

Prozessvisualisierung JETTER VIADUKT

mit

Meldungseditor

JETTER GmbH Steuerungstechnik
Gräterstraße 2
D - 71642 Ludwigsburg
Telefon 07141/2550-0
Fax 07141/2550-25
Modem 07141/59834
Hotline 07141/2550-44

Dokumentation VIADUKT 2
Ausgabe 5
Januar 1994
Softwareversion V1.40

Die JETTER GmbH behält sich das Recht vor, Änderungen an ihren Produkten vorzunehmen, die der technischen Weiterentwicklung dienen. Diese Änderungen werden nicht notwendigerweise in jedem Einzelfall dokumentiert.

Diese Dokumentation und die darin enthaltenen Informationen wurden mit der gebotenen Sorgfalt zusammengestellt. Die JETTER GmbH übernimmt jedoch keine Gewähr für Druck- oder andere Fehler oder daraus entstandene Schäden.

Die in diesem Buch genannten Marken- und Produktnamen sind Warenzeichen oder eingetragene Warenzeichen der jeweiligen Titelhälter.

Inhaltsverzeichnis

1. Einleitung

1.1 Hardwareanforderungen

1.2 Anschluß an die PASE

1.2.1 Anschluß an die PASE-E-CPU, 15-pol.Sub.D

1.2.2 Anschluß an die PASE-E-CPU bzw. PASE-Mikro, 9-pol.Sub.D

1.2.3 Anschluß an PASE-J-CPU

1.3 Anschluß der Maus

1.4 Installation der Software

1.4.1 JETTER-Rechnereinschub

1.4.2 Softwarepaket VIADUKT

2. Maskenerstellung mit dem VIADUKT

2.1 Bedieneroberfläche

2.1.1 Menü "Datei"

2.1.2 Menü "Einstellungen"

2.1.3 Menü "Zeichnen"

2.1.4 Menü "Hilfe"

2.1.5 Menü "Auto"

2.1.6 Rechter Rand: Befehlsauswahl, Ortho, Fang

2.1.7 Unterer Rand: Befehlsanzeige, Texteingabe

2.1.8 Linker Rand: Farb-, Font-, Linientypauswahl

2.1.9 Zeichenoberfläche

2.1.10 Eingabe über Tastatur

2.2 Das Erzeugen von Objekten

2.2.1 Geometrische Grundobjekte

2.2.1.1 Linie

2.2.1.2 Rechteck

2.2.1.3 Balken

2.2.1.4 Kreis

2.2.1.5 Bogen

2.2.1.6 Ellipse

2.2.1.7 Text

2.2.1.8 Füllen

2.2.2 Dynamische Objekte

2.2.2.1 Zahl

2.2.2.2 Bargraph

2.2.2.3 Textvariable

2.2.2.4 Dynamik

2.3 Verändern von Objekten (Editierbefehle)

2.3.1 Objektauswahl

- 2.3.2 Schieben
- 2.3.3 Kopieren
- 2.3.4 Ändern
- 2.3.5 Löschen
- 2.3.6 Änd. Dyn. (Ändern Dynamik)
- 2.3.7 Letztes
- 2.3.8 Ausw.End (Auswahlende)
- 2.4 Erstellen von Masken
 - 2.4.1 Vorgabewerte (Defaulteinstellungen)
 - 2.4.2 Beispiel 1: Meine erste Maske
 - 2.4.3 Beispiel 2: Satzeditor
- 2.5 Aufrufparameter
 - 2.5.1 Updates
- 2.6 Funktionstasten
 - 2.6.1 Grundfunktionen
 - 2.6.2 Menüfunktion
 - 2.6.2.1 Reine Beschriftungsfelder
 - 2.6.2.2 Menüfeld mit Maskenaufruf
- 3. Automatikbetrieb**
 - 3.1 Maskenanzeige
 - 3.2 Funktionstasten
 - 3.3 Zahleneingabe
 - 3.3.1 Cursor-Logik
 - 3.4 Monitorfunktion
- 4. Reservierte PASE-Registerbereiche**
- 5. Reservierte PASE-Merkerbereiche**
- 6. Datentransferfunktion (nicht VIADUKT lite)**
- 7. DOS-Funktionen (nicht VIADUKT lite)**
- 8. Eingabe von ZAHL- und TEXTVARIABLE-Objekten mit der Maus (Automatikbetrieb)**
- 9. PCX-Dateien**
- 10. Laufzeitfehlermeldungen in beliebigen Fremdsprachen**
- 11. Funktionserweiterung durch TSR-Programme (nicht VIADUKT lite)**

12. Spezifikation des Graphikadapters

Stichwortverzeichnis

1. Einleitung

Das Visualisierungssystem **VIADUKT** (**V**ideo- **A**nzeige-, **D**ialog- **U**nd **K**ontroll- **T**erminal) ist als intelligente Bedienerschnittstelle für die Zusammenarbeit mit einer PASE-Steuerung konzipiert. Mit dem VIADUKT lassen sich folgende Anwendungen realisieren:

- o Bedienerführung im Klartext
- o Menütechnik
- o Fehlerdiagnose
- o komfortable Zahlenwert Ein-/Ausgabe
- o Grafische Prozessdarstellung
- o Fehlermeldungen im Klartext oder grafisch

In weiten Bereichen der Automatisierungstechnik sind visuelle Darstellungen von Prozessen und Vorgängen sehr hilfreich. Sie erleichtern den Überblick über komplexe Anlagen und verkettete Vorgänge. Mit dem **JETTER VIADUKT** ist es möglich, die Daten steuerungstechnischer Prozesse in Echtzeit darzustellen und zu verändern. Die Anzeige kann dabei in der jeweiligen physikalischen Einheit erfolgen, also z.B. der Füllstand eines Behälters in Prozent oder Liter, die Geschwindigkeit einer Bewegung in m/s oder die Zielposition einer Achspositionierung in Millimeter. **JETTER VIADUKT** übernimmt die Umrechnung von steuerungsinterner Darstellung in physikalische Einheit und zurück.

Der Zustand von binären Signalen wie Eingängen, Ausgängen und Merkern läßt sich durch einen Farbumschlag von dargestellten Objekten anzeigen. So kann z.B. der Schaltzustand eines Ventils (Ausgang) die Farbe des daran angeschlossenen (und grafisch dargestellten) Rohres beeinflussen.

Das Programm VIADUKT ist lauffähig auf IBM-AT oder 100% kompatiblen Personal-Computern. Es wird eine EGA- oder VGA-Grafikkarte benötigt.

Zur Erstellung von Masken ist eine Maus erforderlich.

JETTER bietet das VIADUKT in zwei Versionen an: Als reines Programmpaket oder als Kombination aus einem PC im 19-Zoll-Einschub mit installiertem Programm.

Das Programm VIADUKT ist als Demoverision auf Diskette vorhanden, mit der alle Eigenschaften des Programms getestet werden können.

VIADUKT lite ist ein VIADUKT-Derivat mit folgenden Einschränkungen:

- o kein Meldungseditor (Schalter /W /L /H)
- o kein Datentransfer (Schalter /D)
- o keine DOS-Funktionen (Schalter /E)
- o keine Übertragung von Zeit und Datum (Schalter /C)
- o keine TSR-Unterstützung (Schalter /U /V)

1.1 Hardwareanforderungen

Das VIADKUT-Programm stellt folgende Anforderungen an die Hardware:

- IBM AT, PS/2 oder kompatibler Rechner (z.B. 286, 386SX, 386, 486).
- EGA oder VGA Grafikkarte.
- Serielle oder MS kompatible Maus.
- Eine serielle Schnittstelle COM1 für die Verbindung zur Steuerung PASE.
- Eine parallele Schnittstelle (Druckerschnittstelle) zum Anschluß des Kopierschutzes ("Hardware-Lock").

Die von JETTER angebotene Hardware besteht aus einem AT im 19-Zoll Format mit folgenden Daten:

- IBM kompatibler Rechner mit einem 80386SX Prozessor der mit 33 MHz getaktet ist.
- 2 MB Ram.
- 170 Megabyte Festplatte.
- 3 1/2 Zoll Diskettenlaufwerk mit 1.44 MB.
- VGA Grafikkarte mit Bildschirm:
 - o monochrome LC-Display
 - o color TFT-LC-Display
- 2 serielle und eine parallele Schnittstelle.
- Fronttastatur mit 14 Funktionstasten, Numerik-Feld und Cursorblock. Anschluß für AT-Tastatur mit 102 Tasten.

Zusätzlich sind noch das für den Anschluß an die Steuerung PASE benötigte Kabel und eine Maus erhältlich.

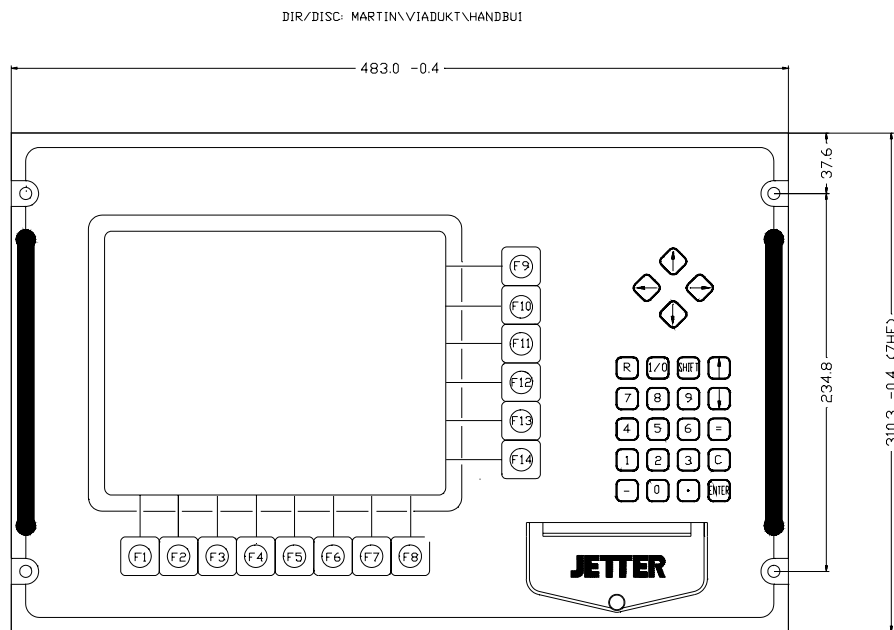


Bild 1 Vorderansicht

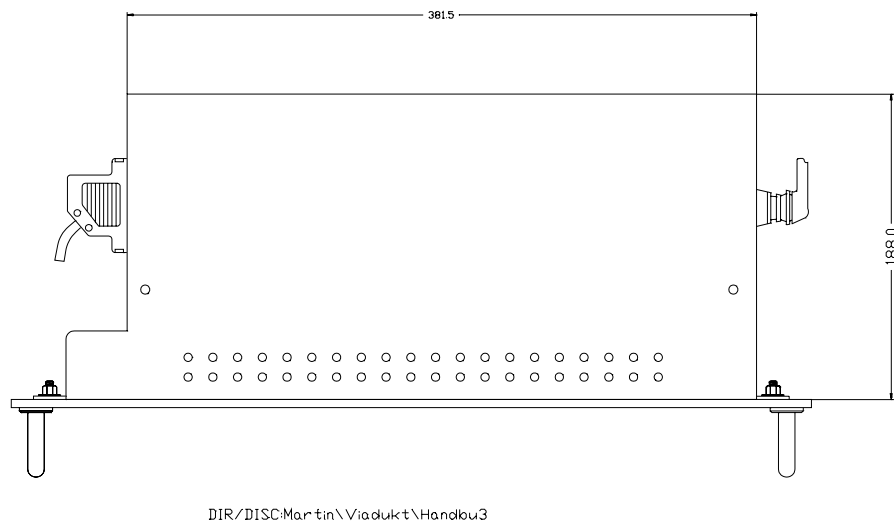


Bild 2 Draufsicht

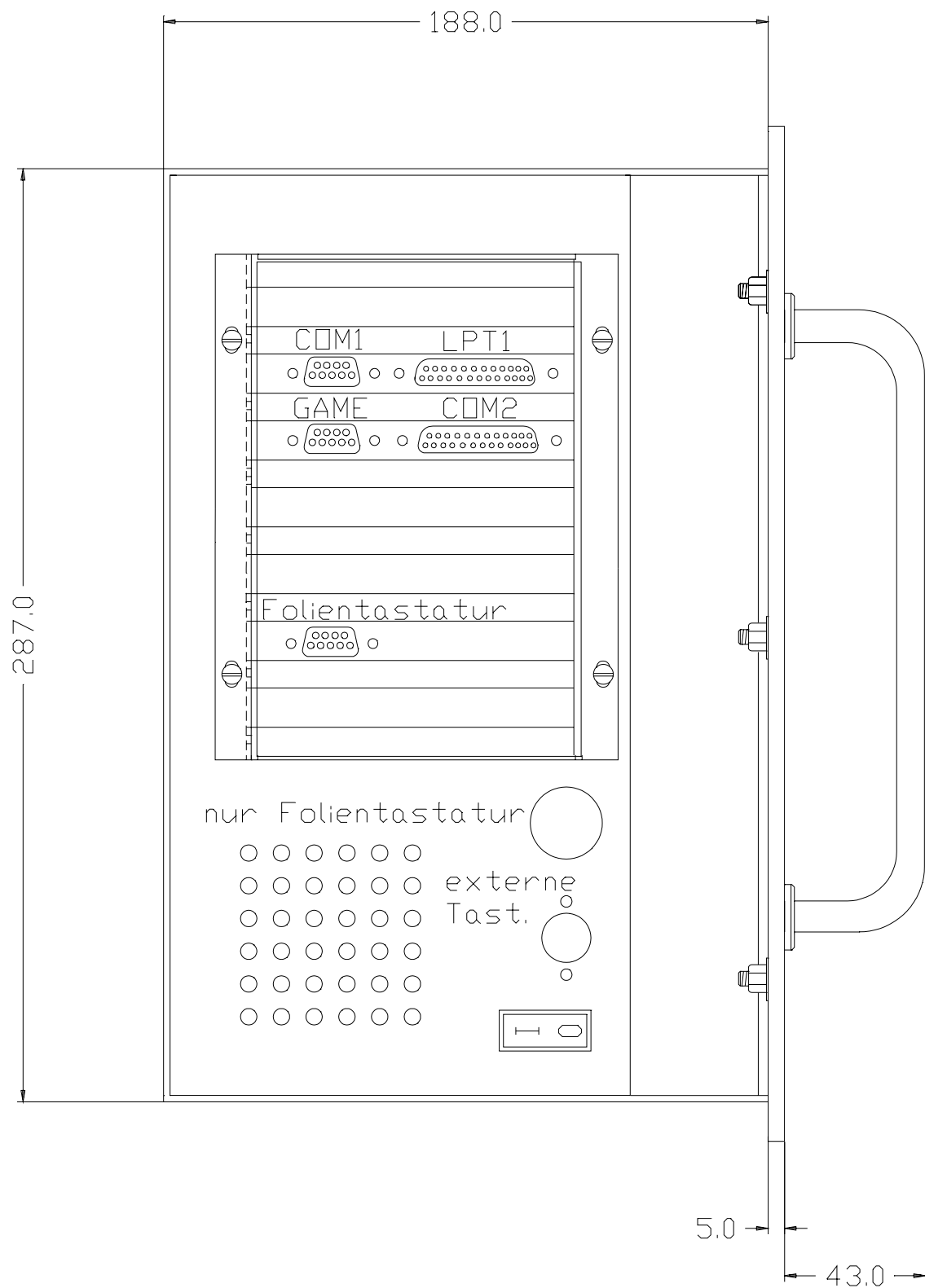


Bild 3 Seitenansicht (Steckerbelegung)

1.2 Anschluß an die PASE

Die Steuerung PASE wird an die serielle Schnittstelle COM1 des Rechners angeschlossen. Auf der PC-Seite ist dies normalerweise ein 9-poliger Sub-D Stiftstecker.

Auf der Steuerungsseite ist die Anschlußart vom verwendeten Steuerungstyp abhängig.

Bei einer PASE-E gibt es zwei Möglichkeiten, ein VIADUKT anzuschließen:

1.2.1 Anschluß an der 15-pol. Sub-D-Buchse der PASE-E-CPU:

Da an die PASE normalerweise nicht gleichzeitig ein VIADUKT **und** ein Anzeigemodul angeschlossen wird, wird die 15-polige Sub-D-Buchse der PASE-E-CPU-Karte zum Anschluß des VIADUKT genutzt. Um diese Verbindung herzustellen, ist folgendes Kabel erforderlich:

15-polige SUB-D Stiftleiste
Kabelseite: PASE-E / CPU Karte

9-polige SUB-D Buchsen-
leiste, Kabelseite: PC-AT (VIADUKT)

PIN	PIN
2-----	2
3-----	3
7-----	5
	+-----7
	+-----8
	+-----6
	+-----1
	+-----4

An die 9-polige Sub-D-Buchse der PASE-E-CPU-Karte kann das Kabel zum PASE-E-Programmiergerät angeschlossen werden. In dieser Konfiguration können also gleichzeitig zwei PCs an die PASE angeschlossen sein.

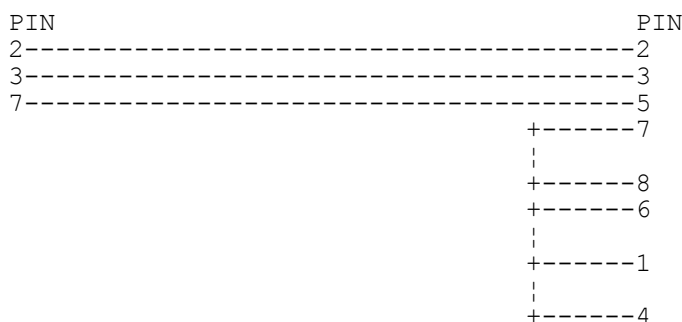
HINWEIS: Auf der CPU-Karte muß in diesem Fall eine Steckbrücke eingesetzt werden. Die Steckbrücke befindet sich auf der CPU-Karte "hinter" der 9-poligen Sub-D-Buchse.

1.2.2 Anschluß an der 9-pol. Sub-D-Buchse der PASE-E-CPU beziehungsweise der PASE-Mikro:

Unter Umständen ist es praktisch, auf ein und demselben PC sowohl mit SYMPAS als auch mit dem VIADUKT-Programm zu arbeiten. Dies ist möglich unter Verwendung des Standard-Programmierskabels der PASE-E:

9-polige SUB-D Stiftleiste
Kabelseite: PASE-E / CPU Karte

9-polige SUB-D Buchsen-
leiste, Kabelseite: PC-AT (VIADUKT)



An die 15-polige Sub-D-Buchse der PASE-E-CPU-Karte kann in diesem Fall zusätzlich ein Anzeigemodul angeschlossen werden.

1.2.3 Anschluß an PASE-J-CPU:

Der Anschluß an die PASE-J ist erst ab der **PASE-J-Betriebssystem-Version 2.0** möglich. Hierzu ist das Kabel vom Typ **J-PK** erforderlich, welches an der Programmierschnittstelle der PASE-J anzuschließen ist.

1.3 Anschluß der Maus

Zur Erstellung der Masken am VIADUKT ist eine Maus erforderlich. Im "Automatikbetrieb", d.h. nach Fertigstellung aller Masken ist keine Maus zum Betrieb des VIADUKT notwendig.

Da VIADUKT die Schnittstelle COM1 zur Kommunikation mit der PASE benötigt, sollte eine Maus an der Schnittstelle COM2 betrieben werden, um häufiges Umstecken zu vermeiden.

Um die Maus zu aktivieren, muß ein Zusatzprogramm ("Maus-Treiber") aufgerufen werden, bevor das Programm VIADUKT gestartet wird. Dieses Programm wird vom Mauslieferanten mitgeliefert. Es hat meistens den Namen "MOUSE.COM". Es ist sinnvoll, dieses Programm innerhalb der AUTOEXEC.BAT Datei aufzurufen. Dadurch ist die Maus bei jedem Neustart des Rechners aktiv. Die entsprechende Aufrufsyntax ist der Mausbeschreibung zu entnehmen.

1.4 Installation der Software

1.4.1 VIADUKT-Rechnereinschub:

Auf dem VIADUKT Rechner ist die Software bereits installiert. Die Dateien befinden sich im Unterverzeichnis C:\VIADUKT. Die Datei AUTOEXEC.BAT enthält u.a. die folgenden Zeilen, die einen Wechsel in dieses Verzeichnis veranlassen und das Programm VIADUKT im Automatikmodus starten:

```
...  
CD C:\VIADUKT  
VIAGR /A
```

1.4.2 Softwarepaket VIADUKT:

Zum Lieferumfang des Softwarepakets **VIADUKT** gehören:

- 1 Diskette mit den Programmdateien
- 1 Kopierschutzadapter ("Hardware-Lock")
- 1 Programmbeschreibung.

Die Installation beschränkt sich auf das Kopieren der Dateien auf die Festplatte und das Anschließen des Hardware-Locks an die parallele Druckerschnittstelle. Es kann wahlweise LPT1 bis LPT3 benutzt werden. Geben Sie zur Installation folgende Zeile ein:

```
A:Install <ENTER>
```

ACHTUNG: Die parallele Schnittstelle hat am Rechner eine 25-polige Sub-D-**Buchse** und ist mit LPT.. bezeichnet. Stecken Sie den Hardware-Lock nur in diese Schnittstelle. Wenn Sie nicht sicher sind, welches die richtige Buchse ist, dann fragen Sie jemanden, der Ihnen kompetent erscheint. Stecken des Hardware-Locks auf die falsche Buchse kann zu dessen Zerstörung führen.

Auf dem Hardware-Lock ist die Steckrichtung mit Pfeilen gekennzeichnet. Sie müssen die **Stiftseite** an den PC anschließen.

Ist das Programm auf der Festplatte installiert, kann es durch Eingabe des Programmnamens (VIAGR oder VLITEGR) aufgerufen werden. **VIADUKT** erkennt selbstständig welche Grafikkarte (EGA oder VGA) der Rechner installiert hat und wählt die bestmögliche Auflösung. Nun erscheint das Eingabefenster von **VIADUKT** an dem die Masken erstellt werden.

In der Mitte des Bildschirms erscheint ein Fadenkreuz, das sich mithilfe der Maus verschieben lässt. Sie sollten jetzt prüfen, ob Ihre Maus funktioniert. Ist dies nicht der Fall, kann das folgende Ursachen haben:

- Maus steckt an der falschen Schnittstelle.
- Der Maus-Treiber ist falsch installiert.
- Die Eintragungen in den Konfigurationsdateien des Rechners wurden nicht oder fehlerhaft vorgenommen (konsultieren Sie Ihre Mausbeschreibung).

Funktioniert die Maus, kann das Fadenkreuz bewegt werden. Außerhalb der Zeichenfläche wird das Fadenkreuz unsichtbar und an der Mausposition erscheint ein Pfeil, mit dem eine der Menüfunktionen ausgewählt werden kann.

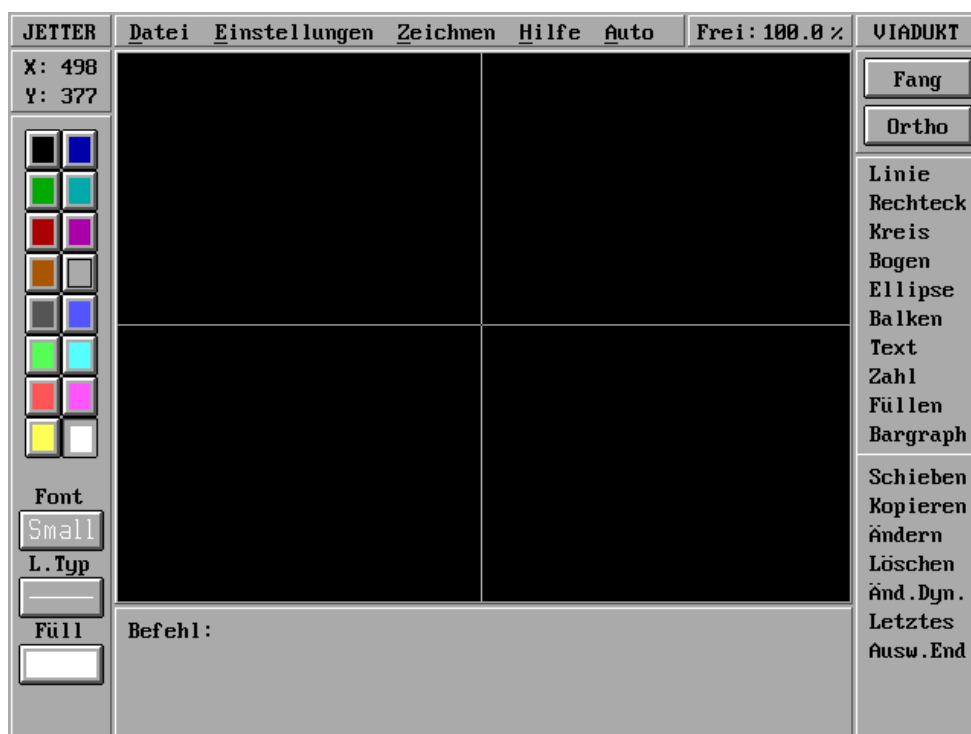
2. Maskenerstellung mit dem VIADUKT

2.1 Die Bedieneroberfläche

Der Begriff "Bedieneroberfläche" bedeutet in diesem Zusammenhang, wie das Programm VIADUKT dem Programmierer von Masken gegenübertritt.

Die Bedieneroberfläche besteht aus den folgenden Bildschirmbereichen:

- Oberer Bildschirmrand (Pull-Down-Menüs)
- Rechte Bildschirmseite (Befehlsauswahl)
- Unterer Bildschirmrand (Befehlseingabe und Befehlsanzeige)
- Linker Bildschirmrand (Farb-, Font-, und Linienauswahl)
- Mitte (Zeichenfläche)



Vorausgeschickt seien einige Informationen über grundsätzliche Editions Hilfsmittel.

Fang Mit dieser Funktion bewegt sich das Fadenkreuz in einem bestimmten Raster, dessen Schrittweite in der Auswahlzeile "Fang" im Pull-Down-Menü "Einstellungen" festgelegt werden kann.

Ortho Diese Funktion gestattet nur orthogonale funktionsrelevante Bewegungen des Fadenkreuzes. Es können zum Beispiel nur Linien gezeichnet werden, die rechtwinklig zueinander stehen.

Beim Selektieren von Objekten im Zeicheneditor (Schieben etc.) kann alternativ zum Cursor ein Fenster zum Auswählen ganzer Gruppen geöffnet werden. Zwei Fensterarten werden unterschieden:

<F> es werden nur Objekte selektiert, deren Zeichengrenzen innerhalb des Fensters liegen.

<K> es werden alle Objekte selektiert, die vom Fenster teilweise oder ganz umschlossen sind.

Durch gleichzeitiges Drücken des SHIFT-Taste kann mit Hilfe des Cursors deselektiert werden.

Mit **ALT-F5** kann der DOS-Bildschirm betrachtet werden, mit einer beliebigen Taste wird in das VIADUKT zurückgekehrt.

F7 Koordinatenanzeige ein- bzw. ausschalten.

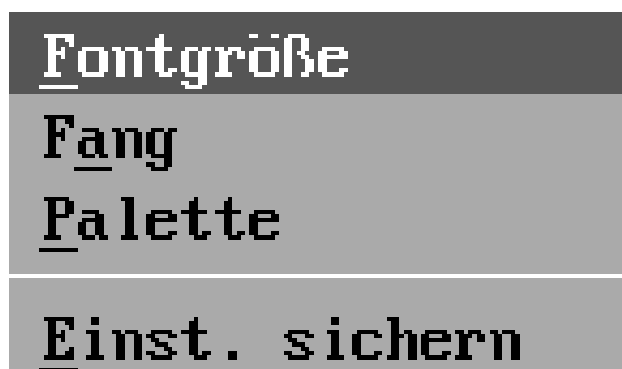
2.1.1 Menü "Datei"

Neu	
Maske laden	F3
Sichern	F2
Sichern als	
Bild laden	
DOS shell	
Exit Viadukt	Alt-X
Info	

- Neu** Die vorhandene Maske wird gelöscht, eine neue kann begonnen oder geladen werden. **VIADUKT** fragt ob die alte Maske vor dem Löschen gespeichert werden soll.
- Maske laden** Datei (Maske) wird geladen und ist editierbar. Durch das "Hinzuladen" weiterer Dateien läßt sich ein Bild aus Einzelbildern zusammenstellen (z.B. Bibliothekselemente).
- Sichern** Hiermit wird eine bereits geladene Datei unter ihrem ursprünglichen Namen im aktuellen Zustand abgespeichert (Die alte Version der Datei wird dabei überschrieben). Masken, die von der PASE aufgerufen werden sollen, müssen **numerische** Namen erhalten ("1" bis "8000000"). Andere Masken (z.B. Bibliothekselemente oder Logo's), die sie als Einzelemente speichern und später zu Masken zusammenstellen wollen, können auch andere Namen mit bis zu acht Zeichen erhalten, z.B. "VENTIL" oder "LOGO".
- Sichern als** Eine Datei wird unter einem Namen abgespeichert, der vor dem Abspeichern erfragt wird.

- Bild laden** Datei (Maske) wird geladen, kann jedoch nicht verändert werden. Diese Funktion kann dazu benutzt werden, das Zusammenpassen mehrerer Masken zu testen (überlappende Bereiche usw.) oder es kann ein "Hintergrund" zu einer editierbaren Maske angezeigt werden. Es können bis zu 10 Masken gleichzeitig geladen werden. Beim Speichern wird nur die editierbare Maske gespeichert, die mit 'Datei laden' geladen wurde.
- DOS-Shell** Aufruf des Befehlsinterpreters COMMAND.COM des DOS-Betriebssystems, Bildschirminhalt von **VIADUKT** bleibt erhalten, zurück durch "EXIT".
- Exit VIADUKT** Programm wird verlassen, der Bildschirminhalt muß vorher gesichert werden.
- Info** Dieses Fenster stellt folgende Informationen zur Verfügung:
- o Versionsnummer
 - o Maskendatei
 - o Aufrufparameter (von VIAGR.EXE)

2.1.2 Menü "Einstellungen"



- Fontgröße** Wahl der Fontgröße für Texte und Zahlen. Beim Fonttyp Normal wird immer auf ganzzahlige Werte gerundet (1.2 wird z.B. auf 1.0 gerundet). Bei den anderen Fonttypen sind auch die Zwischengrößen möglich.
- Fang** Einstellen der Schrittweite des Fangrasters. Wenn die FANG - Funktion aktiv ist, kann das Fadenkreuz nur auf einem Raster bewegt werden, dessen Schrittweite hier anzugeben ist. Diese Funktion ist z.B. hilfreich bei der Konstruktion von tabellarisch angeordneten Eingabefeldern.
- Palette** Mit Hilfe dieser Funktion kann der Benutzer anwendungsspezifische Farbpaletten erstellen und unter dem Namen der VIADUKT-Standardpalette VIADUKT.COL abspeichern.
- Einstellungen sichern** Mit dieser Funktion wird der momentane Stand der Einstellungen in einer Datei mit dem Namen VIADUKT.CFG gespeichert.
Wenn beim Programmstart von VIADUKT die Datei

VIADUKT.CFG gefunden wird, dann wird sie gelesen und die Einstellungen wieder auf diesen Stand gebracht.

