

IBT



 früher
EUROTHERM DRIVES

Serie







Intelligentes Bedienterminal

Typ: IBT

Technisches-Handbuch

Weitere Unterlagen,
die im Zusammenhang mit
diesem Dokument stehen.

Further descriptions,
that relate to this document.

635 - Produkt-Handbuch	UL: 7.1.5.6 	<i>635 - Product manual</i>
637 - Produkt-Handbuch	UL: 7.2.8.3 	<i>637 - Product manual</i>
IBT - Produkt-Beschreibung	UL: 9.5.1 	<i>IBT - Product description</i>
IBT - Erstinbetriebnahme	UL: 9.5.2 	<i>IBT - Getting started</i>

© **EUROTHERM** Antriebstechnik GmbH.
Alle Rechte vorbehalten. Kein Teil der Beschreibung
darf in irgendeiner Form, ohne Zustimmung der Gesell-
schaft vervielfältigt oder weiter verarbeitet werden.

Änderungen sind ohne vorherige Ankündigung
vorbehalten.

EUROTHERM hat für seine Produkte teilweise Waren-
zeichenschutz und Gebrauchsmusterschutz eintragen
lassen. Aus dem Überlassen der Beschreibungen darf
nicht angenommen werden, daß damit eine Übertragung
von irgendwelchen Rechten stattfindet.

Hergestellt in Deutschland, 1998

© **EUROTHERM Drives Limited.**
*All rights reserved. No portion of this description may
be produced or processed in any form without the
consent of the company.*

Changes are subject to change without notice.

EUROTHERM has registered in part trademark
protection and legal protection of designs. The handing
over of the descriptions may not be construed as the
transfer of any rights.

Made in Germany, 1998

Inhalt

1	TesiMod Bedienkonzept	7
1.1	Bedienterminals	7
1.1.1	Ladbare Protokolltreiber	7
1.1.2	Vernetzung von Terminals	7
1.1.3	Schnittstelle für Programmiergeräte	8
1.1.4	Was ist TesiMod und was kann damit erreicht werden?	8
1.1.5	Betriebsarten	8
1.1.6	Anwenderbeschreibung	8
1.1.7	Speicherung der Anwenderbeschreibung	9
1.1.8	Adressierung der Variablen	9
1.1.9	Datenfreigabe	9
1.1.10	Paßwortschutz	10
1.1.11	Editoren	10
1.1.12	Hilfe	10
1.1.13	Funktionstasten	11
1.1.14	Softkeys	11
1.1.15	Variablen	11
1.1.16	Grafik	11
1.1.17	Rezepturen und Datensätze	12
1.1.18	Meldungen	13
1.1.19	Systemmeldungen	13
1.1.20	Projektiersystem	13
1.2	Gemeinsame Merkmale:	15
1.2.1	Hardware	15
1.2.2	Software	15
1.3	Bestellhinweise	15
2	Betriebsarten	16
2.1	Einstellen der Betriebsart	16
3	TesiMod Standard-Mode	17
3.1	Betriebsart einstellen	17
3.2	Verhalten der Terminals beim Einschalten	18
3.2.1	Verhalten ohne gültige Anwenderbeschreibung	19
3.3	Kommunikation im Standard-Mode	19
3.4	Das Bedienkonzept	19
3.4.1	Hierarchische Maskenstruktur bei TSdos	20
3.4.2	Maskenstruktur bei TSwin	20
3.4.3	Externe Maskenanwahl	21
3.4.4	Paßwortschutz, Zugriffsrechte	21
3.5	Masken	25
3.5.1	Maskenparameter	25
3.5.2	Systemmasken	26
3.6	Variablen	32
3.6.1	Ausgabevariablen	32
3.6.2	Eingabevariablen	43
3.6.3	Systemvariablen	45
3.6.4	Editoren	70
3.6.5	Externe Datenfreigabe	77
3.6.6	SPS-Handshake	78
3.6.7	Auffrischen einmaliger Ausgabedaten	78
3.6.8	Änderung der Daten	79
3.7	Grafik	80
3.7.1	Grafische Objekte (TSdos)	80
3.7.2	Bilder (TSwin)	81
3.7.3	Grafik in Bedienterminals	81
3.8	Rezepturen	83
3.8.1	Aufbau einer Rezeptur	84
3.8.2	Rezeptur- und Datensatzbearbeitung	84

3.8.3	Datensatztransfer an / von Steuerung	87
3.8.4	Datensatztransfer an / von PC.....	89
3.8.5	Datensätze drucken.....	91
3.8.6	Speicherbedarf für Datensätze	92
3.9	TesiMod-Meldungssystem	93
3.9.1	Interne Meldungen.....	93
3.9.2	Externe Meldungen	101
3.10	Hilfesystem.....	110
3.10.1	Default-Hilfetext.....	110
3.10.2	Hilfetext zu Masken.....	110
3.10.3	Hilfetext zu Variablen	110
3.11	Funktionstasten.....	110
3.11.1	Direktanwahltasten	111
3.11.2	Funktionstasten der Steuerung.....	111
3.11.3	Softkeytasten	111
3.11.4	Funktionstasten steuern Parallelausgänge	112
3.11.5	Zustands-LEDs der Funktionstasten.....	113
3.12	Systemparameter	113
3.12.1	Systemparameter Allgemeine Parameter.....	114
3.12.2	Systemparameter Pollbereich	114
3.12.3	Systemparameter Terminaluhr.....	114
3.12.4	Systemparameter Betriebsstundenzähler	114
3.12.5	Systemparameter Meldungssystem.....	114
3.12.6	Systemparameter Variantenpuffer.....	114
3.12.7	Systemparameter Paßwortverwaltung	114
3.12.8	Systemparameter Druckerschnittstelle	114
3.12.9	Systemparameter Gateway	115
3.12.10	Systemparameter Datensatztransfer.....	115
3.12.11	Systemparameter parallele Ausgänge.....	115
3.13	Versionsnummer	115
3.14	Betriebsstundenzähler	116
3.15	Parallelausgänge.....	117
3.16	Bildschirmschoner.....	118
3.17	Abbild der Maskennummer.....	118
3.18	Abbild des Betriebsartenschalters	118
3.19	Terminaluhr.....	118
3.19.1	Abbild von Datum und Uhrzeit	119
3.20	Koordinierungsbyte Lesen.....	120
3.20.1	Editieranforderungsbit EA-Bit.....	120
3.20.2	Editierzustandsbit EZ-Bit	120
3.20.3	Refreshanforderungsbit RA-Bit.....	120
3.20.4	Lebensmerker LM-Bit	121
3.20.5	Datensatz-Download aktiv DDA-Bit.....	121
3.21	Koordinierungsbyte Schreiben	121
3.21.1	Externe Datenfreigabe ED-Bit.....	122
3.21.2	Refreshquittierung RQ-Bit	122
3.21.3	Paßwort zurücksetzen.....	122
3.21.4	Lebensmerker LM-Bit	122
3.21.5	Datensatz-Download Freigabe DDF-Bit	122
3.22	Der zyklische Pollbereich.....	122
3.22.1	Byteorientiert.....	123
3.22.2	Wortorientiert	125
3.22.3	Abbild der Zustands-LEDs.....	125
3.22.4	Serieller Meldekanal	126
3.22.5	Pollzeit.....	126
3.22.6	Länge Pollbereich.....	126
3.23	Steuercodes.....	126
3.23.1	Datensatz ausdrucken lassen.....	126
3.23.2	Uhr im Bedienterminal stellen.....	127

3.23.3	Datensatztransfer von Steuerung zu Bedienterminal.....	127
3.23.4	Datensatztransfer von Bedienterminal zur Steuerung	127
3.23.5	Datensatztransfer von Steuerung zu Bedienterminal (einzeln)	127
3.23.6	Meldungssystem auffrischen.....	127
3.24	Zyklische Variablen	128
3.25	Schnittstellenparameter X2, X3	128
3.26	Variablenbeschreibung.....	128
3.26.1	Variablenformate (TSdos).....	128
3.26.2	Variablenliste	128
3.27	Anwendungsprogrammierung.....	130
3.27.1	Systemkonfiguration	130
3.27.2	Projektiersysteme TSdos und TSwIn	131
3.27.3	Einstieg in die Projektierung	134
3.27.4	Projektdokumentation	135
3.27.5	Projektsicherung.....	136
3.27.6	Optimierung der Übertragungsgeschwindigkeit.....	136
3.28	Download der Anwenderbeschreibung	137
3.28.1	Download mit Windows.....	138
3.28.2	Applikationsspeicher	139
3.28.3	Funktion Anwenderbeschreibung laden.....	139
3.28.4	Aktivieren der Downloadfunktion per Software	139
3.28.5	Aktivieren der Downloadfunktion per Hardware	140
3.28.6	Automatische Downloadfunktion.....	140
3.28.7	Downloadkabel	141
3.29	Simulation ohne Steuerung	142
5.22	TesiMod - CAN/CANopen	143
5.22.1	Allgemeines.....	143
5.22.2	Technische Beschreibung.....	143
5.22.3	Indirekte Prozeßdaten-Kommunikation	144
5.22.3.1	Ablauf vom Datenaustausch.....	144
5.22.4	Aufbau von Request- und Responsobjekten.....	144
5.22.5	Aufgaben vom Kommunikationspartner	147
5.22.6	Protokollparameter.....	147
5.22.7	Kommunikationsbeziehungen.....	148
5.22.8	Pollbereich	150
5.22.9	Physikalische Ankopplung.....	151
5.22.10	Fehlermeldungen.....	153
5.22.11	EUROTHERM spezifische Adressen und Definitionen.....	154
5.22.12	Beispiele für Kommunikationsbeziehungen.....	155
	Index	157
	Änderungsliste	160

1 TesiMod Bedienkonzept

1.1 Bedienterminals

Unsere Palette von Bedienterminals wird ständig erweitert und den Bedürfnissen am Markt angepaßt. Alle Geräte sind funktionell identisch und unterscheiden sich lediglich durch die Bauform, die Größe der Anzeige und die Anzahl der Funktionstasten. Aluminiumfrontplatten und verzinkte Stahlblechgehäuse erlauben den störsicheren Einsatz in rauher Industrieumgebung. Bei vielen Geräten wird durch die dichte Frontmontage die Schutzart IP65 erreicht.

Alle Geräte haben in umfangreichen Tests ihre Betriebssicherheit und absolute Industrietauglichkeit unter Beweis gestellt. Prüfungen aus den Bereichen mechanische Festigkeit, EMV, Temperatur und Klima stellen die hohe Qualität der Produkte sicher.

Alle Bedienterminals verfügen über gut ablesbare, mehrzeilige, hinterleuchtete oder selbstleuchtende Displays, die ein breites Anwendungsgebiet abdecken. Als Tastatur kommen entweder mechanische Kurzhubtasten oder Folientastaturen zum Einsatz, die alle über einen definierten Druckpunkt verfügen. Funktionstasten mit integrierten LEDs sparen externe Schalterelemente und Kontrollampen ein. Alle Funktionstasten verfügen über frei beschreibbare Einschubstreifen.

Optional können mit 8 beliebigen Funktionstasten optokoppelte 24 V Parallelausgänge geschaltet werden. Die Ausgänge können direkt zur Ansteuerung von SPS-Eingängen verwendet werden. Die Ausgänge haben den Vorteil einer sehr kurzen Reaktionszeit (<25ms), da sie von der seriellen Übertragung unabhängig sind.

Die Anschaltung der Terminals an die Steuerung erfolgt generell über eine serielle Schnittstelle. Eine zweite serielle Schnittstelle ist für den Druckeranschluß und die Downloadfunktion vorhanden. Die Modularität der Hardware erlaubt die Anpassung an verschiedene Schnittstellenstandards. Im Lieferumfang sind je eine 20 mA Stromschleife und eine RS232-Schnittstelle enthalten. Die Schnittstellen sind gegeneinander und gegen die interne Elektronik galvanisch getrennt. Als Alternativen bieten wir Schnittstellenmodule nach der Norm RS422 oder RS485 an. Ebenso ist die Ankopplung an alle gängigen Feldbussysteme möglich. Je nach Schnittstellenstandard und eingestellter Baudrate können Leitungslängen bis ca. 1000 m realisiert werden.

1.1.1 Ladbare Protokolltreiber

Zur Reduzierung der Gerätevielfalt bieten wir, optional an der Schnittstelle X2, die am häufigsten zum Einsatz kommenden Schnittstellenstandards, 20 mA Stromschleife und RS485 gleichzeitig an. Die Schnittstellen sind dann jeweils mit X2.1 und X2.2 gekennzeichnet. In Verbindung mit den ladbaren Protokolltreibern werden die Bedienterminals in Hard- und Firmware von der Steuerung unabhängig. Mit einem Gerät kann wahlweise mit allen derzeit unterstützten Steuerungen ein Datenaustausch durchgeführt werden. Die Geräte müssen, außer für Feldbusan Kopplungen, nicht protokollspezifisch bestellt werden.

Alle Terminals können zur Ausgabe, Eingabe und Änderung von Daten in SPS-Steuerungen, Mikroprozessorsteuerungen, Prozeßrechnern oder PCs eingesetzt werden.

Die Kommunikation wird entweder über das steuerungsspezifische Protokoll oder ein Standardprotokoll abgewickelt. Bei den steuerungsspezifischen Protokollen ist der Zugriff auf alle System- und Anwenderdaten sowie Ein-/ Ausgänge, Merker, Timer, Zähler usw. gewährleistet. Bei SPS-Steuerungen erfolgt die Anschaltung vorzugsweise über die Programmiergeräteschnittstelle.

1.1.2 Vernetzung von Terminals

Soweit es die steuerungsspezifischen Übertragungsprotokolle erlauben, wird eine Mehrpunkt-Verbindung unterstützt. Bei einigen Protokollen können mehrere CPUs im Netz oder im

Baugruppenträger adressiert werden. Nur bei Feldbusankopplungen besteht die Möglichkeit, mehrere Bedienterminals an einer SPS zu betreiben.

Für andere Protokolle bietet sich eine preiswerte Lösung über den DIN-Meßbus nach DIN66348 an. Bei dieser Mehrpunktverbindung fungiert ein Bedienterminal als Busmaster und Gateway (Protokollumsetzer) zur Steuerung. Alle weiteren Bedienterminals arbeiten als DIN-Meßbus-Slave-Teilnehmer. Die Adressierung der Daten erfolgt in der Syntax der angeschlossenen Steuerung. Über den Bus wird die volle TesiMod-Standard-Funktionalität erreicht. Der Datenengpaß zur Steuerung wird durch geeignete Maßnahmen kompensiert. Als Busmaster kann jedes Bedienterminal verwendet werden. Der Busmaster verliert jedoch die Funktionalität der Druckerfunktion auf der Schnittstelle X3. Diese Schnittstelle wird als Busanschaltung mit RS485-Schnittstelle verwendet. Je nach Terminaltyp ergeben sich dafür unterschiedliche Lösungen.

1.1.3 Schnittstelle für Programmiergeräte

Da Hardware und Protokollsoftware für die Programmierung bereits bei jeder SPS vorhanden sind, kann auf einen zusätzlichen Kommunikationsprozessor verzichtet werden. Das ergibt für den Anwender eine preisgünstige Lösung. Der Programmierer muß keine Kommunikationssoftware für seine SPS erstellen und einbinden, wie es bei einer Kommunikationsbaugruppe erforderlich wäre. Die Protokollsoftware ist fester Bestandteil des SPS- und Terminalbetriebssystems.

1.1.4 Was ist TesiMod und was kann damit erreicht werden?

Unter der Bezeichnung TesiMod stellt die Firma EUROTHERM ein Bedienkonzept vor, das alle technischen Merkmale für eine zukunftsweisende Bedientechnologie enthält. Ein einheitliches und funktionell durchgängiges Bedienkonzept, das die angeschlossene Steuerung von Bedienaufgaben, wie Bedienerführung und Darstellung der Daten, vollständig entlastet.

1.1.5 Betriebsarten

In allen TesiMod-Bedienterminals stehen die Betriebsarten Standard- und Transparent-Mode zur Verfügung.

1.1.5.1 Transparent-Mode

Im Transparent-Mode können TesiMod-Bedienterminals als vollwertige ANSI-Terminals eingesetzt werden. Jede Taste erzeugt einen Drück- und Loslaßcode, der über die Schnittstelle gesendet wird. Die Ansteuerung des Displays und der LEDs erfolgt über ESC-Sequenzen. Es stehen je nach Displaytyp verschiedene Zeichensätze und -attribute zur Verfügung.

1.1.5.2 Standard-Mode

Erst im Standard-Mode entwickelt das TesiMod Bedienterminal seine volle Leistungsfähigkeit. Im Standard-Mode ist keine Tastendekodierung oder Menüsteuerung durch die angeschlossene Steuerung erforderlich. Im Standard-Mode kann eine Bedienerführung ohne großen Steuerungsaufwand erzeugt werden, inklusive Paßwortverwaltung, Rezepturverwaltung, Skalierung von Variablenwerten, Grafischen Elementen, Tabellen und nicht zuletzt auch Meldungen. Dabei kann der Bediener durch Softkeys, Auswahlmenüs, Funktionstasten oder durch die SPS (zwangsweise) durch die Bedienstruktur geführt werden.

1.1.6 Anwenderbeschreibung

Die Projektierungsdaten der Anwenderbeschreibung bestehen im wesentlichen aus drei Arten von Daten.

Globale Daten, sie gelten für das gesamte Projekt.

Sprachspezifische Daten, sie gelten nur für eine bestimmte Sprache innerhalb eines Projekts.

Steuerungsspezifische Daten, sie gelten nur für die ausgewählte Steuerung innerhalb eines Projekts.

Globale Daten sind:

- Systemeinstellungen
- Grafiken

Sprachspezifische Daten sind:

- Maskenbeschreibung
- Textlisten
- Hilfetexte (Hilfemasken)

Steuerungsspezifische Daten sind:

- Kommunikationseinstellungen
- Variablenliste

Durch die Aufteilung der Projektdaten in drei Teilbereiche wird es dem Anwender sehr leicht gemacht, das Projekt mit einer Maskenbeschreibung in einer zusätzlichen Sprache zu erweitern oder den angeschlossenen Steuerungstyp bzw. Steuerungshersteller zu wechseln.

Die gesamte Anwenderbeschreibung wird mit Hilfe einer komfortablen PC-Software (TSDos oder TSwIn) selbst erstellt.

Die Anwendungsbeschreibung erscheint für den Bediener später nur in Form von Texten, Meldungen, Tabellen, Diagrammen, Grafiken und Werten, die am Bedienterminal angezeigt werden. Die angezeigten Elemente werden jeweils für eine Displayseite zusammengestellt. Diese Displayseite wird Maske genannt. Bei TSDos sind für unterschiedliche Anforderungen entsprechende Maskentypen entwickelt worden. Bei TSwIn wird die Funktionalität einer Maske erst durch den Inhalt bestimmt.

1.1.7 Speicherung der Anwenderbeschreibung

Die anwenderspezifische Beschreibung der gesamten Bedien- und Terminalfunktion wird in einem FLASH-Speicher gesichert. Schreiben und Löschen dieses Speichers erfolgt direkt im Terminal, ohne daß dazu ein Programmiergerät benötigt wird. Die Programmierung ist im eingebauten Zustand möglich, ohne daß das Gerät geöffnet werden muß.

Der FLASH-Speicher kann jederzeit durch ein normales UV-EPROM ersetzt werden. Die Funktionalität programmieren und löschen ist für UV-EPROMs im Gerät nicht möglich. Der FLASH-Speicher stellt eine netzausfallsichere Speicherung von Masken und Bedienerführung ohne Batterie dar!

1.1.8 Adressierung der Variablen

Die Variablenliste dient als Referenzliste zur Anbindung an die Steuerungshardware. Hier werden den symbolischen Namen der Variablen die Hardwareadressen wie Merker, Eingänge, Ausgänge oder Datenworte zugeordnet. Die Zuordnung erfolgt in der Originalschreibweise des Steuerungsherstellers.

1.1.9 Datenfreigabe

Um eine Werteänderung am Bedienterminal durchführen zu können, muß eine Datenfreigabe erteilt worden sein. Die Datenfreigabe kann für alle Variablen einer Maske automatisch aktiviert werden (Maskenparameter).

Wird die Datenfreigabe nicht automatisch aktiviert, dann muß sie vom Bediener angefordert werden. Dazu wird die Taste Datenfreigabe am Bedienterminal gedrückt. Die Zustands-LED der Datenfreigabe zeigt dann an, ob eine Datenfreigabe erteilt wurde (LED leuchtet) oder nicht (LED blinkt).

Die Datenfreigabe kann von der Steuerung verweigert werden (externe Datenfreigabe). Dazu müssen Variablen für den Pollbereich (Koordinierungsbyte Schreiben) und das Koordinierungsbyte Lesen eingerichtet werden, in denen entsprechend Bits gesetzt, beziehungsweise zurückgesetzt werden müssen.

1.1.10 Paßwortschutz

Zum Schutz von Daten und Maskeninhalten können bis zu acht Paßworte vergeben werden. Jedem Paßwort ist jeweils ein View- und Editlevel zugeordnet. Jeder Maske wird eine Sperrebene zugeordnet. Dadurch können Masken oder ganze Menübäume benutzerspezifisch gesperrt und freigegeben werden. Alle Informationen innerhalb einer Maske haben die gleiche Zugangsberechtigung. Die Zugangsberechtigung der Paßworte wird in der Projektiersoftware vorgegeben. Die Berechtigung im Bedienterminal wird beim Initialisieren des Terminals, vom Bediener, aus der Steuerung oder durch die Eingabe eines falschen Paßworts rückgesetzt. Default-Paßworte und Sperrebene werden bei der Programmierung der Maskenbeschreibung festgelegt. Paßworte können im Bedienterminal geändert werden. Ein definierbares Masterpaßwort erlaubt die Rücksetzung auf die Grundwerte. Ohne die Programmierung des Paßwortschutzes sind alle Masken und Variablen zugänglich.

1.1.11 Editoren

Die Modifikation von Daten in den Masken ist nicht allein auf die Variablen der Steuerung beschränkt, es können auch terminalspezifische Systemvariablen geändert werden.

Für jeden Datentyp steht ein entsprechend leistungsfähiger Editor zur Verfügung. Alle Variablen werden bei der Eingabe auf Plausibilität geprüft. Der Anwender legt die Bereichsgrenzen in der Variablenbeschreibung fest und kann außerdem einen variablenspezifischen ganzseitigen Hilfetext definieren. Der Hilfetext ermöglicht die Funktion der Variablen und deren zulässiger Wertebereich zu beschreiben.

Die Eingabe von Variablen geschieht wie folgt: Befindet sich in einer E/A-Maske ein editierbarer Wert, so erlaubt die Taste Datenfreigabe in den Editor zu wechseln (unter der Annahme, daß Paßwort- und externe Datenfreigabe erfolgte).

Im Editiermode leuchtet die Zustands-LED der Taste Datenfreigabe, der Cursor steht auf der ersten editierbaren Variablen. Nun läßt sich dieser Wert ändern. Hierzu stehen die numerische Tastatur mit Vorzeichen, Dezimalpunkt und Taste Löschen zur Dateneingabe bereit. Der Wert wird mit der Taste Datenübernahme übernommen. Dabei wird die Plausibilität geprüft und gegebenenfalls eine Systemmeldung erzeugt.

Besonders leistungsfähige Editoren wie der Tabelleneditor und der Auswahlfeldeditor erlauben es auch mit größeren Datenmengen umzugehen. Beim Tabelleneditor können indizierte Variablen spaltenweise editiert werden. Für den Editortyp der Spalte kann aus der Vielzahl der Einzeleditoren wie Festkomma, Gleitkomma, BCD, Hexadezimal usw. gewählt werden. Ebenso dürfen Variablengröße und Variablentyp spaltenweise unterschiedlich sein.

Leistungsfähige Editoren benötigen zusätzliche Tastenfunktionen wie PgUp, PgDn, Spalte links, Spalte rechts usw. Diese werden über Systemvariable und die Softkeys individuell erzeugt.

1.1.12 Hilfe

Jeder Maske und jeder editierbaren Variable kann ein Hilfetext von der Größe einer Displayseite zugeordnet werden. Ist kein spezieller Hilfetext programmiert, wird ein Standardhilfetext (Default-Hilfetext), der für das gesamte System gültig ist, ausgegeben. Der Standardhilfetext kann in der TS-Software für Variablen und Masken vorgegeben werden.

Eine blinkende Zustands-LED der Taste Hilfe weist den Bediener auf eine Fehlfunktion oder eine eingegangene Meldung hin. Beim Betätigen der Taste Hilfe wird die anstehende Systemmeldung mitgeteilt, beispielsweise ob der Wertebereich nach oben oder unten überschritten wurde. Das Editierfeld kann bei einem Fehler mit der Taste Datenübernahme nicht verlassen werden.

Mit der Taste Datenübernahme wird nur ein gültiger Wert eingelesen.

Dagegen verlassen Steuertasten auch dann ein Editierfeld, wenn der Wert ungültig ist, allerdings wird der zuletzt eingegebene Wert durch den alten Wert (Vorzustand) ersetzt. Die Steuertasten dienen im Editiermode zur Anwahl der editierbaren Felder. Der Editiermode wird beim erneuten Betätigen der Taste Datenfreigabe, die Zustands-LED erlischt, wieder verlassen.

1.1.13 Funktionstasten

Neben den Masken spielen die Funktionstasten und deren Zustands-LEDs im TesiMod-Bedienkonzept eine wichtige Rolle. Durch einen Einschubstreifen können die Funktionstasten frei beschriftet oder bedruckt werden. Im Standard-Lieferumfang sind bedruckte und unbedruckte Einschubstreifen enthalten. Die Funktionstasten können vom Anwender programmiert werden. Sie können als Direktanwahltasten für Maskenwechsel und/oder zum Steuern in der Maschine verwendet werden. Für die Funktionsrückmeldung dienen integrierte LEDs, die von der Steuerung beeinflusst werden können.

Sind Funktionstasten als Direktanwahltasten programmiert, ermöglichen sie den direkten Zugang zu Masken oder kompletten Menüstrukturen die sich dahinter verbergen. Außerdem ermöglichen sie dem erfahrenen Bediener eine schnellere Bedienung im Vergleich zur Menüstruktur.

Funktionstasten können immer auch mit Standardbelegungen für eine komplette Maskenbeschreibung vordefiniert werden. Dies schließt eine parallele Benutzung als Softkey nicht aus. Die Definition als Softkeys hat eine höhere Priorität als die Standardfunktion.

Eine weitere Hilfe zur schnelleren Bedienung stellt der Editor zur Verfügung, er merkt sich die zuletzt editierte Stelle in der Maske und kehrt beim erneuten Aufruf der Maske an diese Stelle wieder zurück.

1.1.14 Softkeys

Alle Funktionstasten können auch als Softkeys verwendet werden. Softkeys können wechselnde Funktionen innerhalb einer Maske erfüllen. Die aktuelle Funktion muß am Display angezeigt werden. Zum Beispiel könnte mit dem gleichen Softkey eine Pumpe ein- und wieder ausgeschaltet werden.

1.1.15 Variablen

Welche Variablentypen Bit, Byte, Wort, Doppelwort oder Feld möglich sind, ist von der angeschlossenen Steuerung abhängig. Das Konzept erlaubt Schreib- und Lesezugriffe auf alle in einer Steuerung vorhandenen Datentypen, soweit nicht anders deklariert.

Die Aus- und Eingabe einer Variablen ist formatier- und skalierbar. An Datenformaten stehen Binär, Ganzzahl, Festkomma, Gleitkomma, Alphanumerisch und Auswahlfeld zur Verfügung. Alle Datenformate und Variablentypen sind für die Ein- und Ausgabe zulässig. Durch die Skalierung wird eine einfache Anpassung von Millimeter in Zoll, Grad Celsius in Grad Fahrenheit usw. ermöglicht.

Die Eingabe kann in den unterschiedlichsten Formaten erfolgen. Die Ausgabe einer Variablen kann einmalig oder zyklisch erfolgen. Editierbar gekennzeichnete Variablen werden ebenfalls einmalig oder zyklisch ausgegeben. Die Zykluszeit läßt sich über Systemparameter im Projektiersystem einstellen.

1.1.16 Grafik

Das Einsatzgebiet der grafischen Anzeigen reicht bis zur einfachen grafischen Prozeßvisualisierung. Hier spielen erweiterte Diagnosemöglichkeiten, die leichtere und übersichtlichere Prozeßkontrolle und die Sprachunabhängigkeit der Symbole eine wichtige Rolle.

Teilweise können die aufwendige Übersetzungen in andere Landessprachen entfallen. Grafik ermöglicht eine Bedienerführung durch Piktogramme, es können von den Prozeßvariablen direkt grafische Abbilder erstellt werden. Die grafische Anzeige gibt, wo keine exakten numerischen Daten erforderlich sind, eine schnelle Information anhand der unterstützten Trendanzeigen. Reaktionen vom Bediener, von der Steuerung und aus dem Prozeß können leicht verständlich visualisiert werden. Die eingesetzte Grafik liefert einen schnellen Überblick über die Prozeßzustände in der Anlage oder in

Anlagenteilen. Die Fortsetzung dieser Möglichkeiten führt im weitesten Sinne zu einer textlosen, symbolischen Bedienerführung.

Nachstehend ein paar Beispiele wo Grafik eingesetzt wird:

- allgemein grafisches Abbild von Prozeßgrößen
- Füllstandsanzeigen
- Störstellenanzeige im Maschinenabbild
- darstellen einer Häufigkeitsverteilung
- Temperaturverläufe
- Einschwingvorgänge
- Trendanzeige
- Darstellung des Firmenlogos
- einheitlicher Maskenhintergrund

Alle Texte, Variablen und Grafiken können frei plaziert werden.

Grafiken können mit allen Programmen, die von Windows95 oder WindowsNT unterstützt werden, erstellt werden. Dabei hat der Anwender die Möglichkeit zwischen eingebettetem Objekt oder separat verwaltetem Objekt zu wählen.

1.1.17 Rezepturen und Datensätze

Mit Rezepturen werden die Zusammenfassungen von Texten, Variablen und Einheiten bezeichnet. Die Werte für die Variablen werden als Datensätze separat gespeichert.

Beispiel: Rezeptur Holzrahmen sägen; Datensatz Buche

Text	Variable	Einheit
Drehzahl	1000	min ⁻¹
Vorschub	10	mm/s
Schnittwinkel	35	°

Bei der Erstellung von Rezepturinhalten bestehen grundsätzlich die gleichen Möglichkeiten wie bei Masken.

Die theoretischen Grenzen in einem Bedienterminal liegen im Bereich von max. 250 Rezepturen.

Eine Rezeptur darf maximal 250 Datensätze umfassen. Jeder Datensatz darf bis zu

- 255 Zeilen mit
 - 255 Variablen und
 - 255 Textelementen
- beinhalten.

Die Grenzen des Systems werden außerdem durch den zur Verfügung stehenden Speicher gebildet.

Gegenüber Masken unterscheiden sich Rezepturen dadurch, daß sie unabhängig von ihrer Länge innerhalb eines wählbaren Rezepturfensters in einer E/A-Maske editiert werden können. Die Position und Größe des Rezepturfensters ist in der TS-Software einstellbar.

Neben der Rezeptur können weitere Ein- und Ausgabevariablen in der Maske enthalten sein. Rezepturen dürfen nur Eingabevariablen enthalten. Die Ablage der Variablen erfolgt im Bedienterminal im gepufferten CMOS-RAM. Die Eingabevariablen unterscheiden sich nicht von Maskenvariablen. Auch hier können die Variablen mit Hilfetexten, Formatierung und Skalierung versehen werden.

In der TS-Software vordefinierte Datensätze werden im Flash gespeichert und können für Änderungen ins RAM kopiert werden. Datensätze können im Bedienterminal neu erstellt werden, gelöscht, kopiert und von und zur SPS transferiert werden. Die Verwaltung der Datensätze im Terminalspeicher erfolgt dynamisch, so daß eine optimale Ausnutzung des zur Verfügung gestellten Speicherbereichs erfolgt.

Die Übertragung der Datensätze zur Steuerung kann über einen Kommunikationspuffer oder direkt erfolgen. Die Übertragung des Datensatzes ins Bedienterminal erfolgt immer über einen Kommunikationspuffer. Die Verwaltung der Rezeptur und die Adressierung der Variablen erfolgt

sprachunabhängig. Die Texte in der Rezeptur werden der sprachabhängigen Maskenbeschreibung entnommen. Die Datensätze einer Rezeptur lassen sich über die Schnittstelle X3 auf einen Drucker ausgeben. Mit Hilfe von Systemvariablen kann auch ein Datensatztransfer vom PC zum Bedienterminal und vom Bedienterminal zum PC erreicht werden.

1.1.18 Meldungen

Meldungen sind Betriebszustände, die an das Bedienterminal übermittelt werden. Die eingetroffenen Meldungen werden entweder nach ihrer Priorität (Meldungsnummer) oder chronologisch im Meldungsspeicher abgelegt.

In der momentanen Softwareversion können bis ca. 3000 Meldungen im Meldungsspeicher verwaltet werden. Die maximale Größe des Meldungsspeichers kann in der TS-Software voreingestellt werden. Meldungen, Rezepturen und ladbare Zeichensätze teilen sich den CMOS-RAM-Speicher auf.

Meldungen können in zweifacher Weise in die Bedienterminals übertragen werden. Es wird dabei zwischen einer parallelen und einer seriellen Übertragung unterschieden. Beide Systeme sind gleichberechtigt nebeneinander verwendbar. Neben einem Meldungstext können auch ganzseitige Meldungen ins System integriert werden. Ganzseitige Meldungen können wie Masken erstellt und behandelt werden.

In jedem Meldetext darf eine formatierte und skalierte Variable enthalten sein. Der Variablenwert wird beim Auftreten der Meldung eingefroren.

Der Inhalt des Meldungsspeichers kann komplett, blockweise oder zeilenweise bearbeitet werden. Die zweite Schnittstelle erlaubt die Ausgabe der kompletten Meldung mit Meldungsnummer, Datum, Uhrzeit, Meldetext und Variablenwert auf einen Drucker. Der Ausdruck kann direkt oder zu einem späteren Zeitpunkt erfolgen.

1.1.19 Systemmeldungen

Systemmeldungen werden bei verschiedenen Überwachungsaufgaben vom Bedienterminal generiert. Die textlichen Inhalte werden vom Anwender gestaltet. Es steht jeweils eine Displayseite zur Verfügung. Die Aktivität einer Systemmeldung wird durch das Blinken der Zustands-LED der Taste Hilfe angekündigt. „Leere“ Systemmeldungen werden vom System automatisch unterdrückt.

1.1.20 Projektiersystem

Damit die vielfältigen Möglichkeiten unserer Terminals für Sie einfach nutzbar werden, bieten wir Ihnen die sehr komfortable und einfach handhabbare Projektieroberfläche TSwIn an. Das Programm ist auf allen PCs mit Windows95 oder WindowsNT lauffähig.

Mit diesem Programm lassen sich Ihre Projekte leicht und übersichtlich erstellen und verwalten, sollten dennoch Fragen auftauchen steht ihnen ein umfangreiches Hilfesystem und unsere Hotline zur Verfügung. Bereits bei der Programmerstellung sehen sie die Maske im späteren Displayformat. Die Programmierung und die Funktionalität ist für alle Terminals gleich.

Neben einer vollgrafischen Maskenerstellung enthält das Programmiersystem eine Projektverwaltung, einen Compiler, eine Dokumentationserstellung und ein Datenübertragungsprogramm für die Download-Funktion.

Die Beschreibung der Masken und die Darstellung der Variablen erfolgt protokollunabhängig. Lediglich durch eine Referenzliste der Variablenbeschreibung wird der Bezug zur Ziel-SPS hergestellt.

Bei der Erstellung der Maskenbeschreibung, durch die Programmiersoftware, kann das Terminal auch ohne Ziel-SPS betrieben werden. Dadurch lassen sich Masken und Bedienoberfläche direkt im Terminal kontrollieren und testen.

Für Ihre ersten Versuche bieten wir Ihnen Mustergeräte an.

Haben Sie noch Fragen ?

Sollten sich Fragen ergeben, stehen wir Ihnen mit weiteren Auskünften gerne zur Verfügung.

Rufen Sie unsere Hotline an 07253-9404-0 !

1.2 Gemeinsame Merkmale

1.2.1 Hardware

- mehrzeilige, hinterleuchtete oder selbstleuchtende Displays
- Filterscheiben zur Entspiegelung und Kontrasterhöhung
- Temperaturkompensation der Anzeige (falls erforderlich)
- Funktionstasten mit integrierter grüner LED
- beschreibbare Einschubstreifen für die Funktionstasten
- Editiertasten
- Steuertasten
- im Gerät programmierbare Maskenspeicher (Flash-Speicher)
- gepuffertes RAM für Meldungsspeicher und Rezepturdatenverwaltung
- mindestens zwei serielle Schnittstellen
- verschiedene Schnittstellenstandards
- Batteriespannungsüberwachung
- Echtzeituhr mit Datum, Uhrzeit
- Betriebsartenschalter

1.2.2 Software

- einheitliche Funktionalität der Terminals
- zwei Betriebsarten: Standard- und Transparent-Mode
- hierarchisches Bediensystem mit Direkttasten
- Softkey-Funktionalität auf allen Funktionstasten
- Paßwortschutz mit verschiedenen Zugriffsebenen
- integriertes Hilfesystem
- alle Variablenformate
- Pixelgrafik: Hintergrundbilder, Balkendiagramme, Wechselbild und Kurvendarstellung
- alle Terminals mit Rezepturen
- ladbare Protokolltreiber für alle wichtigen Steuerungsanbieter
- protokollunabhängige Hardware
- Kommunikationsüberwachung
- Anschluß mehrerer Bedienterminals an der PG-Schnittstelle
- BUS-Anschaltungen für Feldbussysteme und DIN-Meßbus
- einheitliche Projektiersoftware für alle Terminals

1.3 Bestellhinweise

Rufen Sie bitte unsere Verkaufsabteilung unter der Telefonnummer 07253-9404-0 an oder senden Sie uns ein Fax unter der Nummer 07253-9404-99 oder eine Email an info@eurotherm.de, wenn Sie eine Bestellung auslösen möchten oder einfach nur eine Frage zu einem Produkt haben.

2 Betriebsarten

In allen TesiMod-Bedienterminals stehen zwei Betriebsarten, der Standard- und der Transparent-Mode zur Verfügung.

Im Transparent-Mode können TesiMod-Bedienterminals als vollwertige ANSI-Terminals eingesetzt werden. Jede Taste erzeugt einen Drück- und Loslaß-Code, der als ASCII-Zeichen über die serielle Schnittstelle gesendet wird. Die Ansteuerung des Displays und der Tasten-LEDs erfolgt über ESC-Sequenzen. Es stehen je nach Displaytyp verschiedene Zeichensätze und -attribute zur Verfügung. Die detaillierte Beschreibung befindet sich im Kapitel "TesiMod Transparent-Mode".

In der Betriebsart Standard-Mode wird das komplette Bediensystem durch TesiMod in das Terminal integriert. Das einheitliche Bedienkonzept erlaubt es, die angeschlossene Steuerung von Bedienerführung und Darstellung der Daten vollständig zu entlasten. Im Standard-Mode ist keine Tastendekodierung oder Maskenanwahl aus der Steuerung erforderlich.

2.1 Einstellen der Betriebsart

Die Betriebsart kann am Betriebsartenschalter (Usermode Switch) eingestellt werden. Die Geräte werden im Standard-Mode ausgeliefert.

Die Einstellung der Betriebsart erfolgt auf der Geräterückseite nach Öffnen des Klappdeckels oder der Gehäuserückwand.



Schalter

ON

OFF

S1

Standard-Mode

Transparent-Mode

S2

frei

frei

S3

Demo (ohne SPS)

Kommunikation mit der SPS

S4

Erase Flash bedingungslos

Flash-Daten erhalten

ggf. S5..S8

frei

3 TesiMod Standard-Mode

Die Betriebsart Standard-Mode gehört neben dem Transparent-Mode zum Leistungsumfang von Bedienterminals mit TesiMod-Bedienkonzept. Wie der Name es bereits andeutet, ist es die Betriebsart, in welcher die Terminals am häufigsten zum Einsatz kommen und in der sie auch ihre höchste Leistungsfähigkeit erbringen. In dieser Betriebsart werden die Terminals zu eigenintelligenten Peripheriegeräten der Steuerung, die eine kontrollierte Vorverarbeitung der Visualisierungsdaten durchführen. Die angeschlossene Steuerung wird von der Bearbeitung der Bedienaufgaben vollständig entlastet. Dies reduziert den Programmieraufwand des Anwenders, die Laufzeit und den Speicherbedarf in der SPS.

Des weiteren wird durch den Verzicht auf Kommunikationsprozessoren in der SPS und durch die konsequente Nutzung von Programmiergeräteschnittstellen eine kostengünstige SPS-Anschaltung erreicht. Die Vielzahl der unterstützten Protokolle machen das Bedienkonzept universell einsetzbar. Es wird erreicht, die Maschinenbedienung unabhängig von Steuerungstyp und -fabrikat zu realisieren.

Der Standard-Mode wird von jedem Bedienterminal unterstützt. Die Flexibilität des Systems erlaubt es, die Bedienerführung ihrer Anwendung anzupassen. Mit Hilfe von Masken, Systemvariablen, Steuertasten und Funktionstasten lassen sich beliebig komplexe Bediensysteme aufbauen. Im Standard-Mode werden alle mit dem Terminal ausführbaren Funktionen, Maskeninhalte, Texte, Meldungen und Variablen in der Anwenderbeschreibung (Maskenbeschreibung) gespeichert. Die Anwenderbeschreibung wird nach Fertigstellung des Projektierens im EPROM oder Flash-Speicher des Bedienterminals abgelegt.

In der weiteren Beschreibung wird der Begriff "*Anwender*" für den Projektant oder Programmierer der Bedienoberfläche verwendet und "*Bediener*" steht für eine Person, die vor Ort Daten der Anlage beobachtet und bedient. Des weiteren wird im folgenden Text immer wieder zwischen TSdos und TSwin unterschieden, da der projektierbare Funktionsumfang unterschiedlich ist. Zur Vereinfachung werden im folgenden Text alle Systemvariablen ohne die Vorsilbe „Sys“ geschrieben.

3.1 Betriebsart einstellen

Die Betriebsart Standard-Mode kann am Betriebsartenschalter eingestellt werden. Die Einstellung erfolgt im ausgeschalteten Zustand des Terminals.

Die Position des Betriebsartenschalters wird im Handbuch des jeweiligen Bedienterminals beschrieben.

Alle Geräte werden mit der Einstellung Standard-Mode ausgeliefert!

Die Stellung ON/OFF ist am Betriebsartenschalter gekennzeichnet.

Sobald am Terminal Spannung angelegt wird, ist die gewählte Betriebsart aktiv.

Schalterstellung für Standard-Mode:

S1	ON	S5	frei
S2	frei	S6	frei
S3	OFF	S7	frei
S4	OFF	S8	frei

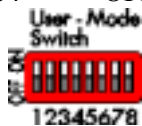


Bild 1: Betriebsartenschalter

3.2 Verhalten der Terminals beim Einschalten

Beim Anlegen der Versorgungsspannung werden zuerst alle LEDs im Terminal eingeschaltet. Im Anschluß daran erfolgt ein Systemtest, der die Überprüfung und Initialisierung der im Bedienterminal befindlichen Module beinhaltet. Vom Systemtest können verschiedene System- und Fehlermeldungen ausgegeben werden. Unter der Voraussetzung, daß sich eine gültige Anwenderbeschreibung im Applikationsspeicher befindet, erscheint auf dem Display als erste Maske, die "Start-Maske" (TSdos), beziehungsweise die Maske, die in den Sprachparametern als Maske für Startup eingetragen wurde (TSwin).

Die Ausgabzeit dieser Maske beträgt 5 Sekunden. Diese Zeit ist fest eingestellt. Während dieser Zeit kann der Bediener die Funktion der LEDs und der Anzeige kontrollieren. Nach Ablauf der Zeit erscheint auf dem Display die „Hauptmaske“(TSdos), beziehungsweise die in den Sprachparametern als Hauptmaske eingetragene. Diese Maske ist gleichzeitig die erste Maske des Bediensystems.

Wird während der Anzeige der „Start-Maske“, die Taste **Datenübernahme** betätigt, erscheint auf dem Display die „Setup-Maske“. In dieser Maske erfolgt die Parametrierung der Schnittstellen und des Bedienterminals.

Wird die Taste Datenübernahme betätigt, bevor die „Start-Maske“ erscheint, wird eine Fehlermeldung beim Überprüfen der Tastatur erzeugt.

```
KEYBOARD ERROR
PLEASE RELEASE KEY
.....
```

Beim Einschalten des Terminals hat der Selbsttest die Tastatur überprüft und dabei eine betätigte Taste erkannt. Kommen sie der Aufforderung nach und lassen die Taste los. Erfolgt die Meldung ohne daß eine Taste betätigt wurde, so deutet dies auf einen Tastatur-Defekt hin. Normalerweise verschwindet beim Loslassen der Taste(n) diese Meldung.

Die Ausgabe der „Start-Maske“ kann sich verzögern, wenn sehr viele Meldungen des seriellen Meldesystems im Statistikspeicher enthalten sind. Die Zeit wird benötigt, um die Strukturen für die Meldungsverwaltung aufzubauen. Durch diese Initialisierungszeit wird das spätere Einsortieren der Meldungen wesentlich beschleunigt. Während dieser Zeit erscheint auf dem Display die Mitteilung:

```
INITIALIZING
MESSAGE BUFFER
.....
```

Beim Einschalten des Terminals werden die im Terminal vorhandenen Meldungen sortiert, dies benötigt eine bestimmte Initialisierungszeit beim Einschalten.

3.2.1 Verhalten ohne gültige Anwenderbeschreibung

Beim Systemtest wird bei der Überprüfung des Applikationsspeichers ein fehlendes, falsches oder defektes Bauteil erkannt und durch die Meldung:

```
NO FLASH EPROM .....  
.....  
.....
```

dem Bediener mitgeteilt.

Wird bei der Speicherprüfung ein korrekter Flashspeicher erkannt, dessen Inhalt jedoch nicht einer gültigen Anwenderbeschreibung entspricht, so wird der Flash-Speicher gelöscht und das Terminal wechselt automatisch in den Download-Betriebszustand. Dabei werden auf dem Display folgende Phasen durchlaufen:

```
ERASE FLASH EPROM .....  
.....  
.....
```

```
FLASH IS ERASED .....  
FLASH 256 kBYTE .....  
HF XXXXX .....
```

```
DOWNLOAD .....  
.....  
.....
```

Auf der Anzeige bleibt die Meldung „Download“ (TSdos) oder „DOWNLOAD 1“ (TSwin) stehen. Die Meldung bedeutet, daß das Terminal nun bereit ist, über die Schnittstelle X3 eine gültige Anwenderbeschreibung zu empfangen.

Solange keine gültige Anwenderbeschreibung im Speicher vorhanden ist, findet keine Kommunikation zu einer an der Schnittstelle X2 angeschlossenen Steuerung statt, ebenso führt die Tastatur keine Funktion aus. Eine Aktivität des Terminals erfolgt erst nach dem Download einer gültigen Anwenderbeschreibung.

3.3 Kommunikation im Standard-Mode

Die Kommunikation zwischen SPS (Hostrechner, etc.) und Bedienterminal kann im Standard-Mode mit jeder Schnittstelle, außer denen für den Protokolldrucker und die Parallelausgänge, erfolgen. Dabei ist die Benutzung einer Schnittstelle immer von der angeschlossenen Gegenseite oder dem Netzwerk abhängig.

Die einzelnen Schnittstellen sind in den entsprechenden Kapiteln der Handbücher zum Bedienterminal näher beschrieben. Für die möglichen Ankopplungen an unterschiedliche SPS-Typen und Netzwerke gibt es detaillierte Beschreibungen in den Kapiteln 5.x *Steuerungs- und Busankopplungen* dieses Handbuchs.

Um eine sichere Ankopplung bieten zu können, gibt es als Zubehör für jede Ankopplungsmöglichkeit ein ca. 3 m langes Standardkabel.

3.4 Das Bedienkonzept

Das Bedienkonzept der TesiMod-Bedienterminals ermöglicht es dem Bediener einen schnellen und übersichtlichen Zugriff auf alle Masken und damit auf alle Daten zu bekommen.

Dabei kann der Anwender den Bediener mit allerlei Hilfestellungen durch das System von hierarchisch angeordneten Masken leiten.

3.4.1 Hierarchische Maskenstruktur bei TSDos

Aus den Basiselementen der Knotenmaske und E/A-Maske können hierarchische Bediensysteme aufgebaut werden. Neben der Auswahl dienen die Maskenparameter, die für jede Maske spezifisch eingestellt werden können, dem Zugang zu weiteren Menüs. Im Bild 2 werden für die Maskenmenüs immer Knotenmasken dargestellt. Die gleiche Funktionalität kann auch mit E/A-Masken erreicht werden, in denen ein Auswahltext für die Maskensprünge eingerichtet wird.

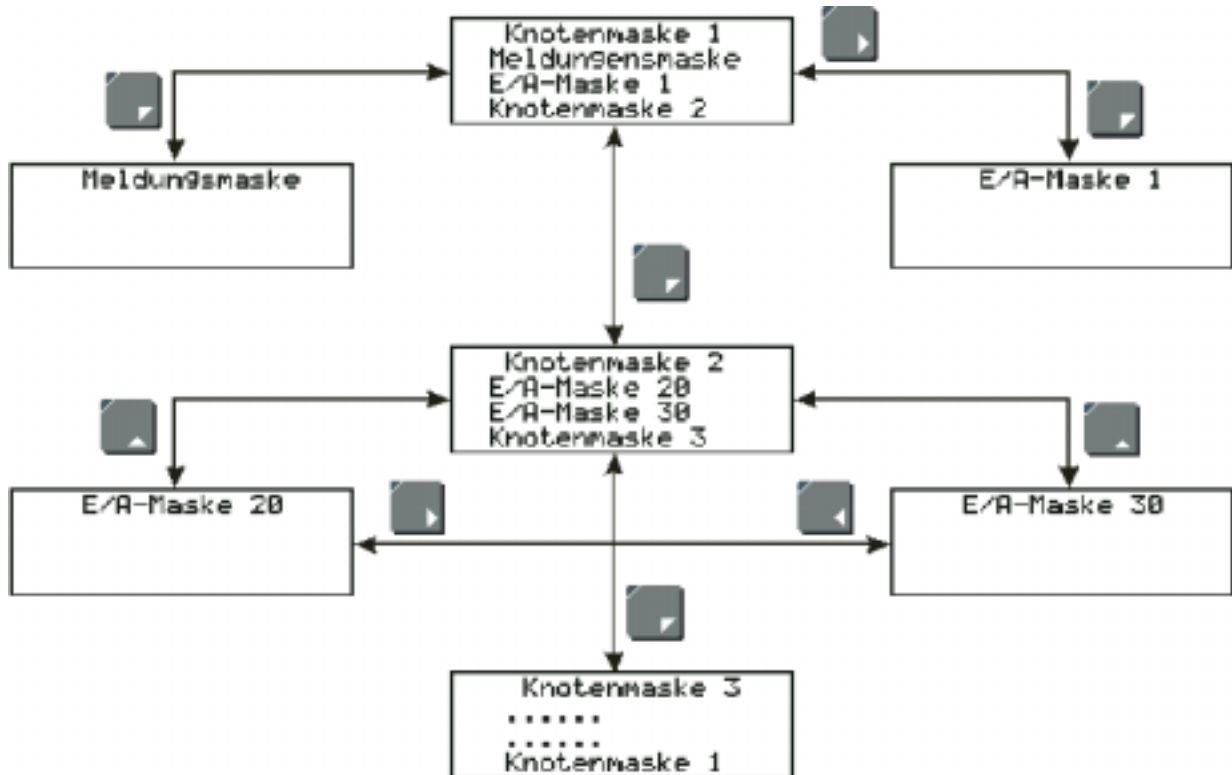


Bild 2: Hierarchische Maskenstruktur

Die Masken unterscheiden sich in den Möglichkeiten, Verbindungen zu anderen Masken herzustellen, in folgender Weise:

- Knotenmasken: Cursortasten Rechts/Links/Home, Funktionstasten und Auswahlpunkte
- E/A-Masken: Cursortasten Rechts/Links/Oben/Unten/Home, Funktionstasten und durch Auswahltexte
- Meldemasken: Cursortaste Home und Funktionstasten
- Zustandsmeldemaske: Cursortaste Home und Funktionstasten

Die Bedienstruktur unterliegt, falls gewünscht, einer kompletten Zugangskontrolle. Zugriffe auf Knoten und darunter liegende Masken lassen sich durch Paßwortschutz verhindern. Auch die Direktanwahl über Funktionstasten wird kontrolliert.

