

Makroprogrammierung der pt-200  
Version x.52

Presence Technology GmbH+Co.KG

12. September 2000

# Inhaltsverzeichnis

<b>8 Makroprogrammierung</b>	<b>3</b>
8.1 Einführung	3
8.2 Anweisungen	4
8.2.1 Zahlen und Bit-Operationen	4
8.2.2 Textbehandlung	5
8.2.3 Timer- und Timeoutbehandlung	7
8.2.4 Dialog-Anweisungen	8
8.2.5 Test- und Vergleichs-Befehle	9
8.2.6 Sonstige Befehle	10
8.2.7 Liste aller Anweisungen	11
8.3 Funktionen und Ereignisse	12
8.3.1 Allgemeine Funktionen und Ereignisse	13
8.3.2 Video-Unterstützung	14
8.3.3 Audio/Telefonie Unterstützung	17
8.3.4 Voice/DTMF Unterstützung	18
8.3.5 Schnittstellen-Unterstützung	19
8.3.6 Auf- und Abbau von Verbindungen	21
8.3.7 Schaltausgänge und Schalteingänge	23
8.3.8 Alarmer	23
8.3.9 CAN-I/O Unterstützung	25
8.3.10 Unterstützung von Benutzer-Interaktion	29
8.3.11 Zeitabhängige Steuerung	29
8.4 GUI-Dialoge	30
8.4.1 Farbwerte	30
8.4.2 Anordnung der Dialoge	30
8.4.3 Pushbutton Dialog	31
8.4.4 Radiobutton Dialog	31
8.4.5 Checkbox Dialog	32
8.4.6 Entry Text, Texteingabefeld	32
8.4.7 Entry Number, Numerisches Eingabefeld	33

8.4.8	Slider, Schieberegler Dialog . . . . .	33
8.4.9	LED, Optische Anzeige . . . . .	34
8.4.10	Numerical Display, Numerische Anzeige . . . . .	35
8.4.11	Text Display, Alphanumerische Anzeige . . . . .	36
8.4.12	Static Text, Beschriftungstext . . . . .	36
8.4.13	Message Text, Mitteilungstext . . . . .	36
8.4.14	LogFile, Virtueller Dateidialog . . . . .	37

# Kapitel 8

## Makroprogrammierung

### 8.1 Einführung

Um die Flexibilität beim Einsatz der pt-200 in verschiedenen Anwendungsfällen zu erhöhen und die Komplexität der über Menues konfigurierbaren System Einstellungen nicht zu gross werden zu lassen, wurde die pt-200 Makroprogrammierbar gemacht.

Die Makroprogrammierung erlaubt dem Programmierer den Zugriff auf viele interne Funktionen des Gerätes. Die pt-200 ist damit auf verschiedene Anforderungen anpassbar. Die Makros werden als Textzeilen in die pt-200 geladen und vom Gerät interpretiert. Die Anweisungen sind sehr kompakt, da Speicherplatz und Interpretationsgeschwindigkeit wichtig sind.

Sprachelemente sind Anweisungen, (Benutzer-)variablen, Systemvariablen, Systemkonstante, Literale (Numerische- und Textkonstanten), Portadressen.

- Benutzerdefinierte Zahl- und Textvariablen beginnen mit einem Kleinbuchstaben. Sie dürfen die Zeichen a..z, A..Z, 0..9 und \_ enthalten. Beispielsweise pt\_ABC
- Systemkonstante bestehen nur aus Großbuchstaben. Beispielsweise HVER
- Zahlen sind Vorzeichenlos und haben den Wertebereich 0..65535. Als Hexadezimalzahlen werden Sie als \$0..\$FFFF notiert. Beispielsweise \$0001, \$AFFE, 45045
- Textkonstanten (manchmal Stringkonstanten genannt) werden durch ' eingeschlossen. Sie haben eine Länge von 0 bis 140 Zeichen. Im Text ist das Zeichen #0 (Code 0) nicht erlaubt, da es als Terminierungszeichen dient. Nicht-druckbare Zeichen können durch vorangestelltes # im Text enthalten sein. Beispielsweise 'Dies ist ein Text'#13#10'mit Zeilenvorschub.'
- I/O-Portadressen werden in rechteckigen Klammern angegeben. Beispielsweise [\$120] für den Parallelen Anschluß.
- Vom System vordefinierte Ereignisse rufen Makros gleichen Namens auf, sofern sie definiert sind. Diese Makros bestehen nur aus Großbuchstaben und Zahlen und beginnen mit #. In Definition des Makros können diese Makronamen mit einem \* beendet werden. Beispielsweise: Makroname #CAM1, wird Umschalten auf Kamera 1 aufgerufen. Makroname #CAM\* wird bei Umschalten einer beliebigen Kamera aufgerufen.
- Makro-, Event- und Variablennamen bestehen aus max. 11 Zeichen.
  - Anweisungen besitzen das Format *Befehl Param1,Param2* oder *Befehl Param1*. Der Befehlsname besteht aus zwei Großbuchstaben.
  - Zwei Anweisungen werden durch ein Semikolon (;) voneinander getrennt. *Param1* und *Param2* kann einer der folgenden Parametertypen sein. Nach der letzten Anweisung eines Makro muß kein Semikolon stehen.