Es werden in VIADUKT.CFG folgende Daten gespeichert:

Aktuelle Farbe, Linientyp, Linienbreite, Füllmuster, Fonttyp, Fontbreite, Fonthöhe, Fang EIN/AUS, Fangraster, Ortho EIN/AUS, Dateiname der aktuellen Maske, Fadenkreuz beziehungsweise Eingaberahmenfarbe.

2.1.3 Menü "Zeichnen"



- Bild löschen** Löschen des Editierfensters. Das neue, leere Fenster nimmt die aktuelle Farbe als Hintergrundfarbe (Auswahl: Aktuelle Farbe oder Abbruch). Alten Bildschirminhalt vorher speichern wenn er noch benötigt wird !
BILD LÖSCHEN ist ein Zeichenbefehl, d.h. wenn eine Maske im Automatikmodus aufgerufen wird, die einen BILD LÖSCHEN Zeichenbefehl enthält, wird der Bildschirm gelöscht.
Wird eine Maskendatei in den VIADUKT-Editor geladen, die mit BILD LÖSCHEN beginnt, so wird der Name der neu hinzugeladenen Datei der aktuelle Dateiname der Maske.
- Einfrieren** Bei EINFRIEREN handelt es sich um einen Zeichenbefehl. EINFRIEREN beendet die zyklische Auffrischung dynamischer Elemente, ohne diese vom Bildschirm zu entfernen. Um die eingefrorenen Elemente wieder aufzutauen, muß die entsprechende Maske erneut aufgerufen werden.

- Textvariable** Mit diesem Objekt können Texte in die Maske und damit in Register der PASE-Steuerung eingegeben werden. Eine ausführliche Beschreibung erfolgt in Kapitel 2.2.2.3.
- Menü** Mit diesem Befehl kann ein Beschriftungsfeld für eine der 14 am VIADUKT vorhandenen Funktionstasten erzeugt werden. Das Feld wird an der Stelle am Bildschirm gezeichnet, an der sich auf der Frontseite des VIADUKT die zugehörige Funktionstaste befindet. Mit der Editierfunktion SCHIEBEN kann ein Menüfeld jedoch anschließend auch an einen anderen Ort auf dem Bildschirm geschoben werden. Der Inhalt des Beschriftungsfelds ist ein zweizeiliger Text mit jeweils 8 Zeichen. Wahlweise kann auch definiert werden, daß durch Druck auf die Funktionstaste eine andere Maske aufgerufen wird.
- Knopf** Mit dieser Funktion können beliebige Buttons (Schalter, Taster) erstellt werden, welche im Automatikbetrieb mit der Maus bedient werden können. Nach dem Anklicken der Auswahlzeile wird mit dem Start- und dem Endpunkt die Größe des Buttons vorgegeben. Nachdem der Endpunkt festgelegt wurde erscheint ein Fenster, in welchem die Farbe des Beschriftungstextes angegeben werden kann. Die Farbe des Buttons selbst entspricht der aktuellen Zeichenfarbe. Jetzt öffnet sich ein Fenster zur Definition folgender Parameter:
- o **Merkernummer:** Dieser Merker wird durch den Button gesetzt (bei betätigtem Button ist der Status des entsprechenden Merkers 1).
 - o **Offset:** bei indirekter Adressierung des Merkers über ein Register wird dieser Offsetwert auf die Merkersummer addiert. Die Summe entspricht der eigentlichen Merkersummer, die durch den Button

gesetzt wird.

o **Typ:** zwei unterschiedliche Buttontypen sind möglich. Der Taster bleibt so lange aktiv als die Maustaste über dem Button gedrückt bleibt. Der Schalter ändert seinen Zustand bei jedem Mausklick und behält diesen bis zum nächsten Mausklick bei.

o **Text:** hier wird der Beschriftungstext für den Button eingetragen.

o **Maskennummer:** neben der Steuerung eines Merkers kann der Button auch eine weitere Maske aufrufen. Hier wird angegeben, welche Maske durch den Button aktiviert werden soll.

Datum-Zeit

Dieses Objekt ermöglicht es das aktuelle Datum und die Zeit in eine Maske zu integrieren. Das Darstellungsformat kann hierbei in weiten Grenzen vom Anwender vorgegeben werden.

Zunächst wird der Startpunkt angegeben ab welchem der Datums- respektive Zeittext erscheinen soll. Danach erscheint ein Fenster, in welchem beliebige Texte zusammen mit folgenden reservierten Befehlen angeordnet werden können:

Befehl:	wird ersetzt durch:
ap	am für Uhrzeiten von 0.00.00 bis 11.59.59 pm für Uhrzeiten von 0.00.00 bis 23.59.59
da	aktueller Tag von 1 bis 31
Da	aktueller Tag von 01 bis 31
date	automatische Generierung des Datums in landesüblicher Form
day	Tagesname in Kurzform (z.B. Mon)
dow	Tagesname in Langform (z.B. Montag)
ha	12-Stundenanzeige
ho	24-Stundenanzeige
mi	aktuelle Minute von 0 bis 59
Mi	aktuelle Minute von 00 bis 59
mo	aktueller Monat von 1 bis 12
Mo	aktueller Monat von 01 bis 12
mon	Monatsname in Kurzform (z.B. Jan)
Mon	Monatsname in Langform (z.B. Januar)
se	aktuelle Sekunde von 0 bis 59
Se	aktuelle Sekunde von 00 bis 59
time	automatische Generierung der Zeit in landesüblicher Form

ye	Jahreszahl 2-stellig (z.B. 93)
year	Jahreszahl 4-stellig (z.B. 1993)
oa	fügt das Ordinalzahlkürzel an die Zahlen 1 bis 31 an.

Beispiel:

Die Formatsdefinition

```
Heute ist dow, der da. Mon year Uhrzeit: ho:Mi:Se
```

würde am Freitag, dem 3. März 1993 um 7 Uhr 5 und 3 Sekunden zu folgender Anzeige führen

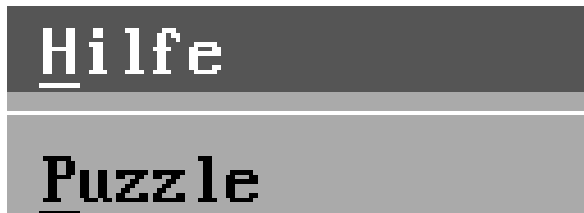
```
Heute ist Montag, der 3. März 1993 Uhrzeit: 7:05:03
```

Include Beliebige VIADUKT-Masken (Extension .VIA) beziehungsweise PCX-Dateien (Extension .PCX) können in eine Maske integriert werden.

Dynamik Alle darstellbaren Objekte können dynamisch ihre Farbe ändern, je nach logischem Zustand einer Variable in der Steuerung (z.B. ein Ausgang). Variable und Umschlagfarbe werden erfragt.

Neu zeichnen dieser Befehl baut den Bildschirm neu auf.

2.1.4 Menü "Hilfe"



Hilfe	Beenden durch Anklicken der linken oberen Ecke mit der linken Maustaste.
Puzzle	Spiel

2.1.5 Menü "Auto"

<u>V</u> ollansicht	F9
<u>A</u> utomatik	Ctrl-F9

Vollansicht Die erstellte Maske kann in Vollansicht betrachtet werden, zurück durch 'Ctrl + End'.

Automatik Betreiben von **VIADUKT** im Automatikmodus. Hierzu ist die Verbindung mit einer Steuerung über COM1 notwendig. Die aktuellen Werte von Zahlen, Bargraphen und Textvariablen werden in Echtzeit angezeigt. Über Eingabevariablen können Registerinhalte verändert werden.

2.1.6 Rechter Rand: Befehlsauswahl, Ortho, Fang

Am rechten Rand der Bedieneroberfläche sind die für das Zeichnen nötigen Befehle aufgelistet. Die einzelnen Befehle werden durch Anklicken mit der linken Maustaste aktiviert. Mit der rechten Maustaste kann ein Befehl abgebrochen werden.

Oben am rechten Rand sind zwei Buttons, die mit FANG und ORTHO beschriftet sind. Durch Anklicken mit der Maus kann ein Button aktiviert und durch nochmaliges Anklicken wieder deaktiviert werden.

Ist die Funktion **ORTHO** aktiviert, werden Linien nur horizontal oder vertikal gezeichnet.

Wenn die Funktion **FANG** aktiv ist, kann das Fadenkreuz nur auf einem Raster bewegt werden, dessen Schrittweite festgelegt werden kann (Siehe Menü 'Einstellungen/Fang' (s. 2.1.2)).

Unter den Feldern Ortho und Fang befinden sich die Auswahlfelder für die Zeichenfunktionen und für die Editierfunktionen.

Direkt am rechten Rand anwählbare Zeichenfunktionen sind: Linie, Rechteck, Kreis, Bogen, Ellipse, Balken, Text, Zahl, Füllen und Bargraph. Weitere Zeichenfunktionen befinden sich im Menü "ZEICHNEN", das über die Hauptmenüzeile am oberen Bildschirmrand erreicht werden kann.

Unter den Feldern für die Zeichenfunktionen befinden sich die Auswahlfelder für die Editierfunktionen: Schieben, Kopieren, Ändern, Löschen, Ändern Dynamik, Letztes und Auswahl-Ende.

Eine genaue Beschreibung der Zeichenfunktionen und der Editierfunktionen befindet sich weiter unten.

2.1.7 Unterer Rand: Befehlsanzeige, Texteingabe

Dies ist das Dialogfenster zwischen VIADUKT und dem Masken-Programmierer. Es erscheint als Eingabeaufforderung der Text

"Befehl:"

Alle Zeichen- und Editierbefehle, die an der rechten Bildschirmseite mit der Maus wählbar sind, können auch hier über die Tastatur eingegeben werden. Die einzige Ausnahme ist der Befehl "LETZTES".

Hier wird die Ausführung der aktuellen Befehle angezeigt. Außerdem wird auch der eingegebene Text angezeigt.

2.1.8 Linker Rand: Farb-, Font-, Linientypauswahl

Farben Am linken Rand der Bedieneroberfläche kann die gewünschte aktuelle Farbe ausgewählt werden. Es stehen 16 verschiedene Farben (Graustufen) zur Verfügung.

Font Nach dem Öffnen eines Fensters kann in diesem der gewünschte Font gewählt werden.

L.typ In einem Fenster kann der gewünschte Linientyp gewählt werden.

Füll Hier kann das Füllmuster zur Füllfunktion festgelegt werden.

Außerdem wird am linken oberen Rand die aktuelle Fadenkreuzposition

(als X- und Y-Position) dargestellt. Während der Eingabe eines Winkels wird die Anzeige automatisch umgestellt auf Anzeige von Radius und Winkel.

Die mitlaufende Fadenkreuzpositionsanzeige kann mit der Funktionstaste F7 ein- und ausgeschaltet werden.

2.1.9 Zeichenoberfläche

In der Mitte des Editierbildschirms ist die Zeichenfläche. Zeichnungen werden in einem "Weltkoordinatensystem" erzeugt, das eine Größe von 1001 x 751 Einheiten hat. Der Wertebereich der Koordinaten ist also:

x-Koordinate: 0 bis 1000

y-Koordinate: 0 bis 750

Die Position (0,0) ist die linke untere Ecke der Zeichenfläche.

Im Editiermodus des VIADUKT wird die gesamte Zeichenfläche abgebildet auf die Fläche innerhalb der 4 Menübereiche. Im Automatikbetrieb wird der gesamte Bildschirm durch die Zeichenfläche ausgefüllt.

Die Anpassung an die real existierenden Auflösungen der vorhandenen Grafikkarten (640 x 350 bei EGA oder 640 x 480 bei VGA) wird vom VIADUKT vorgenommen. Dies bedeutet, daß die Auflösung der später im Automatikmodus vorhandenen Grafikkarte bei der Erstellung der Masken nicht berücksichtigt werden muß. Dadurch lassen sich auch systemunabhängige Bildelement-Bibliotheken erstellen, deren Elemente sowohl auf EGA- als auch auf VGA-Systemen verwendet werden können.

Auf der Zeichenoberfläche erscheint in der Regel ein Fadenkreuz, an der aktuellen Cursorposition.

Wenn das Fangraster z.B. auf eine Schrittweite von 5 eingestellt und die Funktion FANG aktiviert ist, kann das Fadenkreuz auf 201 x-Positionen und 151 y-Positionen bewegt werden (0, 5, 10, 15 usw).

2.1.10 Eingabe über Tastatur

Wie bereits oben erwähnt können bestimmte Befehle und Funktionen auch über die Tastatur eingegeben werden. Dies ist mit allen Befehlen möglich die am rechten Rand der Bedieneroberfläche stehen mit Ausnahme von FANG, ORTHO und LETZTES. AUSWAHL-ENDE kann durch Drücken der 'Leer-Taste' aktiviert werden. Die Befehle des Menüs "Zeichnen" können über Tastatur erreicht werden.

2.2 Das Erzeugen von Objekten

Auf den folgenden Seiten sind die zur Verfügung stehenden Zeichenobjekte beschrieben.

Ein Objekt, das durch Aufruf eines Zeichenbefehls gezeichnet wird, wird immer in der aktuellen Farbe, mit der aktuellen Linienart und Linienbreite und (falls das Objekt Textelemente enthält) mit der aktuellen Einstellung von Textfont, Fontbreite und Fonthöhe gezeichnet. Objekte mit gefüllten Flächen benutzen das aktuelle Füllmuster. Die aktuellen Einstellungen dieser Parameter werden am linken Bildschirmrand angezeigt. Soll das nächste Objekt mit geänderten Einstellungen erzeugt werden, muß die Änderung **vor** Aufruf der Zeichenfunktion erfolgen.

2.2.1 Geometrische Grundobjekte

2.2.1.1 Linie

Aufruf über Maus am rechten Bildschirmrand oder über Tastatur: "Linie".

Zuerst ist der Startpunkt, dann der Endpunkt der Linie mit der Maus zu zeigen. Bei "ORTHO = Ein" werden die Linien immer im rechten Winkel gezeichnet.

Die XY-Koordinaten der aktuellen Cursorposition werden am linken oberen Rand angezeigt bis der Startpunkt gesetzt ist. Zur Angabe des Endpunkts wird die Anzeige auf Länge und Winkel umgeschaltet.

2.2.1.2 Rechteck

Aufruf über Maus am rechten Bildschirmrand oder über Tastatur: "Rechteck".

Zuerst ist der Startpunkt, dann der Endpunkt (= diagonaler Eckpunkt) des Rechtecks mit der Maus zu zeigen. Dabei werden gleichzeitig beide Seitenlängen des Rechtecks festgelegt.

2.2.1.3 Balken

Aufruf über Maus am rechten Bildschirmrand oder über Tastatur: "Balken".

Zuerst ist der Startpunkt, dann der Endpunkt (= diagonaler Eckpunkt) des Balkens mit der Maus zu zeigen. Dabei werden gleichzeitig beide Seitenlängen des Balkens festgelegt. Diese Vorgehensweise entspricht der bei der Definition eines Rechtecks. Der Balken wird jedoch mit der aktuellen Zeichenfarbe und dem aktuellen Füllmuster ausgefüllt.

2.2.1.4 Kreis

Aufruf über Maus am rechten Bildschirmrand oder über Tastatur: "Kreis".

Zuerst ist der Mittelpunkt mit der Maus zu zeigen. Danach kann die Größe des zu erzeugenden Kreises durch Bewegen der Maus festgelegt werden. Die Positionsanzeige zeigt den aktuellen Radius an.

2.2.1.5 Bogen

Aufruf über Maus am rechten Bildschirmrand oder über Tastatur: "Bogen".

Durch das Festlegen von Mittelpunkt, Startpunkt und Endpunkt mit der Maus kann ein beliebiger Bogen gezeichnet werden.

Ist die Funktion ORTHO eingeschaltet, beschränkt sich die Auswahl auf Bögen von 90, 180, 270 und 360 Grad. Winkel und Radius werden links-oben angezeigt. Ist der Startpunkt gesetzt, wird der Bogen in mathematisch positiver Richtung aufgebaut (von 0 nach 360 Grad entgegen dem Uhrzeigesinn).

2.2.1.6 Ellipse

Aufruf über Maus am rechten Bildschirmrand oder über Tastatur: "Ellipse".

Erster anzugebender Punkt ist der Mittelpunkt der Ellipse. Der Winkel bei der Festlegung des 1. Radius legt die Halbachsen der Ellipse fest. Abschließend ist der zweite Radius zu zeigen.

2.2.1.7 Text

Aufruf über Maus am rechten Bildschirmrand oder über Tastatur: "Text".

Vor Aufruf des TEXT-Befehls muß der zu verwendende Fonttyp und die richtige Größe eingestellt sein (Änderungen können im Menü unter 'Einstellungen/Fontgröße' vorgenommen werden). Zunächst ist der Startpunkt des Textes mit der Maus zu zeigen. Danach wird der Text über die Tastatur eingegeben.

Abschluß der Eingaben mit der 'Enter'-Taste. Sollen mehrere Zeilen geschrieben werden, kann die rechte Maustaste zur Befehlswiederholung benutzt werden.

Inhalt und Größe des Textes können nachträglich geändert werden.

2.2.1.8 Füllen

Aufruf über Maus am rechten Bildschirmrand oder über Tastatur: "Füllen".

Dieses Objekt wird benutzt, um eine in sich geschlossene Fläche (z.B. ein Kreis) mit einer bestimmten Farbe und einem bestimmten Muster zu füllen.

Nachdem der Befehl FÜLLEN gewählt wurde, muß ein Punkt auf der zu füllenden Fläche angeklickt werden. Um die nun benötigte Begrenzungsfarbe anzugeben, klickt man einfach die Umrandung der zu füllenden Fläche an. Nach Ausführen dieser Festlegung wird die Fläche ausgefüllt.

ACHTUNG: Das oder die Elemente mit der Begrenzungsfarbe müssen die zu füllende Fläche vollständig umschließen. Wenn dies nicht der Fall ist, wird möglicherweise der gesamte Bildschirm gefüllt.

2.2.2 Dynamische Objekte

Objekte werden als dynamische Objekte bezeichnet, wenn sich eine oder mehrere ihrer Eigenschaften im Automatikbetrieb von einer angeschlossenen Steuerung beeinflussen lassen.

Implizit dynamisch sind die Objekte ZAHL, BARGRAPH und TEXTVARIABLE. Diese können Registerwerte der Steuerung als Dezimalzahl (ZAHL), grafisch (BARGRAPH), oder als Text (TEXTVARIABLE) darstellen. Alle darstellbaren Objekte mit Ausnahme von BILDLÖSCH und MENÜ können zu dynamischen Objekten gemacht werden, indem der Befehl DYNAMISCH auf sie angewendet wird. Diese Objekte können danach ihre Farbe abhängig von Bitvariablen in der Steuerung ändern.

2.2.2.1 Zahl

Mit diesem Objekt können Registerinhalte der PASE am Bildschirm als Dezimalzahlenwerte angezeigt werden. Der dargestellte Zahlenwert wird dabei ständig aktualisiert.

Optional kann ein ZAHL-Objekt auch zur Eingabe von Zahlenwerten in Register der PASE benutzt werden.

Zur Definition des ZAHL - Objekts muß zunächst der Startpunkt mit der Maus gezeigt werden.

Es öffnet sich nun ein Fenster, in das die Parameter des ZAHL-Objekts eingetragen werden müssen:

**Register-
nummer:**

Die Nummer des Registers ist einzutragen, dessen Inhalt im Automatikmodus ständig angezeigt werden soll. Beispiel: Die Istposition der Achse 51 soll angezeigt werden: Registernummer = 15109.

Die Registernummer kann auch indirekt angegeben werden. Hierfür ist der Buchstabe **R** vor der Eingabe der Registernummer einzugeben. Auf diese Weise kann die Bedeutung einer Zahlvariablen von der PASE dynamisch festgelegt werden. Anwendung hierfür z.B. ein Satz-Editor am VIADUKT (Siehe Beispiel weiter hinten).

**Register-
offset:**

Wenn die Registernummer indirekt angegeben wurde, kann zur Vereinfachung der Programmierung von Satz-Editoren der REGISTEROFFSET-Parameter benutzt werden. Die Nummer des Registers, dessen Inhalt angezeigt wird, wird wie folgt ermittelt:

$$(\text{Registernummer}) + \text{Offset}$$

**Vorkommas-
stellen:**

Die Darstellung einer ZAHL am Bildschirm erfolgt formatiert mit einer angebbaren Anzahl von Vor- und Nachkommastellen. Sinnvolle Werte für die Anzahl der Vorkommastellen sind 1 bis 7. Die Zahl wird dann rechtsbündig ausgegeben. Soll die Zahl dagegen linksbündig dargestellt werden, muß die Vorkommastelle auf 0 gesetzt werden. Der Platz für ein eventuelles Minuszeichen wird automatisch reserviert.

**Nachkomma-
stellen:**

Angabe der Anzahl von Nachkommastellen: Sinnvoll sind 0 bis 9 Nachkommastellen.

Hinweise: Die PASE rechnet intern mit 24-Bit Zahlen. Der Zahlenbereich ist ± 8 Mio. , d.h. etwa 7 Stellen. Eine Angabe von 9 Nachkommastellen ist zwar möglich, ergibt aber aus obigem Grund keine zusätzliche Information.

**Anzeige-
faktor:**

Der Anzeigefaktor kann zur Umrechnung des Zahlenwerts von der steuerungsinternen Darstellung in eine dem Anwender vertraute Einheit (z.B. physikalische Einheit) verwendet werden. Beispiel: Angabe der Istposition einer Achse (Register xxx09) ist steuerungsintern in Inkrementen. Zur Darstellung am VIADUKT kann diese mithilfe des Anzeigefaktors umgerechnet werden in Millimeter. Bei z.B. 50 Inkrementen pro Millimeter ist daher ein Anzeigefaktor von 0.02 anzugeben, um eine Anzeige in Millimetern zu erhalten.

Der Vorgabewert für den Anzeigefaktor ist 1.0, so daß die Zahlenwerte am VIADUKT den Zahlenwerten in der PASE entsprechen.

**Anzeige-
offset:**

Analog zum Anzeigefaktor kann auch ein Anzeigeoffset spezifiziert werden. Der angegebene Wert wird nach Multiplikation des Registerinhalts mit dem Anzeigefaktor zum Produkt addiert. Anwendung ist z.B. die Anzeige eines Drucks in mbar, der sich aus einem per PID-Karte übersetzten Analogwert ergibt. Hierbei entspricht beispielsweise der Zahlenwert 0: 1000 mbar, der Zahlenwert 2047: 5000 mbar. Um dieses Register in mbar darzustellen muß ein Anzeigefaktor von $4000/2048 = 1.953$ und ein Anzeigeoffset von 1000 angegeben werden.

Der Vorgabewert für den Anzeigeoffset ist 0.0, so daß die Zahlenwerte am VIADUKT den Zahlenwerten in der PASE entsprechen.

Damit sind die Parameter einer ZAHL definiert, die zur Anzeige von Registerinhalten verwendet wird.

Wenn das ZAHL-Objekt zur Eingabe von Zahlenwerten verwendet werden soll, muß in den Parameter "EINGABE ZULÄSSIG" ein **J** eingetragen werden (siehe Kapitel 3.3).

Minimalwert:**Maximalwert:**

Wenn das Zahlobjekt im Automatikmodus des VIADUKT mit dem Cursor ausgewählt wird, kann der Bediener eine Zahleneingabe durchführen. Diese Eingabe wird mit ENTER beendet. Die eingegebene Zahl wird von VIADUKT auf Zulässigkeit geprüft. Der eingegebene Wert muß größer oder gleich dem Minimalwert und kleiner oder gleich dem Maximalwert sein. Ist dies nicht der Fall, erscheint 1 Sekunde lang eine entsprechende Fehlermeldung und das Eingabefeld kann nicht verlassen werden. Indirekte Adressierung ('R') ist möglich.

Wurde der eingegebene Wert akzeptiert, wird der Anzeigeoffset subtrahiert und das Ergebnis durch den Anzeigefaktor dividiert. Das Ergebnis dieser Rechenoperation wird an die PASE übertragen und in das Register eingetragen, das durch die Parameter Registernummer und Registeroffset spezifiziert wird.

2.2.2.2 Bargraph

Mit diesem Objekt können Registerinhalte der PASE am Bildschirm als Balken mit veränderlicher Größe angezeigt werden. Die Größe des dargestellten Balkens wird dabei ständig aktualisiert.

Die grafische Definition eines BARGRAPH erfolgt in drei Schritten: Zunächst muß der Startpunkt der Basislinie, dann der Endpunkt der Basislinie mit der Maus gezeigt werden. Die Basislinie bestimmt die Breite des Balkens sowie die Orientierung der Bewegung: Der Bargraph wird senkrecht zur Basislinie bewegt.

Als dritter Punkt wird der Endpunkt des Balkens (Balkenmaximum, maximaler "Ausschlag") angegeben. Dies legt sowohl die maximale Größe als auch die Bewegungsrichtung fest (über oder unter der Basislinie bzw. rechts oder links davon).

Es öffnet sich nun ein Fenster, in das die Parameter des BARGRAPH-Objekts eingetragen werden müssen:

Registernummer und Registeroffset haben dieselbe Bedeutung wie beim Objekt ZAHL.

Maximalwert: Wenn das angegebene Register diesen Wert enthält, wird der Balken von der Basislinie bis zum Balkenmaximum gezeichnet.

Minimalwert: Wenn das angegebene Register diesen Wert enthält, hat der Balken die Höhe Null, d.h. er wird unsichtbar.

Alle zwischen Minimalwert und Maximalwert liegenden Werte werden in eine entsprechende Größe des Balkens umgerechnet. Liegen die aktuellen Werte außerhalb dieses Bereichs, werden sie an der Bereichsgrenze abgeschnitten. Indirekte Adressierung ('R') ist möglich, die entsprechenden Registerwerte müssen beim Aufruf der Maske feststehen.

2.2.2.3 Textvariable

Mit diesem Objekt können Registerinhalte der PASE am Bildschirm als ASCII-Zeichensequenz angezeigt werden.

Optional kann ein solches Objekt auch zur Eingabe von Text in Register der PASE benutzt werden. Da 8 Bit genügen, um ein Zeichen vollständig zu beschreiben, können drei solche Zeichen in einem 24-Bit-PASE-Register gespeichert werden. Das **erste Register** einer Zeichenfolge wird gesondert interpretiert:

- o das Längenbyte (Bits 0-7) dieses Registers beinhaltet die Anzahl der Zeichen, die vom VIADUKT als gültige Zeichen angezeigt werden.
- o das Statusbyte (Bits 8-15) gibt die Art und Weise an, wie das VIADUKT die Registerfolge während eines Auffrischzyklusses zu behandeln hat:
 - 0 keine neue Anzeige
 - 1 neue Anzeige mit Quittierung (die Zahl 0 wird dabei in das Statusbyte zurückgeschrieben)
 - 2 neue Anzeige ohne Quittierung

Die Bits 16-23 beinhalten schließlich das erste darzustellende Zeichen.

Zur Definition des TEXTVARIABLE-Objektes muß zunächst der Startpunkt mit der Maus gezeigt werden. Es öffnet sich nun ein Fenster, in das die Parameter des TEXTVARIABLE-Objektes eingetragen werden müssen:

Registernummer und Registeroffset haben dieselbe Bedeutung wie beim Objekt ZAHL.

Textlänge: Gibt die maximale Anzahl Zeichen an, die unabhängig von dem aktuellen Längenbyte auf dem Bildschirm dargestellt werden können. Überschreitet das Längenbyte diesen Wert, wird das "∞"-Zeichen anstelle des letzten möglichen Textzeichens ausgegeben. Es kann zwischen drei verschiedenen Modi gewählt werden:

1. Textlänge > 0: linksbündige Anzeige, wobei die Differenz zwischen der angegebenen Textlänge und der Anzahl gelesener Zeichen mit Leerzeichen aufgefüllt wird
2. Textlänge = 0: wie 1. nur ohne Auffüllung mit Leerzeichen
3. Textlänge < 0: rechtsbündig

Eingabe

zulässig: Wird hier ein "J" eingetragen, kann dieses Objekt zur Eingabe von Text in die entsprechenden PASE-Register verwendet werden.

Im übrigen gilt: Es werden nur Zeichen ab der Ordinalzahl 32 (Leerzeichen) angezeigt beziehungsweise als Eingabe akzeptiert. Steuerzeichen (0 - 31) finden keine Anwendung und werden als umgekehrtes Fragezeichen "¿" ausgegeben.

2.2.2.4 Dynamik

DYNAMIK ist kein eigenständiges Objekt, sondern ein Befehl, der es allen bis hierher aufgeführten Objekten ermöglicht, sich bei Eintreten eines bestimmten Ereignisses (z.B. Merker von 0 nach 1) in einer neuen Farbe darzustellen.

Der Aufruf von DYNAMIK erfolgt mit der Maus im Menü ZEICHNEN oder über die Tastatur mit "DYNAMIK". Mit der Maus muß jetzt das Objekt gezeigt werden, das dynamisiert werden soll.

Jetzt kann die gewünschte Umschlagsfarbe angegeben werden. Bei

Füllobjekten darf hier nicht die Farbe der Umrandung der zu füllenden Fläche gewählt werden!!

Die Dynamisierung erfolgt in Abhängigkeit einer Bit-Variablen der angeschlossenen Steuerung. Hierzu zählen EINGANG, AUSGANG, MERKER und REGISTERBIT.

Die Nummer der Bitvariablen ist anzugeben, bei Registern zusätzlich die Nummer des Bits (0-23) innerhalb des Registers.

Auch hier ist eine indirekte Adressierung (s. Zahl, Bargraph) möglich.