Die Maskenwechsel werden der Steuerung mitgeteilt. Prozeßdaten, die das Terminal im Zusammenhang mit den Masken und Meldungen benötigt, werden aus der Steuerung automatisch abgerufen.

Bei der Tastenauswertung wurde beachtet, daß nachfolgend betätigte Tasten die Ausgabe einer Maske oder die Abfrage von Variablen in der Steuerung überflüssig machen können!

3.4.2 Maskenstruktur bei TSwIn

Bei TSwIn wird ein Netz von E/A-Masken aufgebaut. Das Netz hat keine echte hierarchische Struktur. An Knotenpunkten des Netzes befinden sich E/A-Masken mit einem Auswahlfeld, aus dem die Namen

weiterer Masken ausgewählt werden können. Von jeder E/A-Maske aus kann mit Hilfe von den Steuer- und Funktionstasten jede beliebige andere E/A-Maske erreicht werden. Dadurch entfällt auch die beschränkte Rücksprungfähigkeit aus Meldungsmasken oder Zustandmeldemasken.

3.4.3 Externe Maskenanwahl

Maskenaufrufe von der Steuerung werden in der gleichen Art wie Meldungen getätigt. Dazu muß lediglich 8000H zu der gewünschten Maskennummer vor der Übertragung addiert werden.



Bei der Vergabe von gleichen Nummern für Masken und Meldungen werden bei einer externen Maskenanwahl Maske und Meldung aufgerufen!

Dieser Effekt kann genutzt werden, um in der Maske eine Hilfestellung darzubieten. Ansonsten muß darauf geachtet werden, daß unterschiedliche Nummern für Masken und Meldungen angelegt werden.

Die externe Maske gilt als angewählt, wenn die gewünschte Maskennummer in der Variablen <Abbild der Maskennummer> erscheint. Die Quittung des sequentiellen Datenkanals gibt hier keine ausreichende Gewähr für die Ausgabe der Maske.

Beispiel 1: Maskennummer und Meldungsnummer nicht identisch

Die Masken des Projekts liegen im Nummernbereich 100 bis 200, die erste Meldung hat die Nummer 10.

Maske 118 soll von der Steuerung aufgerufen werden.

In der Steuerung muß addiert werden: $118 + 8000H = 76H + 8000H = 8076H$

Der Wert 8076H muß an die Adresse für den seriellen Meldekanal geschrieben werden.

Angezeigt wird anschließend nur die Maske 118.

Beispiel 2: Maskennummer und Meldungsnummer sind identisch

Die Masken des Projekts liegen im Nummernbereich 1 bis 100, die erste Meldung hat die Nummer 10.

Maske 50 und serielle Meldung 50 sollen von der Steuerung aufgerufen werden.

In der Steuerung muß addiert werden: $50 + 8000H = 32H + 8000H = 8032H$

Der Wert 8032H muß an die Adresse für den seriellen Meldekanal geschrieben werden.

Anschließend wird die Maske 50 angezeigt und die Meldung 50 wird in den Meldungspuffer geschrieben.

3.4.4 Paßwortschutz, Zugriffsrechte

In das TesiMod-Bedienkonzept ist ein Paßwortschutz integriert. Der Paßwortschutz verhindert den unberechtigten Zugang zu Masken und schützt vor unberechtigter Änderung der darin enthaltenen Daten. Die Schutzfunktion ist in allen Terminals verfügbar. Sie wird durch die Einrichtung von Masken mit Sperrebenen und die Vergabe von Paßworten erreicht.

Ohne ausdrückliche Programmierung werden die Sperrebenen aller Masken automatisch auf den niedrigsten Level (=0) gesetzt. Masken mit diesem Level sind auch ohne die Eingabe eines Paßwortes zugänglich.

Jedem Paßwort sind zwei Autorisierungslevel zugeordnet, genannt Editlevel und Viewlevel.

Viewlevel bedeutet, daß nach Eingabe des Paßworts zwar die folgende Maske angeschaut, aber keine Werte darin verändert werden können.

Editlevel bedeutet, daß nach Eingabe des Paßworts die Maske angeschaut und Werte darin verändert werden können.

Bis zu acht verschiedene Paßworte, mit einer Länge von 11 Zeichen, können vorgegeben werden. Bei der Vergabe der verschiedenen Paßworte sollte an eine Strukturierung der Zugangsberechtigungen gedacht werden.

Beispiel:

- Paßwort für den Hersteller der Anlage, Maschine etc.
- Paßwort für den Service vor Ort
- Paßwort für den Maschineneinrichter, Meister, Vorarbeiter
- Paßwort für den Bediener der Anlage

Um die Funktion der Sperrebenen und Level für Edit und View zu erklären soll folgendes Bild dienen:

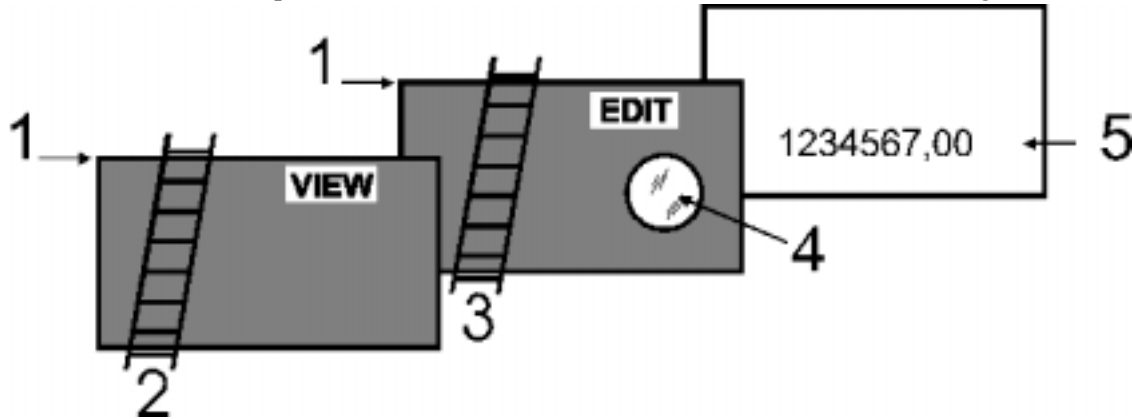


Bild 3: Sperrebenen

Jeder Maske kann eine Sperrebene zugewiesen werden. Hier soll die Sperrebene der Höhe (1) der Mauern gleichgesetzt werden. Einem Paßwort wird ein Viewlevel (Länge der Leiter 2) und ein Editlevel (Länge der Leiter 3) zugewiesen.

Um eine Maske (5) **nur ansehen** zu können, die mit einer Sperrebene versehen ist (Sperrebene größer 0), muß ein Paßwort eingegeben werden. Dem Paßwort muß ein Viewlevel (Leiter 2) zugewiesen sein, der hoch genug ist, um die Sperrebene (Viewmauer) überwinden zu können. Dann kann (durch das Fenster 4) die Maske (5) zwar betrachtet werden, aber Variablenwerte können nicht verändert werden.

Um in der Maske (5) **auch Variablenwerte ändern** zu können, benötigt man ein Paßwort, dem ein genügend hoher Viewlevel (Leiter 2) **und** ein genügend hoher Editlevel (Leiter 3) zugewiesen ist.

Für Paßwörter gilt:

- Eine Sperrebene wird überwunden, wenn der View- bzw. Editlevel höher oder gleich groß wie der Wert der Sperrebene ist.
- Der Editlevel muß gleich oder niedriger als der Viewlevel sein.
- Je höher die Werte für View- und Editlevel, desto höher der Autorisierungsgrad.
- Der gültige Wertebereich für View- und Editlevel beträgt 0 bis 255.
- Die Grundeinstellung beträgt für beide 0.

Wird bei zu niedrigem Editlevel die Datenfreigabetaste betätigt, dann bleibt sie ohne Funktion.

Das Paßwort kann in allen E/A-Masken eingegeben werden. Einen Sonderfall stellt nur die Setup-Maske dar. Zur Eingabe ist die Systemvariable **MskchgPasswd** vorgesehen. Der im Projektiersystem wählbare Paßworteditor erlaubt die verdeckte Eingabe am Terminal. Er gibt für jede eingegebene Stelle ein X-Zeichen aus.

Beispiel:

Eingabe Paßwort: XXXX

Hier wurde eine vierstellige Zahl verdeckt eingegeben. Bei der Eingabe eines falschen Paßwortes werden die Autorisierungslevel automatisch auf 0 gesetzt.

Es wird empfohlen, bei der Projektierung darauf zu achten, daß mindestens ein Paßwort, das Masterpaßwort, mit der höchsten Berechtigung vergeben wird. Das im Projektiersystem an erster Stelle eingegebene Paßwort hat als Masterpaßwort eine besondere Bedeutung. Das Masterpaßwort unterscheidet sich von allen anderen dadurch, daß es im Terminal nicht geändert werden kann. Des weiteren ermöglicht es die Rücksetzung aller geänderten Paßworte auf die im Projektiersystem eingegebenen Grundwerte.

Beispiel:

Maske 5	Sperreebene = 10		
Maske 6	Sperreebene = 20		
Maske 7	Sperreebene = 30		
Paßwort 4712	Editlevel = 15	Viewlevel = 25	

Sobald im Terminal das Paßwort "4712" eingeben ist bestehen folgende Zugänge:

- die Maske 5 wird dargestellt, Werte können geändert werden.
- die Maske 6 wird dargestellt, Werte können **nicht** geändert werden.
- die Maske 7 wird **nicht** dargestellt, Werte können **nicht** geändert werden.

Für die Startup-Maske beträgt die Sperrebene immer 0.

Für die Setup-Maske wurde ein Sonderfall, in Bezug auf Paßwort und externe Datenfreigabe, eingerichtet. Da zum Zeitpunkt der Setup-Maske noch keine Kommunikation stattfindet, entfällt die externe Datenfreigabe. Eine Zugangssicherung ist nur durch ein Paßwort möglich. Falls in der Setup-Maske, als erste editierbare Variable, ein Paßworteditor angegeben wird, bleiben weitere Variablen vor unerlaubtem Zugriff geschützt. Beim Zugang zur Setupmaske bleibt der Viewlevel unberücksichtigt. Die Verzweigung wird immer erlaubt, wenn der gewählte Wert für die Sperrebene der Setup-Maske kleiner oder gleich 254 ist.

Für alle Variablen der Setup-Maske mit Ausnahme des Paßworteditors gilt für den Editlevel die vorgewählte Sperrebene.

Wird für die Setup-Maske die Sperrebene auf 255 gesetzt, so wird generell der Zugang verhindert. Eine Setup-Maske während der Initialisierung des Terminals wird dann nicht mehr angezeigt und ist dann nicht anwählbar. Trotzdem können alle terminalspezifischen Parameter auch in jeder beliebigen E/A-Maske angepaßt werden. Die neuen Parameter werden dann beim erneuten Einschalten des Terminals oder über die Systemvariable **Boot** aktiv.

3.4.4.1 Paßwortschutz wieder aktivieren

Die Zugangsberechtigung zu einer Maske oder Variablen (nur TSwin) wird wieder zurückgenommen, wenn

- das Bedienterminal aus- und eingeschaltet wird
- ein falsches Paßwort eingeben wird
- im <Koordinierungsbyte Schreiben> das Bit 2 gesetzt wird
- die Systemvariable **MskchgResPasswd** aktiviert wird.
- die Option **Paßwort zurücksetzen** in den Maskenparametern der paßwortgeschützten Maske markiert wird.

3.4.4.2 Paßwortverwaltung

Paßwörter werden im Flash-Speicher des Terminals abgelegt. Diese Paßwörter stellen die Grundeinstellung dar, die beim ersten Hochlaufen nach jedem Download übernommen wird. Gleichzeitig werden die Paßwörter im RAM des Bedienterminals abgelegt.

Die im Flash-Speicher abgelegten Paßwörter können durch Schreiben auf die Systemvariable **FlashPasswd** wieder aktiviert werden.

Jedes Paßwort, mit Ausnahme des Masterpaßwortes (erstes Paßwort in der Liste), kann vom Terminal aus geändert werden. Dazu wird zunächst das zu ändernde Paßwort auf die Systemvariable **MskchgPasswd** geschrieben. Anschließend wird das neue Paßwort zweimal auf die Systemvariable **ChangePasswd** geschrieben. Wurde beide Male dasselbe neue Paßwort geschrieben, so ist es ab sofort gültig, ansonsten wird eine entsprechende Systemmeldung ausgegeben und das Paßwort zurückgesetzt. Paßwörter werden als 11 Zeichen lange Strings abgelegt und verglichen. Zur Eingabe der Paßwörter am Terminal wird der alphanumerische Editor verwendet.

Paßwörter sind nicht sprachspezifisch, sondern nur global programmierbar.

3.4.4.3 Paßwortmaske und Paßwortfunktionalität

- Es kann eine Vorschalt-Paßwortmaske (bei TSdos Maske 3) erstellt werden. Sie erscheint immer dann, wenn versucht wird, in eine paßwortgeschützte Maske zu wechseln, ohne vorher ein ausreichend autorisierendes Paßwort eingegeben zu haben. Wenn in ihr ein ausreichend autorisierendes Paßwort eingegeben wird, erfolgt sofort nach betätigen der Datenfreigabetaste ein Wechsel in die zuvor angewählte Maske. Es gibt keine Einschränkungen hinsichtlich des sonstigen Inhalts (Texte, weitere Variablen, Softkeys usw.) der Maske.
- Für jede Maske kann eingestellt werden, ob nach verlassen der Maske der Paßwortschutz wieder aktiviert werden soll.
- Wenn kein gültiges Paßwort eingegeben wurde, muß eine Möglichkeit zum Verlassen der Maske geschaffen sein. Dazu kann beispielsweise die Cursortaste Home programmiert werden.
- Wurde keine Vorschaltmaske erstellt, muß der Bediener ein Paßwort immer in extra dafür vorgesehenen Masken eingeben.
- Der gesamte Paßwortschutz läßt sich, durch Setzen der Systemvariablen **PasswdInactive** auf den Wert 1, deaktivieren. Das Verhalten des Terminals ist dann so, als ob alle Masken mit einem Edit- und Viewlevel von je 0 erstellt worden wären. Die Systemvariable ist batteriegepuffert, das heißt die Deaktivierung wirkt auch nach Wiedereinschalten des Terminals noch.

3.5 Masken

Unter Masken wird im TesiMod-Bediensystem immer der Inhalt einer Displayseite verstanden. Deshalb können Masken je nach Bedienterminal unterschiedliche Größen haben. Masken mit spezifischen Funktionalitäten bilden die Grundbausteine des Bediensystems.

Bei der Programmierung einer Maske wird zuerst die gewünschte Funktionalität festgelegt. Durch die geringe Anzahl unterschiedlicher Maskentypen wird die Strukturierung des Bediensystems sehr erleichtert.

TSdos unterscheidet zwischen folgenden Maskentypen:

- Setup - Maske (Systemmaske)
- Start - Maske (Systemmaske)
- Paßwortmaske (Systemmaske)
- Hauptmaske (Systemmaske)
- E/A-Maske (Anwendermaske)
- Knotenmaske (Anwendermaske)
- Meldungsmaske (Anwendermaske)
- Zustandmeldemaske (Anwendermaske)

3.5.1 Maskenparameter

Maskenparameter sind je nach Maskentyp in unterschiedlicher Anzahl vorhanden. Die Maskenparameter beschreiben die Funktionalität der Steuertasten im Bediensystem. Maskenparameter, die nicht benötigt werden, dürfen unbesetzt bleiben. Falls überhaupt keine Maskenparameter programmiert wurden, kann die jeweilige Maske nur noch über Funktionstasten oder über die externe Maskenanwahl wieder verlassen werden.

Maskenparameter unterliegen außerdem der Zugangskontrolle des Paßwortsystems. Dadurch wird ein unerlaubter Zugang zu einer Maske über die Steuertasten verhindert.

In den Maskenparametern ist neben der Funktionalität der Steuertasten auch die Vorgabe der Sperrebene enthalten. Die Sperrebene gibt an, welchen Wert ein Paßwort (Viewlevel, Editlevel) mindestens haben muß, damit ein Zugang zur Maske und das Verändern von Werten darin erlaubt wird. Der Wert für die Sperrebene ist grundsätzlich auf "0" (das heißt generelle Freigabe) gesetzt.

Mit Hilfe der automatischen Datenfreigabe kann der Bediener direkt eine Eingabe in der Maske ausführen, ohne vorher die Datenfreigabetaste drücken zu müssen.

Um zu erreichen, daß beim erneuten Maskenwechsel zu einer Paßwortgeschützten Maske das Paßwort wieder eingegeben werden muß, kann ein automatisches reaktivieren des Paßwortschutzes zu jeder Maske angewählt werden.

3.5.2 Systemmasken

Systemmasken erleichtern den Einstieg in die Programmierung und führen direkt zu einem lauffähigen System. So wird die Initialisierungsphase fester Bestandteil der Anwenderbeschreibung. Für die Systemmasken von TSdos werden feste Maskennummern vergeben, bei TSwIn kann jede beliebige Maske als Systemmaske ausgewählt werden. Systemmasken basieren auf dem Typ der E/A-Maske mit einigen Einschränkungen. Die Einschränkungen ergeben sich aus der zwangsgeführten Initialisierungsphase und der noch fehlenden Kommunikation zur Steuerung.



Die Systemmasken Setup-Maske und Start-Maske können über die externe Maskenanwahl nicht erreicht werden!

Diese Systemmasken sind bei TSdos fest vergeben:

Maske 1 Setup-Maske

Maske 2 Startup-Maske

Maske 3 Paßwortmaske

Maske 4 Hauptmaske (erste Anwendermaske)

3.5.2.1 Setup-Maske

In der Setup-Maske können nur die terminalspezifischen Parameter eingestellt werden, denn während die Startup- und Setup-Maske angezeigt werden, findet keine Kommunikation mit der angeschlossenen Steuerung statt. Die externe Datenfreigabe ist unwirksam. Ein Datenschutz wird in diesem Fall durch die Vergabe eines Paßwortes erreicht.

Vom Bediensystem aus können die Setup- und Start-Maske über Funktionstasten erreicht werden, solange sie nicht mit einer Sperrebene versehen sind. Jedoch wird beim Verlassen der Masken keine neue Initialisierung durchgeführt.

Beispiel für terminalspezifische Parameter:

- Einstellungen der Druckerschnittstelle
- Grundkontrast- / Grundintensitätseinstellung vom Display
- Einstellung von Datum und Uhrzeit
- aktivieren der Downloadfunktion.

```
BaudrateX2: 9600
Parity:Even D_Bit:8
Übernahme:inaktiv
SN:000 Download:Nein
```

Bild 4: Beispiel für eine Setup-Maske für das IBT

3.5.2.1.1 Paßwortschutz Setup-Maske

Die Setup-Maske wird beim Paßwortschutz besonders behandelt.

Wenn in der Setup-Maske als erste editierbare Variable die Systemvariable **MskchgPasswd** eingerichtet ist, wird die Eingabe des Paßwortes, unabhängig von der Sperrebene (Ausnahme 255), ermöglicht. Dadurch kann auch die Setup-Maske paßwortgeschützt werden.

Für die Setup-Maske wirkt die Sperrebene nur auf den Editlevel, so daß ihr Inhalt für den Bediener immer sichtbar ist.

3.5.2.1.2 Funktion ohne Setup-Maske

Wenn auf die Setup-Maske verzichtet werden kann, verhindert durch setzen der Sperrebene auf den Wert "255" den Zugang. Die Setup-Maske ist dann nicht mehr über die Startup-Maske (mit der Datenfreigabetaste) erreichbar.

3.5.2.2 Startup-Maske

Die Startup-Maske wird nach dem Einschalten des Terminals für etwa 5 s angezeigt. Die Zeit ist fest eingestellt, der Ablauf kann nicht verändert werden.

Der Anwender hat für die Startup-Maske im Projektiersystem nur textliche Gestaltungsfreiheit. Das bedeutet beliebige Kombination von Zeichen, Zeichengröße und Textattributen.

In dieser Maske können nur Systemvariable ausgegeben werden. Eine Eingabe von Variablen ist aufgrund des zeitlichen Ablaufs nicht möglich. Während die Startup-Maske angezeigt wird, kann durch Betätigen der Datenfreigabetaste, in die Setup-Maske gewechselt werden. Ein Wechsel in die Setup-Maske ist nicht möglich, wenn die Sperrebene der Setup-Maske auf 255 eingestellt ist.

In der Startup-Maske können Sie beispielsweise folgende Angaben anzeigen:

- Serviceadresse
- Maschinentyp
- Version des Programms

```
Diese Maske bootet
in 5 Sekunden das
Projekt.
Mit Enter ins Setup!
```

Bild 5: Beispiel einer Startup-Maske für das IBT

3.5.2.3 Paßwortmaske

Die Paßwortmaske ist vom Basistyp der E/A-Maske. Die Paßwort-Vorschaltmaske hat bei TSdos immer die Maskennummer 3. Die Funktionalität ergibt sich aus der Paßwortbeschreibung.

```
Paßworteingabe
Paßwort:xxxx
```

Bild 6: Beispiel einer Paßwortmaske für das IBT

3.5.2.4 Knotenmaske, E/A-Maske mit Auswahltext

Die Knotenmaske als Basiselement der Bedienstruktur ist nur noch in TSdos verfügbar. Da mit Hilfe von einem Auswahltext (verknüpft mit Textliste und Systemvariable **NewMask**) und automatischer Datenfreigabe die Funktionalität von einer Knotenmaske erreicht werden kann, gibt es den Typ Knotenmaske in TSwin nicht mehr.

Die Knotenmaske besitzt einen Kopf- und einen Auswahlteil.

Im Kopfteil werden die Überschriften und Zusatzinformationen zum Menü angegeben. Die Anzahl der Displayzeilen für die Überschrift kann im Projektiersystem angegeben werden.

Im Kopfteil kann nur ein einheitliches Textattribut für alle Textelemente angegeben werden.

Im Anwahlteil sind die wählbaren Untermenüs beschrieben und können durch Cursor Oben und Cursor Unten angewählt werden. Die Anwahl ist mit der Entertaste zu bestätigen. Der Zugang zu Untermenüs kann durch Paßwortschutz verhindert werden.

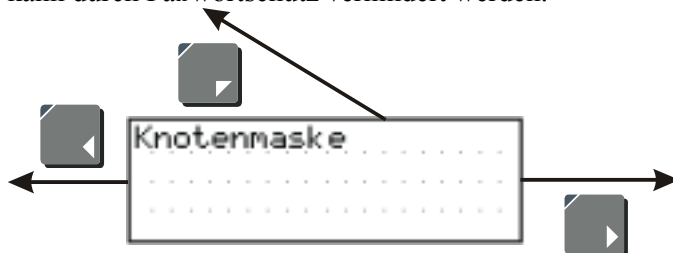


Bild 7: Mögliche Steuertasten der Knotenmaske

Eine Knotenmaske kann außerdem über die Cursortasten Rechts / Links / Home und über programmierte Funktions- und Softkeytasten verlassen werden. Die Cursortasten werden zur direkten

Anwahl von Menüs auf der gleichen Ebene verwendet. Die Cursortaste Home ist vorgesehen, um übergeordnete Menüs zu erreichen.

In den Knotenmasken sind keine Ein-/Ausgaben von Variablen möglich. Sie werden nur zur Verzweigung in weitere Knoten- oder E/A-Masken verwendet. Ein Bediensystem aus Knotenmasken ist deshalb nur in Verbindung mit dem E/A-Maskentyp sinnvoll.

Tastenfunktionen in der Knotenmaske

Cursor Oben	Bewegt den Cursor um einen Auswahlpunkt nach oben.
Cursor Unten	Bewegt den Cursor um einen Auswahlpunkt nach unten.
Cursor Links	Zum Aufruf benachbarter Masken frei belegbar.
Cursor Rechts	Zum Aufruf benachbarter Masken frei belegbar.
Cursor Home	Zum Aufruf benachbarter Masken frei belegbar. Sie verläßt die Knotenmaske in das übergeordnete Menü.
Datenfreigabe	ohne Funktion
Dateneingabe	Startet den Sprung zu der angewählten Maske.
? (Hilfe)	Gibt den zur Maske erstellten Hilfetext aus, solange die Taste betätigt bleibt.

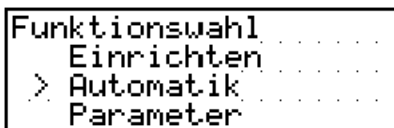


Bild 8: Beispiel einer Knotenmaske

3.5.2.5 E/A-Maske

Der Basistyp E/A-Maske ist in seiner Funktionalität bereits so vielseitig, daß damit komplette Systeme aufgebaut werden können.

E/A-Masken eröffnen folgende Möglichkeiten:

- Auswahlmenüs
- Datenformate definieren
- Werte skalieren
- Werte einmalig oder zyklisch ausgegeben
- Texte anzeigen
- Meldungen ausgegeben
- durch Steuertasten bis zu fünf benachbarte Masken direkt anwählen
- größere Maskeninhalte als mehrere Displayseiten darstellen
- Werte in Tabellen darstellen

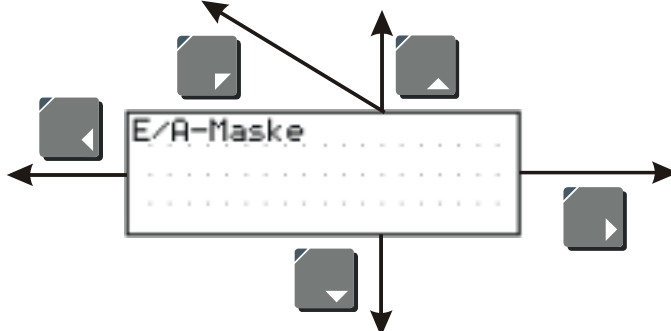


Bild 9: Mögliche Steuertasten der E/A-Maske

Tastenfunktionen in der E/A-Maske

Cursor Oben	Zum Aufruf benachbarter Masken frei belegbar.
Cursor Unten	Zum Aufruf benachbarter Masken frei belegbar.
Cursor Links	Zum Aufruf benachbarter Masken frei belegbar.
Cursor Rechts	Zum Aufruf benachbarter Masken frei belegbar.
Cursor Home	Zum Aufruf benachbarter Masken frei belegbar. Sie verläßt die E/A-Maske in das übergeordnete Menü
Datenfreigabe	Wechselt in den Editor und verläßt den Editor wieder.
Dateneingabe	Besitzt nur eine Funktion in Verbindung mit dem Editor.
?, Hilfe	Gibt den zur Maske, bzw. Variable erstellten Hilfetext aus, solange die Taste betätigt bleibt.

```
Parameter
Position: 1000 Ink.
Geschw.: 750 rpm
Beschl.: 50000 rpm/s
```

Bild 10: Beispiel einer E/A-Maske für das IBT

3.5.2.6 Meldungsmaske, Maske mit Meldungsfeld für serielle Meldungen

Diese Maskenart ist nur noch in TSdos verfügbar, in TSwIn wird diese Maskenart durch die E/A-Maske mit Meldungsfeld für serielle Meldungen ersetzt.

Die Meldungsmaske besteht aus einem Kopfteil und einem Meldungsfeld.

Um eine serielle Meldung am Bedienterminal anzeigen zu lassen, muß deren Nummer in den Pollbereich geschrieben werden. Innerhalb des Pollbereichs ist ein Datenwort (zwei Byte) reserviert, über das serielle Meldungsnummern übertragen werden (serieller Meldekanal).

Die eingehenden Meldungen können direkt vom Bedienterminal nach vorgegebenen Kriterien sortiert werden, bevor sie angezeigt werden.

Meldungen können wahlweise mit

- Meldungsnummer
- Meldungsdatum
- Meldungszeitpunkt
- einer beliebigen Kombination

am Bedienterminal ausgegeben werden.

Die Anzahl der Meldungsnummern, die im Bedienterminal gespeichert werden können, hängt von den Einstellungen in den Systemparametern und der Größe des RAM-Maskenspeichers im Bedienterminal ab.

Die seriellen Meldungen werden solange in der Meldungsmaske dargestellt, bis sie vom Bediener aus dem Meldungsspeicher im Bedienterminal gelöscht werden.

Kopfteil der Maske:

Im Kopfteil werden die Überschrift und eventuell Zusatzinformationen (zum Beispiel Hinweis zur Darstellung des restlichen Meldungsteils) zu den Meldungen angegeben. Die Anzahl der Displayzeilen für die Überschrift kann im Projektiersystem angegeben werden.

Im Kopfteil kann nur ein einheitliches Textattribut für alle Textelemente angegeben werden. In der möglichen Hilfemaske können Hinweise zur Bedienung der Meldungsmaske gegeben werden.

Meldungsfeld der Maske:

Alle eingehenden Meldungen werden im Meldungsspeicher abgelegt und in dieser Maske nach bestimmten Sortierkriterien angezeigt.

Wenn bei TSdos-Projekten eine Meldung länger ist, als eine Zeile vom Display des Bedienterminals, wird sie gekürzt dargestellt. Es gehen aber keine Informationen der Meldung verloren!

Um die gesamte Meldung ansehen zu können, ist es möglich, diese zu markieren und per Tastendruck auf Dateneingabe (Enter) in einer extra Maske darzustellen (Zoomfunktion).

Bei TSwIn-Projekten wird immer der komplette Meldungstext angezeigt, kann aber auch per Zoomfunktion in einer extra Maske dargestellt werden.

Das Zoomen der Meldung gehört zum Funktionsumfang der Meldungsmaske und muß nicht extra projiziert werden! (Siehe Kapitel 3.9.2.2.3 *Zoomen von Meldungen*)

Die Art der Meldungsdarstellung kann durch Festlegungen im Projektiersystem oder Online in einer sogenannten Konfigurationsmaske durch Systemvariablen eingerichtet werden.

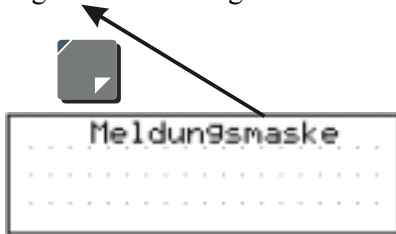


Bild 11: Mögliche Steuertasten der Meldungsmaske

Tastenfunktionen in der Meldungsmaske

Cursor Home	Zum Aufruf beliebiger Masken frei belegbar. Sie verläßt die Meldungsmaske in das übergeordnete Menü.
Cursor Oben	Bewegt den Zeiger (>) nach Oben
Cursor Unten	Bewegt den Zeiger (>) nach Unten
Cursor Rechts	Bewegt den Zeiger (>) um eine Meldungsfeldlänge nach Unten
Cursor Links	Bewegt den Zeiger (>) um eine Meldungsfeldlänge nach Oben
?, Hilfe	Gibt den zur Maske erstellten Hilfetext aus, solange die Taste betätigt wird.
Dateneingabe	Zoomen der Meldung. Alle Zeichen einer Meldung werden dargestellt.
Datenfreigabe	Wechselt in den Meldungseditor und verläßt den Meldungseditor wieder. Wenn der Meldungseditor aktiv ist, können Meldungen markieren werden. Die markierten Meldungen können dann mit der Taste Löschen gelöscht werden oder mit Hilfe der Systemvariablen BlockPrint ausgedruckt werden. Der Ausdruck erfolgt mit den gleichen Einstellungen, wie sie auch für die Anzeige der Meldungen gewählt wurden.

3.5.2.7 Zustandsmeldungsmaske

Die Zustandsmeldungsmaske ist funktionell identisch mit der Meldungsmaske. Es lassen sich die gleichen Einflußnahmen durchführen. Die Inhalte beziehen sich jedoch in erster Linie auf die Meldetexte des parallelen Meldesystems. Diese Maskenart ist nur noch in TSdos verfügbar, in TSwIn wird diese Maskenart durch die E/A-Maske mit Meldungsfeld für parallele Meldungen ersetzt.

Zustandsmeldungen werden über einen separaten Daten- / Adreßbereich an das Bedienterminal gemeldet. Dieser Bereich kann frei gewählt werden. Jedes Bit dieses Datenbereichs repräsentiert eine Zustandsmeldung. Ist das Bit gesetzt, so wird die Meldung am Bedienterminal angezeigt; wird das Bit wieder zurückgesetzt, so erlischt auch die Meldung am Bedienterminal.

Zustandsmeldungen können wahlweise mit

- Meldungsnummer
 - Meldungsdatum
 - Meldungszeitpunkt
 - einer beliebigen Kombination
- am Bedienterminal ausgegeben werden.

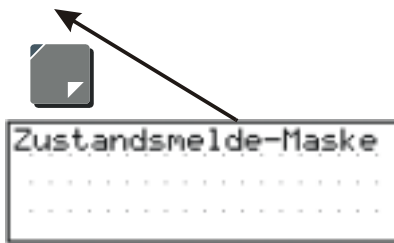


Bild 12: Mögliche Steuertasten der Zustandsmeldungsmaske

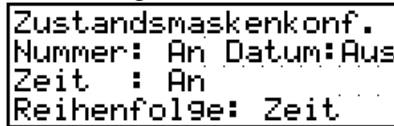


Bild 13: Beispiel einer Konfigurationsmaske

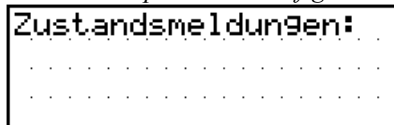


Bild 14: Beispiel einer Zustandsmeldemaske

Tastenfunktionen in der Zustandsmeldungsmaske

Taste	
Cursor Home	Zum Aufruf beliebiger Masken frei belegbar. Sie verläßt die Meldungsmaske in das übergeordnete Menü.
Cursor Oben	Bewegt den Zeiger (>) nach Oben
Cursor Unten	Bewegt den Zeiger (>) nach Unten
Cursor Rechts	Bewegt den Zeiger (>) um eine Meldungsfeldlänge nach Unten
Cursor Links	Bewegt den Zeiger (>) um eine Meldungsfeldlänge nach Oben
?, Hilfe	Gibt den zur Maske erstellten Hilfetext aus, solange die Taste betätigt wird.
Dateneingabe	Zoomen der Meldung. Alle Zeichen einer Meldung werden dargestellt.
Datenfreigabe	Wechselt in den Meldungseditor und verläßt den Meldungseditor wieder. Wenn der Meldungseditor aktiv ist, können Meldungen markieren werden. Die markierten Meldungen können dann mit der Taste Löschen gelöscht werden oder mit Hilfe der Systemvariablen BlockPrint ausgedruckt werden. Der Ausdruck erfolgt mit den gleichen Einstellungen, wie sie auch für die Anzeige der Meldungen gewählt wurden.

3.6 Variablen

Die Anzahl der zulässigen Variablentypen wird durch die angeschlossene Steuerung bestimmt. Von den Bedienterminals werden allgemein verwendbare Standardtypen unterstützt. Mit dem Datentyp liegt auch der mögliche Wertebereich und die Zahl der signifikanten Stellen fest.

<u>Variablentyp</u>	<u>Größe</u>	<u>Wertebereich</u>
Bit	1 Bit	[0, 1]
Byte	1 Byte	[-128 bis +127]
Byte	1 Byte	[0 bis 255]
Wort	2 Byte	[-32768 bis +32767]
Wort	2 Byte	[0 bis 65535]
LWort	4 Byte	[-2147483648 bis +2147483647]
LWort	4 Byte	[0 bis 4294967295]
LWort	4 Byte	[$\pm 1.2 \cdot 10^{-38}$ bis $\pm 3.4 \cdot 10^{+38}$]
ASCII	42 Byte	[0 bis 255]

3.6.1 Ausgabevariablen

Bei Ausgabevariablen handelt es sich um numerische oder alphanumerische Speicherinhalte der angeschlossenen Steuerung. Die Variablen werden aus der Steuerung bei Bedarf angefordert und an der programmierten Stelle unter Berücksichtigung der Darstellungsart, der Formatierung und Skalierung am Display ausgegeben. Allgemein wird zwischen einmaliger und zyklischer Ausgabe unterschieden. Die zusätzlich wählbaren Zeichenattribute sind vom Displaytyp des jeweiligen Bedienterminals abhängig.

Einmalige und zyklische Ausgabe:

Reine Ausgabevariablen werden beim Aufbau der Maske einmalig aus der Steuerung übertragen und in der Maske dargestellt. Die einmalige Ausgabe dient der Entlastung der Kommunikation und kann für alle Variablen wie Sollwerte, Konstanten und Parameter verwendet werden, die sich selten oder nie ändern. Alle Ausgabevariablen können skaliert und formatiert dargestellt werden.

Zyklische Ausgabevariablen werden zur Darstellung von Istwerten, Werte die sich während der Ausgabe einer Maske ständig ändern, verwendet. Die Zykluszeit wird mit der Pollzeit vom Anwender vorgegeben. Damit ist gleichzeitig bekannt, wie oft eine Auffrischung der Istwerte in der Anzeige durchgeführt wird.

Damit die Übertragung zur Steuerung entlastet wird, sollten die Datentypen gleich und die Adressbereiche einer Maske möglichst fortlaufend sein.

Zyklische Ausgabevariablen können ebenfalls skaliert und formatiert werden. Dabei ist besonders bei der Darstellung von Fließkommazahlen darauf zu achten, daß die mögliche Skalierung eine entsprechende Rechenzeit erfordert und die Ausgabe damit nicht in "Echtzeit" erscheint. Hier sollten auf jeden Fall Zykluszeiten > 500 ms gewählt werden.

Allgemein ist zu beachten, daß die Reaktionszeit auf neue Werte von der SPS länger wird, je mehr zyklische Daten übertragen und damit die Pollzeiten verlängert werden müssen

Formatierte Ausgabe:

Die Formatierung kann eine numerische Variable durch das Anzeigeformat an den Ausgabebereich anpassen. Die Anzahl der Ausgabestellen, die Nachkommastellen und die Wahl mit/ohne Vorzeichen kennzeichnen die Formatierung. Durch die Anzahl der Nachkommastellen wird dem Bediener eine Division vorgetäuscht. Eine reale Division findet nicht statt. Die Variable muß in der entsprechend höheren Auflösung in der Steuerung vorliegen.

Beispiel:

Istwert einer Länge in der Steuerung:

Wortvariable Wertebereich 0 bis 65535
 Auflösung 1/100 mm

Istwertanzeige im Bedienterminal:

Anzeigebereich 0,00 bis 655,35 mm

Die Ausgabe erfolgt in Millimeter, es ist keine Umrechnung in der Steuerung erforderlich. In der Variablenbeschreibung wurde als Format gewählt:

SPS-Adresse: Datentyp Wort
TSdos: Positive Ganzzahl (ohne Vorzeichen)
TSwin: Dezimalzahl (ohne Vorzeichen)
Länge: 6 Stellen
Nachkomma: 2 Stellen

Durch die Anwahl „mit Vorzeichen“ würde sich die Ausgabe so verändern:

Anzeigebereich -327,68 bis 327,67 mm

Durch das Vorzeichen hat sich die Ausgabelänge auf 7 Stellen erhöht, dies ist in der Variablenbeschreibung zu berücksichtigen.

<u>Format</u>	<u>Datentyp</u>	<u>formatierbar</u>	<u>skalierbar</u>
Binär	Bit, Byte, Wort, LWort	x	-
Hexadezimal	Byte, Wort, LWort	-	x
Dezimal	Bit, Byte, Wort, LWort	x	x
BCD	Byte, Wort, LWort	-	x
Gleitkomma	LWort	x	x
Auswahltext (kodierter Text)	Bit, Byte, Wort	x	-
Text	erw. ASCII-Zeichensatz	-	-

Die Formatierung entspricht einer Skalierung, ohne daß eine Rechenoperation durchgeführt werden muß.

Skalierte Ausgabe:

Durch die Skalierung der Variablen kann ihr Wertebereich dem Bedienersystem angepaßt werden. Die Skalierung ist für die Dateneingabe und Datenausgabe wirksam. Die Operanden können in der Programmiersoftware eingegeben werden.

Skalierung für Ganzzahlen:

Operand Wertebereich

Faktor	-32768 bis +32767 (ohne den Wert 0!)
Divisor	1 bis +32767
Summand	-32768 bis +32767

Die Operanden Faktor und Divisor müssen >0 sein!

Skalierung für Gleitkomma-Zahlen:

Operand Wertebereich

Faktor	+/-999999999,99999999 (ohne den Wert 0!)
Divisor	+/-999999999,99999999 (ohne den Wert 0!)
Summand	+/-999999999,99999999

Die Operanden werden im IEEE-Format gespeichert.
Zusätzlich behält die Variable für die Formatierung den vollen Wertebereich.

Formel für die Skalierung der Ausgabe:

Die Skalierung erfolgt immer im Bedienterminal. Die in der Steuerung erfaßten Meßwerte sollen durch die Skalierung dem Bedienersystem angepaßt werden. Für die Darstellung der Ausgabe ist nur die *Formel 1* gültig. Der Summand, was einer Offset-Verschiebung entspricht, kann im Gegensatz zu Faktor und Divisor auf den Wert = 0 gesetzt werden.

Eine Variable oder Konstante wird für die Darstellung auf dem Display wie folgt umgerechnet:

$$\boxed{\text{Terminal-Ausgabewert}} = \frac{\boxed{\text{SPS-Variablenwert}} \times \text{Faktor}}{\text{Divisor}} + \text{Summand}$$

Bild 15: Formel 1

Darstellungen der Ausgabe:

Man ist gewohnt, sich bestimmte Variablentypen in ihrer eigenen Darstellung anzuschauen. Die typenspezifische Darstellung vereinfacht das Interpretieren des Inhalts. Aus diesem Grund werden die unterschiedlichsten Darstellungsformen zur Verfügung gestellt.

Beispiele:

- Eingangszustände an Eingangskarte Binärdarstellung
- Füllstand eines Behälters Balkendiagramm
- Temperatur Kurvendarstellung
- Ventilzustände Bilder des Ventils

Beispiel für die Verwendung eines kodierten Textes für die Ausgabe eines Endlagenzustands:

<u>binär</u>	<u>hex</u>	<u>dezimal</u>	<u>Auswahltext (kodierter Text)</u>
0	00	0	Endlage frei
1	20	32	Endlage erreicht

Beispiel für die Verwendung eines kodierten Textes für die Eingabe der Parität einer Schnittstelle:

dezimal	Auswahltext (kodierter Text)
0	Keine
1	Ungerade
2	Gerade

Darstellung mit führenden Nullen:

Die Ausgabe mit führenden Nullen kann mit allen Ganzzahlen kombiniert werden. Der Einsatz ist speziell für die Ausgabe von Datum und Uhrzeit, sowie bei der hexadezimalen oder binären Darstellung erforderlich. Die Möglichkeit der "Führenden Nullen" läßt sich in der Variablenbeschreibung anwählen.

	<u>Darstellung ohne</u>	<u>Darstellung mit</u>
Binärzahl	10 0101	0010 0101
Uhrzeit	8: 2:33	08:02:33
Datum	5. 3.1997	05.03.1997

3.6.1.1 Darstellungstyp Dezimalzahl

Die dezimale Darstellung von Zahlen wird am häufigsten verwendet.

TSwin unterscheidet bei der Darstellung von Dezimalzahlen zwischen Standard, Timer, Zähler und BCD-Format.

In den Bereich der dezimalen Darstellung fallen die Ganzzahlen und Festkommazahlen.

3.6.1.1.1 Variablentyp Standard

Die Wertigkeit der angezeigten Stellen steigt von rechts nach links. Sie erfolgt wahlweise mit Vornullen oder Dezimalpunkt. Die Darstellung bezieht sich auf die Datentypen Bit, Byte, Wort und LWort. Die maximale Länge ist vom Datentyp abhängig. Zwischen den Zeichen befinden sich keine Leerzeichen. Die Variable liegt in der Steuerung im Binärformat oder speziellen Timer- / Zählerformaten vor.

10^3	10^2	10^1	10^0	10^{-1}	10^{-2}	Wertigkeit
0	1	2	3	4	5	Anzeige (123,45 _D)

3.6.1.1.2 Variablentyp Timer

Der Variablentyp Timer hat nur in Kombination mit Simatic S5 Steuerungen eine besondere Bedeutung.

Die Art der Formatierung des Variablentyps Timer hängt davon ab, aus welchem Speicherbereich der SPS der Wert ausgelesen wurde. Wurde er direkt aus einem Zeitwort gelesen, so wird der darin enthaltene 10 Bit große binäre Zeitwert mit der 2 Bit großen Zeitbasis so verrechnet, daß man den Zeitwert zur Basis 10 ms erhält. Wurde er aus einem anderen Speicherbereich, also DW, MW, EW oder AW, gelesen, so wird davon ausgegangen, daß er BCD-kodiert vorliegt (3 Stellen BCD-Code und 2 Bit Zeitbasis). Er wird dann ebenfalls in einen Zeitwert zur Basis 10 ms umgerechnet.

Der so gebildete Zeitwert zur Basis 10 ms kann nun genauso weiterformatiert werden, wie eine Festkommazahl. Es können also Nachkommastellen verwendet und eine Skalierung vorgenommen werden.

Beispiel:

Am Bedienterminal wird ein Sollwert eingegeben: Adresse MW100

Am Bedienterminal wird der aktuelle Istwert angezeigt: Adresse MW200

Darstellung Eingabevariable: Dezimalzahl / Timer / 7 Stellen / 2 Nachkommastellen / Faktor 1 / Divisor 1 / Summand 0

Darstellung Ausgabevariable: Dezimalzahl / Timer / 6 Stellen / 1 Nachkommastelle / Faktor 1 / Divisor 10 / Summand 0

Durch die Befehlssequenz

```
L MW 100
SI T 1
```

wurde in das Merkerwort 100 der BCD-kodierte Zeitwert geladen.

Mit der Befehlssequenz

```
LC T 1          (Aktuellen Timerwert als BCD-Zahl laden)
T MW 200
```

wird in das Merkerwort 200 jetzt der aktuelle Timerwert geladen. Das Bedienterminal liest diesen Wert als BCD-codierten Timerwert und interpretiert ihn, bevor eine Ausgabe stattfindet.

Zunächst wird der Zeitwert zur Basis 10 ms gebildet und anschließend skaliert.

In diesem Beispiel werden die Ausgabewerte mit einer Nachkommastelle angezeigt. Es erscheinen dann folgende Darstellungen am Bedienterminal:

Eingabewert	Ausgabewert	Auflösung	KT-Werte (S5)
0000,01 bis 0000,09	0000,0 bis 0000,0	0,01 s	001.0 bis 009.0
0000,10 bis 0000,99	0000,1 bis 0000,9	0,01 s	010.0 bis 099.0
0001,00 bis 0009,99	0001,0 bis 0009,9	0,01 s	100.0 bis 999.0
0010,00 bis 0099,90	0010,0 bis 0099,9	0,1 s	100.1 bis 999.1
0100,00 bis 0999,00	0100,0 bis 0999,0	1 s	100.2 bis 999.2
1000,00 bis 9990,00	1000,0 bis 9990,0	10 s	100.3 bis 999.3

Durch Verändern der Nachkommastellen und der Skalierung kann die Auflösung und damit die Eingabe- und Ablesegenauigkeit geändert werden.

3.6.1.1.3 Variablentyp Zähler

Der Variablentyp Zähler kann nur in Kombination mit Steuerungen verwendet werden, die diesen Typ auch unterstützen. Als Beispiel wird der Variablentyp Zähler in Kombination mit der Siemens S5 erklärt.

C Hilfsmerker für das Abfragen
 B Flankenmerker für das Setzen, Freigeben, Vorwärts- und Rückwärtszählen
 A Zählwert dualkodiert von 0 bis 999

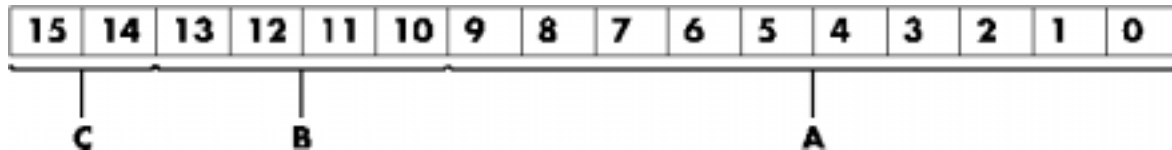


Bild 16: Aufbau des Zählerworts bei Siemens SIMATIC S5-115U

Die Art der Formatierung des Variablentyps Zähler hängt davon ab, aus welchem Speicherbereich der SPS der Wert ausgelesen wurde. Wurde er direkt aus einem Zählwort gelesen, so wird der darin enthaltene 10 Bit große binäre Zählwert direkt übernommen.

Wurde er jedoch aus einem anderen Speicherbereich, also DW, MW, EW oder AW, gelesen, so wird davon ausgegangen, daß er BCD-kodiert vorliegt (3 Stellen BCD-Code). Er wird dann zunächst in einen binären Zählwert umgerechnet.

Der so gebildete binäre Zählwert kann nun genauso weiterformatiert werden, wie eine Ganzzahl. Es kann also eine Skalierung vorgenommen werden.

3.6.1.1.4 Variablentyp BCD-Zahl

Die BCD Darstellung von Zahlen wird teilweise im SPS-Datenbereich verwendet. Die Ausgabe ist in speziellen Fällen erforderlich. Die Wertigkeit der angezeigten Stellen steigt von rechts nach links. Die Darstellung erfolgt mit 0 bis 9. Die Darstellung erfolgt wahlweise mit Vornullern. Sie bezieht sich auf die Datentypen Byte, Wort und LWord. Die maximale Länge beträgt 8 Stellen. Zwischen den Zeichen befinden sich keine Leerzeichen. Die Ablage des Variablenwertes in der Steuerung erfolgt im BCD Format. Der Zahlenbereich umfaßt bei einem Byte 00 bis 99.

10^4	10^3	10^2	10^1	10^0	Wertigkeit
0	1	2	3	4	Anzeige (1234 ₁₀)

3.6.1.2 Darstellungstyp Alphanumerisch

Bei der alphanumerischen Darstellung werden ASCII-Strings im Byteformat aus der Steuerung gelesen und im Display dargestellt. Die Anzahl der darstellbaren Zeichen richtet sich nach den jeweiligen Möglichkeiten des Terminaltyps. Die Länge einer Variablen darf maximal eine Displayzeile lang sein, längere Texte werden abgeschnitten. Die Adresse in der Variablenliste gibt den Anfang des Zeichenstrings an. Ein Längenbyte ist nicht enthalten und nicht erforderlich! Durch die alphanumerische Darstellung besteht eine weitere Möglichkeit, Maskentexte zur Laufzeit zu ändern.

3.6.1.3 Darstellungstyp Auswahltext (Kodierter Text)

Beim Auswahltext (kodierten Text - TSdos) wird einem numerischen Wert ein Textstring zugeordnet. Damit können zwei unterschiedliche Aufgaben gelöst werden:

1. Zustände mit Texten darstellen oder auswählen
2. Maskenwechsel (TSwin)

Der Auswahltext wird in TSwIn genauso benutzt, wie das Auswahlfeld in TSdos.

Der kodierte Text beschränkt sich in TSdos auf eine einzeilige Ausgabe.

Ein Beispiel für die Verwendung von einzeiligem Auswahltext ist die Auswahlmöglichkeit von der Parität einer seriellen Schnittstelle.

Eine Textliste enthält dann die drei folgenden Einträge:

<u>Wert</u>	<u>Text</u>
0	Keine
1	Ungerade
2	Gerade

In der Maske wird eine Variable (**ComParityA** / **ComParityB**) angelegt.

Mittels Standard- oder Mixmodeeditor kann am Bedienterminal nur zwischen den drei gegebenen Möglichkeiten gewählt werden; eine falsche Eingabe wird dann von vornherein verhindert.

Mehrzeilige Auswahltexte werden vor allem zur Menüsteuerung (Ersatz für Knotenmaske) eingesetzt. Im mehrzeiligen Auswahltext wird im Gegensatz zum einzeiligen Auswahltext die Textliste insgesamt oder ausschnittsweise dargestellt. Ist die Höhe vom Auswahltext geringer als die Anzahl der darzustellenden Textelemente, kann mit den Cursortasten durch die Textelemente gerollt werden. Nach Erreichen des letzten Textlisteneintrags beginnt die Auswahl wieder beim ersten Eintrag.

Der angewählte Text ist markiert, wenn die gesamte Auswahltextzeile invers dargestellt wird.

Hier wird ebenfalls eine Textliste benötigt, die den Maskennummern die Maskennamen zuweist:

<u>Wert</u>	<u>Text</u>
10	Maschinenparameter
20	Serielle Meldungsmaske
30	Zustandsmeldungen
40	Meldungskonfiguration
50	Schnittstellenparameter

In der Hauptmaske (Maske 4) wird eine Variable (**NewMask**) vom Typ Auswahltext angelegt, die eine Höhe von maximal 5 Zeilen und eine Länge von 25 Zeichen hat. Die Werte in der Textliste müssen mit den Maskennummern übereinstimmen.

Am Bedienterminal kann dann einer der Einträge (und damit ein Wechsel zur gewünschten Maske) aus dem Auswahlfeld ausgewählt werden.