- Anweisungsfolgen können durch runde Klammern () eingeschlossen werden, um einen Anweisungsblock zu erzeugen. Ein Anweisungsblock kann z.B. im IF THEN oder ELSE-Zweig verwendet werden. Es lassen sich mehrere (max. 10) Anweisungsblöcke schachteln.

- Parametertypen

- Textkonstante beginnend mit # oder '
- Dezimalzahl beginnend mit 0..9
- Hexadezimalzahl beginnend mit \$
- Text-Variable beginnend mit a..z und als Text-Variable definiert
- Word-Variable beginnend mit a..z und als Word-Variable definiert
- undefinierte Variable beginnend mit a..z und noch nicht definiert
- I/O-Port beginnend mit eckiger Klammer [
- Systemvariable/-konstante beginnend mit A..Z

- Falls auf eine undefinierte Variable in einem numerischen Ausdruck lesend zugegriffen wird, dann wird 0 gelesen und die Variable bleibt undefiniert.

- Eine bedingte Verzweigung (IF-Anweisung) wird durch einen Doppelpunkt (:) eingeleitet und hat folgendes Format:

<BOOLScheAnweisung>:<TRUE-Anweisung>!<FALSE-Anweisung>

Die BOOLSche Anweisung ist in der Regel eine Vergleichsanweisung, kann aber auch eine beliebige Anweisung sein, die eine Rückgabewert in der Systemvariablen RES zurückliefert.

Hat RES einen Wert < 256, wird dies von der bedingten Verzweigung als TRUE interpretiert, hat RES einen Wert ≥ 256, wird dies als FALSE interpretiert.

Beispiel max(a,b) bestimmt die größere von zwei Zahlen: GEa,b:SEmax,a!SEmax,b

Beispiel: EQn,1:SEx,'eins'!(EQn,2:SEx,'zwei'!(EQn,3:SEx,'drei'!SEx,'andere'))

Der FALSE-Anweisungszwei ist optional:

EQa,'123':SEx,123;

EQa,'456':SEx,456;

## 8.2 Anweisungen

Im folgenden wird für die Parametertypen der Anweisungen folgende Konvention verwendet:

Typ	Beschreibung
WORD	Dezimal-Zahl, Hexadezimalzahl oder Zahl Variable
TEXT	Text-Literal oder Text-Variable
NUM	Dezimal-Zahl, Hexadezimalzahl, Zahl-Variable oder Port
NVAR	ZahlVariable, undefinierte Variable oder beschreibbarer I/O-Port
TVAR	TextVariable oder undefinierte Variable
WVAR	WordVariable oder undefinierte Variable

Viele Anweisungen liefern in einen numerischen Rückgabewert in der Systemvariablen RES bzw. einen alphanumerischen Rückgabewert in der Systemvariablen NAME zurück. Ein numerischer Rückgabewert kann in der bedingten Verzweigung verwendet werden. Falls der Rückgabewert weiterverarbeitet werden soll, muß er mit im nächsten Befehl z.B. mit SEvar,RES gespeichert oder ausgewertet werden, da folgende Anweisungen die Systemvariable verändern. Ein Makro kann den Systemvariablen RES und NAME selbst einen Rückgabewert zuordnen, indem es z.B. mit SERES,1234 oder SENAME,'info' eine Wertzuweisung vornimmt.

