiedene Formatzeilen zu verteilen - denn generell dürfen ja beliebig viele Formatzeilen in den Text eingebaut werden. Das könnte so aussehen:

```
F x=10 y=42 l=200
F i=780 y=42 x=220
F i=780 y=42 x=430
```

Probier's aus! Du wirst staunen: Die zweite und dritte Spalte sind vertauscht. Warum? In diesem Fall sind beide IF-Befehle gleichzeitig aktiv, d.h. sie treffen gleichzeitig zu. Welcher der beiden zuerst abgearbeitet wird, ist durch die Programmstruktur des Printfox bedingt - in diesem Fall ist's der zweite.

Merke Dir deshalb: IF-Befehle, die der Reihe nach abgearbeitet werden sollen, gehören unbedingt alle in eine einzige Formatzeile! Folglich darf eine Formatzeile auch beliebig lang werden.

Neben dem mehrspaltigen Layout dient der IF-Befehl zur Aussparung der Flächen für Grafiken. Darauf wird in Kapitel 3 ausführlich eingegangen. Zur Übung kannst Du schon den DEMO-Text von der Printfox-Diskette analysieren. Dort kommen eine ganze Reihe von IF-Befehlen vor:

```
F y=40 z=1 l=290 s=1 t=20
F i=220 l=170 i=400 l=290 i=780 y=40
x=330 l=290 i=220 l=170 x=450 i=400 l=290
x=330
```

Der erste IF-Befehl sorgt dafür, daß bei y-Koordinate 220 die Spalte verschmälert wird. Die freie Fläche für das Bild beginnt somit spätestens bei 229 (wir erinnern uns: Unterkante = Oberkante + Zeichenhöhe). Bei frühestens y=400 nimmt die Textspalte wieder die ursprüngliche Breite an (in diesem Fall ist die Zeilenoberkante von Interesse, also keine Umrechnung auf 409!). Das dritte 'i' kommt uns bekannt vor, es sorgt für den Spaltenenumbruch. Der vierte und fünfte IF-Befehl besorgen analog zum ersten und zweiten das Aussparen des Bildes in der rechten Spalte. Im Gegensatz zur linken Spalte muß bei der rechten auch die x-Koordinate geändert werden, da sich der Zeilenanfang verschiebt.

Zugegeben: Dieses Beispiel ist schon recht kompliziert. Doch das Ergebnis rechtfertigt den Denk-Aufwand. Als Kontrastprogramm kannst Du Dir nun das File "NEWS.FT" auf der Printfox-Diskette ansehen: Dort wurde nur das kleine Bild rechts oben mit IF's ausgespart. Die anderen Bilder und insbesondere der Spaltenenumbruch kommen ohne IF's aus. Beim Spaltenenumbruch hätte ein IF auch überhaupt nichts gebracht, denn in der zweiten Spalte beginnt sowieso ein neuer Abschnitt. Dieser soll genau dort und nirgends anders beginnen, also setzt man ihn mit 'x' und 'y' genau dorthin. IF's bringen also nur dann etwas, wenn eine Textstelle nicht an einer bestimmten Position erscheinen soll. Im DEMO zum Beispiel ist es egal, ja sogar unbekannt, an welcher Stelle des Textes nun die Aussparungen für das Bild einsetzen und nach

welcher Zeile umgebrochen wird.

Ich hoffe, Du kannst den IF-Befehl nun anwenden. In Kapitel 3 werden wir ihn noch oft brauchen!

1.4. Tips und Tricks

Alle Befehle für das Textlayout sind nun besprochen: x, y, l, i, t und die Steuerzeichen CTRL T und CTRL I. Der Vollständigkeit halber muß man zu dieser Gruppe noch den Befehl für die Satzart 's' und das Zentrier-Steuerzeichen CTRL C rechnen, doch gibt hierzu die Bedienungsanleitung bereits erschöpfend Auskunft.

Nachdem Du nun die Layout-Befehle perfekt beherrscht, hier noch einige Tips für Perfektionisten:

Bereits bei unserem ersten Beispiel stellten wir fest, daß der Rand rechts neben dem Text breiter war, als er laut Formatanweisungen sein sollte. Oder andersherum ausgedrückt: Die Zeilenlänge wurde nicht eingehalten. Der Grund: Printfox betrachtet ein Zeichen zusammen mit dem folgenden Zeichenabstand als eine Einheit. Der Zeichenabstand wird bekanntlich mit dem Befehl 'h' festgelegt oder, wenn dies nicht geschehen ist, hat er den Defaultwert 1. Printfox füllt die Zeile bis zur angegebenen Länge mit solchen Zeichen-Lücken-Einheiten auf, auch das letzte Zeichen hat noch einen folgenden Zeichenabstand - und dies ist genau der Betrag, der zur vorgegebenen Zeilenlänge scheinbar fehlt.

Folge: Die Zeile ist genau um den Zeichenabstand zu kurz. Bei breiter Schrift verdoppelt sich dieser Betrag. Meist stört das nicht, wahrscheinlich ist es Dir bisher noch gar nicht aufgefallen (oder?). Bei betragsmäßig größeren Zeichenabständen kann dieser Effekt jedoch auffallen. Besonders sei hier die Schreibeschrift (ZS80) genannt, die einen Zeichenabstand von 'h=-8' erfordert. Der Pferdefuß dabei ist das Minus: Die Zeile wird jetzt um bis zu 8 Punkte zu lang! Das ist besonders peinlich, wenn Du den Text bis an den rechten Rand des Grafikspeichers formatierst.

"Bis zu 8 Punkte" sage ich, weil sich der Betrag hier nach der Form des Zeichens richtet. Ein Zeichen mit Oberlänge, zum Beispiel ein 'd', steht weiter über die Zeile hinaus als ein 'm'.

Beseitigen kannst Du diesen Effekt ganz einfach, indem Du zur gewünschten Zeilenlänge den Zeichenabstand addierst. Für eine Zeilenlänge von 600 Punkten müßtest Du bei ZS1 also 'l=601' schreiben, bei ZS80 dagegen 'l=592'. Bei einem Fehler von nur ein oder zwei Punkten ist das den Aufwand wohl nicht wert, doch bei ZS80 wird der Perfektionist die Sache wohl ernst nehmen.

Nun ein weiterer Aspekt für Perfektionisten: In Kapitel 1.1. deutete ich an, daß der dort verwendete y-Wert mit 'y=42' noch nicht befriedigend ist. Warum? Weil an der Trennlinie zwischen erster und zweiter Seitenhälfte eine Zeile zerschnitten wird! Generell stört das nicht, denn Printfox bringt auch zerschnittene Zeilen richtig zu Papier. Beim High-Quality-Druck läßt sich aber eine dünne, weiße Linie in einer zerschnittenen Zeile nicht vermeiden. Deshalb

achtet der Perfektionist darauf, daß eben keine Zeile zerschnitten wird. In unserem Fall befinden sich 6 Punkte der Zeile in der ersten Hälfte, 3 Punkte in der zweiten. Schiebt man den ganzen Text um 3 Punkte nach oben oder um 6 Punkte nach unten, ist die Zeile "aus dem Schneider"!

2. Textgestaltung

Dieses Kapitel befaßt sich mit den Befehlen, die das Aussehen des Textes beeinflußen. Dazu gehört die Wahl des Zeichensatzes ('z'), der Größe ('g' und CTRL B), des Zeichen- und Zeilenabstandes ('h', 'v') sowie die Steuerzeichen für die Optionen Fett (CTRL E), Unterstreichen (CTRL U) und das Hoch- und Tiefstellen (CTRL S, CTRL Pfeil-nach-oben und 'u'). Die Anwendung dieser Befehle ist recht einfach, sodaß sich dieses Kapitel auf einige Tips und Tricks beschränkt.

2.1. Initialen

Es ist im Printfox nicht möglich, verschiedene Zeichensätze in einer Zeile zu mischen. Es gibt jedoch einen Fall, indem dies durchaus wünschenswert wäre: Bei Initialen, also großen Buchstaben am Anfang eines Kapitels oder einer Seite. Mit etwas Kopfrechnen bringt man's natürlich doch hin.

Es gibt zwei Arten, Initialen in den Text zu setzen: Erstens: Das Zeichen steht auf der Grundlinie der ersten Zeile und somit über diese hinaus. Zweitens: Das Zeichen steht nicht über die erste Zeile hinaus und nimmt somit mehrere Zeilen im Text in Anspruch. Auf der folgenden Seite sind beide Möglichkeiten dargestellt.

Beginnen wir mit der ersten Möglichkeit. Wir wollen einen mit ZS10 geschriebenen Text mit einem ZS71-Zeichen einleiten. Hier kommt es darauf an, die y-Koordinaten so zu berechnen, daß ZS71 und ZS10 auf der gleichen Grundlinie

Es war einmal ein kleiner, grüner Frosch, der lebte in einem kleinen, grünen Teich. Plötzlich kam ein kleines, grünes Ufo mit kleinen, grünen Männchen an Bord. Auch eine kleine, grüne Prinzessin war dabei. Sie küßte den kleinen, grünen Frosch und siehe dar: Der kleine, grüne Frosch verwandelte sich in einen großen, gelben Briefkasten. Seitdem kommen keine Ufos mehr auf die Erde.

Es war einmal ein kleiner, grüner Frosch, der lebte in einem kleinen, grünen Teich. Plötzlich kam ein kleines, grünes Ufo mit kleinen, grünen Männchen an Bord. Auch eine kleine, grüne Prinzessin war dabei. Sie küßte den kleinen, grünen Frosch und siehe dar: Der kleine, grüne Frosch verwandelte sich in einen großen, gelben Briefkasten. Seitdem kommen keine Ufos mehr auf die Erde.

stehen. Die Text-Oberkante sei 'y=20'. Da ZS71 eine Höhe von 22 Zeichen ohne Unterlänge hat (siehe Anhang in der Characterfox-Bedienungsanleitung), kommt die Grundlinie auf y-Koordinate 42. ZS10 ist 10 Punkte ohne Unterlänge. Somit müssen wir ihn auf 'y=32' setzen, damit er auf die gleiche Grundlinie kommt. Zusätzlich müssen wir die erste Zeile per Tabulator um die Breite des Initials einrücken. In der Praxis sieht das dann so aus:

```
F y=20 z=71
E
F y=32 z=10 t=20
(CTRL T)s war einmal ....
```

Vergiß nicht, das 'E' mit RETURN abzuschließen, denn wie schon angedeutet, soll der Text vor einer Formatzeile immer mit RETURN abgeschlossen sein. Am Besten siehst Du Dir das ganze im File "INITIALS.FT" an. Dort findest Du auch ein Beispiel für den zweiten Fall:

Hier soll ein mit ZS1 geschriebener Text mit einem doppelt hohen und breiten Buchstaben aus ZS30 beginnen. Die y-Koordinate ist nun für beide Schriften gleich, da ja die Oberkante (und nicht die Grundlinie) übereinstimmen soll. Das Problem ist nun, daß mehrere Zeilen eingerückt werden müssen. Dies tut man am Besten mittels 'i'-Befehl. Eine Einrückung mittels CTRL I ist ungeeignet, da kein Absatzende zur Beendung der Einrückung vorhanden ist. Hier gleich die nötigen Formatzeilen:

```
F z=30 g=3 y=20 x=20 l=290
E
F z=1 g=0 y=20 x=52 l=258 i=52 x=20 l=290
s war einmal ....
```

In der zweiten Formatzeile wird der linke Rand auf 'x=60' verschoben und die Länge entsprechend verkürzt. Ab y-Koordinate 52 darf der Text wieder die volle Breite annehmen. 52 ergibt sich aus $20+2*16$, ich bin also von der Höhe ohne Unterlänge ausgegangen. Hätte das Initialzeichen eine Unterlänge, müßte man dies freilich berücksichtigen ('i=60'). Die Breite eines Initialzeichens, die den Betrag der Einrückung bestimmt, schätzt Du einfach. Wenn's falsch war, kannst Du ja die Breite abzählen und nochmal formatieren.