3.6.1.4 Darstellungstyp Auswahlbild (Kodiertes Bild)

Beim Auswahlbild (kodiertes Bild - TSdos) wird in einer Bildliste einem numerischen Wert ein Bild (Pixelgrafik) zugeordnet. Die Bildliste wird mit der Variablen verknüpft, deren Werte als Bilder

dargestellt werden sollen. Dies ermöglicht eine sprachunabhängige Visualisierung von Betriebszuständen, Eingängen, Ausgängen usw.

Die numerischen Werte zu den Bildern können frei vergeben werden. Sie müssen weder lückenlos noch fortlaufend sortiert sein. Außerdem gehört zu jeder Bildliste ein Defaultbild. Nimmt die Variable einen Wert an, der in der Bildliste nicht vorkommt, wird das Defaultbild angezeigt. Die verwendeten Bilder werden auf das Ausgabeformat beschnitten. Die Darstellung ist auf Bit, Byte oder Wort beschränkt.

Neben der reinen Darstellung ist das Editieren der Auswahlbildvariablen mittels +/- -Taste wie beim Auswahltexteditor möglich. Die eingegebene Änderung wird der Steuerung direkt mitgeteilt.

Ein Beispiel zeigt, wie die Grafiken numerischen Werten zugeordnet sind.

<u>Wert</u>	<u>Bildname</u>
120	Symbol1
34	Symbol2
7	Symbol3
1201	Symbol4

Die Namen der Bilder repräsentieren in der Bildliste die Grafiken. In Wirklichkeit sieht die Liste wie folgt aus:

```
120    
34     
7      
1201 
```

Die Bildliste ist mit vier Einträgen definiert. Es wird jeweils die Grafik dargestellt, die von der Steuerung angewählt wurde.

Alle Bilder einer Bildliste sollten dieselbe Ausgabegröße haben, damit sie sich vollständig überschreiben. In einer E/A-Maske können mehrere Auswahlbildvariablen vorhanden sein. Sie können von der Funktion einmalige Ausgabevariable, zyklische Ausgabevariable und Eingabevariable sein. Bei der zyklischen Ausgabe von Auswahlbildern (“Animation”) ist aufgrund der Hardwarevoraussetzungen von “langsamen” Änderungen auszugehen. Je mehr Bilder ausgegeben werden, desto geringer wird die Ausgabeleistung. Die Auswahlbildvariable sollte deshalb vorrangig für Schaltzustände oder quasistatische Vorgänge zur Anwendung kommen.

3.6.1.5 Darstellungstyp Gleitkommazahl

Für den Darstellungstyp Gleitkommazahl gelten prinzipiell die gleichen Regeln, wie für den Darstellungstyp Dezimalzahl - Variablentyp Standard. Lediglich die Skalierung wird hier mittels einem Faktor im Gleitkommaformat erstellt; der Divisor entfällt damit. Von Gleitkommazahlen kann außerdem der Kehrwert gebildet werden, bevor der Wert angezeigt wird.

Das Format Gleitkommazahl wird nicht von jeder Steuerung unterstützt. Gleitkommazahlen werden in speziellen Gleitkommaformaten in der Steuerung verarbeitet (zum Beispiel IEEE-Format).

3.6.1.6 Darstellungstyp Hexadezimalzahl

Die hexadezimale Darstellung von Zahlen wird häufig bei der Anzeige von Adressen im SPS-Programmierbereich verwendet. Die Anzeige ist nur für Spezialisten geeignet und nicht für den normalen Bediener! Die Wertigkeit der angezeigten Stellen steigt von rechts nach links. Dargestellt werden die Ziffern 0 bis 9 und A bis F. Die Darstellung erfolgt ausschließlich in Großbuchstaben mit führenden Nullen. Sie bezieht sich auf die Datentypen Byte, Wort und LWort. Die maximale Länge beträgt 8 Stellen. Zwischen den Zeichen befinden sich keine Leerzeichen.

16 ⁴	16 ³	16 ²	16 ¹	16 ⁰	Wertigkeit
0	E	4	5	A	Anzeige (0E45A _H)

3.6.1.7 Darstellungstyp Binärzahl

Mit der binären Darstellung können Einzelbits, Byte, Wort oder LWort dargestellt werden. Die Eingabe der "Länge" in der Variablenbeschreibung entspricht der Anzahl der darzustellenden Bits. Die Zählung beginnt immer bei Bit 0. Die Anzahl der Leerstellen gibt an, wieviel Lücken (Spaces) zwischen den einzelnen Bits eingefügt werden. Die Darstellung erfolgt immer horizontal, wobei die Richtung der Wertigkeit in der Variablenbeschreibung eingestellt werden kann.

<u>Bit 3</u>	<u>Bit 2</u>	<u>Bit 1</u>	<u>Bit 0</u>	
0	1	0	0	Richtung = 76543210 (TSwin)
<u>Bit 0</u>	<u>Bit 1</u>	<u>Bit 2</u>	<u>Bit 3</u>	
0	0	1	0	Richtung = 01234567 (TSwin)

0100	Leerstellen = 0
0 1 0 0	Leerstellen = 1
0 1 0 0	Leerstellen = 2

3.6.1.8 Darstellungstyp Balken

Variablen können innerhalb von E/A-Masken als Balken dargestellt werden. Der Anwender kann die Balken horizontal oder vertikal einrichten. Die Balken können sich von einem Bezugspunkt in positive oder negative Richtung ausdehnen, also auch in der Mitte der Anzeige beginnen und sich in beide Richtungen ausdehnen. Der Wertebereich eines Balkens wird durch die Vorgabe zweier Eckwerte beschrieben. Die Balken werden mit Hilfe von vier unterschiedlichen grafischen Objekten, im einfachsten Fall durch Füllmuster, im Terminal dargestellt. Es kann jeweils ein Füllmuster angegeben werden für:

- den leeren Bereich des Balkens
- den gefüllten Balken
- den wertunterschrittenen Balken
- den wertüberschrittenen Balken

In der Projektiersoftware sind sieben verschiedene Füllmuster als Standard vorhanden, mit denen ein Balken vom Bedienterminal dargestellt werden kann. Für die freie Gestaltung der Füllmuster und Formen stehen die bereits erwähnten grafischen Objekte zur Wahl.

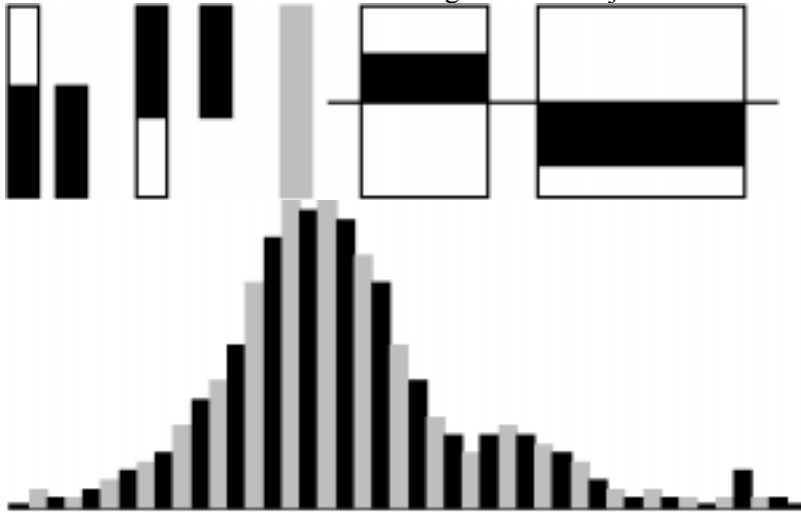


Bild 17: Balkendarstellung in verschiedenen Ausführungen

Für die Darstellung von Balkendiagrammen ist die einmalige, zyklische und ereignisgesteuerte Ausgabe möglich. Balkendiagramme werden zur reinen Istzustandsanzeige verwendet.

Die Ausgabe beschränkt sich auf einen Wertebereich von -32768 bis +32767.

Sind in einer Maske mehrere Balken vorhanden, dann sollten die Variablen so adressiert werden, daß sie gepackt übertragen werden können. Der Balken bewegt sich zwischen einem unteren und oberen Grenzwert, außerhalb der Grenzen erfolgt ein Umschlag in das jeweilige vordefinierte Füllmuster für die Bereichsüberschreitung.

Anwendungsbeispiele:

- vertikale und horizontale Trendanzeige
- optische Überwachung von Grenzwerten
- Histogrammdarstellungen
- Füllstandsanzeigen

Beispiel für die Darstellung eines Füllstandes:



18: Beispiel für die Füllstandsdarstellung

Für die Darstellung von Balkendiagrammen ist die einmalige oder zyklische Ausgabe möglich. Balkendiagramme werden zur reinen Istzustandsanzeige verwendet.

Bei der Balkendarstellung werden Einschränkungen zugunsten der Ausgabegeschwindigkeit vorgenommen. Balkendiagramme erstrecken sich deshalb immer über ein ganzes Vielfaches eines Charakters. Die kleinste Ausgabe hat die Größe eines Zeichens, die größtmögliche Ausgabe entspricht dem Display! Dabei „wächst“ ein Balkendiagramm immer um Pixelbreite.

Sind in einer Maske mehrere Balken vorhanden, dann sollten die Variablen so adressiert werden, daß sie zusammenhängend übertragen werden können.

3.6.1.9 Darstellungstyp Kurve

Die Kurvendarstellung erlaubt die Ausgabe von Wertetabellen als Punktlinie. Zur Kurvendarstellung muß in einer Maske eine Variable vom Typ Kurve definiert werden. Über Länge und Höhe wird die Ausdehnung der Kurve bestimmt. Ein eventuell benötigtes Koordinatensystem kann mit Hilfe von Hintergrundbildern eingeblendet werden.

Die Adresse der Kurvenvariable stellt den Beginn einer Wertetabelle in der SPS dar. Jeder Wert in der Tabelle beschreibt einen einzelnen Bildpunkt der Kurve. Die grafische Darstellung der Wertetabelle erfolgt wie bei zyklischen Ausgabevariablen.

Beispiele für die Darstellung des zeitlichen Verlaufs einer Variablen:

- Ausgabe einmaliger Vorgänge
- Speicherfunktion eines Punkteschreibers
- Füllstandsverläufe

Eine Kurve wird durch folgende Parameter begrenzt:

- Maximale Höhe: Höhe der Anzeige
- Maximale Breite: pro Kurvenvariable 54 Pixel

Soll eine Breite von mehr als 54 Pixel realisiert werden, so können mehrere direkt nebeneinander liegende Kurven ausgegeben werden.

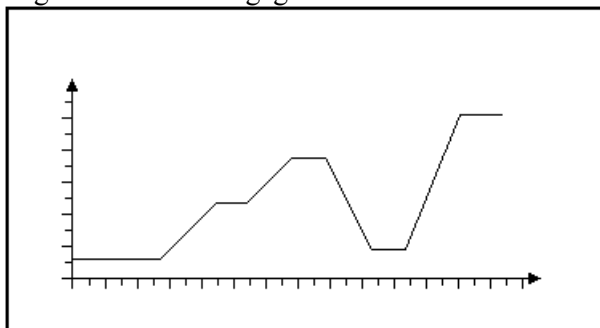


Bild 19: Beispiel einer Kurvendarstellung

Aus der Steuerung werden die Höhenangaben, also die Variablenwerte, aus einem zusammenhängenden Feld mit einem Leseauftrag ausgelesen. Die Höhenangabe in dem Feldelement mit der Anfangsadresse (*Adresse + 0*) wird ganz links dargestellt. Die folgenden Höhenangaben (*Adresse + n*) werden jeweils um eine Pixelposition nach rechts versetzt dargestellt. Die Höhenangaben für die Kurve werden zyklisch aufgefrischt.

Dazu wird in der Ausgabe punktweise gelöscht und dann geschrieben.

Für die Kurvendarstellung werden nur diese Parameter benötigt:

- Breite der Kurve, in Vielfachen der Zeichenbreite (Normalzeichensatz)
- Höhe der Kurve, in Vielfachen der Zeichenhöhe (Normalzeichensatz)

3.6.1.10 Darstellungstyp Auswahlfeld (TSdos)

Beim Auswahlfeld wird, wie beim kodierten Text, einem numerischen Wert aus der Steuerung ein Textstring zugeordnet. Dies erleichtert die Visualisierung von Betriebszuständen, Eingängen, Ausgängen usw. Daneben lassen sich ebenfalls Texte in Masken modifizieren. Im Auswahlfeld wird im Gegensatz zum kodierten Text die Textliste insgesamt oder ausschnittsweise dargestellt. Der angewählte Text ist markiert, indem die gesamte Auswahltextzeile invers dargestellt wird.

Ein Beispiel aus den Musterprojekten zeigt, wie die Namen der Maschinenbediener numerischen Werten zugeordnet sind. Die Zuordnungsnummern können zu den Texten frei vergeben werden. Die Zuordnungsnummern müssen weder lückenlos noch fortlaufend sortiert sein. Die Zeilen werden in Textlisten im Projektiersystem gehalten und können mehrfach verwendet werden. Die Darstellung ist auf Bit, Byte oder Wort beschränkt. Die Interpretation von LWord ist zwar möglich, bewirkt jedoch einen hohen Speicherbedarf.

<u>Wert</u>	<u>Text</u>
9	Alfred
1	Bernd
7	Detlef
4	Erwin
2	

<< hier wurden Leerzeichen eingetragen!

```
Alfred
Bernd
Detlef
Erwin
```

Die Steuerungsvariable enthält den Wert = 7. Ergebnis:

Das Auswahlfeld ist mit vier Einträgen definiert. Werden von der Steuerung Nummern angesprochen, die in der Textliste nicht definiert sind, so erscheinen auf dem Display in der Größe der Feldlänge, Fragezeichen.

Beispiel: Die Variable enthält den Wert = 5

??????

Dagegen erscheint beim Wert = 2 ein Leerstring.

3.6.2 Eingabevariablen

Eingabevariable werden bei erstmaliger Darstellung, also beim Aufbau der Maske, wie einmalige Ausgabevariablen behandelt. Sie werden also unter den gleichen Bedingungen dargestellt. Dies bezieht sich auch auf die Wirkung der Skalierung!

Die Skalierung wirkt aus der Sicht der SPS.

Eingabevariablen werden von Editoren im Terminal bearbeitet. Die eigentliche Funktionalität wird deshalb durch den Editortyp angewählt.

Bei der Eingabe werden die gleichen Typen unterstützt, wie bei der Variablenausgabe. Das Editieren der Variablen unterliegt bestimmten Bedingungen, die der Anwender bei der Projekterstellung beachten muß, zum Beispiel der Paßwortschutz und die externe Datenfreigabe.

Bei der Eingabe von Timern und Zählern müssen Besonderheiten beachtet werden:

Timer

Beim Schreiben auf eine Timervariable wird zunächst eine evtl. vorgenommene Skalierung rückgängig gemacht, so daß man wieder den Zeitwert zur Basis 10 ms erhält. Anschließend wird der BCD-kodierte Zeitwert so berechnet, daß die kleinstmögliche Zeitbasis verwendet wird.

Das Schreiben auf ein Zeitwort in der SPS sollte vermieden werden, da hierdurch auch die Steuerbits unkontrolliert beeinflußt werden.

Zähler

Beim Schreiben auf eine Zählervariable wird zunächst eine evtl. vorgenommene Skalierung rückgängig gemacht. Anschließend wird der BCD-kodierte Wert berechnet und dieser in die SPS übertragen.

Das Schreiben auf ein Zählwort in der SPS sollte vermieden werden, da hierdurch auch die Steuerbits unkontrolliert beeinflußt werden.

Formel für die Skalierung der Eingabe

Ein Eingabewert der aus der SPS nach der *Formel 1* berechnet ausgegeben wurde, wird nach dem Editieren mit der Umkehrfunktion nach *Formel 2* auf die SPS zurückgeschrieben. Die Umkehrfunktion wird im Terminal automatisch gebildet. Der Anwender hat die Operanden immer aus der Sicht der Steuerung zu bestimmen, auch bei der Eingabe!

$$\boxed{\text{SPS- Variablenwert}} = \frac{\boxed{\text{Terminal- Eingabewert}} - \text{Summand}}{\text{Faktor}} \times \text{Divisor}$$

Bild 20: Formel 2

Die positive Obergrenze ist ebenfalls bei der Rundung zu beachten. Die Berechnung erfolgt automatisch so, daß die letzte Stelle gerundet wird.

$$\boxed{(\text{Eingabewert} \times \text{Faktor}) < (\text{pos. Obergrenze} - \text{Divisor}/2)}$$

Bild 21: Formel 3

Plausibilitätskontrolle

Für alle Eingabevariablen wird eine Plausibilitätskontrolle durchgeführt. Dabei wird der eingegebene Wert gegen die, in der Variablenbeschreibung hinterlegten, Bereichsgrenzen geprüft. Beim Überschreiten der Grenzen werden entsprechende Systemmeldungen erzeugt. Der fehlerhafte Wert wird nicht in die Steuerung geschrieben. Im Fehlerfall bleibt der bis dahin gültige Wert erhalten.

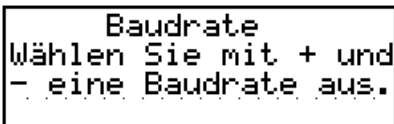
Werden die Systemmeldetexte „Wert ist zu groß“ und „Wert ist zu klein“ gelöscht, wird

- bei Wertunterschreitung der Wert der Untergrenze
- bei Wertüberschreitung der Wert der Obergrenze

eingetragen. Eine Systemmeldung wird dabei nicht ausgegeben.

Hilfetexte zu Eingabevariablen

Zu jeder Eingabevariablen kann ein ganzseitiger Hilfetext beschrieben werden. Die Ausgabe erfolgt innerhalb des Editors beim Betätigen der Hilfetaste. Wichtige Inhalte könnten die Bereichsgrenzen, Auswirkung der Variablen auf den zu bearbeitenden Prozeß oder eventuelle Abhängigkeiten zu weiteren Variablen sein.



```
Baudrate
Wählen Sie mit + und
- eine Baudrate aus.
```

Bild 22: Beispiel für Hilfetext zu Variable

Wird auf einen Hilfetext zur Variablen verzichtet, dann erscheint immer der Default-Hilfetext. Dieser ist ebenfalls ganzseitig und kann in der Projektiersoftware verändert werden. Ein Default-Hilfetext ist immer vorhanden, gegebenenfalls wird eine leere Displayseite angezeigt.

3.6.3 Systemvariablen

Mit Hilfe von Systemvariablen können bedienterminalinterne Funktionen gesteuert werden. Systemvariablen werden entweder mit Funktionstasten beziehungsweise Softkeytasten oder wie alle anderen Variablen in Masken zur Ein- und Ausgabe angelegt.

Wird eine Systemvariable mit einer Funktions- oder Softkeytaste verknüpft, gilt:

- einen Maskenwechsel und eine Systemvariable **nicht mit der gleichen Taste** verknüpfen
- Setzen (1) und Zurücksetzen (0) der Systemvariablen muß nicht mit der gleichen Funktions- oder Softkeytaste erfolgen
- Tipbetrieb wird erreicht durch Setzen (1) und Zurücksetzen (0) mit der gleichen Funktions- oder Softkeytaste

Die Namen von Systemvariablen dürfen nicht mit in die Variablenliste aufgenommen werden. Wird der Name einer Systemvariable auf eine SPS-Adresse bezogen, verliert sie ihre bedienterminalinterne Funktion.

Im den folgenden Unterkapiteln werden die Systemvariablen, nach Anwendungsgebieten gruppiert, aufgelistet und deren Funktion erklärt.

In TSdos haben alle Systemvariablen als Erkennungsmerkmal die Silbe **Sys** vorangestellt. In TSwin fehlt diese Silbe. In den folgenden Beschreibungen ist die Silbe **Sys** nicht mit angefügt.

3.6.3.1 Grundfunktionen

IntEraseEprom

Ermöglicht das Löschen der Anwenderbeschreibung im Flash-Speicher und bereitet das Terminal auf einen neuen Download über die Schnittstelle X3 vor. Es findet dabei keine Kommunikation mit einer eventuell angeschlossenen SPS statt. Das Schreiben einer "1" auf diese Variable löscht die Anwenderbeschreibung bedingungslos. Anschließend wird die Variable automatisch auf "0" zurückgesetzt.

Datentyp:	numerisch	
Editor:	Auswahltext (kodierter Text), positive Dezimalzahl	
Ausgabe:	---	
Mögliche Werte:	(0) inaktiv	Ruhelage
	(1) aktiv	löscht die Anwenderbeschreibung

MainVersion

Damit kann sich der Anwender die aktuelle Firmware-Version des Bedienterminals anzeigen lassen.

Datentyp:	alphanumerisch
Editor:	---
Ausgabe:	alphanumerisch, 8 Zeichen
Mögliche Werte:	vom Hersteller festgelegtes Format

ComVersion

Dient zur Anzeige von Typ und Versionsnummer des aktuellen Kommunikationsprotokolls.

Datentyp:	alphanumerisch
Editor:	---
Ausgabe:	alphanumerisch, 8 Zeichen
Mögliche Werte:	vom Hersteller festgelegtes Format

UserVersion

Teilt dem Bediener die Versionsnummer der Projektbeschreibung mit. Die Nummer wird vom Anwender bei der Projektierung im Projektiersystem eingegeben.

Datentyp: numerisch
Editor: ---
Ausgabe: numerisch, 3 Stellen
Mögliche Werte: ---

Boot

Mit dieser Variablen kann das Terminal zum Booten (Neustart des Systems) gebracht werden. Diese Variable ist dazu geeignet die Setup-Funktion in das normale Bediensystem zu integrieren. So können in jeder E/A-Maske Terminalparameter geändert werden, die anschließend eine Grundinitialisierung (Neustart des Systems) erforderlich machen. Das Schreiben einer "1" auf diese Variable läßt das Terminal booten. Anschließend wird die Variable automatisch auf "0" zurückgesetzt.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text),
positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) inaktiv Ruhelage
(1) aktiv Restart des Systems

LCDContrast

Dient zur Änderung des Kontrasts bei Terminals mit LCD-Display.

Datentyp: numerisch
Editor: Ganzzahl
Ausgabe: ---
Mögliche Werte: -127 bis +127 (je nach Terminaltyp weiter einschränken)

LCDBackground

Dient zur Einstellung des Anzeigehintergrundes. Diese Variable hat nur eine Funktion beim Bedienterminals mit entsprechendem Display; siehe dazu die Hinweise im jeweiligen technischen Handbuch.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Normaldarstellung
(1) inverser Hintergrund

LCDBackLight

Dient zur Einstellung der Anzeigehintergrundbeleuchtung. Diese Variable hat nur eine Funktion beim Bedienterminals mit entsprechendem Display; siehe dazu die Hinweise im jeweiligen technischen Handbuch. (Erst ab Firmware 6.40 verfügbar)

Datentyp: numerisch
Editor: beliebig
Ausgabe: ---
Mögliche Werte: (0) Hintergrundbeleuchtung AUS
(1 bis x) Hintergrundbeleuchtung EIN (gedimmt)

TurnOnTemp

Dient dazu, die LCD-Anzeige des Bedienterminals erst ab einer bestimmten Umgebungstemperatur einzuschalten. Diese Variable hat nur eine Funktion beim Bedienterminals mit entsprechender Funktionalität in Verbindung mit einem entsprechenden Display; siehe dazu die Hinweise im jeweiligen technischen Handbuch. (Erst ab Firmware 6.40 verfügbar)

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl

Ausgabe: ---
Mögliche Werte: (0) LCD-Anzeige AUS
(1 bis x) Temperaturwert, bei dessen Erreichen LCD-Anzeige EIN geschaltet wird

OsLanguage

Dient zur Online-Sprachumschaltung bei mehrsprachigen Anwenderbeschreibungen.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text)
Ausgabe: ---
Mögliche Werte: (0) erste Sprache
(n) n-te Sprache

3.6.3.2 Kommunikationsbereich X2

ComDataLenA

Definiert die Anzahl der Datenbits für die Kommunikation über die Schnittstelle X2 (X2.1).

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) 5 Bit/Char.
(1) 6 Bit/Char.
(2) 7 Bit/Char.
(3) 8 Bit/Char.

ComParityA

Bestimmt die Bildung und Prüfung der Parität für die Kommunikation über die Schnittstelle X2 (X2.1).

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Keine Parität
(1) Ungerade Parität
(2) Gerade Parität

ComStopBitsA

Definiert die Anzahl der Stoppbits für die Kommunikation über die Schnittstelle X2 (X2.1).

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) 1 Bit
(1) 1,5 Bit
(2) 2 Bit

ComBaudrateA

Definiert die Baudrate für die Kommunikation über die Schnittstelle X2 (X2.1).

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) 300 Baud
(1) 600 Baud
(2) 1200 Baud
(3) 2400 Baud
(4) 4800 Baud
(5) 9600 Baud
(6) 19200 Baud
(7) 38400 Baud
(8) 57600 Baud (nur BT35 / BT35C / TP35)

ComHandshakeA

Gibt die Art des Handshakes für die Kommunikation über die Schnittstelle X2 (X2.1) an.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Kein Handshake
(1) RTS/CTS Hardwarehandshake
(2) XON/XOFF Softwarehandshake

ComDefaultA

Mit dieser Variablen können neue Parameter für die Schnittstelle (X2) programmiert werden.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text),
positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) inaktiv Ruhelage
(1) aktiv Die in den zugehörigen Systemvariablen eingegebenen Daten werden als neue Parameter übernommen.
(1) aktiv Die im Flash-Speicher von der Projektiersoftware abgelegten Daten werden als neue Parameter übernommen.

ComTimeout

Gibt die Timeout-Überwachungszeit für die Schnittstelle X2 (X2.1) an. Der Wert (0) schaltet die Timeoutüberwachung ab.

Datentyp: numerisch
Editor: positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) inaktiv Ruhelage
(1 bis 65535) Timeout-Überwachung aktiv, Zeit in ms

ComRetryTimeout

Gibt die Zeit in Millisekunden an, die nach einem erfolglosen Verbindungsaufbau an der Kommunikationsschnittstelle X2, bis zum nächsten Versuch gewartet wird. Damit kann die SPS-abhängige Power-Up-Phase überbrückt werden, ohne daß eine Fehlermeldung generiert wird.

Datentyp: numerisch
Editor: positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: 0 bis 65535 ms

ComSlaveNr

Enthält die in der Vernetzung verwendete Slavennummer des Geräts. Mit dieser Adresse kann das Bedienterminal am Bus angesprochen werden.

Datentyp: numerisch
Editor: positive Dezimalzahl
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 255

3.6.3.3 Fehlerstatistik Schnittstelle X2**ComParityCount**

Fehlerzähler zur Überwachung der Paritätsfehler auf der Schnittstelle X2 zur SPS. Wird bei jedem Download gelöscht.

Datentyp: numerisch
Editor: (möglich)
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 65535

ComOverrunCount

Fehlerzähler zur Überwachung der Overrunfehler auf der Schnittstelle X2 zur SPS. Wird bei jedem Download gelöscht.

Datentyp: numerisch
Editor: (möglich)
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 65535

ComFrameCount

Fehlerzähler zur Überwachung der Framingfehler auf der Schnittstelle X2 zur SPS. Wird bei jedem Download gelöscht.

Datentyp: numerisch
Editor: (möglich)
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 65535

ComStatisticsTab

Die Systemvariable zeigt auf den Anfang der Protokollstatistik-Tabelle mit 6 Elementen zu je 4 Byte.

Datentyp: positive Dezimalzahl
Editor: ---
Ausgabe: Tabelle
Mögliche Werte: ---

ComErrorTab

Die Systemvariable zeigt auf den Anfang einer Tabelle mit 16 Elementen in denen die letzten 16 Kommunikationsfehler abgelegt sind.

Datentyp: positive Dezimalzahl
Editor: ---
Ausgabe: Tabelle
Mögliche Werte: ---

ComSubcodeTab

Die Systemvariable zeigt auf den Anfang einer Tabelle mit 16 Elementen in denen die Subcodes der Kommunikationsfehler abgelegt sind

Datentyp: positive Dezimalzahl
Editor: ---
Ausgabe: Tabelle
Mögliche Werte: ---

3.6.3.4 Kommunikationsbereich X3

ComDataLenB

Definiert die Anzahl der Datenbits auf der Schnittstelle (X3).

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) 5 Bit/Char.
(1) 6 Bit/Char.
(2) 7 Bit/Char.
(3) 8 Bit/Char.

ComParityB

Bestimmt die Bildung und Prüfung des Paritäts-Bit für die Schnittstelle (X3).

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Keine Parität
(1) Ungerade Parität
(2) Gerade Parität

ComStopBitsB

Definiert die Anzahl der Stoppbits auf der Schnittstelle (X3).

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) 1 Bit
(1) 1,5 Bit
(2) 2 Bit

ComBaudrateB

Definiert die Baudrate auf der Schnittstelle (X3).

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) 300 Baud
(1) 600 Baud
(2) 1200 Baud
(3) 2400 Baud
(4) 4800 Baud
(5) 9600 Baud
(6) 19200 Baud
(7) 38400 Baud
(8) 57600 Baud (nur BT35 / BT35C / TP35)

ComHandshakeB

Gibt die Art des Handshakes auf der Schnittstelle (X3) an.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Kein Handshake
(1) RTS/CTS Hardwarehandshake
(2) XON/XOFF Softwarehandshake

ComDefaultB

Mit dieser Variablen können neue Parameter für die Schnittstelle (X3) programmiert werden.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text),
positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) inaktiv Ruhelage
(1) aktiv Die in den zugehörigen Systemvariablen eingegebenen Daten
werden als neue Parameter übernommen.
(1) aktiv Die im Flash-Speicher von der Projektiersoftware abgelegten
Daten werden als neue Parameter übernommen.

3.6.3.5 Echtzeituhr

RTCSec

Mit dieser Variablen werden die Sekunden der Uhr angezeigt und gestellt.

Datentyp: positive Dezimalzahl
Editor: positive Dezimalzahl
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 59

RTCMin

Mit dieser Variablen werden die Minuten der Uhr angezeigt und gestellt.

Datentyp: positive Dezimalzahl
Editor: positive Dezimalzahl
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 59

RTCHour

Mit dieser Variablen werden die Stunden der Uhr angezeigt und gestellt.

Datentyp: positive Dezimalzahl
Editor: positive Dezimalzahl
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 23

RTCDay

Mit dieser Variablen wird der Tag des Datums angezeigt und gestellt.

Datentyp: positive Dezimalzahl
Editor: positive Dezimalzahl
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 31 vom Monat abhängig, Uhr korrigiert Fehleingaben beim nächsten
Datumswechsel

RTCMonth

Mit dieser Variablen wird der Monat des Datums angezeigt und gestellt.

Datentyp: positive Dezimalzahl
Editor: positive Dezimalzahl
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 12

RTCYear

Mit dieser Variablen wird das Jahr des Datums angezeigt und gestellt.

Datentyp: positive Dezimalzahl
Editor: positive Dezimalzahl
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 99 (es wird nur Jahr und Jahrzehnt beeinflusst)

RTCDayofWeek

Mit dieser Variablen wird der Wochentag angezeigt und festgelegt. Die Variable nimmt die Werte von 0 bis 6 an. Der Zähler läuft modulo. Die Ausgabe sollte über einen kodierten Text erfolgen. Die Zuordnung und der Anfangspunkt können beliebig festgelegt werden.

Datentyp: positive Dezimalzahl
Editor: Auswahltext (kodierter Text)
Ausgabe: Auswahltext (kodierter Text)
Mögliche Werte: 0 bis 6

RTCDateFmt

Dient zur Eingabe des Datumsformats für die Meldungsausgabe.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Europäisch TT MM JJ
(1) USA MM TT JJ
(2) JAPAN JJ MM TT

RTCYear2000

Mit dieser Variablen wird das Jahr des Datums mit vier Stellen angezeigt und gestellt.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), Auswahlbild, Alphanumerisch, Dezimal
Ausgabe: ---
Mögliche Werte: 0 bis 9999

3.6.3.6 Serielles Meldesystem

RepmanSortCrit

Dient zur Eingabe des Sortierkriteriums für die serielle Meldungsausgabe:

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Priorität der Meldungsnummer
(1) Zeitliches Eintreffen (neueste zuerst)
(2) Zeitliches Eintreffen (älteste zuerst)

ClearRepBuf

Dient zum Löschen des kompletten seriellen Meldungspuffers. Wahlweise Direktansteuerung durch Funktionstasten, Softkeys oder mittels Editor. Das Löschen des Meldungspuffers gegebenenfalls durch Paßworte oder spezielle Masken kontrollieren.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Daten erhalten
(1) Meldungspuffer löschen

RepmanRepPrint

Schaltet die Ausgabe von Meldungen über den Drucker ein.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) ausgeschaltet
(1) eingeschaltet

RepoutNr

Schaltet die Ausgabe der Meldungsnummer in der Meldungsmaske.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) aus
(1) ein

RepoutDate

Schaltet die Ausgabe des Datums der Meldung in der Meldungsmaske.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) aus
(1) ein

RepoutTime

Schaltet die Ausgabe der Uhrzeit der Meldung in der Meldungsmaske.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) aus
(1) ein

RepoutAnzYear

Gibt die Anzahl der Stellen für die Darstellung der Jahreszahl vor.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Jahreszahl zweistellig
(1) Jahreszahl vierstellig

RepoutRepText

Dient zur Ausgabe der aktuellsten seriellen Meldung. Die Meldung wird so angezeigt, wie sie in der Meldungsmaske angezeigt würde, das heißt entsprechend der gewählten Meldungsparameter.

Datentyp: alphanumerisch
Editor: ---
Ausgabe: alphanumerisch
Mögliche Werte: beliebig

RepoutRepText21

Dient zur Ausgabe der aktuellsten Meldung des seriellen Meldesystems, ab der 21. Stelle mit variabler Länge (20, 40 oder 60). Die Meldung wird so angezeigt, wie sie in der Meldungsmaske angezeigt würde, das heißt entsprechend der gewählten Meldungsparameter.

Datentyp: alphanumerisch
Editor: ---
Ausgabe: alphanumerisch
Mögliche Werte: beliebig

RepoutRepText41

Dient zur Ausgabe der aktuellsten Meldung des seriellen Meldesystems, ab der 41. Stelle mit variabler Länge (40 oder 60). Die Meldung wird so angezeigt, wie sie in der Meldungsmaske angezeigt würde, das heißt entsprechend der gewählten Meldungsparameter.

Datentyp: alphanumerisch
Editor: ---
Ausgabe: alphanumerisch
Mögliche Werte: beliebig

RepoutRepText61

Dient zur Ausgabe der aktuellsten Meldung des seriellen Meldesystems, ab der 61. Stelle. Gibt die letzten 20 Zeichen einer Meldung aus. Die Meldung wird so angezeigt, wie sie in der Meldungsmaske angezeigt würde, das heißt entsprechend der gewählten Meldungsparameter.

Datentyp: alphanumerisch
Editor: ---
Ausgabe: alphanumerisch
Mögliche Werte: beliebig

3.6.3.7 Paralleles Meldesystem

RepmanSortCritP

An dieser Variablen erkennt das Terminal, nach welchem Kriterium die Sortierung der Meldungen bei der Anzeige erfolgen soll. Der Grundwert wird in der Anwenderbeschreibung festgelegt. Späteres Editieren dieser Variablen erlaubt die nachträgliche Anpassung.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Priorität der Meldungsnummer
 (1) Zeitliches Eintreffen (neueste zuerst)
 (2) Zeitliches Eintreffen (älteste zuerst)

RepoutNrP

An dieser Variablen erkennt das Terminal, ob im parallelen Meldesystem die Meldungsnummer mit dem Meldetext ausgegeben werden soll oder nicht. Die Grundeinstellung wird in der Anwenderbeschreibung festgelegt. Späteres Editieren auf dieser Variablen erlaubt die nachträgliche Anpassung.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) aus
(1) ein

RepoutDateP

An dieser Variablen erkennt das Terminal, ob im parallelen Meldesystem das Datum mit dem Meldetext ausgegeben werden soll oder nicht. Die Grundeinstellung wird in der Anwenderbeschreibung festgelegt. Späteres Editieren auf dieser Variablen erlaubt die nachträgliche Anpassung.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) aus
(1) ein

RepoutTimeP

An dieser Variablen erkennt das Terminal, ob im parallelen Meldesystem die Zeit mit dem Meldetext ausgegeben werden soll oder nicht. Der Grundwert wird in der Anwenderbeschreibung festgelegt. Späteres Editieren auf dieser Variablen erlaubt die nachträgliche Anpassung.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) aus
(1) ein

RepoutAnzYearP

Gibt die Anzahl der Stellen für die Darstellung der Jahreszahl vor.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Jahreszahl zweistellig
(1) Jahreszahl vierstellig

RepoutRepTextP

Mit dieser Variable gibt das Terminal die aktuellste Meldung des parallelen Meldesystems aus.

Datentyp: alphanumerisch
Editor: ---
Ausgabe: alphanumerisch
Mögliche Werte: beliebig

RepoutRepTextP21

Dient zur Ausgabe der aktuellsten Meldung des parallelen Meldesystems, ab der 21. Stelle mit variabler Länge (20, 40 oder 60). Die Meldung wird so angezeigt, wie sie in der Meldungsmaske angezeigt würde, das heißt entsprechend der gewählten Meldungsparameter.

Datentyp: alphanumerisch
Editor: ---
Ausgabe: alphanumerisch
Mögliche Werte: beliebig

ReputRepTextP41

Dient zur Ausgabe der aktuellsten Meldung des parallelen Meldesystems, ab der 41. Stelle mit variabler Länge (40 oder 60). Die Meldung wird so angezeigt, wie sie in der Meldungsmaske angezeigt würde, das heißt entsprechend der gewählten Meldungsparameter.

Datentyp: alphanumerisch

Editor: ---

Ausgabe: alphanumerisch

Mögliche Werte: beliebig

ReputRepTextP61

Dient zur Ausgabe der aktuellsten Meldung des parallelen Meldesystems, ab der 61. Stelle. Gibt die letzten 20 Zeichen einer Meldung aus. Die Meldung wird so angezeigt, wie sie in der Meldungsmaske angezeigt würde, das heißt entsprechend der gewählten Meldungsparameter.

Datentyp: alphanumerisch

Editor: ---

Ausgabe: alphanumerisch

Mögliche Werte: beliebig

3.6.3.8 Druckersteuerung

StopPrint

Beim Aktivieren der Systemvariablen wird ein gerade laufender Druckvorgang abgebrochen.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
 (1) Druck abbrechen

BlockPrint

Beim Aktivieren der Systemvariablen wird der in der Meldemaske markierte Block ausgedruckt.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
 (1) Block ausdrucken

PrintAllRep

Beim Aktivieren der Systemvariablen wird der komplette, serielle Meldespeicher ausgedruckt. Die Ausgabe erfolgt in der voreingestellten Weise.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
 (1) Ausdruck mit Formatierung
 (2) Ausdruck mit voller Länge

PrintAllState

Beim Aktivieren der Systemvariablen wird der komplette, parallele Meldespeicher (Zustandsmeldungen) ausgedruckt.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
 (1) Meldepuffer ausdrucken

BlockPrintLong

Beim Aktivieren der Systemvariablen wird der markierte Bereich in einer Meldungsmaske in voller Länge ausgedruckt. Die Einstellungen, die beim Projektieren gemacht wurden, bleiben unberücksichtigt.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Markierten Bereich drucken

3.6.3.9 Menüsteuerung / Tasten

NewMask

Nach Beschreiben der Systemvariablen mit einer Maskennummer wird anschließend ein Maskenwechsel in die entsprechende Maske veranlaßt.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), Auswahlfeld
Ausgabe: ---
Mögliche Werte: 1 bis 9999

VarTablenR0

Erzeugt eine fortlaufende Numerierung in Tabellen. Die Numerierung beginnt bei "0". Die Systemvariable dient außerdem zur Ausgabe von konstanten Tabellentexten.

Datentyp: numerisch
Editor: ---
Ausgabe: Auswahltext (kodierter Text), positive Dezimalzahl
Mögliche Werte: 0 bis n

VarTablenR1

Erzeugt eine fortlaufende Numerierung in Tabellen. Die Numerierung beginnt bei "1". Die Systemvariable dient außerdem zur Ausgabe von konstanten Tabellentexten.

Datentyp: numerisch
Editor: ---
Ausgabe: Auswahltext (kodierter Text), positive Dezimalzahl
Mögliche Werte: 1 bis n

HardCopy

Dient zum Upload von PCX/oder ASCII-Dateien mit dem Displayinhalt über die Schnittstelle X3 an den angeschlossenen PC.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Hardcopy starten (immer aktuelle Displayseite)

TabLeft

Mit Hilfe dieser Systemvariable können Spaltenwechsel nach links in Tabellen ausgeführt werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) eine Spalte nach links

TabRight

Mit Hilfe dieser Systemvariable können Spaltenwechsel nach rechts in Tabellen ausgeführt werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) eine Spalte nach rechts

Datentyp: numerisch

TabPgUp

Mit Hilfe dieser Systemvariable können Softkeys zum Seitenwechsel (nach oben) in Tabellen verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) eine Spalte nach oben blättern

TabPgDn

Mit Hilfe dieser Systemvariable können Softkeys zum Seitenwechsel (nach unten) in Tabellen verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) eine Spalte nach unten blättern

Shift

Bei gesetzter Systemvariable können Sie alphanumerische Zeichen eingeben lassen. Durch Drücken der Tasten vom numerischen Tastenblock erhält man automatisch die zugeordneten Buchstaben im Alphabet. Durch mehrfaches Drücken werden die zugeordneten Buchstaben durchlaufen.

Dabei werden nur Großbuchstaben zur Verfügung gestellt.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Shiftmode 1 aktiv, Großbuchstaben

<u>Taste</u>	<u>Buchstaben (Zeichen)</u>	<u>Taste</u>	<u>Buchstaben (Zeichen)</u>
Punkt	: ? ! .	Drei	Y Z % 3
Minus	\ * / -	Vier	J K L 4
Plus	< = > +	Fünf	M N O 5
Null	() ° 0	Sechs	P Q R 6
Eins	S T U 1	Sieben	A B C 7
Zwei	V W X 2	Acht	D E F 8

ShiftCase

Bei gesetzter Systemvariable können Sie alphanumerische Zeichen eingeben lassen. Durch Drücken der Tasten vom numerischen Tastenblock erhält man automatisch die zugeordneten Buchstaben im Alphabet. Durch mehrfaches Drücken werden die zugeordneten Buchstaben durchlaufen.

Dabei werden Klein- und Großbuchstaben zur Verfügung gestellt.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Shiftmode 2 aktiv, Klein- und Großbuchstaben

<u>Taste</u>	<u>Buchstaben (Zeichen)</u>	<u>Taste</u>	<u>Buchstaben (Zeichen)</u>
Punkt	: ? ! .	Drei	Y Z % y z % 3
Minus	\ * / -	Vier	J K L j k l 4
Plus	< = > +	Fünf	M N O m n o 5
Null	() ° 0	Sechs	P Q R p q r 6
Eins	S T U s t u 1	Sieben	A B C a b c 7
Zwei	V W X v w x 2	Acht	D E F d e f 8

KeyCursLeft

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Cursor Links* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Cursor Links* behandeln

KeyCursRight

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Cursor Rechts* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Cursor Rechts* behandeln

KeyCursUp

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Cursor Oben* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Cursor Oben* behandeln

KeyCursDown

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Cursor Unten* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Cursor Unten* behandeln

KeyHome

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Cursor Home* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Cursor Home* behandeln

KeyHelp

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Hilfe* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Hilfe* behandeln

KeyDot

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Dezimalpunkt* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Dezimalpunkt* behandeln

KeyClear

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Löschen* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Löschen* behandeln

Key0

Mit Hilfe dieser Systemvariable können Softkeys als Taste *0 (Null)* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *0 (Null)* behandeln

Key1

Mit Hilfe dieser Systemvariable können Softkeys als Taste *1* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *1* behandeln

Key2

Mit Hilfe dieser Systemvariable können Softkeys als Taste 2 verwendet werden.

Datentyp: numerisch

Editor: Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste 2 behandeln

Key3

Mit Hilfe dieser Systemvariable können Softkeys als Taste 3 verwendet werden.

Datentyp: numerisch

Editor: Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste 3 behandeln

Key4

Mit Hilfe dieser Systemvariable können Softkeys als Taste 4 verwendet werden.

Datentyp: numerisch

Editor: Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste 4 behandeln

Key5

Mit Hilfe dieser Systemvariable können Softkeys als Taste 5 verwendet werden.

Datentyp: numerisch

Editor: Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste 5 behandeln

Key6

Mit Hilfe dieser Systemvariable können Softkeys als Taste 6 verwendet werden.

Datentyp: numerisch

Editor: Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste 6 behandeln

Key7

Mit Hilfe dieser Systemvariable können Softkeys als Taste 7 verwendet werden.

Datentyp: numerisch

Editor: Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste 7 behandeln

Key8

Mit Hilfe dieser Systemvariable können Softkeys als Taste *8* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *8* behandeln

Key9

Mit Hilfe dieser Systemvariable können Softkeys als Taste *9* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *9* behandeln

KeyPlus

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Plus* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Plus* behandeln

KeyMinus

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Minus* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Minus* behandeln

KeyEnter

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Eingabe* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Eingabe* behandeln

KeyEdit

Mit Hilfe dieser Systemvariable können Softkeys als Taste *Freigabe* verwendet werden.

Datentyp: numerisch
Editor: Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Softkeytaste als Taste *Freigabe* behandeln

3.6.3.10 Paßwort

MskchgPasswd

Dient zur Eingabe des Paßworts innerhalb einer Maske.

Datentyp: numerisch (alphanumerisch nur bei Terminals mit entsprechenden Tasten)

Editor: alphanumerisch

Ausgabe: ---

Mögliche Werte: 11 Zeichen

MskchgResPasswd

Dient zum Löschen des aktuell eingegebenen Paßworts, setzt die Zugriffsberechtigung zurück. Die Zugriffslevel werden bei jedem Power-Up zurückgesetzt.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
(1) Zugriffsberechtigung rücksetzen

ChangePasswd

Mit Hilfe der Systemvariablen können die Paßworte im Terminal geändert werden.

Datentyp: numerisch (alphanumerisch nur bei Terminals mit entsprechenden Tasten)

Editor: alphanumerisch

Ausgabe: ---

Mögliche Werte: 11 Zeichen

FlashPasswd

Mit Hilfe dieser Systemvariablen werden die Paßworte auf die in der Programmiersoftware angegebenen Codewerte zurückgesetzt. Nützlich bei Verlust der Paßworte. Das Masterpaßwort sollte vorher gesichert werden.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste

Ausgabe: ---

Mögliche Werte: (0) Ruhelage
(1) Paßworte rücksetzen

Passwdlnactive

Dient zum Deaktivieren des Paßwortschutzes. Die Systemvariable ist batteriegepuffert. Die letzte Einstellung bleibt beim Ausschalten erhalten.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text), positive Dezimalzahl

Ausgabe: ---

Mögliche Werte: (0) Paßwortschutz aktiv, (Ruhelage bei Erstinitialisierung)
(1) Paßwortschutz inaktiv (Edit- und Viewlevel = 255)

3.6.3.11 Rezepturen

SelectDSNr

Die Variable enthält die Nummer des aktiven Datensatzes. Die Variable kann nur mit dem zugehörigen Auswahltexteditor bearbeitet werden.

Datentyp: numerisch
Editor: Auswahltext (Auswahlfeld für TSdos)
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 250

SelectDSName

Die Variable enthält den Namen des aktiven Datensatzes. Die Variable kann nur mit dem zugehörigen Auswahltexteditor bearbeitet werden.

Datentyp: alphanumerisch
Editor: Auswahltext (Auswahlfeld für TSdos)
Ausgabe: alphanumerisch
Mögliche Werte: 15 Zeichen

DestDSNr

Die Variable enthält die Nummer des Zieldatensatzes beim Kopieren von Datensätzen.

Datentyp: numerisch
Editor: Auswahltext (Auswahlfeld für TSdos)
Ausgabe: positive Dezimalzahl
Mögliche Werte: 0 bis 250

DSCopy

Mit dieser Variablen kann der aktive Datensatz auf das unter **DestDSNr** angegebene Ziel kopiert werden.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) Kopiert mit Kopierziel in **DestDSNr**
(2) Kopiert und sucht automatisch einen freien Datensatz
(3) Kopiert mit Kopierziel in **DestDSNr** und überschreibt einen eventuell vorhandenen Datensatz

DSDelete

Mit dieser Variablen kann der aktive Datensatz gelöscht werden. Der erste Datensatz aus der gleichen Rezeptur wird der neue aktive Datensatz.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) löscht den Datensatz.

ActDSName

Die Variable enthält den Namen des aktuellen Datensatzes. Es kann gelesen und bei RAM-Datensätzen auch geschrieben werden.

Datentyp: alphanumerisch
Editor: alphanumerisch
Ausgabe: alphanumerisch
Mögliche Werte: 15 Zeichen

SelectRezeptNr

Die Variable enthält die aktive Rezeptur. Die Variable kann auch außerhalb der Rezepturmaske geändert werden.

Datentyp: numerisch
Editor: numerisch, Auswahltext (kodierter Text)
Ausgabe: numerisch, Auswahltext (kodierter Text)
Mögliche Werte: 1 bis 250

DSDownload

Mit Hilfe dieser Variablen kann der aktive Datensatz auf die Steuerung geschrieben werden.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) schreibt den Datensatz.

DSDnloadBreak

Mit Hilfe dieser Variablen kann ein laufender Datentransfer zur Steuerung abgebrochen werden.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text), positive Dezimalzahl, Softkey, Funktionstaste
Ausgabe: ---
Mögliche Werte: (0) Ruhelage
(1) beendet den Datentransfer.

DSDnloadState

Mit Hilfe dieser Variablen läßt sich der Datentransfer zur Steuerung überwachen.

Datentyp: numerisch
Editor: ---
Ausgabe: Auswahltext (kodierter Text)
Mögliche Werte: (0) Grundzustand
(1) Datentransfer wurde angefordert, aber von der Steuerung noch nicht freigegeben.
(2) Datentransfer aktiv

LoadDSName

Diese Variable enthält den zuletzt an die Steuerung übertragenen Datensatznamen. Falls der Datensatz inzwischen gelöscht wurde, erscheinen Fragezeichen.

Datentyp: alphanumerisch
Editor: ---
Ausgabe: alphanumerisch
Mögliche Werte: 15 Zeichen

StartSave

Mit dieser Variablen können Datensätze zum PC übertragen werden.

Datentyp: numerisch
Editor: Auswahltext (kodierter Text)
Ausgabe: ---
Mögliche Werte: (0) Grundzustand
(1) Einzelnen Datensatz an den PC übertragen
(2) Alle Datensätze einer Rezeptur an den PC übertragen
(3) Alle Datensätze im Terminal an den PC übertragen

SaveState

Die Variable gibt bei einer Übertragung an den PC den aktuellen Übertragungsstatus an.

Datentyp: numerisch

Editor: ---

Ausgabe: Auswahltext (kodierter Text)

Mögliche Werte: (0) Grundzustand
(1) Ein einzelner Datensatz wird in den PC übertragen
(2) Alle Datensätze einer Rezeptur werden in den PC übertragen
(3) Alle Datensätze im Terminal werden in den PC übertragen

StartRestore

Die Variable steuert den Ladevorgang vom PC ins Terminal.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text)

Ausgabe: ---

Mögliche Werte: (0) Grundzustand
(1) Das Terminal geht in Empfangsbereitschaft
(2) Das Terminal bricht eine laufende Übertragung ab.

RestoreState

Die Variable zeigt den Status der Übertragung vom PC ins Terminal an.

Datentyp: numerisch

Editor: ---

Ausgabe: Auswahltext (kodierter Text)

Mögliche Werte: (0) Grundzustand
(1) Datentransfer läuft

RestoreLineNr

Die Variable zeigt die aktuelle Zeilennummer der Datensatzdatei an. Dient zur Aktivitätskontrolle und im Fehlerfall zur Lokalisierung des Fehlers.

Datentyp: numerisch

Editor: ---

Ausgabe: numerisch

Mögliche Werte: 1 bis 255

StartRezPrint

Mit dieser Variable können Datensätze ausgedruckt werden.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text)

Ausgabe: ---

Mögliche Werte: (0) Grundzustand
(1) Startet das Ausdrucken des Datensatzes
(2) Der Druckvorgang wird abgebrochen.

RezPrintState

Die Variable zeigt den aktuellen Druckzustand beim Datensatzdrucken an.

Datentyp: numerisch

Editor: ---

Ausgabe: Auswahltext (kodierter Text)

Mögliche Werte: (0) Grundzustand
(1) Ausdrucken von Datensatz läuft

StartUpload

Mit dieser Variablen kann für die aktive Rezeptur ein Datensatz aus der Steuerung ausgelesen und im Terminal gespeichert werden.