### 8.2.1 Zahlen und Bit-Operationen

Anweisung	Param 1	Param 2	Funktion
SE	NVAR	NUM	Set. Wertzuweisung von NUM an I/O-Port oder Variable
AN	NVAR	NUM	AND. Bitweise Und-Verknüpfung NVAR := NVAR AND NUM
OR	NVAR	NUM	OR. Bitweise Oder-Verknüpfung NVAR := NVAR OR NUM
NA	NVAR	NUM	NAND. Bitweise NAND-Verknüpfung NVAR := NVAR AND NOT NUM
XO	NVAR	NUM	XOR. Bitweise Exklusiv-Oder Verknüpfung NVAR := NVAR XOR NUM
SL	NVAR	NUM	Shift Left. Linksschieben um NUM Bits: SHL (NVAR, NUM)
SR	NVAR	NUM	Shift Right. Rechtsschieben um NUM Bits: SHR (NVAR, NUM)
MU	NVAR	NUM	Multiplikation. NVAR := NVAR*NUM
DI	NVAR	NUM	Division. NVAR := NVAR DIV NUM
MO	NVAR	NUM	Modulo. Restberechnung NVAR := NVAR MOD NUM
IN	NVAR	NUM	Inkrementierung (Addition). NVAR := NVAR+NUM
DE	NVAR	NUM	Dekrementierung (Subtraktion). NVAR := NVAR-NUM
SI			Signed Prefix. Wirkt nur auf die nächste Anweisung. Der nächste Vergleichs-, Konvertierungs- oder Arithmetikoperation wird vorzeichenbehaftet ausgeführt. Ein WORD-Parameter wird als vorzeichenbehafteter INTEGER Wert interpretiert.

## 8.2.2 Textbehandlung

Bei Verwendung von Textvariablen ist zu beachten, daß deren Länge bei Textoperationen nicht über 140 Zeichen erreicht.

Anweisung	Param 1	Param 2	Funktion
SE	TVAR	TEXT	Set.

Anweisung	Param 1	Param 2	Funktion												
			Wertzuweisung von TEXT an Text-Variable												
IN	TVAR	TEXT	Insert. Fügt den Text TEXT am Anfang von TVAR ein												
IN	TVAR	NUM	Insert. Fügt das Zeichen NUM am Anfang von TVAR ein												
DE	TVAR	WORD	Delete. WORD ist in HIGH und LOW Teil aufgeteilt (jeweils ein Byte): <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>HIGH</th> <th>Operation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Entfernt in TVAR das Zeichen an Position LOW</td> </tr> <tr> <td>255</td> <td>Entfernt in TVAR das letzte Zeichen</td> </tr> <tr> <td>sonst</td> <td>Entfernt an Position HIGH LOW Zeichen.</td> </tr> </tbody> </table>	HIGH	Operation	0	Entfernt in TVAR das Zeichen an Position LOW	255	Entfernt in TVAR das letzte Zeichen	sonst	Entfernt an Position HIGH LOW Zeichen.				
HIGH	Operation														
0	Entfernt in TVAR das Zeichen an Position LOW														
255	Entfernt in TVAR das letzte Zeichen														
sonst	Entfernt an Position HIGH LOW Zeichen.														
WT	WORD	WORD	Write Text. Gibt das Zeichen des zweiten Parameters an der COM-Schnittstelle deren Nummer im ersten Parameter steht aus. Zeichencodes 0..255 sind erlaubt.												
WT	WORD	TEXT	Write Text. Gibt den Text auf der COM-Schnittstelle, deren Nummer im ersten Parameter steht, aus. COM-Schnittstellen: <table border="1" style="margin-left: 20px;"> <tbody> <tr> <td>0</td> <td>LPT (Printer)</td> </tr> <tr> <td>1</td> <td>V.24/RS232C (Modem/Remote-I/O)</td> </tr> <tr> <td>3</td> <td>LCD</td> </tr> <tr> <td>5</td> <td>ISDN</td> </tr> <tr> <td>6</td> <td>TV/Monitor</td> </tr> <tr> <td>8..15</td> <td>Parallelport PD0..PD7 Serielle TX-Emulation</td> </tr> </tbody> </table>	0	LPT (Printer)	1	V.24/RS232C (Modem/Remote-I/O)	3	LCD	5	ISDN	6	TV/Monitor	8..15	Parallelport PD0..PD7 Serielle TX-Emulation
0	LPT (Printer)														
1	V.24/RS232C (Modem/Remote-I/O)														
3	LCD														
5	ISDN														
6	TV/Monitor														
8..15	Parallelport PD0..PD7 Serielle TX-Emulation														
RT	WORD	WVAR	Read Text. Liest ein Zeichen von COM-Schnittstelle (Parameter 1) in die Variable WVAR ein. Falls kein Zeichen verfügbar ist, wird 0 geliefert. Es wird nicht gewartet.												
RT	WORD	TVAR	Read Text. Hängt solange Zeichen von der COM-Schnittstelle (Parameter 1) an die Variable TVAR an, bis 10ms lang kein Zeichen verfügbar ist, oder TVAR die maximale Länge von 140 Zeichen überschreitet.												
CH	NVAR	TEXT	Convert Hexadecimal. Wandelt die als Text übergebene Hexadezimal-Repräsentation um und weist sie an NVAR zu.												
CH	TVAR	WORD	Convert Hexadecimal. Wandelt die Zahl WORD in eine Hexadezimalzahl-Repräsentation (ohne führendes Dollar-Zeichen) um und weist sie an die Text-Variable zu.												
CD	NVAR	TEXT	Convert Decimal.												