Neben der Verwendung der normalen Zeichensätze für Initialen gibt's noch die Rieseninitialen auf der Characterfox-Diskette. Wegen ihrer Größe sind sie als Grafiken gespeichert und werden auch so behandelt: Du setzt den gewünschten Buchstaben im Grafik-Editor an seinen Platz, mißt die Koordinaten aus und formatierst nun den Text mittels 'i'-Befehl wie im obigen Beispiel. Eine ausführlichere Besprechung dieser Vorgehensweise findest Du in Kapitel 3, wo es um Grafiken geht.

2.2. Special Effects

Wie Du weißt, kannst Du mittels Characterfox besonders effektvolle Zeichensätze erstellen: Outline und Shadow sind nur zwei der vielen Möglichkeiten, die Du im Characterfox realisieren kannst. Doch Printfox kann solche Effekte mit den normalen Zeichensätzen auch realisieren. Es können sogar Effekte erreicht werden, die mit dem Characterfox gar nicht möglich wären, wie die Demos auf dieser Seite beweisen.

Special Effects **Special Effects**

Special Effects **Special Effects**

Special Effects

Special Effects

Das Textfile, mit dem diese Demos erstellt wurden, findest Du unter dem Namen "EFFECTS.FT" auf Disk 3. Lade dieses File und drucke es mittels 'C= P' in den Grafikspeicher. Im Grafikeditor invertierst Du nun Bildschirm 1 ('I'), gehst dann in den Texteditor zurück und druckst nun die zweite Seite des Files. Dabei mußt Du die Frage "Grafik löschen" mit 'N' für Nein beantworten. Im Grafikeditor invertierst Du

Bildschirm 1 dann ein zweites Mal.

Nun die Erklärung: Die ersten beiden Demos sind ganz einfach: Hier wurde der Text ein zweites mal um einen Punkt in x- und y-Richtung verschoben gedruckt. Dies ergibt einen schattierten Fettdruck, den Du natürlich auch mit dem Characterfox realisieren kannst. Braucht man einen Effect jedoch nur schnell für eine Überschrift, so lohnt es sich kaum, dafür einen eigenen Zeichensatz zu erstellen.

Interessanter, weil mit dem Characterfox nicht realisierbar, sind die folgenden beiden Effekte. Dabei entspricht der erste genau der Vorgangsweise, wie man auch im Characterfox einen Outline-Zeichensatz erstellt: Die Schrift wird zunächst neunmal, nämlich dreimal neben- und untereinander, gedruckt. Aus der mittleren dieser neuen Positionen wird die Schrift wieder herausgelöscht, indem sie in die invertierte Grafik gedruckt wird (invertiertes Drucken = Löschen). Was ist nun der Unterschied zum Characterfox? Beim Characterfox wird jedes einzelne Zeichen getrennt mit einem Rand versehen, dagegen beim hier beschriebenen Effekt gilt das ganze Wort als eine Einheit. Besonders deutlich wird dies, wenn man die Buchstaben eng zusammenrückt, wie hier mit 'h=-1' geschehen. Dadurch entstehen zwischen den einzelnen Buchstaben keine trennenden Linien mehr. Die ersten drei Buchstaben von "Effects" zum Beispiel ergeben ein zusammenhängendes Zeichen!

Nach dem gleichen Prinzip funktioniert auch das letzte und eindrucksvollste der vier Demos. Hier wird die Zeile zunächst 18 mal, nämlich 6 mal nebeneinander und 3 mal untereinander, gedruckt. Um die Zeile nicht

18 mal in den Textspeicher schreiben zu müssen, habe ich den Fettdruck verwendet. Beim Fettdruck wird ein Zeichen zweimal, um einen Punkt versetzt, nebeneinander gedruckt. Da durch den Fettdruck die Zeichen um einen Punkt breiter werden, muß der Zeichenabstand auf 'h=0' verkleinert werden.

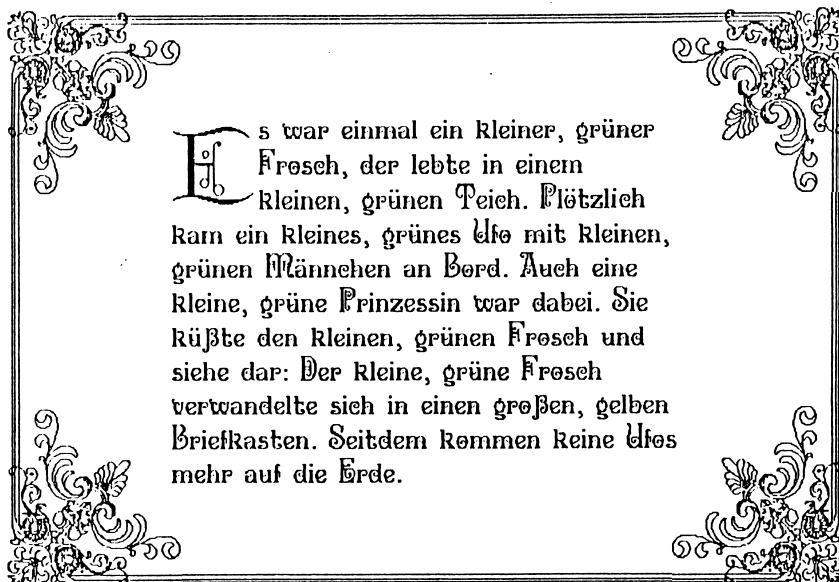
Der Aufwand für dieses letzte Beispiel ist zugegebenermaßen schon ziemlich hoch, doch Du siehst daran einmal mehr, daß Dir Printfox kaum Begrenzungen auflegt. Mit etwas Phantasie und Knobeln kannst Du aus diesem Programm Ungeahntes herausholen!

3. Text und Grafik

Nun kommen wir endlich zur Grafik, genauer gesagt zur Kombination von Text und Grafik. An zwei Beispielen wird in diesem Kapitel Schritt für Schritt die Erstellung kompletter Layouts, bestehend aus Text und Grafik, erklärt.

3.1. Plazieren, Rahmen und Initialen

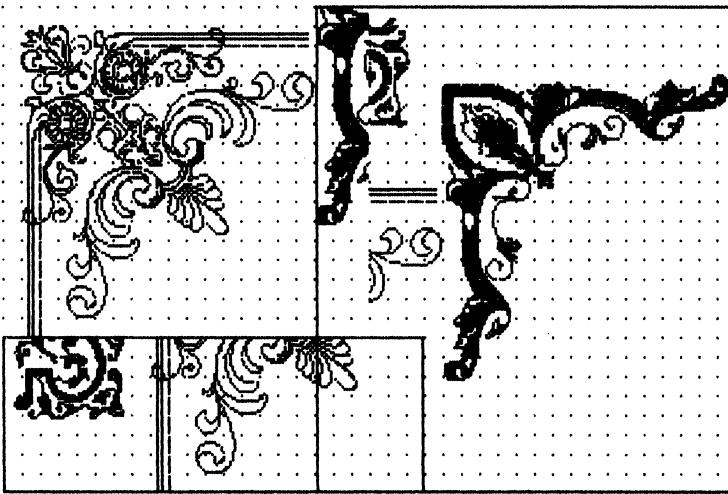
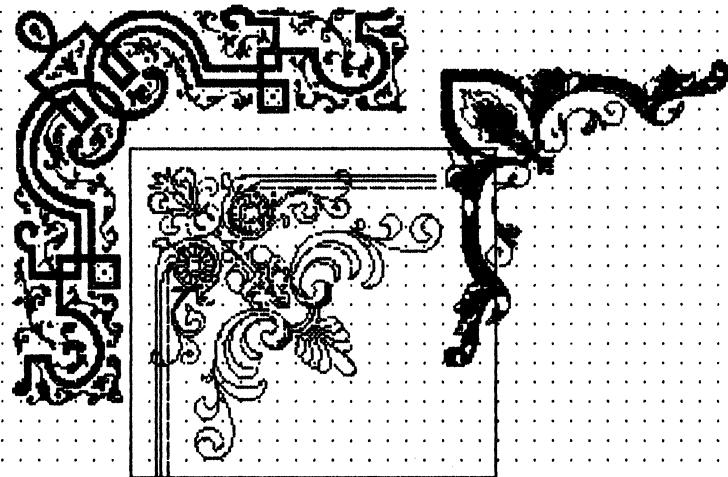
Als erstes Beispiel wollen wir das Märchen vom kleinen, grünen Frosch, das wir schon aus Kapitel 2 kennen, in eine wirklich märchenhafte Form bringen:



Anhand dieses Beispiels soll die Plazierung von Grafiken und dabei insbesondere die Benutzung des Move-Befehls im Grafik-Editor demonstriert werden.

Als erstes wollen wir den schönen Rahmen erstellen. Wir benutzen dazu das File "ECKEN2.GB" von der Characterfox-Diskette. Lade dieses File in den Grafikeditor. Dieses Gesamtbild enthält die Ecken für vier verschiedene Rahmen. Da der Printfox nicht drehen und spiegeln kann, sind in den vier Bildschirmen die Elemente für alle vier Ecken eines Rahmens enthalten. Für Superscanner-Besitzer hätte freilich ein Bildschirm gereicht, die anderen kann der Superscanner durch Drehen und Spiegeln erzeugen.

Beginnen wir nun in Bildschirm 1 mit der Plazierung: Das Element unten in der Mitte soll nach links oben verschoben und alles übrige gelöscht werden. Dies ist mit nur drei Move-Operationen möglich. Als erstes schaltest Du das Punktraster ein. Da es die 8*8-Punkte-Auflösung des Move-Befehls markiert, ist es ein wertvolles Hilfsmittel bei allen Verschiebe- und Lösch-Funktionen. Markiere den Quellbereich etwa so, wie in der folgenden Abbildung angedeutet: Links und oben läßt Du noch ein Kästchen dran, damit wir dem Element nichts abzwacken und es nicht zu nah an den Rand quetschen. Wie weit Du den Bereich nach rechts ziehst, ist egal. Zur Markierung des Zielbereiches setzt Du den Cursor in die Home-Position und drückst den Feuerknopf.