Datentyp: numerisch

Editor: Auswahltext (kodierter Text)

Ausgabe: ---

Mögliche Werte:

- (0) Grundzustand
- (1) Variablen werden einzeln aus ihren spezifizierten Adressen gelesen
- (2) Variablen werden als Block aus dem für die Rezeptur angegebenen Puffer gelesen
- (3) Variablen werden von ihren spezifizierten Adressen gelesen. Es wird automatisch ein freier Datensatz gesucht. Systemmeldung 18 erscheint, wenn kein freier Datensatz vorhanden ist.
- (4) Variablen werden im Block aus dem für die Rezeptur angegebenen Puffer gelesen. Es wird automatisch ein freier Datensatz gesucht. Systemmeldung 18 erscheint, falls kein freier Datensatz vorhanden ist.

UploadDSNr

Die Variable gibt die Datensatznummer an, auf die der hochgeladene Datensatz im Terminal geschrieben werden soll.

Datentyp: numerisch

Editor: numerisch

Ausgabe: ---

Mögliche Werte: 1 bis 250

UploadState

Diese Variable zeigt den Status beim Hochladen von Datensätzen zum Terminal an.

Datentyp: numerisch

Editor: ---

Ausgabe: Auswahltext (kodierter Text)

Mögliche Werte:

- (0) Grundzustand
- (1) Hochladen von Datensatz läuft

3.6.3.12 Betriebsstundenzähler

Counter1

Counter2

Counter3

Counter4

Counter5

Counter6

Counter7

Counter8

Dient in Verbindung mit der Start/Stop-Funktion des zugehörigen Bits als Betriebsstundenzähler. Solange das Bit gesetzt ist, wird der Zähler erhöht.

Datentyp: numerisch

Editor: positive Dezimalzahl

Ausgabe: positive Dezimalzahl

Mögliche Werte: 0 bis 4.294.967.295

3.6.3.13 Durchschleifbetrieb

Pg2Sps

Aktiviert und deaktiviert abwechselnd den Durchschleifbetrieb (Toggle-Funktion). Das PG-Protokoll muß dafür ausgestattet sein!

Datentyp: numerisch

Editor: ---

Ausgabe: Auswahltext (kodierter Text), positive Dezimalzahl

Mögliche Werte: (0) Grundzustand
(1) Erste 1 aktiviert, zweite 1 deaktiviert Durchschleifbetrieb

Pg2SpsState

Zeigt den Status des Durchschleifbetriebs an.

Datentyp: numerisch

Editor: ---

Ausgabe: Auswahltext (kodierter Text), positive Dezimalzahl

Mögliche Werte: (0) Grundzustand (AUS)
(1) Durchschleifbetrieb anfordern
(2) Durchschleifbetrieb möglich
(3) Durchschleifbetrieb aktiv

3.6.3.14 Ladbarer Zeichensatz

ChrsetName

Dient zur Anzeige vom aktuellen Zeichensatznamen.

Der Name des Zeichensatzes darf eine Länge von 8 Zeichen nicht überschreiten!

Datentyp: alphanumerisch

Editor: ---

Ausgabe: ---

Mögliche Werte: (Standard) Darstellung mit Terminalzeichensatz
(Zeichensatzname) Darstellung mit selbst entworfenem Zeichensatz

3.6.3.15 Wartung

User1

User2

User3

User4

User5

Universelle Variablen, die dem Anwender die Ablage von Informationen im Terminal ermöglichen.

Die Daten bleiben im gepufferten RAM erhalten.

Datentyp: beliebig

Editor: beliebig

Ausgabe: beliebig

Mögliche Werte: 16 Bit

LCDADCInput

Dient zum Auslesen vom aktuellen Eingangswert des AD-Wandlers, der für die LCD-Kontrastregelung verwendet wird. Dieser Wert ist nur für Diagnosezwecke sinnvoll.

Datentyp: numerisch

Editor: beliebig

Ausgabe: ---

LCDDACOutput

Dient zum Auslesen vom aktuellen Eingangswert des DA-Wandlers, der für die LCD-Kontrastregelung verwendet wird. Dieser Wert ist nur für Diagnosezwecke sinnvoll.

Datentyp: numerisch
Editor: beliebig
Ausgabe: ---

Break

Der aktuelle Editor wird unterbrochen und die eingegebenen Werte werden nicht an die Steuerung übergeben. Lösen Sie die Funktion Break mit einer Funktions- oder Softkeytaste aus.

Datentyp: numerisch
Editor: Funktionstaste, Softkeytaste, Auswahltext, Dezimal-, Binär-, Hexadezimalzahl
Ausgabe: ---
Mögliche Werte: (0) Ruhestellung
(1) Abbruch des Editiervorgangs

3.6.3.16 Editoren

EditInvers

Gibt an, ob der Editor während der Eingabe invers geschaltet werden soll.

Datentyp: numerisch
Editor: beliebig
Ausgabe: ---
Mögliche Werte: (0) normale Darstellung
(1) inverse Darstellung

EditEnter

Gibt an, wie sich der Editor bei Betätigung der Datenfreigabetaste verhalten soll.

Datentyp: numerisch
Editor: ---
Ausgabe: ---
Mögliche Werte: (0) Editor wechselt auf das nächste Eingabefeld (Standardeinstellung)
(1) Editor bleibt auf der aktuellen Position stehen. Die Weiterschaltung muß mittels Cursor erfolgen.

StatePerm

Gibt den Status der Zustands-LED von der Taste Datenfreigabe an.

Datentyp: numerisch
Editor: beliebig
Ausgabe: 0, 1, 2
Mögliche Werte: (0) Zustands-LED AUS
(1) Zustands-LED EIN
(2) Zustands-LED BLINKT

3.6.3.17 Hilfe

StateHelp

Gibt den Status der Zustands-LED von der Taste Hilfe an.

Datentyp: numerisch
Editor: beliebig
Ausgabe: 0, 1, 2
Mögliche Werte: (0) Zustands-LED AUS
(1) Zustands-LED EIN
(2) Zustands-LED BLINKT

Message

Tritt ein Systemfehler auf, wird die Nummer der Systemmeldung in diese Variabel geschrieben.

Datentyp: numerisch
Editor: beliebig
Ausgabe: ---
Mögliche Werte: (0) Ruhezustand
(1 bis 29) Systemmeldungsnummer

QuitMessage

Quittiert die Systemmeldung, die aktuell durch die Systemvariable **Message** angezeigt wird.

Datentyp: numerisch
Editor: beliebig
Ausgabe: ---
Mögliche Werte: (0) Ruhezustand
(1) Quittieren

3.6.4 Editoren

In E/A-Masken stehen Editoren für die verschiedenen Datentypen zur Wahl. Unter Editoren versteht man Programmteile, die die komfortable Änderung der Eingabevariablen erlauben. Diese zur Änderung erforderliche Ablaufsteuerung ist komplett in das Terminal integriert. In der Steuerung ist dazu keine zusätzliche "Software" erforderlich.

Auf die Editoren werden beeinflusst durch

- die Datenfreigabe (insbesondere von der angeschlossenen Steuerung aus)
- das zyklische Auffrischen der Werte
- das Hilfesystem
- und die Plausibilitätskontrolle.

Die Editoren werden mittels Editiertasten und Steuertasten bedient. Die jeweilige Funktion ist der einzelnen Editorbeschreibung zu entnehmen.

Durch verschiedene Editoren können die Variablen der Steuerung modifiziert werden. Die numerischen Variablen werden bei der Eingabe auf Plausibilität geprüft. Der Anwender legt die Bereichsgrenzen in der Variablenbeschreibung fest und kann außerdem einen variablenspezifischen Hilfetext beifügen. In diesem Text kann die Funktion der Variablen und deren zulässiger Wertebereich beschrieben sein.

Die Eingabe von Variablen geschieht wie folgt:

Befindet sich in einer E/A-Maske ein editierbarer Wert, so erlaubt die Taste Datenfreigabe in den Editor zu wechseln. Im Editiermode leuchtet die Zustands-LED der Taste Datenfreigabe, die Schreibmarke zeigt auf die erste editierbare Variable. Nun läßt sich dieser Wert über die numerische Tastatur und/oder mit den Tasten Plus/Minus ändern. Der Wert wird beim Betätigen der Taste Dateneingabe (Enter) übernommen. Dabei wird die Plausibilität geprüft und bei einem Fehler blinkt die Zustands-LED der Taste Hilfe.

Die blinkende Zustands-LED der Taste Hilfe soll den Bediener auf eine Fehlfunktion hinweisen. Beim Betätigen der Taste Hilfe wird die Fehlfunktion beschrieben. Das Editierfeld kann bei einem Fehler mit der Taste Dateneingabe nicht verlassen werden.

Dagegen verlassen Steuertasten auch dann ein Editierfeld, wenn der Wert ungültig ist, allerdings wird der zuletzt eingegebene Wert durch den alten Wert (Vorzustand) ersetzt.

Die Steuertasten dienen im Editiermode zur Anwahl der editierbaren Variablen. Der Editiermode wird beim erneuten Betätigen der Taste Datenfreigabe wieder verlassen und die Zustands-LED erlischt.

Derzeit stehen folgende Editoren bei der Auswahl in der Variablenbeschreibung zur Verfügung:

TSdos-Editoren	TSwin-Editoren
Ganzzahl	Dezimalzahl / keine Nachkommastellen
Festkommazahl	Dezimalzahl / Nachkommastellen
Fließkommazahl	Gleitkommazahl
Auswahltext (kodierter Text)	Auswahltext
Kodiertes Bild	Auswahlbild
Auswahlfeld	Auswahltext
Kurvendiagramm	Kurve
Balkendiagramm	Balken
Alphanumerisch	Alphanumerisch
Hexadezimalzahl	Hexadezimalzahl
Binärzahl	Binärzahl
BCD-Zahl	Dezimalzahl / Attribut BCD
Timer	Dezimalzahl / Attribut Timer
Zähler	Dezimalzahl / Attribut Zähler
Paßwort	Alphanumerisch / Paßwort

Die Tastenfunktionen von allen numerischen Editoren sind identisch.

Tastenfunktion numerischer Editor

Taste: Cursor Oben	Bewegt die Schreibmarke auf dem Display zur nächsthöheren editierbaren Variable und markiert sie. Wenn die Schreibmarke die oberste editierbare Variable erreicht hat, wird die unterste Variable als nächstes markiert.
Taste: Cursor Unten	Bewegt die Schreibmarke auf dem Display zur nächstniedrigeren editierbaren Variable und markiert sie. Wenn die Schreibmarke die unterste editierbare Variable erreicht hat, wird die oberste Variable als nächstes markiert.
Taste: Cursor Links	Bewegt die Schreibmarke innerhalb der editierbaren Variable um eine Stelle nach links bis maximal zur Feldgrenze.
Taste: Cursor Rechts	Bewegt die Schreibmarke innerhalb der editierbaren Variable um eine Stelle nach rechts bis maximal zur Feldgrenze.
Taste: Plus	<ol style="list-style-type: none"> 1. Variable im Moment angewählt: Wert wird gelöscht, neuer Wert kann eingegeben werden 2. Schreibmarke wurde innerhalb eines positiven Werts bewegt: Keine Änderung 3. Schreibmarke wurde innerhalb eines negativen Werts bewegt: Löscht das negative Vorzeichen
Taste: Minus	<ol style="list-style-type: none"> 1. Variable im Moment angewählt: Wert wird gelöscht, negatives Vorzeichen wird an niederwertigster Position eingefügt, neuer Wert kann eingegeben werden 2. Schreibmarke wurde innerhalb eines positiven Werts bewegt: Negatives Vorzeichen wird dem Wert vorangestellt 3. Schreibmarke wurde innerhalb eines negativen Werts bewegt: Keine Änderung
Taste: Löschen	Löscht die Stelle, an der sich die Schreibmarke befindet (auch Vorzeichen).

3.6.4.1 Dezimalzahleditor

Der Dezimalzahleditor ist ein numerischer Editor, wahlweise nur für positive oder positive/negative Dezimalzahlen.

Zusätzlich können für Dezimalzahlen Variablentypen ausgewählt werden.

- Standardtyp Ganzzahlen / Festpunktzahlen
- Timer Timerwerte im S5-Format
- Counter Zählerwerte im S5-Format
- BCD-Zahl Dekadenschalter / Inkrementeditor

Die Stellenzahl (Länge) ist auf die maximal darstellbare 4 Byte Zahl begrenzt. Dezimalzahlen können mit führenden Nullen ausgegeben werden. Durch Angabe von einer bestimmten Anzahl von Nachkommastellen werden Festpunktzahlen dargestellt.

Die Skalierungsfaktoren, die Bereichsgrenzen, das Meldeverhalten und die Darstellungsattribute können zusätzlich gewählt werden.

Mit diesem Editor können auf Bit, Byte, Wort und LWort-Variablen zugegriffen werden. Die maximalen Bereichsgrenzen ergeben sich aus dem Speicherabbild.

Die Typen Timer und Counter werden wie der Standardtyp dargestellt und ihre Werte werden ebenso am Bedienterminal verändert.

Der Typ BCD-Zahl hat eine Besonderheit.

Der Editor für BCD-Zahlen wird auch "Dekadenschalter" genannt.

Bei skalierten Variablen ändert sich der Wert in der Steuerung um +/- 1, jedoch der dargestellte Editierwert ist zusätzlich von der vorgegebenen Skalierung abhängig! Dies bedeutet besondere Aufmerksamkeit, falls eine Skalierung erforderlich ist, ansonsten ist der Editor zum Feinabgleich oder als Dekadenschalter mit Dezimalübertrag zu verwenden.

Mögliche Datentypen:

- positive Ganzzahlen (ohne Eingabe von Vorzeichen)
- Ganzzahlen
- positive Kommazahlen (ohne Eingabe von Vorzeichen)
- Kommazahlen

Tastenfunktionen BCD-Zahleditor

Tasten: 1 bis 9	1.) Standard Direkteingabe der Zahlenwerte 2.) Mixmode Direkteingabe der Zahlenwerte 3.) Inkrement Keine Funktion
Taste: Plus	1.) Standard Keine Funktion 2.) Mixmode und Inkrement erhöht den Wert an der Cursorposition und beeinflusst die höherwertigen Stellen bei Bereichsüberschreitung (mit dynamischer Repeatfunktion)
Taste: Minus	1.) Standard Keine Funktion 2.) Mixmode und Inkrement Verringert den Wert an der Cursorposition und beeinflusst die höherwertigen Stellen bei Bereichsunterschreitung (mit dynamischer Repeatfunktion)
Taste: Cursor links	Bewegt die Schreibmarke innerhalb der editierbaren Variable eine Stelle nach links bis maximal zur Feldgrenze.
Taste: Cursor rechts	Bewegt die Schreibmarke innerhalb der editierbaren Variable eine Stelle nach rechts bis maximal zur Feldgrenze.

3.6.4.2 Gleitkommazahleneditor

Der Gleitkommazahleneditor unterstützt LWord-Variablen die im IEEE-Format abgelegt sind. Dieser Variablentyp wird nicht von allen Steuerungen unterstützt. Sollte dennoch eine Fließkommazahl benötigt werden, dann könnte es mit der Festkommazahl im LWord-Format versucht werden. Auch mit den Skalierungsfaktoren können sehr genaue gebrochene Zahlen dargestellt werden.

Die Funktionen der Steuer- und Editiertasten ist mit denen des Dezimalzahleneditors identisch.

3.6.4.3 Hexadezimaleditor

Mit diesem Editor wird die Eingabe hexadezimaler Werte ermöglicht. Die Vorzeichentasten haben dabei eine besondere Bedeutung. Mit ihnen können die Zeichen der nicht vorhandenen alphanumerischen Tasten eingegeben werden.

Tastenfunktion Hexadezimaleditor

Tasten: 1 bis 9	Direkteingabe der Zahlenwerte
Taste: Plus	Gibt an der gewählten Stelle ASCII-Zeichen 0 bis 9, A bis F in steigender Reihenfolge aus
Taste: Minus	Gibt an der gewählten Stelle ASCII-Zeichen F bis A, 9 bis 0 in fallender Reihenfolge aus
Taste: Cursor Links	Bewegt die Schreibmarke innerhalb der editierbaren Variable eine Stelle nach links bis maximal zur Feldgrenze.
Taste: Cursor Rechts	Bewegt die Schreibmarke innerhalb der editierbaren Variable eine Stelle nach rechts bis maximal zur Feldgrenze.

3.6.4.4 Alphanumerischer Editor

Die Eingabe alphanumerischer Zeichen, mittels einer numerischen Tastatur, ist grundsätzlich möglich. Wir stellen die Eingabe mit Hilfe eines speziellen "Inkrementeditors" zur Verfügung. Er erlaubt das Editieren von Textstrings mit alphanumerischen Zeichen in Groß- und Kleinschreibung.

Tastenfunktion alphanumerischer Editor

Tasten: 1 bis 9	Direkteingabe der Zahlen 0 bis 9.
Taste: Plus	gibt an der gewählten Stelle ASCII-Zeichen 0 bis 9, A bis Z, a bis z in steigender Reihenfolge aus.
Taste: Minus	gibt an der gewählten Stelle ASCII-Zeichen z bis a, Z bis A, 9 bis 0 in fallender Reihenfolge aus.
Taste: Cursor Links	Bewegt die Schreibmarke innerhalb der editierbaren Variable eine Stelle nach links bis maximal zur Feldgrenze.
Taste: Cursor Rechts	Bewegt die Schreibmarke innerhalb der editierbaren Variable eine Stelle nach rechts bis maximal zur Feldgrenze.
Löschen	Löscht das Zeichen an der Stelle der Schreibmarke und zieht den Text von rechts nach.

Feldtyp Paßwort

Der alphanumerische Editor in Verbindung mit dem Feldtyp Paßwort dient der verdeckten Eingabe numerischer Werte für das Paßwort. Der Feldtyp Paßwort hat die gleichen Funktionen wie der normale alphanumerische Editor, jedoch werden die eingegebenen Zeichen nicht angezeigt. Die eingegebene Stelle wird durch ein "X"-Zeichen dargestellt. Die Eingabe arbeitet im Einfügemodus. Die Funktion des Feldtyps Paßwort ist an die Systemvariable **MskchgPasswd** gebunden. Um die Eingabe des Paßworts unverdeckt zu ermöglichen, muß eine alphanumerische Variable ohne Feldtyp Paßwort verwendet werden. Diese Variable muß mit der Systemvariablen **MskchgPasswd** verknüpft werden.

3.6.4.5 Auswahltexteditor (kodierter Text)

Der Auswahltexteditor bietet die Texte einer Textliste zur Auswahl an. Jeder Text einer Textliste wird einem numerischen Wert zugeordnet. Bei Eingabevariablen wird der, dem jeweiligen Text zugehörige, numerische Wert in der Steuerungsvariablen abgebildet. Entsprechend wird bei Ausgabevariablen der Text zum numerischen Wert am Bedienterminal angezeigt.

Tastenfunktion Auswahltexteditor

Taste: Plus	Anwählen in steigender Reihenfolge (nach Erreichen des letzten Wertes der Textliste erfolgt Neubeginn am Anfang der Textliste)
Taste: Minus	Anwählen in umgekehrter Reihenfolge (nach Erreichen des ersten Wertes der Textliste erfolgt Rücksprung zum Ende der Textliste)

3.6.4.6 Auswahlbildeditor (kodierte Bild)

Der Auswahlbildeditor bietet die Bilder einer Bildliste zur Auswahl an. Jedem Bild einer Bildliste wird ein numerischer Wert zugeordnet. Bei Eingabevariablen wird der dem jeweiligen Bild zugehörige numerische Wert in der Steuerungsvariablen abgebildet. Bei Ausgabevariablen wird das Bild am Bedienterminal angezeigt, das dem numerischen Wert in der Bildliste zugeordnet ist.

Tastenfunktion Editor für kodierte Bild

Taste: Plus	anwählen in steigender Reihenfolge
Taste: Minus	anwählen in umgekehrter Reihenfolge

3.6.4.7 Tabelleneditor

Begriffsdefinitionen:

Zeilenzahl:	Anzahl der auf der Anzeige dargestellten Zeilen der Tabelle.
Anzahl der Elemente:	Anzahl der Variablen, die für jede Spalte in der SPS abgelegt sind.
Tabellenoffset:	Offset der in der ersten Zeile dargestellten Variablen relativ zum Tabellenanfang. Beim Neuaufbau einer Maske wird der Tabellenoffset immer auf 0 gesetzt. Nach dem ersten seitenweisen Blättern ist er zum Beispiel gleich der Zeilenzahl. Der Tabellenoffset ist immer für alle Spalten gleich.
Einzelvariable:	Jede in einer Maske außerhalb der Tabelle stehende Variable.
Spaltenvariable:	Jede Tabellenspalte ist als Ganzes eine Spaltenvariable.
Maskenvariable:	Spalten- oder Einzelvariable.
Zeilenvariable:	Jede Zeile innerhalb einer Tabellenspalte enthält eine Zeilenvariable.
Letzte Zeilenvariable:	Zeilenvariable in der untersten Tabellenzeile bei maximalem Tabellenoffset.
Erste Zeilenvariable:	Zeilenvariable in der obersten Tabellenzeile bei Tabellenoffset = 0.

Funktionsbeschreibung:

- Jede Zeilenvariable wird, unmittelbar bevor sie zum Editieren aktiviert wird, nochmals aus der SPS gelesen.
- Eingabe- und zyklische Ausgabevariablen werden auch in der Tabelle ständig aktualisiert; mit Ausnahme von der Zeilenvariablen, die gerade editiert wird. Einmalige Ausgabevariablen werden bei jedem Maskenaufbau (also zum Beispiel auch nach dem Loslassen der Hilfetaste) und immer, wenn sich der Tabellenoffset geändert hat, aktualisiert.
- Bewegung zwischen den Variablen erfolgt mit Hilfe der Cursortasten und Softkeys, die mit Systemvariablen verknüpft sind. Die Reihenfolge beim Wechsel zwischen Maskenvariablen entspricht der Reihenfolge beim Anlegen der Variablen in der Maske (Einzelvariablen) und im Tabellenfeld. Die folgende Tabelle zeigt die Systemvariablen und deren Funktion, die für die Steuerung in Tabellenfeldern relevant sind.

Systemvariable	Funktion im Tabellenfeld	Anlageort
TabPgDn	Blättern vor (PgDn)	Funktionstaste / Softkeytaste
TabPgUp	Blättern zurück (PgUp)	Funktionstaste / Softkeytaste
TabLeft	Sprung zur nächsten linken Spalte	Funktionstaste / Softkeytaste
	Bei äußerst linker Spalte: Sprung zur vorherigen Maskenvariablen	
TabRight	Sprung zur nächsten rechten Spalte	Funktionstaste / Softkeytaste
	Bei äußerst rechter Spalte: Sprung zur nächsten Maskenvariablen	
VarTablenR0	Zeilennummer beginnend bei 0 anzeigen	Im Tabellenfeld am linken Rand
VarTablenR1	Zeilennummer beginnend bei 1 anzeigen	Im Tabellenfeld am linken Rand

- Die Variablen in einem Tabellenfeld können nach erfolgter Datenfreigabe (Zustands-LED der Taste Datenfreigabe leuchtet) durch die Tasten *Cursor Oben*, *Cursor Unten*, *Datenübernahme* (Enter), *Blättern zurück* (PgUp) und *Blättern vor* (PgDn) durchgesehen und geändert werden. Ohne Datenfreigabe (Zustands-LED der Taste Datenfreigabe leuchtet nicht) können die Variablen nur durch die Tasten *Blättern zurück* und *Blättern vor* gesichtet, aber nicht geändert werden.
- Alle Variablen einer Tabellenspalte müssen mit einem Kommunikationsauftrag gelesen werden können. Dies kann bei Verwendung von 4 Byte Variablen zu einer Beschränkung der Zeilenzahl führen.

Tastenfunktion Tabelleneditor

Taste: Cursor Oben	<p>Wechselt auf die vorhergehende Maskenvariable, wenn der Cursor auf einer Einzelvariablen oder auf der ersten Zeilenvariablen steht.</p> <p>Wechselt auf die vorhergehende Zeilenvariable, wenn der Cursor auf einer anderen als der ersten Zeilenvariablen steht. Dabei wird nur dann zeilenweise gescrollt, wenn der Cursor schon auf der obersten Zeilenvariablen steht.</p> <p>Beim Wechsel auf eine neue Spaltenvariable wird der Cursor auf die unterste Zeilenvariable (also letzte Zeilenvariable) gesetzt.</p> <p>Beim Wechsel auf eine Einzelvariable wird die Zeilennummer nicht verändert.</p>
Taste: Cursor Unten	<p>Wechselt auf die nächste Maskenvariable, wenn der Cursor auf einer Einzelvariablen oder auf einer letzten Zeilenvariablen steht.</p> <p>Wechselt auf die nächste Zeilenvariable, wenn der Cursor auf einer anderen als der letzten Zeilenvariablen steht. Dabei wird nur dann zeilenweise gescrollt, wenn der Cursor schon auf der untersten (nicht letzten) Zeilenvariablen steht.</p> <p>Beim Wechsel auf eine neue Spaltenvariable wird die Zeilennummer auf 0 und der Cursor auf die oberste Zeilenvariable (also erste Zeilenvariable) gesetzt.</p> <p>Beim Wechsel auf eine Einzelvariable wird die Zeilennummer nicht verändert.</p>
Softkey: TabLeft	<p>Wechselt in die nächste linke Spalte in der gleichen Zeile. Wechselt auf die vorhergehende Maskenvariable, wenn sich der Cursor in der äußerst linken Spalte befindet.</p>
Softkey: TabRight	<p>Wechselt in die nächste rechte Spalte in der gleichen Zeile. Wechselt auf die nächste Maskenvariable, wenn sich der Cursor in der äußerst rechten Spalte befindet.</p>
Taste: Blättern zurück	<p>Verringert den Tabellenoffset um die Zeilenzahl. Wenn dies möglich ist, bleibt die Zeilennummer unverändert, ansonsten wird die Zeilennummer um die noch fehlende Anzahl von Zeilen vermindert. Wenn die Zeilennummer ebenfalls den Minimalwert erreicht, hat die Taste keine Auswirkung mehr. Insbesondere ist kein Wechsel in eine andere Maskenvariable möglich.</p>
Taste: Blättern vor	<p>Erhöht den Tabellenoffset um die Zeilenzahl. Wenn dies möglich ist, bleibt die Zeilennummer unverändert, ansonsten wird die Zeilennummer um die noch fehlende Anzahl von Zeilen erhöht. Wenn die Zeilennummer ebenfalls den Maximalwert erreicht, hat die Taste keine Auswirkung mehr. Insbesondere ist kein Wechsel in eine andere Maskenvariable möglich.</p>
Taste: Dateneingabe	<p>Funktion wie Cursor Unten, jedoch wird die Variable zusätzlich noch in die SPS geschrieben.</p>

3.6.5 Externe Datenfreigabe

Die externe Datenfreigabe beschränkt die Eingabemöglichkeit von Variablenwerten durch die Steuerung. Um die Datenfreigabe von der Steuerung aus beeinflussen zu können, muß:

- eine Variable für das Koordinierungsbyte Lesen angelegt sein
- eine Variable für den Pollbereich angelegt sein.

Durch Betätigen der Taste Datenfreigabe in einer E/A-Maske wechselt der Zustand des Terminals in den Editiermode. Unter der Voraussetzung, daß die externe Datenfreigabe von der Steuerung vorliegt, leuchtet die Zustands-LED der Taste Datenfreigabe. Die externe Datenfreigabe wird durch das Bit 0 im ersten Byte des zyklischen Pollbereichs der Steuerung ermöglicht (siehe Kapitel Pollbereich). Sobald die SPS in dieses Bit den Wert 0 einträgt, wird das Editieren von Daten verhindert. Dieser Zustand wird gekennzeichnet, indem die Zustands-LED der Taste Datenfreigabe blinkt.

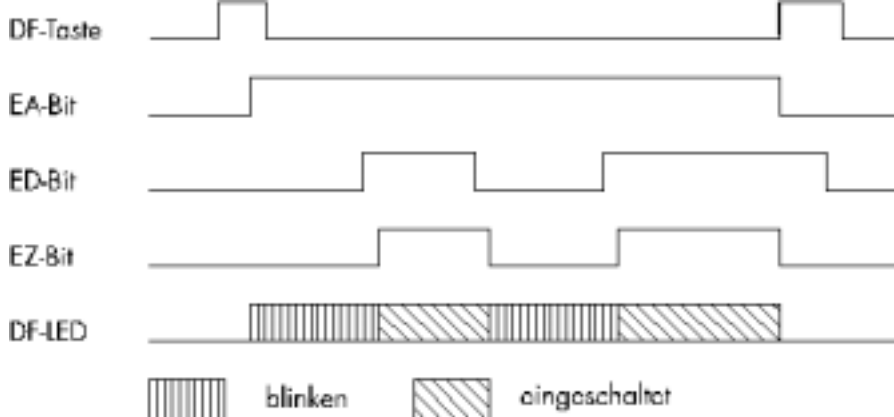


Bild 23: Impulsdiagramm externe Datenfreigabe

- DF-Taste Taste Datenfreigabe
- EA-Bit Editieranforderungsbit im Koordinierungsbyte Lesen
- ED-Bit Externe Datenfreigabe im Koordinierungsbyte Schreiben
- EZ-Bit Editierzustandsbit im Koordinierungsbyte Lesen
- DF-LED Zustands-LED der Taste Datenfreigabe

Eine blinkende Zustands-LED der Taste Datenfreigabe bedeutet immer eine fehlende externe Datenfreigabe. Hat die SPS den Wert 1 in Bit 0 eingetragen, dann ist das Editieren erlaubt, die Zustands-LED der Taste Datenfreigabe leuchtet. Die Schreibmarke blinkt auf dem ersten Eingabefeld der Maske. In Abhängigkeit der gewählten Variablen wird der vom Anwender spezifizierte Editor aufgerufen.

Die externe Datenfreigabe wird wie folgt behandelt:

- Die SPS schaltet die externe Datenfreigabe im Bit ED des Koordinierungsbytes Schreiben.
- Wird die Datenfreigabetaste gedrückt, wird das Bit EA im Koordinierungsbyte Lesen gesetzt und die Zustands-LED der Taste Datenfreigabe blinkt.
- Ca. 100 ms später wird das Koordinierungsbyte Schreiben vom Terminal ausgelesen. Ist die externe Datenfreigabe vorhanden (ED im Koordinierungsbyte Schreiben gesetzt), kann editiert werden, die Zustands-LED der Taste Datenfreigabe leuchtet und das Bit EZ im Koordinierungsbyte Lesen wird gesetzt. Ist die externe Datenfreigabe nicht vorhanden (ED im Koordinierungsbyte Schreiben nicht gesetzt), kann nicht editiert werden und die Zustands-LED der Taste Datenfreigabe blinkt weiter.
- Während des Editierens kann die SPS die Dateneingabe jederzeit über das Bit ED sperren. Immer wenn vom Terminal erkannt wird, daß das Bit ED zurückgesetzt wird, ist die Dateneingabe gesperrt, die Zustands-LED der Taste Datenfreigabe blinkt und das Bit EZ wird ebenfalls zurückgesetzt. Immer wenn vom Terminal erkannt wird, daß das Bit ED gesetzt wird, ist die Dateneingabe freigegeben, die Zustands-LED der Taste Datenfreigabe leuchtet und das Bit EZ wird ebenfalls gesetzt.

- Wird der Editor durch nochmaliges Betätigen der Taste Datenfreigabe beendet, wird der zuletzt editierte Wert bestätigt, das heißt an die Steuerung übertragen, und anschließend die Bits EA und EZ im Koordinierungsbyte Lesen gelöscht.

3.6.6 SPS-Handshake

Der Handshake wird wie folgt behandelt:

- Wird vom Terminal auf eine Variable geschrieben, in deren Beschreibung die Handshakekennung enthalten ist, so wird im Koordinierungsbyte Lesen das Bit RA (Refreshanforderung) gesetzt und die Dateneingabe gesperrt.
- Von der SPS kann nun zum Beispiel ein neuer Datensatz für andere Eingabevariablen bereitgestellt werden. Danach wird von der SPS das Bit RQ (Refreshquittierung) im Koordinierungsbyte Schreiben gesetzt.
- Wenn das Terminal das Setzen des Bits RQ erkannt hat, liest es zunächst alle in der gerade angewählten Maske enthaltenen Eingabevariablen nochmals neu aus der SPS aus, gibt anschließend die Dateneingabe wieder frei und setzt das Bit RA wieder zurück.
- Wenn das Bit RQ bereits beim Schreiben auf die Variable, deren Beschreibung die Handshakekennung enthält, gesetzt ist, erfolgt sofort eine Neuausgabe der anderen Eingabevariablen und die Dateneingabe wird nicht gesperrt.

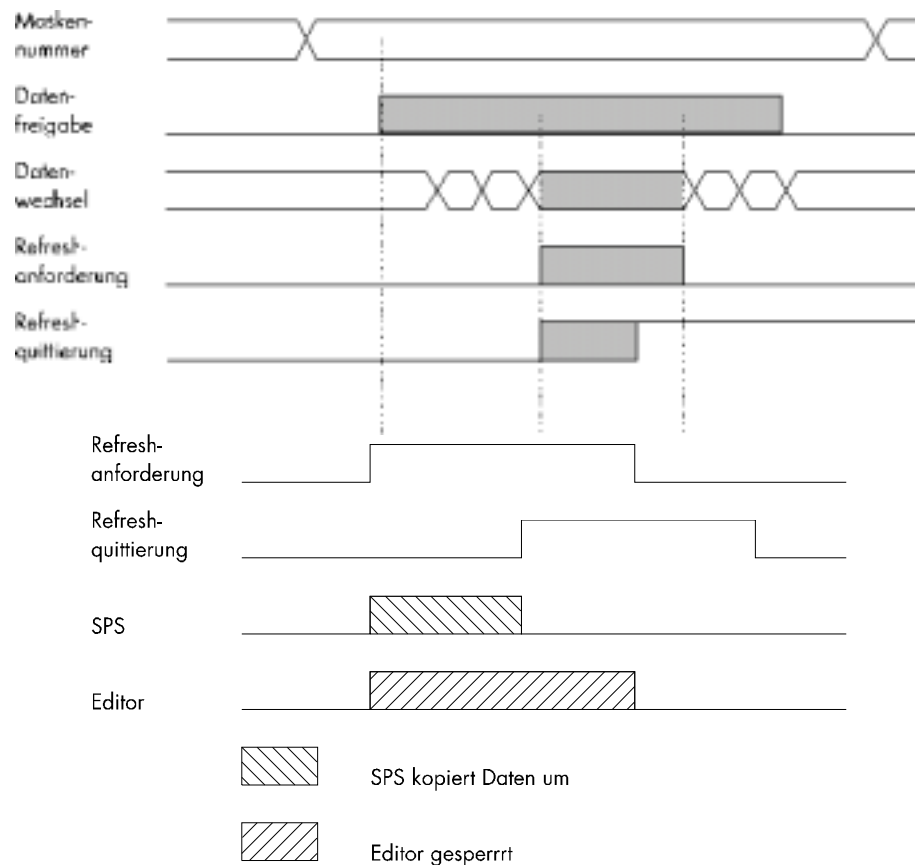


Bild 24: Impulsdiagramm SPS-Handshake

3.6.7 Auffrischen einmaliger Ausgabedaten

Ändern sich in einer Maske einmalige Ausgabe- oder Eingabevariablen, so können durch die erneute Ausgabe der Maske (externe Maskenanwahl) die Variableninhalte aufgefrischt werden.

3.6.8 Änderung der Daten

Wie die Änderung der Daten erfolgt, das heißt unter welchen Bedingungen das Bedienterminal die neuen Daten an die Steuerung überträgt, legt der Anwender in der Variablenbeschreibung fest. Hier wird unterschieden, zu welchem Zeitpunkt die Übertragung erfolgt.

Wahlweise:

- mit der Taste Dateneingabe (Standard)
- mit den +/- Tasten oder der Taste Dateneingabe (Inkrement-Editor)
- immer melden (jeder Zwischenstand wird übertragen)

Beim Inkrement-Editor wird jeder Zwischenwert übertragen.

Die Möglichkeit **immer melden** ist beschränkt auf Werte, die innerhalb der definierten Grenzen bleiben. Wird ein Wert eingegeben, der außerhalb der Unter- oder Obergrenze liegt, wird dem Bediener eine entsprechende Systemmeldung angezeigt.

Bei Eingabe eines mehrstelligen Wertes, der

- die Obergrenze überschreitet,
wird die zuletzt eingegebene Stelle nicht an die Steuerung übertragen.
- die Untergrenze unterschreitet,
wird der vorher gültige Wert wieder eingesetzt.
- die Untergrenze unterschreitet (bis zur Eingabe der letzten Stelle) und dann die Obergrenze überschreitet (nach Eingabe der letzten Stelle)
wird der vorher gültige Wert wieder eingesetzt.

3.6.8.1 Plausibilitätskontrolle der Eingabe

Die Plausibilitätskontrolle wird bei allen editierbaren numerischen Variablen durchgeführt. Dazu wird in der Variablenbeschreibung die Obergrenze und Untergrenze vorgegeben. Die Bereichsgrenzen werden im passenden Ausgabeformat dargestellt. Bei Fehleingaben werden Systemmeldungen abgesetzt. Für weitere Informationen siehe Kapitel Editoren und Systemmeldungen.

3.7 Grafik

3.7.1 Grafische Objekte (TSdos)

Ein Bild oder andere grafische Informationen werden im Terminal in Form von grafischen Objekten benutzt. Als Basis für ein grafisches Objekt dient ein Quellbild, das im PCX-Dateiformat vorliegt.

Die Bilder oder Grafiken können mit Standard-Entwicklungswerkzeugen, zum Beispiel unter Windows (1) erstellt werden. Bereits vorhandene Grafikprogramme lassen sich problemlos nutzen. Durch die Bilderstellung mit Standardwerkzeugen entfällt die zusätzliche Einarbeitung des Anwenders. Die einzubindende Bitmap (Grafik) wird als monochrome PCX-Datei unter der TS-Programmiersoftware verwaltet. Die verwendeten Bitmaps können dabei Ausschnittsweise definiert sein.

Die angegebenen Bitmaps werden von der Programmiersoftware als "Grafische Objekte" in einer Bibliothek verwaltet. Jedes grafische Objekt erhält vom Anwender einen eindeutigen symbolischen Namen. Unter diesem Namen können die Objekte in die Masken oder Bildlisten aufgenommen werden.

Die grafischen Objekte sind wiederverwendbar und können mittels einer Bibliotheksliste ausgewählt werden. Sie stehen somit dem Anwender nicht nur in verschiedenen Masken sondern auch bei verschiedenen Projekten automatisch zur Verfügung. Die Projektunabhängigkeit bietet den Vorteil, daß einmal definierte grafische Objekte im System immer wieder zur Verfügung stehen. Die Größe einer Bitmap kann minimal Font-Größe (zum Beispiel 6 x 8, 6 x 9) und maximal Displaygröße (zum Beispiel 256 x 128, 240 x 128, 256 x 64) annehmen. Sie ist immer ein ganzzahliges Vielfaches eines Zeichens (Charakterkoordinaten).

Grafische Objekte werden generell "sprachunabhängig" verwaltet. Das hat den Vorteil, daß in allen Sprachen die gleichen grafischen Objekte zur Verfügung stehen.

Das grafische Objekt entspricht einem Bildausschnitt aus dem Quellbild. Der Bildausschnitt wird durch das linke, obere Eck innerhalb des Quellbildes und die Höhe und Breite (Einheit Charakter) beschrieben. Über einen frei definierbaren, aus 8 Zeichen bestehenden, symbolischen Namen, kann das grafische Objekt überall im Projektiersystem angesprochen werden.

Bei der Übersetzung in eine ladbare Anwenderbeschreibung, wird die PCX-Datei vom Assembler auf das jeweilige Display und den Displaycontroller optimal abgestimmt und komprimiert. Dadurch wird die Ausgabezeit und der Speicherbedarf für die Grafik optimiert. Zur Reduzierung der Rechenzeit wird die Bitmap-Objektdatei nur dann erzeugt, wenn die PCX-Datei ein neueres Datum erhalten hat, oder die Terminalkennung in der Bitmap-Objektdatei ungleich dem Terminaltyp ist, für den der Assembler momentan übersetzt.

3.7.2 Bilder (TSwin)

Die Projektiersoftware TSwIn kann jedes vom Anwender installierte Programm als Server für Bilder nutzen.

Prinzipiell ist es möglich Bilder auf 2 Arten für TSwIn nutzbar zu machen:

- 1.) Es können bereits erstellte Bilder eines beliebigen Programms ausgewählt werden. Ein Duplikat vom Quellbild wird in der Projektdatenbank von TSwIn gespeichert und in eine Pixelgrafik mit den angegebenen Ausmaßen umgerechnet.
- 2.) Es kann ein beliebiges Programm als Server für die Bilderstellung aus TSwIn heraus gestartet werden. Nach Fertigstellung des Bildes wird der Server beendet und das Bild in der Projektdatenbank von TSwIn gespeichert. Für die Darstellung in TSwIn und am Bedienterminal wird das Bild in eine Pixelgrafik mit den angegebenen Ausmaßen umgerechnet.

Ein Doppelklick mit der Maus auf ein Bild in TSwIn öffnet automatisch den ursprünglichen Server, um das Bild bearbeiten zu können. Die Gestaltungsmöglichkeiten der Bilder richten sich nur nach den Möglichkeiten des Serverprogramms. In TSwIn können Bilder, je nach Bedienterminaltyp, Pixel- oder Rasterorientiert plaziert werden.

3.7.3 Grafik in Bedienterminals

Die Einbindung von Grafik in die Anwenderbeschreibung erfolgt durch statische Hintergrundbilder zu jeder E/A-Maske, dynamische Variablendarstellung als Balkendiagramm, als Punktelinie einer Kurve oder in Form von Auswahlbildern (Symbolen). Durch diese Basiselemente kann der Anwender eine vollgrafische Gestaltung seiner Bedienoberfläche realisieren. Die zyklische Aktualisierung der Kurvenverläufe, Balkendiagramme und Auswahlbilder ermöglicht auch eine einfache Animation.

3.7.3.1 Hintergrundbilder

Mit Hilfe von grafischen Objekten können im Terminal Hintergrundbilder zu Masken anlegt werden. Pro Maske können bis zu 10 Grafische Objekte als Hintergrundbilder angegeben werden. Hintergrundbilder werden über den Namen des grafischen Objekts und die Position innerhalb der Maske beschrieben. Für jedes Hintergrundbild sind Anzeigeattribute wählbar. Zur Wahl stehen die Attribute: normal, invers oder blinkend. Die dargestellte Größe wird von der Definition des grafischen Objektes bestimmt.

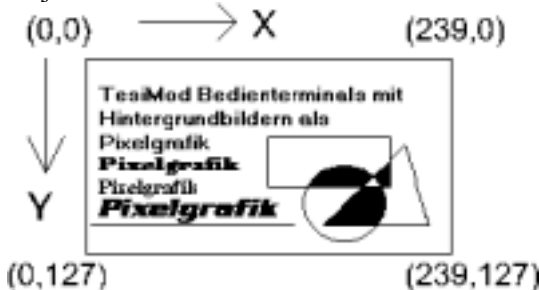


Bild 25: Koordinatenbeispiel

Hintergrundbilder werden nacheinander auf das Display ausgegeben. Die Reihenfolge wird im Projektiersystem durch die Reihenfolge der Eingabe bestimmt. Die Bilder werden im angewählten Mode in den Anzeigespeicher kopiert. Bitmaps für Hintergrundbilder können maximal die Auflösung und Ausdehnung der eingesetzten Displays erreichen. Die Modularisierung in Teilbilder hat den Vorteil, daß sie im System mehrfach genutzt werden können, wodurch in erster Linie eine Speicheroptimierung erreicht wird. Hintergrundbilder bieten sich für alle statischen Anwendungen an, wie:

- feste Raumaufteilung des Displays
- symbolische Darstellung der Softkeyleiste
- Koordinatensysteme
- Infos über Fertigungsabläufe
- Statische Bilder oder Symbole
- Übersichtsbilder der Anlagen und Prozesse

Wurden mehrere Hintergrundbilder für eine Maske definiert, können sich diese auch überlagern. Dabei kann der Überschreibmodus definiert werden.

3.8 Rezepturen

In einer Rezeptur können verschiedene logisch zusammengehörende Variablen zu einer Einheit zusammengefaßt werden. Im Unterschied zu Maskenvariablen werden Rezepturvariablen nicht sofort nach der Eingabe in die Steuerung übertragen, sondern in Form von Datensätzen netzausfallsicher im Terminal abgelegt. Diese Datensätze werden dann bei Bedarf als Ganzes in die Steuerung geladen.

Bei der Projektierung können maximal 250 Rezepturen erstellt werden. Für jede dieser Rezepturen können dann bis zu 250 Datensätze angelegt werden. Die Datensätze werden entweder während der Projektierung erstellt und zusammen mit der Anwenderbeschreibung im Flash-Speicher des Terminals abgelegt oder sie werden während des Betriebs am Terminal eingegeben und sind dann im batteriegepufferten RAM abgelegt. Im Flash-Speicher abgelegte Datensätze können nicht direkt geändert werden, für Änderungen müssen die Datensätze ins RAM kopiert werden. Geänderte Datensätze verbleiben im batteriegepufferten RAM.

Beispiel für den Einsatz von Rezepturen :

Maschineneinstellung für die Fertigung von verschiedenen Artikeln

	Variable	Wert	Einheit
Artikel <i>Bügel</i> benötigt :	Material	ST37-3	
	Vorschub	25,00	mm/s
	Sollwert Achse 1	43,50	mm
	Sollwert Achse 2	56,30	mm
	Schnittwinkel	30	°
	Schnittgeschwindigkeit	110	mm/s
Artikel <i>Welle</i> benötigt :	Material	X20Cr13	
	Vorschub	20,00	mm/
	Sollwert Achse 1	45,60	mm
	Sollwert Achse 2	51,20	mm
	Schnittwinkel	45	°
	Schnittgeschwindigkeit	76	mm/s

Die Variablen Material, Vorschub, Sollwert Achse 1, Sollwert Achse 2, Schnittwinkel und Schnittgeschwindigkeit können zu einer Rezeptur "Maschineneinstellung für Artikel" zusammengefaßt werden. Die Variablen Vorschub, Sollwert Achse 1 und Sollwert Achse 2 werden als Gleitkomma- oder Festpunktzahlen definiert. Die Variable Schnittwinkel als Ganzzahl und die Variable Material als Auswahltext (kodierter Text).

Die Werte für die Fertigung der Artikel *Bügel* und *Welle* müssen als Datensatz gespeichert werden. Je nach Artikel wird der jeweilige Datensatz zum Zeitpunkt der Fertigungsumstellung in die Steuerung geladen.

Die folgende Checkliste enthält alle Elemente, die für die Erstellung und Bedienung einer Rezeptur mit Datensätzen erforderlich und hilfreich sind:

- Die Rezeptur selbst (Texte und Variablen)
- Datensätze mit Datensatznummer, Datensatzname und Variablenoffset
- E/A-Maske für die Rezeptur
- Rezepturfeld in der Maske
- Rezepturpuffer (Adresse für Datenbereich in der Steuerung)
- Variable Datensatznummer für Übertragung vom Terminal
- Variable Rezepturnummer für Übertragung vom Terminal
- Variable Datensatznummer für Anforderung von Steuerung
- Variable Rezepturnummer für Anforderung von Steuerung

- Systemvariablen:

Variablenname	Verknüpfungspartner	Beschreibung
SelectDSNr	Auswahltext/Dezimalzahl	Datensatznummer anzeigen/auswählen
SelectDSName	Auswahltextvariable	Datensatznamen anzeigen/auswählen
DestDSNr	positive Dezimalzahl	Zieldatensatznummer für Kopiervorgang
DSCopy	Softkeytaste/Auswahltextvariable	Datensatz kopieren aktivieren
DSDelete	Softkeytaste/Auswahltextvariable	Datensatz löschen
DSDownload	Softkeytaste/Auswahltextvariable	Datensatz in Steuerung laden
ActDSName	Alphanumerische Variable	Namen für RAM-Datensatz eingeben
SelectRezeptNr	Auswahltext/Dezimalzahl	Rezeptnummer anzeigen/auswählen
TabPgUp	Softkeytaste	Blättern zurück
TabPgDn	Softkeytaste	Blättern vor
Break	Softkeytaste	Eingabe abbrechen

3.8.1 Aufbau einer Rezeptur

Eine Rezeptur umfaßt maximal 255 Variablen. Zusätzlich können bis zu 255 erläuternde Texte projektiert werden. Die Variablen und Texte können auf bis zu 255 Zeilen (jeweils über die gesamte Anzeigenbreite) verteilt werden. Zu jeder Variablen kann ein Hilfetext programmiert werden.

Die Rezeptur wird innerhalb einer E/A-Maske in einem Rezepturfeld dargestellt, das sich über die gesamte Anzeigenbreite erstreckt. Die Höhe des Rezepturfelds ist einstellbar von einer Zeile bis zur gesamten Anzeigenhöhe. Innerhalb des Rezepturfelds kann eine umfangreiche Rezeptur mit Hilfe der Tasten Blättern vor / zurück und den Cursortasten durchgeblättert werden.

Für die Rezepturvariablen stehen sämtliche einzeiligen Formate und Editoren zur Verfügung, die auch in E/A-Masken verwendet werden können. Die Verwendung von mehrzeiligen Formaten ist nicht möglich (zum Beispiel mehrzeiliges Auswahlfeld, Tabelle etc). Ebenso kann weder bei Variablen noch bei Texten die Zoomdarstellung verwendet werden.

3.8.2 Rezeptur- und Datensatzbearbeitung

Die meisten der im Folgenden beschriebenen Operationen beziehen sich auf den aktiven Datensatz. Um einen Datensatz zu aktivieren, muß zunächst die Rezeptur, zu der er gehört, und anschließend der Datensatz selbst, angewählt werden. Wie diese Anwahlen erfolgen können, wird in den nächsten beiden Abschnitten erläutert.

3.8.2.1 Rezeptur anwählen

Rezepturen werden Nummern im Bereich 1 - 250 zugeordnet, die bei der Programmierung vergeben werden.

Eine Rezeptur kann auf folgende Weisen angewählt werden:

- Durch eine feste Kopplung einer Rezeptur mit einer Maske. Das heißt, immer wenn die entsprechende Maske aufgerufen wird, wird in dem darin enthaltenen Rezepturfenster die bei der Maskenprojektierung angegebene Rezeptur angezeigt. Wird bei der Projektierung einer Maske mit Rezepturfenster keine Rezeptur fest zugeordnet, dann wird beim Aufruf der Maske die Rezeptur angezeigt, die als letztes bearbeitet wurde.
- Über die Systemvariable **SelectRezeptNr**. Die Systemvariable kann prinzipiell mit beliebigen Editoren bearbeitet werden. Es empfiehlt sich jedoch einen Auswahltext (kodierten Text) oder ein Auswahlfeld (nur TSDos) zu verwenden und den Rezepturnummern sinnvolle Rezepturnamen zuzuordnen.

3.8.2.2 Datensatz anwählen

Datensätze können sowohl Nummern im Bereich 1 - 250, als auch Namen zugeordnet werden.

Die Datensatznummern und -namen werden bei der Erstellung eines Datensatzes vergeben, also entweder bei der Projektierung für die im Flash-EPROM abgelegten Datensätze oder am Terminal für die im RAM abgelegten Datensätze. Ein Datensatzname kann maximal 15 Zeichen lang sein. Die Datensatznamen müssen nicht (sollten jedoch) eindeutig sein.

Ein Datensatz kann auf folgende Weisen angewählt werden:

- Anwahl einer neuen Rezeptur. Mit einer neuen Rezeptur wird automatisch auch der zugehörige Datensatz mit der niedrigsten Nummer angewählt.
- Über die Systemvariable **SelectDSNr**. Die Systemvariable kann nur als Auswahltext (kodierter Text) oder Auswahlfeld (nur TSdos) editiert werden. Hierbei werden dann jeweils nur die Nummern der für die aktive Rezeptur vorhandenen Datensätze angezeigt.
- Über die Systemvariable **SelectDSName**. Die Systemvariable kann nur als Auswahltext (kodierter Text) oder Auswahlfeld (nur TSdos) editiert werden. Hierbei werden dann jeweils nur die Namen der für die aktive Rezeptur vorhandenen Datensätze angezeigt.

3.8.2.3 Datensatz kopieren

Nur der aktive Datensatz kann kopiert werden. Hierzu wird auf die Systemvariable **DestDSNr** die Nummer des Zieldatensatzes und anschließend der Wert 1 auf die Systemvariable **DSCopy** geschrieben.

Voraussetzungen für ein erfolgreiches Kopieren sind:

- Die Nummer des Zieldatensatzes muß im Bereich zwischen 1 und 250 liegen. (DSCopy mit Wert 2 sucht freien Datensatz, DSCopy mit Wert 3 überschreibt vorhandenen Datensatz)
- Es darf für die aktive Rezeptur kein Datensatz mit derselben Nummer bereits existieren (außer DSCopy wird mit Wert 3 verwendet)
- Der aktive Datensatz darf nicht gerade editiert werden.
- Es muß noch genügend RAM-Speicher im Terminal vorhanden sein.