Anweisung	Param 1	Param 2	Funktion
			Wandelt die als Text übergebene Dezimalzahl-Repräsentation um und weist sie an NVAR zu.
CD	TVAR	WORD	Convert Decimal. Wandelt die Zahl WORD in eine Dezimal-Repräsentation um und weist sie an die Text-Variable zu.
CB	NVAR	TEXT	Convert Binary. Wandelt die als Text übergebene Binärzahl-Repräsentation um und weist sie an NVAR zu.
CB	TVAR	WORD	Convert Binary. Wandelt die Zahl WORD in eine Binärzahl-Repräsentation um und weist sie an die Text-Variable zu.
LE	WVAR	TEXT	Length. Liefert in WVAR die Länge (=Anzahl Zeichen) des Textes TEXT.
AP	TVAR	TEXT	Append. Hängt den TEXT an das Ende der Text-Variablen an.
AP	TVAR	WORD	Append. Hängt das Zeichen aus WORD an das Ende der Text-Variablen an.
ST	TEXT	WORD	Subtext Extract. Liefert den Subtext aus TEXT von Position LOBYTE(WORD) mit der Länge HIBYTE(WORD). Falls HIBYTE(WORD)=0 ist wird ein Länge=1 angenommen und genau ein Zeichen geliefert. Das Ergebnis wird in der Systemvariablen NAME zurückgeliefert. In der Systemvariablen RES wird der Zeichencode des Ergebnisstrings zurückgeliefert. Falls die Länge des Resultatstrings größer 1 ist, werden die ersten beiden Zeichen des Strings als WORD interpretiert zurückgeliefert.
ST	TEXT	TEXT	Subtext Position. Liefert die Position (1...n), an der der 2.Text im ersten Text enthalten ist. Ist der 2.Textparameter kein Bestandteil des ersten Parameters, dann wird in der Systemvariable RES \$FFFF (FALSE) zurückgeliefert.

### 8.2.3 Timer- und Timeoutbehandlung

Bei Ablauf eines Timers wird das entsprechende Makro gleichen Namens aufgerufen. Der Timer wird nicht automatisch neu gesetzt. Um einen periodischen Timer zu erhalten, muß der Timer in diesem Makro neu gesetzt werden.



Die Timer werden während einer Bildverbindung höchstens alle 500ms überprüft. Im Standby ca. alle 10ms. Die Genauigkeit hängt stark von der aktuellen Systembelastung ab, da ein Aufruf des Timeout-Ereignis-Namens z.B. nicht während einer Wählaktion, einer Bildübertragung oder Ausführung eines Makros geschieht.

Die Timer können durch die TT-Anweisung getestet werden, so daß es auch möglich ist, Timeouts durch Polling und nicht nur Ereignisgesteuert abzufragen.