Nun muß das Ganze etwa so wie in der zweiten Abbildung aussehen: Das gewünschte Element ist an seiner endgültigen Stelle. Alles übrige löscht Du nun in zwei Bereichen, wie in der Abbildung angedeutet, indem Du diese beiden Bereiche mit sich selbst EXOR-verknüpft. Du markierst also den zu löschenen Bereich, setzt den Cursor in dessen linke, obere Ecke, tippst 'X' und drückst dann den Feuerknopf (oder RETURN). Diese Art des Löschens ist für große Flächen die schnellste. Soll's mal etwas genauer gehen, so benutzt Du den Sprite-Radiergummi im Erase-Modus. Den dicken oder gar den dünnen Pinsel im Draw-Modus wirst Du nur für Feinarbeiten brauchen.

Nun ziehst Du mittels Line-Befehl noch die drei Linien des Rahmens bis an den Bildschirmrand nach rechts und unten und schon ist der erste Bildschirm fertig.

Wenn Du mit dem Superscanner arbeitest, kannst Du nun diesen Bildschirm in die drei anderen kopieren und durch Drehen und Spiegeln in die endgültige Position bringen, um den ganzen Rahmen fertigzustellen. Beim Printfox dagegen mußt Du die Bildschirme 2 bis 4 in der gleichen Weise wie den ersten bearbeiten, also den gewünschten Rahmen plazieren und den Rest löschen. Im Unterschied zum Bildschirm 1 kannst Du bei der Bestimmung des Zielbereichs den Cursor nun nicht mehr in die Home-Position setzen. Vielmehr muß der Cursor nun so positioniert werden, daß der verschobene Bereich gerade bis an den rechten oder/und unteren Rand reicht. Natürlich könntest Du diese Position durch Kästchenzählen herausknobeln, doch so kompliziert machen wir das nicht. Setz ganz einfach den Cursor "pi mal Daumen", also geschätzt, und drück den Feuerknopf. Tut sich

gar nichts, so bist Du zu nah an der rechten oder unteren Kante. In diesem Fall fährst Du den Cursor ein Kästchen von der Kante weg und versuchst es noch einmal. Wurde der Bereich zwar verschoben, ist aber zu weit von der Kante entfernt, so revidierst Du die Operation mittels Undo (Pfeil nach links), holst die Markierung des Quellbereiches zurück (Pfeil nach oben), verschiebst den Cursor um die Anzahl der Kästchen, um die's im ersten Versuch gefehlt hat, und drückst wieder den Feuerknopf.

Hast Du in allen vier Bildschirmen die Ecken richtig plaziert, also mit einem Kästchen Abstand bis zum Bildschirmrand, so passen auch die Rahmenlinien nahtlos zusammen. Dies ist der Vorteil der 8-Punkte-Auflösung des Move-Befehls: Könnte dieser Befehl punktgenau verschieben, so hättest Du höchstwahrscheinlich um ein, zwei oder drei Punkte daneben getroffen. Sich dagegen um 8 Punkte zu verschätzen, ist schon sehr unwahrscheinlich.

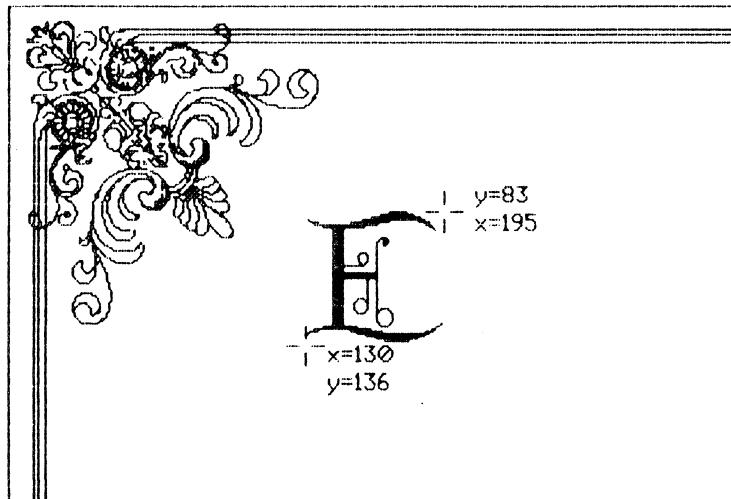
Nun müssen wir das große 'E' in die Grafik setzen. Die Initialen sind alphabetisch auf die beiden Grafikfiles "INITIALS.BS1" und "INITIALS.BS2" auf der Characterfox-Diskette verteilt. Da das 'E' ziemlich weit vorn im Alphabet steht, wird es wohl in "INITIALS.BS1" enthalten sein. Doch bevor Du dieses File frisch-fröhlich lädst, überlege Dir, wohin. Denn einerseits soll der fertige Rahmen nicht überdeckt werden, andererseits brauchst Du links oben Platz, denn dort soll das 'E' hinkommen. Also wählst Du am Besten Bildschirm 4 an und scrollst so weit nach links und oben, daß vom Rahmen gerade nichts mehr zu sehen ist. In dieses freie Plätzchen kannst Du nun "INITIALS.BS1" laden!

Mittels Move holst Du nun das 'E', wählst Bildschirm 1 an und plazierst es dort. Wo genau, bleibt eigentlich Dir überlassen. Damit jedoch die Koordinaten bei der Textformatierung mit den in diesem Beispiel angegebenen Werten übereinstimmen, solltest Du versuchen, das 'E' etwa bei x=130 und y=83 hinzubekommen. Auch hier kannst Du mit Undo und Recall herumexperimentieren, wenn Du die richtige Stelle nicht auf Anhieb triffst.

Nun löscht Du noch den Bildschirm mit den Initialen und fertig ist die plazierte Grafik! Wie in der Printfox-Bedienungsanleitung schon erwähnt, solltest Du dieses Gesamtbild unbedingt abspeichern. Denn wenn Du es durch einen Fehler bei der Textformatierung zerstörst, mußt Du die ganze Plazierung nochmal vornehmen. In diesem Beispiel wär's allerdings nicht so schlimm: Auf Disk 3 ist dieses File unter dem Namen "MARCHEN.GB" vorhanden! Du kannst es zum Vergleich laden, um zu kontrollieren, ob Du alles richtig gemacht hast. Lade dazu dieses File ganz einfach auf Dein Werk drauf, indem Du die Frage 'Mischen' mit 'J' beantwortest.

Nun geht's an die Textformatierung. Im Prinzip geht das ähnlich wie im zweiten Frosch-Beispiel aus Kapitel 2, denn auch dort wurden mehrere Zeilen für den Initial-Buchstaben eingerückt. Lade das File "INITIALS.FT" in den Texteditor und lösche den gesamten ersten Block, entsprechend dem ersten Frosch-Beispiel aus Kapitel 2. Vom zweiten Block kannst Du die erste Formatzeile und das 'E' löschen, denn dieses befindet sich bereits in der Grafik. Nun geht es darum, die verbleibende Formatzeile für das aktuelle Beispiel anzupassen. Dazu sind folgende Fragen zu beantworten:

Bei welcher x- und y-Koordinate soll der Text hinter dem 'E' beginnen? Wann darf er die volle Breite einnehmen und welche x-Koordinate ergibt sich dabei? Um diese Fragen zu beantworten wechselst Du in den Grafikeditor, schaltest die Koordinatenanzeige ein und mißt die relevanten Punkte aus. In der folgenden Abbildung sind diese Meßpunkte dargestellt.



Nun zur Formatzeile: Zunächst wollen wir den märchenhaft schönen ZS70 benutzen. Dann setzen wir den Text einfach an die gemessenen Koordinaten 'x=195' und 'y=83'. Wenn die y-Koordinate den Wert 136 überschreitet, also 'i=136', beginnt der Text bei 'x=130'. Die Zeilenlänge bei voller Textbreite, also nach dem IF-Befehl, ist $640 - 2 \cdot 130$, also 'l=380'. Vor dem IF-Befehl ist der Text um 65 Punkte eingerückt ($195 - 130$), also ergibt sich dort eine Länge von 'l=315'. Hier die komplette Formatzeile:

F z=70 y=83 x=195 l=315 i=136 x=130 l=380

Wenn Du nun den Text formatierst, also in den Grafikspeicher druckst und dort noch die plazierte Grafik steht, dann antworte auf die Frage 'Grafik löschen' mit 'N'. Der Text wird dann in die Grafik hineingedruckt, ohne daß diese gelöscht wird.

Du wirst bei der Betrachtung des fertigen Werkes nun feststellen, daß der Text subjektiv etwas zu tief steht. Der Grund ist wohl, daß ZS70 sehr lange Oberlängen hat, von denen jedoch in der ersten Zeile nicht viele vorkommen, und daß drei Zeilen dieses Zeichensatzes gerade etwas höher als das 'E' sind. Optisch wirkt es besser, wenn Du den Textanfang noch etwas nach oben schiebst, beispielsweise auf 'y=75'.

Wenn Du den Text ein zweites Mal in den Grafikspeicher druckst, mußt Du dafür sorgen, daß die Reste des ersten Versuches gelöscht werden. Generell gibt es hier zwei Möglichkeiten:

Erstens: Du lädst die plazierte Grafik (Mischen 'N') und druckst den Text in die Grafik (Grafik löschen 'N')

oder zweitens: Du druckst den Text in den Grafikspeicher (Grafik löschen 'J') und mischt dazu die plazierte Grafik von Disk (Mischen 'J').

Wie Du's machst, ist egal. Wenn's Dir zu kompliziert ist, entscheide Dich einfach für eine Möglichkeit, die Dir besser zusagt. Wenn Du aber nicht gerne Disketten wechselst, verwendest Du die erste Möglichkeit immer dann, wenn gerade Deine Datendiskette mit dem Grafikfile im Laufwerk liegt und die zweite dann, wenn die Zeichensatzdiskette drin ist. Bei aufeinanderfolgenden Versuchen wechseln

sich die Möglichkeiten somit ab und im Endeffekt sparst Du pro Versuch einen Diskettenwechsel.

Zur Kontrolle Deiner Arbeit ist neben dem Grafikfile "MÄRCHEN.GB" auch das Textfile "MÄRCHEN.FT" auf Disk 3 vorhanden.

3.2. Große Grafiken

In den vorhergegangenen Kapiteln hast Du bereits alles gelernt, was Du für die professionelle Layoutgestaltung mit dem Printfox brauchst. Du kannst Texte formatieren und gestalten, weißt, wie man Grafiken plaziert und die Formatzeilen so erstellt, daß die Texte um die Grafiken herum Platz finden.

Was nun kommt, sind nur noch Feinheiten, zusätzliche Tips und natürlich Übung für das bereits gelernte. So geht es auch in folgenden Beispiel wieder darum, den Text um die Grafik herum zu formatieren. Neu dabei ist, daß die Grafik diesmal größer als ein Bildschirm ist. Das erfordert etwas Nachdenken beim Plazieren, insbesondere im Hinblick auf den Move-Befehl, der nur maximal bildschirmgroße Grafiken verschieben kann.

Doch nun zur Aufgabenstellung: Es geht um ein zweispaltiges Layout mit einer großen Grafik. Auf der folgenden Seite siehst Du bereits das gewünschte Ergebnis. In der Praxis hast Du das Ergebnis natürlich nicht vorliegen. Es ist deshalb durchaus empfehlenswert (und auch für einen Computerprofi keine Sünde), zu Bleistift und Papier zu greifen und sich das gewünschte Layout grob zu skizzieren.