Falls eine dieser Voraussetzungen nicht erfüllt ist, wird der Kopiervorgang nicht durchgeführt und es erfolgt eine entsprechende Systemmeldung.

Nach dem Kopieren wird der Zieldatensatz zum aktiven Datensatz.

Der Name des Zieldatensatzes enthält nach dem Kopieren nur Leerzeichen. Er kann mit Hilfe der Systemvariablen **ActDSName** geändert werden.

3.8.2.4 Datensatz löschen

Nur der aktive Datensatz kann gelöscht werden. Hierzu wird auf die Systemvariable **DSDelete** der Wert 1 geschrieben.

Voraussetzungen für ein erfolgreiches Löschen sind:

- Der aktive Datensatz darf nicht gerade editiert werden.
- Der Datensatz muß im RAM-Speicher abgelegt sein.

Falls eine dieser Voraussetzungen nicht erfüllt ist, wird der Löschvorgang nicht durchgeführt und es erfolgt eine entsprechende Systemmeldung.

Nach dem Löschen wird der Datensatz mit der niedrigsten Nummer, der zur aktuellen Rezeptur gehört, zum aktiven Datensatz.

3.8.2.5 Datensatz ändern

Der aktive Datensatz kann, sofern er im RAM abgelegt ist, geändert werden.

Die Änderung des Inhalts erfolgt durch Editieren der Variablen im Rezepturfenster. Allerdings werden die neu eingegebenen Werte nicht sofort nach Betätigen der Taste Dateneingabe (Enter) in den Datensatz geschrieben, sondern zunächst in einem temporären Puffer gehalten. Erst nach Betätigen der Taste Datenfreigabe werden die Werte in den Datensatz übernommen. Falls diese Übernahme nicht erfolgen soll, kann durch Schreiben des Wertes 1 auf die Systemvariable **Break** der Pufferinhalt auch verworfen werden. Die Systemvariable **Break** wird sinnvollerweise über einen entsprechend programmierten Softkey angesteuert.

Bevor der Pufferinhalt nicht übernommen oder verworfen wurde, kann kein anderer Datensatz angewählt werden.

Wenn während der Datensatzänderung ein Maskenwechsel durch die Steuerung erfolgt oder die externe Datenfreigabe weggenommen und anschließend die Datenfreigabetaste betätigt wird, so wird der Pufferinhalt ebenfalls verworfen.

Der geänderte Datensatz wird nicht automatisch in die Steuerung übertragen. Dies muß explizit durch den Bediener oder die Steuerung ausgelöst werden.

3.8.3 Datensatztransfer an / von Steuerung

3.8.3.1 Transfer zur Steuerung

Ein Datensatztransfer zur Steuerung kann auf folgende Arten ausgelöst werden:

- Durch Schreiben des Wertes 1 auf die Systemvariable **DSDownload** kann der aktive Datensatz in die Steuerung übertragen werden.
- Anforderung eines beliebigen Datensatzes durch die Steuerung. Hierzu schreibt die Steuerung auf die Adressen der Variablen für die Rezeptur- und Datensatznummer die gewünschten Werte. Anschließend schreibt die Steuerung den Wert **7FFBH** auf die Adresse des seriellen Meldekanals (siehe Kapitel 3.22 *Der zyklische Pollbereich*).

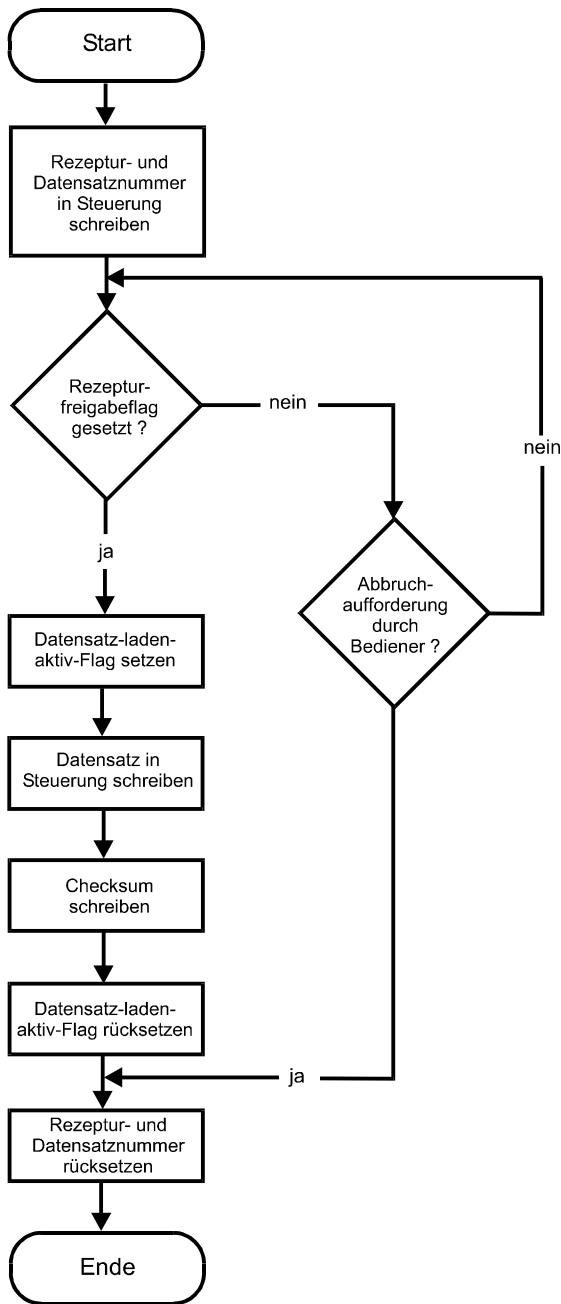


Bild 26: Ablauf des Datensatztransfers zur Steuerung

Der Transfer eines beliebigen Datensatzes, ausgelöst durch das Terminal, soll im folgenden nochmal chronologisch erklärt werden.

Der Vorgang wird aus der Sicht des Terminals betrachtet:

1. Schreiben der Rezepturnummer auf die vereinbarte Variablenadressen für Rezepturnummern.
2. Schreiben der Datensatznummer auf die vereinbarte Variablenadresse für Datensatznummern.
3. Warten, bis Datensatztransfer von der Steuerung freigegeben wird. Dazu muß die Steuerung das DDF-Bit im Koordinierungsbyte Schreiben auf logisch 1 setzen.
Während dieser Zeit kann der Transfer durch Schreiben des Wertes 1 auf die Systemvariable **DSDnloadBreak** abgebrochen werden. Ist die Freigabe durch die Steuerung einmal erteilt worden, kann sie während des Transfers nicht mehr durch Rücksetzen des DDF-Bit im Koordinierungsbyte Schreiben entzogen werden.
4. Das Bit Datensatz-Download-aktiv im Koordinierungsbyte Lesen setzen.
Dieses Bit kann von der Steuerung dazu benutzt werden, einen unvollständigen Datensatztransfer (Kommunikationsunterbrechung, Stromausfall) zu erkennen. Ein unvollständiger Transfer wird allerdings nach dem erneuten Hochlaufen des Terminals wiederholt.
5. Datensatz in den vereinbarten Rezepturpuffer in der Steuerung schreiben. Ein Rezepturpuffer kann für jede Rezeptur getrennt programmiert werden. Der Rezepturpuffer muß um mindestens ein Byte größer als der Datensatz sein (wegen Checksumme) und außerdem muß er eine gerade Anzahl von Bytes groß sein (also gegebenenfalls nochmal ein Byte größer).
6. Checksumme schreiben. Auf die dem letzten Byte des Datensatzes folgende Adresse wird die byteweise XOR-Checksumme (vom gesamten Datensatz) geschrieben. Diese kann von der Steuerung zur zusätzlichen Absicherung gegen die Verwendung von unvollständig übertragenen Datensätzen benutzt werden.
7. Das Bit Datensatz-Download-aktiv im Koordinierungsbyte Lesen zurücksetzen.
8. Rücksetzen der Rezeptur- und Datensatznummer in die für diesen Zweck in der Steuerung projektierten Variablen. Die Rezepturnummer wird auf 0 und die Datensatznummer auf 255 gesetzt.

Mit Hilfe der Systemvariablen **DSDnloadState** können einzelne Schritte des Ablaufs vom Bediener überwacht werden.

Für jede Rezeptur wird im Terminal der Namen des zuletzt in die Steuerung geladenen Datensatzes gespeichert. Für die aktive Rezeptur kann dieser Name über die Systemvariable **LoadDSName** angezeigt werden. Wenn zu einer Rezeptur noch kein Datensatz in die Steuerung geladen wurde oder wenn der in die Steuerung geladene Datensatz bereits gelöscht wurde, erscheint ein Fragezeichen.

3.8.3.2 Transfer von der Steuerung

Bei manchen Anwendungen kann es erforderlich werden, daß einzelne Datensatzwerte in der Steuerung verändert werden (zum Beispiel Teachen) und der so abgeänderte Datensatz wieder im Terminal abgelegt wird. Deshalb besteht auch die Möglichkeit Datensätze von der Steuerung an das Terminal zu transferieren.

Ein Datensatztransfer von der Steuerung an das Terminal kann auf folgende Arten ausgelöst werden:

- Durch Schreiben auf die Systemvariable **StartUpload** wird das Lesen eines Datensatzes für die aktive Rezeptur ausgelöst. Wird der Wert 1 geschrieben, so werden die Variablen einzeln von den bei der Variablenprojektierung angegebenen Adressen gelesen. Wird der Wert 2 geschrieben, so werden die Variablen als Block aus dem für die Rezeptur projektierten Datensatzpuffer gelesen. Die Nummer, unter der der Datensatz im Terminal abgelegt werden soll, kann in der Systemvariablen **UploadDSNr** angegeben werden. Diese Systemvariable ist mit 0 vorbesetzt. Wird also keine Zieldatensatznummer eingegeben, so wird der transferierte Datensatz vorübergehend unter der Nummer 0 abgelegt und muß von da auf eine gültige Nummer (1 - 250) umkopiert werden.
- Anforderung des Lesens eines beliebigen Datensatzes durch die Steuerung. Hierzu schreibt die Steuerung auf die für diesen Zweck vereinbarten Variablen die Rezeptur- und Datensatznummer

(dieselben wie beim Transfer zur Steuerung auf Anforderung der Steuerung) des Zieldatensatzes. Anschließend muß eine der Meldungsnummern **7FFDH** (Variablen werden einzeln von ihren projektierten Adressen gelesen) oder **7FFAH** (Variablen werden als Block aus dem projektierten Datensatzpuffer gelesen) abgesetzt werden. Nach erfolgreichem Abschluß des Datensatztransfers wird die Rezepturnummer mit 0 überschrieben. Wenn im Terminal kein Speicher mehr vorhanden ist oder wenn die angegebene Datensatznummer bereits im Flash-Speicher existiert, wird die Rezepturnummer mit 255 überschrieben. Die Datensatznummer wird in keinem Fall überschrieben.

Überschreibt der gelesene Datensatz einen unter derselben Nummer im RAM bereits vorhandenen Datensatz, so wird dessen Name für den gelesenen Datensatz übernommen. Ist unter der Nummer noch kein Datensatz vorhanden, besteht der Namen aus Leerzeichen.

Mit Hilfe der Systemvariablen **UploadState** kann vom Bediener der Lesevorgang überwacht werden.

3.8.4 Datensatztransfer an / von PC

Um die im Terminal gespeicherten Datensätze zu sichern, die Daten weiter zu verarbeiten oder das Terminal mit neuen Datensätzen zu versorgen, gibt es die Möglichkeit über die Schnittstelle X3 Datensätze an bzw. von einem PC zu transferieren.

Die Sicherung der Datensätze ist insbesondere auch erforderlich, wenn eine neue Anwenderbeschreibung in das Terminal geladen wird, da in diesem Fall alle Datensätze, die sich im RAM befinden, gelöscht werden. Sie können jedoch, sofern am Aufbau der Rezepturen nichts geändert wurde, nach dem Laden der Anwenderbeschreibung, ebenfalls wieder in das Terminal geladen werden. Wurde am Aufbau einer einzelnen Rezeptur (Anzahl der Variablen, Position der Variablen innerhalb des Datensatzpuffers, etc) eine Änderung vorgenommen, so können nur noch die Datensätze der anderen, nicht veränderten, Rezepturen wieder in das Terminal geladen werden.

Die Datensätze werden in einem mit einem Texteditor bearbeitbaren Format transferiert (Siehe Kapitel 3.8.4.3 *Aufbau der Datensatzdatei*).

Die Schnittstellenparameter für die Schnittstelle X3 können über die entsprechenden Systemvariablen frei eingestellt werden. Es ist lediglich sicherzustellen, daß auf der PC-Seite dieselben Parameter verwendet werden. Der Empfang bzw. das Senden auf der PC-Seite kann mit einem beliebigen dafür geeigneten Programm erfolgen, wie zum Beispiel Windows Terminal (1).

3.8.4.1 Transfer zum PC

Der Transfer von Datensätzen zum PC wird durch Schreiben auf die Systemvariable **StartSave** ausgelöst. Die Anzahl der transferierten Datensätze hängt vom geschriebenen Wert ab. Folgende Werte sind zulässig:

Systemvariablenwert = 1: Nur der aktive Datensatz wird transferiert.

Systemvariablenwert = 2: Alle Datensätze der aktiven Rezeptur werden transferiert.

Systemvariablenwert = 3: Alle Datensätze aller Rezepturen werden transferiert.

Mit Hilfe der Systemvariablen **SaveState** kann vom Bediener der Vorgang überwacht werden.

3.8.4.2 Transfer vom PC

Durch Schreiben des Wertes 1 auf die Systemvariable **StartRestore** wird das Terminal auf Empfangsbereitschaft gestellt. Anschließend können vom PC die Datensätze gesendet werden. Das Ende des Datensatztransfer wird vom Terminal anhand der empfangenen Daten selbständig erkannt. Es geht dann wieder in den Grundzustand.

Soll die Empfangsbereitschaft wieder zurückgenommen werden, ohne daß Daten empfangen wurden, kann dies durch Schreiben des Wertes 2 auf dieselbe Systemvariable **StartRestore** erfolgen.

Ob das Terminal empfangsbereit ist, kann mit Hilfe der Systemvariablen **RestoreState** überwacht werden.

Wird in den empfangenen Daten ein Formatfehler erkannt, so wird eine entsprechende Systemmeldung ausgegeben und der Empfang beendet. Die Position des Formatfehlers kann mit Hilfe der Systemvariablen **RestoreLineNr** zumindest näherungsweise lokalisiert werden. In dieser Systemvariablen wird die Nummer der gerade empfangenen Zeile angezeigt.

Datensätze können vom Terminal nur abgespeichert werden, wenn ihr Aufbau noch mit dem übereinstimmt, der in der Anwenderbeschreibung für die jeweilige Rezeptur beschrieben ist. Diese Übereinstimmung kann vom Terminal mittels einer Versionsnummer (siehe Aufbau der Datensatzdatei) überprüft werden. Wird ein in dieser Hinsicht unzulässiger Datensatz empfangen, so wird er unter Ausgabe einer entsprechenden Systemmeldung verworfen. Der Empfang wird jedoch nicht abgebrochen.

Ist ein Datensatz mit derselben Nummer, wie sie der transferierte Datensatz hat, bereits im Flash-Speicher vorhanden, so wird der empfangene Datensatz verworfen, ohne daß dies dem Bediener angezeigt wird.

Ist ein Datensatz mit derselben Nummer, wie sie der transferierte Datensatz hat, bereits im RAM vorhanden, so hängt es von einem ebenfalls in den empfangenen Daten enthaltenen Parameter (siehe Aufbau der Datensatzdatei) ab, ob der bereits vorhandene Datensatz überschrieben wird oder nicht. Wenn er nicht überschrieben werden soll und ein Datensatz mit derselben Nummer bereits im Terminal vorhanden ist, dann wird der empfangene Datensatz ebenfalls verworfen, ohne daß dies dem Bediener angezeigt wird.

3.8.4.3 Aufbau der Datensatzdatei

Die zum PC transferierten Datensätze werden im allgemeinen in einer Datei abgelegt.

Dient diese Datei nur zur Datensicherung, so muß der Benutzer nichts über ihren Aufbau wissen. Die Datei wird lediglich im Bedarfsfall wieder unverändert in das Terminal transferiert.

Sollen die Daten jedoch, zum Beispiel im Rahmen einer Betriebsdatenerfassung, weiterverarbeitet werden, so muß der Benutzer den Aufbau der Datei kennen.

Sämtliche in der Datensatzdatei enthaltenen Daten werden mit einer eigens dafür entwickelten, sehr einfachen Sprache dargestellt.

Die Elemente dieser Sprache sind:

Schlüsselwörter:	S + zwei weitere Buchstaben. Sie stehen normalerweise am Zeilenanfang. Beispiel: SDW oder SFA
Dezimalzahl:	Beliebige Anzahl der Ziffern 0-9 mit evtl. vorangestelltem Minuszeichen. Beispiel: 999 oder -1234567
Hexadezimalzahl:	H + beliebige Anzahl der Ziffern 0-9 oder Buchstaben A-F oder a-f. Beispiel: H999 oder H123abCD4
Hexadezimal-String:	C + beliebige geradzahlige Anzahl der Ziffern 0-9 oder Buchstaben A-F oder a-f. Beispiel: C12 oder CAFF33
ASCII-String:	Beliebige Zeichenkette zwischen zwei \-Zeichen eingeschlossen. Beispiel: \Dies ist 1 ASCII-String\
Kommentar:	beliebige Zeichenkette zwischen zwei \$-Zeichen eingeschlossen. Kommentare können an beliebiger Stelle in der Datensatzsicherungsdatei auch über mehrere Zeilen hinweg eingefügt werden. Beispiel: \$Dies ist 1 Kommentar\$

Zwischen diesen Sprachelementen können beliebig viele Trennzeichen (Leerzeichen, Tabulatorzeichen oder Zeilenvorschubzeichen) stehen.

Aus diesen Sprachelementen wird eine Datei mit folgender Struktur aufgebaut:

- **Dateianfangskennung**
 - **beliebig viele Datensätze**
 - **Dateiendekennung**
- Ein Datensatz besteht aus:
- **Datensatzkopf**

- **beliebig vielen Datensatzvariablen**

- **Datensatzendekennung**

Dateianfangskennung:

Schlüssel: SFA

Parameter: keine (Datum und Uhrzeit werden vom Terminal als Kommentar ausgegeben)

Dateiendekennung:

Schlüssel: SFE

Parameter: keine

Datensatzkopf:

Schlüssel: SDK

Parameter: Rezepturnummer, Datensatznummer, Datensatzname (als ASCII-String), Datensatzgröße in Byte, Rezepturversionsnummer, Überschreibkennung

Datensatzvariable:

Schlüssel: SDW

Parameter: Offset der Variablen innerhalb der Rezeptur, Variablengröße in Byte, Wert der Variablen (als Hexadezimal-String)

Datensatzendekennung:

Schlüssel: SDE

Parameter: keine

Erläuterungen:

Rezepturversionsnummer:

Diese Versionsnummer wird beim Erstellen bzw. Ändern der Rezepturbeschreibung im Projektiersystem immer dann automatisch erhöht, wenn sich der Aufbau der Datensätze geändert hat. Soll ein Datensatz vom PC ins Terminal geladen werden, muß die heruntergeladene Versionsnummer mit der, im Terminal für die entsprechende Rezeptur abgelegten Versionsnummer übereinstimmen. Bei ungleichen Versionsnummern wird der heruntergeladene Datensatz nicht abgespeichert.

Überschreibkennung:

Wert 1 bedeutet, daß ein im Terminal evtl. bereits vorhandener Datensatz mit derselben Nummer durch den heruntergeladenen Datensatz überschrieben werden soll. Wert 0 bedeutet, daß der heruntergeladene Datensatz verworfen werden soll, wenn bereits ein Datensatz mit derselben Nummer vorhanden ist. Überschreiben ist selbstverständlich nur bei solchen Datensätzen möglich, die sich nicht im Flash-Speicher befinden, das heißt die zusammen mit der Anwenderbeschreibung in das Terminal geladen wurden.

3.8.5 Datensätze drucken

Das Ausdrucken von Datensätzen kann sowohl vom Bedienterminal aus gestartet werden, wie auch von der Steuerung.

Um einen Ausdruck am Bedienterminal bewirken zu können, muß entweder die Systemvariable **StartRezPrint** in einer Maske aufgenommen, oder eine Softkeytaste entsprechend belegt werden.

Durch Schreiben des Wertes 1 auf die Systemvariable wird der aktive Datensatz über die Schnittstelle X3 ausgedruckt.

Durch Schreiben des Wertes 2 auf dieselbe Systemvariable kann der Druckvorgang abgebrochen werden.

Vor jedem Datensatz wird eine Titelzeile mit Rezepturnummer, Datensatznummer und Datensatznamen ausgedruckt.

Der Status des Druckvorgangs kann über die Systemvariable **RezPrintState** angezeigt werden.

Um einen Ausdruck von der Steuerung aus zu steuern, müssen zunächst die Datensatznummer und die Rezepturnummer in die entsprechenden Variablen eingetragen werden. Anschließend wird der Ausdruck durch schreiben des Wertes **7FF8H** auf die Adresse des seriellen Meldekanals ausgelöst.

Wird der Datensatz ausgedruckt, dann signalisiert dies der Wert **0** (Null) in der Variablen für die Rezepturnummer (für Anforderung vom Terminal).

Sollte der Drucker bereits einen anderen Druckjob ausführen und er kann den geforderten Datensatz nicht ausdrucken, wird der Wert **255** in die Variable für die Rezeptnummer (für Anforderung vom Terminal) geschrieben.

3.8.6 Speicherbedarf für Datensätze

Der im Terminal vorhandene, nicht vom System belegte RAM-Speicher, von ca. 110.000 Byte wird zur Speicherung von Meldungen und von im RAM abgelegten Datensätzen gemeinsam benutzt.

Die Größe des Meldungspuffers ist einstellbar. Für jede Meldung werden 24 Byte benötigt. Dies sind also für die standardmäßig eingestellte Meldungspuffergröße von 500 Meldungen insgesamt 12000 Byte, so daß für die Datensatzspeicherung noch 98000 Byte zur Verfügung stehen.

Außerdem wird zusätzlich noch Platz zur Speicherung des Datensatznamens und von Verwaltungsinformationen benötigt. Für diesen Zweck werden zusätzlich noch 28 Byte pro Datensatz benötigt.

Beispiel:

Bei einer projektierten Datensatzgröße von 22 Byte können (Meldungspuffergröße: 500) insgesamt $98000 / (22 + 28) = 1960$

Datensätze im RAM abgespeichert werden. Zusätzlich können noch weitere festprogrammierte Datensätze im Flash-Speicher abgelegt werden.

3.9 TesiMod-Meldungssystem

Das Meldungssystem ist ein fester Bestandteil des TesiMod-Bedienkonzeptes. Meldungen sind Reaktionen auf Ereignisse, um diese, dem Bediener in einer verständlichen Form mitzuteilen. Je nach Ereignisort, wird zwischen intern und extern erzeugten Meldungen unterschieden. Die folgende Grafik zeigt die Struktur des Meldesystems.

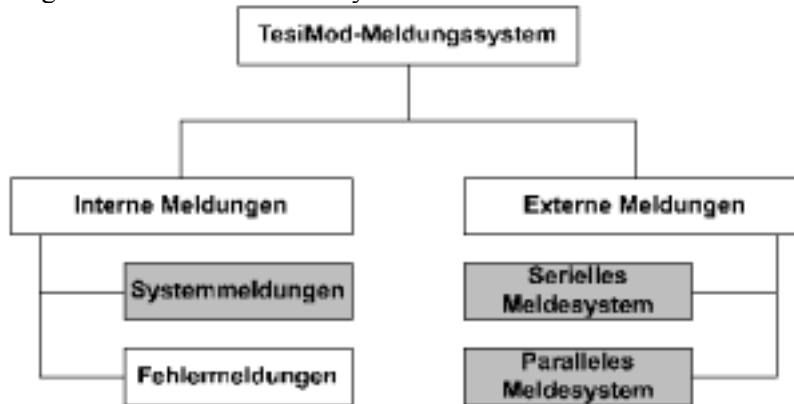


Bild 27: Aufbau TesiMod-Meldungssystems

Die grau hinterlegten Bereiche der Meldungen können vom Anwender frei gestaltet werden.

3.9.1 Interne Meldungen

In den Bereich der internen Meldungen fallen alle vom Betriebssystem erzeugten Ausgaben an den Bediener. Es wird nochmals zwischen den Systemmeldungen und den Fehlermeldungen unterschieden. Der Anwender hat keinen Einfluß auf die Erzeugung dieser Meldungen.

3.9.1.1 Systemmeldungen

Systemmeldungen werden vom Betriebssystem, aufgrund interner Plausibilitätsprüfungen, erzeugt. Eine Aktivierung erfolgt sofort nachdem ein entsprechendes Ereignis eingetreten ist.

Anstehende Systemmeldungen werden durch Blinken der Hilfetaste dem Bediener signalisiert. Nach Betätigen der Hilfetaste erscheint für die Dauer des Tastendrucks der ganzseitige Systemmeldetext. Beim gleichzeitigen Auftreten von mehreren Systemmeldungen erfolgt die Ausgabe in der Reihenfolge ihrer Systemnummern. Dabei hat, bei der Darstellung auf dem Display, die Systemmeldung mit der Nummer 1 die höchste Priorität.

Die Texte der Systemmeldungen können mit der Projektiersoftware vom Anwender editiert werden. Für jede Systemmeldung steht eine Displayseite zur Verfügung. Mit dem terminalspezifischen Zeichensatz läßt sich die Systemmeldung textlich gestalten. Zusätzliche Zeichenattribute oder Grafiken sind **nicht** möglich.

Die Ausgabe der Texte erfolgt sprachabhängig, so daß bei mehrsprachiger Bedienung, die Systemmeldungen in der jeweiligen Landessprache ausgegeben werden.

Die Systemmeldungen werden in der Projektiersoftware über die Systemmeldungsnummer zugeordnet. Die Systemmeldungsnummer steht für ein vordefiniertes Ereignis. Zur Erklärung der Systemnummer wird eine 20 Zeichen lange Kurzinfo verwendet. Die Länge der Texte ist so gestaltet, daß sie auch auf dem kleinsten Display der TesiMod-Bedienterminalsreihe in einer Zeile direkt darstellbar sind. Bei der Auswahl in der Projektiersoftware ist neben der Nummer die Kurzinfo als Meldungsname angegeben. Folgende Systemmeldungen mit Kurzinfo sind in einem neu angelegten System enthalten:

Systemmeldung-Nr.	Kurzinfo
1	Falsches Format
2	Wert ist zu groß
3	Wert ist zu klein
4	Batterie wechseln
5	Meldungs-Überlauf
6	Neue Meldung
7	Meldungspuffer voll
8	Ungültige Maskennr.
9	Ungültige Meldungsnr.
10	Druckprot. ungültig.
11	Schnittstelle belegt
12	Ungültiges Passwort
13	Passwort unverändert
14	Überspannung
15	Datensatz geschützt
16	Datensatz unzulässig
17	Datensatz unbekannt
18	Datensatzmemory voll
19	Datensatz aktiv
20	Datensatztransfer
21	Passwort fehlt
22	Editiermode aktiv
23	Datensatzdateifehler
24	Datensatzformat
25	Fließkommaz. ungült.
26	Durchschleifen aktiv
27	Keine Datensatzadr
28	Rezeptur unbekannt
29	Datensatzdownload

Systemmeldung 1 "Falsches Format"

Der Text erscheint beim Editieren, wenn versucht wird, in einem Variablenfeld des numerischen Editors ein ungültiges Datenformat einzugeben. Beispielsweise ist die Anzahl der eingegebenen Vorkommastellen größer als die Vorgabe durch die Applikationsbeschreibung.

Systemmeldung 2 "Wert ist zu groß"

Die Meldung erscheint beim Editieren, wenn versucht wird in einem Variablenfeld des Editors einen größeren Wert als die obere Bereichsgrenze der Variablen einzugeben. Der obere Grenzwert wird in der Anwenderbeschreibung festgelegt. Wird der Text aus der Systemmeldung gelöscht, wird nach Eingabe von einem zu großen Wert keine Systemmeldung ausgegeben, sondern der größtmögliche Wert eingetragen.

Systemmeldung 3 "Wert ist zu klein"

Die Meldung erscheint, wenn versucht wird in einem Variablenfeld des Editors einen kleineren Wert als die untere Bereichsgrenze der Variablen einzugeben. Der untere Grenzwert wird in der Anwenderbeschreibung festgelegt. Wird der Text aus der Systemmeldung gelöscht, wird nach Eingabe von einem zu kleinen Wert keine Systemmeldung ausgegeben, sondern der kleinstmögliche Wert eingetragen.

Systemmeldung 4....."Batterie wechseln"

Der Text erscheint, wenn bei der Kapazitätskontrolle der Batterie der Grenzwert unterschritten wurde. Die Prüfung wird alle 60 min wiederholt. Wird nach erstmaligem Auftreten der Meldung innerhalb von 2-3 Tagen die Batterie ausgetauscht, so bleiben die gespeicherten Rezepturdaten und die Daten im Meldungsspeicher erhalten. Damit beim Batteriewechsel kein Datenverlust entsteht, müssen die Hinweise im Handbuch des jeweiligen Bedienterminals beachten werden. Bei der Entnahme der Batterie erscheint die gleiche Meldung, jedoch gehen beim Ausschalten des Gerätes alle gepufferten Daten verloren!

Systemmeldung 5....."Meldungs-Überlauf"

Der Text erscheint, wenn externe Meldungen vom System nicht schnell genug bearbeitet werden können. Wenn diese Meldung erscheint ist bereits eine Meldung verloren gegangen.

Systemmeldung 6....."Neue Meldung"

Der Text erscheint, beim Betätigen der Hilfetaste, wenn im Terminal eine neue externe Meldung eingegangen ist, deren Priorität den programmierten Schwellwert überschritten hat und keine Direktanwahltaste der Meldungsmaske zugeordnet wurde.

Systemmeldung "Meldungspuffer voll"

Der Text erscheint, wenn die Gefahr besteht, daß die nächsten externen Meldungen die ältesten Meldungen oder die mit der niedrigsten Priorität (je nach Konfiguration) überschreiben.

Systemmeldung "Ungültige Maskennr"

Der Text erscheint, wenn aus der SPS über den seriellen Meldekanal eine nicht vorhandene Maskennummer übertragen wurde.

Systemmeldung "Ungültige Meldungsnr"

Der Text erscheint, wenn aus der SPS eine nicht programmierte Meldungsnummer angewählt wurde.

Systemmeldung 10....."Druckprot. ungültig"

Vom Bediener oder aus der SPS wurde versucht ein nicht programmiertes Druckprotokoll zu starten.

Systemmeldung 11....."Schnittstelle belegt"

Die Schnittstelle X3 ist bereits durch eine andere Druckausgabe belegt. Es wurde versucht den Drucker gleichzeitig für die Ausgabe von verschiedenen Daten zu nutzen (zum Beispiel Rezepturausdruck und Meldungen).

Systemmeldung 12....."Ungültiges Paßwort"

Bediener hat ein nicht in der Paßwortliste enthaltenes Paßwort eingegeben. Mit der Meldung werden die bisherige Zugangsberechtigung (View- und Edit-Level) zurückgesetzt.

Systemmeldung 13....."Paßwort unverändert"

Bediener hat nicht zweimal hintereinander dasselbe neue Paßwort eingeben.

Systemmeldung 14....."Überspannung"

Das Gerät hat eine zu hohe Versorgungsspannung erkannt. Gerät sofort abschalten bevor es Schaden nimmt. Versorgungsspannung kontrollieren.

Systemmeldung 15....."Datensatz geschützt"

Es wurde versucht, einzelne Werte eines im Flash abgelegten Datensatzes zu ändern oder den gesamten Datensatz zu löschen.

Systemmeldung 16....."Datensatz unzulässig"

Als Kopierziel beim Datensatz kopieren wurde eine Datensatznummer angegeben, die bereits existiert oder außerhalb des zulässigen Bereichs (zum Beispiel Flash) liegt.

Systemmeldung 17 "Datensatz unbekannt"

Es wurde ein Datensatz angewählt, dessen Nummer nicht im Datensatz-Verzeichnis enthalten ist.

Systemmeldung 18 "Datensatzmemory voll"

Es wurde versucht einen neuen Datensatz zu erzeugen, obwohl der Datensatzspeicher bereits voll ist.

Systemmeldung 19 "Datensatz aktiv"

Es wurde versucht den aktiven Datensatz zu löschen oder auf ihn zu kopieren oder einen Datensatz anzuwählen, obwohl der aktive Datensatz gerade editiert wird.

Systemmeldung 20 "Datensatztransfer"

Es wurde versucht einen Datensatztransfer in die Steuerung auszulösen, obwohl der zuvor ausgelöste Transfer noch nicht abgeschlossen war.

Systemmeldung 21 "Paßwort fehlt"

Es wurde versucht in eine paßwortgeschützte Maske zu wechseln, beziehungsweise in einer paßwortgeschützten Maske zu editieren, ohne ein ausreichend autorisiertes Paßwort eingegeben zu haben.

Systemmeldung 22 "Editiermode aktiv"

Es wurde versucht einen Maskenwechsel durchzuführen, während sich das Terminal im Editiermode befand.

Systemmeldung 23 "Datensatzdateifehler"

Die vom PC ins Terminal geladene Datensatzdatei enthält einen Syntaxfehler. Die Lokalisierung des Fehlers kann mit Hilfe der Zeilennummer-Systemvariablen erfolgen.

Systemmeldung 24 "Datensatzformat"

In einem vom PC in das Terminal geladenen Datensatz stimmt die Größe oder Offsetverteilungsversion nicht mit den entsprechenden Werten in der Projektiersoftware überein.

Systemmeldung 25 "Fließkommaz. ungült."

Aus der Steuerung wurde für eine Fließkommazahl ein unzulässiges Bitmuster gelesen. Die Zahl wird als 0,0 auf die Anzeige ausgegeben.

Systemmeldung 26 "Durchschleifen aktiv"

Die gewählte Aktion wurde wegen aktivem Durchschleifbetrieb verhindert.

Systemmeldung 27 "Keine Datensatzadr"

Die Adressen für den Datensatztransfer bei Anforderung durch die Steuerung sind nicht vorhanden.

Systemmeldung 28 "Rezeptur unbekannt"

Es wurde versucht, eine Rezeptur anzuwählen, die im Terminal nicht vorhanden ist.

Systemmeldung 29 "Datensatzdownload"

Es wurde ein Datensatztransfer zur Steuerung (Download) eingeleitet, aber das Bit Datensatz-Download-Freigabe im Koordinierungsbyte Schreiben (Bit 4) wurde von der Steuerung noch nicht erteilt.

3.9.1.1.1 Ausgabe der Systemmeldung verhindern

Das Bedienkonzept gibt Ihnen die Möglichkeit, die Ausgabe von Systemmeldungen zu unterdrücken. Durch Löschen einer Systemmeldung in der Projektiersoftware verhindern Sie die Ausgabe während des Betriebes.

Beispiel:

Die Systemmeldung 7 "Meldungspuffer voll" soll unterdrückt werden. Alte Meldungen oder solche mit niedriger Priorität sollen überschrieben werden.

Der Systemmeldungstext in der Projektiersoftware wird gelöscht.

Dadurch, dass diese Systemmeldung nicht mehr angezeigt wird, ist der Anwender damit einverstanden, dass die eingehenden Meldungen die ältesten Meldungen oder die mit der niedrigsten Priorität automatisch überschreiben, sobald der Meldungspuffer voll ist.

3.9.1.2 Fehlermeldungen

Die hier aufgeführten Meldungen werden vom Betriebssystem ausgegeben. Die Meldetexte sind Bestandteil des Betriebssystems und werden in englischer Sprache ausgegeben. Die Größe der Texte wurde so gewählt, dass sie auf allen Terminals darstellbar sind. Die Ausgabe der Texte kann nicht verhindert werden, ebenso ist es nicht möglich die Inhalte der Texte zu ändern. Der Begriff „Fehlermeldung“ wurde gewählt, weil während der Ausgabe dieser Meldungen kein Betrieb des Terminals im eigentlichen Sinne vom TesiMod Standard-Mode stattfindet. Neben echten Systemfehlern werden auch Zustände und Abläufe beschrieben.

```
COMMUNICATION ERROR
CODE                X
SUBCODE             XX
RETRIES             XXXX
```

Diese Meldung erscheint bei allen Protokoll- und Schnittstellenfehlern. Die Fehlercodes (CODE X) und SUBCODE (X) sind protokollabhängig und der entsprechenden Applikationsbeschreibung zu entnehmen. Die Verbindung zum Kommunikationspartner ist unterbrochen. Unter RETRIES wird die Anzahl der Fehlversuche für einen Verbindungsaufbau angezeigt. Die Anzahl summiert sich, solange das Gerät eingeschaltet ist. Die Häufigkeit der Wiederholungen ist vom jeweiligen Protokoll abhängig.

```
ADDRESS ERROR
.....
.....
```

Diese Meldung kann während der Download-Funktion erscheinen. Durch die S3-Datei werden physikalische Adressen im Terminal angesprochen. Die Übertragung bricht ab, sobald dabei ungültige Adressen erkannt werden. Die Startadresse des fehlerhaften S3-Satzes wird hexadezimal angegeben.

```
FLASH MEMORY FAILURE
.....
.....
```

Erscheint während Download, wenn Flash-EPROM nicht programmiert werden kann. Die Meldung deutet auf einen Defekt vom Applikationsspeicher hin. Die Startadresse des fehlerhaften S3-Satzes wird hexadezimal angegeben.

```
CHECKSUM ERROR
.....
.....
```

Fehler bei der Übertragung der Anwenderbeschreibung. Es traten entweder Fehler bei der seriellen Übertragung auf, die S3-Datei enthält ungültige Zeilen oder es wurde keine gültige S3-Datei übertragen. Applikationsbeschreibung neu übersetzen und erneute Übertragung versuchen.

```
BYTECOUNT OVERFLOW
.....
.....
```

Fehler bei der Übertragung der Anwenderbeschreibung. Es wurde Fehler in der S3-Datei der Applikationsbeschreibung erkannt. In einer Übertragungszeile wurden mehr Bytes empfangen als im Bytecount angegeben waren.

```
FORMAT ERROR
.....
.....
```

Übertragungsformat der Anwenderbeschreibung fehlerhaft. Es wurde keine Ausgabedatei des Projektiersystems verwendet. Die übertragene Datei enthielt keine S0, S3 oder S7- Sätze, es war kein S3-Format.

```
TURN POWER OFF
RESET DIP-SW 4
OTHERWISE ALL FLASH-
DATA WILL BE LOST
```

Beim Einschalten der Versorgungsspannung am Terminal, war der Betriebsartenschalter S4 eingeschaltet. Bei Einhaltung der Vorgaben bleiben die Flash-Daten erhalten. Gerät ausschalten, S4 ausschalten, Gerät einschalten - Daten bleiben erhalten und Gerät arbeitet wie bisher. S4 ausschalten unter Spannung - Daten gehen verloren, Gerät wechselt in den Download-Mode!

```
DIFFERENT MASK VERS
          EPROM FLASH
VERSION   XXX   XXX
REVISION  XXX   XXX
```

Die Programmversion des Projektiersystems TS und das Betriebssystem im Terminal passen nicht zueinander. Dieser Fehler tritt auf, wenn bei der Übersetzung der Anwenderbeschreibung die falsche Betriebssystemversion angewählt wurde. Die beiden Programmversionen müssen zueinander passen.

```
DIFFERENT DRIV VERS
          EPROM FLASH
VERSION   XXX   XXX
REVISION  XXX   XXX
```

Der über das Projektiersystem TS geladene Protokoll-Treiber und Betriebssystem des Terminals differieren. Die beiden Programmversionen müssen zueinander passen.

```
!!!! WARNING !!!!
NONE DEFAULT PARA-
METERS ON SERIAL
PORT X2 USED
```

Die Parameter der Schnittstelle X2 wurden verändert. Die Verbindung ist nur funktionsfähig wenn beide Kommunikationspartner auf die neuen Parameter angepaßt worden sind. Dient als Hinweis, falls kein Verbindungsaufbau zum Kommunikationspartner zustande kommt.

```
!!!! ERROR !!!!
NO PROTOCOL-DRIVER
IN MASK DESCRIPTION
FOUND
```

Das Betriebssystem findet in der geladenen Anwenderbeschreibung keinen Protokolltreiber. Anwenderbeschreibung mit angewähltem Protokoll neu übersetzten und erneut über Download laden.

```
!!!! ERROR !!!!
PLC TYPE MISMATCH
BETWEEN TERMINAL
AND MASK DESCRIPTION
```

Das im Projektiersystem gewählte Protokoll bei der Erstellung der Applikationsbeschreibung entspricht nicht der Hardware des Terminals. Es wurde zum Beispiel der Interbus Protokolltreiber in ein Gerät mit Standard-Schnittstellen geladen.

```
KEYBOARD ERROR
PLEASE RELEASE KEY
.....
```

Beim Einschalten des Terminals wird ein Selbsttest durchgeführt und die Tastatur überprüft, dabei dürfen keine Tasten betätigt sein. Kommen sie bitte der Aufforderung nach. Erfolgt die Meldung ohne daß eine Taste betätigt wurde, so liegt ein Tastatur-Defekt vor!

```
INITIALIZING
MESSAGE BUFFER
.....
```

Beim Einschalten des Terminals werden die im Terminal vorhandenen Meldungen sortiert, dies benötigt je nach Anzahl der gespeicherten Meldungen eine bestimmte Initialisierungszeit. Die Meldung kommt immer, ist jedoch meist sehr kurz oder gar nicht sichtbar.

```
ERASE FLASH EPROM
.....
```

Erscheint auf dem Display während der Maskenspeicher gelöscht wird. Zu diesem Zeitpunkt gehen die vorher programmierten Daten verloren.

```
FLASH IS ERASED
.....
```

Diese Mitteilung erscheint auf dem Display, wenn der Löschvorgang abgeschlossen ist. Die Schnittstelle X3 wird für den Download-Betriebszustand initialisiert.

```
DOWNLOAD .....  
.....  
.....
```

Diese Mitteilung erscheint auf dem Display, wenn das Terminal bereit ist, über die Schnittstelle X3 eine neue Anwenderbeschreibung zu empfangen. (nur bei TSdos)

```
DOWNLOAD 1 .....  
FLASH XXX kByte .....
```

Das Bedienterminal zeigt seine Downloadbereitschaft mit der Baudrate 19200 Bd über die Schnittstelle X3 an. Jetzt kann ein neues Projekt geladen oder neue Schnittstellenparameter ausgetauscht für die Übertragung ausgetauscht werden. (nur bei TSwIn)

```
DOWNLOAD 2 .....  
.....  
.....
```

Das Bedienterminal zeigt Downloadbereitschaft mit den neuen Schnittstellenparametern an. Empfängt das Bedienterminal innerhalb von 20 s keine Daten, fällt es in den Zustand DOWNLOAD 1 zurück. (nur bei TswIn)

```
AUTO REBOOT .....  
.....  
.....
```

Innerhalb der nächsten Sekunden führt das Bedienterminal einen Neustart durch.

```
INITIALIZING .....  
CPU XX MHz .....  
FLASH XXX kByte .....  
XXXXXXXXX YYYYYYYYY
```

Das Bedienterminal meldet beim Startvorgang seine Parameter:

- CPU-Frequenz in MHz
- Größe vom Flash-Speicher in kByte
- Eprom-Versionsnummer XXXXXXXXX
- Geladener SPS-Treiber YYYYYYYYY

```
IDENTIFY MEMORY-TYP .....  
.....  
.....
```

Der vorhandene Flash-Speichertyp wird identifiziert.

```
!!! HIGH VOLTAGE !!! .....  
.....  
.....
```

Es liegt Überspannung am Bedienterminal an! Diese Meldung erlischt es wieder nach Erreichen der spezifizierten Versorgungsspannung.

```
ERROR ASYNCHRONOUS .....  
SERIAL I/O UNIT 0 .....
```

Die serielle Schnittstelle (Unit 0 oder Unit 1) konnte nicht erfolgreich initialisiert werden.

```
SUCONETK-MODUL .....  
HARDWARE-VERSION .....  
NOT CONFORM TO .....  
DRIVER VERSION .....
```

Der Programmstand der SUCOnet K Platine ist nicht kompatibel zum aktuellen Protokolltreiber. Das Bedienterminal muß umgerüstet werden oder die entsprechende Treiberversion verwendet werden. Der Subcode gibt den Stand der SUCOnet K-Platine an.

```
KEYBOARD-MODUL .....  
HARDWARE-VERSION .....  
NOT CONFORM TO .....  
DRIVER VERSION .....
```

Der Programmstand der Tastaturplatine ist nicht kompatibel zur aktuellen Firmware. Das Bedienterminal muß umgerüstet werden. Der Subcode gibt den Stand der Tastaturplatine an

```
FIRMWARE UPDATE .....  
SUCCESSFUL .....  
AUTO-REBOOT .....
```

Der Updatevorgang wurde erfolgreich durchgeführt. Das Bedienterminal führt einen automatischen Neustart durch.

```
SYSTEM ERROR
CODE      : XX
SUBCODE   : XXXX
RETRIES   : XXXXX
```

Ein schwerwiegender Fehler ist aufgetreten. Wenn diese Fehlermeldung auftritt, müssen Sie sich mit EUROTHERM Antriebstechnik GmbH, Im Sand 14, 76669 Bad Schönborn, Hotline-Nr.: 07253 / 94 04 0 in Verbindung setzen. Geben Sie dann bitte die Nummern für den Firmwarestand und die Hardwareversion durch.

```
UNEXPECTED INTERRUPT
NR =
IP =
CALL HOTLINE
```

Ein unerwarteter Interrupt ist aufgetreten. Setzen Sie sich mit EUROTHERM Antriebstechnik GmbH, Im Sand 14, 76669 Bad Schönborn, Hotline-Nr.: 07253 / 94 04 0 in Verbindung. Geben Sie dann bitte die Interruptnummer (NR) und den Stand vom Programmzähler (IP) durch.

```
FLASH NOT ERASABLE
.....
.....
```

Erscheint nach dem Einschalten oder vor dem Download, falls der Flash-Speicher nicht gelöscht werden kann.

```
WRONG S3 FILES
.....
.....
```

Erscheint zu Beginn des Downloads, falls die S3-Datei nicht zu dem eingesetzten Terminal paßt.

```
NO FLASH EPROM
.....
.....
```

Diese Mitteilung erscheint auf dem Display, wenn kein vom Programmieralgorithmus unterstütztes Flash erkannt wird.

```
FLASH CHECKSUM ERROR
.....
.....
```

Die im Flash befindliche Anwenderbeschreibung ist fehlerhaft. Fehler kann am Ende einer Übertragung auftreten, zum Beispiel wenn die Übertragung unvollständig war oder nach dem Einschalten des Gerätes wenn ein Speicherdefekt vorliegt.

```
TERMINAL-TYP IS XXXX
.....
.....
```

Es wurde versucht eine S3-Datei zu laden, die für einen anderen Terminaltyp bestimmt war. Beim Auftreten des Fehlers erscheint in der Anzeige an der "XXXX"-Stelle der für dieses Terminal richtige Typ. Mit dieser Anwahl muß im Projektiersystem neu übersetzt werden.

```
MEMORY IS FLASH XXXX
.....
.....
```

Es wurde versucht eine S3-Datei zu laden, welche für einen größeren Maskenspeicher erstellt worden ist. Die angeforderte Speichergröße durch die S3-Datei stimmt mit der im Terminal verfügbaren nicht überein. Beim Auftreten des Fehlers erscheint in der Anzeige an der "XXX"-Stelle die im Gerät befindliche Speichergröße in kByte. Dieser Wert ist in der Projektiersoftware bei der Übersetzung einzustellen.

```
FATAL ERROR
CODE      : XXXX
SUBCODE   : XXXX
CALL HOTLINE
```

Eine Fehlermeldung die eigentlich nie auftreten dürfte, dennoch ist sie vorhanden. Vom Betriebssystem des Terminals wird dieser Fehler erzeugt, wenn mangels fehlender Plausibilität eine sinnvolle Weiterarbeit nicht möglich ist. Für eine Rekonstruktion des Vorfalls ist es notwendig die Code- und Subcodennummer, sowie die Softwareversionen von Betriebssystem und Projektieroberfläche zu kennen. Ein Anruf bei der Hotline hilft hier sicher weiter!

```
FIRMWARE NOT CONFORM
TO HARDWARE
FIRMWARE 1
HW-VERSION 2
```

Wenn diese Fehlermeldung auftritt, müssen Sie sich mit EUROTHERM Antriebstechnik GmbH, Im Sand 14, 76669 Bad Schönborn, Hotline-Nr.: 07253 / 94 04 0 in Verbindung setzen. Geben Sie dann bitte die Nummern für den Firmwarestand und die Hardwareversion durch. Damit kein Gerätedefekt auftreten kann, springt das Betriebssystem des Bedienterminals in eine Endlosschleife.

```
DATASET STORAGE
FAILURE
```

Bei der Überprüfung der Speicherbereiche der Rezepturdatensätze wurde ein Prüfsummenfehler festgestellt. Entweder ist die Batterie oder der RAM-Speicher defekt.

3.9.2 Externe Meldungen

Alle externen Meldungen sind Informationen, die von der angeschlossenen Steuerung aufgrund des zu überwachenden Prozesses generiert und dem Terminal mitteilt werden. Der Anwender hat die Möglichkeit zwei voneinander unabhängige Meldesysteme zum Einsatz zu bringen. Je nach Anforderung kann die Meldung seriell oder parallel an das Terminal übertragen werden. Es ist dabei unerheblich, ob es sich dabei um Betriebs- oder Störmeldungen handelt.

Eine Meldung kann aus dem Meldetext und einer skalierten und formatierten Variablen bestehen. Jeder im System vorhandene Variablentyp ist zulässig.

Die Informationen des Meldespeichers können zu statistischen Auswertungen verwendet werden. Die Zuordnung der Meldung zwischen Terminal und Steuerung erfolgt über die Meldungsnummer. Die zugehörigen Texte mit Variablenspezifikation werden mit der Anwenderbeschreibung im Terminal gespeichert. Die Funktion der Meldung und deren textlicher Inhalt wird vom Anwender beim Erstellen der Anwenderbeschreibung im Projektiersystem festgelegt.

Alle externen Meldungen werden, chronologisch oder nach der Priorität sortiert, im Meldespeicher abgelegt. Bei TSwIn-Projekten gibt es die Möglichkeit, parallel abgesetzte Meldungen in den Speicher für serielle Meldungen zu übernehmen, damit diese ebenfalls statistisch ausgewertet werden können. Der Wert einer in der Meldung enthaltenen Variablen wird im Meldungsspeicher eingefroren.

3.9.2.1 Aufbau einer externen Meldung

Eine externe Meldung besteht aus

- einer Meldungsnummer von 1 bis 9999
- einem Meldungstext mit bis zu 80 Zeichen (TSdos) oder
- einem Meldungstext mit bis zu 255 Zeichen (TswIn)
- höchstens einer Variablen.

Bei der Erstellung einer neuen Applikation, können bestehende Meldungen einzeln oder komplett übernommen werden.

3.9.2.1.1 Vergabe der Meldungsnummern

Bei externen Meldungen wird durch die Meldungsnummer gleichzeitig die Priorität der Meldung festgelegt, wobei die Meldung Nr. 1 die höchste und die Nr. 9999 die niedrigste Priorität hat. Die Meldungsnummern müssen nicht fortlaufend vergeben werden, dadurch ist eine sinnvolle Gruppierung der Meldungen nach ihrem Inhalt möglich.

Die Vergabe der Meldungsnummern im Bereich der Zustandsmeldungen beginnt immer bei "1". Es ist darauf zu achten, daß sich das serielle und parallele Meldesystem nicht überschneidet. Falls beide Meldesysteme unabhängig voneinander sein sollen, ist darauf zu achten, daß die Meldungsnummern des seriellen Systems oberhalb der Zustandsmeldungen beginnen. Ebenso müssen Meldungsnummern

und Maskennummern bei der Projektierung entsprechend koordiniert werden (Siehe Kapitel 3.9.2.3.1 *Ganzseitige Meldungsangabe*).

Vom System aus ist es möglich auch Zustandsmeldetexte im seriellen Meldesystem zu verwenden.

3.9.2.1.2 Größe des Meldungspuffers

Für den Meldungspuffer ist eine Gesamtgröße zur Verwaltung von bis zu 3000 Meldungseinträgen vorgesehen. Diese hohe Anzahl von Daten erfordert entsprechende Rechenleistung beim Einsortieren der Meldungen, beim Umsortieren und beim Initialisieren. Da diese Zahl sehr häufig nicht benötigt wird, wurde die Einstellung der maximale Puffergröße im System dem Anwender überlassen. Die Grundeinstellung der Puffergröße erlaubt 500 Einträge.