Anweisung	Param 1	Param 2	Funktion
TO	TEXT	WORD	Timeout Milliseconds. Setzt den Timeout für den Timer (Name in Parameter 1) auf die im 2.Parameter übergebenen Millisekunden. Falls Parameter 2=0 ist, wird der Timer abgeschaltet und liefert keinen Timeout mehr.
TS	TEXT	WORD	Timeout Seconds. Setzt den Timeout für den Timer (Name in Parameter 1) auf die im 2.Parameter übergebenen Sekunden.
TM	TEXT	WORD	Timeout Minutes. Setzt den Timeout für den Timer (Name in Parameter 1) auf die im 2.Parameter übergebenen Minuten.
TH	TEXT	WORD	Timeout Hours. Setzt den Timeout für den Timer (Name in Parameter 1) auf die im 2.Parameter übergebenen Stunden.

## 8.2.4 Dialog-Anweisungen

Dialog-Anweisungen dienen während einer Online-Verbindung (z.B. via Netz, Modem, ISDN) zur Kommunikation mit dem Benutzer. Die Darstellung der Dialogelemente ist abhängig von der Benutzeroberfläche der Gegenseite. Überlicherweise wird im #ONL Ereignismakro ein Dialog-Initialisierungsmakro aufgerufen, daß alle MD-Anweisungen aufruft, um die Dialoge auf der Benutzeroberfläche darzustellen.

Bei Verwendung der ptWin-Bediensoftware lassen sich die Dialogelemente von Seite 8.4 verwenden.

Anweisung	Param 1	Param 2	Funktion
MD	TEXT	TEXT	Make Dialog. Erzeugt ein Dialogelement auf dem Bildschirm des Hosts, falls eine Verbindung besteht. Im 1.Parameter wird der Name für den Dialog übergeben (max.11 Zeichen), im zweiten Parameter die Beschreibung des Dialogelementes (Syntax s.u.).
WD	TEXT	TEXT	Write Dialog. Gibt den Text des zweiten Parameters im Dialog mit dem Namen des ersten Parameters aus.
WD	TEXT	NUM	Write Dialog. Gibt den Zahlwert des zweiten Parameters im Dialog mit dem Namen des ersten Parameters aus.

Anweisung	Param 1	Param 2	Funktion
QD	TEXT		Query Dialog. Fordert den Dialog-Status an. Wird dann über ein Event mit dem Namen des ersten Parameters mitgeteilt.
KD	TEXT		Kill Dialog. Entfernt den Dialog, der im Parameter TEXT angegeben ist vom Bildschirm des Hosts, falls eine Online-Verbindung besteht.
RE	TEXT	TEXT WORD	Remote Event. Erzeugt ein Ereignis mit Namen, der als 1.Parameter übergeben wird. Dies führt während der Verbindung auf der Remote-Seite zum Aufruf eines Makros gleichen Namens. Hierdurch lassen sich bei Box-zu-Box Verbindungen Dialogmakro-Eingaben simulieren. Optional kann ein TEXT- oder WORD-Parameter übergeben werden, der als Parameter RES/NAME auf der Remote-Seite abfragbar ist.

### 8.2.5 Test- und Vergleichs-Befehle

Diese Anweisungen liefern ein bool'sches Resultat (0000=TRUE, FFFF=FALSE). Das Resultat ist direkt nach Aufruf der Anweisung in der Systemvariablen RES verfügbar. Das Resultat kann z.B. mit der bedingten Anweisung : abgefragt werden.

Anweisung	Param 1	Param 2	Funktion
TT	TEXT		Timer-Test. Prüft ob Timer noch läuft. Ein Rückgabewert FALSE wird geliefert, wenn der Timer abgelaufen ist.
TB	NUM	WORD	Test Bit. Prüft ob das Bit (Nummer als 2.Parameter WORD) in Port, Wert oder Variable des ersten Parameters (NUM) gesetzt ist.
MS	NUM	NUM	Mask Set. Führt eine bitweise UND-Verknüpfung der beiden Parameter durch (ohne das Ergebnis zu speichern) und liefert TRUE, wenn das Ergebnis gleich dem zweiten Parameter ist. Dadurch lässt sich prüfen, ob in einem I/O Port oder einer Variablen alle gewünschten Bits gesetzt sind.
MC	NUM	NUM	Mask Cleared. Führt eine bitweise UND-Verknüpfung der beiden Parameter durch (ohne das Ergebnis zu speichern) und liefert TRUE, wenn das Ergebnis 0 ist. Dadurch lässt sich prüfen, ob in einem I/O Port oder einer Variablen alle gewünschten Bits gelöscht sind.