Printfox, der schlauer Druckerfuchs

Gestalten: Printfox. Ich bin ein ganz schlauer Fuchs, der Deinen Drucker zu neuem Leben erweckt. Dazu biete ich Dir ein Textverarbeitungsprogramm und ein Grafikprogramm in einem. Aber keineswegs irgendwelche Sparversionen, sondern von jedem das Beste: Der Texteditor ist weitgehend befehlkompatibel zu Uzawrite, der Grafikeditor ist fast identisch mit dem Superscanner II und Hi-Eddi plus. Doch was mich so unheimlich stark macht, ist die Kombination aus den beiden Programmen! Den Text, den Du im Texteditor schreibst, gebe ich nämlich nicht direkt auf den Drucker aus, sondern "drücke" ihn zunächst in den Grafikspeicher. Dadurch ergeben sich irre Möglichkeiten:



→Zig verschiedene Schriftarten, Formen und Größen, von der einfachen Pica bis hin zu extra großen und schönen Zierschriften!

→Frei und punktgenau einstellbarer Zeichen- und Zeilenabstand, Proportionalsschrift, echte Unterlängen, deutsche und internationale Sonderzeichen, auch auf den Commodore-Druckern MPS 801/803!

→Uoll mischbar mit Grafik! Auch mit dem Superscanner gescannte Bilder können eingelesen werden.

→Unglaubliche Formatiermöglichkeiten, hier nur einige Stichpunkte: Links- oder Rechtsbündig, gleitender Blocksatz (bei dem alle Spaces punktgenau gleich lang sind), zentriert, mehrspaltig mit automatischem Spaltenumbruch, Einrückung, Tabulator

Da ich mir die Zeichensätze von der Diskette nachlade, ist der Uorrat praktisch unbegrenzt. Auf der Programmdiskette werden bereits fünf Zeichensätze mitgeliefert, eine Zusatzdiskette mit mindestens zehn weiteren befindet sich in Urrbereitung.

Jede Schriftart kann in doppelter Breite und/oder Höhe gedruckt werden, dies ist z.B. breite Pica.

Jetzt haben wir doppelt hohe Pica, die ich nun auch breit schreiben kann.

Alle die oben gezeigten Schriftarten und Größen kann ich auch **fett drucken und unterstreichen**. Außerdem darfst Du mir mitteilen, wie viel Platz ich zwischen den Zeichen und den Zeilen lassen soll: Hier lasse ich jetzt z.B. 3 Punkte zwischen den Zeichen, vorher war's nur ein Punkt. Und im Folgenden

probier' ich's mal mit 2 Punkten:

Und dann habe ich eine ganze Menge Satzarten auf'm Kasten. Neben Blocksatz, links- und rechtsbündig kann ich z.B. Zentrieren, wie Du hier siehst. Auch verschiedene Schriftbreiten innerhalb einer Zeile sind für mich kein Problem.

Und ganz nebenbei: Auch Tabulatoren springe ich immer punktgenau an, trotz Proportionalsschrift, Blocksatz und verschiedener Schriftbreiten – versuch' das mal mit einem anderen Textverarbeiter.

Sicherlich ist Dir auch schon die Einteilung dieser Seite aufgefallen: Zweispaltig, und um das Bild herum habe ich den Text zur Seite gerückt. Mit wenigen Befehlen teilst Du mir die "Form" mit, in die ich den Text einpasse.

Kommen wir nun nochmal auf die beiden Programmteile zu sprechen:

In Texteditor benehme ich mich fast wie Uzawrite: Echtes Word-Wrapping, das auch bei heftigem Herumeditieren voll erhalten bleibt, buchstabengenaue Blockoperationen wie Löschen, Verschieben, Kopieren... Ich kann sogar noch einiges mehr: Suchen und Ersetzen kann ich wahlweise mit oder ohne Beachtung der Groß/Klein-Schreibung und mit Joker, ferner habe ich einen komfortablen Einfüge-Modus, eine Caps-Lock-Funktion für Großschreibung, und sowohl Trennvorschläge als auch unterdrückte Trennung. Für die Trennvorschläge mußt Du übrigens kein extra Steuerzeichen eingeben, sondern Du schreibst ganz einfach den Silbenanfang groß, z.B. so PrintFox! Im Text erscheint das natürlich als "Printfox", aber ich weiß, daß ich bei Bedarf dort trennen kann.

Und dann der Grafikeditor, hier ein kurzer Überblick: Grafikspeicher mit 640*400 Punkten, also entsprechend 4 Bildschirmen, frei scrollbar, Punkte, Kreise, Linien und Rechtecke zeichnen, Spray-Funktion, Flächen voll oder gerastert ausfüllen, Bereiche verschieben, kopieren, logisch verknüpfen, echte Zoom-Funktion und Sprite-Editor und eine Undo-Funktion, mit der die letzte Operation rückgängig gemacht wird. Und als ganz besonderes Bonbon kann ich den gesamten Grafikspeicher in Bildschirmgröße darstellen, damit Du einen optimalen Überblick über Dein Werk hast.

Beim Ausdrucken lasse ich dann nochmal mein ganzes Können aufblitzen: Auf Epsom-Druckern und kompatiblen, die über einen Grafikmodus für 1920 Punkte/Zeile und einen Zeilenvorschub in 1/216 verfügen, erzeuge ich echte Near-Letter-Quality-Schrift!

Natürlich braucht das nicht sorgfältig ausgeführt zu sein: Die Grafik deustest Du nur mit ein paar Strichen an, den Text stellst Du als Kasten dar. In diese Skizze kannst Du auch die Koordinaten einträgen, die Du im Grafikeditor aus der plazierten Grafik mißt und für die Formatierung brauchst.

Womit geht's los? Mit dem Plazieren der Grafik! Also holen wir uns diese aus dem File "GIRLS1.GB" auf Disk 1. Zunächst wird mit der EXOR-Methode und, wenn's knapp wird, mit Erase wieder alles, was wir nicht brauchen, gelöscht. Nun wollen wir die Lady schön in die Mitte rücken, außerdem möchten wir sie ein Bißchen nach unten schieben, damit oben genug Platz für eine große Überschrift bleibt. Nachdem bis zum unteren Rand des Gesamtbildes, also der Trennlinie zwischen den Seiten, noch reichlich Platz ist und sich diese Linie als unterer Bildrand ('i=400') anbietet, können wir um zwei Kästchen, also 16 Punkte, nach unten verschieben. Für die genaue Mitte müssen wir etwas rechnen: Der Rock geht ungefähr von x-Koordinate 200 bis 430. Der Rand links ist also etwa 200, der rechts 210 Punkte breit. Wir müßten genau genommen um 5 Punkte verschieben. Da wir jedoch nur im 8-Punkte-Raster schieben können und zudem der rechte Rockzipfel weiter hervorsteht als der linke, verschieben wir um 8 Punkte nach rechts.

Jetzt geht's darum, eine Grafik, die etwa zwei Bildschirme bedeckt, mittels Move zu verschieben. Weiter kein Problem: Wir verschieben die obere und die untere Hälfte extra, jede ein Kästchen nach rechts und zwei nach unten. Wenn Du allerdings mit der oberen anfängst, überdeckst Du einen zwei Kästchen breiten Streifen der unteren Hälfte, dessen Fehlen sich in der Figur der Lady unangenehm

bemerkbar macht. Du mußt deshalb mit dem Verschieben der unteren Hälfte beginnen! Der Einfachheit halber markieren wir die Bereiche nicht mit dem Cursor, sondern verschieben immer ganze Bildschirme, die man bekanntlich mit 'SHIFT-Pfeil-nach-oben' markieren kann. Wähle den Bildschirmausschnitt so, daß Du die untere Hälfte der Lady siehst, scroll' dann den Bildschirm um zwei Schritte (=Kästchen) nach oben und markiere den ganzen Bildschirm. Dann scrollst Du wieder zwei Schritte nach unten, einen nach rechts, setzt den Cursor in die Home-Position und drückst RETURN. Sodann holst Du die obere Hälfte in den Bildschirm, markierst ihn, scrollst wieder die zwei Schritte nach unten und einen nach rechts und drückst RETURN.

Dies ist nun die fertige, plazierte Grafik und es empfiehlt sich, diese abzuspeichern.

Jetzt soll der Text formatiert werden. Zunächst brauchen wir einen Text, wir benutzen dazu wieder den "PROBETEXT.FT". Die Überschrift soll diesmal in einer größeren Schriftart, zum Beispiel in ZS40, dargestellt werden. Die Formatzeile vor der Überschrift könnte so aussehen:

F z=40 y=10

Wie Du siehst, habe ich die Angaben für 'x' und 'l' weggelassen, womit die Defaultwerte 'x=20' und 'l=600' wirksam sind. Eigentlich sind diese Werte für die Überschrift auch egal, sie müssen nur eine Bedingung erfüllen: Die Überschrift muß in der Mitte stehen. Dies könnte man auch mit 'x=30' und 'l=580' oder 'x=0' und 'l=640' erreichen.

Nun zur zweiten Formatzeile, die unmittelbar vor dem Text einzufügen ist. Als y-Wert nehmen wir diesmal 'y=48', da uns dieser Wert bei ZS1 und Zeilenabstand 2 garantiert, daß keine Zeile an der Trennlinie zwischen erster und zweiter Seitenhälfte zerschnitten wird (siehe Kapitel 1.4.). Der neben der Grafik zur Verfügung stehende Platz ist etwa 200 bis 210 Punkte breit, wie wir von vorher wissen. Um den Text nicht zu dicht an die Grafik zu kleben, entscheiden wir uns für 190 Punkte, wovon 20 für den Rand neben dem Text bleiben sollen. Das ergibt 'x=20' und 'l=170'. Ab y-Koordinate 400 darf dann der Text die volle Zweispalten-Breite von 'l=290' einnehmen. Die x-Koordinate der oberen Hälfte der zweiten Spalte errechnet sich aus 640-190 ergibt 'x=450'. Somit haben wir alle Angaben für die Formatzeile:

```
F z=1 x=20 y=48 l=170 i=400 l=290 i=780
y=48 x=450 l=170 i=400 x=330 l=290
```

Diese Zeile ergibt nun einen wunderschönen Bereichsfehler, weil das letzte Wörtchen des Textes nicht mehr auf die Seite paßt. Jetzt können wir die vorgesehene Reserve von einer Zeile beanspruchen, indem wir 'i=790' statt 'i=780' schreiben, und schon klappt's!

Wenn Du willst, kannst Du nun noch mit der Satzart spielen. Wie Du siehst, habe ich die erste Spalte rechtsbündig, die zweite linksbündig gedruckt. Natürlich kannst Du auch alles im Blocksatz formatieren. Allerdings müssen bei einer Zeilenlänge von nur 170 Punkten viele Trennvorschläge vorhanden sein, damit beim Blocksatz keine zu langen Lücken entstehen.

3.3. Weitere Techniken

Bisher habe wir uns darauf beschränkt, die Grafiken in den Speicher zu laden, zu plazieren und den Text dazu passend zu formatieren. Dies ist der Normalfall und man wird in den allermeisten Fällen auch so vorgehen.