Beispiel: Der Zustand des Endschaltes einer Servoachse soll durch Farbumschlag eines ausgefüllten Kreises dargestellt werden. Wenn der Endschalter aktiv ist, soll ein Farbumschlag von grün nach rot erfolgen. Hierzu wird das Füll-Objekt, das den Kreis normalerweise grün füllt dynamisiert. Als NUMMER für die Dynamisierung wird (da der Endschalter an einer SV-Karte und nicht an einer Eingangskarte angeschlossen ist) das Statusregister der SV-Karte angegeben, also z.B. 13101. Als Bitnummer wird 4 angegeben (negativer Endschalter), und als Umschlagfarbe ROT.

2.3 Verändern von Objekten (Editierbefehle):

Zu den Editierbefehlen zählen SCHIEBEN, KOPIEREN, ÄNDERN und LÖSCHEN, ÄND.DYN. (Ändern Dynamik), AUSW.END (Auswahl Ende). Um Objekte zu editieren, müssen diese zunächst ausgewählt werden.

2.3.1 OBJEKTAUSWAHL:

Die Auswahl geschieht mit der Maus (linke Maustaste).

- Linien, Rechtecke, Kreise, Bögen:
Diese geometrischen Objekte werden durch das Anklicken der jeweiligen Objekte an einer beliebigen Stelle ausgewählt.
- Eine Ellipse muß an einem ihrer 4 Scheitelpunkte ausgewählt werden.
- Bargraph, Balken, Knopf, Bild löschen:
Diese Objekte werden jeweils durch das Anklicken ihres Randes ausgewählt (Bild löschen am Rand der Zeichenfläche).
- Text, Zahl, Textvariable, Datum-Zeit:
Texte und Zahlen wählt man durch das Anklicken ihres Startpunktes aus. Zu diesem Zweck wird der Startpunkt durch einen kleinen PUNKT markiert. Diese Markierung ist im Automatikmodus nicht sichtbar.
- Füllmuster (Füllfarbe):
Ein Füllmuster wird ebenfalls durch das Anklicken seines Startpunktes ausgewählt. Zu diesem Zweck wird der Startpunkt durch einen kleinen Punkt markiert, der jeweils eine andere als die Füllfarbe besitzt. Im Automatikmodus ist dieser Punkt nicht sichtbar.
- Menü: Innerhalb des Rechtecks.
- Einfrieren, Include: Durch Drücken der Taste <u> öffnet sich ein Auswahlfenster, in dem "unsichtbare Objekte" ausgewählt werden können.

Außer der direkten Wahl der Objekte auf der Zeichenfläche mit der Maus gibt es noch die Möglichkeit, das letzte Objekt, das gezeichnet wurde, auszuwählen. An der rechten Menüleiste kann dazu das Feld

LETZTES angeklickt werden.

Wenn alle Objekte gewählt sind, die für die nächste Funktion selektiert werden sollen, muß die Auswahl mit der Funktion AUSW.END an der rechten Menüleiste oder mit der LEER-Taste beendet werden.

2.3.2 SCHIEBEN

Mit Hilfe dieser Funktion ist es möglich ein oder mehrere Objekte zu verschieben. Dabei wird zunächst nach den gewünschten Objekten gefragt. Die Auswahl erfolgt wie unter 2.3.1 angegeben. Sind die Objekte gewählt, muß ein Verschiebevektor durch zeigen des Basis- und des Endpunkts angegeben werden ("von wo" "nach wo").

Die Funktion 'Schieben' löscht die ausgewählten Objekte am ursprünglichen Ort und zeichnet sie am neuen Ort erneut.

Hinweis: Die Objekte Bild löschen, Einfrieren und Include sind nicht verschiebbar.

2.3.3 KOPIEREN

Der Ablauf dieses Befehls entspricht dem von 'Schieben'. Das ursprüngliche Objekt bleibt erhalten.

Hinweis: Die Objekte BILD LÖSCHEN, EINFRIEREN und MENÜ sind nicht kopierbar.

2.3.4 ÄNDERN

Dieser Befehl ermöglicht das Ändern von Parametern der Objekte TEXT, ZAHL, BARGRAPH, TEXTVARIABLE, KNOPF, MENÜ, DATUMZEIT, INCLUDE. Nach Auswahl eines Objekts erscheinen dessen änderbare Parameter nacheinander in der Kommandozeile und/oder in einem Parameterfenster. Mit der Bestätigung durch <RETURN> (Kommandozeile) oder 'OK' (Parameterfenster) werden die neuen Parameterwerte übernommen. Zum Ändern des virtuellen Objektes INCLUDE die "u"-Taste betätigen.

2.3.5 LÖSCHEN

Hiermit können beliebige Objekte gelöscht werden. Nach Auswahl der Objekte und Beenden der Auswahl mit AUSW.END (oder LEER-Taste) werden alle gewählten Objekte gelöscht! Vor AUSW.END kann LÖSCHEN mit der rechten Maustaste abgebrochen werden. Zum Löschen virtueller Objekte die "u"-Taste betätigen.

2.3.6 ÄND.DYN. (Ändern Dynamik)

Mit Hilfe dieser Funktion können dynamisierte Objekte geändert werden (siehe 2.2.2.4 Dynamik). Nach dem Aufruf von Änd.Dyn. erscheint ein rechteckiger Cursor mit welchem die zu ändernden Objekte markiert werden können. Danach öffnen sich Fenster in welchen die entsprechenden Einstellungen zur Dynamik verändert werden können. Gelöscht wird die Dynamik eines Objektes mit LÖSCHEN oder der Taste 'L'.

2.3.7 LETZTES

Hiermit werden das - oder bei wiederholtem Einsatz die - jeweils vorhergehenden Objekte selektiert. Mit SHIFT-LETZTES kann in umgekehrter Reihenfolge wieder reselektiert werden. Zuvor muß eine Funktion aufgerufen werden für die eine Objektauswahl mit der Funktion LETZTES getroffen werden soll.

2.3.8 AUSW.END (Auswahlende)

Jedes Auswählen von Objekten für Funktionen wie SCHIEBEN, LÖSCHEN, ÄNDERN, etc. muß mit AUSW.END abgeschlossen werden. Danach können das oder die selektierten Objekte modifiziert, verschoben, etc. werden.

2.4 Erstellen von Masken

2.4.1 Die Vorgabewerte (Defaulteinstellungen)

Font:	Small
Fontgröße:	Breite 2.0 Höhe 2.0
Linienart:	Durchgezogen, dünn
Zeichenfarbe:	Weiß
Füllmuster:	Ausgefüllt
Hintergrundfarbe:	Schwarz

2.4.2 Meine erste Maske

An dem folgenden kleinen Beispiel soll exemplarisch das Erstellen einer Maske beschrieben werden:

Als erstes soll ein Rechteck gezeichnet werden, in dem sich ein Kreis befindet, der dynamisch seine Farbe ändert. Die Abhängigkeit soll sein:

Der Ausgang 401 ist nicht aktiv --> Grün.
Der Ausgang 401 ist aktiv --> Rot.

Darunter soll sich eine Zahl befinden, die in Echtzeit den Inhalt von Register 8704 anzeigt.

Von der DOS-Kommandozeile aus geben Sie VIAGR ein. Es erscheint die Editor-Bedieneroberfläche. Es kann nun sofort mit dem Zeichnen

begonnen werden, der Masken-Name wird erst beim Abspeichern benötigt. Die Zeichnungsschritte werden im folgenden stichwortartig beschrieben.

2.4.2.1 Zeichnen der geometrischen Objekte

- Den Befehl 'Rechteck' an der rechten Menüleiste wählen und durch Drücken der linken Maustaste aktivieren. Die beiden Eckpunkte angeben.
- Weiß als aktuelle Farbe wählen. Den Befehl 'Kreis' anklicken, den Mittelpunkt und den gesamten Kreis innerhalb des Rechtecks plazieren.
- Mit dem Befehl 'Füllen' die Kreisfläche z.B. grün gestalten.
 - Grün als aktuelle Farbe wählen
 - 'Füllen' anklicken
 - Kreisfläche anklicken (zu füllender Bereich)
 - Kreisrand anklicken (Begrenzungsfarbe)
 - Wieder weiß als aktuelle Farbe wählen.

An dieser Stelle ist es sinnvoll, die bisher geleistete Arbeit zu sichern. Wählen Sie hierzu aus dem am oberen Bildrand dargestellten Menü "DATEI" und in dem sich öffnenden Fenster die Zeile SICHERN.

Es erscheint jetzt die Frage nach einem Dateinamen für die neue Maske, die Sie gerade erstellen. Geben Sie 100 ein. Die Maskendaten werden daraufhin in einer Datei mit dem Namen 100.VIA im aktuellen Verzeichnis gesichert.

2.4.2.2 Dynamik

- Um die Abhängigkeit der Füllfarbe des Kreises vom Ausgang 401 herzustellen, wird nun der Befehl DYNAMIK aufgerufen:
- Im Menü unter Einstellungen 'Dynamik' anklicken
- Als gefordertes Objekt den Startpunkt des FÜLL-Objekts anklicken
- Als gewünschte Umschlagfarbe rot eingeben
- 401 als Antwort auf NUMMER eingeben und angeben, daß es sich um eine Ausgangnummer handelt.

2.4.2.3 Definition der Zahl zur Anzeige von Register 8704

- Die Eingaben für das ZAHL-Objekt im Beispiel sind:
 - Den Befehl 'Zahl' anklicken.
 - Den Startpunkt wählen.
 - Die Registernummer des Registers wählen, dessen Werte zyklisch abgefragt werden sollen (in diesem Beispiel: 8704).
 - Anzahl Vorkommastellen angeben: 5
 - Anzahl Nachkommastellen angeben: 0
 - Anzeigefaktor und -Offset müssen nicht geändert werden.
 - Unter Eingabe "N" (Nein) wählen.
 - Parameterfenster mit OK verlassen.

Die Beispielmaste ist jetzt fertig definiert. Sichern Sie die Maskendaten jetzt nochmals. Sie können sich den aktuellen Dateinamen mit der Funktion INFO im Menüpunkt DATEI anzeigen lassen.

Um die Maske in der vollen Bildschirmgröße sehen zu können, wählen Sie unter dem Menüpunkt AUTO die Funktion VOLLANSICHT. Die Maske, die aktuell im Editor geladen ist, wird jetzt so dargestellt, wie sie auch später im Automatikmodus dargestellt wird, d.h. auf den gesamten Bildschirmbereich vergrößert.

Um diese kleine Beispielmaste im Automatikbetrieb zu aktivieren, muß eine PASE an das VIADUKT angeschlossen werden. Starten Sie dann den Automatikbetrieb mit Menü AUTO und AUTOMATIK. Die Maste wird auf dem Bildschirm dargestellt, wenn Sie ins Register 1 der PASE den Wert 100 eintragen (z.B. mit SYMPAS Inbetriebnahmemodus).

2.4.3 Beispiel SATZEDITOR (mit indirekter Adressierung)

Eine typische Anwendung des VIADUKT ist die Ein-/Ausgabe von Datensätzen. Das nachfolgende Beispiel zeigt eine Ein-/Ausgabemaske für 100 Datensätze mit je 4 Parametern.

Aufgabenstellung:

Ein 2-Achs-Roboter soll hintereinander 100 Positionen anfahren. Diese Positionen sollen am Bildschirm dargestellt, bzw. eingegeben werden können. Es sind also 100 Datensätze darzustellen, von denen jeder aus 4 Parametern besteht:

X-Position, Verfahrgeschw. X, Y-Position, Verfahrgeschw. Y

Diese Variablen müssen in der PASE in Registern abgespeichert werden. Die Registeraufteilung soll wie folgt sein:

Satz 1: Reg. 1000:Pos. X
 Reg. 1001:Verfahrgeschw. X
 Reg. 1002:Pos. Y
 Reg. 1003:Verfahrgeschw. Y

Satz 2: Reg. 1004:Pos. X
 Reg. 1005:Verfahrgeschw. X
 Reg. 1006:Pos. Y
 Reg. 1007:Verfahrgeschw. Y

.....

Satz 100: Reg. 1396:Pos. X
Reg. 1397:Verfahrgeschw. X
Reg. 1398:Pos. Y
Reg. 1399:Verfahrgeschw. Y

Auf dem Bildschirm soll nun eine Maske definiert werden, die 5 solcher Datensätze gleichzeitig darstellt. Die Nummer des ersten dargestellten Satzes soll wählbar sein.

Dies wird nun mit Hilfe der indirekten Adressierung realisiert: es wird ein Zeiger definiert (in diesem Beispiel das Register 100), der mit Hilfe des Anzeigefaktors (= 0,25) und des Offsets (= -249) bei der Zahleneingabe als Registernummer der X-Position des ersten anzuzeigenden Satzes ermittelt wird.

$$\text{Register 100} = (\text{Satznummer} - \text{Offset}) / \text{Faktor}$$

Wird beispielsweise die Satznummer "2" vorgewählt, ergibt sich als Zeiger (Register 100) der Wert

$$\text{Register 100} = (2 - -249) / 0,25 = 1004$$

was der X-Position von Satz 2 entspricht.

Die Parameter, die dargestellt werden sollen (in diesem Beispiel die Sätze 2, 3, 4, 5 und 6) sind nun in den Registern ab 1004 abgelegt. Es genügt daher, bei der Parameterdefinition alle Parameter indirekt auf Register 100 zu beziehen, und über einen aufsteigenden Registeroffset die Reihenfolge festzulegen.

Die Maske wird daher wie folgt definiert:

```

Satznummer:          Register    = 10
=====             AnzFaktor   = 0,25
                    AnzOffset   = -249

```

Satz	X-Pos	X-Geschw	Y-Pos	Y-Geschw.
Reg=100 RegOffset=0 AnzFakt=0.25 AnzOffs=-249	Reg=R100 RegOffset=0	Reg=R100 RegOffset=1	Reg=R100 RegOffset=2	Reg=R100 RegOffset=3
Reg=100 RegOffset=0 AnzFakt=0.25 AnzOffs=-248	Reg=R100 RegOffset=4	Reg=R100 RegOffset=5	Reg=R100 RegOffset=6	Reg=R100 RegOffset=7
Reg=100 RegOffset=0 AnzFakt=0.25 AnzOffs=-247	Reg=R100 RegOffset=8	Reg=R100 RegOffset=9	Reg=R100 RegOffset=10	Reg=R100 RegOffset=11
Reg=100 RegOffset=0 AnzFakt=0.25 AnzOffs=-246	Reg=R100 RegOffset=12	Reg=R100 RegOffset=13	Reg=R100 RegOffset=14	Reg=R100 RegOffset=15
Reg=100 RegOffset=0 AnzFakt=0.25 AnzOffs=-245	Reg=R100 RegOffset=16	Reg=R100 RegOffset=17	Reg=R100 RegOffset=18	Reg=R100 RegOffset=19

Im letzten dargestellten Satz werden in diesem Beispiel (Satz 2 => Register 100 = 1004) demzufolge dargestellt:

```

Satznummer      : (1004 * 0,25) + -245 = 6
Position X      : Inhalt von Reg(1004 + 16), also Inhalt von R1020
Geschw. X       : Inhalt von Reg(1004 + 17), also Inhalt von R1021
Position Y      : Inhalt von Reg(1004 + 18), also Inhalt von R1022
Geschw. Y       : Inhalt von Reg(1004 + 19), also Inhalt von R1023

```

2.5 Die Aufrufparameter des VIADUKT (DOS-Kommandozeile)

Durch Aufruf von VIADUKT über den Dateinamen (ohne Parameter /A), wird der Editor aktiviert und die in 2.1 beschriebene Bedieneroberfläche angezeigt:

VIAGR

Eine Demoversion wird am Fehlen des Hardware-Locks erkannt und als solche aufgerufen.

Beim Aufruf von VIADUKT können Parameter angegeben werden, die die Arbeitsweise u.a. im Automatikbetrieb beeinflussen:

VIAGR /A

Das Programm startet sofort den Automatikbetrieb. In diesem Fall ist ein Maustreiber zur Funktion des Programmes nicht unbedingt erforderlich.

Dieser Parameter wird normalerweise angegeben, nachdem alle Masken erstellt worden sind und das VIADUKT kundenseitig installiert wurde.

Beim VIADUKT (Hardware) steht diese Zeile am Ende der AUTOEXEC.BAT Datei, so daß nach einem Neustart des Rechners automatisch das VIADUKT-Programm aufgerufen wird und dort automatisch der Automatikmodus aktiviert wird.

VIAGR /Rxxxx

Nummer des Registers, das die darzustellende Maskennummer enthält. Voreinstellung: 1.

Das VIADUKT fragt zur Ermittlung der Nummer der darzustellenden Maske zyklisch ein bestimmtes Register der angeschlossenen Steuerung ab. Wenn beim Start von VIADUKT nichts anderes angegeben wurde, wird Register Nr. 1 für diesen Zweck benutzt. Soll ein anderes Register diese Funktion

übernehmen, muß es beim Aufruf von VIADUKT angegeben werden. Bsp: Register 1000 soll die Maskennummer enthalten (Siehe auch 3.1):

VIAGR /R1000

VIAGR /Gaaa

VIADUKT erkennt beim Programmstart automatisch den im System vorhandenen Grafikadapter (EGA oder VGA). Über die Angabe /GEGA kann erzwungen werden, daß selbst auf Systemen mit VGA-Karte die Grafiken im EGA-Modus dargestellt werden.

"aaa" steht für Adaptername. Hier ist "EGA" oder "VGA" zulässig.

VIAGR /Fxxxx

Nummer des ersten Merkers, der durch die Funktionstasten des VIADUKT beeinflusst werden soll. Ohne Angabe dieses Parameters wird F1 auf Merker 201 abgebildet, F2 auf 202 usw.

VIAGR /Dnnnn

Aktivierung der Datentransferfunktionen des VIADUKT. Die anzugebende Nummer ist die Nummer des ersten Registers des Transfer-Kontrollblocks (Siehe Kapitel 6).

VIAGR /Px

Bestimmung der Steuerung, mit der das VIADUKT im Visualisierungsbetrieb gekoppelt ist. "x" steht für den Steuerungstyp. Hier ist "E" für PASE-E, "M" für PASE-Mikro oder "J" für PASE-J zulässig (Voreinstellung ist PASE-E).

VIAGR /X <cc> oder <Dateiname>

Dieser Schalter dient der näheren Spezifikation des Graphikadapters des VIADUKT. Dabei kann einmal durch einen Zahlenwert <cc> die Farbe des Fadenkreuzes beziehungsweise des Eingaberahmens definiert werden, zum anderen können über die Angabe einer Definitionsdatei <Dateiname> weitgehende Definitionen zum Graphikadapter getroffen werden.

VIAGR /Cxxxx (nicht VIADUKT lite)

Mit dieser Funktion können das Datum und die Uhrzeit aus dem VIADUKT in Register der PASE-Steuerung übertragen werden. Der dem Schalter mitgegebene Parameter **xxxx** definiert das Anfangsregister eines Registerblockes in der PASE-Steuerung. Die einzelnen Register übernehmen dabei folgende Funktionen:

```
xxxx    = Handshakeregister (0: Übertragung findet statt,  
                             1: keine Übertragung des Datums und der Zeit)  
xxxx+1  = Minuten (0..59)  
xxxx+2  = Stunden (0..23)  
xxxx+3  = Wochentag (0 = Sonntag, ...)  
xxxx+4  = Tag      (1..31)  
xxxx+5  = Monat   (1..12)  
xxxx+6  = Jahr    (1980..2099)
```

Folgende Aufrufparameter werden in den genannten Kapiteln beziehungsweise Beschreibungen erklärt.

- o /W, /H, /L siehe Beschreibung zum Meldungseditor.
- o /U, /V siehe Kapitel Funktionserweiterung durch TSR-Programme respektive die Datei USERPROG.PAS.
- o /E siehe Kapitel DOS-Funktionen.

Es können auch mehrere der obigen Parameter innerhalb der Aufrufzeile erscheinen. Beispiel: VIADUKT im Automatikmodus starten, Register 99 als Maskenregister und die Merker 1 bis 14 als Merker für die Funktionstasten definieren:

```
VIAGR /A /R99 /F1
```

Für die tägliche Arbeit mit dem VIADUKT kann es hilfreich sein, diese Kommandozeilenparameter nicht immer erneut eingeben zu müssen. Abhilfe kann hier eine Stapeldatei schaffen, die die Aufrufzeile komplett enthält. Sie ist dann jedoch auf die eingetragenen Werte fixiert.

Eine flexiblere Methode bietet VIADUKT an: Tragen Sie die Parameter der Aufrufzeile in eine Datei mit der Dateierweiterung ".CMD" ein und geben Sie diesen Dateinamen als Parameter der Aufrufzeile an. Zusätzlich können Sie weitere Aufrufparameter angeben, wobei immer die zuletzt gesetzten Parameter Gültigkeit haben.

Beispiel: Die Datei M.CMD enthalte folgende Zeile:

```
/R99 /F1 /D100
```

Ein Aufruf von VIADUKT mit folgender DOS-Zeile

```
VIAGR M.CMD /A
```

ruft VIADUKT auf. VIADUKT liest zunächst die Datei M.CMD und stellt dadurch das Maskennummerregister auf Nr. 99, Merker für Funktionstasten auf Merker Nr. 1 bis 14 und schaltet die Dateitransferfunktion ein. Anschließend wird der Rest der Aufrufzeile bearbeitet. In diesem Beispiel wird direkt der Automatikmodus gestartet. Wenn das **/A** weggelassen wird, kann mit denselben Einstellungen der VIADUKT-Editor aufgerufen werden.

2.5.1 Updates (neue Programmversionen)

Wie jedes andere Programm, ist auch VIADUKT möglicherweise nicht frei von Fehlfunktionen, die erst nach Auslieferung entdeckt werden. Um den Vorgang der Installation einer neuen Programmversion zu vereinfachen, bietet VIADUKT die Möglichkeit, diesen UPDATE-Vorgang zu automatisieren. Durch diese Funktion kann die Installation einer neuen VIADUKT-Version auch durch den Endkunden erfolgen.

VIADUKT bietet hierzu beim Aufrufparameter **/A** eine Option an:

Wird VIADUKT mit **/A** gefolgt von einer Ziffer zwischen 1 und 9 aufgerufen (also z.B. VIAGR /A5), hat der Bediener nach einem Start des VIADUKT-Programms 5 Sekunden lang die Möglichkeit, ein automatisches Update anzufordern. Hierzu muß innerhalb der angegebenen Zeit die Funktionstaste F6 gedrückt werden. VIADUKT fordert den Bediener dann auf die Diskette mit der neuen Programmversion in das Diskettenlaufwerk einzulegen und anschließend die ENTER-Taste zu betätigen.

Das VIADUKT-Programm wird dadurch beendet und ein DOS-ERRORLEVEL von 97 an DOS zurückgeliefert. Das Programm, das VIADUKT aufgerufen hat, sollte jetzt dafür sorgen, daß die notwendigen Befehle zur Überspielung der neuen Programmversion durchgeführt werden.

Stapeldatei UPDATBSP.BAT, die diese Funktion beispielhaft benutzt, ist auf der Originaldiskette von VIADUKT enthalten.

2.6 Funktionstasten

2.6.1 Grundfunktionen der Funktionstasten

Die Funktionstasten des VIADUKT können direkt für Programmverzweigungen in PASE-Programmen benutzt werden. Zu diesem Zweck werden die Funktionstasten auf Merker innerhalb der Steuerung abgebildet. Ohne Verwendung der Kommandozeilenoption /F gilt folgende Zuordnung:

F1	Merker 201
F2	Merker 202
F3	Merker 203
usw.	
F14	Merker 214

Über die Kommandozeilenoption /F kann eine andere Zuordnung definiert werden (siehe 2.5)

Durch Drücken einer Funktionstaste am VIADUKT wird der zugeordnete Merker in der Steuerung gesetzt. Umgekehrt wird der zugehörige Merker nach Loslassen der Taste wieder zurückgesetzt.

Sind die Funktionstasten F11 bis F14 nicht vorhanden, kann ihre Funktion auch über SHIFT-F1 bis SHIFT-F4 erzeugt werden.

2.6.2 Menü-Funktion

Mit Hilfe der Funktion 'Menü' im Hauptmenü unter 'Zeichnen' wird die Beschriftung der Funktionstasten vereinfacht. Außerdem können Menüstrukturen erstellt werden, die vom Programmablauf in der PASE unabhängig sind.

2.6.2.1 Reine Beschriftungsfelder

Die Beschriftungsfelder erscheinen am rechten und unteren Rand einer Maske in Form eines Rahmens mit zwei kurzen Texten. Bei Bedarf können sie mit SCHIEBEN beliebig verschoben werden.

Der Ablauf solch einer Menüerstellung sieht so aus:

- Anklicken der Funktion 'Menü' im Hauptmenü unter 'Zeichnen'. Daraufhin öffnet sich ein Fenster.
- Eingabe der Funktionstaste, für die ihre Bedeutung angezeigt werden soll (Zahlen von 1 bis 14 möglich).
- Eingabe der ersten Zeile des Textes (Max. 8 Zeichen).
- Eingabe der zweiten Zeile des Textes (Max. 8 Zeichen).

Hinweis: Wird kein Text eingegeben geht VIADUKT davon aus, daß auch kein Rahmen gezeichnet werden soll. Dies ist insbesondere dann interessant, wenn ein Menüfeld mit einem Maskenaufruf gekoppelt ist (s. u.) und eigene Menütexte in beliebiger Größe dargestellt werden sollen.

2.6.2.2 Menüfeld mit Maskenaufruf

Die Reihenfolge der Eingabe ist wie unter 2.6.2.1 angegeben. Zusätzlich wird jetzt die Nummer der Maske angegeben, die aufgerufen werden soll, wenn die ausgewählte Funktionstaste gedrückt wird. VIADUKT trägt diese Nummer in das Istmaskenregister (Default: Register Nr. 3) ein.

Die durch eine Funktionstaste aufgerufene Maske kann wiederum neue MENÜ-Befehle enthalten, die die Funktionstasten mit anderen Funktionen belegen. Auf diese Art und Weise kann eine Menüstruktur erstellt werden, die es unabhängig vom PASE-Programm erlaubt verschiedenste Masken aufzurufen.

Die jeweils zuletzt geladene MENÜ-Funktion einer Funktionstaste bleibt aktiv, bis sie durch eine MENÜ-Funktion derselben Funktionstastenummer überschrieben wird, oder bis eine Maske angezeigt wird, die einen BILD-LÖSCH - Befehl enthält.

Wird ein BILD-LÖSCH-Befehl ausgeführt, sind anschließend alle MENÜ-Definitionen der Funktionstasten gelöscht.

3. Automatikbetrieb

Der Automatikbetrieb wird erreicht durch

- Aufruf mit Parameter /A in der Aufrufzeile
- oder
- Auswahl von AUTOMATIK im Menü AUTO.

Mit der Tastenkombination 'CTRL+End' wird der Automatikmodus wieder verlassen. Wurde er von der Zeichenoberfläche aus aufgerufen, gelangt man dorthin zurück. Wurde der Automatikmodus von DOS direkt über Parameter /A aufgerufen, beendet 'CTRL+End' das Programm und kehrt zu DOS zurück.

Im Automatikmodus übernimmt das VIADUKT-Programm folgende Funktionen:

- Zyklische Abfrage des Registers, das die Nummer der darzustellenden Maske enthält. Falls sich diese Nummer ändert, wird die neue Nummer als Name einer Maske betrachtet und diese dargestellt (falls eine Datei mit diesem Namen existiert). (Siehe Kapitel 3.1)
- Beobachtung der Funktionstasten: Durch Drücken oder Loslassen einer Funktionstaste wird der zugeordnete Merker gesetzt bzw. rückgesetzt.
Wurden MENÜ-Funktionen definiert und einer Funktionstaste eine Maskennummer zugeordnet, wird die entsprechende Maske bei Tastendruck angezeigt und die Nummer ins Istmaskenregister der PASE eingetragen (2.6).
- Falls Eingabevariablen auf dem Bildschirm dargestellt sind, wird eine Zahleneingabe vom Bediener akzeptiert. (3.3)
- Zyklische Abfrage der Register des Datentransfer-

kontrollblocks. Falls hier Kommandowerte eingetragen sind, werden diese ausgeführt (Siehe Kapitel 6).

- Die Darstellung der Meldungsfenster (Aufrufparameter /W, /H, /L).
- Die Zeitübertragung an die PASE (Aufrufparameter /C).

Ist die PASE nicht angeschlossen oder gibt es Probleme mit der Verbindung zwischen PASE und Rechner bzw. **VIADUKT**, erscheint die Fehlermeldung "PASE nicht bereit". Mit der Tastenkombination 'CTRL + End' kann auch hier der Automatikmodus verlassen werden.

3.1 Maskenanzeige

Innerhalb des Automatikmodus erfragt das VIADUKT periodisch den Inhalt des Registers (**Sollmaskenregister**), das die Maskennummer enthält. Dies ist normalerweise das Register Nr. 1. Unter Verwendung des Kommandozeilenparameters /R (siehe 2.5) kann jedoch ein beliebiges Register diese Funktion erhalten.

In jedem Fall werden auch die zehn nachfolgenden Register von VIADUKT belegt. Es handelt sich unter anderem um das sogenannte **Maskenquittungsregister** sowie um das **Istmaskenregister**.

VIADUKT versucht eine Datei zu lesen, deren Namen sich aus dem Zahlenwert des Soll-Maskenregisters sowie der Dateierweiterung ".VIA" zusammensetzt. Wenn diese Datei gefunden und die Maske angezeigt wurde, setzt VIADUKT das sogenannte Maskenquittungsregister auf den Wert, der im Sollregister steht. Daran kann die PASE erkennen, daß die Maske angezeigt wurde. Gleichzeitig wird auch in das Istmaskenregister (enthält immer die aktuell angezeigte Maskennummer, auch wenn die Maske via Menü-Funktion

aktiviert wurde) die neue Maskennummer eingetragen.

Sollte die gewünschte Maske nicht vorhanden sein oder ein Fehler beim Lesen der Datei geschehen, wird das Maskenquittungsregister auf den Wert -1 gesetzt.

3.2 Funktionstasten

Die Funktionstasten auf der VIADUKT-Frontplatte (F1 bis F14) bzw. die Funktionstasten F1 bis F12 einer PC-Tastatur werden dazu benutzt, bestimmte Merker in der PASE zu steuern. Wenn eine Funktionstaste gedrückt wird, dann wird der zugeordnete Merker in der PASE gesetzt. Wird die Taste wieder losgelassen, dann wird der zugeordnete Merker wieder zurückgesetzt. Die Zuordnung zwischen Funktionstasten und Merkern ist wie folgt:

F1	Merker 201
F2	Merker 202
F3	Merker 203
usw.	
F14	Merker 214

Über die Kommandozeilenoption /F kann eine andere Zuordnung definiert werden (siehe Kapitel 2.5).