Anweisung	Param 1	Param 2	Funktion
EQ	NUM	NUM	Equal. Prüft beide Werte auf Gleichheit.
EQ	TEXT	TEXT	Equal. Prüft beide Texte auf Gleichheit.
NE	NUM	NUM	Not Equal. Prüft beide Werte auf Ungleichheit.
NE	TEXT	TEXT	Not Equal. Prüft beide Texte auf Ungleichheit.
GE	NUM	NUM	Greater or Equal. Liefert TRUE, wenn NUM1 >= NUM2 ist.
GE	TEXT	TEXT	Greater or Equal. Liefert TRUE, wenn TEXT1 >= TEXT2 ist.
LE	NUM	NUM	Less or Equal. Liefert TRUE, wenn NUM1 <= NUM2 ist.
LE	TEXT	TEXT	Less or Equal. Liefert TRUE, wenn TEXT1 <= TEXT2 ist.
LT	NUM	NUM	Less Than. Liefert TRUE, wenn NUM1 < NUM2 ist.
LT	TEXT	TEXT	Less Than. Liefert TRUE, wenn TEXT1 < TEXT2 ist.
GT	NUM	NUM	Greater Than. Liefert TRUE, wenn NUM1 > NUM2 ist.
GT	TEXT	TEXT	Greater Than. Liefert TRUE, wenn TEXT1 > TEXT2 ist.
CT	TEXT	TEXT	Contains. Liefert TRUE, wenn TEXT2 in TEXT1 enthalten ist.

## 8.2.6 Sonstige Befehle

Anweisung	Param 1	Param 2	Funktion
FN	WORD	...	Function call. Ruft Systemfunktion (siehe unten) auf.
JP	TEXT		Jump to Makro. Springt in das Makro dessen Name als Parameter übergeben wird. Hinweis: Der Makro-name muss in ' Zeichen stehen. Hinweis: Alle Zeichen nach dem Sprungbefehl im aktuellen Makro werden ignoriert.
CA	TEXT		Call Makro.

Anweisung	Param 1	Param 2	Funktion
			Ruft das Makro TEXT als Unterprogramm auf. Max. 12 Unterprogramme, Funktionsaufrufe und Makros lassen sich ineinander verschachteln.
KI	VAR		Kill. Entfernt die Text- oder Zahl-Variable bzw. das Makro. Sie/Es ist anschliessend undefiniert.

## 8.2.7 Liste aller Anweisungen

Anweisung	Param 1	Param 2	Funktion
AN	NVAR	NUM	Bitwise AND.
AP	TVAR	TEXT	Append text.
AP	TVAR	WORD	Append character.
CA	TEXT		Call Makro.
CB	NVAR	TEXT	Convert binary string to numerical value.
CB	TVAR	WORD	Convert numerical value to binary string.
CD	NVAR	TEXT	Convert cecimal string to numerical value.
CD	TVAR	WORD	Convert numerical value to decimal string.
CH	NVAR	TEXT	Convert hexadecimal string to numerical value.
CH	TVAR	WORD	Convert numerical value to hexadecimal string.
CR	0		CAN Test Message available.
CR	WVAR		Read CAN Message. Only Header.
CR	WVAR	TVAR	Read CAN Message. Long Value.
CR	WVAR	WVAR	Read CAN Message. Short Value.
CS	WORD		Send CAN Message. No Value.
CS	WORD	TEXT	Send CAN Message. Long Value.
CS	WORD	WORD	Send CAN Message. Short Value.
CT	TEXT	TEXT	Test if text contains other text.
DE	NVAR	NUM	Decrement Variable.
DE	TVAR	WORD	Delete Text.
DI	NVAR	NUM	Division.
EQ	NUM	NUM	Test numerical equal.
EQ	TEXT	TEXT	Test text equal.
FN	WORD	...	Function call.
GE	NUM	NUM	Test numerical Greater or Equal.
GE	TEXT	TEXT	Test text Greater or Equal.
GT	NUM	NUM	Test numerical Greater Than.
GT	TEXT	TEXT	Test text Greater Than.
IN	NVAR	NUM	Increment Variable.
IN	TVAR	NUM	Insert Character.
IN	TVAR	TEXT	Insert Text.
JP	TEXT		Jump to Makro.
KD	TEXT		Kill GUI Dialog.
KI	VAR		Kill Makro or Variable.
LE	NUM	NUM	Test numerical Less or Equal.
LE	TEXT	TEXT	Test text Less or Equal.
LE	WVAR	TEXT	Length of text.
LT	NUM	NUM	Test numerical Less Than.
LT	TEXT	TEXT	Test text Less Than.
MC	NUM	NUM	Clear Masked Bits.
MD	TEXT	TEXT	Make GUI Dialog.
MO	NVAR	NUM	Modulo (Remainder of Division).
MS	NUM	NUM	Set Masked Bits.