Generell sind aber auch andere Techniken vorstellbar, Techniken, bei denen die Grafik erst nachträglich, auf dem Papier, in das Layout eingefügt wird. Warum sollte man so etwas Merkwürdiges tun, was doch eigentlich den Sinn und Zweck eines kombinierten Text-Grafik-Programmes in Frage stellt? Nun, es gibt durchaus Fälle, die der Printfox nicht abdeckt. Als Beispiel sei hier die Extension zum Superscanner II genannt:

In dieser Extension ist auch ein High-Quality-Zusatz enthalten, der auf Epson-Druckern und kompatiblen Grafiken in einer Qualität ausdrückt, wie sie sonst nur Laserdrucker erreichen. Zwar kennt auch der Printfox einen High-Quality-Ausdruck, doch werden dabei "nur" durch einen Algorithmus Zwischenpunkte errechnet, die insbesondere Schrägen und Kurven glätten. Außerdem erreicht der Printfox mit seinem High-Quality-Modus "nur" eine sechsfach höhere Auflösung (1920*1600 Punkte pro Seite), was etwa der Auflösung der NLQ-Schrift moderner Matrixdrucker entspricht. Die Scanner-Extension bietet jedoch eine bis zu neunfach höhere Auflösung, und dabei werden keine Punkte errechnet, sondern alle "sind echt", werden also auch gescannt. Der Preis für diese Superqualität ist ein immenser Speicherbedarf: Ein 32 KByte großes Gesamtbild ergibt in der höchsten Auflösung gerade noch einen Ausdruck von 5*8

cm. Aus diesem Grund verfügt der Printfox auch nicht über diese Fähigkeit.

Für besonders anspruchsvolle Werke kann es aber durchaus sinnvoll sein, diese Super-Quality-Bilder in Printfox-Layouts einzubauen. Dazu druckt man zuerst das Printfox-Layout, spannt dann das bereits bedruckte Papier ein zweites Mal ein und druckt in die freien Stellen mit dem Superscanner II die gewünschten Grafiken. Natürlich ist diese Methode sehr aufwendig, besonders das genaue Treffen der richtigen Stellen ist recht kritisch. Ein oder mehrere Fehlversuche werden sich dabei kaum vermeiden lassen. Es ist deshalb wichtig, bei jedem Versuch sich die Papiereinstellung (z.B. auf Abreißkante beim FX) genau zu merken und aus eventuellen Fehlversuchen auf die richtige Position umzurechnen.

Die Scanntronik-Anzeige in der 64'er 1/87 und folgenden ist übrigens auf diese Weise entstanden: Der Text und der Rahmen wurden vom Printfox ausgedruckt, die Bilder wurden mit der Scanner-Extension eingefügt. Einen Ausschnitt aus dieser Anzeige in Originalgröße (in der 64'er ist sie auf 2/3 verkleinert) siehst Du auf der folgenden Seite.



Neu Neu Neu Neu COLOURPRINTER

Der erste Farbaufrüstsatz
"normale" Matrixdrucker



Vorbei sind die grauen Zeiten, in denen Ihnen bunten Bildschirm-Gemälden nur weiße Hardcopsys zu Papier bringen. Jetzt drucken Sie farbig, ohne teure drucker! COLOURPRINTER macht's möglich.

Druckt Grafiken aller gängigen Zeichen- und M (Hi-Eddi, Superscanner II, Paint Magic, Ko Blazing Paddles, Doodle, Vidcom 64, Art St Farben und verschiedenen Größen, Double Str Bilder über-, neben- und aufeinander (Collage-Layouts mit Farbbildern versehen, Umfang Farbtöne frei editierbar



Lieferbar für Epson RX-80, FX80/85, in Vort Star NL-10 und andere (nachfragen!).
Preis für Diskette mit Software, vielen Farbband-Set:
DM 138.-

Klebeschiene (für Drucker ohne Papier-Rücksicht)
DM 14.-

Ersatz-Farbband-Set: **DM 49.-**

Ein weiterer Fall, in dem nachträgliches Drucken der Grafik in ein Printfox-Layout sinnvoll ist, ist der Colourprinter, der Farbaufrüstsatz für Schwarz-Weiß-Drucker. Zwar kann der Colourprinter fertige Printfox-Layouts laden und sie, mit Farbbildern versehen, in einem Rutsch ausdrucken, doch hat diese Methode einen Nachteil: Der Colourprinter verfügt über keinen High-Quality-Modus. Wer also höchste Qualitätsansprüche stellt, muß etwas mehr Aufwand in Kauf nehmen.

Neben diesen ausgeklügelten Fällen gibt's auch noch den ganz trivialen Fall, die Bildchen von Hand in das fertige Layout zu malen oder Zeichnungen, Fotos oder anderes einzukleben. Für den eingefleischten Computerfreak ist dies natürlich ein schwerer Stilbruch, aber wenn das Ergebnis stimmt, why not? Schließlich muß man nicht immer alles auf Biegen und Brechen mit dem Computer machen!

4. Sonstiges

Hier noch ein paar Hinweise zum Printfox und seiner Erweiterung auf der Charcterfox-Diskette.

Zunächst zum Printfox: In einem Test des Printfox in der RUN hieß es, der Printfox könne keine Files auf der Diskette überschreiben. Er kann schon, nur muß man dem Filenamen anstelle des Klammeraffen das Paragraphen-Zeichen ('\$') voranstellen, wegen des deutschen Zeichensatzes. Allerdings rate ich von dieser Vorgehensweise ab, denn gelegentlich baut das Floppy-DOS beim Überschreiben Chaos, sodaß Files verloren gehen können. Besser speichert man die Files mit fortlaufender Nummer ab und scratched dann die alten.

Zum Thema Floppy-Speeder: So ein Tempomacher ist eine feine Sache und jeder, der einmal damit gearbeitet hat, möchte ihn nicht mehr missen. Doch ist Speeder nicht gleich Speeder: Insbesondere die seriellen (Software-Speeder oder Hypra-Kernels, kurz alle, die kein zusätzliches Kabel brauchen) sind in Sachen Betriebssicherheit und Verträglichkeit den parallelen unterlegen. So muß man bei einigen den Drucker am seriellen Bus ausschalten - für ein Druckerprogramm wie den Printfox natürlich nicht akzeptabel - und konkret das 64'er-DOS läuft mit Prinfox offenbar überhaupt nicht. Keine Probleme dagegen gibt's mit den gängigen Parallelspeedern wie Speeddos oder Dolphindos. Lediglich den Turbo-Access muß man für den Setup - und nur dort - ausschalten, sonst könnte es Ärger geben.

Ein anderes Kapitel, das gelegentlich für Frust sorgt, sind Hardware-Interfaces, besonders jenes von Görlitz: Es verträgt keine Zeilen mit mehr als 1792 Bytes, die im High-Quality-Modus aber durchaus vorkommen können. Abhilfe: Görlitz bietet eine Aufrüstung seiner Interfaces auf die neuen DIN-Interfaces an, die diesen Fehler nicht mehr haben.

Daß der Drucker bei allen Hardware-Interfaces im High-Quality-Modus vor jeder Zeile eine Bedenkpause einlegt oder kläglich hin- und herruckelt, liegt an der bekannteren langsamem Übertragungsgeschwindigkeit des seriellen Busses. Der billigste und wirkungsvollste Druckerspeeder ist hier ein simples Userport-Kabel!

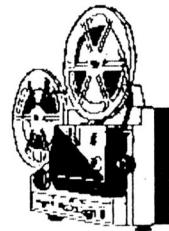
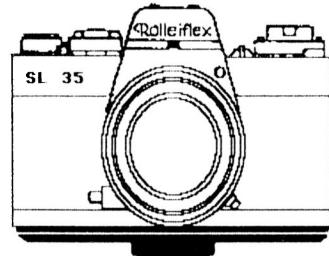
Die Printfox-Erweiterung 'XF' auf der Characterfox-Diskette, die mit dem Befehl 'C= X' aufgerufen wird, kann nicht nur fremde Textfiles und Tastaturbelegungen laden. Sie kann auch einen neuen Bildschirmzeichensatz laden und Printfox-Texte im Format anderer Textprogramme speichern. Für neue Bildschirmzeichensätze empfiehlt es sich, den vorhandenen Zeichensatz als Ausgangsbasis zu verwenden. Um ihn aus dem Printfox herauszubekommen, brauchst Du einen Maschinensprachemonitor. Mit diesem speicherst Du, nachdem Du den Printfox verlassen hast, den Bereich \$3800 bis \$3CA0 auf Diskette. Dies ist der Zeichensatz, den Du nun im Grafikeditor laden kannst. Von dort mußt Du ihn nach Veränderung unkomprimiert, also mit vorangestelltem 'Ø:', saven. Nun kopierst Du 'XF' mit auf die Diskette und kannst nun, nach Aufruf der Erweiterung, diesen Zeichensatz aus dem Directory laden. Um ein Textfile im Format eines anderen Textverarbeiters zu saven, gehst Du zunächst wie beim Laden vor: 'C= X'

und dann eine Convertierungstabelle ('.CT') aus dem Directory laden. Bei der Aufforderung 'Textdisk einlegen' drückst Du nun nicht irgendeine Taste, sondern 'S'! Allerdings wird beim Saven auf diese Art nur die Convertierung der Codes vorgenommen, eventuelle Filekennungen für andere Textverarbeiter fehlen. So können zum Beispiel auf diese Art erzeugte Viza-Files nur mittels Merge-Befehl in Vizawrite geladen werden. Andere Programme, die über eine solche Funktion nicht verfügen, können die Files wahrscheinlich nicht laden.

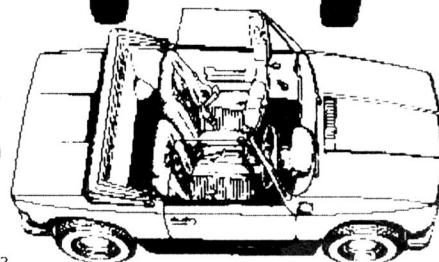
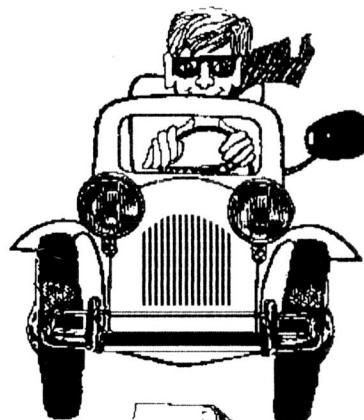
Und dann ist da noch dieses "Achtz", in der 64'er zur Fehlermeldung des Monats gekürt. Printfox spuckt diese Meldung immer dann aus, wenn ihm die Worte fehlen (z.B. beim Laden eines Grafikfiles im Texteditor). Einige User meinten, hier wäre eine aussagekräftigere Fehlermeldung angebracht gewesen. Doch diese Achtz ist wörtlich zu nehmen: Printfox nutzt den Speicher des C64 bis auf das letzte Byte aus, für eine längere Fehlermeldung ist wirklich kein Platz mehr!

Achtz!