Wenn weniger als 14 Funktionstasten vorhanden sind, können die darüber hinaus gehenden Merker mit 'SHIFT + Funktionstaste' erreicht werden. Sind zum Beispiel 12 Funktionstasten vorhanden, wird Merker 213 mit 'SHIFT + F3' und Merker 214 mit 'SHIFT + F4' angesprochen.

F11	=	SHIFT F1
F12	=	SHIFT F2
F13	=	SHIFT F3
F14	=	SHIFT F4

3.3 Zahleneingabe

Wird am Bildschirm eine Maske dargestellt, die ZAHL-Variablen enthält, welche zur Eingabe zugelassen wurden, kann der Bediener den Zahlenwert des dieser ZAHL-Variablen zugeordneten Registers verändern. Dazu kann ein CURSOR auf allen angezeigten und für Eingabe zugelassenen ZAHL-Variablen bewegt werden. Der CURSOR ist ein RECHTECK, das als Rahmen um die eingebare Zahl dargestellt wird. Nach Aufruf einer Maske ist der CURSOR zunächst nicht sichtbar, befindet sich jedoch "unsichtbar" auf der ersten für Eingabe zugelassenen ZAHL-Variablen. Bei mehreren Variablen ist dies diejenige, deren Definitionspunkt am weitesten links oben dargestellt wird.

Der Cursor wird dargestellt, sobald eine der folgenden Tasten betätigt wird:

- ENTER "Öffnet" die Eingabe der aktuellen Zahlvariablen. Als Startwert der Eingabe wird der momentan aktuelle Wert angezeigt.
- Cursorpfeile Bewegt den Cursor auf die nächste Zahlvariable in Richtung der Pfeiltaste.
- 0 bis 9 Öffnet die Eingabe der aktuellen Zahlvariablen. Die eingegebene Ziffer wird als Teil der damit begonnenen Eingabe betrachtet (Der aktuelle Zahlenwert am Beginn der Eingabe ist damit gelöscht).

Das Eingabefeld bleibt aktiv, bis die Eingabe auf eine der unten angegebenen Arten beendet wird. Solange die Eingabe aktiv ist, wird diese Variable nicht durch den dynamischen Anzeigevorgang verändert.

Während der Eingabe können folgende Tasten betätigt werden:

- 0 bis 9 Eingabe der Zahl

- BACKSPACE Löschen der zuletzt eingegebenen Ziffer
- DELETE oder C Löschen der gesamten eingegebenen Zahl. Es wird wieder der Zahlenwert angezeigt, der am Beginn der Eingabe aktuell war.

Die Eingabe wird durch eine der folgenden Tasten beendet:

- ENTER Beenden der Eingabe. Cursor bleibt auf aktueller Variable, wird jedoch unsichtbar.

Pfeiltasten Beenden der Eingabe, Cursor wird auf das nächste Eingabefeld in der gewählten Richtung gesetzt (siehe 3.3.1 Cursor-Logik).

- ESC (= F14 auf VIADUKT-Hardware)
Beenden der Eingabe. Es erfolgt keine Übertragung eines Werts an die PASE. Der Cursor bleibt auf der aktuellen Variable, wird jedoch unsichtbar.

Bei Betätigen der ENTER-Taste oder der Pfeiltasten wird die eingegebene Zahl gegen die in der ZAHL-Definition festgelegten Grenzen geprüft. Falls der Wert innerhalb der Grenzen liegt, wird der neue Wert akzeptiert und (nach Subtraktion des Offset und Division durch den Anzeigefaktor) an die PASE übertragen.

Wenn die eingegebene Zahl nicht innerhalb der Grenzen liegt, ist ein Verlassen des Eingabefeldes nicht möglich. Es muß also zunächst ein gültiger Wert eingegeben werden (Mit DELETE kann der ursprünglich aktuelle Wert wieder zur Anzeige gebracht werden).

3.3.1 Cursor-Logik

Grundlage für den Algorithmus, der die Weichen für das nächste Eingabefeld bei gedrückter Pfeiltaste stellt, ist der Startpunkt einer jeden Zahl. Priorität hat dabei immer das nächste Eingabefeld, das auf gleicher Achse in Pfeilrichtung folgt. Dieser Umstand kann beim Erstellen einer Masken am einfachsten dadurch erzwungen werden, indem man die Fang-Funktion aktiviert und deren Raster entsprechend einstellt.

Stehen die Eingabefelder dagegen wahllos im Raum, wird die am nächsten mögliche Zahl zunächst auf einen 90-Grad-Bereich (Winkelhalbierende ist die Pfeilrichtungssachse) ausgehend vom aktuellen Eingabefeld beschränkt, in dem schließlich das Feld mit dem geringsten Abstand ausgewählt wird. Sollten sich in diesem Bereich keine Eingabefelder befinden, wird das nächste in Pfeilrichtung mit dem kürzesten Abstand ausgewählt. Wurde auch jetzt nichts gefunden, bleibt der Cursor auf der aktuellen Eingabezahl und wird unsichtbar.

3.4 Monitorfunktion

Die Monitorfunktion des VIADUKT ermöglicht es, in Anlehnung an das Anzeige/Tastaturmodul LCD9/10/11/12 (siehe Kapitel III / 7), Speicherbereiche der PASE zu lesen und falls erlaubt auch zu beschreiben.

Ein entsprechendes Eingabefenster kann durch die Tasten "R" und "I" ("I/O" bei VIADUKT Hardware-Einschub) im Visualisierungsbetrieb aktiviert werden, wobei der gerade angezeigte Prozeß zwischenzeitlich im Hintergrund "eingefroren" wird.

Registerfunktionen werden immer durch die Taste "R" eingeleitet. Zweimaliges Betätigen der Taste "R" aktiviert hingegen die **Merkerfunktionen**. Dreimaliges Betätigen der Taste "R" aktiviert die **Textvariablenfunktion**.

Ähnliches gilt für die **Ausgangs-** und **Eingangsfunktionen**. Jene werden durch die Taste "I" ("I/O") eingeleitet.

Nun muß die Nummer der gewählten Funktion eingegeben werden, wobei nur folgende Wertebereiche akzeptiert werden:

0 - 65535 Register, wobei	
50000 - 59999 Netzwerkregister sind	(PASE-E, PASE-Mikro)
0 - 4095 Merker	(PASE-E)
1 - 2655 Merker	(PASE-Mikro)
101 - 3216 Ausgang / Eingang	(PASE-E)
1 -128 Eingang	(PASE-Mikro)
1 - 80 Ausgang	(PASE-Mikro)

Nach Bestätigung durch die Taste <ENTER> wird der jeweilige Inhalt in gleichnamigen Fenster angezeigt und in kurzen Zyklen erneuert. Aus dieser Ebene (Anzeigemodus) heraus können durch nachfolgend beschriebene Tasten weitere Schritte unternommen werden:

"R"	Register-, Merker-, Textvariablenfunktion (siehe oben).
"I"	Eingangs-, Ausgangsfunktion (siehe oben).
"SHIFT-R"	Aktuelle Programmzeile, Taskstatusanzeige siehe Anmerkung unten.
"ESC", "ENTER"	Fenster wird geschlossen und Prozeßvisualisierung wird fortgesetzt.

" = ", "LEER":

Wenn es sich bei der eingestellten Funktion nicht um Eingänge handelt bzw. die anderen Funktionen nicht durch das Register 8224 (PASE-E), Register 65 (PASE-Mikro) gesperrt sind, wird hierdurch die Eingabe eines neuen Wertes eingeleitet.

Ein solcher kann folgendermaßen editiert werden:

"0".. "9", "-": Bestimmung der Wertigkeit

"." : Dezimalpunkt bei Fließkommazahlregistern
(8960..9015; nur bei PASE-E)
 "Del", "C" : Eingabezahl = 0 --> Abbruch (Anzeigemodus)
 Eingabezahl < > 0 --> Eingabezahl = 0
 "Esc": Abbruch und zurück zu Anzeigemodus
 "ENTER" : Eingebenen Wert an PASE senden;
 Merker, Ausgang:
 Eingabewert = 0 --> Sendewert = 0
 Eingabewert < > 0 --> Sendewert = 1
 Register:
 Sendewert = Eingabewert
 Danach zurück in Anzeigemodus.

Anmerkung zu **SHIFT-R**: Zur Darstellung des Taskstatus ist die RT-Datei des aktuellen PASE-Programmes notwendig, die beim Kompilieren des PASE-Programmes von SYMPAS generiert wird. Der Pfad der Datei muß dem VIADUKT mitgeteilt werden, wenn diese nicht im aktuellen Verzeichnis abgelegt worden ist. Dies geschieht mit dem DOS-Befehl:

```
SET RT_DIRECTORY=Pfad
```

4. Reservierte PASE-Registerbereiche

Hier werden die Registerbereiche beschrieben, die vom VIADUKT in der PASE-Steuerung verwendet werden. Vorab ist zu erwähnen, daß der Beginn des Registerblocks in der PASE-Steuerung über welchen das Zusammenwirken zwischen PASE und VIADUKT gesteuert wird vom Anwender definiert werden kann (Schalter /Rxxxx). Wird von seiten des Anwenders keine Spezifikation vorgenommen weist die Defaulteinstellung dem ersten Register dieses Registerblocks die Nummer 1 zu (**PASE-Register 1**).

Der Registerblock der das Zusammenwirken von PASE-Steuerung und VIADUKT regelt sieht folgendermaßen aus:

Register x	=	Sollmaskenregister
x+1	=	Quittungsregister
x+2	=	Istmaskenregister
x+3	=	Statusregister (VIADUKT)
x+4	=	Maskenoffsetregister
x+5	=	Statusregister (PASE)
x+6	=	Eingaberegister
x+7	=	reserviert
x+8	=	reserviert
x+9	=	reserviert

Sollmaskenregister

In diesem Register ist die Nummer der Maske gespeichert, die vom VIADUKT dargestellt wird (im Automatikbetrieb). Die in dem Register enthaltene Maskennummer korrespondiert zu einer VIA-Datei. Befindet sich in dem Register der Zahlenwert 110 so wird die Maske mit dem Dateinamen 110.VIA, welche zuvor im VIADUKT-Editor erstellt wurde, im Automatikbetrieb auf dem VIADUKT-Bildschirm dargestellt.

Quittungsregister

In das Quittungsregister trägt das VIADUKT stets die Maskennummer ein, welche im Moment auf dem VIADUKT-Bildschirm dargestellt wird (Automatikbetrieb). Wurde die im Sollmaskenregister angeforderte Maske nicht auf dem VIADUKT-Bildschirm angezeigt, erscheint der Wert '-1' im Quittungsregister.

Istmaskenregister

Wird von einer Maske, welche über das Sollmaskenregister aufgerufen wurde, über eine Menüfunktion zu einer weiteren Maske verzweigt so bleibt der Zahlenwert der Basismaske im Quittungsregister erhalten und ist somit identisch zum Zahlen- respektive Maskenwert im Sollmaskenregister. Im Istmaskenregister jedoch befindet sich die Nummer der Maske, welche von der Basismaske aus über eine Menüfunktion aufgerufen wurde (Automatikbetrieb).

Statusregister (VIADUKT)

Die einzelnen Bits dieses Registers haben folgende Bedeutung:

Bit 0 : 0 = keine Eingabe aktiv
1 = ein ZAHL- oder TEXTVARIABLE-Objekt wird gerade eingegeben

Maskenoffsetregister

Ein hier eingetragener Wert wird zu dem Wert des Sollmaskenregisters addiert. Aus der Summe resultiert die Nummer der Maske, welche auf dem Bildschirm des VIADUKT dargestellt wird (Automatikbetrieb).

Statusregister (PASE)

Die einzelnen Bits dieses Registers haben folgende Bedeutung:

- Bit 0 : 0 = die Eingabe in ZAHL- oder TEXTVARIABLE-Objekte ist erlaubt
1 = die Eingabe ist nicht erlaubt - noch nicht abgeschlossene Eingaben werden unterbrochen
- Bit 1 : 0 = der Inhalt des Lese- und Schreibregisters wird direkt als Dateiname verwendet (siehe Datentransferfunktion)
1 = der Inhalt des Lese- und Schreibregisters wird als Zeiger auf einen Dateinamen verwendet. (Enthält dieser Wildcards so wird ein Dateiauswahlfenster angeboten)
- Bit 2 : 0 = siehe Bit 2 = 1
1 = wurde eine DA-Datei über ein Dateiauswahlfenster aktiviert, so wird der gewählte Dateiname in das Textregister übernommen
- Bit 3 : 0 = siehe Bit 3 = 1
1 = wird eine Maske über das MENÜ-Objekt aufgerufen, so wird zusätzlich kein Merker gesetzt (die Definition dieses Bits muß vor dem Start von VIADUKT erfolgen)
- Bit 4 : 0 = Dateiname wird neu angelegt. Eine alte Datei mit gleichem Namen wird dabei überschrieben.
1 = Datenblock wird an die bereits bestehende Datei mit dem angegebenen Namen angehängt. Existiert noch keine solche Datei, wird diese zuerst angelegt.

Eingaberegister

Wird eine Eingabe in ein ZAHL- oder TEXTVARIABLE-Objekt vorgenommen so erscheint in diesem Register die Nummer des Register auf das in diesem Zusammenhang zugegriffen wird. Der Wert '-1' im Eingaberegister weist darauf hin, daß keine Eingabe stattfindet.

5. Reservierte PASE-Merkerbereiche

Hier werden die Merkerbereiche beschrieben, die vom VIADUKT in der PASE-Steuerung verwendet werden. Vorab ist zu erwähnen, daß der Beginn des Merkerblocks in der PASE-Steuerung, über welchen das Zusammenwirken zwischen PASE und VIADUKT gesteuert wird, vom Anwender definiert werden kann (Schalter /Fxxxx). Wird von seiten des Anwenders keine Spezifikation vorgenommen weist die Defaulteinstellung dem ersten Merker dieses Merkerblocks die Nummer 201 zu (**PASE-Merker 201**).

Der Merkerblock der das Zusammenwirken von PASE-Steuerung und VIADUKT unterstützt sieht folgendermaßen aus (Defaulteinstellungen vorausgesetzt, siehe Ausführung oben):

Funktionstaste auf dem VIADUKT	PASE-Merker
F1	201
F2	202
F3	203
F4	204
F5	205
F6	206
F7	207
F8	208
F9	209
F10	210
F11	211
F12	212
F13	213
F14	214
↑	221
↓	222
←	223
→	224
Bild ↑	225
Bild ↓	226

Die PASE-Merker 201 bis 250 sind beim Einsatz des VIADUKT reserviert.

6. Datentransferfunktion (nicht VIADUKT lite)

Bei flexiblen Bearbeitungsmaschinen, die eine Vielzahl unterschiedlicher Parametersätze für die zu bearbeitenden Werkstücke speichern müssen, bietet die Datentransferfunktion einem PASE-Programm die Möglichkeit, Registersätze auf die Festplatte des VIADUKT-Rechners auszulagern. Die Festplatte wird als "externer" Speicher für PASE-Register benutzt. Die PASE kann hierzu Kommandos zum Lesen und Speichern (Schreiben) von Dateien auf der Festplatte geben. Beim Schreiben kann die PASE festlegen, welcher Registerbereich auf der Festplatte unter welchem Namen gespeichert werden soll.

Um die Datentransferfunktion benutzen zu können, muß beim Aufruf von VIADUKT in der Aufrufzeile der /D - Parameter angegeben werden. Im Anschluß an den Buchstaben D ist die Nummer eines Registers anzugeben, das den Beginn des Transfer-Kontrollblocks darstellt.

Der Kontrollblock ist ein Block von 4 zusammenhängenden Registern der PASE, die die unten beschriebene Bedeutung haben. Wird z.B. folgende Kommandozeile angegeben

VIAGR /D200

so beginnt der Kontrollblock mit dem Register 200 und umfaßt die Register 200 bis 203. Die Register des Blocks haben folgende Namen:

200 Leseregister
201 Schreibregister
202 Startregisternummer
203 Endregisternummer

Die Nummern (200 bis 203) beziehen sich auf das angegebene Beispiel /D200.

VIADUKT liest im Automatikmodus periodisch den Inhalt des Leseregisters und des Schreibregisters. Diese Register muß das PASE-Programm verwenden, um die Datentransferfunktion zu aktivieren.

Im einzelnen haben die Register folgende Bedeutung:

Leseregister:

Wenn das PASE-Programm in dieses Register eine positive Zahl größer Null schreibt, wird das VIADUKT dadurch aufgefordert, eine Datei zu lesen und deren Inhalt an die PASE zu senden. Der Name der Datei ergibt sich aus dem Inhalt des Leseregisters (1 bis 8000000) und der Dateierweiterung ".DA".

Eine solche Datei kann mit der Datentransferfunktion "schreiben" oder mit dem Programm SYMPAS erzeugt werden.

Nach der Übertragung schreibt das VIADUKT einen Fehlercode in das Leseregister. Dieser kann durch den Anwender ausgewertet werden:

- 0 = Datenübertragung fehlerfrei
- 1 = Dateiauswahlfenster wurde mit ESC abgebrochen
- 2 = Datei nicht gefunden
- 3 = Verzeichnis nicht gefunden
- 18 = das Verzeichnis enthält keine DA-Dateien
- 101 = Laufwerk voll
- 150 = Diskette ist schreibgeschützt
- 152 = Floppy Drive nicht bereit

Schreibregister:

Wenn die PASE in dieses Register einen positiven Wert größer Null einträgt, ist dies die Aufforderung an das VIADUKT, Register von der PASE zu lesen und in eine Datei zu schreiben. Der Name der Datei ergibt sich aus dem Inhalt des Schreibregisters und der Dateierweiterung ".DA". Die Daten werden in diese Datei in einem Format geschrieben, das anschließend von der Lesefunktion oder vom Programm SYMPAS (Funktion Werte Lesen) verarbeitet werden kann.

Wenn alle Daten aufgezeichnet sind, setzt VIADUKT das Schreibregister auf den Wert 0. Falls ein Fehler während der

Aufzeichnung auftritt, wird das Schreibregister auf einen Wert kleiner 0 gesetzt. Befindet sich die DA-Datei nicht im aktuellen Verzeichnis, so kann VIADUKT das entsprechende Verzeichnis mit folgendem DOS-Befehl mitgeteilt werden:

```
SET DA_DIRECTORY=Pfad
```

Startregisternummer

Dieses Register enthält die Nummer des ersten Registers der PASE aus dem Daten gelesen werden sollen.

Endregisternummer

Dieses Register enthält die Nummer des letzten Registers der PASE aus dem Daten gelesen und aufgezeichnet werden sollen.

Beispiel:

Der Inhalt der Register 100 bis 140 soll auf Festplatte gespeichert werden. Wenn vom obigen Beispiel des VIADUKT-Aufrufs ausgegangen wird (/D200), muß das PASE-Programm zunächst die Register für die Start- und Endregisternummer besetzen:

```
LADE-REGISTER [ 202 mit 100 ]  
LADE-REGISTER [ 203 mit 140 ]
```

Anschließend kann der Aufzeichnungsvorgang durch Zuweisen des Dateinamens an das Schreibregister gestartet werden.

```
LADE-REGISTER [ 201 mit 3333 ]
```

Jetzt wird die Datei "3333.DA" auf der Festplatte erzeugt, die Werte der Register 100 bis 140 von der PASE gelesen und in die Datei geschrieben. Die Datei wird geschlossen und das Schreibregister 201 auf den Wert 0 gesetzt. Die PASE kann auf diese "Fertigmeldung" des VIADUKT warten z.B. mit den Zeilen

```
SOBALD
  REG 201
  <
  1
DANN
FALLS
  REG 201
  <
  0
DANN
  Fehler....
```

Weitere Funktionen können im Statusregister (PASE) über die Bits 1 und 2 codiert werden (siehe Kapitel "4. Reservierte PASE-Registerbereiche").

Statusregister (PASE)

- Bit 1 : 0 = der Inhalt des Lese- und Schreibregisters wird direkt als Dateiname verwendet (siehe Datentransferfunktion)
1 = der Inhalt des Lese- und Schreibregisters wird als Zeiger auf einen Dateinamen verwendet. (Enthält dieser Wildcards so wird ein Dateiauswahlfenster angeboten)
- Bit 2 : 0 = siehe Bit 2 = 1
1 = wurde eine DA-Datei über ein Dateiauswahlfenster aktiviert, so wird der gewählte Dateiname in das Textregister übernommen
- Bit 4 : 0 = Dateiname wird neu angelegt. Eine alte Datei mit gleichem Namen wird dabei überschrieben.
1 = Datenblock wird an die bereits bestehende Datei mit dem angegebenen Namen angehängt. Existiert noch keine solche Datei, wird diese zuerst angelegt.

7. DOS-Funktionen (nicht VIADUKT lite)

Diese Anwendung erlaubt es dem Bediener des Prozessvisualisierungssystems in der Maske Fenster zu öffnen und mit Hilfe hier definierter Befehlszeilen DOS-Funktionen auszuführen. Folgende DOS-Befehle beziehungsweise Mechanismen werden vom VIADUKT zur Verfügung gestellt:

- **Dir**
- **Copy**
- **Del**
- die Ausführung von **Batchdateien**
- das Aufrufen einer **DOS-Shell** (PC-Tastatur erforderlich)

Über Anweisungsdateien sind die gewünschten DOS-Funktionen vom Anwender in das VIADUKT zu integrieren. Diese Anweisungsdateien generieren zur Laufzeit durch Tastendruck des Bedieners Fenster, aus denen dann die gewünschten DOS-Funktionen gewählt werden können.

In den Anweisungsdateien definiert der Anwender durch Schlüsselwörter die DOS-Funktionen, welche zur Laufzeit dem Bediener in den angesprochenen Auswahlfenstern zur Verfügung stehen.

Beim Starten des VIADUKT sind diesem in einem Aufrufparameter die Anweisungsdatei und somit die darin definierten DOS-Funktionen anzugeben. Die Anweisungsdatei wurde vom Anwender vorher in einem beliebigen ASCII-Editor geschrieben.

Folgende **Schlüsselwörter** sind festgelegt:

1. **WinTitle** Titel des Hauptauswahlfensters (max. 36 Zeichen)

- 2. Userkey** Tasten (-Kombination) zur Aktivierung der Funktion.
Zulässige Tasten sind:
- o F1 - F12
 - o SHIFT-Taste
- 3. Lock** **Y** = die Taste zur Aktivierung der Funktion muß in der aktuellen VIADUKT-Maske als MENÜ-Taste definiert worden sein. In diesem Fall darf keine Kombination mit der SHIFT-Taste verwendet werden.
- N** = (Defaulteinstellung) die Taste zur Aktivierung der Funktion öffnet das in der Anweisungsliste definierte Auswahlfenster ohne den Inhalt der aktuellen VIADUKT-Maske zu berücksichtigen. Hier sollte die Kombination zwischen SHIFT- und Funktionstaste genutzt werden, um die nicht kombinierten Funktionstasten der MENÜ-Funktion zu überlassen.
- 4. Function** ausführbare DOS-Befehle und Funktionen sind:
- o Dir
 - o Copy
 - o Del
 - o Convert
 - o Batch
 - o Shell
- 4.a MenuTitle** hier kann eine Funktionsüberschrift angegeben werden, die stellvertretend für die eigentliche Funktionsdefinition in dem Auswahlfenster erscheint.
- 4.b SourcePath** hier wird das Quellverzeichnis der Funktion angegeben:
Quellverzeichnis + Name (Wildcards möglich)

- 4.c DestPath** hier werden das Zielverzeichnis der Funktion oder für eine BATCH-Funktion die Parameter angegeben:
Zielverzeichnis [*.EXT] oder Parameter (BATCH)
- 4.d AllText** werden Dateinamen mit Wildcards definiert, so werden durch diese in das Auswahlfenster integrierte Zeile alle Dateien mit gleichem Namen und gleicher Extension von der Funktion erfasst. Zur näheren Erläuterung siehe Beispiel.
- 4.e PostErase** steht nur der CONVERT-Funktion zur Verfügung. Die Quelldateien werden durch diese Definition nach der Konvertierung gelöscht.
- 5. Position** Diesem Schlüsselwort werden 4 Parameter hintenangestellt, welche der Reihenfolge nach folgende Bedeutung haben:
- X: Dieser Parameter definiert die X-Koordinate der linken, oberen Fensterecke. (Ist der Parameter 0 so wird die Defaulteinstellung aktiviert: Vermittlung der X-Achse)
 - Y: Dieser Parameter definiert die Y-Koordinate der linken, oberen Fensterecke. (Ist der Parameter 0 so wird die Defaulteinstellung aktiviert: Vermittlung der Y-Achse)
 - B: Definiert die Breite des Meldungsfensters (min. 15, max. 70).
 - H: Definiert die Höhe des Fensters (2...12).

Folgende **Vereinbarungen** sind beim Schreiben der Anweisungsdateien in einem ASCII-Editor einzuhalten:

- o zwischen Schlüsselwort und Schlüsselwortparameter muß ein

- Gleichheitszeichen stehen; daneben sind Leerzeichen zur optischen Gliederung erlaubt.
- o Kommentare sollten mit einem Semikolon eingeleitet werden.
 - o Zeilen mit Schlüsselwörtern dürfen keine Kommentare enthalten.
 - o Pfade müssen komplett mit Laufwerksbuchstaben angegeben werden.
 - o unvollständige oder fehlerhafte Schlüsselwortbeschreibungen führen dazu, daß die Funktion nicht in das Auswahlménü aufgenommen wird.
 - o Fehlermeldungen werden zur Laufzeit ausgegeben.

Folgender **Aufrufparameter** ist beim Start des VIADUKT anzugeben:

VIAGR /a /eDateiname.Ext

Anhand eines **Beispiels** sollen die Zusammenhänge erläutert werden:

Aufgabenstellung: es sollen alle Dateien mit der Extension EX1, die sich im Verzeichnis C:\TESTVER1 befinden in das Verzeichnis C:\TESTVER2 kopiert werden. Dabei soll die Dateierweiterung EX1 in EX2 umbenannt werden.

Der Anwender definiert jetzt in einer Anweisungsdatei wie das Auswahlfenster aussehen soll, welches der Bediener mit einer Funktionstaste öffnen kann um daraus die gewünschte DOS-Funktion auszuwählen.

Diese Anweisungsdatei kann in einem beliebigen ASCII-Editor erstellt werden. Zum Beispiel ist diese Datei folgendermaßen aufgebaut:

```
;Datei DOSFUNC1.DFC
;
WinTitle = Bitte wählen Sie
UserKey = F10 shift
;
;
Function = Copy
MenuTitle = kopiere C:\..\*.EX1 nach C:\..\*.EX2
SourcePath = c:\testver1\*.EX1
DestPath = c:\testver2\*.EX2
AllText = Alle Dateien
;
```

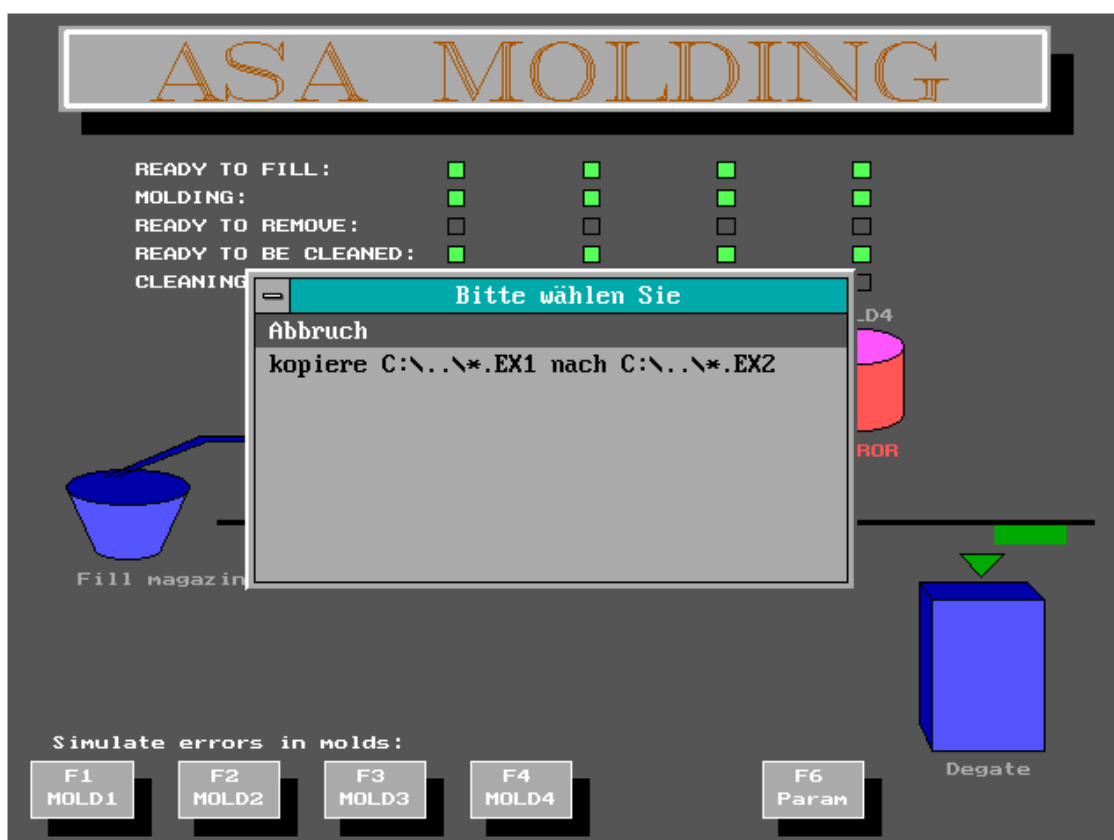
Diese Datei wird jetzt unter einem beliebigen Namen auf Festplatte gespeichert. In unserem Beispiel verwenden wir den Dateinamen DOSFUNC1.DFC.

Diese Datei muß sich entweder in gleichem Verzeichnis wie VIAGR.EXE befinden oder im Aufrufparameter zu dieser Anweisungsdatei muß der gesamte Pfad spezifiziert werden.