Anweisung	Param 1	Param 2	Funktion
MU	NVAR	NUM	Multiplication.
NA	NVAR	NUM	Bitwise NAND.
NE	NUM	NUM	Test numerical Not Equal.
NE	TEXT	TEXT	Test text not Equal.
OC	WORD		CAN Open Clear Output.
OG	TEXT	TVAR	CAN Open Read Object long value.
OG	WORD	WVAR	CAN Open Read Object short Value.
OI	WORD		CAN Open Read Input.
OO	WORD		CAN Open Read Output.
OS	WORD		CAN Open Set Output.
OW	TEXT	TEXT	CAN Open Write Object long Value.
OW	TEXT	TEXT	CAN Open Write Object long value.
OW	WORD	WORD	CAN Open Write Object short Value.
OW	WORD	WORD	CAN Open Write Object short Value.
OR	NVAR	NUM	Bitwise OR.
QD	TEXT		Query GUI Dialog.
RE	TEXT	TEXT	Remote Event. Call Remote Macro with text parameter.
RE	TEXT	WORD	Remote Event. Call Remote Macro with numerical parameter.
RT	WORD	TVAR	Read Text from COM-Port.
RT	WORD	WVAR	Read Character from COM-Port.
SE	NVAR	NUM	Variable Assignment.
SE	TVAR	TEXT	Set Text. Variable-Assignment.
SI			Signed Prefix.
SL	NVAR	NUM	Shift Left.
SR	NVAR	NUM	Shift Right.
ST	TEXT	WORD	Subtext string to NAME/RES.
ST	TEXT	TEXT	Subtext Position to RES.
TB	NUM	WORD	Test Bit.
TH	TEXT	WORD	Set Timeout in hours.
TM	TEXT	WORD	Set Timeout in minutes.
TO	TEXT	WORD	Set Timeout in milliseconds.
TS	TEXT	WORD	Set Timeout in seconds.
TT	TEXT		Timer-Test.
WD	TEXT	NUM	Write number to GUI Dialog.
WD	TEXT	TEXT	Write text to GUI Dialog.
WT	WORD	TEXT	Write Text to COM-Port.
WT	WORD	WORD	Write Character to COM-Port.
XO	NVAR	NUM	Bitweise XOR, Exclusive Or.

### 8.3 Funktionen und Ereignisse

Funktionen werden mit der Anweisung FN aufgerufen. Beispielsweise wird die Delay-Funktion, die 1000 Millisekunden verzögern soll, durch FN1,1000 aufgerufen. Im folgenden werden die Funktionsnummern als Hexdezimalzahlen angegeben, um die Systematik hervorzuheben. Es ist natürlich möglich die Funktionsnummer als Dezimalzahl zu schreiben z.B. statt FN\$1C,cam auch FN28,cam.

Systemerzeugte Ereignisse werden nach vordefinierten Bedingungen aufgerufen und durch ein # Zeichen eingeleitet. Der Name besteht nur aus Großbuchstaben und wird nicht länger als 11 Zeichen inclusive eines übergebenen hexadezimalen Parameters. Der hexadezimale Parameter wird vor dem Anhängen an den Namen soweit gekürzt, so daß er ohne führende Nullen darstellbar ist und mindestens ein Zeichen lang ist.

Ein Makro gleichen Namens wird aufgerufen, sobald ein Ereignis eintritt. Anstelle des vollständigen Ereignisnamens kann der Makroname am Ende ein \* enthalten, was bedeutet, das alle Ereignisse, unabhängig vom Parameter dieses Makro aufrufen. Beispielsweise wird bei drücken der DTMF-Taste

5 das Ereignis #DTMF5 aufgerufen. Ein Makro #DTMF\* würde auf alle Tasten #DTMF0 bis #DTMF9 reagieren.