APPARATE



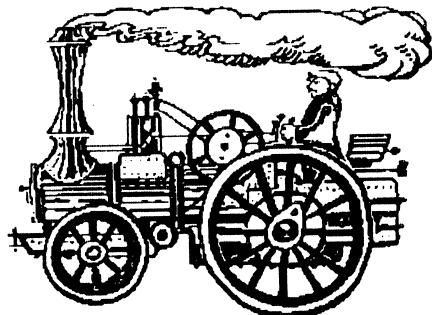
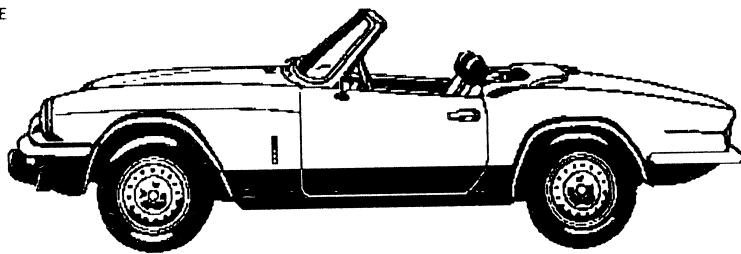
AUTOS



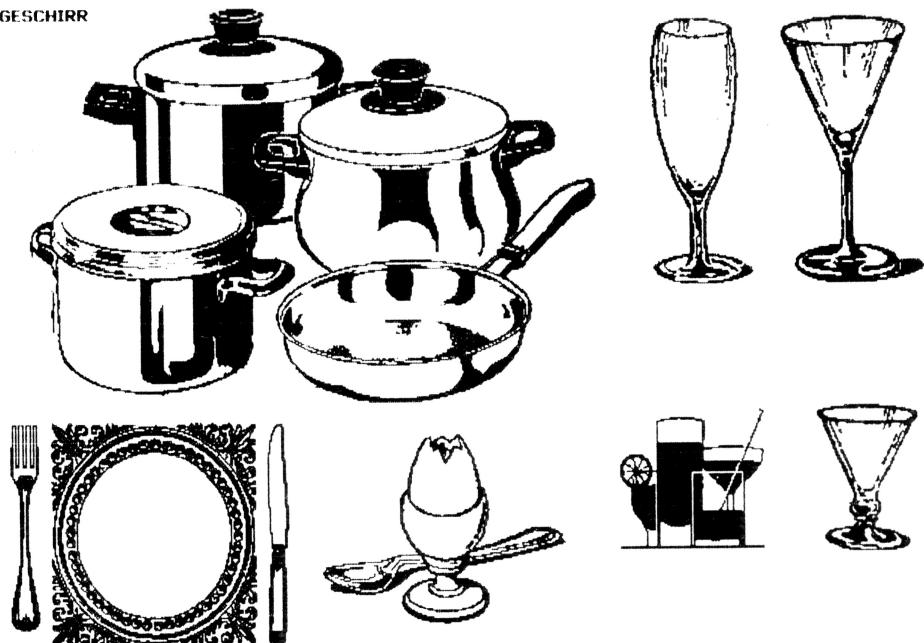
DSCHUNGEL



FAHRZEUGE



GESCHIRR



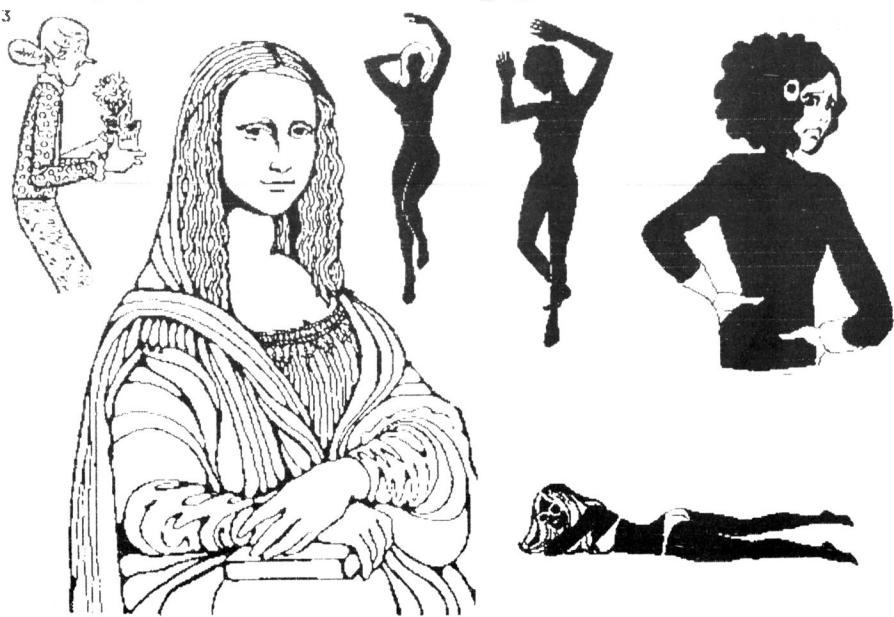
GIRLS!



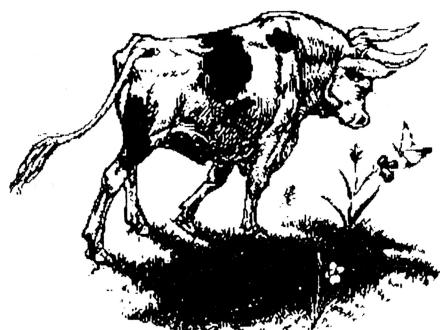
GIRLS2



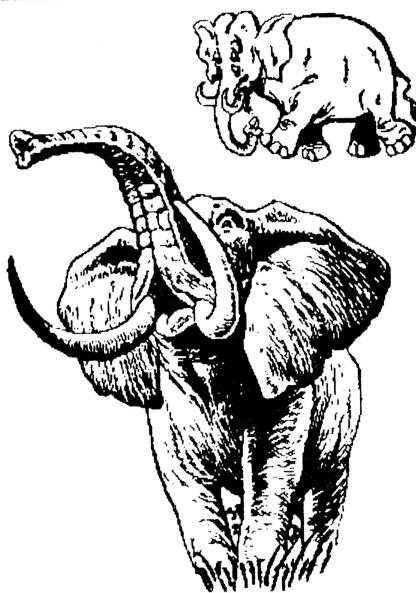
GIRLS3



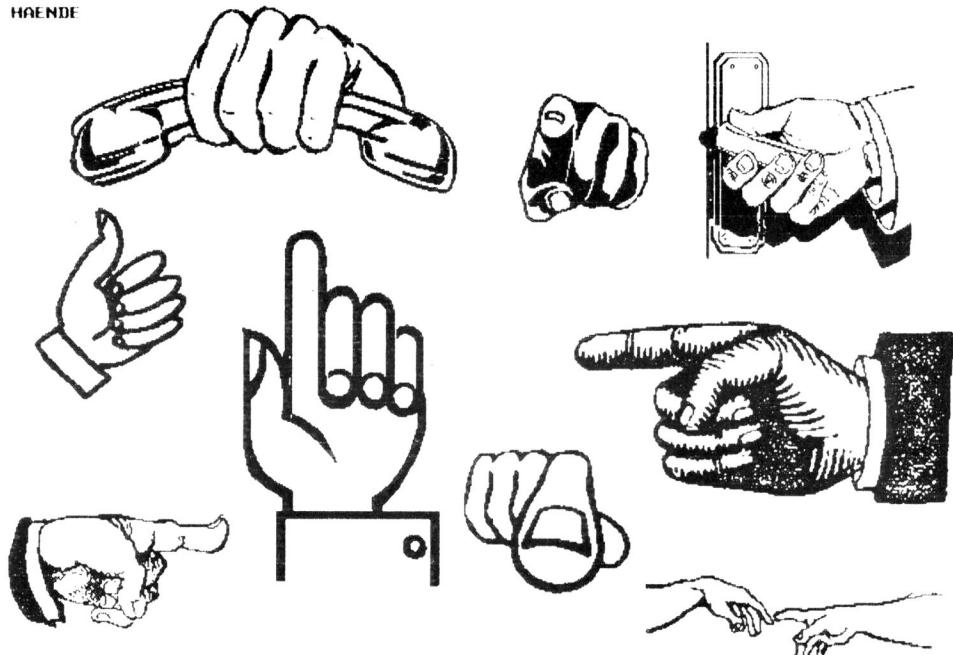
GROSSWILD1



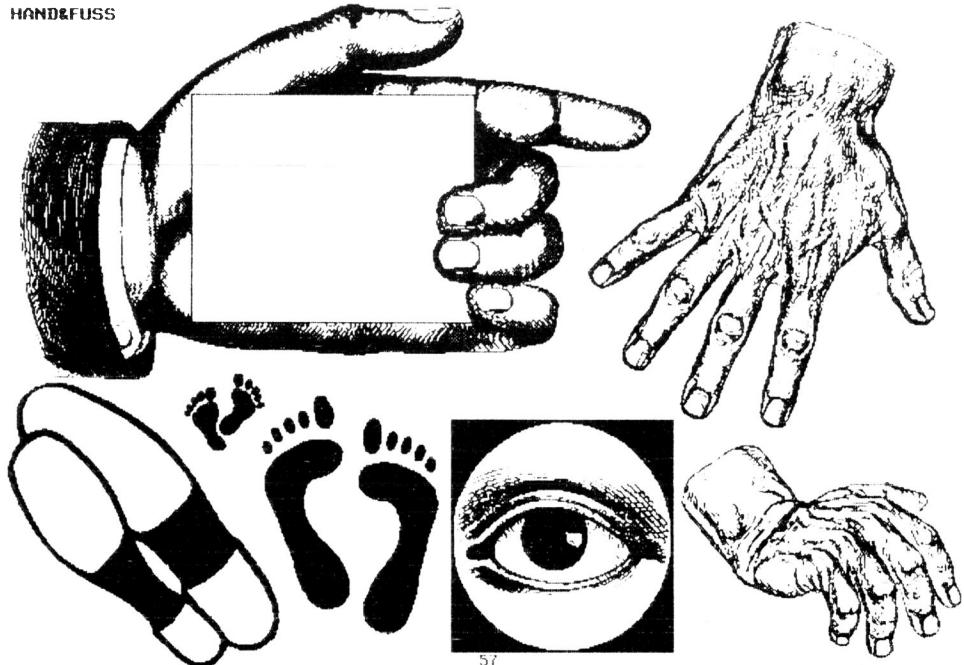
GROSSWILD2



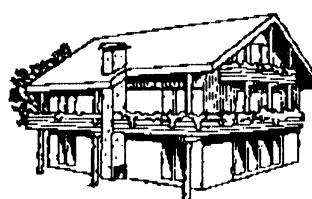
HAENDE



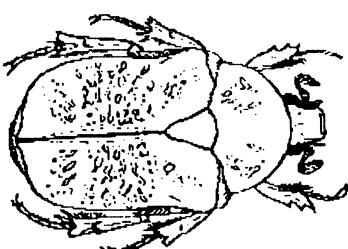
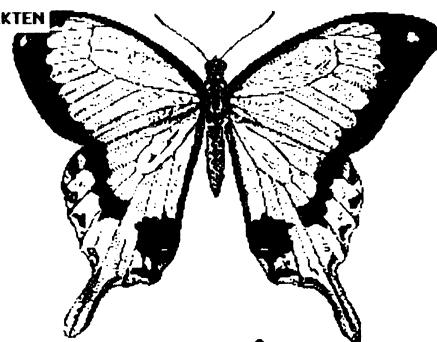
HAND&FUSS