Bei der Einstellung ist zu beachten, daß bei einem kompletten Ausdruck des Meldungspuffers mit zum Beispiel 3000 Meldungen, ca. 50 Seiten Druckerpapier benötigt werden!

Die Ausgabe des Meldungspuffers erfolgt in der Meldungsmaske. Die Sortierung der Meldungen kann durch eine Systemvariable festgelegt werden.

3.9.2.1.3 Meldetext, Variable

Die Textlänge inklusive einer formatierten Variable, darf 80 (TSdos), bzw 255 (TSwin) Zeichen nicht überschreiten. Die Eingabe längerer Texte wird vom Projektiersystem verhindert. Alle Zeichen des jeweils spezifizierten Terminals in Normalgröße sind zulässig. In jedem Meldetext darf eine Ausgabevariable enthalten sein. Das Ausgabeformat der Variablen ist identisch zu einmaligen Ausgabevariablen in E/A-Masken. So können beispielsweise durch kodierte Texte einzelne Meldungen modifiziert oder für mehrere Zustände verwendet werden. Das Ausgabeformat der Meldungszeile kann in einer Konfigurationsmaske zur Meldungsmaske Online verändert werden.

Die Möglichkeiten sind für serielle und parallele Meldungen gleich.

Beispiel: Vollständiges Meldungsformat:

```
Nr. Datum Uhrzeit Text1 Variable Text2
1234 25:08:92 11:30:00 Temperatur 285 °C an Meßst.07
```

Die Aufteilung ergibt sich wie folgt:

1234 4-stellige Meldungsnummer

25.08.92 Datum, wird beim Erfassen der Meldung im Terminal gespeichert.

11:30:00 Uhrzeit, wird beim Erfassen der Meldung im Terminal gespeichert.

Temperatur Text(1) vor der Variablen

285 Wert der Variablen zum Zeitpunkt der Meldung.

°C an Meßst.07 Text(2) nach der Variablen

3.9.2.1.4 Sortieren von Meldungen

Meldungen können wahlweise nach dem zeitlichen Eintreffen oder deren Priorität in der Meldungsmaske dargestellt werden. Die Vorbelegung, welche Reihenfolge gezeigt werden soll, kann bereits bei der Projektierung erfolgen. Die Angaben können beim Vorhandensein beider möglicher Meldesysteme getrennt vorgewählt werden. Die Angaben werden in den Systemvariablen **RepmanRepSortCrit** und **RepmanSortCritP** gespeichert. Die Einstellungen können Online am Terminal, zum Beispiel in einer Konfigurationsmaske, geändert werden. Wird dem Bediener keine Möglichkeit der Einstellung gegeben, gelten die Einstellungen, die der Anwender projektiert hat.

Sortieralgorithmus vom seriellen Meldesystem:

0 - nach Priorität

1 - nach zeitlichem Eintreffen (neueste zuerst)

2 - nach zeitlichem Eintreffen (älteste zuerst)

Sortieralgorithmus vom parallelen Meldesystem:

- 0 - nach Priorität
- 1 - nach zeitlichem Eintreffen (neueste zuerst)
- 2 - nach zeitlichem Eintreffen (älteste zuerst)

3.9.2.1.5 Meldungspriorität für Direktanzeige

Die Priorität der Meldung wird durch deren Meldungsnummer bestimmt. Je höher die Meldungsnummer, desto niedriger die Priorität. In der Projektiersoftware kann in der Parameterdatei (TSdos), beziehungsweise in den Systemparametern des Meldungssystems (TSwin) ein Wert eingetragen werden, der die Obergrenze der Meldungsnummer darstellt, welche beim Eintreffen durch Blinken der Melde-LED oder Ausgabe einer Systemmeldung angezeigt wird.

Wird hier der Wert = 0 eingetragen, dann wird das Signalisieren neu eingehender Meldungen generell unterdrückt!

3.9.2.1.6 Ausdruck des Meldespeichers

Die Meldespeicher des seriellen und des parallelen Meldesystems können komplett oder auszugsweise ausgedruckt werden.

Der gesamte Meldungsspeicher des **seriellen** Meldungssystems wird ausgedruckt, wenn die Systemvariable **PrintAllRep** auf den Wert 1 (Ausdruck mit Formatierung) oder den Wert 2 (Ausdruck in voller Länge) gesetzt wird.

Der gesamte Meldungsspeicher des **parallelen** Meldungssystems wird ausgedruckt, wenn eine Softkeytaste betätigt wird, die mit der Systemvariablen **PrintAllState** verknüpft ist.

Um auszugsweise auszudrucken, werden die dafür vorgesehenen Meldungen in der Meldemaske markiert. Dazu wird in der Meldungsmaske die Taste Datenfreigabe betätigt und anschließend mit den Tasten Cursor Oben oder Cursor Unten die Meldungen im Meldungsfeld markiert. Um den Ausdruck zu starten, muß dann eine Softkeytaste betätigt werden, die mit der Systemvariablen **BlockPrint** (Sichtbaren Teil vom markierten Block ausdrucken) oder **BlockPrintLong** (Meldungen vom markierten Block in voller Länge ausdrucken) verknüpft ist.

Die Systemvariablen können zusätzlich in einer Konfigurationsmaske aufgenommen und Online verändert werden.

3.9.2.2 Meldungsmaske, Zustandsmeldungsmaske

Die Meldungsmaske ist nur bei TSdos eine spezielle E/A-Maske, die für die Ausgabe der Meldungen modifiziert wurde. Bei TSwIn wird eine E/A-Maske mit einem Meldungsfeld versehen, um die gleichen Funktionen bieten zu können. Eine Meldungs- bzw Zustandsmeldemaske verfügt über eine spezielle Tastenbelegung zur Bedienung der Ausgabe des sehr umfangreichen Meldungsspeichers.

- Datenfreigabe nicht aktiv	Cursor Oben: Cursor Unten: Cursor Links: Cursor Rechts: Minus: Plus: Löschen: Datenfreigabe: Dateneingabe:	Bewegt den Zeiger um eine Meldung nach Oben. Nach Erreichen der obersten Meldung keine weitere Bewegung. Bewegt den Zeiger um eine Meldung nach Unten. Nach Erreichen der untersten Meldung keine weitere Bewegung. Bewegt den Zeiger um eine Displayseite nach Oben (Repeatfunktion). Nach Erreichen der obersten Meldung keine weitere Bewegung. Bewegt den Zeiger um eine Displayseite nach Unten (Repeatfunktion). Nach Erreichen der untersten Meldung keine weitere Bewegung. Bewegt den Zeiger zur untersten Meldung. Bewegt den Zeiger zur obersten Meldung. inaktiv Wechselt in den Editiermode, unter der Voraussetzung, daß die externe Datenfreigabe erteilt wurde und der Zugriffslevel des eingegebenen Paßwortes ausreicht. Stellt die Meldung an der aktuellen Zeigerposition komplett auf einer Displayseite dar.
-Datenfreigabe aktiv	Cursor Oben: Cursor Unten: Cursor Links: Cursor Rechts: Löschen: Minus: Plus: Datenfreigabe:	Markiert von aktueller Zeigerposition an nach Oben einzelne Meldungen. Markiert von aktueller Zeigerposition an nach Unten einzelne Meldungen. Markiert von aktueller Zeigerposition an nach Oben seitenweise bis zur obersten Meldung. Keine Repeatfunktion. Markiert von aktueller Zeigerposition an nach Unten seitenweise bis zur untersten Meldung. Keine Repeatfunktion. Löscht alle markierten Meldungseinträge. Ohne Markierung wird die Meldung an der Cursorposition gelöscht. Markiert von aktueller Zeigerposition an bis zur letzten Meldung. Markiert von aktueller Zeigerposition an bis zur ersten Meldung. Verläßt den Editiermode, unter der Voraussetzung, daß die externe Datenfreigabe noch erteilt war.

3.9.2.2.1 Direktanwahl der Meldemaske

In der Projektiersoftware kann eine Funktionstaste mit einer Meldungs-
maske verknüpft werden. Über diese Funktionstaste kann dann aus jeder Maske
aus in die Meldungs-
maske gewechselt werden. Die Meldemaske wird dann neben dem Zugang aus
einem Auswahlmenü, auch mit der Funktionstaste erreicht. Die integrierte
LED der Funktionstaste übernimmt dann das Signalisieren neu eingegan-
gener Meldungen. In diesem Falle blinkt die LED bei einer neuen Meldung.

Beim Betätigen der blinkenden Funktionstaste wird direkt in die Meldung-
maske gewechselt. Beim erneuten Betätigen der Funktionstaste erfolgt der
Rücksprung in die vorige Maske automatisch. Die normalerweise blinkende
Hilfetasten-LED entfällt bei der Projektierung einer Funktionstaste.

3.9.2.2.2 Ausgabeformate der Meldungen

Zu jeder externen Meldung gibt es folgende Informationen:

- Meldungsnummer
- Datum
- Uhrzeit
- Meldetext
- Wert einer Variablen, zum Zeitpunkt des Auftretens der Meldung (nur falls vorhanden)

Die Darstellung der Meldung in der Meldemaske oder auf dem Drucker kann durch
verschiedene Systemparameter beeinflusst werden. In einer eventuell projektierten
Konfigurationsmaske könnten diese Parameter dann eingestellt werden.

Die An- und Abwahl von Meldungselementen erfolgt dann über Systemvariablen.

Serielle Meldungen	Parallele Meldungen	Beeinflußt
- RepoutNr	RepoutNrP	Meldungsnummer
- RepoutDate	RepoutDateP	Datum
- RepoutTime	RepoutTimeP	Uhrzeit
- RepoutAnzYear	RepoutAnzYearP	2- oder 4-stellige Ausgabe vom Jahr

Damit läßt sich die Länge der Meldungszeile beeinflussen. Ausgabenänderungen
beeinflussen nicht die gespeicherte Information.

Folgende Ausgabe-Varianten sind einstellbar:

Vollständiges Meldungsformat:

```
Nr. Datum Uhrzeit Text1 Variable Text2
1234 25:08:92 11:30:00 Temperatur 285 °C an Meßst.07
```

Varianten:

```
1234 25:08:92 Temperatur 285 °C an Meßst.07
1234 11:30:00 Temperatur 285 °C an Meßst.07
1234 Temperatur 285 °C an Meßst.07
25:08:92 11:30:00 Temperatur 285 °C an Meßst.07
11:30:00 Temperatur 285 °C an Meßst.07
25:08:92 Temperatur 285 °C an Meßst.07
Temperatur 285 °C an Meßst.07
```

3.9.2.2.3 Zoomen von Meldungen

Die Meldungen werden in der Meldungsmaske wegen des besseren Überblicks einzeilig dargestellt. Da jedoch Meldungen mehr Zeichen beinhalten können, besteht die Möglichkeit, die angewählte Meldung durch betätigen der Dateneingabetaste vollständig auszugeben.

Zeile der Meldungsmaske zum Beispiel im BT20

1234 25:08:92 11:30:00 Die Meßstelle im

Zoomdarstellung:

**1234 25:08:92 11:30:00
Die Meßstelle 137 im Ofen hat eine
Temperatur von 285 °C**

Die Zoomdarstellung erscheint, solange die Datenfreigabetaste gedrückt bleibt. Bei kleineren Displays zum Beispiel IBT mit 4 x 20 Zeichen wird nur der Meldetext gezoomt. Für den richtigen Zeilenumbruch ist bei der Projektierung der jeweilige Terminaltyp zu berücksichtigen.

3.9.2.2.4 Quittierung von Meldungen

Meldungen können in der Steuerung über Variable quittiert werden. Dazu eignen sich verschiedene Editoren oder Funktionstasten (Softkeys). Durch die Quittung kann die Steuerung ein Löschen der Meldung und eine erneute Überprüfung auslösen.

3.9.2.3 Serielles Meldesystem

Im zyklischen Pollbereich wird ein 2 Byte großes Fach für die Übertragung der seriellen Meldungen verwendet. Die Byte-Order ist vom verwendeten Datentyp des Pollbereichs abhängig (siehe Pollbereich). In dieses Sendefach legt die SPS eine 16 Bit Meldungsnummer ab. Das TesiMod Bedienterminal holt aus der SPS den gesamten Pollbereich zyklisch ab und überträgt dabei auch die serielle Meldung. Sobald eine Meldung (Meldungsnummer > 0) erkannt wird, wird diese im internen Meldungsspeicher des Bedienterminals abgelegt und das Fach in der SPS auf Null zurückgesetzt. Der Wert Null im Fach bedeutet für die SPS, daß das Terminal die Meldung abgeholt hat. Die Pollzeit des zyklischen Datenbereichs ist einstellbar.

In der gleichen Art werden externe Masken und Meldungsmasken angesprochen. Sobald die übertragene Nummer einer Maskennummer entspricht, wird diese auf dem Display dargestellt. Ist von der Nummer gleichzeitig eine Maske und ein Meldetext vorhanden, so wird die Maske (Meldungsmaske, ganzseitiger Störmeldetext) auf dem Display dargestellt und der zugehörige Meldetext in den Meldungsspeicher aufgenommen.

Es ist darauf zu achten, daß die Meldungsnummer immer als 16 Bit Befehl in das serielle Datenfach eingetragen wird. Durch die asynchrone Bearbeitung mancher Übertragungsprotokolle kann die Auswertung der Meldungsnummer zu Problemen führen, wenn die Meldungsnummer mit Einzelbyte-Befehlen eingetragen wurde.

3.9.2.3.1 Ganzseitige Meldungsangabe

Die ganzseitige Meldung ist eine Verknüpfung der Meldungsbearbeitung und der externen Maskenanwahl. Zur Durchführung der ganzseitigen Meldemaskenangabe wird unter der gleichen Nummer eine Maske und ein Meldetext projiziert. Die Steuerung ruft über den seriellen Meldekanal die "externe Maske" auf. Beim Aufruf wird die Maske auf dem Display dargestellt und der zugehörige Meldetext in den Meldungsspeicher übernommen. Da der Displayinhalt frei gestaltet werden kann, besteht die Möglichkeit eine Meldemaske, ganzseitige Fehlerangabe oder sonstige Einblendungen zu realisieren. Möchte man anschließend in das vorherige Menü zurückkommen, dann ist mindestens ein Maskenparameter auf "Vorgängermaske" zu programmieren. Meldemaske können natürlich aus mehreren Masken oder kompletten Strukturen zur Fehlerbeseitigung bestehen. Natürlich ist bei jeder ganzseitigen Meldung auch ein eigener ganzseitiger Hilfetext projektierbar.

3.9.2.3.2 Meldungen direkt auf Protokolldrucker

Beim direkten Protokollieren von seriellen Meldungen läuft der Drucker stets mit. Jede neu ankommende Meldung über den seriellen Meldekanal wird sofort ausgedruckt und parallel dazu in den Meldungsspeicher aufgenommen. Es ist hier zu beachten, daß der Drucker immer nur einen Druckauftrag bearbeiten kann. Bevor eine weitere Druckeranforderung vom System gestartet wird, muß die vorherige beendet werden.

Die Ausgabe der Meldungen auf den Drucker kann durch die Systemvariable **RepmanRepPrint** beeinflusst werden.

Wird die formatierte Art des Ausdrucks gewählt, gelten Einstellungen, die auch für die Anzeige der Meldungen in der Meldemaske getroffen wurden.

Die Einstellungen für den Ausdruck können Online am Terminal geändert werden.

Da es sich bei der Ausgabe um eine reine Textdatei handelt, kann die Meldung auch von einem Hostrechner oder PC eingelesen werden. Mit einer weiteren Systemvariable **PrintAllRepLong** kann generell die volle Länge der Meldung ausgegeben werden.

3.9.2.3.3 Meldespeicher von extern löschen

Der interne Meldespeicher des seriellen Meldesystems kann von extern, also von der Steuerung gelöscht werden. Dazu ist in der Projektiersoftware in der Parameterdatei unter Meldesystem ein symbolischer Variablenname für die Löschvariable anzugeben. Für die Variable sind 2 Byte in der Steuerung erforderlich.

Das Terminal prüft nach Erhalt der Löschsequenz (Löschcode **7FFE_H** über den seriellen Meldekanal) immer die Löschvariable in der Steuerung. Wenn in der Löschvariablen das Bitmuster **E216_H** steht, wird der interne Meldespeicher gelöscht. Die Löschvariable erhöht die Sicherheit vor unbeabsichtigtem Löschen.

Falls kein Löschen gewünscht wird, sollte die Variable zurückgesetzt oder in der Projektiersoftware kein symbolischer Name angegeben werden.

3.9.2.3.4 Hinweise zum seriellen Meldesystem

Wie gelangen die Meldungen ins Terminal?

In der Steuerung wird eine Wort-Variable, als Teil des zyklischen Pollbereichs, für die Meldungsnummer reserviert. Das Bedienterminal pollt diese Variable. Wenn der Variablenwert größer als null ist, wird dies als numerischer Wert einer Meldung, der Meldungsnummer, erkannt. Die Meldungsnummer wird in den Meldungsspeicher des Terminals aufgenommen und die Variable in der Steuerung mit null überschrieben. Dies gilt als Quittung für die Steuerung, daß das Terminal die Meldung abgeholt hat. Die nächste serielle Übertragung von Meldungen kann stattfinden.

Wie wird eine neue Meldung erkannt?

Das Terminal signalisiert dem Bediener durch Blinken der Hilfetaste oder eine blinkende Funktionstaste, daß eine neue Meldung eingegangen ist. Diese optische Mitteilung erfolgt nur wenn ein Grenzwert (Meldungspriorität für Direktanzeige) unterschritten wurde.

Wie wird die Meldemaske konfiguriert?

Vor die Meldungsmaske wird eine E/A-Maske vorgeschaltet, die Konfigurationsmaske, in ihr können durch verschiedene Systemvariablen die Ausgabeformate bestimmt werden.

Wie kann die aktuellste Meldung in einer beliebigen E/A-Maske ausgegeben werden?

Es steht eine Systemvariable **RepoutRepText** zur Verfügung, die immer die zuletzt eingegangene Meldung oder die mit der höchsten Priorität enthält. Der Inhalt bleibt bis zur nächsten Änderung erhalten. Die Ausgabe erfolgt immer linksbündig und wie im Konfigurationsmenü festgelegt. Daneben gibt es für die vollständige oder mehrzeilige Ausgabe noch die Systemvariablen:

RepoutRepText21

RepoutRepText41

RepoutRepText61

siehe dazu die Beschreibung der Systemvariablen.

Wo werden die Meldungen angezeigt?

In der Meldungsmaske. Die Meldungsmaske kann über die Knotenmaske, E/A-Maske, Steuertasten oder Softkeys erreicht werden.

Aufgrund der Verwendung von vielen Steuertasten innerhalb der Meldungsmaske kann diese nur über die Home-Taste oder Funktionstasten wieder verlassen werden.

3.9.2.4 Paralleles Meldesystem (Zustandsmeldungen)

Das parallele Meldungssystem ergänzt das serielle Meldesystem. Die Meldungen werden parallel übertragen und im Terminal ausgewertet. Der aktuelle Meldungsstatus wird dabei mit dem Vorzustand im Terminal verglichen. Die nicht mehr vorhandenen Meldungen werden aus dem Speicher automatisch gelöscht, die neuen Meldungen zusätzlich in den Speicher aufgenommen. Es kann der aktuelle Status der Meldungen ausgegeben werden.

Alle Meldungen sind mit Datum und Uhrzeit versehen, so daß festgestellt werden kann, wann eine Meldung erstmals aufgetreten ist.

Die Länge des Meldungspuffers darf maximal

- 64 Datenwörter oder 128 Byte (TSdos) oder

- 256 Byte (TSwin) betragen.

Die Länge ist in der Projektiersoftware in der Parameterdatei (TSdos), bzw. in den Systemparametern für das Meldungssystem (TSwin) einzustellen. Bei der Länge können sich, je nach Protokoll, Einschränkungen ergeben (siehe dazu Kapitel Steuerungs- und Busanpassungen).

Zustandsmeldungen bleiben im Meldespeicher nur solange erhalten, wie sie durch die Steuerung gemeldet werden.

Zustandsmeldungen können zeit- und/oder ereignisgesteuert übermittelt werden.

3.9.2.4.1 Anzahl Bytes für Zustandsmeldungen

Dieser Parameter gibt die Anzahl der zu übertragenden Bytes beim parallelen Meldesystem an. Die Größe richtet sich nach der Anzahl der Zustandsmeldungen. Die absolute Größe ist auch vom definierten Datentyp (Adresse) abhängig. Wurde hier beispielsweise eine Wortadresse angegeben und die Anzahl der Bytes ist ungerade, so wird automatisch die Anzahl aufgerundet.

3.9.2.4.2 Abbild der Zustandsmeldungen

Bei den Zustandsmeldung wird hier die Anfangsadresse des Speicherbereichs eingetragen, der die Information der Meldungen enthält. Der symbolische Name wird einer festen Adresse zugewiesen. Die Zuordnung der Meldungsnummer zur Bit-Information ist vom Datentyp der Steuerungsvariablen abhängig. Da in den unterstützten Steuerungen die Abbildungen von Byte, Wort und LWort nicht identisch sind, wurde der Datentyp protokollspezifisch bearbeitet. Wichtig ist hier, daß eine einmal festgelegte Zuordnung nicht wieder geändert wird!

Zuordnung byteorientiert:

Byteadresse + 0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byteadresse + 1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byteadresse + 2	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byteadresse + 3	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
:	:	:	:	:	:	:	:	:
Byteadresse + n	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Meldung 1 = Byte 0, Bit 0
 Meldung 2 = Byte 0, Bit 1
 Meldung 3 = Byte 0, Bit 2
 :
 Meldung 8 = Byte 0, Bit 7
 Meldung 9 = Byte 1, Bit 0
 Meldung 10 = Byte 1, Bit 1

Zuordnung wortorientiert

Wortadresse + 0	Bit 15	Bit 14	Bit 13	Bit 12.....Bit 3	Bit 2	Bit 1	Bit 0
Wortadresse + 1	Bit 15	Bit 14	Bit 13	Bit 12.....Bit 3	Bit 2	Bit 1	Bit 0
Wortadresse + 2	Bit 15	Bit 14	Bit 13	Bit 12.....Bit 3	Bit 2	Bit 1	Bit 0
Wortadresse + 3	Bit 15	Bit 14	Bit 13	Bit 12.....Bit 3	Bit 2	Bit 1	Bit 0
:	:	:	:	:	:	:	:
Wortadresse + n	Bit 15	Bit 14	Bit 13	Bit 12.....Bit 3	Bit 2	Bit 1	Bit 0

Meldung 1 = Wort 0, Bit 0
 Meldung 2 = Wort 0, Bit 1
 Meldung 3 = Wort 0, Bit 2
 :
 Meldung 16 = Wort 0, Bit 15
 Meldung 17 = Wort 1, Bit 0
 Meldung 18 = Wort 1, Bit 1

Für andere Datenformate gelten entsprechend die gleichen Regeln.

3.9.2.4.3 Zeitgesteuerte Übertragung der Zustandsmeldung

Die Übertragung der parallelen Meldungen wird zeitgesteuert aus dem Terminal oder ereignisgesteuert aus der Steuerung aktiviert. Der komplette Meldepuffer wird auf einmal übertragen.

Die beiden Übertragungsverfahren dürfen nebeneinander aktiviert werden. Dadurch können größere Pollzeiten (alle 5 bis 10 s) vorgegeben werden. Bei kritischen oder wichtigen Meldungen kann dann zusätzlich ereignisgesteuert übertragen werden. Die Zustandsmeldungen werden nach Ablauf der Pollzeit automatisch aus der Steuerung angefordert. Die Pollzeit sollte nicht zu kurz gewählt werden, da die Übertragung der Zustandsmeldungen, je nach Anzahl und Protokoll, recht zeitintensiv sind.

Wird die Pollzeit auf 0,0 s gesetzt, findet keine zeitgesteuerte Übertragung statt.

3.9.2.4.4 Ereignisgesteuerte Übertragung der Zustandsmeldung

Die ereignisgesteuerte Übertragung wird aus der Steuerung aktiviert. Die Steuerung schreibt auf den seriellen Datenkanal den Ereigniscode **7FFF_H**. Dieser Code bewirkt einen Updatezyklus auf das parallele Meldesystem. Das Bedienterminal hat dann die aktuellen Meldungen im parallelen Meldungspuffer.

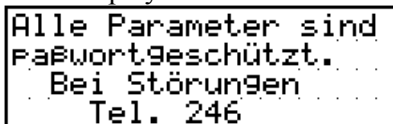
Stehen in der Parameterdatei der symbolische Variablenname und die Länge der Zustandsmeldungen, dann kann ereignisgesteuert übertragen werden (auch bei Pollzeit 0,0 s)

3.10 Hilfesystem

Jeder Maske und jeder editierbaren Variablen kann ein Hilfetext von der Größe einer Displayseite zugeordnet werden. Ist kein spezieller Text projektiert, wird ein Standardtext (Default-Hilfetext) ausgegeben. Der Default-Hilfetext kann im Projektiersystem vom Anwender frei gestaltet werden.

3.10.1 Default-Hilfetext

Wird zu einer bestimmten Maske kein Hilfetext definiert, so wird der im Projektiersystem angegebene Default-Hilfetext ausgegeben. Dieser Text ist immer vorhanden, im unprogrammierten Zustand als leere Displayseite!



```
Alle Parameter sind
paßwortgeschützt.
Bei Störungen
Tel. 246
```

Bild 28: Beispiel einer Default-Hilfemaske

3.10.2 Hilfetext zu Masken

Zu jeder projektierten Maske kann eine Hilfemaske erstellt werden. In den Maskenparametereinstellungen wird die aktuelle Maske mit einer Hilfemaske verknüpft.

Für die Hilfemaske können alle zulässigen Zeichen verwendet werden. Die Hilfetexte können generell über die Hilfetaste direkt aufgerufen werden. Solange keine Datenfreigabe angefordert ist, erscheint der Hilfetext zur Maske. Sowie die Datenfreigabe angefordert wird, erscheint der Hilfetext zur Variablen.

3.10.2.1 Hilfetext zur Meldungsmaske

Zu Meldungsmasken können ebenfalls Hilfemasken erstellt werden. Hier wird aber nur ein Hilfetext zur Maske angeboten. Für die eventuell eingebetteten Variablen ist keine Verknüpfung mit einer Hilfemaske möglich.

3.10.3 Hilfetext zu Variablen

Hier gilt die gleiche Funktionalität, wie bei Hilfetext zu Masken. Speziell wurde hier an die Ausgabe des zulässigen Wertebereichs gedacht. Die Hilfemaske wird in den Variablenparametern mit der Variablen verknüpft.

3.11 Funktionstasten

Neben den Masken spielen die Funktionstasten und deren LEDs im TesiMod-Bedienkonzept eine wichtige Rolle. Funktionstasten lassen sich vom Anwender projektieren. Sie können als Direktanwahltasten für Maskenwechsel oder als Steuertasten für die Maschine verwendet werden. Bei der Verwendung als Steuertasten dienen die integrierten LEDs als Funktionsrückmeldung.

Sind Funktionstasten als Direktanwahltasten projektiert so ermöglichen sie einen schnellen und direkten Zugang zu Masken oder komplette Menü-Strukturen.

Ist das Bedienterminal mit parallelen Ausgängen ausgestattet, können acht beliebige Funktionstasten diese Ausgänge direkt ansteuern. Die Reaktionszeit auf den Tastendruck beträgt ca. 30 ms. Der Ausgang wird erst zur Verfügung gestellt, wenn die Taste nach der Entprellung sicher als "betätigt" erkannt wird.

In der Projektiersoftware kann die Kombination Direkttasten und gleichzeitig steuernde Funktion für Funktionstasten und Softkeys projiziert werden. In dieser Betriebsart sollten nur die Startflanken der Tasten ausgewertet werden, denn je nach Länge des Tastendrucks und zugeordneter Maske könnte sich der Stop-Code bereits geändert haben!

3.11.1 Direktanwahltasten

Als Direktanwahltasten werden Funktionstasten bezeichnet, die für den direkten Aufruf einer Maske projiziert wurden. Beim Betätigen der Funktionstaste wird ein Maskenwechsel veranlaßt.

Der Maskenwechsel ist **nicht** möglich, wenn in einer Maske ohne automatische Datenfreigabe die Datenfreigabe angefordert wurde (Zustands-LED der Taste Datenfreigabe blinkt oder leuchtet).

Durch Direktanwahltasten kann die Bedienung im System wesentlich beschleunigt werden.

3.11.2 Funktionstasten der Steuerung

Neben der Belegung von Funktionstasten als Direktanwahltasten können diese auch eine Funktion in der SPS auslösen. Dazu wird in der Anwenderbeschreibung anstelle eines Maskenwechsels der Funktionstaste der symbolische Name einer Steuerungsvariablen zugeordnet. Die Taste kann dabei die Variable beim Betätigen "setzen" oder "rücksetzen", die gleiche Funktionalität ist ihr beim Loslassen zugeordnet. Wird der Taste die Funktion "setzen" zugeordnet, dann wird der Datentyp mit dem eingetragenen Wert besetzt. Das heißt:

Ist als Wert die Ziffer 1 eingetragen, erhält ein

- Merkerbit logisch "1"
- Merkerbyte den Wert 01_H
- Merkerwort den Wert 0001_H
- Doppelwort den Wert 00000001_H

Für höhere Werte als „1“ muß für die Variable mindestens eine Byteadresse angegeben werden. Höhere Werte als „1“ können nur bei TSwIn zugewiesen werden.

Ist als Wert die Ziffer 3 eingetragen, erhält ein

- Merkerbyte den Wert 03_H
- Merkerwort den Wert 0003_H
- Doppelwort den Wert 00000003_H

3.11.3 Softkeytasten

Softkeys sind im TesiMod Standard-Mode Funktionstasten, die je nach Maske eine andere Funktion erhalten. Die gerade aktuelle Funktion einer Softkeytaste wird in der aktuellen Maske beschrieben. Dazu können Bilder, Hintergrundbilder, Auswahlbilder, statische Texte und Auswahltexte (kodierter Text) verwendet werden.

Falls die Beschriftung der Softkeys durch einen Auswahltext (kodierten Text) dargestellt wird, dann kann die Funktionstaste auch innerhalb einer Maske für mehrere Funktionen verwendet werden.

Die auszuführende Aktion wird in der Steuerung aus der Verknüpfung von

- Maskennummer
- Nummer des Auswahltextes (kodierter Text)
- Variablenwert, der mit der Softkeytaste übergeben wird bestimmt.

Je nach Terminal können unterschiedlich viele Tasten als Softkeys verwendet werden.

Beispiel: Ein Softkey (F1) soll in Maske 10 eine Pumpe ein- und ausschalten können

1. Textliste (Pumpe) mit zwei Einträgen erstellen

Wert	Text
0	PUMPE AUSSCHALTEN
1	PUMPE EINSCHALTEN

2. Variablen definieren

- Softkeybeschriftung M 100.0
- Softkeystatus M 100.1
- Maskenabbild MW 110

3. Maske erstellen (Nummer 10)

- neben oder über einer Funktionstaste eine Steuerungsvariable (M 100.0) einrichten:
Auswahltextvariable für zyklische Ausgabe und mit der Textliste (Pumpe) verknüpfen
- Funktionstaste F1 der Maske mit der Variablen Softkeystatus (M 100.1) verknüpfen
(setzen/rücksetzen)

4. SPS-Programm erstellen: Pumpe soll mit A32.0 geschaltet werden

- Maskennummer (MW 110) auswerten (muß den Wert 10 haben)
- Flankenauswertung für M 100.1 erstellen
- ELTACO-Funktion für Pumpenausgang A 32.0 erstellen
- Merker M 100.0 auf Wert 0 setzen, wenn Pumpe eingeschaltet ist
- Merker M 100.0 auf Wert 1 setzen, wenn Pumpe ausgeschaltet ist

3.11.3.1 Reaktionszeit von Funktions- und Softkeytasten

Sobald Funktionstasten SPS-Variablen beeinflussen müssen, werden diese über das Protokoll mit höchster Priorität übertragen. Bei der Übertragung treten protokollspezifische Reaktionszeiten im Bereich von 60 bis 120 ms auf. Diese Zeit vergeht zwischen dem Betätigen der Taste, bis in der SPS ein Ausgang gesetzt oder rückgesetzt wird. Die Reaktionszeit ist vom Protokoll selbst, der Protokollbelastung (zyklische Daten etc) und der Zykluszeit der SPS abhängig.

Es soll nochmals darauf hingewiesen werden, daß sich die Reaktionszeit durch die Pollzeiten der Variablen, Meldungen und vom Abbild der LEDs beeinflussen läßt.

3.11.3.2 Steuertasten als Funktionstasten

Die Steuertasten können alternativ als Funktionstasten Reaktionen an der SPS auslösen. Dabei werden sie funktionell wie Funktionstasten behandelt, das heißt sie können einer Variable den Wert 1 -setzen- oder 0 -rücksetzen- (TSdos), bzw beliebige Werte (TSwin) zuweisen. Die Übertragung geschieht unabhängig von der Belegung der Maskenparameter. Falls also die Steuertasten in einer Maske eine besondere Bedeutung haben sollen, dann ist es sinnvoll, diese nicht gleichzeitig für Maskenwechsel zu projektieren. Die maskenbezogene Auswertung ist identisch zur Funktionstaste.

3.11.4 Funktionstasten steuern Parallelausgänge

Jeweils 8 wahlfreie Funktionstasten können mit Parallelausgängen (Halbleiterausgänge) verknüpft werden. Die Tasten werden per Software eingelesen, entprellt und dann auf den Ausgängen abgebildet. Die Reaktionszeit auf die Ausgänge beträgt ca. 30 ms. Die Option Parallelausgänge hat für den Anwender den großen Vorteil daß die Tasten sehr schnell und protokollunabhängig der SPS mitgeteilt werden können. Die ausgewählten Tasten sind ideal geeignet zur Steuerung von Achsen oder Realisierung des Tippbetriebs!

Mit der Ausgangsleistung können direkt SPS-Eingänge angesteuert werden.

Wurde für die Funktionstaste neben dem Ausgang auch eine Variable der SPS programmiert, so wird diese natürlich zusätzlich zum Ausgang ebenfalls mitgeführt, allerdings mit einem kleinen zeitlichen Versatz.

3.11.5 Zustands-LEDs der Funktionstasten

Für jede Zustands-LED einer Funktionstaste wird im zyklischen Pollbereich eine 2 Bit Information zur Verfügung gestellt. Jeweils ein Bit dient dazu, die betreffende Zustands-LED ein- bzw. auszuschalten, das andere Bit stellt das Blink-Attribut der Zustands-LED dar. Die Zustands-LEDs können nur von der Steuerung beeinflusst werden.

Bit ein/aus	Bit blinken	Zustand der LED
0	0	LED aus
1	0	LED ein
1	1	LED blinkt
0	1	LED aus, Bit blinken darf gesetzt bleiben

Eine Ausnahme davon entsteht, wenn

- eine Funktionstaste als Direktanwahltaste auf eine Meldungsmaske programmiert ist
- für die Meldungspriorität ein Wert größer 0 (Null) eingegeben wurde.

In diesem Fall kann die LED dieser Funktionstaste nicht von der SPS beeinflusst werden! Die Ansteuerung geschieht dann rein nur über die Meldungsfunktionalität.

Wenn das eingesetzte TesiMod Bedienterminal weniger Zustands-LEDs hat, als hier angesteuert werden können, so sind die überzähligen Bits ohne Bedeutung.

Die Anordnung der Bits für die einzelnen Zustands-LEDs ist dem Abschnitt vom zyklischen Pollbereich zu entnehmen.

Um die Übertragungszeiten zu reduzieren, sollte die Länge vom Pollbereich so gewählt werden, daß nur die notwendigen Bytes für Zustands-LEDs übertragen werden.

3.12 Systemparameter

Alle Systemparameter sind mit Grundwerten voreingestellt.

Sie werden mit dem erstellten Projekt zusammen in das Bedienterminal geladen. Mit den Systemparametern werden Werte für

- die allgemeinen Parameter
 - den Pollbereich
 - die Terminaluhr
 - die Betriebsstundenzähler
 - das Meldungssystem
 - den Variantenpuffer
 - die Paßwortverwaltung
 - die Druckerschnittstelle
 - den Gateway
 - den Datensatztransfer
 - die parallelen Ausgänge (optional)
- eingestellt.

3.12.1 Systemparameter Allgemeine Parameter

Die allgemeinen Parameter beziehen sich auf allgemein gültige Funktionen des Bedienterminals.

- Pollzeit für zyklische Variablen
- Einstellungen für den Bildschirmschoner (optional)
- Automatischer Download (nur TSwin)
- Symbolische Adressen für :
 - Abbild der Maskennummer
 - Abbild des DIP-Schalters
 - Koordinierungsbyte Lesen
 - Tabellenindex
 - Tastaturabbild

3.12.2 Systemparameter Pollbereich

Als Systemparameter für den Pollbereich gelten die Symbolische Adresse der Variablen, die Pollzeit und die Größe des Pollbereichs. Bei den *EUROTHERM Reglern* muß dieser eingetragene symbolische Name die Adresse „W 0“ haben und eine Größe von 6 Byte.

3.12.3 Systemparameter Terminaluhr

Für die Terminaluhr können die Systemparameter Symbolische Adresse der Variablen, die Pollzeit, die Übertragungsparameter (Datum, Uhrzeit, Wochentag) und die Variable für das Einstellen der Terminaluhr von der Steuerungsseite spezifiziert werden.

3.12.4 Systemparameter Betriebsstundenzähler

Für die Betriebsstundenzähler können die Systemparameter Adressen für Controlbyte und Resetbyte sowie die Pollzeit angegeben werden.

3.12.5 Systemparameter Meldungssystem

Für das Meldungssystem können allgemeine Parameter angegeben werden. Dazu gehören die Größe des Meldungspuffers und die Meldungspriorität für die direkte Anzeige einer Meldung.

Das serielle Meldungssystem kann weiterhin mit den Parametern für einen Protokolldrucker und dem symbolischen Namen der Variablen, mit der Meldungen von der Steuerung aus gelöscht werden können spezifiziert.

Für das parallele Meldungssystem können die Parameter Größe des Meldungspuffers, die Pollzeit und der symbolische Name der Variablenadresse definiert werden.

Zusätzliche Parameter für das serielle, wie für das parallele Meldungssystem sind die Sortierungsweise und die Darstellungsweise der Meldungen.

3.12.6 Systemparameter Variantenpuffer

Als Systemparameter für den Variantenpuffer gelten hier der symbolische Name der Variablen und die Größe des Variantenpuffers.

3.12.7 Systemparameter Paßwortverwaltung

Die Systemparameter für die Paßwortverwaltung sind gleichzeitig die Paßwörter selbst, sowie die entsprechenden Autorisierungslevel. Außerdem werden die Initialisierungswerte für die Autorisierungslevel definiert.

3.12.8 Systemparameter Druckerschnittstelle

Für die Druckerschnittstelle werden die Parameter Baudrate, Parität, Datenbits, Stopbits und Handshake eingestellt.

3.12.9 Systemparameter Gateway

Die Einstellung der Gatewayparameter kommt nur für Bedienterminals in Frage, die mit einer entsprechenden Firmware ausgerüstet sind.

Für solche Geräte können die Parameter

- kleinste mögliche Slavenummer
 - größte mögliche Slavenummer
 - Pollzeit für Textliste
 - Cachegröße
 - Pollzeit für Cache
 - Variable für Cacheadresse
 - Variable für Netzstatusadresse
- eingestellt werden.

3.12.10 Systemparameter Datensatztransfer

Für den Datensatztransfer benötigt man Parameter für den Transfer vom Terminal an die Steuerung:

- Variable für Rezepturnummer (Bei EUROTHERM Reglern Adresse „W 12“)
- Variable für Datensatznummer (Bei EUROTHERM Reglern Adresse „W 13“)

Für den Datensatztransfer von der Steuerung (auf Anforderung):

- Variable für Rezepturnummer (Bei EUROTHERM Reglern Adresse „W 10“)
- Variable für Datensatznummer (Bei EUROTHERM Reglern Adresse „W 11“)

3.12.11 Systemparameter parallele Ausgänge

Die Systemparameter für die parallelen Ausgänge sind der symbolische Name der Variablen und die Pollzeit dieser Variablen. Jeder Ausgang kann separat freigegeben werden.

3.13 Versionsnummer

Die Versionsnummer ist für den Anwender reserviert. Gültige Werte liegen im Bereich von 0 bis 255. Der Wert wird in einer Systemvariablen abgespeichert. Diese Systemvariable kann sich der Anwender in jeder beliebigen E/A-Maske ausgeben lassen.

Eine weitere funktionelle Verknüpfung im Bedienterminal besteht nicht mehr. Ein Editieren der Versionsnummer im Online-Betrieb ist nicht möglich.

3.14 Betriebsstundenzähler

TesiMod Bedienterminals stellen dem Anwender 8 Betriebsstundenzähler zur Verfügung. In der Projektiersoftware geben Sie einen Variablennamen für das Controlbyte an, unter dessen Adresse die Steuerung die Betriebsstundenzähler beeinflussen kann.

Jedes Bit von diesem Controlbyte steht für einen Betriebsstundenzähler. Ist ein Bit auf logisch 1 gesetzt, wird der entsprechende Betriebsstundenzähler im Zyklus der eingestellten Pollzeit inkrementiert.

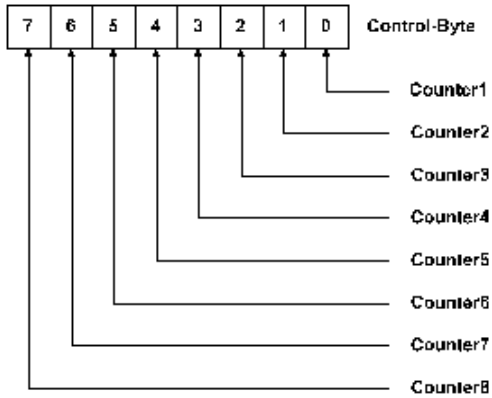


Bild 29: Aufbau Controlbyte

Beispiel: Für ein Wartungsintervall von 50 Stunden soll ein Betriebsstundenzähler eingerichtet werden

Pollzeit für Zähler: 60 Sekunden (jede Minute wird der Zähler um 1 erhöht)

Einstellung: Systemvariable **Counter1**
Festkommazahl (TSdos)
Dezimalzahl (TSwin)
4 Stellen; 1 Nachkommastelle
nur positiv
Faktor 1; Divisor 6; Summand 0

Anzeige am Terminal nach 150 Pollzyklen:
Formatierung: 150 : 6 = 25
formatierte Anzeige: 2,5 Stunden

Dieses Beispiel hat eine Genauigkeit von +/- 6 Minuten.

Mit der Variablen „Reset-Byte“ haben Sie die Möglichkeit, jeden einzelnen Betriebsstundenzähler von der Steuerung aus zurückzusetzen. Dazu müssen Sie das Bit für den entsprechenden Betriebsstundenzähler auf logisch 1 setzen.

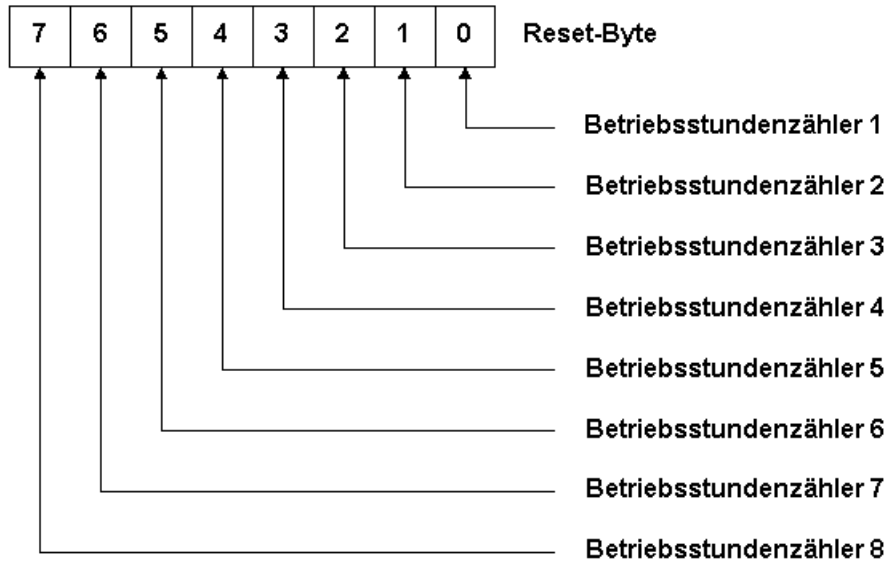


Bild 30: Aufbau Reset-Byte

3.15 Parallelausgänge

Die Parallelausgänge des Bedienterminals können direkt von den Funktionstasten und von der Steuerung aus angesteuert werden. Um die Parallelausgänge von der Steuerung aus bedienen zu können, muß eine Variable für das Steuerwort vereinbart werden. Kann die Steuerung nur byteweise zugreifen, so muß das Wort in zwei Einzelbytes aufgeteilt werden. Die Byteorder richtet sich dabei nach der verwendeten Steuerung.



Bild 31: Aufbau Steuerwort

Der Aufbau des Steuerworts zeigt, daß immer ein Bitpaar einen Ausgang steuert. Für jedes Bitpaar gilt folgende Wahrheitstabelle:

0	0	Ausgang AUS
1	0	Ausgang EIN
1	1	Ausgang BLINKT

Bild 32: Wahrheitstabelle für Parallelausgänge

In der Projektiersoftware muß zusätzlich zum Variablennamen auch die Pollzeit angegeben werden, mit der das Steuerwort zyklisch ausgelesen werden soll.

Die Ausgänge können auch mit den Funktionstasten angesteuert werden, wenn dies in der Projektiersoftware vereinbart worden ist. Ein Ausgang ist solange angesteuert (EIN), wie die Funktionstaste gedrückt wird.

Die Projektiersoftware überprüft, ob Funktionstasten und die Steuerung auf die gleichen Ausgänge zugreifen wollen.

3.16 Bildschirmschoner

In einigen Terminals ist ein Bildschirmschoner (Screen Saver) eingebaut. Dabei wird die Ausgabe auf das Display überwacht. Sobald keine Ausgabe mehr auf die Anzeige erfolgt, beginnt die Überwachungszeit zu laufen. Nachdem die Zeit abgelaufen ist, wird die Anzeige abgeschaltet und die Zustands-LED der Taste Hilfe blinkt. Durch Betätigen einer beliebigen Taste wird die Anzeige wieder aktiviert.

Die Aktivierung des Bildschirmschoners kann von der Anzeige zyklischer Variablen abhängig gemacht werden. Bei Projekten mit TSdos kann der Bildschirmschoner generell nicht aktiviert werden, wenn zyklische Daten angezeigt werden.

Die Überwachungszeit läßt sich im Raster von 0,1 Sekunden eintragen.

Wenn für die Zeit der Wert "0" eintragen wird, bleibt der Bildschirmschoner ausgeschaltet.

3.17 Abbild der Maskennummer

Die Maskennummer stellt einen maskenspezifischen Code dar, der vom Terminal in die hier eingetragene Variable der Steuerung geschrieben wird.

Das Terminal schreibt bei jedem Maskenwechsel in diese Variable die gerade aktuelle Maskennummer. Die Zuweisung der Variable erfolgt in der Variablenliste. Hier muß dem symbolischen Namen eine Wortadresse der Steuerung zugewiesen werden.

Bei den *EUROTHERM* Reglern muß dieser eingetragene symbolische Name die Adresse „W 15“ haben.

An dieser Adresse kann dann die jeweilige Maskennummer ausgelesen und weiterverarbeitet werden.

Für die Maskennummer ist immer eine 16 Bit-Variable zu reservieren.

Siehe dazu auch: Auswertung der Funktionstasten und Softkeys, etc.

3.18 Abbild des Betriebsartenschalters

Das Abbild des Betriebsartenschalters (User-Mode-Switch) wird im Standard-Mode nach der Initialisierung an die Steuerung übertragen. Freie DIP-Schalter können vom Anwender in der Steuerung ausgewertet werden. Dadurch können Sie individuelle Programme aufrufen oder in einer Serviceroutine eine Abfrage erstellen.

3.19 Terminaluhr

Jedes Bedienterminal ist mit einer Echtzeituhr ausgestattet. Die Parameter der Echtzeituhr können mit den Systemparametern eingestellt werden.

Die Echtzeituhr legt den aktuellen Zeitwert, das Datum und den Wochentag mit jedem Pollzyklus in der vereinbarten Variable in der angeschlossenen Steuerung ab.

Die angeschlossene Steuerung kann seinerseits auch die aktuelle Zeit, das Datum und den Wochentag an das Terminal übertragen.

Dies dient einmal zur Synchronisation der Uhrzeit einer eventuell in der Steuerung vorhanden Uhr oder stellt bei Steuerungen ohne Uhr die Echtzeit zur Verfügung.

Das Format und der Inhalt der Variablen müssen mit dem im Kapitel „Abbild von Datum und Uhrzeit“ identisch sein. Um das Terminal dazu anzuregen, diese Variable auszulesen, muß die Steuerung zunächst den Steuercode **7FF9H** in den seriellen Meldekanal übertragen.

3.19.1 Abbild von Datum und Uhrzeit

Die Uhrzeit, das Datum und der Wochentag werden im BCD-Format abgelegt.

Es wird eine Speichergröße von maximal 7 Byte benötigt. Die Adresse des ersten Byte wird in der Variablenliste angegeben. Der Dateninhalt ist wie folgt festgelegt:

Anfangsadresse +0 [Byte 0]	J	J	Jahr [0..99]
Anfangsadresse +1 [Byte 1]	M	M	Monat [1..12]
Anfangsadresse +2 [Byte 2]	T	T	Tag [1..31]
Anfangsadresse +3 [Byte 3]	h	h	Stunde [0..23]
Anfangsadresse +4 [Byte 4]	m	m	Minute [0..59]
Anfangsadresse +5 [Byte 5]	s	s	Sekunde [0..59]
Anfangsadresse +6 [Byte 6]	W	W	Wochentag [0..6]

Bild 33: Aufbau Steuerbyte für Uhrzeit und Datum

Die Variable für den Wochentag ist unabhängig vom Kalender und läuft immer modulo 6. Die Zuordnung der Nummer zu einem bestimmten Wochentag im Kalender legt der Anwender beim Setzen des Datums und der Definition des Tagestextes fest.

Mögliche Zuordnung in der Textliste:

- 0 Sonntag
- 1 Montag
- 2 Dienstag
- 3 Mittwoch
- 4 Donnerstag
- 5 Freitag
- 6 Samstag

Es ist nun Aufgabe des Anwenders bei der Eingabe des Datums die Variable für den Wochentag auf den richtigen Wert zu setzen.

Beispiel:

21.02.1997 (Freitag) Wochentag nach obiger Tabelle: 5

3.20 Koordinierungsbyte Lesen

Die Bezeichnung **Koordinierungsbyte Lesen** (KBL) bedeutet, daß die Steuerung dieses Byte liest. Das KBL wird nur vom Bedienterminal geschrieben!

Dieses Byte dient dem Handshake und der Datenkoordination mit der Steuerung. Das Terminal teilt dazu seinen aktuellen Status der Steuerung unter dem hier eingetragenen symbolischen Namen mit. Bei den *EUROTHERM Reglern* muß dieser eingetragene symbolische Name die Adresse „W 9“ haben. Die einzelnen Bits haben eine unabhängige Bedeutung voneinander. Der Aufbau sieht folgendermaßen aus:

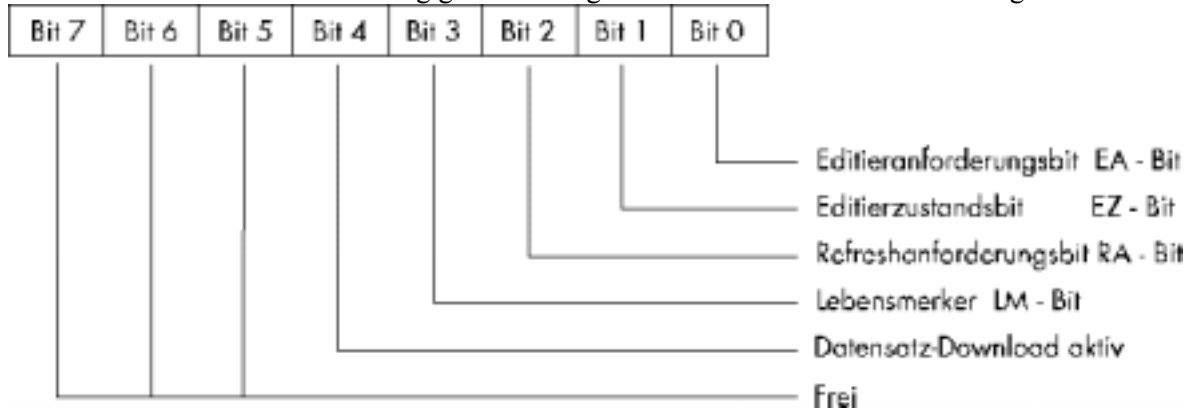


Bild 34: Aufbau des Koordinierungsbyte Lesen

Im Koordinierungsbyte Lesen wird der SPS der Status des Terminals mitgeteilt.