Jetzt wird das **VIADUKT** mit folgendem **Aufrufparameter gestartet**:

```
VIAGR /a /eDOSFUNC1.DFC
```

Drückt der Bediener jetzt die Tastenkombination SHIFT-F10 so erscheint über der aktuellen Maske folgendes Auswahlfenster:



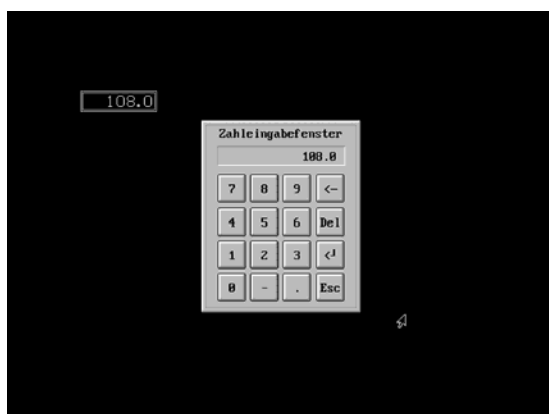
Aus dem Fenster kann der Bediener nun die gewünschte Funktion auswählen. In dem hier vorgeführten Beispiel kann zwischen ABBRUCH und der oben definierten Kopieraktion gewählt werden. Durch entsprechend umfangreiche Anweisungsdateien können dem Anwender beliebig viele DOS-Funktionen in einem Auswahlfenster zur Verfügung gestellt werden.

8. Eingabe von ZAHL- und TEXTVARIABLE-Objekten mit der Maus (Automatikbetrieb)

Im Automatikbetrieb können vom Bediener Werteingaben in die ZAHL-Objekte oder auch Texteingaben in die TEXTVARIABLE-Objekte vorgenommen werden. Die verschiedenen Eingabefelder werden nach einer intern festgelegten Reihenfolge mit den Cursorstasten aufgerufen, können modifiziert, diese Änderung dann quittiert (ENTER) oder der Werteingabevorgang abgebrochen werden (ESC).

Wird das VIADUKT auch während des Automatikbetriebes mit einer Maus (Trackball, etc.) ausgestattet, so kann die Eingabe in ZAHL- oder auch TEXTVARIABLE-Objekte mit diesen Zeigegeräten realisiert werden. Dazu muß der Bediener mit dem Zeigegerät auf den Zahlenwert oder den Text deuten, der geändert werden soll. Es öffnen sich entsprechend der Art des Eingabeobjektes eine numerische Eingabetastatur oder eine komplette PC-Tastatur, über die der Bediener mit Hilfe des Eingabegerätes alle notwendigen Eingaben bewerkstelligen kann.

Eingabe von **Zahlen** oder **Texten** mit den jeweiligen Eingabefenstern:



Einfach auf den zu ändernden Text oder die zu ändernde Zahl mit dem Zeigegerät deuten, und in dem sich öffnenden Fenster den Text respektive die Zahl editieren.

Beide Eingabefenster, sowohl das Zahleingabefenster für die Eingabe von Zahlen als auch das Tastaturfenster für die Eingabe von Texten können vom Anwender selbst erstellt und somit mit Zeichen belegt werden, die in den beiden vorgefertigten Eingabefenstern (auf VIADUKT-Diskette) nicht vorhanden sind.

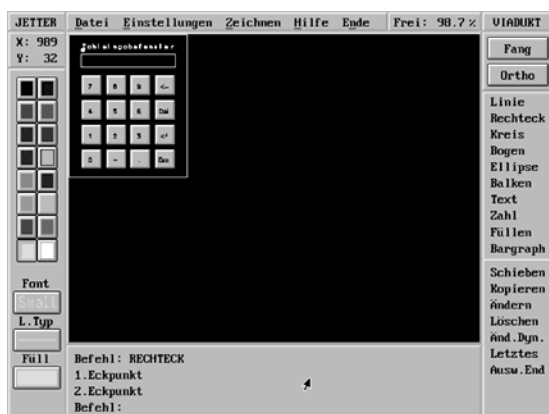
Deutet der Bediener des VIADUKT mit dem Zeigegerät auf ein ZAHL-Objekt so sucht VIADUKT nach einer Datei mit dem Namen INNUMBER.VIA. Findet das VIADUKT eine solche Datei wird diese in Form des oben abgebildeten Zahleingabefensters geöffnet.

Deutet der Bediener des VIADUKT mit dem Zeigegerät auf ein TEXTVARIABLE-Objekt so sucht VIADUKT nach einer Datei mit dem Namen INSTRING.VIA. Findet das VIADUKT eine solche Datei wird diese in Form des oben abgebildeten Texteingabefensters geöffnet.

Sowohl die Datei INNUMBER.VIA als auch die Datei INSTRING.VIA sind gewöhnliche VIADUKT-Masken, in denen jedoch nur bestimmte Objekte Verwendung finden dürfen.

Zur Basismaske für die beiden oben angeführten Eingabefenster werden diese Dateien durch ihre Dateinamen.

Die folgenden Abbildungen zeigen INNUMBER.VIA und INSTRING.VIA als Masken im Editor:



Einige Konventionen müssen eingehalten werden, damit die Masken als Eingabefenster akzeptiert werden beziehungsweise korrekt funktionieren. Außerdem muß die Maske für das Zahleingabefenster unter den Namen INNUMBER.VIA die Maske für das Texteingabefenster unter dem Namen INSTRING.VIA abgespeichert werden.

Für das Erstellen der Eingabemasken gilt:

- o es werden nur die Objekte RECHTECK, BUTTON und TEXT registriert.
- o mit dem Objekt **RECHTECK** kann sowohl die Größe des Dialogfensters als auch die Größe des Anzeigefensters angegeben werden.
 - o Größe des Dialogfensters: Hierzu muß ein Rechteck in roter Farbe definiert werden.
 - o Größe des gesamten Eingabefensters: Hierzu muß ein Rechteck in gelber Farbe definiert werden.Das Fenster wird im Automatikmodus zentriert, so daß die Position des Eingabefenster im Editorbereich nicht relevant ist.
- o mit dem Objekt **KNOPF** kann eine beliebige Anzahl von Eingabetasten definiert werden. Neben der Position und Größe des Buttons kann ein Beschriftungstext und ein Code definiert werden, der durch anklicken des entsprechenden Buttons (Taste) ausgewertet wird.
Dieser Code ist im Eingabefeld "Merkernummer" anzugeben. Der Code eines Button, der zusammen mit einem definierten CAPS-LOCK-Button "gedrückt" wird, ist im Feld "Offset" einzutragen.

Folgende Tastencodes werden unterstützt:

- o Steuerzeichen sind:
 - 8 = BACKSPACE
 - 13 = ENTER
 - 27 = ESCAPE
- o weiter gelten die ASCII-Zeichen 32 bis 255.

- o Codes über 255 werden als erweiterte Tastaturcodes gewertet:
 - 313** = CAPS-LOCK: wird ein Buchstabe zwischen 'a' und 'z' (Code 97 bis 122) angeklickt, während CAPS-LOCK aktiv ist, so wird der entsprechende Großbuchstabe eingetragen. **Ausnahme:** Ein Wert im Offsetfeld hat Priorität sofern ein solcher spezifiziert wurde.
 - 338** = DEL: löscht die komplette Eingabe und zeigt den ursprünglichen Wert wieder an.

- o mit dem Objekt **TEXT** können beliebige Texte innerhalb des Dialogfensters erstellt werden.

- o es wird empfohlen zur Definition solcher Fenster den Font "Normal" zu verwenden, um im Editorfenster in etwa dieselben Größenverhältnisse wie im Fenster des Automatikbetriebes zu gewährleisten.

9. PCX-Dateien

In das VIADUKT können PCX-Dateien eingebunden werden (S/W oder 16 Farben). Diese PCX-Dateien können als Hintergrundmasken dienen, über denen der Anwender weitere Elemente einfügen kann.

Ebenso können vom VIADUKT-Bildschirm (Vollansicht und Automatik) Hardcopies angefertigt werden, die als PCX-Dateien auf Festplatte gespeichert werden können, um anschließend zum Beispiel für Dokumentationszwecke Verwendung zu finden.

Mit der Taste **CTRL-P** werden Hardcopies von den Bildschirmen der Vollansicht respektive des Automatikmodus gestartet. Ein Fenster erscheint, in welchem die Hardcopy näher spezifiziert werden kann. Dann wird ein Fenster geöffnet, in dem der Dateiname der PCX-Datei angegeben werden kann unter welchem diese auf die Festplatte gespeichert wird.

Folgende Auswahl kann in dem Fenster "PCX-Datei" getroffen werden:

Vollbild	der gesamte Bildschirm wird kopiert.
Aktives Fenster	das aktive Fenster wird kopiert.
Ausschnitt	ein rechteckiger Ausschnitt, welcher mit der Maus definiert werden kann, wird kopiert.
Aktives Fenster (0/0)	das aktive Fenster wird kopiert, wobei die Koordinaten auf den Ursprung projiziert werden.
Ausschnitt (0/0)	ein rechteckiger Ausschnitt wird kopiert, wobei die Koordinaten auf den Ursprung projiziert wird.

In eine Maske eingefügt werden die PCX-Dateien mit der **INCLUDE**-Funktion im Pull-Down-Menü "Zeichnen". Hier kann der Anwender zwischen den Importformaten .VIA und .PCX wählen.

Bindet der Anwender eines der beiden Dateiformate über diese Funktion in eine Maske ein, so sind die Importdateien - also die .VIA oder .PCX Dateien - kein direkter Bestandteil der aktuellen Maske. In dieser befindet sich nur ein Includevermerk der auf die entsprechende, einzuschließende Datei hinweist. Damit erklärt sich auch der

Unterschied zwischen dem Laden ("Maske laden") und dem Includen einer **.VIA**-Datei (dieser Sachverhalt gilt nicht für PCX-Dateien, denn diese können ohnehin nur mit INCLUDE in die Maske integriert werden).

Eine in eine aktuelle Maske zusätzlich eingeladene ("Maske laden") weitere Maske wird zum integralen Bestandteil der Basismaske, ihre Daten werden dieser beigefügt und mit abgespeichert.

Wird eine zusätzliche Maske über INCLUDE eingeladen, so weist nur ein Includevermerk auf die Existenz der zusätzlichen Maske hin, ihre Daten verbleiben in der ursprünglichen Datei.

PCX-Dateien können in beliebigen handelsüblichen Zeichen- und Designprogrammen erstellt (Paintbrush, CorelDraw, etc.) und in das VIADUKT integriert werden.

Wird das Programm Paintbrush (Windows) verwendet so sollte die Palette VWINDOWS.COL in Paintbrush geladen werden, um ein Höchstmaß an Kompatibilität zwischen Paintbrush und VIADUKT-Farben zu gewährleisten.

Außerdem sind die Dateien vor dem Import in das VIADUKT mit dem Programm PCX16VIA.EXE zu konvertieren. Detaillierte Informationen können der Datei PCX16VIA.DOK entnommen werden.

10. Laufzeitfehlermeldungen in beliebigen Fremdsprachen

Fehlermeldungen, die VIADUKT zur Laufzeit im Automatikbetrieb ausgibt, können in jede beliebige Sprache übertragen werden. Dazu muß die Datei VIAMSG.MSG modifiziert werden. Hierzu werden die deutschsprachigen Fehlermeldungen mit den fremdsprachigen Pendants überschrieben, wobei auf die genaue Einhaltung des Zeilenschemas der Datei VIAMSG.MSG unbedingt zu achten ist.

Auch die Gesamtzeilenanzahl der Datei VIAMSG.MSG darf nicht verändert werden. Andernfalls wird die Datei ignoriert und es werden keine Fehlermeldungen ausgegeben.

11. Funktionserweiterung durch TSR-Programme (nicht VIADUKT lite)

Diese Option ermöglicht die Kommunikation zwischen VIADUKT und einem vom Anwender selbst geschriebenen (speicherresidenten) Programm. Da dies spezielle PC-Programmierkenntnisse voraussetzt, soll hier nicht näher darauf eingegangen werden.

Die im Lieferumfang enthaltene Datei **USERPROG.PAS** enthält alles Wissenswerte zu diesem Thema.

12. Spezifikation des Graphikadapters

Mit folgendem Schalter kann das VIADUKT an den Graphikadapter angepasst werden.

VIAGR /X <cc> oder <Dateiname>

Dieser Schalter dient der näheren Spezifikation des Graphikadapters des VIADUKT. Dabei kann einmal durch einen Zahlenwert <cc> die Farbe des Fadenkreuzes beziehungsweise des Eingaberahmens definiert werden, zum anderen können über die Angabe einer Definitionsdatei <Dateiname> weitgehende Definitionen zum Graphikadapter getroffen werden.

Farbdefinition des Eingaberahmens und des Fadenkreuzes mit VIAGR /Xcc. Dabei sind folgende Werte für 'cc' möglich:

0 = Schwarz	8 = Dunkelgrau
1 = Blau	9 = Hellblau
2 = Grün	10 = Hellgrün
3 = Türkis	11 = helles Türkis
4 = Rot	12 = Hellrot
5 = Fuchsinrot	13 = helles Fuchsinrot
6 = Braun	14 = Gelb
7 = Hellgrau	15 = Weiß

Defaulteinstellung: Hellgrau

Angaben in der Definitionsdatei '**Dateiname**':

Folgende Schlüsselwörter werden in einer Definitionsdatei unterstützt:

- **GrDriver**
- **GrMode**

Mögliche Werte für **GrDriver** und **GrMode** (eingerückte Zahlenwerte) sind:

```
3 = EGA (> 64 kB)
  0 = 640 * 200 mit 16 Farben
  1 = 640 * 350 mit 16 Farben
  3 = 640 * 350 mit  2 Farben
4 = EGA (64 kB)
  0 = 640 * 200 mit 16 Farben
  1 = 640 * 350 mit  4 Farben
  3 = 640 * 350 mit  2 Farben

9 = VGA
  0 = 640 * 200 mit 16 Farben
  1 = 640 * 350 mit 16 Farben
  2 = 640 * 480 mit 16 Farben
```

Beispiel: EGA-Karte mit Speicher >64kB in einer Auflösung von 640*350 Pixeln mit 16 Farben setzt folgende Definitionen voraus:

- GrDriver = 3
- GrMode = 1

Weitere Schlüsselwörter, die in der Definitionsdatei unterstützt werden, sind:

- **Xpixel**
- **Ypixel**

Anzahl Pixel des Monitors in X- und Y-Richtung, wobei Xpixel und Ypixel nicht die Auflösung haben müssen, die dem gewünschten Graphikmodus entspricht.

Beispiel:

- Xpixel = 320
- Ypixel = 256

- **ScrWidth**
- **ScrHeight**

Mit diesen Schlüsselwörtern wird die physikalische Höhe und Breite des Monitors angegeben aus deren Angabe sich das notwendige Höhen/Breiten-Verhältnis errechnen lässt.

Beispiel:

- ScrWidth = 960
- ScrHeight = 768

- **monochrome**

Hiermit kann die vom Graphikadapter unterstützte Bitplane angegeben werden. Folgende Werte können spezifiziert werden:

- 1 = blaue Bitplane
- 2 = grüne Bitplane
- 4 = rote Bitplane

Beispiel:

- monochrome = 2

Zwischen den Definitionen können Kommentarzeilen eingefügt werden, die mit einem Semikolon ';' einzuleiten sind.

Stichwortverzeichnis

<F>	15
<K>	15
AllText	79, 81
ALT-F5	15
ÄND.DYN.	42, 44
ÄNDERN	2, 24, 27, 34, 42, 44, 45
Anschluß an die PASE	1, 9, 10
Anschluß der Maus	1, 11
Anweisungsdateien	77, 79, 82
Aufrufparameter	2, 17, 52, 54-56, 61, 77, 80, 81
Ausgang	4, 24, 41, 46, 48, 67, 68
Ausgangs- und Eingangsfunktionen	67
AUSW.END	2, 42-45
AUTOEXEC.BAT	11, 12, 52
Automatik	26, 49, 60, 87
Automatikbetrieb	2, 11, 21, 29, 34, 49, 52, 60, 69, 70, 83, 89
Automatikmodus	12, 20, 26, 29, 35, 37, 42, 48, 52, 55, 60, 61, 74, 85, 87
Balken	1, 27, 32, 38, 42
Bargraph	1, 27, 34, 38, 41, 42, 44
Batch	78, 79
Batchdateien	77
Bibliothekselemente	16
Bild laden	17
Bild löschen	20, 42, 43
BILD-LÖSCH	59
Bitplane	93
Bogen	1, 27, 32, 33, 42
Button	21, 22, 27, 85
COM1	6, 9, 11, 26
Convert	78, 79
Copy	77, 78, 81
CorelDraw	88
DA-Datei	71, 76

DA-Dateien	74
Datei laden	17
Datentransferfunktion	2, 71, 73, 74, 76
Datum-Zeit	23, 42, 44
Del.....	68, 77, 78, 86
Demoversion.....	5, 52
DestPath	79, 81
Dir	77, 78
DOS-Funktionen	2, 5, 54, 77, 82
DOS-Kommandozeile	46, 52
DOS-Shell	17, 77
Druckerschnittstelle	6, 12
Dynamik	1, 2, 24, 27, 40, 42, 44, 48
Dynamische Objekte	1, 34
EGA	5, 6, 13, 29, 53, 92
Eingaberegister.....	69, 71
Eingang	41, 67
Eingangsfunktionen.....	67
Einstellungen sichern.....	18
Ellipse	1, 27, 33, 42
Endregisternummer	73, 75
Exit VIADUKT	17
Fadenkreuzpositionsanzeige	29
Fang	1, 15, 18, 19, 27, 30, 66
Farbdefinition	91
Festplatte	6, 12, 13, 73, 75, 81, 87
Fontgröße.....	18, 33, 46
Füll	28, 41, 48
FÜLL-Objekt.....	41
Füllen	1, 27, 34, 47
Füllmuster.....	19, 28, 31, 32, 42, 46
Function	78, 81
Funktionstasten.....	2, 6, 21, 53, 55, 57-60, 63, 78
Grafikadapter	53
Grafikkarte.....	5, 6, 13, 29
Graphikadapter	54, 91, 93
GrDriver.....	91, 92
GrMode	91, 92

Grundfunktionen der Funktionstasten.....	57
Hardcopies.....	87
Hardware-Lock.....	6, 12, 13
Hardwareanforderungen.....	1, 6
Hintergrundfarbe.....	20, 46
Include.....	24, 42-44, 87, 88
Info.....	17, 48
INNUMBER.VIA.....	84, 85
Installation der Software.....	1, 12
INSTRING.VIA.....	84, 85
Istmaskenregister.....	58, 60, 61, 69, 70
Knopf.....	21, 42, 44, 85
Kommandowerte.....	61
Kommandozeilenoption.....	57, 63
Kopieren.....	2, 12, 27, 42, 43
Kreis.....	1, 27, 32, 34, 41, 46, 47
Laufzeitfehlermeldungen.....	2, 89
Leseregister.....	73-75
LETZTES.....	2, 27, 28, 30, 43, 45
Linie.....	1, 27, 31
Linienart.....	31, 46
Lock.....	6, 12, 13, 78, 85, 86
LÖSCHEN.....	2, 16, 20, 27, 42-45, 65
Maskenanzeige.....	2, 61
Maskenoffsetregister.....	69, 70
Maus.....	1, 2, 5, 6, 11, 13, 21, 27, 28, 31-35, 38-40, 42, 83, 87
MENÜ.....	1, 15, 16, 18, 20, 21, 25-27, 33, 34, 40, 42-44, 47-49, 57-61, 71, 78, 87
Menü "Auto".....	1, 26
Menü "Datei".....	1, 16, 47
Menü "Einstellungen".....	1, 15, 18
Menü "Hilfe".....	1, 25
Menü "Zeichnen".....	1, 20, 27, 87
Menü-Funktion.....	57, 59, 61, 78
Menüfeld.....	2, 21, 58
MenuTitle.....	78, 81
Merker.....	21, 40, 41, 53, 55, 57, 60, 63, 67, 68, 71, 72
Merkerfunktionen.....	67

Monitorfunktion.....	2, 66
monochrome.....	6, 93
Neu zeichnen	24
Ortho	1, 15, 19, 27, 30, 31, 33
Paintbrush	88
Palette.....	18, 88
PASE-E.....	1, 9, 10, 53, 67, 68
PASE-J.....	1, 10, 53
PASE-Merkerbereiche.....	2, 72
PASE-Registerbereiche	2, 69, 76
PC-Tastatur	63, 77, 83
PCX-Dateien	2, 24, 87, 88
PostErase	79
Quittungsregister	69, 70
Rechteck	1, 27, 32, 46, 47, 64, 85
Registerbit	41
Registerfunktionen.....	67
RT_DIRECTORY = Pfad	68
Satzeditor	2, 49
Schieben	2, 15, 21, 27, 42, 43, 45, 58
Schreibregister	73-75
ScrHeight	93
ScrWidth	93
Sichern als.....	16
Sollmaskenregister.....	61, 69, 70
SourcePath	78, 81
Startregisternummer	73, 75
Statusregister (PASE).....	69, 71, 76
Statusregister (VIADUKT).....	69, 70
SYMPAS	10, 49, 68, 74
Textvariable	1, 2, 21, 34, 39, 42, 44, 70, 71, 83, 84
TEXTVARIABLE-Objekte.....	71, 83
TSR	2, 5, 54, 90
TSR-Programme	2, 54, 90
Updates.....	2, 56
UserKey	78, 81
VGA	5, 6, 13, 29, 53, 92
VIADUKT lite	2, 5, 54, 73, 77, 90

VIAGR.....	12, 13, 17, 46, 52-56, 73, 80, 81, 91
VIAMSG.MSG	89
Vollansicht.....	26, 48, 87
VWINDOWS.COL	88
WinTitle.....	77, 81
ZAHL- oder TEXTVARIABLE-Objekte	71
ZAHL-Objekt	35, 37, 48, 84
Zahleneingabe	2, 37, 50, 60, 64
Zeichenfarbe	21, 32, 46

VIADUKT

Meldungseditor

(Nicht VIADUKT lite)

Inhaltsverzeichnis

1. Übersicht	1
2. Starten des Meldungseditors	3
3. Die Editoroberfläche	4
4. Die Pulldown-Menüs	6
4.1 Menü „≡“	6
4.2 Menü „Datei“	8
4.3 Menü „Bearbeiten“	10
4.4 Menü „Suchen“	12
4.5 Menü „Ausgabe“	13
4.6 Menü „Keywords“	16
4.6.1 Direktive	16
4.6.2 Variable	21
4.6.3 Symbol	27
4.7 Menü „Optionen“	29
4.8 Menü „Fenster“	31
5. Dateien, Extensions, etc.	33
6. Schalter (Aufrufparameter)	35
6.1 Schalter im Meldungstext hinter der Direktive LOGFILE (optional)	35
6.2 Schalter beim Kommandozeilenaufruf von VIAGR (VIADUKT)	36
7. Fehlermeldungen	37
8. Beispiele	41
8.1 Beispiel 1	41
8.2 Beispiel 2: Ausgabe in Fenstern, auf den Drucker, in ein Logfile	43
8.3 Beispiel 3	46
8.4 Beispiel 4	48
8.5 Beispiel 5: Symboldefinitionen	49
8.6 Beispiel 6: @AUTODATE, @AUTOTIME	51
8.7 Beispiel 7	53
8.8 Beispiel 8: Schreiben in ein Logfile	55
8.9 Beispiel 9	57
8.10 Beispiel 10: Logfile-Optionen	59
8.11 Beispiel 11: Systemsymbole zur Logfilenamens-Definition	62
8.12 Beispiel 12: Ausgabe auf den Drucker, Textboxen	64

1. Überblick

09/93

Mit dem Meldungseditor zum VIADUKT können Meldungstexte beliebiger Art und Länge in Fenstern, auf einem Drucker oder in ein Logfile ausgegeben werden. Ein Logfile ist eine Datei, in welche Meldungen gespeichert werden; wobei neue Meldungen stets am Dateiende angehängt und mitgespeichert werden.

Bei den Meldungen kann es sich um Fehler- oder Statusmeldungen einer Maschine oder des gesteuerten Prozesses handeln, oder aber um Handlungsanweisungen an das Bedienungspersonal einer Maschine. Diese Handlungsanweisungen können sich kontextsensitiv auf den jeweiligen Fehlerfall oder einen besonderen Maschinenstatus (z.B. wechseln des Bohrkopfes wegen Verschleißes, Werkzeugwechsel allgemein, etc.) beziehen.

Die Fehlermeldungen werden auf dem VIADUKT (und unter dessen Regie auf Drucker oder Logfile) ausgegeben. Dies geschieht in einem Fenster, auf einem an das VIADUKT angeschlossenen Drucker oder auf die im VIADUKT integrierte Festplatte.

Auch sind Kombinationen der eben genannten Ausgaben möglich. So kann eine Meldung in einem Fenster auf dem VIADUKT-Bildschirm und gleichzeitig die gesamte Meldung, oder nur Teile von dieser, in einem Logfile abgelegt werden.

Das Schreiben und Austesten der Meldungen geschieht im Meldungseditor (VIATXTGR.EXE), der auf der VIADUKT-Diskette mitgeliefert wird. Um die Meldungstexte verwenden zu können, ist mindestens die VIADUKT-Version V1.30 (oder höher) notwendig.

Die Kontrolle der Fehlermeldungen, das heißt deren Initialisierung, Darstellung in Fenstern, Ausgabe auf dem Drucker, Abspeichern in Logfiles, oder aber Kombinationen hiervon wird von VIADUKT und PASE-Steuerung gemeinsam übernommen.

Wobei die PASE-Steuerung über Register definiert, welche Fehlermeldung der Prozess erfordert und das VIADUKT dann für Darstellung, Speicherung oder den Ausdruck der jeweiligen Meldung sorgt.

Auf der anderen Seite kann die PASE-Steuerung abfragen, ob die Meldung, welche zum Beispiel in einem Fenster auf dem VIADUKT gerade dargestellt wird, vom Bediener quittiert, und somit zur Kenntnis genommen wurde.

Über eine Vielzahl von zusätzlichen Funktionen, die über den Kommandozeilenaufruf des VIADUKT (VIAGR.EXE) als optionale Schalter angefügt werden können lassen sich weitere Möglichkeiten zur Prozessunterstützung realisieren.

Die Meldungstexte, welche in einem Fenster, auf dem Drucker oder auf ein Logfile ausgegeben werden sollen, werden im Meldungseditor geschrieben und ausgetestet.

Ausgetestet deshalb, weil in die Meldung an sich Variablen, zum Beispiel der Zustand eines Ausganges der Steuerung, integriert werden können. Dies setzt einige logische Fähigkeiten des gesamten Meldungsmanagements voraus, was das Austesten des Meldungstextes in einem Kompilierungslauf notwendig macht. Diese kompilierte Datei wird dann vom VIADUKT zur Entnahme der entsprechenden Fehlermeldung verwendet. Es werden je nach Ausgabemodus drei verschiedene kompilierte Meldungsdateien angelegt, welche sich durch unterschiedliche Extensions unterscheiden. Je nach Datei, unterschieden durch die jeweilige Extension, erfolgt dann die Ausgabe in ein Fenster, auf den Drucker oder in ein Logfile. Auch Kombinationen der Ausgabemöglichkeiten sind realisierbar. In einer solchen Meldungsdatei können sich beliebig viele Meldungen befinden, welche von der PASE-Steuerung nach den jeweiligen Erfordernissen aufgerufen werden können.

Die Meldungsdatei muß sich in dem Verzeichnis auf der Festplatte befinden, das bei dem Kommandozeilenaufruf von VIADUKT definiert wird (Namen mit Pfadangabe).

Nach dem Starten des Meldungseditors erscheint eine Editoroberfläche, realisiert in Fenstertechnik, mit Pull-Down-Menüs und Mausunterstützung. Hier kann jetzt mit Hilfe zahlreicher Editorfunktionen in den Pull-Down-Menüs die Gesamtheit der Meldungstexte geschrieben werden.

Nachdem dies geschehen ist, kann man mit zusätzlichen Funktionen, die über einen normalen Editor hinausgehen, die Meldungen auf Lauffähigkeit überprüfen. Während dieses Kompilierlaufes wird die Ausgabe, welche später vom VIADUKT aus auf den Bildschirm, den Drucker oder die Festplatte vorgenommen wird, simuliert. Die Fehlermeldungen erscheinen so als würden sie später in einem Fenster ausgegeben werden.

Nach dem Kompilieren der entsprechenden Meldungsdatei, kann der Meldungseditor verlassen werden. Das VIADUKT greift nun auf ein definiertes PASE-Register zu, in welchem die Nummer der darzustellenden Meldung gespeichert ist, und dort von der PASE-Steuerung modifiziert werden kann. Entsprechend der Nummer wird die Meldung dargestellt bis sie quittiert wurde oder eine Timeoutzeit abgelaufen ist. Die Quittierung durch den Anwender wiederum kann von der PASE-Steuerung zur Kenntnis genommen werden.

2. Starten des Meldungseditors

Der Meldungseditor wird mit folgender DOS-Kommandozeile gestartet:

```
C:\..\VIATXTGR
```

Nachdem Sie diese Zeile mit RETURN quittiert haben erscheint die Grundoberfläche des Meldungseditors.

3. Die Editoroberfläche

Die Bedieneroberfläche des Meldungseditors besteht aus folgenden Elementen:

- o Die Auswahlzeile der einzelnen Pull-Down-Menüs (erste Zeile)
- o Das oder die Fenster in denen Texte editiert, kompiliert, überprüft und korrigiert werden können
- o Die Auswahlzeile der wichtigsten Menüfunktionen (letzte Zeile)

Ein Fenster, von diesen können mehrere auf einmal auf der Bedieneroberfläche dargestellt werden, stellt folgende Funktionen zur Verfügung:

[■] Durch Anklicken dieses Symboles mit der linken Maustaste wird das Fenster geschlossen (Position: links oben)

Zahl Nummer des geöffneten Fensters (Position: rechts oben; in Verbindung mit der ALT-Taste wird das entsprechende Fenster aktiviert).

C:\... Aktueller Dateiname mit Pfadangabe (Position: Mitte oben)

[⇕] Mit diesem Symbol kann, die jeweils letzte Fenstergröße wiederhergestellt werden (Position: rechts oben)

* Das Erscheinen dieses Symboles zeigt an, daß die Datei geändert wurde. Fehlt das Symbol, wurden an der Datei keine Änderungen vorgenommen, d.h. die Datei auf der Festplatte entspricht der Datei im Fenster. (Position: links unten)

Zahl:Zahl Die erste Zahl steht für die Zeile, die zweite Zahl für die Spalte in der sich der Cursor befindet. (Position: links unten)

Funktionsbereiche:

In einem Fenster kann man 4 Funktionsbereiche nutzen.