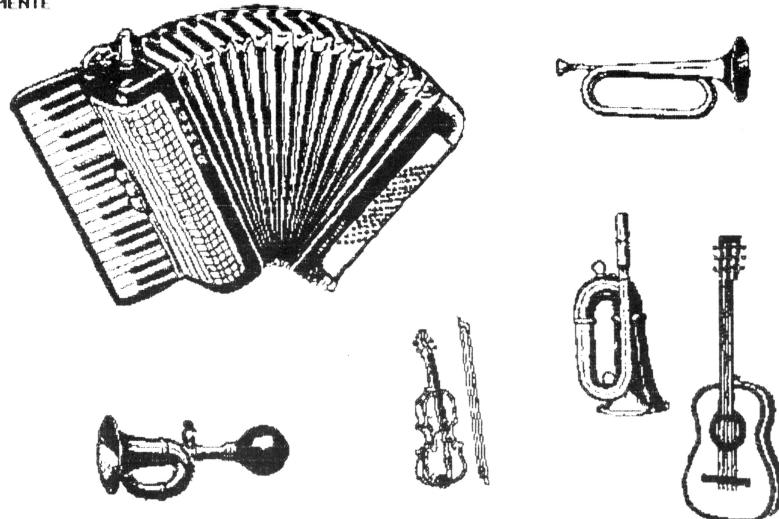
HÄUSER



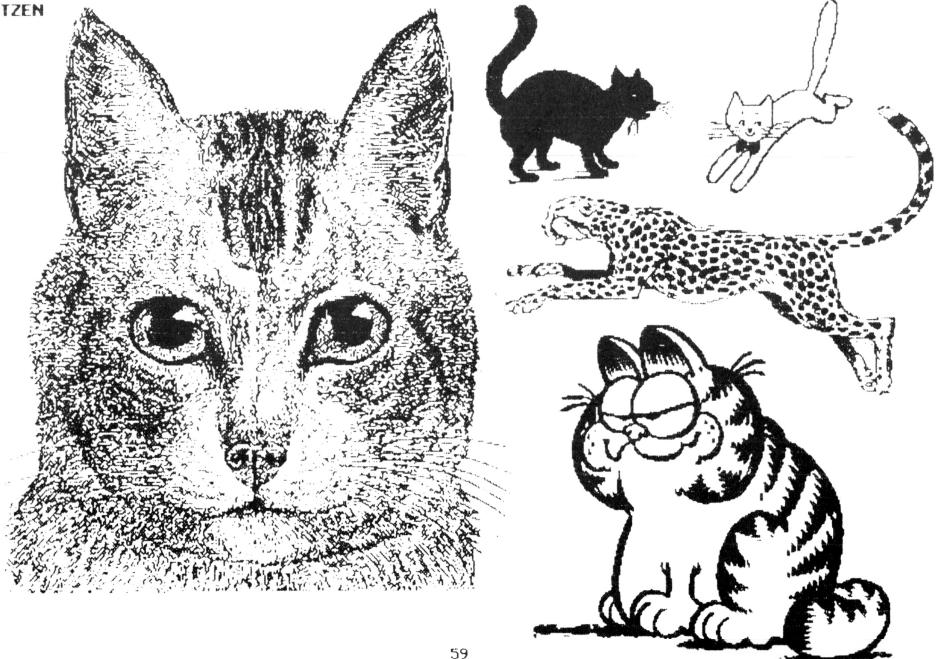
INSEKTEN



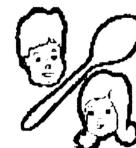
INSTRUMENTE



KATZEN



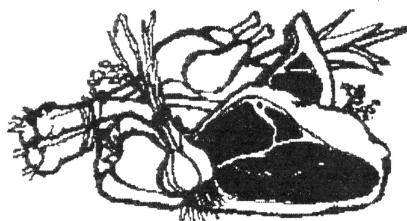
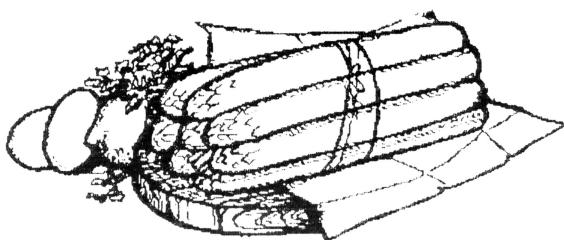
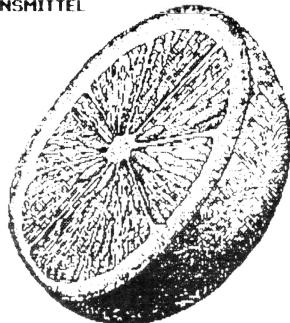
KINDER



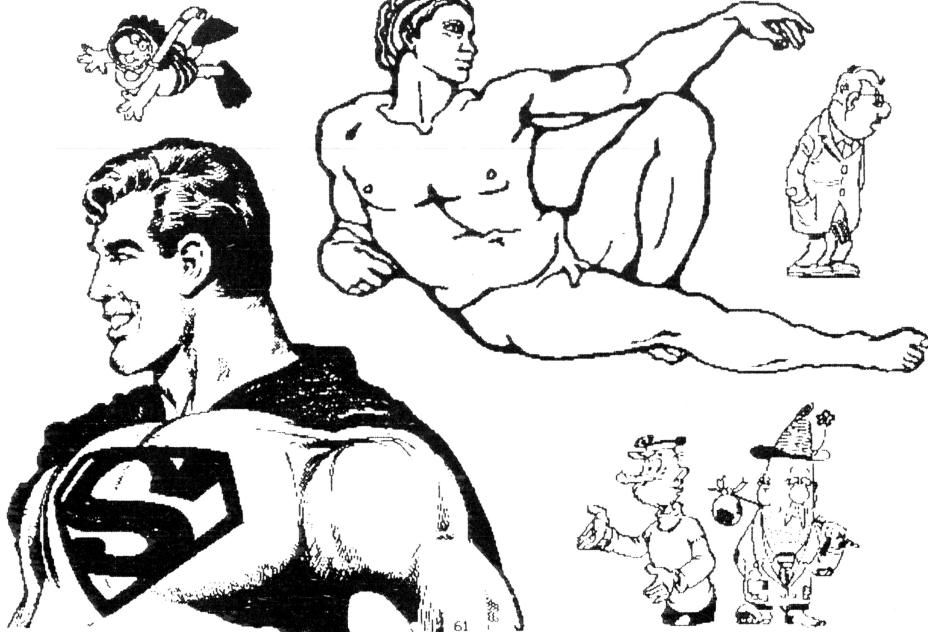
KOEPFE



LEBENSMITTEL



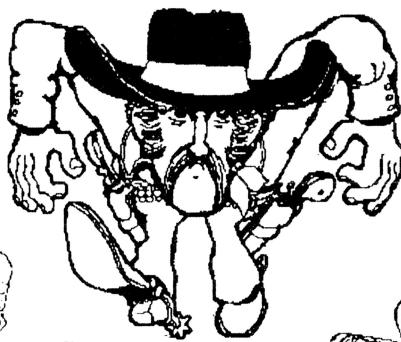
MAENNER



MAENNER2



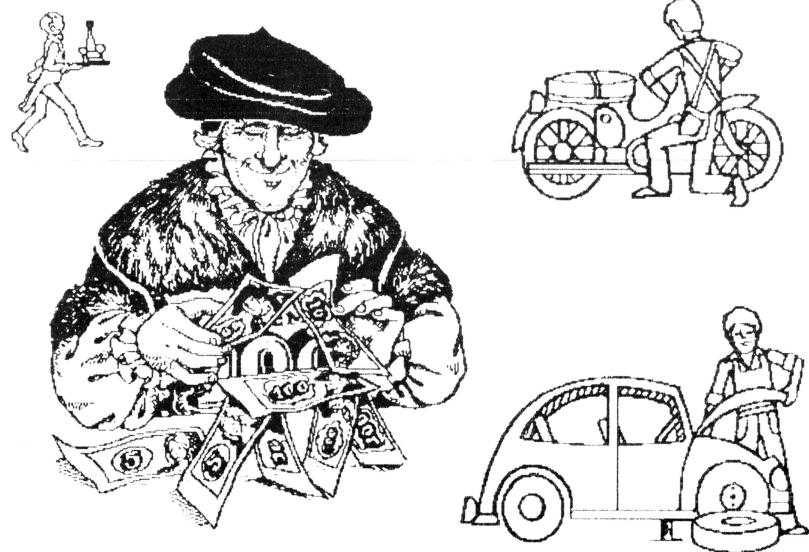
MAENNER3



MAENNER4



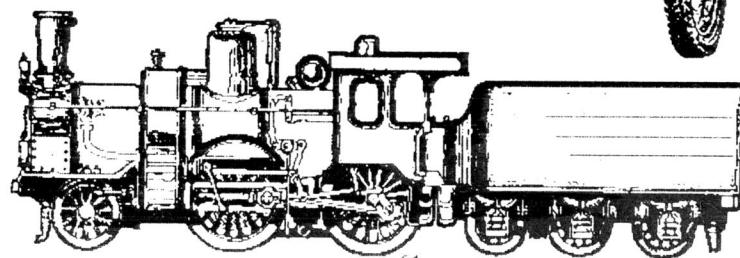
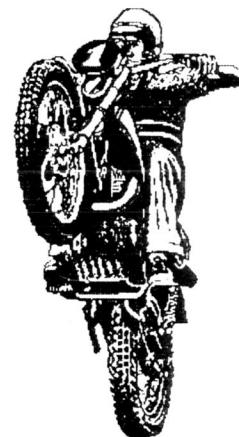
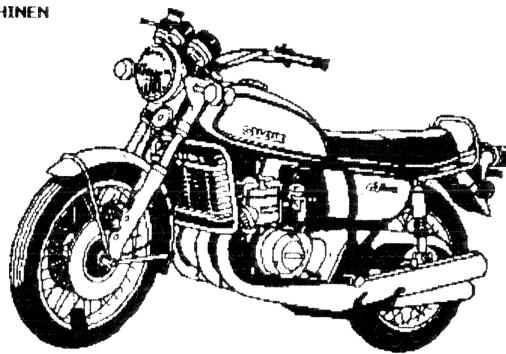
MAENNER5