Die Funktionalität der einzelnen Bits sind den entsprechenden Kapiteln zu entnehmen.

Die Funktionalität <Koordinierungsbyte Lesen>, wird vom Terminal nur dann unterstützt, wenn gleichzeitig auch das **Koordinierungsbyte Schreiben** im zyklischen Pollbereich aktiviert wurde. Zur Aktivierung von beiden Koordinierungsbytes müssen deshalb bei der Projektierung die Länge des Pollbereichs, die Pollzeit und in der Variablenliste die Adressen von KBL und Pollbereich angegeben werden.

3.20.1 Editieranforderungsbit EA-Bit

Mit dem Editieranforderungsbit wird der Steuerung signalisiert, daß am Terminal der Wert einer Variablen geändert werden soll. Der Bediener drückt dazu die Taste Datenfreigabe. Solange die Steuerung die Editierfreigabe nicht erteilt hat, blinkt die Zustands-LED der Taste Datenfreigabe. Erst, nachdem die Steuerung das Bit **Externe Datenfreigabe** im **Koordinierungsbyte Schreiben** auf logisch 1 gesetzt hat, leuchtet die Zustands-LED der Taste Datenfreigabe und der Wert kann vom Bediener geändert werden.

3.20.2 Editierzustandsbit EZ-Bit

Das Editierzustandsbit wird vom Terminal automatisch auf logisch 1 gesetzt, nachdem von der Steuerung die Editierfreigabe erteilt wurde. Sobald der Bediener die Taste Dateneingabe (Enter) drückt, wird das Bit wieder auf logisch 0 zurückgesetzt.

3.20.3 Refreshanforderungsbit RA-Bit

Die Refreshanforderung wird durch eine Eingabevariable mit dem Attribut SPS-Handshake eingerichtet. Die Datenfreigabe für die nächste Eingabevariable, in der gleichen Maske, wird dann, bis zur Refreshquittierung durch die Steuerung, zurückgehalten.

Die Steuerung kann dadurch auf Anfrage (RA-Bit =1) den Wert der nächsten Variable auffrischen und dann erst die Refreshquittierung senden (RQ-Bit =1 im KBS), damit dem Bediener der aktuelle Wert vorliegt.

3.20.4 Lebensmerker LM-Bit

In einigen Kommunikationsprotokollen ist es nicht möglich die Funktionsfähigkeit der Schnittstelle auf der Steuerungsseite zu kontrollieren. Aus diesem Grund wurde die Funktion des "Lebensmerkers" eingeführt. Eine einfache Funktionalität, die sich in der Praxis als sehr wirkungsvoll erwiesen hat.

Immer wenn die SPS wissen will, ob die Verbindung noch besteht, schreibt sie in das Bit 3 vom **Koordinierungsbyte Schreiben** (KBS) eine logische "1", später eine logische "0".

Das Bedienterminal überwacht ständig den Lebensmerker im **KBS** und vergleicht ihn mit dem Zustand des Lebensmerkers im **Koordinierungsbyte Lesen** (KBL).

Sobald eine Differenz auftritt kopiert das Bedienterminal das **LM-Bit** aus dem **KBS** in das **KBL**.

Die Steuerung hat nun die Aufgabe, innerhalb einer Timeoutzeit ebenfalls zu kontrollieren, ob beide Zustände gleich sind. Bei der Festlegung der Timeoutzeit sind die Übertragungszeiten und Pollzeiten zu berücksichtigen.

3.20.5 Datensatz-Download aktiv DDA-Bit

Solange ein Datensatz transferiert wird, hat das Bit **Datensatz-Download aktiv** den Zustand logisch 1. Nachdem alle Daten gesendet wurden, wird das Bit wieder zurückgesetzt und die Steuerung kann mit den neuen Werten in der Rezeptur arbeiten.

3.21 Koordinierungsbyte Schreiben

Die Bezeichnung **Koordinierungsbyte Schreiben** (KBS) bedeutet, daß die Steuerung dieses Byte beschreibt.

Das KBS wird nur vom Bedienterminal gelesen.

Dieses Byte dient zusammen mit dem **Koordinierungsbyte Lesen** dem Handshake und der Datenkoordination mit der Steuerung. Die Steuerung teilt hier ihren aktuellen Status dem Terminal mit.

Die einzelnen Bits sind unabhängig voneinander. Das **KBS** ist das erste Byte des zyklischen Pollbereichs.

Der Aufbau sieht folgendermaßen aus:

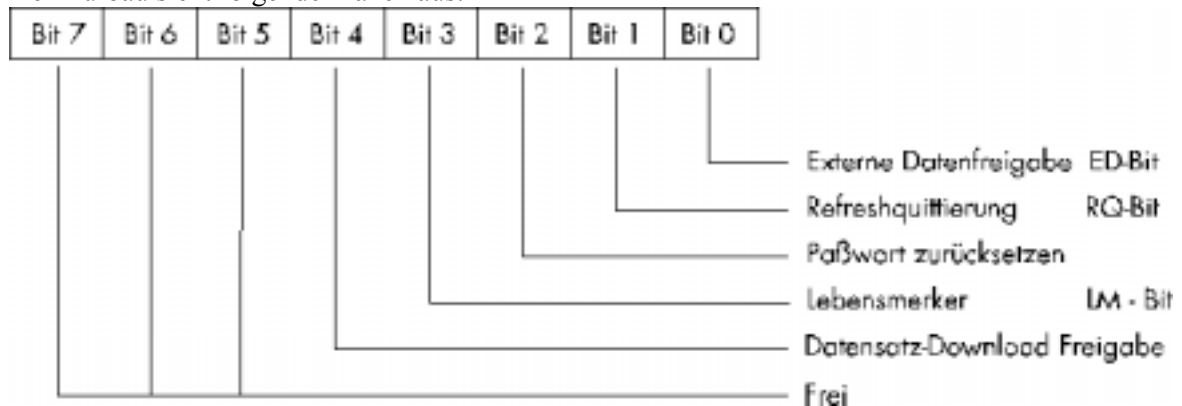


Bild 35: Aufbau vom **Koordinierungsbyte Schreiben**

3.21.1 Externe Datenfreigabe ED-Bit

Mit Hilfe der externen Datenfreigabe kann von der Steuerungsseite der Zeitpunkt bestimmt werden, ab wann am Terminal der Wert einer Variablen geändert werden darf. Solange keine externe Datenfreigabe erfolgt ist, blinkt die Datenfreigabe-LED am Bedienterminal. Der Wert kann noch nicht geändert werden. Sobald das Datenfreigabebit auf logisch 1 gesetzt wird, leuchtet die Datenfreigabe-LED und der Bediener kann einen neuen Wert eingeben. Das Bedienterminal setzt nach dem Editiervorgang das Editierzustandsbit wieder auf logisch 0 zurück und signalisiert damit das Ende des Editiervorgangs.

3.21.2 Refreshquittierung RQ-Bit

Mit dem Refreshanforderungsbit wird der Steuerung mitgeteilt, daß in der aktuellen Maske die Werte der Variablen aktualisiert werden sollen. Nach erfolgreicher Übertragung der Werte in das Terminal setzt die Steuerung das Refreshquittierungsbit auf logisch 1, um anzuzeigen, daß der Vorgang abgeschlossen ist (vgl Kapitel 3.20.3 *Refreshanforderungsbit RA-Bit*).

3.21.3 Paßwort zurücksetzen

Sowie der Bediener eine Sperrebene verläßt, die mit einem Paßwort geschützt wird, sollte der Paßwortschutz wieder aktiviert werden. Dies kann von der Steuerung erzwungen werden, indem das Bit **Paßwort zurücksetzen** auf logisch 1 gesetzt wird.

3.21.4 Lebensmerker LM-Bit

Siehe Kapitel Lebensmerker vom **Koordinierungsbyte Lesen** (3.21.4).

3.21.5 Datensatz-Download Freigabe DDF-Bit

Mit Hilfe vom DDF-Bit kann die Steuerung den Zeitpunkt bestimmen, ab dem ein Datensatztransfer an die Steuerung gestartet werden darf.

3.22 *Der zyklische Pollbereich*

Zusätzlich zum wahlfreien Schreib- und Lesezugriff auf Steuerungsvariablen wird in der Anwenderbeschreibung ein Speicherbereich von 23 Byte als zyklischer Pollbereich definiert.

Der zyklische Pollbereich besteht aus:

- 1 Byte **Koordinierungsbyte Schreiben** (KBS)
- 2 Byte **serieller Meldekanal** (Low- und Highbyte)
- einer terminalspezifischen Anzahl von **Steuerbytes für die Zustands-LEDs** der Funktionstasten

Der Pollbereich wird von der Steuerung beschrieben und vom Bedienterminal zyklisch gelesen. Dabei können von der Steuerung Aktionen, die mit dem Terminal zusammenhängen, ausgelöst, kontrolliert, freigegeben oder gesperrt werden. Darüber hinaus kann mit dem Lebensmerker vom **Koordinierungsbyte Schreiben** kontrolliert werden, ob die Verbindung zum Terminal noch besteht. Über den seriellen Meldekanal werden serielle Meldungen an das Bedienterminal gemeldet oder Aktionen ausgelöst (zum Beispiel: Datensätze transferieren, Maskenwechsel). Mit den Bits der Steuerbytes können die Zustands-LEDs der Funktionstasten ein- bzw ausgeschaltet oder blinkend geschaltet werden.

Randbedingungen des Speicherbereichs für den Pollbereich sind:

- Die SPS greift bei KBS und Zustands-LEDs bitweise zu und beim seriellen Meldekanal byte- oder wortweise
- Das Terminal greift byte- oder wortweise zu
- Der Speicherbereich muß lückenlos sein

Der symbolische Name für den zyklischen Pollbereich wird im Projektiersystem bei den Parametern des Pollbereichs angegeben. Die Zuweisung der Anfangsadresse für den Bereich wird in der Variablenliste eingetragen.

Es ist zu beachten, daß der Zugriff auf die Byte- und Wortstruktur nicht identisch sind. Eine einmal getroffene Wahl sollte beibehalten werden.

3.22.1 Byteorientiert

Wurde in der Variablenliste für den zyklischen Datenbereich die Zuordnung zu Bytes gewählt, dann ergibt sich folgende Aufteilung des Datenbereichs.

Beispiel: Der zyklische Pollbereich wird im Projektiersystem auf Merkerbyte MB 12 gesetzt.

Der Zugriff in der SPS erfolgt über:

Byteadresse	MB	Beschreibung
Byteadresse +0	MB12	Koordinierungsbyte Schreiben
Byteadresse +1	MB13	Meldekanal-Lowbyte
Byteadresse +2	MB14	Meldekanal-Highbyte
Byteadresse +3	MB15	Zustands-LEDs 1 bis 4
Byteadresse +4	MB16	Zustands-LEDs 5 bis 8
Byteadresse +5	MB17	Zustands-LEDs 9 bis 12
Byteadresse +6	MB18	Zustands-LEDs 13 bis 16
Byteadresse +7	MB19	Zustands-LEDs 17 bis 20
Byteadresse +8	MB20	Zustands-LEDs 21 bis 24
Byteadresse +9	MB21	Zustands-LEDs 25 bis 28
Byteadresse +10	MB22	Zustands-LEDs 29 bis 32

Der Datenbereich ist 11 Byte groß.

Im Einzelnen ist der byteorientierte zyklische Pollbereich folgendermaßen aufgebaut:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byteadresse + 0	frei	frei	frei	DDF	LM	PL	RQ	ED
Byteadresse + 1	serieller Meldekanal Lowbyte							
Byteadresse + 2	serieller Meldekanal Highbyte							
Byteadresse + 3	LED1	LED1	LED2	LED2	LED3	LED3	LED4	LED4
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 4	LED5	LED5	LED6	LED6	LED7	LED7	LED8	LED8
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 5	LED9	LED9	LED10	LED10	LED11	LED11	LED12	LED12
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 6	LED13	LED13	LED14	LED14	LED15	LED15	LED16	LED16
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 7	LED17	LED17	LED18	LED18	LED19	LED19	LED20	LED20
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 8	LED21	LED21	LED22	LED22	LED23	LED23	LED24	LED24
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 9	LED25	LED25	LED26	LED26	LED27	LED27	LED28	LED28
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 10	LED29	LED29	LED30	LED30	LED31	LED31	LED32	LED32
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 11	LED33	LED33	LED34	LED34	LED35	LED35	LED36	LED36
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 12	LED37	LED37	LED38	LED38	LED39	LED39	LED40	LED40
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 13	LED41	LED41	LED42	LED42	LED43	LED43	LED44	LED44
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Byteadresse + 14	LED45	LED45	LED46	LED46	LED47	LED47	LED48	LED48
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken

Bild 36: Byteorientierter Pollbereich

3.22.2 Wortorientiert

Der wortorientierte zyklische Pollbereich hat folgende Struktur:

	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Wortadresse + 0	frei	frei	frei	DDF	LM	PL	RQ	ED	frei	frei	frei	frei	frei	frei	frei	frei
Wortadresse + 1	serieller Meldekanal Highbyte								serieller Meldekanal Lowbyte							
Wortadresse + 2	LED1	LED1	LED2	LED2	LED3	LED3	LED4	LED4	LED5	LED5	LED6	LED6	LED7	LED7	LED8	LED8
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Wortadresse + 3	LED9	LED9	LED10	LED10	LED11	LED11	LED12	LED12	LED13	LED13	LED14	LED14	LED15	LED15	LED16	LED16
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Wortadresse + 4	LED17	LED17	LED18	LED18	LED19	LED19	LED20	LED20	LED21	LED21	LED22	LED22	LED23	LED23	LED24	LED24
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Wortadresse + 5	LED25	LED25	LED26	LED26	LED27	LED27	LED28	LED28	LED29	LED29	LED30	LED30	LED31	LED31	LED32	LED32
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Wortadresse + 6	LED33	LED33	LED34	LED34	LED35	LED35	LED36	LED36	LED37	LED37	LED38	LED38	LED39	LED39	LED40	LED40
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken
Wortadresse + 7	LED41	LED41	LED42	LED42	LED43	LED43	LED44	LED44	LED45	LED45	LED46	LED46	LED47	LED47	LED48	LED48
	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken	ein/aus	blinken

Bild 37: Wortorientierter Pollbereich

Beispiel: Der zyklische Pollbereich wird im Projektiersystem auf DW 21 gesetzt:

Wortadresse	DW	Highbyte	Lowbyte
Wortadresse +0	DW21	Koordinierungsbyte Schreiben	Reserviert
Wortadresse +1	DW22	Meldekanal Highbyte	Meldekanal Lowbyte
Wortadresse +2	DW23	Zustands-LEDs 1 bis 4	Zustands-LEDs 5 bis 8
Wortadresse +3	DW24	Zustands-LEDs 9 bis 12	Zustands-LEDs 13 bis 16
Wortadresse +4	DW25	Zustands-LEDs 17 bis 20	Zustands-LEDs 21 bis 24
Wortadresse +5	DW26	Zustands-LEDs 25 bis 28	Zustands-LEDs 29 bis 32

3.22.3 Abbild der Zustands-LEDs

Über das Abbild der LEDs wird die Steuerung in die Lage versetzt, die Zustands-LEDs der Funktionstasten zu steuern. Für jede Zustands-LED kann die Funktion EIN, AUS oder BLINKEN gesetzt werden. Sobald die Steuerung ein Bit setzt, wird die zugeordnete LED im Terminal entsprechend beeinflusst.

Wichtig ist hier, daß zusätzlich die Länge des Pollbereichs und die Pollzeit richtig gesetzt wurde. Ohne diese zusätzlichen Parameter können gegebenenfalls Probleme bei der LED-Ansteuerung auftreten.

Bei einer Funktionstaste, die direkt zur Meldungsmaske führt, wird die Zustands-LED vom Meldesystem beeinflusst. Damit zeigt das Meldesystem an, daß eine neue Meldung eingetroffen ist und noch nicht quittiert wurde. Um die Zustands-LED dieser Funktionstaste von der Steuerung beeinflussen zu können, muß die Meldungspriorität auf 0 (Null) gesetzt werden (vgl Kapitel 3.11.5 *Zustands-LEDs der Funktionstasten*).

0	0	Zustands-LED AUS
1	0	Zustands-LED EIN
1	1	Zustands-LED BLINKT

Bild 38: Wahrheitstabelle für Zustands-LEDs

3.22.4 Serieller Meldekanal

Der serielle Meldekanal ist Teil des zyklischen Pollbereichs und dient der Übertragung von 16 Bit-Informationen. Über diesen Datenkanal werden die Nummern serieller Meldungen, die Anwahl der Meldemasken, die externe Maskenanwahl und die Übertragung von Steuercodes ermöglicht.

Die Übertragung von Informationen läuft über folgenden Handshake ab: die SPS legt in diesem Datenwort einen Wert (> 0) ab. Nachdem der Wert an das Bedienterminal übertragen wurde, schreibt das Bedienterminal wieder den Wert 0 in dieses Datenwort. Das bedeutet für die SPS, daß der nächste Wert abgesetzt werden kann. Der Wert wird vom Bedienterminal interpretiert und deren Funktion ausgeführt.

Als Werte gelten:

- Meldungsnummern
- Maskennummern (Maskennummer + 8000_H)
- Steuercodes

3.22.5 Pollzeit

Die Pollzeit gibt an, in welchen Zeitabständen die Variable für den zyklischen Pollbereich vom Terminal gelesen werden soll. Sie wird in den Systemparametern für den Pollbereich eingestellt. Das Pollen dieser Variablen ersteckt sich gleichzeitig auf das **Koordinierungsbyte Schreiben**, den **seriellen Meldekanal** und das **Abbild der Zustands-LEDs**. Bei den meisten Protokollen haben sich Einstellungen im Bereich einer halben Sekunde bewährt. Wird die Zykluszeit zu klein gewählt, kann das Schnittstellenprotokoll den Anforderungen nicht mehr folgen, die Reaktionen werden wieder schlechter. Ein Universalrezept gibt es jedoch nicht. Die Möglichkeiten sind zu stark von der individuellen Anwenderbeschreibung abhängig. Es sollten jedoch Zeiten mindestens größer 100 ms vorgewählt werden. Eine Beratung durch unsere Hotline kann hier weiterhelfen.

3.22.6 Länge Pollbereich

Die Länge beträgt je nach Datentyp und Terminal maximal 23 Byte. Die Eingabe ermöglicht die Anpassung an den tatsächlich genutzten Bereich, falls auf das Abbild der LEDs oder einen Teil davon verzichtet werden kann. Die Grundeinstellung für alle Bedienterminals ist eine Länge von 12 Byte.

3.23 Steuercodes

In vorherigen Abschnitten wurden Steuercodes genannt, mit denen bestimmte Funktionen oder Aktionen am Bedienterminal gestartet werden können. Alle Aktionen werden von der Steuerung angefordert. Dazu muß der Steuercode in den seriellen Meldekanal innerhalb vom Pollbereich geschrieben werden.

Die Steuercodes sollen hier nochmal, nach Einsatzfall geordnet, erklärt werden.

3.23.1 Datensatz ausdrucken lassen

Mit Hilfe des folgenden Hexcodes kann von der Steuerung aus erreicht werden, daß der am Bedienterminal angeschlossene Drucker den aktuellen Datensatz ausdruckt.

Hexadezimalcode: **7FF8_H**

Vom Bedienterminal können zur Überwachung des Druckvorgangs zwei Werte an die Adresse von der Variablen für die Rezepturnummer geschrieben werden.

- 0** Datensatzausdruck OK.
- 255** Kein Ausdruck von Datensatz mit der gewünschten Datensatznummer möglich.

3.23.2 Uhr im Bedienterminal stellen

Mit Hilfe des folgenden Hexcodes kann von der Steuerung aus erreicht werden, daß die Echtzeituhr im Bedienterminal nach den Vorgaben, die in einem vereinbarten Steuerwort stehen, gestellt wird.

Hexadezimalcode: **7FF9_H**

3.23.3 Datensatztransfer von Steuerung zu Bedienterminal

Mit Hilfe des folgenden Hexcodes kann von der Steuerung aus erreicht werden, daß ein Datensatz von der Steuerung an das Bedienterminal übertragen wird. Von der Steuerung müssen vorher die Rezeptur- und die Datensatznummer angegeben werden.

Die Übertragung erfolgt blockweise.

Folgende Variablen müssen definiert sein:

- Rezepturnummer für Anforderung von Steuerung (Systemparameter - Datensatztransfer)
- Datensatznummer für Anforderung von Steuerung (Systemparameter - Datensatztransfer)

Hexadezimalcode: **7FFA_H**

3.23.4 Datensatztransfer von Bedienterminal zur Steuerung

Mit Hilfe des folgenden Hexcodes kann von der Steuerung aus erreicht werden, daß ein Datensatz vom Bedienterminal zur Steuerung übertragen wird. Von der Steuerung müssen vorher die Rezeptur- und die Datensatznummer angegeben werden.

Folgende Variablen müssen definiert sein:

- Rezepturnummer für Anforderung von Steuerung (Systemparameter - Datensatztransfer)
- Datensatznummer für Anforderung von Steuerung (Systemparameter - Datensatztransfer)

Hexadezimalcode: **7FFB_H**

3.23.5 Datensatztransfer von Steuerung zu Bedienterminal (einzeln)

Mit Hilfe des folgenden Hexcodes kann von der Steuerung aus erreicht werden, daß ein Datensatz von der Steuerung an das Bedienterminal übertragen wird.

Von der Steuerung müssen vorher die Rezeptur- und die Datensatznummer angegeben werden.

Die Übertragung erfolgt einzeln.

Folgende Variablen müssen definiert sein:

- Rezepturnummer für Übertragung vom Terminal (Systemparameter - Datensatztransfer)
- Datensatznummer für Übertragung vom Terminal (Systemparameter - Datensatztransfer)

Hexadezimalcode: **7FFD_H**

3.23.6 Meldungssystem auffrischen

Mit Hilfe des folgenden Hexcodes kann von der Steuerung aus erreicht werden, daß das Bedienterminal die neuesten parallelen Meldungen einliest.

Dadurch wird ein ereignisgesteuertes Meldungssystem realisiert.

Hexadezimalcode: **7FFF_H**

3.24 Zyklische Variablen

Unter zyklischen Variablen werden Daten verstanden, die sich während einer Maskenausgabe ständig ändern können, im weitesten Sinne ISTWERTE aller Art. Die aktuellen Werte müssen vom Terminal zyklisch aus der Steuerung angefordert werden. Die hier eingestellte Zeit gibt an, in welchem zeitlichen Abstand diese Werte einer Maske aufgefrischt werden.

Neben zyklischen Variablen wird aus der Steuerung der neue Status vom Koordinierungsbyte Schreiben, vom seriellen Meldekanal und von den LEDs aus der Steuerung angefordert (Update). Dies wird je nach Protokoll zeit- oder ereignisgesteuert vorgenommen. Bei zeitgesteuerten Protokollen wird eine Pollzeit eingetragen. Wird die Zeit auf den Wert = 0 gesetzt, findet kein Pollen statt. Ohne das Pollen dieser Variablen ist beispielsweise keine externe Steuerung der Datenfreigabe möglich.

3.25 Schnittstellenparameter X2, X3

Die im Projektiersystem eingestellten Schnittstellenparameter werden im Maskenspeicher hinterlegt. Die Daten können jedoch im Bedienterminal noch angepaßt werden. Die im Bedienterminal veränderten Werte werden im gepufferten RAM gehalten. Es besteht jedoch immer die Möglichkeit zu der Ureinstellung zurück zu kommen.

Die Parameter der Schnittstelle am Stecker X2 werden ausschließlich durch das Protokoll bestimmt. Sollten die Parameter geändert werden, können Fehlfunktionen in der Kommunikation auftreten. Untypische Werte werden beim Initialisieren vom Terminal erkannt und durch eine Meldung gekennzeichnet.

Die Parameter der Schnittstelle X3 sind ausschließlich für die Ansteuerung des Druckers und bei der Übertragung einer Hardcopy auf den PC bestimmt (Hardcopy nicht bei Bedienterminals mit 386er Prozessor). Hier müssen beide Seiten passend zueinander eingestellt werden. Für den Download gelten die eingestellten Parameter nicht, hier wird immer auf die maximal mögliche Übertragungsrate automatisch gewechselt.

3.26 Variablenbeschreibung

Die Variablenbeschreibung definiert in der Anwenderbeschreibung Format, Typ, Wertebereich, Skalierung und Attribut der Darstellung. Es sind hier alle steuerungsunabhängigen Parameter hinterlegt. Bei editierbaren Variablen können zusätzlich der Editor und die Übertragung zur Steuerung beeinflusst werden. Die Variablenbeschreibung enthält den symbolischen Namen der Variablen, die Formatbeschreibung und, falls erforderlich, Textlisten bei Auswahltextvariablen (kodierte Textvariablen).

3.26.1 Variablenformate (TSdos)

Die in einer Variablenbeschreibung einmal definierten Formatinhalte werden unter einem Formatnamen im Projektiersystem gespeichert. Dadurch ist es möglich, sehr einfach verschiedene Variablen im gleichen Format auszugeben, ohne jeweils alle Parameter wieder neu eingeben zu müssen.

3.26.2 Variablenliste

Die Variablenliste beinhaltet die Zuweisung der symbolischen Variablennamen zur Zielhardware. Durch ihre Gestaltung und Nähe zur SPS ist diese Liste immer herstellerspezifisch.

Bei TSdos sind neben der Adressierung der Variablen hier auch die protokollspezifischen Parameter, Schnittstellenparameter, Timeoutzeiten etc enthalten. Die Variablenliste wird bei TSdos als Datei mit der Dateierweiterung **.TSV** geführt.

TSwin führt die Variablenliste als Teil einer Datenbank. Es können beliebig viele Variablenlisten angelegt werden.

Die Variablenliste enthält nur die notwendigen herstellerspezifischen Anwenderbeschreibungsteile. Ihr Umfang ist deshalb in Bezug zum Gesamtprojekt gering. Durch eine Änderung der Variablenliste an andere SPS-Spezifikationen, läßt sich das Gesamtprojekt sehr schnell an verschiedene Steuerungen anpassen.

Eine Variablenliste kann im Projektiersystem auf zwei Wegen erstellt werden.

Erste Möglichkeit

TSdos:

Man beginnt mit der Eingabe der gesamten Bedienoberfläche und anschließend erstellt das Projektiersystem auf Anforderung eine protokollspezifische Variablenliste mit allen verwendeten Variablennamen. Nach Erstellung der Liste, wird diese auch als Namensliste in der Maskenprogrammierung verwendet. Die Liste enthält noch keine Information über Hardwarereferenzen der Steuerung. Dies wird vom Anwender in der herstellereigenen Schreibweise erst in der Variablenliste eingegeben. Bei dieser Variante wird sichergestellt, daß keine Variablen in der Variablenliste fehlen können.

TSwin:

Eine in einer Maske definierte Variable (mit oder ohne Adreßangabe) wird gleichzeitig in der Variablenliste eingetragen.

Zweite Möglichkeit

TSdos:

Der Anwender erstellt zuerst aufgrund seines SPS-Programms die Variablenliste mit symbolischen Namen und allen Referenzen. Hier können dann die gleichen Namen, wie in der SPS verwendet werden. Es ist dann sehr einfach, bei der Maskenerstellung den jeweiligen Variablennamen aus der Namensliste zu entnehmen.

TSwin:

Die Einträge in der Variablenliste werden zuerst erstellt, können aber auch aus jedem geeigneten Programm, über die Zwischenablage, eingefügt werden. Die Einträge stehen in TSwIn sofort global zur Verfügung.

Beide Arbeitsweisen können beliebig kombiniert werden.

Eine Alternative für Anwender von TSdos, die über entsprechende Dokumentationen oder Datenbanken ihrer SPS-Programme verfügen, besteht darin, sich die Variablenliste aus der entsprechenden SPS-Dokumentation (Druckdatei) selbst herzustellen. Der Aufbau der Liste (TSV-Datei) entspricht einer einfachen Textdatei. Die Definition ist in den Demos als Kommentar enthalten. Es muß beachtet werden, daß sich das Beschreibungsformat protokollspezifisch ändert.

Beispiel für Siemens-SPS mit PG-Anschaltung:

Mit "Co" beginnen alle Kommentarzeilen.

Auszug aus einer Variablenliste:

Co Für Siemens PG Schnittstelle

Co Vp Siemens PG Variable

Co Vp \Variablenname\,\Datentyp\,\Parameter 1\,\Parameter 2

Co oder für Siemens L1 Schnittstelle

Co Vl Siemens L1 Variable

Co Vl \Variablenname\,\Datentyp\,\Parameter 1\,\Parameter 2\,\Slave\Co

Co Datentyp : E,A,M,EB,EW,AB,AW,MB,MW,DW,DL,DR,t,z

Co Parameter 1 : Bei Datentyp DW, DL, DR Bausteinnummer, sonst

Co Byte oder Wortoffset

Co Parameter 2 : Bei Datentyp E,A,M Bitnummer 0 bis .7

Co Bei Datentyp DW,DR,DL Datenwortnummer

Co sonst keine Bedeutung

Co Slave Adresse : Nur bei L1 Protokoll

Vp \Maskennummer\,\MW\,\76\,\0\

Vp \ZyklischeDaten\,\MB\,\0\,\0\

Vp \Dipschalter\,\MB\,\120\,\0\

3.27 Anwendungsprogrammierung

Das TesiMod-Bedienkonzept basiert auf einer Maskenstruktur, die der Anwender nach seinen Bedürfnissen erstellen kann und Systemvariablen, die wichtige Funktionen im Bedienterminal beeinflussen. Dazu stehen unterschiedliche Arten von Masken zur Verfügung, um Standardaufgaben leichter lösen zu können.

Jede Maskenstruktur beginnt mit vier Masken, die vom Anwender inhaltlich beeinflusst werden können.

Bei TSwin kann jede beliebige E/A-Maske als eine Systemmasken projektiert werden. Nur für TSdos gelten die folgenden Masken unveränderlich.

- Setup-Maske (1)
- Start-Maske (2)
- Paßwort-Vorschaltmaske (3)
- Haupt-Maske (4)

Nach dem Einschalten beginnt das Bedienterminal mit der Initialisierung. Während dieser Zeit wird die Setup-Maske angezeigt. Anschließend wird für kurze Zeit die Start-Maske angezeigt. Wird dabei die Taste Dateneingabe gedrückt, springt das Bedienterminal zur Setup-Maske zurück.

Nach der Start-Maske springt das Bediensystem zur Hauptmaske (Maske 4). Ab hier beginnt das individuelle Bediensystem des Anwenders. Die Haupt-Maske ist üblicherweise mit einem Menü ausgestattet, das die Auswahl der untergeordneten Masken in Form von Menüpunkten ermöglicht (Knotenmaske).

Der Anwender hat die Möglichkeit, zu jeder Maske einen ganzseitigen Hilfetext anzugeben, der dem Bediener als Online-Hilfe über die Hilfetaste zur Verfügung steht.

Die Funktionen der Systemvariablen werden im Kapitel 3.6.3 kurz erläutert. In den folgenden Kapiteln wird auf einige Systemvariablen vorab Bezug genommen, um Teile der Anwenderprogrammierung besser erklären zu können.

3.27.1 Systemkonfiguration

Für die Projektiersoftware benötigen Sie einen PC, der den Anforderungen entspricht, die im Handbuch zur Software beschrieben sind. Eine serielle Schnittstelle des Rechners (COM1 oder COM2) wird über das Downloadkabel (ebenfalls als Zubehör erhältlich) mit der Schnittstelle X3 des Bedienterminals verbunden. Über ein weiteres, steuerungsspezifisches Schnittstellenkabel wird das Bedienterminal mit der SPS verbunden (hier an X2). Dabei ist auf den richtigen Typ des Schnittstellenkabels zu achten. Des Weiteren ist das Bedienterminal mit einer Stromquelle zu verbinden, die den Anforderungen der technischen Daten des Bedienterminals entsprechen muß.

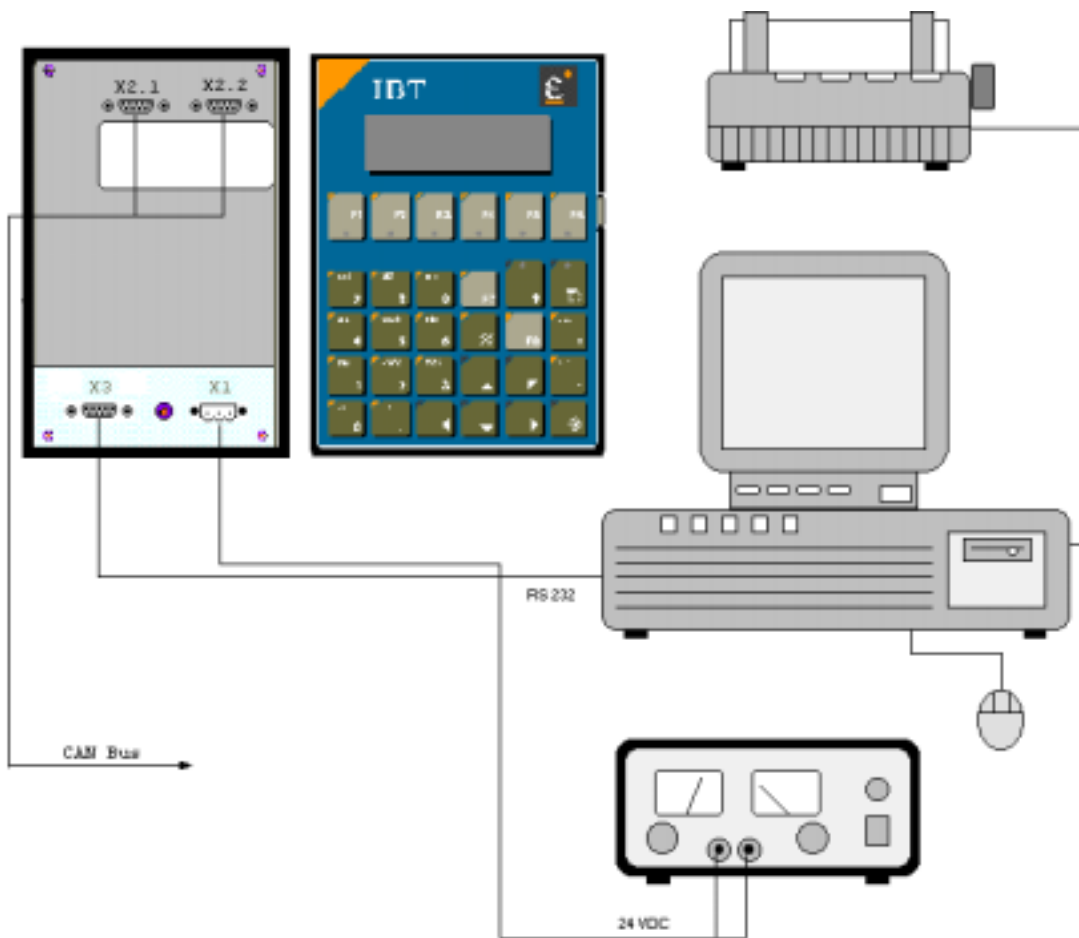


Bild 39: Systemkonfiguration

Sind alle Vorbereitungen getroffen, kann mit der Projektdurchführung begonnen werden. In dieser Konfiguration sind alle Funktionen des Bedienterminals ausführbar und prüfbar. Beim Betrieb ohne SPS (siehe Simulation ohne SPS) können alle Bedienmöglichkeiten, bis auf die Reaktionen der SPS, getestet werden.

3.27.2 Projektiersysteme TSdos und TSwIn

Mit den Projektiersystemen TSdos und TSwIn bieten wir alle Möglichkeiten, um schnell und einfach alle Möglichkeiten zu projektieren, die ein TesiMod-Bedienterminal bieten kann. Das Projektiersystem TSdos ist ein, unter dem Betriebssystem MSDOS lauffähiges, PC-Programm. Das Programm entspricht einer standardisierten SAA-Oberfläche im Textmode, die Bedienung erfolgt wahlweise über Hotkeys, Alt-Tastenkombinationen oder mit der Maus.

Im Gegensatz dazu benötigt TSwIn als Betriebssystem entweder Windows95 oder Windows NT-4.0. TSwIn bietet Windows-konforme Bedienbarkeit in Verbindung mit allen Mitteln, die Windows zur Verfügung stellen kann (OLE, COPY/CUT/PASTE, WYSIWYG). TSwIn verfügt über ein Hilfesystem, das über einen Hilfebutton oder die Funktionstaste F1 von allen Stellen des Programms erreichbar ist (Onlinehilfe).

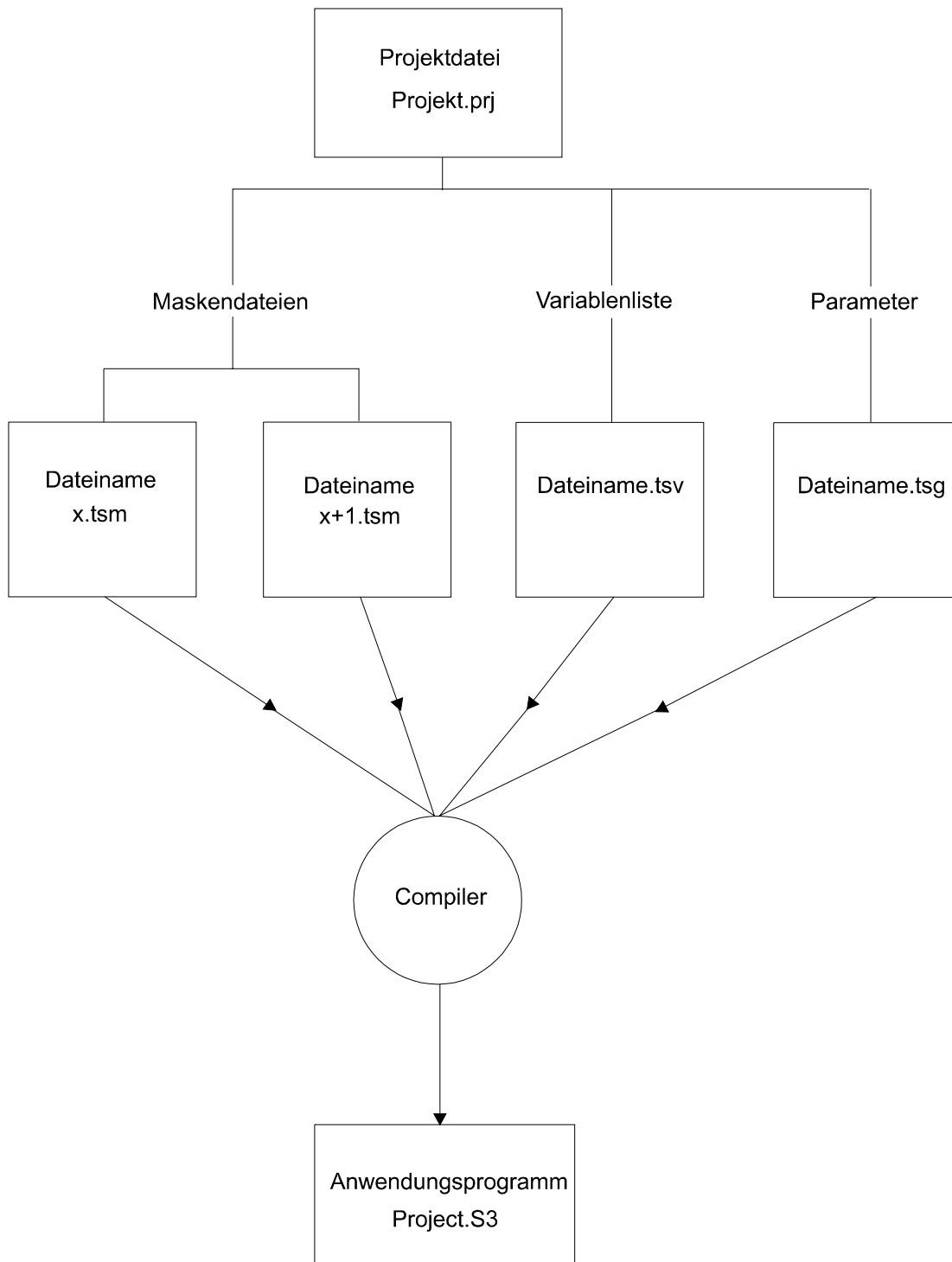


Bild 40: Projektdateien des TSdos-Projektiersystems

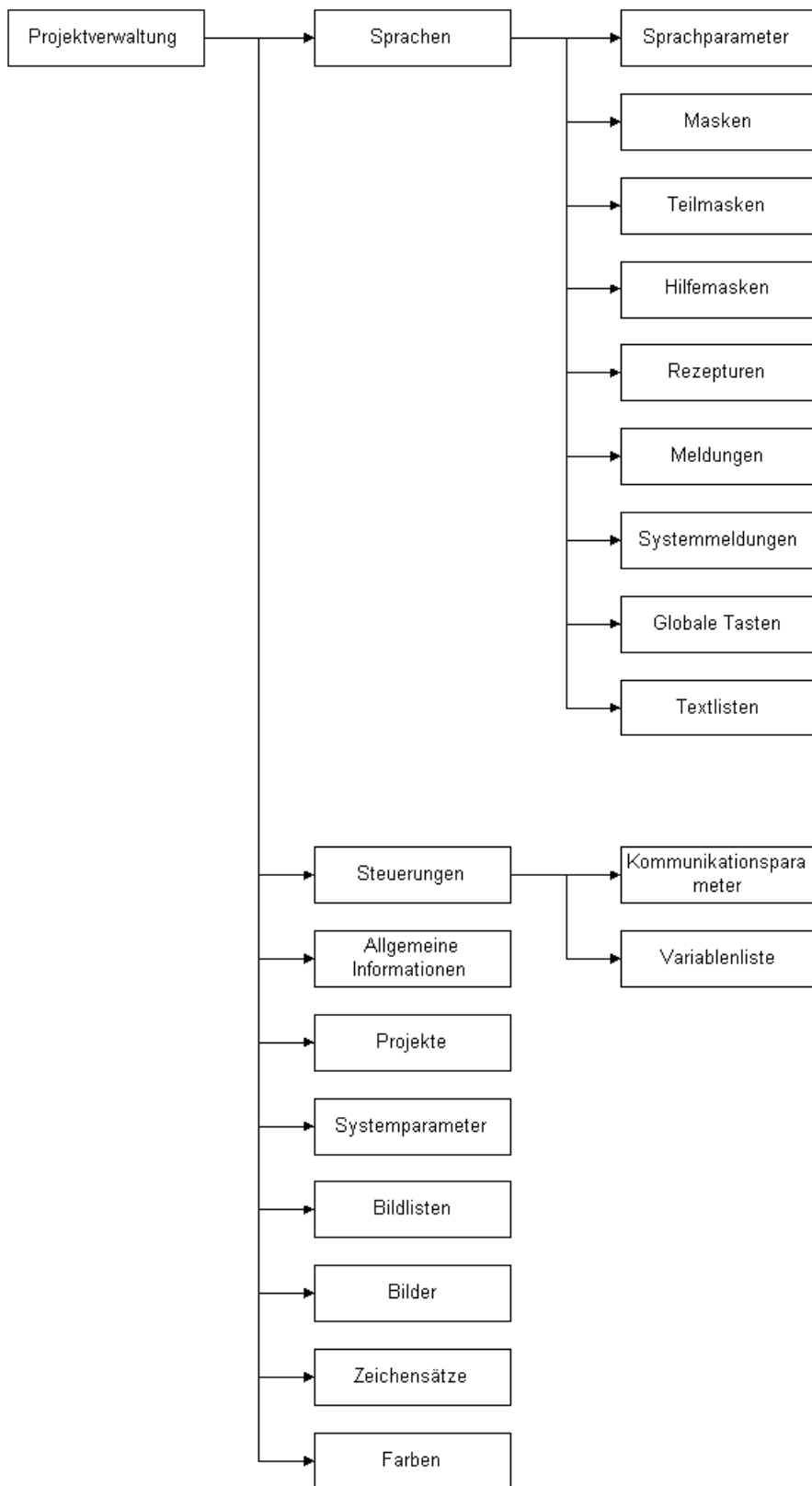


Bild 41: Struktur von TSwin-Projektiersystem

Über die Funktion und Bedienung des Projektiersystems TSdos steht ein weiteres umfangreiches Handbuch zur Verfügung. Außerdem unterstützt eine Onlinehilfe über die F1-Taste die Bedienung der Software. Mit den Projektiersystemen können alle für ein Projekt erforderlichen Arbeiten durchgeführt

werden. Mit TSDos lassen sich verschiedene Masken-, Parameter- und Variablendateien erstellen und zu einem Projekt zusammenfassen.

Das TSwIn-Projektiersystem arbeitet wie eine Datenbank. Alle projektbezogenen Daten werden in dieser Datenbank gesammelt und zum Schluß zu einem Projekt zusammengefaßt.

Anschließend kann das Projekt in eine S3-Datei übersetzt und mit der Download-Funktion an das Terminal übertragen werden.

3.27.3 Einstieg in die Projektierung

Der Einstieg in die Projekterstellung wird dem Anwender durch Demoprojekte erleichtert. Die Demoprojekte sind komplett erstellte Applikationen, die alle wichtigen Projektiermöglichkeiten aufzeigen.

Sollte eine Applikation für Ihren Bedienterminaltyp nicht zur Verfügung stehen, setzen Sie sich bitte mit unserem Vertrieb in Verbindung.

3.27.3.1 Projektbeschreibung

Die Projektbeschreibung besteht aus:

mindestens einer Anwenderbeschreibung (für jede Sprache eine Anwenderbeschreibung)
einer Variablenliste (steuerungsspezifisch)
den globalen Daten (Parametern)
und, falls verwendet, Teilen aus der grafischen Bibliothek.

In der Anwenderbeschreibung werden alle steuerungsunabhängigen Teile zusammengefaßt. Für alle steuerungsspezifischen Bezeichnungen werden in der Anwenderbeschreibung symbolische Namen eingesetzt.

In der Variablenliste werden die symbolischen Namen den entsprechenden Adressen in der Steuerung zugewiesen. Dadurch ergibt sich der Vorteil, daß beim Wechsel des Steuerungsherstellers die Masken, Variablenbeschreibungen und die Funktionalität der Bedienung nicht neu erstellt werden muß. Durch Austausch der Variablenliste kann die Anpassung an einen anderen Steuerungstyp sehr schnell vorgenommen werden.

3.27.3.2 Mehrsprachige Projekte

Die Anwenderbeschreibung bestimmt durch die Maskeninhalte auch die Landessprachen, in der die Texte dargestellt werden. Damit eine Bedienoberfläche in mehreren Sprachen zur Verfügung steht, werden die Anwenderbeschreibungen der einzelnen Sprachen unter Sprachnummern abgelegt. Durch die Zuordnung der Sprachnummer kann die jeweilige Sprache über eine Systemvariable online angewählt werden.

Die Anzahl der hinterlegbaren Sprachen wird nur durch die Maskenspeichergröße begrenzt. Es wird darauf hingewiesen, daß in den verschiedenen Sprachen weder gleiche Textlängen oder Variablen, noch die gleiche Maskenanzahl vorhanden sein muß.

Die Nummer der aktuell verwendeten Sprache wird im Bedienterminal gespeichert und geht auch nach dem Ausschalten nicht verloren. Das Terminal wechselt nach dem Einschalten wieder zu der gespeicherten Sprache. Im Falle eines Datenverlustes wird automatisch die Sprache angewählt, die dem Wert 0 entspricht.

Die Anwahl der Sprache am Terminal kann mit Hilfe von Auswahltext realisiert werden. Die Sprache kann dann mit den Tasten Cursor Oben und Cursor Unten ausgewählt werden.

Zusammengefaßt:

1. Textliste mit Sprachnamen anlegen (Beispiel: Deutsch, English, Francaise)
2. Maske anlegen
3. Variable in Maske anlegen:

OsLanguage, Darstellungsart Auswahltext, Textlistenname wie unter

1. angelegt



Achtung: Systemvariablen dürfen nicht in die Variablenliste eingetragen werden! Dadurch würde nicht mehr auf interne Bedienterminaladressen zugegriffen, sondern auf Steuerungsadressen.

3.27.3.3 Varianten-Projekte

Mit der Systemvariablen **OsLanguage** kann auch eine Variantenverwaltung durchgeführt werden. Wird beispielsweise eine Maschine mit Optionen gebaut und den Bediener sollen diese Optionen bei der Bedienung nicht stören, so kann die jeweilig gültige Variante online am Terminal gewählt werden.

Beispiel:

<u>WERT</u>	<u>FUNKTION</u>
0	Variante 1
1	Variante 2
2	Variante 3

Die Varianten können sich sowohl in der Bedienstruktur, der Anzahl der Masken als auch in den verwendeten Variablen unterscheiden. Der große Vorteil liegt in der nur einmal zu wartenden Anwendungsbeschreibung der Gesamtanlage.

Beispiel:

Eine Anwendung ergibt sich sehr schnell, wenn eine Anlage mit metrischen und angelsächsischen Maßangaben betrieben werden soll. Hier könnten dann neben der Landessprache auch die Variableneingaben und Variablenausgaben in Format und Skalierung von Millimeter zu Inch angepaßt werden.

3.27.4 Projektdokumentation

Mit der Projektiersoftware kann gleichzeitig eine umfangreiche Dokumentation des Projekts erstellt werden. Detaillierte Hinweise auf die Verwendung der Variablen, Skalierungsparameter, Darstellungsarten und Inhalte der Masken, sowie des Bediensystems, erleichtern die Kontrolle der Anwenderbeschreibung.

Anhand der Dokumentation kann die Anwenderbeschreibung auch bei Datenverlust wieder neu erstellt werden, sie enthält alle notwendigen Informationen.

3.27.4.1 Druckdateien von TSdos

Zur Dokumentation eines Projekts können in TSdos Druckdateien (Listen) mit folgenden Inhalten erzeugt werden:

- Anwenderbeschreibung
- Einzelanwenderbeschreibung
- Parameterliste
- Variablenliste
- Mapliste

Die Listen werden im Standard-ASCII-Format oder im erweiterten ASCII-Format (IBM-Mode) erstellt. Die Listen können dann, mit Hilfe von Befehlen des Betriebssystems, auf einem Drucker ausgegeben werden.

3.27.4.2 Hardcops der Masken von TSdos

Die Hardcopy des Displays soll in erster Linie die Arbeit bei der Erstellung einer Dokumentation durch den Anwender erleichtern. Deshalb wurde die Ausgabe der Hardcops bei grafikfähigen Displays so gestaltet, daß sie in eine Dokumentation eingebunden werden kann. Der Upload zum PC erfolgt über die Schnittstelle X3.

In einem Empfangsprogramm wie beispielsweise "Terminal" unter Windows 3.1x (1) werden die ankommenden Dateien eingelesen. Die eingelesene Datei entspricht dem PCX-Format, das von verschiedenen Grafik- oder Desktop-Publishing-Programmen eingelesen und weiterverarbeitet werden kann.

Über die Schnittstelle X3 kann von jeder Maske im Terminal eine Hardcopy erstellt werden. Dabei wird von grafikfähigen Anzeigen eine Pixelbeschreibung, von alphanumerischen Anzeigen eine ASCII-Beschreibung an den PC gesendet.

Das Aktivieren der Hardcopy erfolgt über die Systemvariable **HardCopy**. Es ist sinnvoll, während der Projektierung für die Erstellung der Hardcops, diese Systemvariable einer Funktionstaste zuzuweisen. Es können dann die gewünschten Masken angewählt und durch Betätigen der Funktionstaste an den PC übertragen werden.

Die für die Schnittstelle X3 eingestellten Parameter sind gültig und werden durch die Funktion der Hardcopy nicht verändert.

3.27.4.3 Dokumentationserstellung von TSwIn

TSwin bietet einen umfangreichen Dialog an, in dem die Elemente angewählt werden können, die dokumentiert werden sollen.

Die Dokumentation von TSwIn kann auf einfache Weise mit Textverarbeitungs- oder Desktop-Publishingprogrammen weiterverarbeitet werden.

3.27.5 Projektsicherung

Die Projektsicherung erfolgt durch Speicherung auf unterschiedlichen Datenträgern, wie vom PC-Hersteller empfohlen. Die Möglichkeiten sind im Handbuch des Betriebssystems ihres PCs beschrieben.



Wir empfehlen Ihnen, die erzeugten Versionen mit den Dokumentationshilfen auf Papier auszugeben.

3.27.6 Optimierung der Übertragungsgeschwindigkeit

Das TesiMod-Bedienkonzept unterstützt den freien Zugriff auf unterschiedliche Datentypen und Datenlängen.