- o Die erste Zeile (Doppellinie) kann mit der linken Maustaste gegriffen - linke Maustaste gedrückt halten und Maus bewegen - und auf diese Weise das gesamte Fenster verschoben werden.
- o Der Scrollbalken am rechten Fensterrand scrollt das Fenster von oben nach unten, der Scrollbalken am unteren Fensterrand scrollt das Fenster von links nach rechts und umgekehrt.
- o Die rechte untere Bildschirmecke (hier keine gesonderte Markierung) dient zur Größenverstellung eines Fensters. Die rechte untere Ecke des Fensters mit der linken Maustaste anklicken und Taste gedrückt halten. Jetzt kann die Größe des Fensters mit der Maus verändert werden. Nach der Größenveränderung des Fensters kann dieses durch Greifen in der ersten Zeile (Doppellinie) in seiner Gesamtheit verschoben werden - jetzt in der neuen Größe.

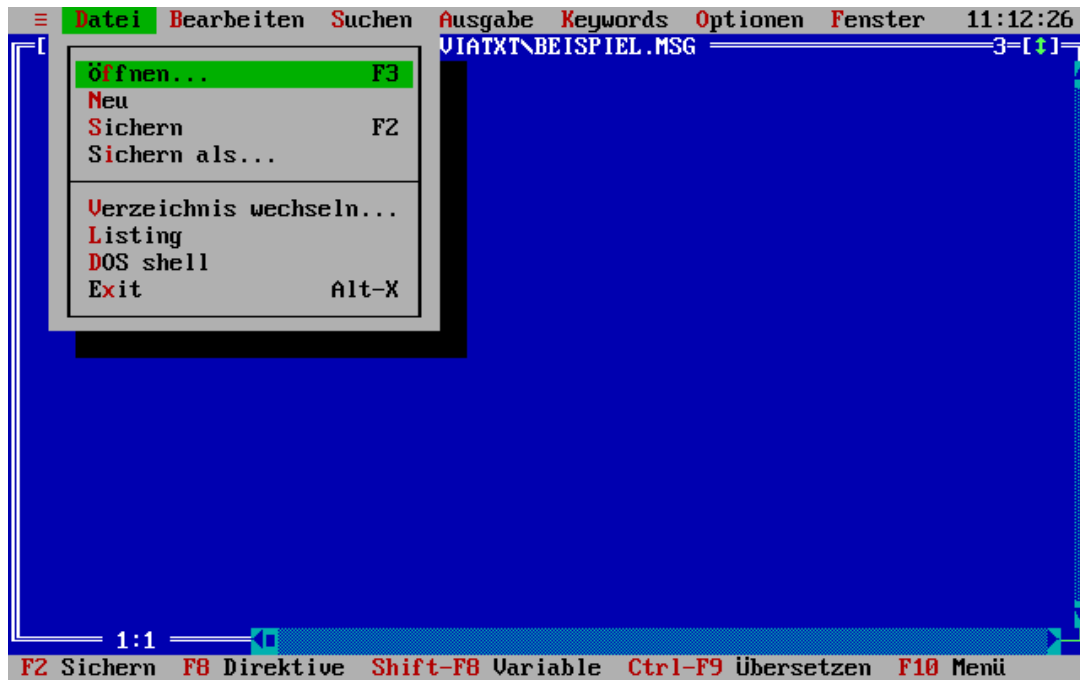
ASCII-Tabelle

In dieser Tabelle sind die ASCII-Codes des zur Verfügung stehenden Zeichensatzes aufgeführt (hexadezimal und dezimal). Eine mögliche Anwendung mag das Zeichnen von Textboxen für Meldungen sein - hierzu kann der notwendige ASCII-Code in der Tabelle nachgeschlagen werden und über die ALT-Tastenkombination in den Editor eingegeben werden.

Rechner

Taschenrechner

4.2 Menü "Datei"

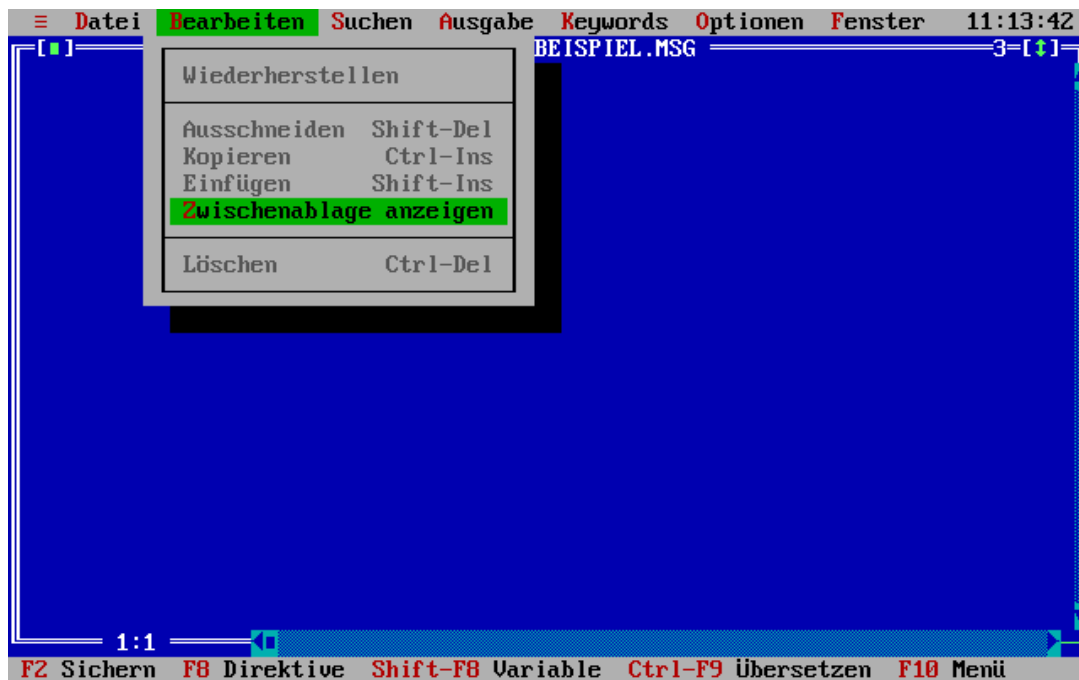


Öffnen

Mit dieser Funktion werden Dateien in den Editor geladen. Nach dem anklicken der Auswahlzeile erscheint ein Fenster in dem Dateiname und Pfad definiert werden können. Entweder kann der Name eingetippt werden, oder aber aus der angebotenen Dateiauswahl mit der Maus ausgewählt werden. Alle Quellcodedateien haben per Default die Extension ".MSG" (können aber auch anderweitig benannt werden). Kompilierte Objektcodedateien die im Fenster simuliert werden haben die Extension ".LST".

Neu	Hiermit wird ein neues Fenster geöffnet, welches zunächst den Namen "Untitled" trägt. Mit der Auswahlzeile "Sichern als" kann der Datei ein Name zugeordnet und die Datei auf Platte gespeichert werden.
Sichern	Diese Funktion sichert eine Datei unter ihrem Namen auf Festplatte. Der Namen samt Pfad der Datei wird in der oberen Fensterzeile eingeblendet.
Sichern als	Im Unterschied zur obigen Funktion "Sichern" wird hier der Datei ein neuer Name (evtl. Pfad) zugewiesen, unter welchem diese dann auf Festplatte gespeichert wird.
Verzeichnis wechseln	Über diese Funktion kann das Arbeitsverzeichnis gewechselt werden. Gleichzeitig wird Überblick durch die integrierte Baumfunktion gewährleistet.
Listing	Diese Funktion gibt das aktuelle Fenster auf dem Drucker aus.
DOS shell	Ausgang zu DOS. Nachdem Sie auf DOS-Ebene alle gewünschten Kommandos ausgeführt haben, kehren Sie mit EXIT wieder in den Meldungseditor zurück.
Exit	Mit dieser Funktion wird der Meldungseditor beendet, das Programm kehrt wieder zu DOS zurück.

4.3 Menü "Bearbeiten"

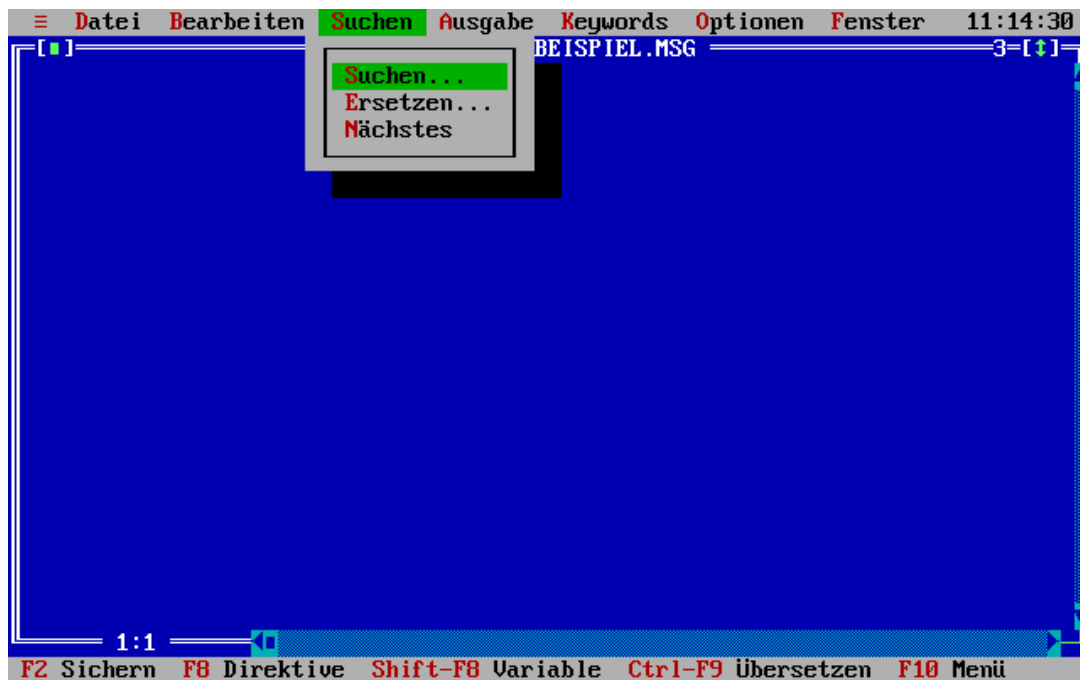


Wiederherstellen Diese Funktion restauriert die zuletzt gelöschte Zeile. Der Zwischenpuffer umfasst also nur eine Zeile die nach der Ausführung verschiedener Funktionen wiederhergestellt werden kann.

Ausschneiden Mit dieser Funktion und den Cursortasten lassen sich Blöcke aus dem Text ausschneiden. Der Text wurde zuvor entweder mit der gedrückten linken Maustaste oder der Kombination SHIFT-Cursortaste markiert und kann dann mit der Funktion "Ausschneiden" aus dem Text gelöscht und in die Zwischenablage übertragen werden.

Kopieren	Diese Funktion kopiert den markierten Text in die Zwischenablage - der Text bleibt also im Gegensatz zur obigen Funktion bestehen.
Einfügen	Der Inhalt der Zwischenablage wird an der aktuellen Cursorposition eingefügt.
Zwischenablage anzeigen	Der Inhalt der Zwischenablage wird angezeigt.
Löschen	Der markierte Text wird gelöscht.

4.4. Menü "Suchen"



Suchen

Nach dem Aktivieren der Auswahlzeile erscheint ein Fenster in dem der Suchbegriff und einige Suchkriterien definiert werden können. Außerdem kann ein weiteres Fenster geöffnet werden, das eine Auswahl an Suchbegriffen anbietet, die schon definiert wurden.

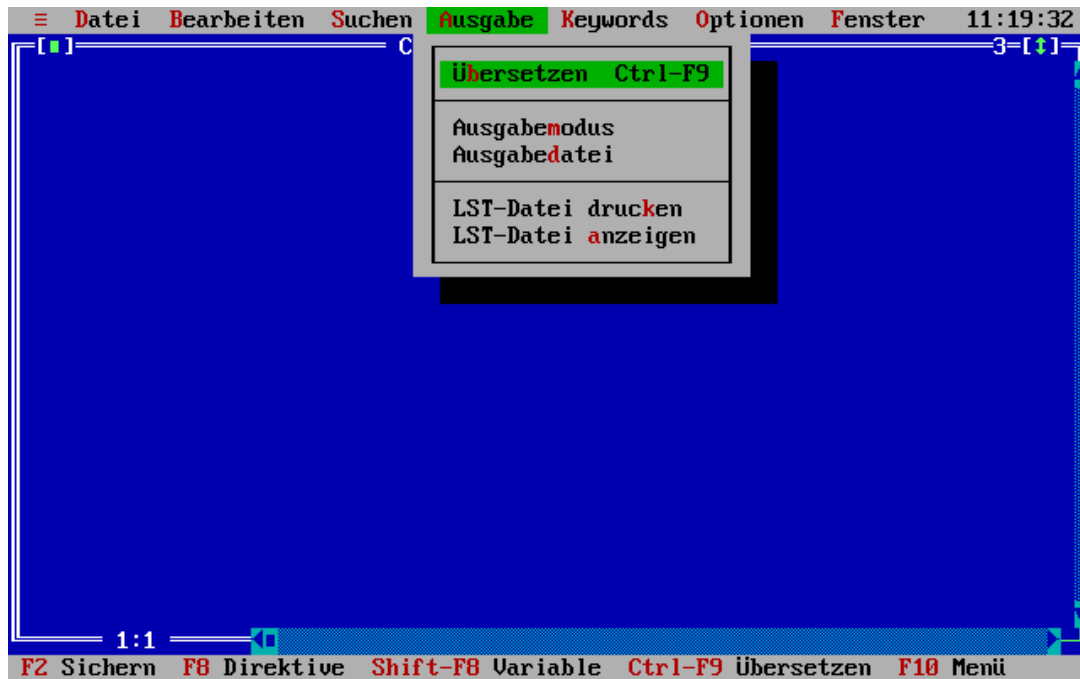
Ersetzen

Es erscheint ein Fenster, in welchem der Suchbegriff, der Begriff durch welchen dieser ersetzt werden soll und verschiedene Such- und Ersetzkriterien definiert werden können.

Nächstes

Die letzte aktivierte Funktion ("Ersetzen" oder "Suchen") wird mit den zuletzt vorgenommenen Definitionen wiederholt.

4.5 Menü "Ausgabe"



Übersetzen Mit dieser Funktion wird die Kompilierung der Meldung gestartet. Auf Fehler in dem Meldungstext wird mit Hilfe roter Fehlerbalken hingewiesen. Sobald der Meldungstext fehlerfrei ist startet der Kompiler und stellt die Meldungen in simulierten Fenstern dar. Dieses Fenster mit der Extension ".LST" sollte nach erfolgter Begutachtung wieder geschlossen werden (z.B. mit ALT-F3).

Ausgabemodus In dieser Funktion wird definiert, auf welchem Ausgabemedium die Fehlermeldung erscheinen soll - im Fenster, auf dem Drucker oder in einem Logfile.

Entsprechend dieser Definition werden vom Kompiler Objectcodedateien mit den Extensions ".W", ".H" oder ".L" generiert.

Desweiteren lassen sich folgende Optionen vor der Kompilierung in dem Ausgabemodusfenster einstellen.

- o es kann festgelegt werden ob die Indexangaben im .LST-Fenster angezeigt werden.
- o es kann angekreuzt werden ob die Direktiven COLOR und DISPLAYTIME zusammen mit ihren entsprechenden Parameterangaben beim Simulationslauf zusätzlich zu den resultierenden Meldungsfenstern erscheinen sollen.
- o es kann definiert werden, ob die LST-Datei angezeigt werden soll.

Durch die Angabe der Landeskennzahl werden die spezifischen Darstellungskriterien bei den Variablen DATE und TIME berücksichtigt.

Um diese Funktion nutzen zu können muß in der CONFIG.SYS folgende Zeile erscheinen:

```
INSTALL = C:\DOS\NLSFUNC.EXE C:\DOS\COUNTRY.SYS
```

Ausgabedatei

In diesem Fenster können Name und Pfad der Ausgabedatei modifiziert werden.

Im wesentlichen werden vom Texteditor drei Dateien bearbeitet bzw. erarbeitet, welche sich durch verschiedene Extensions unterscheiden:

- Name.MSG: Sourcecode Meldungsdatei
- Name.LST: Testdatei zur Meldungsfenstersimulation
- Name.W oder .H oder .L: Objectcode **Ausgabedatei**;

Diese Datei wird vom VIADUKT zur Darstellung der Fehlermeldungen verwendet. Die anderen beiden Dateien sind Arbeitsdateien.

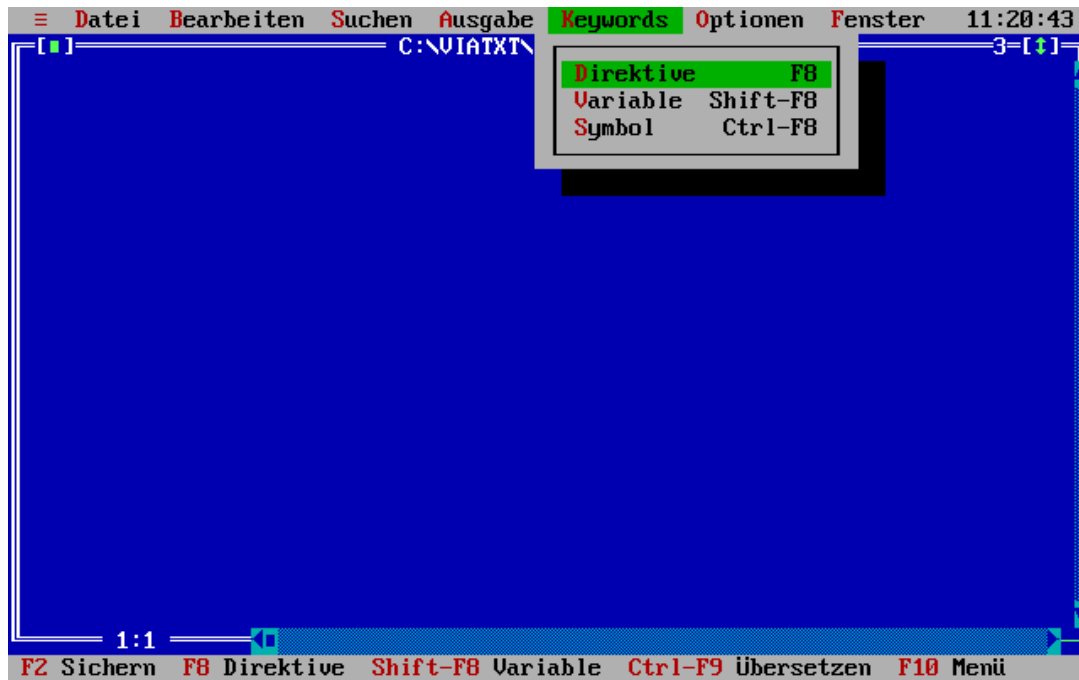
LST-Datei
drucken

Mit dieser Funktion werden die Dateien zur Fenstersimulation, welche nach einem Kompilierlauf angelegt werden, ausgedruckt.

LST-Datei anzeigen

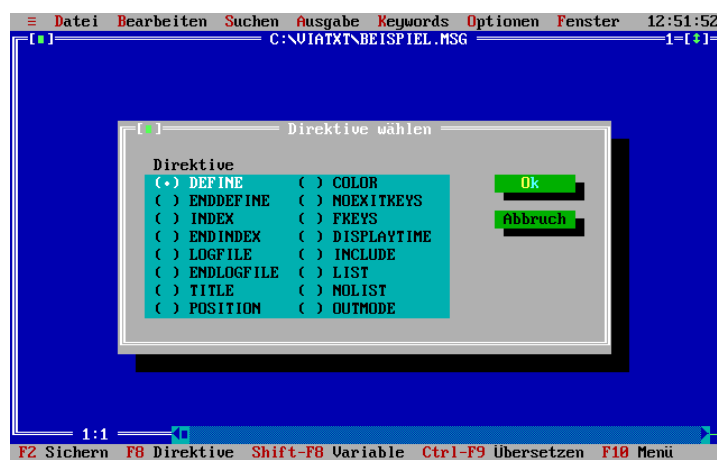
Mit dieser Funktion werden die Dateien zur Fenstersimulation, welche nach einem Kompilierlauf angelegt werden, auf dem Bildschirm dargestellt.

4.6 Menü "Keywords"



4.6.1 Direktive

Nach dem Aktivieren dieser Auswahlzeile erscheint folgendes Pop-Up-Window:



DEFINE	Diese Direktive markiert den Beginn eines Definitionsblockes. In einem Definitionsblock können Definitionen von Variablen, Dateinamen, etc. vorgenommen werden. (Darstellung siehe Beispiele in Kapitel 8)
ENDDEFINE	Diese Direktive markiert das Ende eines Definitionsblockes.
INDEX	Diese Direktive steht am Anfang des jeweiligen Meldungstextes. Nach dem Keyword folgt eine Zahl, welche die Identifikationsnummer für den späteren Aufruf der jeweiligen Meldung darstellt. Diese Zahl wird zur Laufzeit durch die PASE-Steuerung in einem Register definiert; das VIADUKT fragt dieses Register periodisch ab, und stellt die entsprechenden Meldungen auf dem jeweiligen Ausgabemedium (Fenster, Drucker, Logfile) dar.
ENDINDEX	An dieser Direktive endet der Text einer Meldung.
LOGFILE	Text, der dieser Direktive folgt, wird sowohl in einem Fenster dargestellt, als auch auf ein Logfile ausgegeben. Sinn dieser Möglichkeit ist es die Ausgabe in ein Logfile auf die wichtigsten Fakten zu beschränken, in einem Fenster dem Anwender jedoch umfassende, weiterführende Informationen zugänglich zu machen.
ENDLOGFILE	Diese Direktive beendet den Text, der in das Logfile geschrieben werden wird.

TITLE Dieser Direktive kann als Parameter eine Zeichenkette in Anführungsstrichen beigegeben werden. Dieser String erscheint in dem jeweiligen Meldungsfenster als Fenstertitelzeile.

POSITION Dieser Direktive werden 4 Parameter hintenangestellt, welche der Reihenfolge nach folgende Bedeutung haben:

X: Dieser Parameter definiert die X-Koordinate der linken, oberen Fensterecke. (Ist der Parameter 0 so wird die Defaulteinstellung aktiviert: Vermittlung der X-Achse)

Y: Dieser Parameter definiert die Y-Koordinate der linken, oberen Fensterecke. (Ist der Parameter 0 so wird die Defaulteinstellung aktiviert: Vermittlung der Y-Achse)

B: Definiert die Breite des Meldungsfensters (min. 15, max. 70).

H: Definiert die Höhe des Fensters (2...12).

COLOR Dieser Direktive werden 4 Parameter hintenangestellt, welche der Reihenfolge nach folgende Bedeutung haben:

P1: Farbe Textvordergrund

P2: Farbe Texthintergrund

P3: Farbe Überschriftvordergrund

P4: Farbe Überschrift hintergrund

Folgende Farben sind den Parameterwerten zugeordnet:

0	schwarz	1	blau	2	grün
3	türkis	4	rot	5	fuchsinrot
6	braun	7	hellgrau	8	dunkelgrau
9	hellblau	10	hellgrün	11	helles türkis
12	hellrot	13	helles fuchsinrot	14	gelb
15	weiß				

- NOEXITKEYS** Durch diese Direktive wird verhindert, daß ein Meldungsfenster durch die Tasten RETURN oder ESC abgebrochen werden kann. Somit muß das Fenster von der PASE-Steuerung oder durch die Timeoutfunktion wieder zurückgenommen werden.
- FKEYS** Die Parameter in der Form (Fx = Merker-Nr., [...]) weisen den Funktionstasten des VIADUKT Merkernelnummern zu, welche während der Darstellung eines Meldungsfensters aktiviert werden können.
Die Zuordnung der Funktionstasten zu den Merkernelnummern während der Darstellung eines Meldungsfensters ist nicht identisch mit der Zuordnung der Funktionstasten im Betrieb ohne Meldungsfenster. D.h. während der Darstellung eines Fensters können den Funktionstasten des VIADUKT andere Funktionen zugewiesen werden, definiert durch die Merkerzuordnung im Parameter der Direktive FKEYS.
- DISPLAYTIME** Der Parameter dieser Direktive definiert die Timeoutzeit des Meldungsfensters. Das heißt, nach dem Ablauf dieser Zeit (Parameterwert in Einheit Sekunden) wird das Fenster wieder vom VIADUKT-Bildschirm genommen. Die Meldung geht in diesem Fall verloren.
- INCLUDE** Dieser Direktive wird der Name der Datei hintenangestellt, die als Includedatei Verwendung finden soll. Eine Includedatei ist vom Aufbau mit einer Meldungsdatei identisch was bedeutet, daß jede Meldungsdatei als Includedatei eingebunden werden kann.
Die Schachtelung von Includedateien ist nicht möglich.

- LIST** Soll beim Kompilierungslauf nicht die gesamte Meldungsdatei (.MSG) in einem Fenstersimulationslauf (.LST-Datei) dargestellt werden sondern nur Teile von dieser, so kann mit LIST und NOLIST der Bereich, der dargestellt werden soll, definiert werden. Die Defaulteinstellung ist LIST. D.h. mit NOLIST muß der Auszugrenzende Teil eingeleitet werden, um dann mit List den Beginn des Teiles zu definieren der im Simulationslauf wieder darzustellen ist.
- NOLIST** Siehe Direktive LIST.
- OUTMODE** Diese Direktive sieht drei Parameter vor:
- o Window
 - o Printer
 - o Logfile
- Mit dieser Direktive und dem entsprechenden Parameter kann am Anfang einer Meldungsdatei definiert werden wo die Meldungen, welche in dieser Datei definiert werden, erscheinen sollen - in einem Meldungsfenster, auf dem Drucker oder in einem Logfile.
- Mit dieser Angabe zu Beginn einer Meldungsdatei kann die Angabe des Ausgabemodus im Pull-Down-Menü "Ausgabe", Auswahlzeile "Ausgabemodus" ersetzt werden. Die Definition in der Meldungsdatei hat Priorität.

4.6.2 Variable

Nach dem Aktivieren dieser Auswahlzeile erscheint folgendes Pop-Up-Window.



REGISTER

Mit dieser Systemvariablen können Inhalte beliebiger Register in die Meldung integriert werden. Nach dem Aktivieren der Auswahlzeile erscheint ein weiteres Pop-Up-Window, in welchem folgende Parameter definiert werden:

- o Gerät: Hier befindet sich bis auf weiteres die Zahl 0.
- o Nummer: Hier wird die Registernummer definiert.
- o Offset: Dieser Parameter wird zur oben angegebenen Registernummer addiert, wenn diese indirekt adressiert wurde.
- o Vorkommastellen: Anzahl der Vorkommastellen.
- o Nachkommastellen: Anzahl der Nachkommastellen.

- o Anzeigefaktor: Dieser Faktor wird mit dem Registerinhalt multipliziert, bevor dieser als Variable in dem Meldungstext erscheint.
- o Anzeigeoffset: Dieser Wert wird zu dem Registerinhalt addiert, bevor dieser als Variable in dem Meldungstext erscheint.

Die beiden letzten Parameter dienen der Einheitenumrechnung der jeweiligen Registerinhalte.

MERKER

Mit dieser Systemvariablen können Zustände beliebiger Merker in die Meldung integriert werden. Nach dem Aktivieren der Auswahlzeile erscheint ein weiteres Pop-Up-Window, in welchem folgende Parameter definiert werden:

- o Gerät: Hier befindet sich bis auf weiteres die Zahl 0.
- o Nummer: Hier wird die Merckernummer definiert.
- o Offset: Dieser Parameter wird zur oben angegebenen Merckernummer addiert, wenn diese indirekt adressiert wurde.
- o Feldlänge: Dieser Parameter legt die Darstellungsbreite der unten angegebenen Statustexte fest. Dabei kann zwischen drei Darstellungsarten unterschieden werden:
 1. Feldlänge > 0: linksbündige Anzeige, wobei die Differenz zwischen der angegebenen Feldlänge und der Anzahl gelesener Zeichen mit Leerzeichen aufgefüllt wird.
 2. Feldlänge = 0: wie 1. nur ohne Auffüllung mit Leerzeichen
 3. Feldlänge < 0: rechtsbündig
- o Text 0: Dieser Text erscheint in der Meldung, wenn der Status des Merkers 0 ist.

- o Text 1: Dieser Text erscheint in der Meldung, wenn der Status des Merkers 1 ist.

EINGANG

Mit dieser Systemvariablen können Zustände beliebiger Eingänge in die Meldung integriert werden. Nach dem Aktivieren der Auswahlzeile erscheint ein weiteres Pop-Up-Window, in welchem folgende Parameter definiert werden:

- o Gerät: Hier befindet sich bis auf weiteres die Zahl 0.
- o Nummer: Hier wird die Eingangsnummer definiert.
- o Offset: Dieser Parameter wird zur oben angegebenen Eingangsnummer addiert, wenn diese indirekt adressiert wurde.
- o Feldlänge: Dieser Parameter legt die Darstellungsbreite der unten angegebenen Statustexte fest. Dabei kann zwischen drei Darstellungsarten unterschieden werden:
 1. Feldlänge > 0: linksbündige Anzeige, wobei die Differenz zwischen der angegebenen Feldlänge und der Anzahl gelesener Zeichen mit Leerzeichen aufgefüllt wird.
 2. Feldlänge = 0: wie 1. nur ohne Auffüllung mit Leerzeichen
 3. Feldlänge < 0: rechtsbündig
- o Text 0: Dieser Text erscheint in der Meldung, wenn der Status des Eingangs 0 ist.
- o Text 1: Dieser Text erscheint in der Meldung, wenn der Status des Eingangs 1 ist.