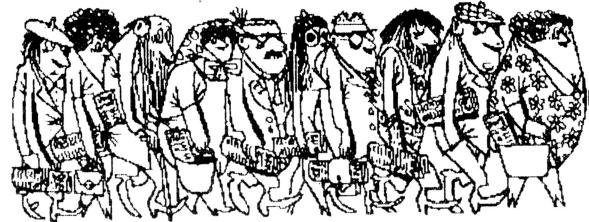
MAENNER6



MASCHINEN



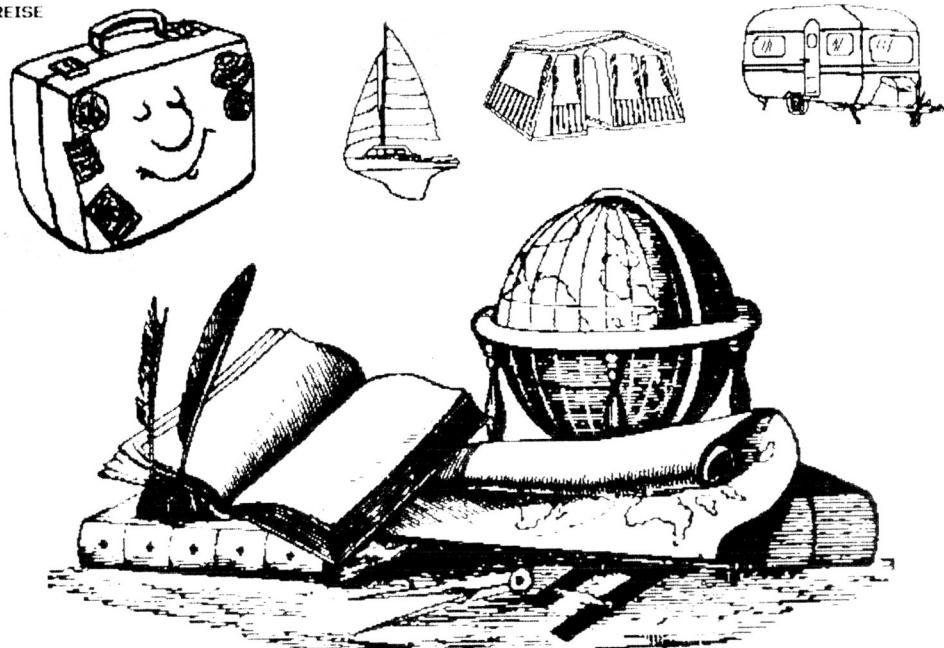
MENSCHEN



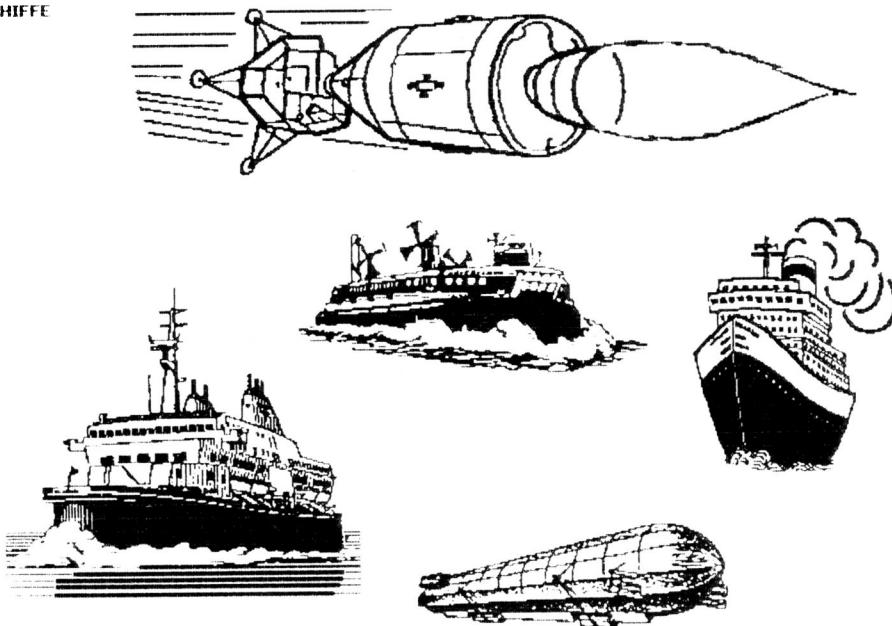
MUSIKER



REISE



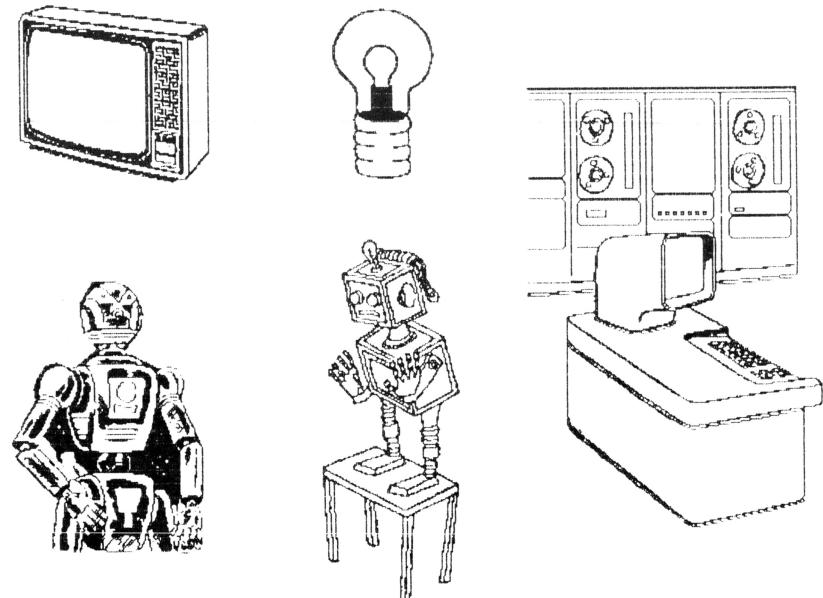
SCHIFFE



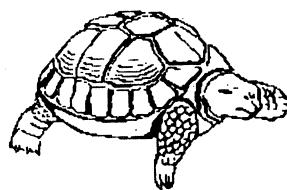
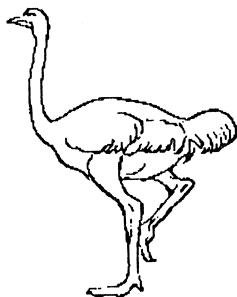
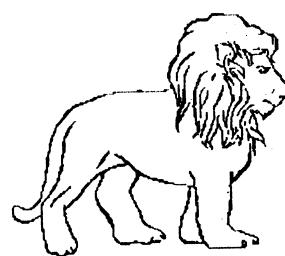
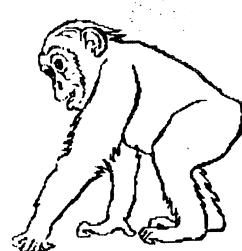
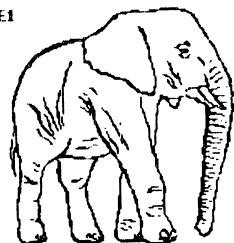
SCHUHE & HÜTE



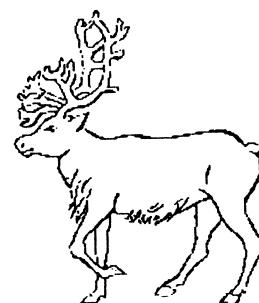
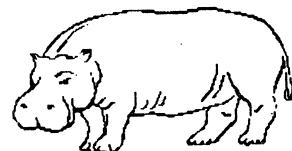
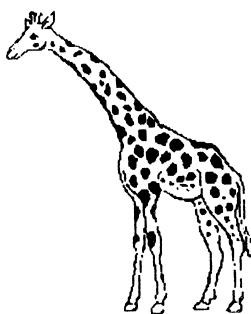
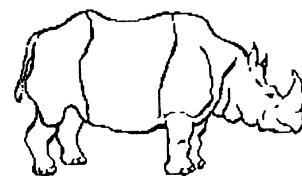
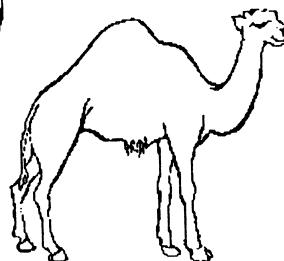
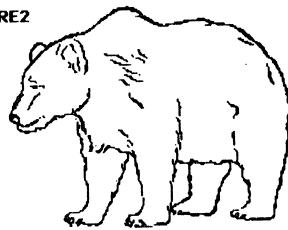
TECHNIK



TIERE1



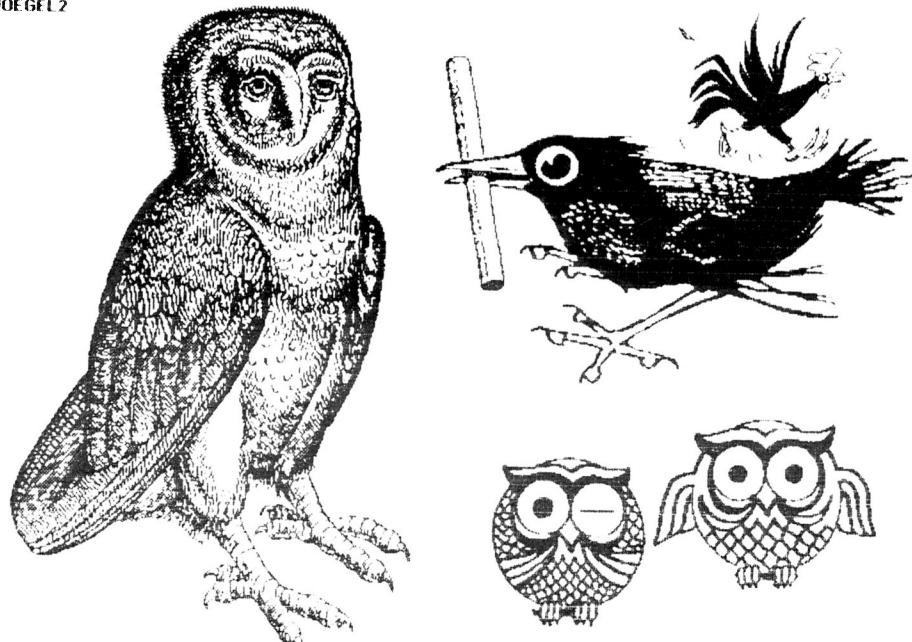
TIERE2



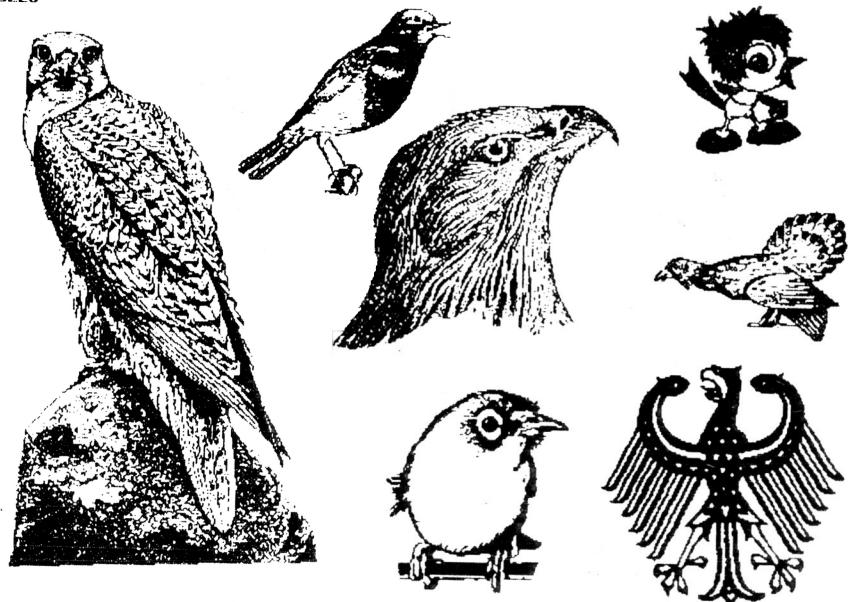
VOEGEL 1



VOEGEL 2



VOEGEL 3



WERKZEUG