Diese Freiheit wird von den einzelnen Protokollen nur sehr unterschiedlich unterstützt. Die Firmware der Terminals optimiert die Zugriffe maskenspezifisch, damit die kürzesten Zugriffszeiten erreicht werden.

Diese Optimierung versucht mehrere Variablen mit einer Übertragung zur Anzeige zu bringen. Der Anwender kann dies aktiv unterstützen, indem er in einer Maske so wenig wie möglich Variablentypen kombiniert und gleiche Typen möglichst fortlaufend adressiert (kleine Lücken spielen keine Rolle). Der Vorteil liegt in den kürzeren Auffrischzeiten und der geringeren Belastung der Schnittstelle.

3.28 Download der Anwenderbeschreibung

Die Funktion Download beschreibt wie eine neues Projekt in den im Terminal befindlichen FLASH-Speicher geladen wird. Dazu ist das Terminal in den Download-Betriebszustand zu versetzen.

Dies wird erreicht, indem

- in einer E/A-Maske auf die Systemvariable **IntEraseEprom** eine "1" geschrieben wird.
- die Versorgungsspannung abgeschaltet, DIP-Schalter S4 auf ON geschaltet, das Gerät wieder eingeschaltet und nach erscheinen der Systemmeldung

```
TURN POWER OFF
RESET DIP-SW 4
OTHERWISE ALL FLASH-
DATA WILL BE LOST
```

unter Betriebsspannung der DIP-Schalter S4 wieder auf OFF geschaltet wird.

Falls anstelle eines FLASH-Speichers ein normales UV löschbares EPROM im Terminal enthalten ist, wird dies erkannt und ein Lösch- oder Programmiervorgang verhindert.

Es erscheint die Fehlermeldung:

```
FLASH MEMORY FAILURE
.....
.....
```

die aussagt, daß ein Schreibvorgang nicht erfolgreich beendet werden konnte.

Während der Projektierung mit TSwIn ist es empfehlenswert die automatische Download-Funktion zu aktivieren. Durch starten des Downloads am PC wird das Bedienterminal automatisch in den Downloadzustand versetzt.

Beim Aufruf der Download-Funktion wird die momentan im Terminal geladene Anwenderbeschreibung gelöscht und auf dem Display erscheint auf der Schreibmarkenposition (1,1) die Information:

```
DOWNLOAD .....
.....
.....
```

In diesem Zustand läßt sich aus dem Projektiersystem im Menü der Projektverwaltung eine neue Anwenderbeschreibung an das Terminal übertragen. Dazu muß der PC über das Download-Kabel mit der Schnittstelle X3 des Terminals verbunden werden.

Um ein Löschen der Anwenderbeschreibung im Betrieb zu verhindern, kann die Maske mit der Download-Funktion mit einer entsprechend hohen Sperrebene versehen werden, so daß für den Zugang ein Paßwort erforderlich ist. Liegt diese Maske auf einer Steuertaste, dann wird für den normalen Bediener die Existenz einer solchen Maske nicht erkennbar.

Eine weitere Möglichkeit besteht darin, die Eingabe durch die angeschlossene Steuerung zu verhindern. Ein Paßwort-Schutz ist dazu nicht erforderlich. Für Maskennummer mit der Download-Funktion wird von der Steuerung nur dann eine Datenfreigabe gegeben, wenn die von Anwender gewünschten Bedingungen erfüllt sind.

Sind die Lösungen für den Anwender nicht praktikabel so kann auch ganz auf die Softwarelösung über sie Systemvariable verzichtet werden. Ohne die Systemvariable **IntEraseEprom** zu verwenden muß die alte Variablenbeschreibung über einen strengen Ablauf mit Hilfe des DIP-Switch S4 gelöscht werden. Anschließend geht das Terminal automatisch in den Download-Betriebszustand.

Terminal in den Download-Betriebszustand versetzen:

Nach der Aktivierung der Systemvariablen wird das Flash-EPROM gelöscht, während des Löschvorganges erscheint die Meldung:

```
ERASE FLASH EPROM ..  
.....  
.....
```

Nach dem Löschen erscheint die Meldung:

```
FLASH IS ERASED .....
```

Anschließend wechselt das Terminal automatisch in den Download-Mode. Auf der Anzeige erscheint:

```
DOWNLOAD .....
```

Das Terminal ist nun bereit über die Schnittstelle X3 eine neue Anwenderbeschreibung zu empfangen und im FLASH-EPROM zu speichern. Während der Datenübertragung werden ">>>>>"-Zeichen auf die Anzeige ausgegeben, die Zeichenlänge ändert sich ständig!

Am Ende der Übertragung erscheint auf der Anzeige:

```
READY .....
```

Das Bedienterminal beginnt sofort nach der Übertragung mit der Initialisierung und gibt die in der Anwenderbeschreibung enthaltene Start-Maske aus.

Während der Übertragung oder zu Beginn der Initialisierung können verschiedene Fehlermeldungen auf der Anzeige erscheinen:

ADDRESS ERROR	Fehler in der Adreßberechnung des Compilers
FLASH MEMORY FAILURE	Die geladene Anwenderbeschreibung ist fehlerhaft, oder unvollständig.
CHECKSUM ERROR	Die Anwenderbeschreibung ist fehlerhaft, bitte neu übersetzen
BYTECOUNT OVERFLOW	Die Anwenderbeschreibung ist fehlerhaft, bitte neu übersetzen.
FORMAT ERROR (S0,S3,S7)	Die übertragene S3-Datei weist fehlerhafte Sequenzen auf.

Erscheint eine Fehlermeldung auf der Anzeige so ist die übertragene Anwenderbeschreibung unvollständig oder fehlerhaft. Wird das Terminal in diesem Zustand aus-/eingeschaltet dann wird automatisch der Download-Betriebszustand wieder erreicht.

3.28.1 Download mit Windows

Der Download kann auch ohne die Projektiersoftware durchgeführt werden. Eine Möglichkeit dazu ergibt sich unter MS-Windows. Dazu wird das Terminal-Programm aus der Gruppe "Zubehör" verwendet.

Es sind folgende Einstellungen erforderlich:

1. PC COM1 oder COM2 mit BTxx-X3 via Downloadkabel verbinden
2. Terminal-Icon unter Windows anklicken (Doppelklick)
3. Parameter einstellen:

Menü:

Settings:

Communication: 19200Bd
7 Bit
odd-parity
xon/xoff
1 stop-bit

Transfers:

Send textfile:
(Dateinamen eingeben) xxxxx.S3

3.28.2 Applikationsspeicher

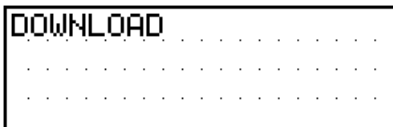
Als Applikationsspeicher kann ein Flashspeicher oder UV-EPROM eingesetzt werden. Bei der Auslieferung der Terminals ist dieses mit einem 128 kByte großen Flashspeicher bestückt. Dieser Flashspeicher steckt in einem 32pol. DIL-Präzisionssockel, der den Austausch mittels eines Ausziehwerkzeugs erlaubt. Der Speicher kann durch ein normales UV-EPROM entsprechender Größe und Geschwindigkeit ersetzt werden.

Die vom Projektiersystem erzeugte S3-Datei kann von den meisten EPROM-Programmiergeräten verarbeitet werden. Ein UV-EPROM kann im Terminal nicht gelöscht oder programmiert werden. Im Terminal ist nur die Programmierung von Flashspeicher vorgesehen.

Übrigens, Flashspeicher können auch von neueren EPROM-Programmiergeräten programmiert werden, dadurch könnte beim Service ein Download vor Ort entfallen.

3.28.3 Funktion Anwenderbeschreibung laden

Zum Laden der Anwenderbeschreibung ist an der Schnittstelle X3 ein PC über das Download-Kabel anzuschließen. Das Terminal ist in den Download-Zustand zu versetzen. Auf der Anzeige des Terminals muß zu lesen sein:



```
DOWNLOAD .....  
.....  
.....
```

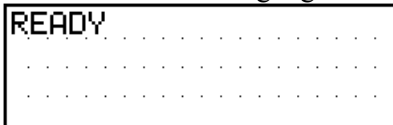
Das Terminal ist nun bereit über die Schnittstelle X3 eine neue Anwenderbeschreibung zu empfangen und im Flashspeicher zu speichern.

In der Projektierungssoftware ist nun im Menü der Projektverwaltung der Download einer fehlerfreien Anwenderbeschreibung "name.S3" zu starten.

Im Terminal werden, während der Datenübertragung, ständig ">>>>>"-Zeichen auf die Anzeige ausgegeben, die Zeichenlänge ändert sich ständig!

Im Projektiersystem werden "*" -Zeichen für jeden Übertragungsblock ausgegeben.

Am Ende der Übertragung erscheint auf der Anzeige im Terminal:



```
READY .....  
.....  
.....
```

Die Übertragung ist nun abgeschlossen. Das Terminal beginnt sofort nach der Übertragung mit der Initialisierung und gibt die in der Anwenderbeschreibung enthaltene Start-Maske aus.

3.28.4 Aktivieren der Downloadfunktion per Software

Dazu ist das Terminal in den Download-Betriebszustand zu versetzen. Dies wird erreicht, indem in einer E/A-Maske auf die Systemvariable **IntEraseEprom** eine "1" geschrieben wird.

Ohne die Systemvariable **IntEraseEprom** zu verwenden muß die alte Variablenbeschreibung über einen strengen Ablauf mit Hilfe von DIP-Schalter S4 gelöscht werden. Anschließend geht das Terminal automatisch in den Download-Betriebszustand.

3.28.5 Aktivieren der Downloadfunktion per Hardware

Falls die Systemvariable zum Löschen des Flashspeichers in den Masken nicht aktiviert werden kann oder gänzlich fehlt, besteht immer noch die Möglichkeit, durch eine definierte Schalterstellung des Betriebsartenschalters den Flashspeicher zu löschen und das Terminal in den Download-Zustand zu bekommen.

Schalterstellung für Flashspeicher bedingungslos löschen:

S1	ON	S5	frei
S2	frei	S6	frei
S3	OFF	S7	frei
S4	ON	S8	frei

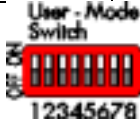


Bild 42: Betriebsartenschalter

Sobald am Terminal Spannung angelegt wird, erscheint auf der Anzeige folgende Meldung:

```
TURN POWER OFF
RESET DIP-SW 4
OTHERWISE ALL FLASH-
DATA WILL BE LOST
```

Diese Meldung soll ein ungewolltes Löschen der Anwenderbeschreibung verhindern. Wenn die Meldung eingehalten wird, dann bleibt die Anwenderbeschreibung erhalten.

Also zuerst die Spannungsversorgung ausschalten, Betriebsartenschalter S4 umschalten, dann Gerät neu einschalten. Das Bedienterminal arbeitet mit der bisherigen Anwenderbeschreibung.

Wird jedoch der Schalter S4 ohne abschalten der Versorgungsspannung umgeschaltet, dann wird der FLASH-Inhalt bedingungslos gelöscht!

3.28.6 Automatische Downloadfunktion

Die automatische Downloadfunktion kann nur beim Projektieren mit TSwIn aktiviert werden. Dazu muß in den Systemparametern -Allgemeine Parameter- ein entsprechendes Kontrollkästchen markiert werden.

Die Anwenderbeschreibung, die diese Funktion enthält, muß erstmals noch per erzwungenem Download (DIP-Schalter S4 auf ON) in das Terminal geladen werden. Jeder anschließende Download wird automatisch dadurch erzeugt, daß der PC die S3-Datei der neuen Anwenderbeschreibung in das Terminal laden will. Mitten im Betrieb des Bedienterminals kann ein Download erzwungen werden. Dabei gehen alle

- Meldungsdaten aus dem Meldungsspeicher
- alle RAM-Datensätze, die nicht gesichert wurden
- alle online veränderten Systemparameter (Werte von Systemvariablen, Schnittstellen, Paßwörter verloren!

3.28.7 Downloadkabel

Belegung des Verbindungskabels von der Schnittstelle X3 des Bedienterminals zum PC (außer IBT).

TesiMod
Bedienterminal

Personal
Computer

D-Subminiatur
Stiftstecker
9-polig

D-Subminiatur
Buchsenstecker
9-polig

Der Schirm liegt beidseitig auf dem Metallgehäuse.

Belegung des Verbindungskabels von der Schnittstelle X3 des Bedienterminals zum PC (nur BT2 / IBT).

TesiMod
Bedienterminal IBT / BT2

Personal
Computer

Der Schirm liegt beidseitig auf dem Metallgehäuse.

3.29 *Simulation ohne Steuerung*

Durch Einstellen des Betriebsartenschalters S3 auf ON läßt sich das Terminal ohne Steuerung betreiben. Durch den Betriebsartenschalter S3 wird auch automatisch die externe Datenfreigabe gesetzt. Das gesamte Bediensystem mit den Hilfetexten kann simuliert werden. Die Variablen sind automatisch einer gemeinsamen Variablenadresse im Terminal zugeordnet. Dadurch lassen sich die Editoren mit ihren Bereichsgrenzen testen, die Werte bleiben jedoch nicht erhalten.

Es findet keine Kommunikation zur Steuerung statt!

5.22 TesiMod - CAN/CANopen

Abkürzungen und Begriffe:

CAL CAN-Applikation-Layer, standardisiert durch CiA (CiA Draft Standard 102, 201, 205, 207)

CiA CAN in Automation International Users and Manufacturers Group e.V.

CANopen-Communication-Profile, CiA Draft Standard 301

5.22.1 Allgemeines

TesiMod Bedienterminals lassen sich auf einfache Weise an CAN-Bussysteme ankoppeln. Deshalb sind TesiMod Bedienterminals als Mensch-Maschine-Schnittstelle (MMI) für ihre CAN/CANopen-Applikation bestens geeignet.

Der CAN-Bus ist als High-Speed-Bus nach ISO-DIS 11898 ausgelegt. Alle Verbindungen sind galvanisch getrennt.

Die Anwendung von CAN in TesiMod Bedienterminals lehnt sich an die Spezifikation **CANopen Communication Profile** an.

CANopen verwendet eine Untermenge der Kommunikations-Dienste von CAL.

Bisher wurde von CiA noch kein Kommunikationsprofil für Mensch-Maschine-Schnittstellen (MMI) standardisiert. Deshalb hat die Fa. Süttron electronic GmbH eine eigene **Indirekte Prozeßdaten-Kommunikation** entwickelt.

5.22.2 Technische Beschreibung

Der Datenverkehr wird durch das Kommunikationsprofil des CANopen bestimmt. Basierend darauf werden Profile definiert, über die gleichartige Module gleichartig angesprochen werden können.

Mit Hilfe des Projektiersystems können acht verschiedene Kommunikationsbeziehungen definiert werden.

5.22.2.1 Datenobjekte

Alle Daten, die im CANopen-Netz auszutauschen sind, werden als Datenobjekte über ein Objektverzeichnis angesprochen. Bei CANopen werden Prozeßdatenobjekte (PDO) zusammengestellt. Diese Übertragungspakete können mehrere Datenobjekte (bis 8 Byte) enthalten. PDOs werden für das Senden und Empfangen von Daten zusammengestellt.

PDOs für das Senden von Daten heißen Request-PDO (PDO tx), die für das Empfangen heißen Response-PDO (PDO rx).

Für die Übertragung der PDOs werden Dienste der untersten Ebene der CAL verwendet, die keinerlei Protokoll-Overhead benötigen.

5.22.2.2 Identifier

Bei CANopen wird jedem PDO ein Identifier zugeordnet.

Jeder CANopen-Teilnehmer enthält maximal zwei Sende-PDOs und zwei Empfangs-PDOs, also vier Identifier.

Durch diese Einschränkung können auf einfachem Wege maximal 16 Byte gesendet und 16 Byte empfangen werden.

5.22.3 Indirekte Prozeßdaten-Kommunikation

Bei der Kommunikation mit Bedienterminals werden im CANopen-Netz Datenmengen ausgetauscht, die größer als 16 Byte sind. In zwei PDOs können diese Datenmengen dann nicht mehr untergebracht werden.

Deshalb wird vom Bedienterminal ein Datenobjekt nicht direkt angesprochen, sondern über spezielle Kommunikationsobjekte, die Request- und Responseobjekte. Diese teilen dem Partner mit, welches Datenobjekt gelesen oder beschrieben werden soll.

5.22.3.1 Ablauf vom Datenaustausch

Die für den Betrieb des Terminals notwendigen Dienste gehen alle vom Bedienterminal aus, das heißt das **Bedienterminal besitzt Client-Funktionalität**.

Der Kommunikationspartner reagiert lediglich auf die Anforderung des Terminals, das heißt der **Kommunikationspartner besitzt Server-Funktionalität**.

Das Bedienterminal sendet also Requestobjekte und der Kommunikationspartner sendet Responseobjekte.

Empfängt der Kommunikationspartner ein Requestobjekt vom Bedienterminal, so interpretiert er die ersten 4 Byte nach Datenrichtung und angesprochenem Datenobjekt.

Bei einer Schreibanforderung speichert er die angekommenen Daten und quittiert mit einem leeren Responseobjekt.

Bei einer Leseanforderung sendet er im Responseobjekt die angeforderten Daten an das Bedienterminal.

Sende-PDO und Empfangs-PDO sind immer paarweise vorhanden. Ein Paar steht für eine Kommunikationsbeziehung zwischen Bedienterminal und **einem** Teilnehmer.

Mit TesiMod-Bedienterminals können bis zu acht Kommunikationsbeziehungen aufgebaut werden.

5.22.4 Aufbau von Request- und Responseobjekten

Die Request- und Responseobjekte sind je 8 Byte groß. In den ersten 4 Byte dieser Objekte wird das angesprochene Datenobjekt definiert und in den restlichen 4 Byte stehen die Daten.

1. Byte	2. Byte	3. Byte	4. Byte	5. Byte	6. Byte	7. Byte	8. Byte
Status	Index:Low	Index:High	Subindex	Nutzdaten	Nutzdaten	Nutzdaten	Nutzdaten

Bild 1: Aufbau von Request- und Responseobjekten

5.22.4.1 Aufbau vom Statusbyte

Innerhalb von Request- und Responseobjekt hat das Statusbyte folgenden Aufbau:

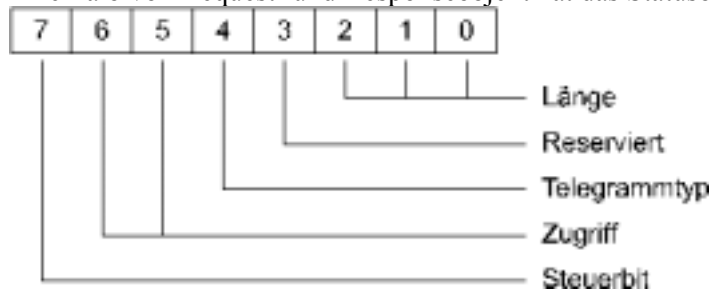


Bild 2: Aufbau vom Steuerbyte

Länge - Bit 0 bis Bit 2

1, 2	Bitzugriff	Länge der Bitmaske zum Setzen oder Löschen von
Bits		
1, 2, 3, 4	Bytezugriff	Anzahl in Bytes
2, 4	Wortzugriff	Anzahl in Bytes
4	Doppelwortzugriff	Anzahl in Bytes

Telegrammtyp - Bit 4

0	Request - Anforderungstelegramm
1	Response - Antworttelegramm

Zugriff - Bit 5 bis Bit 6

0	Bitzugriff	Schreibender Bitzugriff (setzen / löschen)
1	Bytezugriff	Byte lesen oder schreiben
2	Wortzugriff	Wort lesen oder schreiben
3	Doppelwortzugriff	Doppelwort lesen oder schreiben

Steuerbit - Bit 7

Requestobjekt

0	Schreiben
1	Lesen

Responseobjekt

0	O.K.
1	Fehler aus Steuerung bei Telegrammbearbeitung Die Fehlernummer steht im ersten Datenbyte

5.22.4.2 Indexbytes

Der Wert spezifiziert den Index aus dem Objektverzeichnis des Kommunikationspartners.

Der Index zeigt auf

- eine Variable ohne Subindex
- den Anfang eines Arrays
- den Anfang eines Records.

Bei Anwendung ohne CANopen kann hier eine beliebige Numerierung zwischen den Kommunikationspartnern vereinbart werden.

5.22.4.3 Subindexbyte

Der Wert spezifiziert den Subindex aus dem Objektverzeichnis des Kommunikationspartners. Der Subindex zeigt genau auf eine Variable (Basic-CAN-Variable) von der Größe des Zugriffs.

Bei Anwendung ohne CANopen kann hier eine beliebige Numerierung zwischen den Kommunikationspartnern vereinbart werden.

5.22.4.4 Nutzdatenbytes vom Request- / Responseobjekt

Nutzdaten bitorientiert

In einer Byteadresse:

5. Byte	6. Byte	7. Byte	8. Byte
AND-Maske		OR-Maske	

Bild 3: Nutzdaten in einer Byteadresse

In einer Wortadresse:

5. Byte	6. Byte	7. Byte	8. Byte
AND-Maske LOW-Byte	AND-Maske HIGH-Byte	OR-Maske LOW-Byte	OR-Maske HIGH-Byte

Bild 4: Nutzdaten in einer Wortadresse

Die Bitmasken sind in der Steuerung mit der angegebenen Adresse zu verODERN und zu verUNDEN, wobei die AND-Maske Bits löscht und die ODER-Maske Bits setzt.

Nutzdaten byteorientiert

5. Byte	6. Byte	7. Byte	8. Byte
1. Byte	2. Byte	3. Byte	4. Byte

Bild 5: Nutzdaten byteorientiert

Nutzdaten wortorientiert

5. Byte	6. Byte	7. Byte	8. Byte
1. Wort		2. Wort	
Low-Byte	High-Byte	Low-Byte	High-Byte

Bild 6: Nutzdaten wortorientiert

Nutzdaten doppelwortorientiert

5. Byte	6. Byte	7. Byte	8. Byte
Doppelwort			
Lowest-Byte	Byte	Byte	Highest-Byte

Bild 7: Nutzdaten doppelwortorientiert

5.22.4.5 Responseobjekt mit Fehler

Im Falle eines Kommunikationsfehlers wird ein Responseobjekt mit folgendem Aufbau zurückgesendet:

5. Byte	6. Byte	7. Byte	8. Byte
Fehler-Nr.			

Bild 8: Aufbau Responseobjekt mit Fehler

5.22.5 Aufgaben vom Kommunikationspartner

Der Kommunikationspartner des Bedienterminals hat Serverfunktionalität. Als Server muß der Kommunikationspartner die eingehenden Telegramme interpretieren und mit einem Responseobjekt beantworten. Dabei muß die in den Protokollparametern eingestellte Wartezeit eingehalten werden. Jedes Requestobjekt muß mit einem Responseobjekt beantwortet werden. Im Fehlerfall sendet der Kommunikationspartner gleichzeitig eine Fehlernummer an das Bedienterminal.

5.22.5.1 Ablauf bei fehlerfreier Kommunikation

- Das Statusbyte des Requestobjekts wird mit folgender Änderung in das Responseobjekt umkopiert:
Bit 7 Steuerbit = 0 fehlerfreie Aktion
Bit 4 Telegrammtyp = 1 Responseobjekt
- Index und Subindex werden vom Requestobjekt in das Responseobjekt umkopiert.
- Schreibt das Bedienterminal, so sind die Nutzdaten auf die angesprochene Adresse zu kopieren
- Liest das Bedienterminal Daten, so sind die Daten von der angesprochenen Adresse an die Adresse der Nutzdatenbytes zu kopieren.

5.22.5.2 Ablauf bei fehlerhafter Kommunikation

Stellt der Kommunikationspartner einen Fehler im Inhalt vom Requestobjekt fest, so kann dies durch eine Fehlermeldung im Responseobjekt dem Bedienterminal mitgeteilt werden. Die Fehlernummern können vom Anwender frei definiert werden. Grundsätzlich zeigt das Bedienterminal einen Communication-Error 100 am Display an. Der Subcode der Fehlermeldung enthält die Fehlernummer.

- Das Statusbyte des Requestobjekts wird mit folgender Änderung in das Responseobjekt umkopiert:
Bit 7 Steuerbit = 1 Fehler erkannt
Bit 4 Telegrammtyp = 1 Responseobjekt
- Index und Subindex werden vom Requestobjekt in das Responseobjekt umkopiert.
- Fehlernummer in Byte 5 eintragen (wird als Subcode am Bedienterminal angezeigt)

5.22.6 Protokollparameter

5.22.6.1 Baudrate

Für die Baudrate können folgende Einstellungen ausgewählt werden:

- 20 kBaud
- 125 kBaud
- 500 kBaud
- 1 MBaud

Grundsätzlich wird die Baudrate von 20 kBit/s vorgegeben.

5.22.6.2 Wartezeit für Antwort

Innerhalb der Wartezeit für Antwort muß der Kommunikationspartner mit einem Response-PDO geantwortet haben. Andernfalls wird eine entsprechende Fehlermeldung ausgegeben.

Die Wartezeit kann zwischen 0 ms und 65536 ms liegen.

Als Grundeinstellung wird der Wert 1000 ms eingetragen.

5.22.6.3 Zeit bis zu neuem Verbindungsaufbau

Das TesiMod-Bedienterminal wartet mit der Aufnahme der Kommunikation, bis die eingestellte Zeit verstrichen ist. Dadurch wird vermieden, daß eine fehlerhafte Kommunikation aufgebaut wird, bevor das Bussystem betriebsbereit ist.

Die Wartezeit kann zwischen 1 s und 255 s eingestellt werden.

Als Grundwert wird eine Zeit von 5 s eingetragen.

5.22.6.4 Terminal-Modulnummer

Im Modus CANopen wird mit Hilfe der Modulnummer die Vergabe der Identifier für Request und Response berechnet.

Für die Modulnummern können Werte zwischen 0 und 127 eingetragen werden.

Als Grundwert wird der Wert 1 vorgegeben.

5.22.6.5 CANopen Netzdienste

Der Anwender kann im Projektiersystem definieren, ob CANopen Netzdienste wahrgenommen werden sollen.

- NMT-Dienst mit Identifier = 0
 - Byte 1 = 1 Aktivieren des Teilnehmers (Knotens)
 - Byte 1 = 2 Deaktivieren des Teilnehmers (Knotens)
 - Byte 1 = 3 Deaktivieren des Teilnehmers (Knotens)
 - Byte 1 = 128 Deaktivieren des Teilnehmers (Knotens)
 - Byte 1 = 129 Reset des Teilnehmers (Knotens)
 - Byte 1 = 130 Kommunikation beim Teilnehmer (Knoten) zurücksetzen
- Das Bedienterminal sendet beim Hochlaufen eine Emergency-Nachricht mit Identifier = 128 + Terminal-Modulnummer

5.22.7 Kommunikationsbeziehungen

Für TesiMod Bedienterminals können acht unabhängige Kommunikationsbeziehungen definiert werden. Sehen Sie dazu das Bild 10 im Kapitel 5.22.11.

5.22.7.1 Modus

Die Vergabe der Identifier für die Kommunikationsbeziehungen kann auf zwei verschiedenen Wegen vorgenommen werden:

- Manuell: Der Request- und Responseidentifier wird direkt in den entsprechende Feldern eingetragen.
- CANopen: In diesem Fall richtet sich die Vergabe der Identifier nach dem Verteilschlüssel der Definition nach CANopen-Kommunikation-Profil DS301. Zu einer Basisnummer wird die Modulnummer des Slave, des Kommunikationspartners im CANopen-Netz, addiert und als Identifier eingetragen.

5.22.7.2 Verbindungspartner

Die Auswahl eines Verbindungspartners kann nur im CANopen-Modus erfolgen. Durch die Auswahl des Verbindungspartners wird der Basis-Identifizier bestimmt. Als Verbindungspartner kann entweder **Master** oder **Slave** ausgewählt werden.

- Master: Das Terminal kommuniziert mit dem CANopen-Netzmaster.
 - Es wird der erste PDO-Kanal verwendet.
 - Zur Basisnummer wird die Modulnummer vom Bedienterminal addiert.
- Slave: Das Bedienterminal kommuniziert mit einem anderen Slave im CANopen-Netz.
 - Es wird der zweite PDO-Kanal verwendet.
 - Zum Basisidentifizier wird die Modulnummer des anderen Slaves addiert.

5.22.7.3 Modulnummer

Die Eingabe einer Modulnummer kann nur im CANopen-Modus in Verbindung mit einer Kommunikationsbeziehung zu einem anderen Slave erfolgen.

Hier kann direkt die Modulnummer des Kommunikationspartners eingegeben werden.

5.22.7.4 Requestidentifizier

Die Eingabe eines Requestidentifiziers kann nur im Manuell-Modus erfolgen.

Während im CANopen-Modus der Requestidentifizier aus Basisidentifizier und Modulnummer berechnet wird, kann im Manuell-Modus direkt der Identifizier für das Requestobjekt eingegeben werden.

5.22.7.5 Responseidentifizier

Die Eingabe eines Responseidentifiziers kann nur im Manuell-Modus erfolgen.

Während im CANopen-Modus der Responseidentifizier aus Basisidentifizier und Modulnummer berechnet wird, kann im Manuell-Modus direkt der Identifizier für das Responseobjekt eingegeben werden.

5.22.7.6 Adreßtyp

Beim Zugriff auf Daten des Kommunikationspartners muß dem Bedienterminal für interne Adreßberechnungen bekannt sein, welche Adressierungsart der Datenbereich hat, auf den zugegriffen wird.

Folgende Werte können für den Adreßtyp eingetragen werden:

- Byteadresse 1
- Wortadresse 2
- Doppelwortadresse 4

5.22.8 Pollbereich

Für den Datenbereich des Pollbereichs sind folgende Bedingungen zu beachten:

- Die Steuerung muß einzelne Bits manipulieren können (Ansteuerung der LEDs)
- Das Bedienterminal muß diesen Datenbereich wortweise lesen können
- Vom Bedienterminal ist **nur Wortzugriff** erlaubt
- Der Adreßtyp des Datenbereichs in der Steuerung muß eine **Byte- oder Wortadresse** sein

Beispiel für die Adressierung des Pollbereichs:

Byteadressiert

Wort-Adresse + 0	Index 100	Subindex 0
Wort-Adresse + 1	Index 100	Subindex 2
Wort-Adresse + 2	Index 100	Subindex 4
Wort-Adresse + 3	Index 100	Subindex 6
Wort-Adresse + 4	Index 100	Subindex 8
Wort-Adresse + 5	Index 100	Subindex 10

Wortadressiert

Wort-Adresse + 0	Index 100	Subindex 0
Wort-Adresse + 1	Index 100	Subindex 1
Wort-Adresse + 2	Index 100	Subindex 2
Wort-Adresse + 3	Index 100	Subindex 3
Wort-Adresse + 4	Index 100	Subindex 4
Wort-Adresse + 5	Index 100	Subindex 5

	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Wort-Adr. + 0	frei	frei	frei	DDF	LM	PL	RQ	ED	frei	frei	frei	frei	frei	frei	frei	frei
Wort-Adr. + 1	Sequentieller Meldekanal High-Byte								Sequentieller Meldekanal Low-Byte							
Wort-Adr. + 2	LED 1 ein/aus	LED 1 blinken	LED 2 ein/aus	LED 2 blinken	LED 3 ein/aus	LED 3 blinken	LED 4 ein/aus	LED 4 blinken	LED 5 ein/aus	LED 5 blinken	LED 6 ein/aus	LED 6 blinken	LED 7 ein/aus	LED 7 blinken	LED 8 ein/aus	LED 8 blinken
Wort-Adr. + 3	LED 9 ein/aus	LED 9 blinken	LED 10 ein/aus	LED 10 blinken	LED 11 ein/aus	LED 11 blinken	LED 12 ein/aus	LED 12 blinken	LED 13 ein/aus	LED 13 blinken	LED 14 ein/aus	LED 14 blinken	LED 15 ein/aus	LED 15 blinken	LED 16 ein/aus	LED 16 blinken
Wort-Adr. + 4	LED 17 ein/aus	LED 17 blinken	LED 18 ein/aus	LED 18 blinken	LED 19 ein/aus	LED 19 blinken	LED 20 ein/aus	LED 20 blinken	LED 21 ein/aus	LED 21 blinken	LED 22 ein/aus	LED 22 blinken	LED 23 ein/aus	LED 23 blinken	LED 24 ein/aus	LED 24 blinken
Wort-Adr. + 5	LED 25 ein/aus	LED 25 blinken	LED 26 ein/aus	LED 26 blinken	LED 27 ein/aus	LED 27 blinken	LED 28 ein/aus	LED 28 blinken	LED 29 ein/aus	LED 29 blinken	LED 30 ein/aus	LED 30 blinken	LED 31 ein/aus	LED 31 blinken	LED 32 ein/aus	LED 32 blinken
Wort-Adr. + 6	LED 33 ein/aus	LED 33 blinken	LED 34 ein/aus	LED 34 blinken	LED 35 ein/aus	LED 35 blinken	LED 36 ein/aus	LED 36 blinken	LED 37 ein/aus	LED 37 blinken	LED 38 ein/aus	LED 38 blinken	LED 39 ein/aus	LED 39 blinken	LED 40 ein/aus	LED 40 blinken
Wort-Adr. + 7	LED 41 ein/aus	LED 41 blinken	LED 42 ein/aus	LED 42 blinken	LED 43 ein/aus	LED 43 blinken	LED 44 ein/aus	LED 44 blinken	LED 45 ein/aus	LED 45 blinken	LED 46 ein/aus	LED 46 blinken	LED 47 ein/aus	LED 47 blinken	LED 48 ein/aus	LED 48 blinken

Bild 9: Pollbereich wortorientiert

5.22.9 Physikalische Ankopplung

Für eine Integration eines Bedienterminals in eine CAN-Struktur stehen die optoentkoppelten Schnittstellen X2.1 und X2.2 zur Verfügung. Der CAN-Bus ist als High-Speed-Bus nach ISO-DIS 11898 ausgelegt.

Für die Verbindungsleitungen kommen abgeschirmte Kabel mit paarverseilten Adern in Frage. Der CAN-Bus muß an beiden Enden mit Terminationswiderständen abgeschlossen werden. Die Belegung der Schnittstellenstecker am Bedienterminal entsprechen CiA Draft Standard 102.

Steckverbinder im Terminal: 9-polige SubminD Buchsenleiste X2.1

Belegung:

Pin	Bel.	Bezeichnung	Funktion
1	x	res	Reserviert
2	x	CANL	CAN_L Busleitung (Dominant LOW)
3	x	0V_C	CAN Ground
4	x	res	Reserviert
5	x	res	Reserviert
6	x	0V_C	CAN Ground
7	x	CANH	CAN_H Busleitung (Dominant HIGH)
8	x	res	Reserviert
9	x	res	Reserviert

Steckverbinder im Terminal: 9-polige SubminD Stiftleiste X2.2

Belegung:

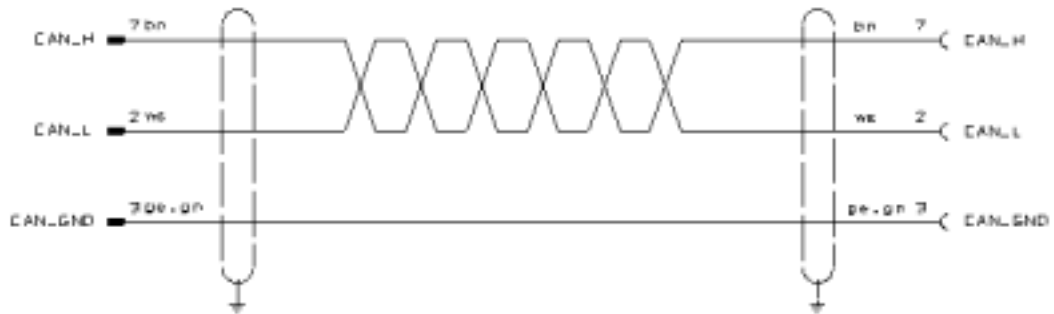
Pin	Bel.	Bezeichnung	Funktion
1	x	res	Reserviert
2	x	CANL	CAN_L Busleitung (Dominant LOW)
3	x	0V_C	CAN Ground
4	x	res	Reserviert
5	x	res	Reserviert
6	x	0V_C	CAN Ground
7	x	CANH	CAN_H Busleitung (Dominant HIGH)
8	x	res	Reserviert
9	x	res	Reserviert

Alle Signalleitungen sind von X2.1 zu X2.2 gebrückt. Die Verbindungskabel sollten an alle Pins, auch die reservierten, angeschlossen werden. Dadurch können die Kabel auch bei zukünftigen Erweiterungen der Busspezifikationen noch eingesetzt werden.

5.22.9.1 Verbindungskabel TesiMod Bedienterminal - CAN-Bus

TesiMod
Bedienterminal

Nächster
CAN-Bus-Teilnehmer



D-Subminiatur
Stiftstecker
9-polig

D-Subminiatur
Buchsenstecker
9-polig

Schirm liegt beidseitig auf Metallgehäuse.

Das Kabel ist entgegen den Empfehlungen des CiA Draft Standard 102 nur mit den Verbindungen ausgestattet, die, nach den derzeitigen Anforderungen, für eine Kommunikation nötig sind.

5.22.10 Fehlermeldungen

Code		
1	E_SLAVE_NOT_READY	Slave zur Zeit nicht bereit
2	E_PROTOKOL	Reihenfolge der Pakete
3	E_FRAME	Protokollrahmen-Fehler
4	E_TIMEOUT	Timeout-Fehler
5	E_CRC_BCC	CRC-Fehler
6	E_PARITY	Parity-Fehler
7	E_SEND_ABORT	Sendeabbruch
8	R_REC_ABORT	Empfangsabbruch
9	E_BUF_SIZE	Zyklischer Puffer zu klein
10	E_NO_DEFINE	Keine zyklischen Daten definiert
12	E_DEFINE	Zyklische Daten bereits definiert.
15	E_NO_PROTOCOL	Gewähltes Protokoll wird nicht unterstützt
16	E_OVERRUN	Empfangsüberlauf
40	E_SYS_ADDRESS	Undefinierte Systemvariable
50	E_CAN_ERROR	Fehler aus CAN-Controller
Subcode		
1		Stuff-Error
3		Terminal hat keine Verbindung zum Bus
4,5		Am Bus sind keine Teilnehmer angeschlossen
6		Busleitung hat Schluß nach logisch 0 oder 1 CRC-Error
Code		
51	E_RESPONSE-TIMEOUT	Keine Antwort vom Kommunikationspartner Kein Partner für Requestobjekt
53	E_NO_RESPONSE_BIT	Kein Empfänger für gesendeten Identifier Falsches Responseobjekt
54	E_RESPONSE OF NO_REQUEST	Nachricht ohne Response-Bit Falsches Responseobjekt
55	E_NO_HARDWARE	Response ohne Request Terminal findet keine CAN-Hardware CAN-Hardware in Terminal fehlt oder ist defekt
60	E_RESET_FROM_MASTER	NMT 0 Message mit Kommando 129
100	E_DATA_ERROR	Fehler vom Kommunikationspartner Fehlernummer steht im Subcode Subcode ist frei definierbar.

5.22.11 EUROTHERM spezifische Adressen und Definitionen

Für die Ankopplung der Terminals an die EUROTHERM Regler sind einige Besonderheiten zu beachten:

1. die Definition der Adreßvorwahlen, und
2. die Definition der Adreßbereiche und
3. die Definition der Identifier.

5.22.11.1 Definition der Adressvorwahlen

Adressvorwahl	Reglervariable
BY	BIAS Merker
W	Steuerbereich
DW	BIAS Variablen

5.22.11.2 Definition der Adressbereiche

Vorwahl	Bereich (Index)	Subindex	Adressweite	Erklärung
BY	0...255	0	1	BIAS Merker 0...255
W	0	0	2	Pollbereich
W	9	0	2	Koordinierungsbyte Lesen
W	10	0	2	Rezeptnummer an IBT
W	11	0	2	Datensatznummer an IBT
W	12	0	2	Rezeptnummer vom IBT
W	13	0	2	Datensatznummer vom IBT
W	15	0	2	Maskennummer
DW	0...255	0	4	BIAS Variablen 0...255
DW	1000	0	4	Istposition 1
DW	1001	0	4	Istposition 2 (X40)
DW	1002	0	4	Istposition 3 (Absolutwertgeber)

5.22.11.2 Definition der Identifier

Bei der Vergabe der Request- und Responseidentifier ist darauf zu achten, daß folgende Einstellungen eingehalten werden:

- Modus „manuell“
- Request-ID 0...2046
- Response-ID Request-ID+1
- Default-Adressweite 4

5.22.12 Beispiele für Kommunikationsbeziehungen

Mit Hilfe des Projektiersystems TSwIn lassen sich auf übersichtliche Weise acht Kommunikationsbeziehungen definieren. Das folgende Bild zeigt das Fenster von TSwIn, in dem die Beziehungen definiert werden und macht gleichzeitig deutlich, welche Einstellungen möglich sind.



Bild 10: Einstellungen der Kommunikationsbeziehungen in TSwIn als Beispiel

Passend zu den oben dargestellten Kommunikationsbeziehungen ist hier eine Variablenliste dargestellt, wie sie ebenfalls mit TSwIn erstellt wird. In der rechten Spalte der Tabelle werden die Kommunikationsbeziehungen eingetragen.

	Variablenname	Adresse	Komm.-bez.:
0	Var1	B 0:1.3	1
1	Var2	B 100:10.12	5
2	Var3	B 100:10.12	1 / 2
3	Var4	BY 1:35	1
4	Var5	BY 10:78	1
5	Var6	W 11:24	1
6	Var7	W 100:102	5
7	Var8	W 130:12	1 / 2
8	Var9	DW 100:102	5
9	Var10	DW 130:20	1 / 2
10	Var11	DW 50:25	6
11	Var12	DW 200:20	1 / 4

Bild 11: Variablenliste in TSwIn als Beispiel

In der Kommunikations-Beziehung 1 gilt folgendes:

- Index 0 - 99 byteadressiert,
- Index 100 - 199 wortadressiert
- Index 200 -300 doppelwortadressiert.

Variablenname	Zugriff	Index	Subindex	Bitnummer	Komm.-Beziehg.	Adreßtyp
Var1	Bit	0	1	3	1	

Bitzugriff: Bit 3 von Index 0, Subindex 1 von Kommunikations-Beziehung 1.

Ist für Adress-Typ nichts eingetragen, so gilt der Defaultwert in der Tabelle "Definition der Kommunikations-Beziehungen", also hier bei Index 0 Byteadresse.

Variablenname	Zugriff	Index	Subindex	Bitnummer	Komm.-Beziehg.	Adreßtyp
Var3	Bit	100	10	12	1	2

Bitzugriff: Bit 12 von Index 100, Subindex 10 von Kommunikations-Beziehung 1.

Index 100 in der Kommunikations-Beziehung 1 ist eine Wortadresse, default ist aber Byteadresse eingestellt. Deshalb muß hier eine 2 bei Adress-Typ eingegeben werden.

Variablenname	Zugriff	Index	Subindex	Bitnummer	Komm.-Beziehg.	Adreßtyp
Var2	Bit	100	10	12	5	

Bitzugriff: Bit 12 von Index 100, Subindex 10 von Kommunikations-Beziehung 5.

Variablenname	Zugriff	Index	Subindex	Bitnummer	Komm.-Beziehg.	Adreßtyp
Var5	Byte	10	10	78	1	

Bytezugriff: Index 10, Subindex 78, Kommunikations-Beziehung 1

Variablenname	Zugriff	Index	Subindex	Bitnummer	Komm.-Beziehg.	Adreßtyp
Var8	Word	130	12		1	2

Wortzugriff: Index 130, Subindex 12, Kommunikations-Beziehung 1

Index 100- 199 in Komm.-Bez. 1 ist eine Wortadresse, default ist aber Byteadresse eingestellt. Deshalb muß hier eine 2 bei Adress-Typ eingegeben werden.

Variablenname	Zugriff	Index	Subindex	Bitnummer	Komm.-Beziehg.	Adreßtyp
Var12	Doppelword	200	20	3	1	4

Doppelwortzugriff: Index 200, Subindex 20, Kommunikations-Beziehung 1

Index 200- 300 in Komm.-Bez. 1 ist eine Doppelwortadresse, default ist aber Byteadresse eingestellt. Deshalb muß hier eine 4 bei Adress-Typ eingegeben werden.

Variablenname	Zugriff	Index	Subindex	Bitnummer	Komm.-Beziehg.	Adreßtyp
Var11	Doppelword	50	25		6	

Doppelwortzugriff: Index 50, Subindex 25, Kommunikations-Beziehung 6

Index

A

Anzahl der hinterlegbaren Sprachen 134
Auffrischzeiten 137

B

Bedienung der TS-Software 133
Beschreibungsformat 129
Betriebsartenschalter
 Standard-Mode 17
Betriebsstundenzähler
 Reset-Byte 117

C

CAL 143
CANopen 143
CiA 143

D

Darstellung mit führenden Nullen 35
Darstellungen der Ausgabevariablen 34

E

Eingabevariable
 Timer 43
 Zähler 43
Einmalige und zyklische Ausgabe der Variablen 32
EUROTHERM-Definitionen 154
Externe Datenfreigabe 77

F

Falsches Format, Systemmeldungen 94
Firmware 136
Formatierte Ausgabe der Variablen 33
Formel für die Skalierung der Ausgabevariablen 34
Formel für die Skalierung der Eingabe 43

H

Hilfetaste 70
Hilfetext 10
 zur Eingabevariablen 44

K

Kommunikationsbaugruppe 8
Konfigurationsmaske 102

M

Maskennummer 118
Maskentypen 25
Masterpaßwort 22
Meldungen 13
Meldungsmaske 102
Meldungspuffer voll, Systemmeldungen 95

N

Nachkommastellen 33
Neue Meldung, Systemmeldungen 95
Nutzdaten bitorientiert, CAN 146
Nutzdaten byteorientiert, CAN 146
Nutzdaten doppelwortorientiert, CAN 146
Nutzdaten wortorientiert, CAN 146

P

Parallel-Ausgänge 7
Paßwort 21
Passwort 10
Paßworteditor 73
Paßwortmaske
 Systemmasken 26
Passwortschutz 10
Paßwortschutz
 Autorisierungslevel 21
 Setup-Maske 23
 Sperrzebene 21
 Startup-Maske 23
 Viewlevel 21
Plausibilität 70
Plausibilitätskontrolle
 Eingabevariable 44
Projektierung 128
Protokollspezifische Variablenliste 129

R

Randbedingungen 123
Reaktionszeit 7

S

Screen Saver 118
Skalierte Ausgabe der Variablen 33
Skalierte Variable 13
Symbolische Namen 118
Systemmasken
 Hauptmaske 26
 Setup-Maske 26
 Startup-Maske 26
Systemmeldungen
 Batterie wechseln 95
 Datensatz aktiv 96
 Datensatz geschützt 95
 Datensatz unbekannt 96
 Datensatz unzulässig 96
 Datensatzdateifehler 96
 Datensatzdownload 96
 Datensatzformat 96
 Datensatzmemory voll 96
 Datensatztransfer 96
 Druckprot. ungültig 95
 Durchschleifen aktiv 96
 Editiermode aktiv 96
 Fließkommazahl ungültig 96
 Keine Datensatzadr 96
 Meldungs-Überlauf 95
 Paßwort fehlt 96
 Paßwort unverändert 95
 Rezeptur unbekannt 96
 Schnittstelle belegt 95
 Überspannung 95
 Ungültige Maskennr 95
 Ungültige Meldungsnummer 95
 Ungültiges Paßwort 95
 Wert ist zu groß 94
 Wert ist zu klein 94
Systemnummern 93

Systemvariablen

- Betriebsstundenzähler
 - Counter1 67
 - Counter2 67
 - Counter3 67
 - Counter4 67
 - Counter5 67
 - Counter6 67
 - Counter7 67
- Druckersteuerung
 - BlockPrint 56
 - BlockPrintLong 57
 - PrintAllRep 56
 - PrintAllState 56
 - StopPrint 56
- Durchschleifbetrieb
 - Pg2Sps 68
 - Pg2SpsState 68
- Echtzeituhr
 - RTCDateFmt 52
 - RTCDay 51
 - RTCDayofWeek 52
 - RTCHour 51
 - RTCMin 51
 - RTCmonth 52
 - RTCsec 51
 - RTCYear 52
 - RTCYear2000 52
- Editoren
 - EditEnter 69
 - EditInvers 69
 - StatePerm 69
- Fehlerstatistik Schnittstelle X2
 - ComErrorTab 49
 - ComFrameCount 49
 - ComOverrunCount 49
 - ComParityCount 49
 - ComStatisticsTab 49
 - ComSubcodeTab 50
- Grundfunktionen
 - Boot 46
 - ComVersion 45
 - Handshake 126
 - IntEraseEprom 45
 - LCDBackground 46
 - LCDBackLight 46
 - LCDContrast 46
 - MainVersion 45
 - OsLanguage 47
 - TurnOnTemp 47
 - UserVersion 46
- Hilfe
 - Message 70
 - QuitMessage 70
 - StateHelp 69
- Kommunikationsbereich X2
 - ComBaudrateA 48
 - ComDataLenA 47
 - ComDefaultA 48
 - ComHandshakeA 48
 - ComParityA 47
 - ComRetryTimeout 48
 - ComSlaveNr 49
 - ComStopBitsA 47
 - ComTimeout 48
- Kommunikationsbereich X3
 - ComBaudrateB 50
 - ComDataLenB 50
 - ComDefaultB 51
 - ComHandshakeB 51
 - ComParityB 50
 - ComStopBitsB 50
- Ladbarer Zeichensatz
 - ChrsetName 68
- Menüsteuerung / Tasten
 - HardCopy 57
 - Key0 60
 - Key1 60
 - Key2 61
 - Key3 61
 - Key4 61
 - Key5 61
 - Key6 61
 - Key7 61
 - Key8 62
 - Key9 62
 - KeyClear 60
 - KeyCursDown 59
 - KeyCursLeft 59
 - KeyCursRight 59
 - KeyCursUp 59
 - KeyDot 60
 - KeyEdit 62
 - KeyEnter 62
 - KeyHelp 60
 - KeyHome 60
 - KeyMinus 62
 - KeyPlus 62
 - NewMask 57
 - Shift 58
 - ShiftCase 59
 - TabLeft 57
 - TabPgDn 58
 - TabPgUp 58
 - TabRight 58
 - VarTablenR0 57
 - VarTablenR1 57
- Paralleles Meldesystem
 - RepmanSortCritP 54
 - RepoutAnzYearP 55
 - RepoutDateP 55
 - RepoutNrP 54
 - RepoutRepTextP 55
 - RepoutRepTextP21 55
 - RepoutRepTextP41 56
 - RepoutRepTextP61 56
 - RepoutTimeP 55
- Paßwort
 - ChangePasswd 63
 - FlashPasswd 63
 - MskchgPasswd 63
 - MskchgResPasswd 63
 - Passwdlnactive 63
- Paßwortschutz
 - Editlevel 21
- Rezepturen
 - ActDSName 64
 - DestDSNr 64
 - DSCopy 64
 - DSDelete 64
 - DSDnloadBreak 65
 - DSDnloadState 65

- DSDownload 65
- LoadDSName 65
- RestoreLineNr 66
- RestoreState 66
- RezPrintState 66
- SaveState 66
- SelectDSName 64
- SelectDSNr 64
- SelectRezeptNr 65
- StartRestore 66
- StartRezPrint 66
- StartSave 65
- StartUpload 67
- UploadDSNr 67
- UploadState 67
- Serielles Meldesystem
 - ClearRepBuf 53
 - RepmanRepPrint 53
 - RepmanSortCrit 52
 - RepoutAnzYear 53
 - RepoutDate 53
 - RepoutNr 53
 - RepoutRepText 54
 - RepoutRepText21 54
 - RepoutRepText41 54
 - RepoutRepText61 54
 - RepoutTime 53
- Wartung
 - Break 69
 - LCDADCInput 68
 - LCDDACOutput 69
 - User1 68
 - User2 68
 - User3 68
 - User4 68
 - User5 68

T

Taste

- Datenfreigabe 70

Tastenfunktionen

- Alphanumerischer-Editor 73
- Editor für kodierten Text 74
- Hex-Editor 73
- in der E/A-Maske 29
- in der Knotenmaske 28
- in der Meldungsmaske 30, 31
- Inkrement-/Dekrement-Editor 72
- numerischer Editor 71

V

Variablentyp

- ASCII 32
- Bit 32
- Byte 32
- LWort 32
- Wort 32

W

Wertebereich 33

Z

Zustands-LED

- Datenfreigabe 77

Änderungsliste

Version	Änderungsgrund	Modification	Kapitel Chapter	Datum Date	Name Name	Bemerkung Comment
V02.49TB98	Ur-Version	<i>Source version</i>		03.12.98	T. Beul	im Eurotherm-Format <i>in Eurotherm design</i>