AUSGANG Mit dieser Systemvariablen können Zustände beliebiger Ausgänge in die Meldung integriert werden. Nach dem Aktivieren der Auswahlzeile erscheint ein weiteres Pop-Up-Window, in welchem folgende Parameter definiert werden:

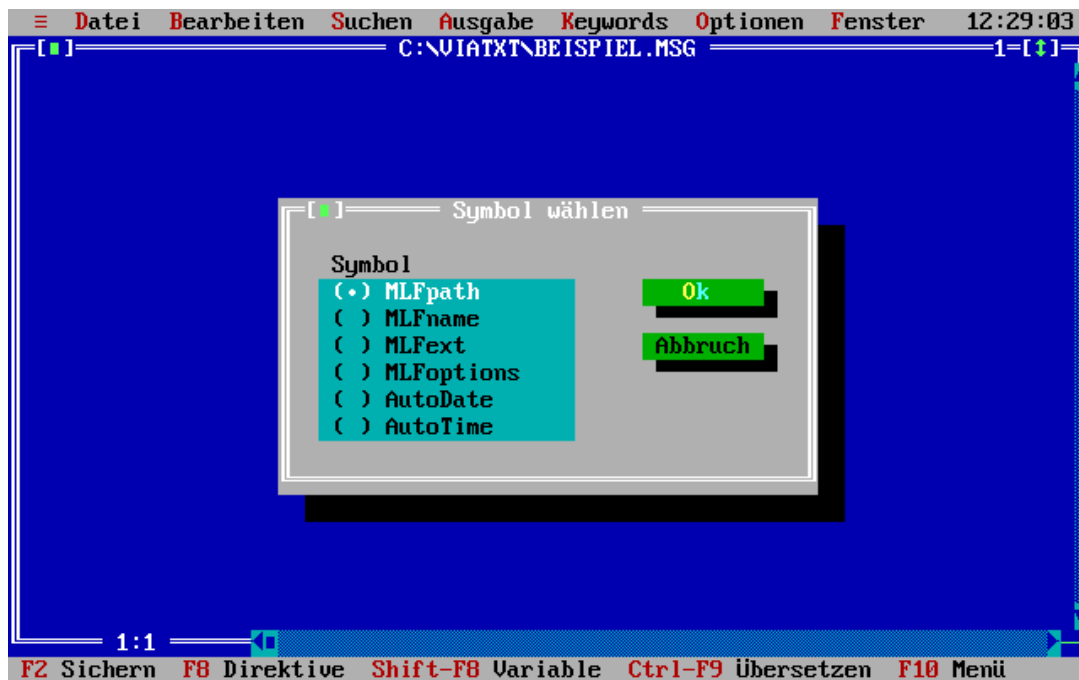
- o Gerät: Hier befindet sich bis auf weiteres die Zahl 0.
- o Nummer: Hier wird die Ausgangsnummer definiert.
- o Offset: Dieser Parameter wird zur oben angegebenen Ausgangsnummer addiert, wenn diese indirekt adressiert wurde.
- o Feldlänge: Dieser Parameter legt die Darstellungsbreite der unten angegebenen Statustexte fest. Dabei kann zwischen drei Darstellungsarten unterschieden werden:
 1. Feldlänge > 0: linksbündige Anzeige, wobei die Differenz zwischen der angegebenen Feldlänge und der Anzahl gelesener Zeichen mit Leerzeichen aufgefüllt wird.
 2. Feldlänge = 0: wie 1. nur ohne Auffüllung mit Leerzeichen
 3. Feldlänge < 0: rechtsbündig
- o Text 0: Dieser Text erscheint in der Meldung, wenn der Status des Ausganges 0 ist.
- o Text 1: Dieser Text erscheint in der Meldung, wenn der Status des Ausganges 1 ist.

- REGISTERBIT** Mit dieser Systemvariablen können Zustände beliebiger Registerbits in die Meldung integriert werden. Nach dem Aktivieren der Auswahlzeile erscheint ein weiteres Pop-Up-Window, in welchem folgende Parameter definiert werden:
- o Gerät: Hier befindet sich bis auf weiteres die Zahl 0.
 - o Nummer: Hier wird die Registernummer definiert.
 - o Offset: Dieser Parameter wird zur oben angegebenen Registernummer addiert, wenn diese indirekt adressiert wurde.
 - o Bitnummer: Hier wird die Bitnummer des oben genannten Registers bestimmt.
 - o Feldlänge: Dieser Parameter legt die Darstellungsbreite der unten angegebenen Statustexte fest. Dabei kann zwischen drei Darstellungsarten unterschieden werden:
 1. Feldlänge > 0: linksbündige Anzeige, wobei die Differenz zwischen der angegebenen Feldlänge und der Anzahl gelesener Zeichen mit Leerzeichen aufgefüllt wird.
 2. Feldlänge = 0: wie 1. nur ohne Auffüllung mit Leerzeichen
 3. Feldlänge < 0: rechtsbündig
 - o Text 0: Dieser Text erscheint in der Meldung, wenn der Status des Registerbits 0 ist.
 - o Text 1: Dieser Text erscheint in der Meldung, wenn der Status des Registerbits 1 ist.

- TEXT** Mit dieser Systemvariablen können beliebige Textsequenzen in die Meldung aufgenommen werden. Die Texte sind als ASCII-Zeichen in aufeinanderfolgenden Registern gespeichert und können durch die Angabe des Anfangsregisters unterschieden werden:
- o Gerät: Hier befindet sich bis auf weiteres die Zahl 0.
 - o Nummer: Hier wird die Registernummer definiert, ab der die Textsequenz in Form von ASCII-Zeichen gespeichert wird.
 - o Offset: Dieser Parameter wird zur oben angegebenen Registernummer addiert, wenn diese indirekt adressiert wurde.
 - o Feldlänge: Dieser Parameter legt die Darstellungsbreite der variablen Textsequenz fest. Dabei kann zwischen drei Darstellungsarten unterschieden werden:
 1. Feldlänge > 0: linksbündige Anzeige, wobei die Differenz zwischen der angegebenen Feldlänge und der Anzahl gelesener Zeichen mit Leerzeichen aufgefüllt wird.
 2. Feldlänge = 0: wie 1. nur ohne Auffüllung mit Leerzeichen
 3. Feldlänge < 0: rechtsbündig
- ZEIT** Diese Systemvariable gibt in der Meldung die aktuelle Zeit an.
- DATUM** Diese Sytemvariable gibt in der Meldung das aktuelle Datum an.
- DATEIDATUM** Diese Systemvariable generiert Dateinamen, welche sich aus dem aktuellen Datum zusammensetzen (z.B. 921001 für 01.10.1992). Diese Variable kann Verwendung finden um die Namensgebung von Logfiles sinnvoll zu automatisieren.

4.6.3 SYMBOL

Nach dem Aktivieren dieser Auswahlzeile erscheint folgendes Pop-Up-Window.



MLFpath

Dieses Symbol dient zur Definition des Pfades eines Logfiles. Die Defaulteinstellung ist der aktuelle Pfad. Eine Neudefinition dieser Pfad-angabe im Defineteil des Meldungstextes ist möglich.

MLFname

Dieses Symbol dient zur Definition des Dateinamens eines Logfiles. Die Defaulteinstellung ist ein Dateiname der sich aus dem aktuellen Datum zusammensetzt (921001). Eine Neudefinition dieser Namensangabe im Defineteil des Meldungstextes ist möglich.

MLFext

Dieses Symbol dient zur Definition der Extension eines Logfiles. Die Defaulteinstellung ist ".LOG".

Eine Neudefinition dieser Extensionangabe ist möglich.

MLFoptions

Mit diesem Symbol können im Defineteil eines Meldungstextes Optionen festgelegt werden, die bei dem Bearbeiten eines Logfiles zusätzlich ausgeführt werden:

/Rxxxx: Nummer des Registers, dessen Inhalt als Zeichenkette interpretiert und zwischen den Logfilenamen und die Extension eingefügt wird (Registerinhalt = 1234):
C:\Verzeichnis\Dateiname1234.ext

/O: das Logfile wird neu geöffnet, der alte Inhalt wird neu überschrieben.

/A: die Meldung wird an das existierende Logfile angehängt (Defaulteinstellung).

/E: das Logfile wird gelöscht.

/P: das Logfile wird auf dem Drucker ausgegeben.

/V [xxxx]: das Logfile wird in einem Dateianzeigefenster ausgegeben. Optional kann zusätzlich ein Register angegeben werden, in welchem die Zeilennummer definiert wird, ab der das Logfile ausgegeben wird.

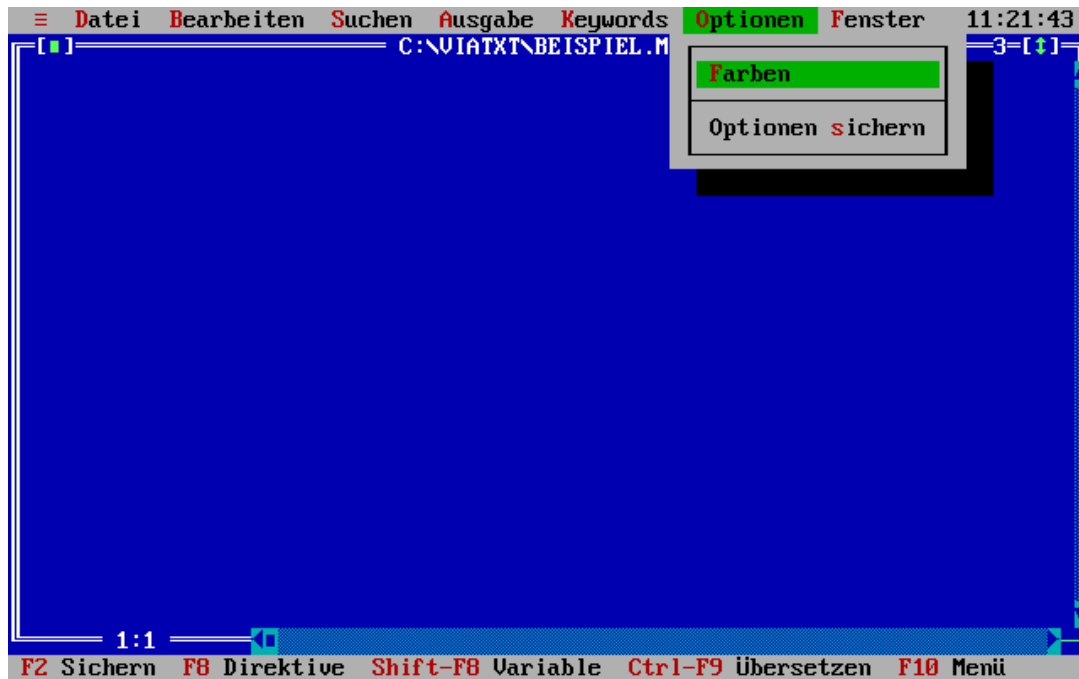
AUTODATE

Dieses Symbol kann im Defineteil eines Meldungstextes ein- oder ausgeschaltet werden. Ist die Funktion aktiviert, wird an den Anfang jeder Meldung automatisch das aktuelle Datum geschrieben.

AUTOTIME

Dieses Symbol kann im Defineteil eines Meldungstextes ein- oder ausgeschaltet werden. Ist die Funktion aktiviert, wird an den Anfang jeder Meldung automatisch die aktuelle Zeit gesetzt.

4.7 Menü "Optionen"



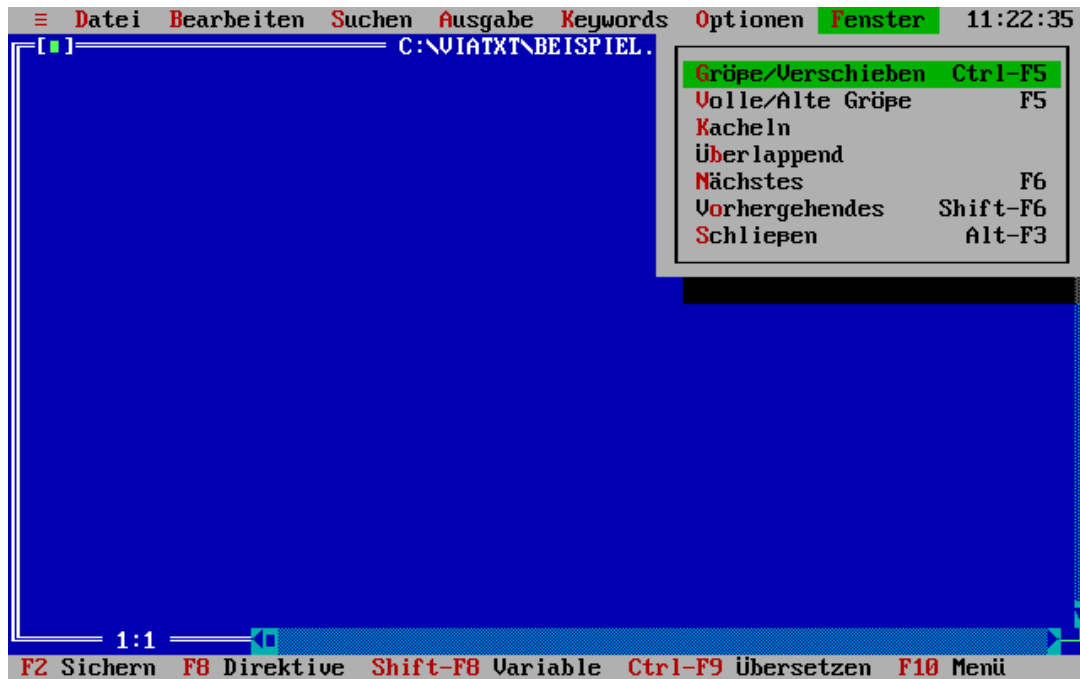
FARBEN

In dieser Auswahlzeile können die Farbeinstellungen für den gesamten Meldungseditor getroffen werden. Nach dem Bestätigen der Auswahlzeile erscheint ein Fenster, in dem wiederum 4 Unterfenster angewählt werden können. Mit der TAB-Taste (oder der Maus) werden die Fenster "Gruppe", "Detail", "Vordergrund" und "Hintergrund" der Reihe nach aktiviert. Mit der SHIFT-TAB-Tastenkombination werden die Fenster in umgekehrter Reihenfolge aktiviert. In den einzelnen Fenstern bewegt man sich mit den Cursor-tasten oder der Maus. Für jeden Punkt des Fensters "Gruppe" finden sich im Fenster "Detail" ein oder mehrere Unterpunkte, welchen man in den Fenstern "Vordergrund" und "Hintergrund" eine entsprechende Farbe zuweisen kann.

Mit der ESC-Taste wird die Farbeinstellung abgebrochen, mit der Taste RETURN quittiert. Unterhalb des Fensters "Hintergrund" befindet sich ein Test-Text, an welchem die Farbeinstellungen exemplarisch dargestellt werden.

OPTIONEN SICHERN Mit dieser Funktion werden die im obigen Punkt vorgenommenen Farbeinstellungen gesichert.

4.8 Menü "Fenster"



GRÖßE/ VERSCHIEBEN

Nach Bestätigung dieser Auswahlzeile mit RETURN läßt sich die Position des Fensters mit den Cursor-Tasten verschieben. Mit den SHIFT-Cursortasten läßt sich die Größe des Fensters verändern. Mit der RETURN oder der ESC-Taste wird diese Funktion abgebrochen. (Siehe auch Kapitel "3. Die Editoroberfläche")

VOLLE / ALTE GRÖßE

Mit dieser Funktion kann zwischen der vollen Fenstergröße und der letzten Einstellung hin und her geschaltet werden.

KACHELN	Bei der Verwendung mehrerer Fenster werden diese automatisch nebeneinander, in Spalten und Reihen, abhängig von der Fensteranzahl, angeordnet.
ÜBERLAPPEND	Bei der Verwendung mehrerer Fenster werden diese überlappend angeordnet.
NÄCHSTES	Mit dieser Funktion wird das nächste Fenster aktiviert. Diese Funktion ist nur bei mehreren Fenstern sinnvoll, welche entweder durch Mausklick oder aber diese Funktion aktiviert werden können.
VORHERGEHENDES	Diese Auswahlzeile entspricht der letzten mit dem Unterschied, daß die Reihenfolge der Fensteraktivierung nun in umgekehrter Reihenfolge geschieht.
SCHLIEßEN	Mit dieser Funktion wird das aktivierte Fenster geschlossen.

5. Dateien, Extensions, etc.

Folgende Dateien werden vom Meldungseditor verwendet bzw. angelegt.

- o **NAME.MSG**

In dieser Datei werden die Meldungen geschrieben. Sie ist die Basis- oder Sourcedatei, aus welcher bei einem Kompilierungslauf eine Fenstersimulationsdatei NAME.LST und entsprechend der Vorauswahl eine Ausgabedatei Fenster (NAME.W), Drucker (NAME.H) oder Logfile (NAME.L) angelegt generiert werden.

- o **NAME.W**

Ausgabedatei in ein Fenster, welche bei einem Kompilierungslauf mit der Auswahlzeile "Übersetzen" im Hauptmenü "Ausgabe" generiert wird.

- o **NAME.H**

Ausgabedatei auf den Drucker, welche bei einem Kompilierungslauf mit der Auswahlzeile "Übersetzen" im Hauptmenü "Ausgabe" generiert wird.

- o **NAME.L**

Ausgabedatei in ein Logfile, welche bei einem Kompilierungslauf mit der Auswahlzeile "Übersetzen" im Hauptmenü "Ausgabe" generiert wird.

- o **NAME.LST**

Datei simuliert die Ausgabe der Meldungen in Fenster, nachdem der Kompilierungslauf mit "Übersetzen" im Hauptmenü "Ausgabe" gestartet wurde.

- o **NAME.CFG**

In dieser Datei sichert der Meldungseditor alle veränderten Farbeinstellungen. Die Datei wird beim Ausführen der Auswahlzeile "Optionen sichern" angelegt.

6. Schalter (Aufrufparameter)

6.1 Schalter im Meldungstext hinter der Direktive LOGFILE (optional)

- /A:** append: die anfallenden Meldungen werden an das Logfile angehängt und komplett gespeichert. Die Direktive Logfile generiert automatisch Datei und Pfad. (Die Funktion des Schalters entspricht der Defaulteinstellung der Direktive LOGFILE)
- /O:** overwrite: löscht jeweils den Inhalt des bestehenden Logfiles und speichert dann die aktuell anfallende Meldung ab. Somit ist im Logfile nur die jeweils letzte Meldung abrufbar. Die Direktive Logfile generiert automatisch Datei und Pfad.
- /E:** erase: das Logfile wird gelöscht.
- /P:** print: diese Option druckt das Logfile aus; solange dies geschieht können vom VIADUKT aus keine weiteren Funktionen ausgeführt werden.
- /Rxxxx:** Der Inhalt des definierten Registers (xxxx) wird zwischen den bestehenden Logfilenamen und dessen Extension eingefügt.
- /V [xxxx]:** das Logfile wird in einem Dateianzeigefenster ausgegeben. Optional kann zusätzlich ein Register angegeben werden, in welchem die Zeilennummer definiert wird, ab der das Logfile ausgegeben wird.

6.2 Schalter beim Kommandozeilenaufruf von VIAGR (VIADUKT) zur Kontrolle von Meldungen

VIAGR /Wxxxx"Dateiname.W"

Dieser Schalter aktiviert die Meldungsfunktion. Die Meldungsnummer wird im Register mit der Nummer xxxx definiert. Die so definierten Meldungen werden der Meldungsdatei mit dem Namen "Dateiname.W" entnommen und in einem Fenster dargestellt. Befindet sich die Meldungsdatei nicht im gleichen Verzeichnis wie VIAGR, so ist zum Dateinamen der vollständige Pfad anzugeben.

VIAGR /Hxxxx"Dateiname.H"

Dieser Schalter aktiviert die Meldungsfunktion. Die Meldungsnummer wird im Register mit der Nummer xxxx definiert. Die so definierten Meldungen werden der Meldungsdatei mit dem Namen "Dateiname.H" entnommen und auf dem Drucker ausgegeben. Befindet sich die Meldungsdatei nicht im gleichen Verzeichnis wie VIAGR, so ist zum Dateinamen der vollständige Pfad anzugeben.

VIAGR /Lxxxx"Dateiname.L"

Dieser Schalter aktiviert die Meldungsfunktion. Die Meldungsnummer wird im Register mit der Nummer xxxx definiert. Die so definierten Meldungen werden der Meldungsdatei mit dem Namen "Dateiname.L" entnommen und in ein Logfile gespeichert. Befindet sich die Meldungsdatei nicht im gleichen Verzeichnis wie VIAGR, so ist zum Dateinamen der vollständige Pfad anzugeben.

7. Fehlermeldungen

Beim Kompilieren einer geschriebenen Meldungsdatei berücksichtigt der Kompiler folgende Fehlermöglichkeiten, und stellt diese in einem roten Fehlerbalken dar, wobei der Cursor die Fehlerstelle markiert.

Fehler 1: "(" nicht gefunden

An dieser Stelle erwartet der Kompiler eine öffnende Klammer.

Fehler 2: "," nicht gefunden

An dieser Stelle erwartet der Kompiler ein Komma.

Fehler 3: ")" nicht gefunden

An dieser Stelle erwartet der Kompiler eine schließende Klammer.

Fehler 4: Ganze Zahl erwartet

An dieser Stelle erwartet der Kompiler eine Integer-Konstante.

Fehler 5: Gleitkommazahl erwartet

An dieser Stelle erwartet der Kompiler eine Gleitkommazahl.

Fehler 6: Textsequenz erwartet

An dieser Stelle erwartet der Compiler eine Textsequenz.

Fehler 7: Ungültiger Wertebereich

Der Wertebereich der durch den Cursor markierten Zahl wurde überschritten.

Fehler 8: Textsequenz zu lang

Die durch den Cursor markierte Textsequenz ist zu lang.

Fehler 9: Die Direktive ist an dieser Stelle unzulässig

Entweder wurden `#LIST` und/oder `#NOLIST` zwischen `#INDEX` und `#ENDINDEX` angegeben oder `#OUTMODE` wurde nicht an den Textbeginn gestellt.

Fehler 30: #ENDDEFINE erwartet

Die Direktive `#DEFINE` wurde nicht mit `#ENDDEFINE` abgeschlossen.

Fehler 31: #ENDDEFINE ohne #DEFINE

Die Direktive `#ENDDEFINE` wurde ohne vorhergehendes `#DEFINE` verwendet.

Fehler 32: " = " erwartet

Der Compiler erwartet an der durch den Cursor markierten Stelle ein Gleichheitszeichen.

Fehler 33: " = " nicht gefunden

Bei der Funktionstastendefinition mit #FKEYS fehlt ein Gleichheitszeichen hinter Fxx.

Fehler 35: #ENDINDEX erwartet

Die Direktive #INDEX wurde nicht mit #ENDINDEX abgeschlossen.

Fehler 36: #ENDINDEX ohne #INDEX

Die Direktive #ENDINDEX wurde ohne vorhergehendes #INDEX verwendet.

Fehler 40: #ENDLOGFILE erwartet

Die Direktive #LOGFILE wurde nicht mit #ENDLOGFILE abgeschlossen.

Fehler 41: Dateiname erwartet

Nach der Direktive #INCLUDE oder #LOGFILE (nur wenn innerhalb einer Ausgabedatei für Logfiles verwendet) fehlt die Angabe des Dateinamens.

Fehler 42: #ENDLOGFILE ohne #LOGFILE

Die Direktive #ENDLOGFILE wurde ohne vorhergehendes #LOGFILE verwendet.

Fehler 43: Ungültiger Parameter

Als Logfileoption wurde ein ungültiger Parameter verwendet.

Fehler 50: Unerlaubtes #INCLUDE

Es wurde mehr als eine Includefileebene verwendet. Eine Schachtelung von Includefiles ist nicht möglich.

Fehler 51: Includefilename.Ext nicht gefunden

Der Kompiler hat das angegebene Includefile nicht gefunden.

Fehler 52: F1...F26 erwartet

Allgemeiner Fehler bei der Definition von Funktionstasten mit #FKEYS.

Fehler 53: Ungültiges Ausgabegerät

Hinter der Direktive #OUTMODE wurde kein gültiges Ausgabegerät definiert. Gültige Ausgabegeräte sind 'Window', 'Printer' und 'Logfile'.

8. Beispiele

Anhand der nun folgenden aufeinander aufbauenden Meldungsbeispiele wird die Funktion der einzelnen Elemente erklärt. Die meldungsspezifischen Funktionen, die über die Funktion eines Editors hinausgehen, befinden sich in den Pull-Down-Menüs "AUSGABE" und "KEYWORDS".

8.1 Beispiel 1

```
;Datei: vt_ex01d.msg
;
;Dieses Beispiel zeigt ein erstes einfaches Beispiel des VIADUKT Meldungs-
;editors.
;Wie Sie schon bemerkt haben werden, wird eine mit einem Semikolon star-
;tende Zeile als Kommentarzeile gewertet, und vom Meldungsprozessor
;ignoriert.
;Jede Meldung muß mit dem Keyword #INDEX beginnen. Dem Keyword folgt eine
;Zahl - die Meldungsnummer. Diese Zahl wird von der PASE-Steuerung vorge-
;geben, um die entsprechende Meldung auf dem VIADUKT-Bildschirm auszugeben.
;Die Reihenfolge der Meldungsnummern ist beliebig.
;
#INDEX 3
;Diese Zeile leitet die Meldungstextdefinition ein. Der Text wird von der
;PASE-Steuerung mit der Zahl 3 aktiviert.
Meldung, mit der Meldungsnummer 3. Diese Meldung ist 68 Zeichen breit
und besteht aus 8 Zeilen.

Meldungszeilen wie diese können zur Fehlermeldungs Ausgabe, oder aber
auch zur Informationsausgabe über Maschinenzustände, etc., genutzt
werden.

                                WEITER MIT <ENTER>
#ENDINDEX
;Das Keyword #ENDINDEX ist notwendig, um den Meldungstext des #INDEX 3
;logisch zu beenden.

#INDEX 4
;Nun ein anderer Text:
----- WARNUNG -----

    Temperaturalarm in Ofen 1.

    WEITER MIT <ENTER>

#ENDINDEX
```

```
;Diese Meldung kann man jetzt zu Testzwecken mit CTRL-F9 kompilieren.
;Dieser Schritt wird eine Datei mit dem Namen VT_EX01D.W erzeugen. Dann
;wird ein Fenster mit dem Namen ...\\VT_EX01D.LST geöffnet, in welchem der
;Inhalt eines Meldungsfensters im VIADUKT simuliert wird. Nach der Über-
;prüfung des Fensters ist dieses mit der ALT-F3 Tastenkombination oder der
;Maus zu schließen.
;Danach kann das VIAGR mit der Kommandozeile
;    VIAGR /a /w111"VT_EX01D.MSG"
;gestartet werden.
;Jetzt muß das Register 111 mit Wert 3 beschrieben werden, um den Aufruf
;der Meldung mit dem Index 3 im VIADUKT zu veranlassen.
;Das Register 111 kann per PASE-Programm oder SYMPAS beschrieben werden.
;Nachdem die erste Fehlermeldung im VIADUKT mit der RETURN-Taste be-
;stätigt wurde, wird das Register 111 auf 0 gesetzt. Jetzt kann das Regi-
;ster mit 4 (Index 4) beschrieben werden, und die zweite Meldung wird er-
;scheinen.

;Das ist alles, was ein PASE-Programm zu leisten hat um einen vordefinier-
;ten Text in einem Pop-Up-Window erscheinen zu lassen.

;Weiter mit VT_EX02D.MSG
```

8.2 Beispiel 2: Ausgabe in Fenster, auf den Drucker, in ein Logfile

```
;Datei: vt_ex02d.msg
;
;Zweites Beispiel VIADUKT Meldungseditor.

#INDEX 4
;Die Meldung im Index 4 wird zusätzlich mit dem Datum und der aktuellen
;Uhrzeit versehen. Die Keywords @DATE und @TIME können im SHIFT-F8-Menü
;ausgewählt, oder einfach geschrieben werden.
@DATE @TIME
----- WARNUNG -----

    Temperaturalarm in Ofen 1.

        Weiter mit <ENTER>
#ENDINDEX

;Diese Datei mit CTRL-F9 übersetzen. Jetzt wird eine Datei mit dem Namen
;VT_EX02D.W angelegt. Danach können Sie VIAGR mit folgender Kommandozeile
;starten
;
; VIAGR /a /w111"VT_EX02D.W"
;
;Mit der PASE oder dem VIADUKT ist nun in Register 111 der Wert 4 (für
;INDEX ;4) einzutragen.
;Der Warntext wird nun erscheinen mit dem aktuellen Datum und der Uhrzeit
;im Meldungskopf.

;Fortsetzung mit VT_EX03D.MSG
```

Diese Meldungsdatei können Sie jetzt mit der Tastenkombination CTRL-F9 übersetzen. Der Compiler generiert eine Testdatei mit der Extension ".LST" und stellt diese auf dem Bildschirm dar. Diese Datei simuliert eine spätere Meldungsangabe im VIADUKT. Dazu werden die Meldungen so dargestellt als würde ihre Ausgabe im Fenster erfolgen. Nach der Begutachtung der Simulationsdateien im Meldungseditor sollten sie die entsprechenden Fenster mit ALT-F3 schließen, um eine unnötige Anhäufung von Fenstern auf der Editoroberfläche zu verhindern.

Das Ausgabegerät ist mit Hilfe der Auswahlzeile "Ausgabemodus" im Pull-Down-Menü "Ausgabe" zu definieren. Entsprechend dieser Definition generiert der Compiler eine Meldungsdatei mit den Extensions ".W", ".H", ".L".

Folgende Darstellung wird durch einen Kompilationslauf, der durch die Auswahlzeile "Übersetzen" im Pull-Down-Menü "Ausgabe" zu aktivieren (oder CTRL-F9) ist, auf dem Bildschirm erzeugt:

```
////////// Index = 4 \\\\\\\\\\\\\\\\\\\\\\\
```

PASE-Meldung
10-16-1992 3:11:03pm ----- WARNUNG ----- Temperaturalarm in Ofen 1. Weiter mit <ENTER>

In der ersten Zeile, der Indexzeile, erscheint die Fehlernummer unter welcher die Meldung in dem definierten Register (siehe Kapitel 6.2 Schalter beim Kommandozeilenaufruf von VIAGR) aktiviert wird.

Auf welchem Ausgabemedium die Fehlermeldung erscheint ist beim Kommandozeilenaufruf vom VIADUKT auf DOS-Ebene festzulegen:

```
VIAGR /A /Wxxxx"Dateiname.W"
```

bringt die Meldungen in einem Fenster. Die Meldungsnummer, definiert durch die Zahl, welche der Direktive #INDEX im Meldungstext folgt, wird von der PASE-Steuerung in das Register xxxx geschrieben. Das VIADUKT fragt zyklisch den Inhalt des Registers xxxx ab, und stellt die entsprechende Meldung dar.

```
VIAGR /A /Hxxxx"Dateiname.H"
```

druckt die Meldungen auf einem Drucker aus. Die Meldungsnummer, definiert durch die Zahl, welche der Direktive #INDEX im Meldungstext folgt, wird von der PASE-Steuerung in das Register xxxx geschrieben. Das VIADUKT fragt zyklisch den Inhalt des Registers xxxx ab, und druckt die entsprechende Meldung aus.

```
VIAGR /A /Lxxxx"Dateiname.L"
```

speichert die Meldungen in ein Logfile. Die Meldungsnummer, definiert durch die Zahl, welche der Direktive #INDEX im Meldungstext folgt, wird von der PASE-Steuerung in das Register xxxx geschrieben. Das VIADUKT fragt zyklisch den Inhalt des Registers xxxx ab, und speichert die entsprechende Meldung in das Logfile. Der Dateiname des Logfiles ergibt sich entweder aus der Defaulteinstellung (entsprechend dem aktuellen Datum: z.B. 921020 für den 20.10.1992) oder kann durch den Anwender selbst definiert werden. (Siehe #LOGFILE).

8.3 Beispiel 3

```

;Datei vt_ex03d.msg
;
;Drittes Beispiel zum VIADUKT Meldungseditors.

#INDEX 4
;Die Mitteilung in Index 4 wird jetzt mit mehr Informationen angereichert:
;Anstelle der einfachen Übertemperaturmeldung, wird der momentane Tempera-
;turwert angezeigt. Der Temperatursollwert ist in Register 230 gespeichert,
;die obere Alarmgrenze in Register 430 und die Isttemperatur in Register
;162. Der Registerwert 0 entspricht -40 °C, der Wert 1023, 160°C. Um die
;Temperatur in Grad Celsius anzuzeigen, ist eine Einheitenumrechnung not-
;wendig.
;
;Das Keyword REGISTER wird mit der Tastenkombination SHIFT-F8 ausgewählt.
;Jetzt können alle erforderlichen Werte in das Eingabefeld eingegeben wer-
;den. Durch das Eingabefeld bewegt man sich mit der TAB oder der SHIFT-TAB
;Taste. (Oder mit der Maus). Nach der richtigen Eingabe aller Werte mit
;RETURN oder dem Mausschalter OK bestätigen.
;
;Um eine bereits definierte Variable zu modifizieren, ist der Cursor an
;einer beliebigen Stelle des Variablentextes zu plazieren und SHIFT-F8 zu
;drücken.
;Das Eingabefenster öffnet sich, und die Werte können geändert werden.
;
;Probieren Sie das ruhig mal aus. Plazieren Sie den Cursor auf der
;@REGISTER-Variablen und drücken Sie SHIFT-F8.
@DATE @TIME
----- WARNUNG -----

    Temperaturalarm in Ofen 1.

    Aktuelle Temperatur: @REGISTER(0,162,0,3,1,0.1955,-40) °C
    Sollwert           : @REGISTER(0,230,0,3,1,0.1955,-40) °C
    Obere Grenze       : @REGISTER(0,430,0,3,1,0.1955,-40) °C

    Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.

    Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                  einem Schraubenzieher öffnen. Danach alle Fenster
                  öffnen und die Feuerwehr rufen.

                Weiter mit <ENTER>

#ENDINDEX

;Diese Datei mit CTRL-F9 übersetzen. Jetzt wird eine Datei mit dem Namen
;VT_EX03D.W angelegt. Danach können Sie VIAGR mit folgender Kommandozeile
;starten
;
; VIAGR /a /w111"VT_EX03D.W"
;
;Fortsetzung mit VT_EX04D.MSG

```

Folgende Meldungssimulation erscheint nach einem Übersetzungslauf auf dem Bildschirm (CTRL-F9):

```
////////// Index = 4 \\\\\\\\\\\\\\\\\\\
```

PASE-Meldung	
10-16-1992	3:16:34pm
----- WARNUNG -----	
Temperaturalarm in Ofen 1.	
Aktuelle Temperatur:	±###.# °C
Sollwert	: ±###.# °C
Obere Grenze	: ±###.# °C
Die Maschine ist im \$\$\$\$\$\$\$\$ Modus.	
Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder einem Schraubenzieher öffnen. Danach alle Fenster öffnen und die Feuerwehr rufen.	
Weiter mit <ENTER>	

Anhand dieser Darstellung können Sie nun überprüfen, ob die Meldungsdarstellung ihren Vorstellungen entspricht und gegebenenfalls, nach dem schließen des Simulationsfensters mit ALT-F3, Korrekturen im Meldungstext machen. Ein weiterer Kompilationslauf gibt über den Erfolg der Änderungen Auskunft.

8.4 Beispiel 4

```
;Datei: vt_ex04d.msg
;
;Beispiel 4 des VIADUKT Meldungseditors.

#INDEX 4
;Zusätzlich zu den aktuellen Prozeßwerten wird nun ein Hilfstext einge-
;blendet, der dem Anwender eine Problemlösung anbietet.
;Auch wird der gegenwärtige Betriebsstatus der Maschine mit Hilfe des
;Merker-Keywords angezeigt. Jede boolesche Variable (Merker, Eingang,
;Ausgang, RegBit) kann entsprechend dem jeweiligen Status der Variablen
;zwei verschiedene Texte anzeigen.
;
;Der Automatikmodus-Merker des nächsten Beispieles ist 10.

@DATE @TIME
----- WARNUNG -----

    Temperaturalarm in Ofen 1.

    Aktuelle Temperatur: @REGISTER(0,162,0,3,1,0.1955,-40) °C
    Sollwert           : @REGISTER(0,230,0,3,1,0.1955,-40) °C
    Obere Grenze       : @REGISTER(0,430,0,3,1,0.1955,-40) °C

    Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.

    Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                  einem Schraubenzieher öffnen. Danach alle Fenster
                  öffnen und die Feuerwehr rufen.

                Weiter mit <ENTER>

#ENDINDEX

;Diese Datei mit CTRL-F9 übersetzen. Jetzt wird eine Datei mit dem Namen
;VT_EX04D.W angelegt. Danach können Sie VIAGR mit folgender Kommandozeile
;starten
;
; VIAGR /a /w111"VT_EX04D.W"
;
;Fortsetzung mit VT_EX05D.MSG
```


8.5 Beispiel 5: Symboldefinitionen

```
;Datei: vt_ex05d.msg
;
;Beispiel 5 des VIADUKT Meldungseditors.

;Neu ist in diesem Beispiel die Verwendung von Symboldefinitionen. Der
;Anwender kann Symbole für den späteren Gebrauch definieren. Dies ist
;besonders hilfreich für Variablendefinitionen, wenn die gleiche Variable
;oft und/oder in mehreren Fenstern verwendet wird. Siehe Beispiel unten:
;Drei Registervariablen werden zwischen #DEFINE und #ENDDFINE definiert.
;Um die Variablen jetzt aufzurufen, ist nun der zugewiesene Name (z.B.
;%templ) zu verwenden.
;Wichtig: Jedes Symbol muß mit dem % Zeichen beginnen.

#DEFINE
%templ    = @REGISTER(0,162,0,3,1,0.1955,-40)
%set1     = @REGISTER(0,230,0,3,1,0.1955,-40)
%limit1   = @REGISTER(0,430,0,3,1,0.1955,-40)
#ENDDFINE

#INDEX 4
@DATE @TIME
----- WARNUNG -----

    Temperaturalarm in Ofen 1.

    Aktuelle Temperatur: %templ  °C
    Sollwert           : %set1   °C
    Obere Grenze       : %limit1 °C

    Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.

    Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                  einem Schraubenzieher öffnen. Danach alle Fenster
                  öffnen und die Feuerwehr rufen.

    Weiter mit <ENTER>

#ENDINDEX
```

```
#INDEX 5
* * * * * * * * * * ALARM * * * * * * * * * *
```

Der Sollwert für den Ofen 1 ist %set1 °C mit einer oberen Grenze von %limit1 °C. Deshalb wurde die Evakuierung der Firma angeordnet. Verständigen Sie Herrn Boss unter der Rufnummer 0815-4711 und versuchen Sie dann den Raum möglichst schnell zu verlassen.

Drücken Sie nicht <ENTER>, sondern verlassen Sie schnell den Raum!

```
#ENDINDEX
```

```
;Diese Datei mit CTRL-F9 übersetzen. Jetzt wird eine Datei mit dem Namen
;VT_EX05.W angelegt. Danach können Sie VIAGR mit folgender Kommandozeile
;starten
```

```
;  
; VIAGR /a /w111"VT_EX05.W"  
;
```

```
;Fortsetzung mit VT_EX06.MSG
```

8.6 Beispiel 6: @AUTODATE, @AUTOTIME

```
;Datei: vt_ex06d.msg
;
;Dieses Beispiel stellt einige der vordefinierten System-Symbole vor.
;Diese können mit der DEFINE Funktion geändert werden.
;
;Statt der Symbole @DATE und @TIME können in jedem Indextext auch
;die Systemsymbole %AUTODATE und %AUTOTIME verwendet werden. Die Default-
;einstellung dieser Symbole ist OFF. Werden sie auf ON gesetzt so wird
;automatisch am Beginn jeder Indextmeldung das Datum bzw. die aktuelle
;Zeit eingefügt.

#DEFINE
%temp1   = @REGISTER(0,162,0,3,1,0.1955,-40)
%set1    = @REGISTER(0,230,0,3,1,0.1955,-40)
%limit1  = @REGISTER(0,430,0,3,1,0.1955,-40)

%AutoDate = ON
%AutoTime = ON
#ENDDEFINE

#INDEX 4
;Bitte nehmen Sie zur Kenntnis, daß diese Zeilen nun Kommentar sind: @DATE
;@TIME
----- WARNUNG -----

    Temperaturalarm in Ofen 1.

    Aktuelle Temperatur: %temp1 °C
    Sollwert           : %set1 °C
    Obere Grenze       : %limit1 °C

    Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.

    Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                  einem Schraubenzieher öffnen. Danach alle Fenster
                  öffnen und die Feuerwehr rufen.

    Weiter mit <ENTER>

#ENDINDEX
```

```
#INDEX 5
* * * * * * * * * * ALARM * * * * * * * * * *
```

Der Sollwert für den Ofen 1 ist %set1 °C mit einer oberen Grenze von %limit1 °C. Deshalb wurde die Evakuierung der Firma angeordnet. Verständigen Sie Herrn Boss unter der Rufnummer 0815-4711 und versuchen Sie dann den Raum möglichst schnell zu verlassen.

Drücken Sie nicht <ENTER>, sondern verlassen Sie schnell den Raum!

```
#ENDINDEX
```

```
;Diese Datei mit CTRL-F9 übersetzen. Jetzt wird eine Datei mit dem Namen
;VT_EX06.W angelegt. Danach können Sie VIAGR mit folgender Kommandozeile
;starten
```

```
;  
; VIAGR /a /w111"VT_EX06.W"  
;
```

```
;Fortsetzung mit VT_EX07.MSG
```

8.7 Beispiel 7

```
;Datei: vt_ex07.msg
;
;Dieses Beispiel führt das Logfile ein. In diesem können Informationen,
;wie in einem Logbuch, auf der Festplatte abgelegt werden.
;
;Um dies zu erreichen sind zwei Wege möglich:
;
; Einmal kann der Ausgabemodus LOGFILE von VIATXTGR benutzt werden, um
; dann VIAGR mit dem Kommandozeilenbefehl /L aufzurufen.
; Jetzt werden alle Informationen, welche sonst (/W) in einem Fenster dar-
; gestellt werden in ein Logfile geschrieben.
;
; Die andere Möglichkeit besteht darin den Text, welcher für das Log-
; file bestimmt ist zwischen die Keywords #LOGFILE und #ENDLOGFILE zu
; schreiben.
;
;Um den ersten Fall zu realisieren ist im AUSGABE Pull-Down-Menü die
;Einstellung AUSGABEMODUS auf LOGFILE zu stellen.
;Dann ist die Datei zu kompilieren (die im Bezug auf den Inhalt zwischen
;INDEX und ENDINDEX mit der Datei aus Beispiel 6 identisch ist).

#DEFINE
%templ    = @REGISTER(0,162,0,3,1,0.1955,-40)
%set1     = @REGISTER(0,230,0,3,1,0.1955,-40)
%limit1   = @REGISTER(0,430,0,3,1,0.1955,-40)

%AutoDate = ON
%AutoTime = ON
#ENDDEFINE

#INDEX 4
;Bitte nehmen Sie zur Kenntnis, daß diese Zeilen nun Kommentar sind: @DATE
;@TIME
----- WARNUNG -----

    Temperaturalarm in Ofen 1.

    Aktuelle Temperatur: %templ  °C
    Sollwert           : %set1   °C
    Obere Grenze       : %limit1 °C

    Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.

    Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                  einem Schraubenzieher öffnen. Danach alle Fenster
                  öffnen und die Feuerwehr rufen.

    Weiter mit <ENTER>

#ENDINDEX
```

```
#INDEX 5
* * * * * * * * * * ALARM * * * * * * * * * *
```

Der Sollwert für den Ofen 1 ist %set1 °C mit einer oberen Grenze von %limit1 °C. Deshalb wurde die Evakuierung der Firma angeordnet. Verständigen Sie Herrn Boss unter der Rufnummer 0815-4711 und versuchen Sie dann den Raum möglichst schnell zu verlassen.

Drücken Sie nicht <ENTER>, sondern verlassen Sie schnell den Raum!

```
#ENDINDEX
```

```
;Um dieses Logfile zu benutzen, ist der Ausgabemodus auf LOGFILE zu
;stellen und die Übersetzung mit der Tastenkombination CTRL-F9 einzuleiten.
;Jetzt wird die Datei VT_EX07D.L generiert.
;Dann können Sie VIAGR mit der Kommandozeile
;VIAGR /a /L111"VT_EX07D.L"
;starten.
```

```
;Nachdem der Wert 4 in das Register 111 geschrieben wurde wird ein Logfile
;im aktuellen Verzeichnis angelegt, welches einen Dateinamen und die
;Extension .LOG hat. Der Inhalt entspricht dem des INDEX 4. Immer wenn das
;Register 111 auf 4 gesetzt wird, wird dieser Text an das Logfile angehängt
;werden. Der Dateiname setzt sich aus dem aktuellen Datum und der Extension
;.LOG zusammen.
```

```
;Fortsetzung mit VT_EX08D.MSG
```

8.8 Beispiel 8: Schreiben in ein Logfile

```
;Datei: vt_ex08d.msg
;
;Dieses Beispiel stellt das Schreiben in ein Logfile dar. Die Datei ansich
;soll in einem Fenster erscheinen (Fenster Textdatei; Option /W), so daß
;der Ausgabemodus auf Fenster zu stellen ist.

#DEFINE
%templ    = @REGISTER(0,162,0,3,1,0.1955,-40)
%set1     = @REGISTER(0,230,0,3,1,0.1955,-40)
%limit1   = @REGISTER(0,430,0,3,1,0.1955,-40)

%AutoDate = ON
%AutoTime = ON
#ENDDFINE

#INDEX 4
----- WARNUNG -----

#LOGFILE Warnung.log
  Temperaturalarm in Ofen 1.

  Aktuelle Temperatur: %templ  °C
  Sollwert           : %set1   °C
  Obere Grenze       : %limit1 °C

  Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.
#ENDLOGFILE

  Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                einem Schraubenzieher öffnen. Danach alle Fenster
                öffnen und die Feuerwehr rufen.

                Weiter mit <ENTER>

#ENDINDEX
```

```
#INDEX 5
* * * * * * * * * * ALARM * * * * * * * * * *
```

Der Sollwert für den Ofen 1 ist %set1 °C mit einer oberen Grenze von %limit1 °C. Deshalb wurde die Evakuierung der Firma angeordnet. Verständigen Sie Herrn Boss unter der Rufnummer 0815-4711 und versuchen Sie dann den Raum möglichst schnell zu verlassen.

Drücken Sie nicht <ENTER>, sondern verlassen Sie schnell den Raum!

```
#ENDINDEX
```

```
;Um diese Datei zu benutzen, ist der Ausgabemodus auf FENSTER zu
;stellen und die Übersetzung mit der Tastenkombination CTRL-F9 einzuleiten.
;Jetzt wird die Datei VT_EX08D.W generiert.
;Dann können Sie VIAGR mit der Kommandozeile
;VIAGR /a /W111"VT_EX08D.W"
;starten.
```

```
;Register 111 ist nun mit dem Wert 4 zu beschreiben und nach dem Erscheinen
;des Fensters wird es alle Informationen des INDEX 4 enthalten.
;Zusätzlich wird das Logfile WARNUNG.LOG angelegt. In dieses wird der Text
;zwischen den Direktiven LOGFILE und ENDLOGFILE geschrieben.
;Immer wenn das Register 111 auf den Wert 4 gesetzt wird, wird der Text
;zwischen den Direktiven LOGFILE und ENDLOGFILE an die Datei "WARNUNG.LOG"
;angehängt.
```

```
;Fortsetzung mit VT_EX09D.MSG
```


8.9 Beispiel 9

```
;Datei vt_ex09d.msg
;
;Soll die Ausgabe auf das gleiche Logfile erfolgen, welches an ver-
;schiedenen Textstellen erscheint, so macht es Sinn ein
;Symbol zu definieren: Siehe Eintrag %file1 in dem DEFINE-Abschnitt unten.

#DEFINE
%templ    = @REGISTER(0,162,0,3,1,0.1955,-40)
%set1     = @REGISTER(0,230,0,3,1,0.1955,-40)
%limit1   = @REGISTER(0,430,0,3,1,0.1955,-40)
%AutoDate = ON
%AutoTime = ON

%file1    = "Warnung.log"
#ENDDFINE

#INDEX 4
----- WARNUNG -----

#LOGFILE %file1
  Temperaturalarm in Ofen 1.

  Aktuelle Temperatur: %templ °C
  Sollwert           : %set1 °C
  Obere Grenze       : %limit1 °C

  Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.
#ENDLOGFILE

  Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                 einem Schraubenzieher öffnen. Danach alle Fenster
                 öffnen und die Feuerwehr rufen.

                 Weiter mit <ENTER>

#ENDINDEX

#INDEX 5
* * * * * ALARM * * * * *

Der Sollwert für den Ofen 1 ist %set1 °C mit einer oberen Grenze von
%limit1 °C. Deshalb wurde die Evakuierung der Firma angeordnet.
Verständigen Sie Herrn Boss unter der Rufnummer 0815-4711 und versuchen
Sie dann den Raum möglichst schnell zu verlassen.

Drücken Sie nicht <ENTER>, sondern verlassen Sie schnell den Raum!

#ENDINDEX
```

```
;Um diese Datei zu benutzen, ist der Ausgabemodus auf FENSTER zu  
;stellen und die Übersetzung mit der Tastenkombination CTRL-F9 einzuleiten.  
;Jetzt wird die Datei VT_EX09D.W generiert.  
;Dann können Sie VIAGR mit der Kommandozeile  
;VIAGR /a /W111"VT_EX09D.W"  
;starten.
```

```
;Register 111 ist nun mit dem Wert 4 zu beschreiben und nach dem Erscheinen  
;des Fensters wird es alle Informationen des INDEX 4 enthalten.  
;Immer wenn das Register 111 auf den Wert 4 gesetzt wird, wird der Text  
;zwischen den Direktiven LOGFILE und ENDLOGFILE an die Datei "WARNUNG.LOG"  
;angehängt.
```

```
;Fortsetzung mit VT_EX10D.MSG
```

8.10 Beispiel 10: Logfile-Optionen

```
;Datei: vt_ex10d.msg
;
;Diese Datei stellt die Optionen zum Keyword LOGFILE dar.
; INDEX 5 zeigt wie ein Logfile überschrieben wird,
; INDEX 6 zeigt wie ein Logfile gelöscht wird,
; INDEX 7 zeigt wie Logfile-Dateinamen definiert werden.
;%file1 im Definitionsteil unten.

#DEFINE
%templ    = @REGISTER(0,162,0,3,1,0.1955,-40)
%set1     = @REGISTER(0,230,0,3,1,0.1955,-40)
%limit1   = @REGISTER(0,430,0,3,1,0.1955,-40)
%AutoDate = ON
%AutoTime = ON

%file1 = "warnung.log"

#ENDDDEFINE

;INDEX 4 ist identisch mit Beispiel 9.
#INDEX 4
----- WARNUNG -----

#LOGFILE %file1
  Temperaturalarm in Ofen 1.

  Aktuelle Temperatur: %templ °C
  Sollwert           : %set1 °C
  Obere Grenze       : %limit1 °C

  Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.
#ENDLOGFILE

  Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                 einem Schraubenzieher öffnen. Danach alle Fenster
                 öffnen und die Feuerwehr rufen.

                 Weiter mit <ENTER>

#ENDINDEX

;INDEX 5 zeigt das Überschreiben des Logfiles.
;
; Immer wenn INDEX 5 aufgerufen wird, wird die vorhergehende
; Datei überschrieben. Somit bleiben immer nur die Fehlermeldungen
; aus dem letzten INDEX 5 Aufruf erhalten.
;
;
; Nach dem Aufruf von INDEX 5 wird das Logfile gelöscht bevor ein
; neuer Eintrag erfolgt. Danach kann man, wenn gewünscht, mit INDEX 4
; wieder Informationen an das gleiche Logfile anhängen.
;
;
; Die Überschreibfunktion wird mit dem Schalter /o in der LOGFILE
; Zeile aktiviert
```

```
#INDEX 5
----- WARNUNG -----

#LOGFILE %file1 /o
  Temperaturalarm in Ofen 1.

  Aktuelle Temperatur: %templ °C
  Sollwert           : %set1 °C
  Obere Grenze       : %limit1 °C

  Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.
#ENDLOGFILE

  Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                 einem Schraubenzieher öffnen. Danach alle Fenster
                 öffnen und die Feuerwehr rufen.

                 Weiter mit <ENTER>

#ENDINDEX

;INDEX 6 zeigt wie ein Logfile gelöscht wird.
;Dies wird mit der Option /e realisiert. Natürlich kann keine
;Information in die Datei geschrieben werden.

#INDEX 6
----- LÖSCHE LOGFILE -----

#LOGFILE %file1 /e
#ENDLOGFILE

  Die Information im Logfile %file1
  gelöscht.

  Weiter mit <ENTER>

#ENDINDEX

;INDEX 7 stellt die Eigendefinition von Logfilenamen dar.
;
; Mit der Option /r kann die Information auf einen laufzeitabhängigen
; Dateinamen ausgegeben werden. Nach der /r Option ist eine Register-
; nummer anzugeben. Nachdem der Zugriff auf ein Logfile angefordert wird,
; schaut VIAGR in dem spezifizierten Register nach einer Zahl welche an
; den Dateinamen angehängt wird. Die Informationen werden jetzt an das
; eben definierte Logfile angehängt.
;
; Beispiel: Register hat einen Inhalt zwischen 1 und 21 (3 Schichten pro
; Tag, 7 Tage pro Woche -> 21 Logfiles).
; Das PASE-Programm verwaltet den Zahlenwert in Register 190.
; INDEX 7 Texte werden daher auf die Dateien mit folgenden Namen geleitet:
;           SCHICH1.LOG bis SCHICH21.LOG
;
;
;
```

```
#INDEX 7
----- WARNUNG -----

#LOGFILE SCHICH /r190
  Temperaturalam in Ofen 1.

  Aktuelle Temperatur: %temp1 °C
  Sollwert           : %set1 °C
  Obere Grenze       : %limit1 °C

  Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.
#ENDLOGFILE

  Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                 einem Schraubenzieher öffnen. Danach alle Fenster
                 öffnen und die Feuerwehr rufen.

                 Weiter mit <ENTER>

#ENDINDEX

; Die Logfile-Optionen können auch kombiniert genutzt werden. Die Zeile
;
; LOGFILE "A.LOG" /r190 /o
;
; liest Register 190 aus (Zahlenwert z.B. 12), löscht die Datei A12.LOG,
; sofern diese existiert, und gibt die neuen Informationen auf dieselbe
; Datei aus.
;
; Um diese Datei zu benutzen, ist der Ausgabemodus auf FENSTER zu
; setzen und die Datei mit der Tastenkombination CTRL-F9 zu starten.
; Jetzt wird die Datei VT_EX10D.W erzeugt. Dann können Sie VIAGR mit der
; Kommandozeile starten
;
; VIAGR /a /W111"VT_EX10D.W"
;
; Fortsetzung mit VT_EX11D.MSG
```

8.11 Beispiel 11: Systemsymbole zur Logfilenamens-Definition

```
;Datei: vt_ex11d.msg
;
;Vier System-Symbole werden in dieser Datei erklärt.
; MLFpath   MainLogFile path
;           Für dieses Symbol gibt es keine Defaulteinstellung.
;           Wird es ohne zusätzliche Definition benutzt so findet das
;           aktuelle Verzeichnis Verwendung.
;           Wird das Symbol neu definiert, so wird der neue Pfad ver-
;           wendet. Es ist wichtig auf gültige Pfadnamen zu achten.
;           Beispiel:
;           %MLFpath = c:\projekt1\logfiles\
;           Das \ am Ende der Pfadangabe ist unbedingt notwendig.
;
;
; MLFname   MainLogFile name
;           Die Defaulteinstellung für dieses Symbol ist die System-
;           variable @Dateidatum. Wenn keine Änderungen erfolgen wird ein
;           Dateiname aus dem aktuellen Datum kreiert (16.09.1992 wird zu
;           "920916").
;           Als Neudefinition kann jeder DOS-Dateiname verwendet werden.
;           Beispiel:
;           %MLFname = Alarm
;
;
; MLFext    MainLogFile extension
;           Die Defaulteinstellung ist ".LOG". Um eine andere Extension
;           zu erhalten muß das Symbol umdefiniert werden.
;           Beispiel:
;           %MLFextension = .DAT
;
;
; MLFOptions MainLogFile options
;           Alle Optionen, welche in Beispiel 10 beschrieben sind können
;           für das MainLogFile definiert werden. Die Defaulteinstellung
;           ist /a (Anhänge-Modus).
;           Beispiel:
;           %MLFOptions = /r190 /o
;
;
;Werden die Symbole nicht durch den Anwender definiert, so verwendet das
;System die entsprechenden Defaulteinstellungen. Die Symbole werden zur
;Definition der Logfilenamens innerhalb von Meldungsdateien verwendet (/L -
;Kommandozeilenoption). Außerdem finden sie immer dann Verwendung wenn
;keine Dateiname nach einem LOGFILE Keyword angegeben wird.
;
;Der erste Index dieser Datei (Index 4) schreibt in das Default-Logfile
;(das aktuelle Datum ist der Dateiname, LOG die Extension), der zweite
;Index wird wegen der Neudefinition eine andere Datei anlegen (MLFxxx).
```

```
#DEFINE
%templ    = @REGISTER(0,162,0,3,1,0.1955,-40)
%set1     = @REGISTER(0,230,0,3,1,0.1955,-40)
%limit1   = @REGISTER(0,430,0,3,1,0.1955,-40)
%AutoDate = ON
%AutoTime = ON

#ENDDFINE

#INDEX 4
----- WARNUNG -----

#LOGFILE
  Temperaturalarm in Ofen 1.

  Aktuelle Temperatur: %templ °C
  Sollwert           : %set1 °C
  Obere Grenze       : %limit1 °C

  Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.
#ENDLOGFILE

  Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                  einem Schraubenzieher öffnen. Danach alle Fenster
                  öffnen und die Feuerwehr rufen.

                  Weiter mit <ENTER>

#ENDINDEX

#DEFINE
%MLFname = Alarm
#ENDDFINE

#INDEX 5
----- WARNUNG -----

#LOGFILE
  Temperaturalarm in Ofen 1.

  Aktuelle Temperatur: %templ °C
  Sollwert           : %set1 °C
  Obere Grenze       : %limit1 °C

  Die Maschine ist im @FLAG(0,10,0,9,"HAND","AUTOMATIK") Modus.
#ENDLOGFILE

  Gegenmaßnahme: Ventil "Kalt Wasser" mit dem Taster "K.W." oder
                  einem Schraubenzieher öffnen. Danach alle Fenster
                  öffnen und die Feuerwehr rufen.

                  Weiter mit <ENTER>

#ENDINDEX

;Um diese Datei zu benutzen ist der Ausgabemodus auf FENSTER zu setzen
;Jetzt wird die Datei VT_EX11D.W erzeugt.
;Danach ist VIAGR mit folgender Kommandozeile zu starten:
;VIAGR /a /W111"VT_EX11D.W"
;Fortsetzung mit VT_EX12D.MSG
```

8.12 Beispiel 12: Ausgabe auf dem Drucker, Textboxen

```
;Datei: vt_ex12d.msg
;
;Die Kommandozeilenoption /H des Programmes VIAGR erlaubt die Definition
;eines Steuerregisters für Drucker Meldungen.
;
;Meldungsdateien für Drucker Meldungen werden in der fast gleichen Weise
;erstellt wie Meldungsdateien für Pop-Up-Fenster. Die einzigen Unterschiede
;sind:
;   o Der Ausgabemodus Drucker ist im Ausgabemodus-Menü zu wählen
;   o Es wird eine Datei mit der Extension .H erzeugt
;   o Es können so viele Zeichen in einer Zeile Verwendung finden
;     wie vom Drucker unterstützt werden
;
;
;Dieses Beispiel zeigt ein Schichtprotokoll für den Ausdruck am Ende einer
;Arbeitsschicht. Die graphischen Zeichen können mit ALT-Tastenkombinationen
;erzeugt werden. Die Eingabenummern können der ASCII-Tabelle im Pull-Down-
;Menü ≡ entnommen werden.
;
;In diesem Beispiel ist es sehr nützlich einen Voreindruck vom Ausdruck in
;dem VT_EX12.LST Fenster zu bekommen. Sie sehen wie der Text auf dem Dru-
;cker erscheinen wird, ohne zahlreiche Seiten Papier verschwenden zu müssen
;bis alle Zeichen da erscheinen, wo sie erscheinen sollen.
;
;Vergessen Sie nicht das *.LST-Fenster zu schließen, bevor Sie erneut über-
;setzen. Sonst werden sich immer mehr *.LST-Fenster ansammeln, ohne daß das
;Fenster des letzten Testlaufes bestimmt werden könnte. Sie gewinnen einen
;Überblick über alle Fenster, indem Sie im Pull-Down-Menü Fenster die Aus-
;wahlzeile "Kacheln" aktivieren.
```

#INDEX 4

@DATE	@TIME	Produktions-Protokoll
		Maschinen-Nummer: @REGISTER(0,180,0,1,0,1.0,0.0)
		Anzahl bearbeiteter Teile: @REGISTER(0,2020,0,6,0,1.0,0.0)
		Anzahl defekter Teile: @REGISTER(0,2021,0,6,0,1.0,0.0)
		Zusammenfassung Toleranzen:
		Minimale Länge "a": @REGISTER(0,2058,0,3,2,.01,0.0)
		Maximale Länge "a": @REGISTER(0,2059,0,3,2,.01,0.0)
		Durchschnittslänge "a": @REGISTER(0,2060,0,3,2,.01,0.0)
		Einstellen der Werkzeuge ist @FLAG(0,101,0,5,"nicht","") notwendig.

#ENDINDEX

In diesem letzten Beispiel wird dargestellt wie Sie Textboxen erzeugen können. Geben Sie die entsprechenden Zeichen über ALT-Tastenkombinationen ein und gestalten Sie so die gewünschten Textboxen.