### 8.3.1 Allgemeine Funktionen und Ereignisse

#### Allgemeine Funktionen

Funktion	Parameter	Beschreibung
\$01	WORD ms	Delay ms. Beispiel: FN1,1000: Warte 1 Sekunde. Die Programmausführung sollte nicht länger als einige Sekunden angehalten werden. Besser ist es längere Verzögerungen mittels Timer/Timeout Konstruktion durchzuführen.
\$02	WORD com	Clear Com Buffer. Löscht den Eingabepuffer der Schnittstelle z.B. com=1: Serielle Schnittstelle.
\$03		System Reset.
\$04		Get Current Label. Liefert (durch Systemvariable NAME abfragbar) den Namen des aktuellen Makros (=Events).
\$05		Write EEPROM Config. Aktuelle Einstellungen ins EEPROM sichern. Nur möglich, falls Offline oder Online mit Config-Permission.

#### Allgemeine Ereignisse

Ereignis	Parameter	Beschreibung
#RESET	0	Aufruf bei Reset des Gerätes vor der Initialisierung von Harddisk, Kameras, Modem, ISDN etc.
#ERROR	1 code	Aufruf bei Reset des Gerätes nach obiger Initialisierung. Wird aufgerufen, wenn eine Störung im Gerät auftritt. Der code ist der hexadezimale Errorcode.

## Allgemeine Systemvariablen

Variable	R/W	Typ	Beschreibung
RES	R/W	WORD	Resultat des letzten Vergleichs. Da die Systemvariable beschreibbar ist, kann ihr Wert auch durch SERES,xxx verändert werden.
NAME	R/W	TEXT	Resultat einer Textoperation (siehe FN...) Bzw. Eingangs Parameter eines Makroaufrufs.
PAR	R	WORD	Parameter des letzten Event/Makro Aufrufs, der bei CALL oder vom Host übergeben wurde. Z.B. der Wert 0000 oder FFFF, der den aktuellen Button Status widerspiegelt.
BNAME	R	TEXT	"Boxname" (Name des Gerätes, max. 20 Zeichen)
BNR	R	TEXT	Boxrufnummer (kann Ländervorwahl enthalten) dieses Gerätes.
TEMP	R	WORD	Innentemperatur der pt200 in Grad Celsius.

## 8.3.2 Video-Unterstützung

### Video-Funktionen

Funktion	Parameter	Beschreibung
\$10	WORD cam	Select Video Input. Setze Videoeingang auf cam (1..3) Die Systemvariable RES enthält TRUE, wenn ein Videosignal an diesem Eingang erkannt wurde, anderenfalls FALSE.
\$11	WORD 0 1	Set Video State. Switch Video off. Switch Video on.
\$12	WORD 5 4 9 8	Store Image. Speichert Vollbild von aktueller Kamera auf Harddisk. Speichert Differenzbild von aktueller Kamera auf Harddisk. Speichert Vollbild von aktueller Kamera ins RAM. Speichert Differenzbild von aktueller Kamera ins RAM..
\$13	WORD size	Set screen size. Setzt Spalten und Zeilen des Text-Schirms der Monitor-Ausgabe (COM 6). LOBYTE(size)=Spalten HIBYTE(size)=Zeilen
\$14	WORD cam	Get Camera Name. Speichert den Namen der Kamera cam (1..3) in der System-Variablen NAME.
\$pp15	WORD val	Video/Audio Remote Control. Falls diese Box als Bildempfänger arbeitet lassen sich damit die Kamera- und Audioparameter der Remote-Box einstellen:

Funktion	Parameter	Beschreibung								
\$0015	cam	Video Input. Kameraumschaltung cam=1..3								
\$0115	mode 0 1	Videomodus. HDelta64 DCT (Verlustbehaftet)								
\$0215	size	Set Image Size. <table border="1" style="margin-left: 20px;"> <tr> <td>0</td> <td>176x144</td> </tr> <tr> <td>1-9</td> <td>Teilbildauschnitt wie auf der Zehnertastatur</td> </tr> <tr> <td>14</td> <td>96x72</td> </tr> <tr> <td>15</td> <td>352x288</td> </tr> </table>	0	176x144	1-9	Teilbildauschnitt wie auf der Zehnertastatur	14	96x72	15	352x288
0	176x144									
1-9	Teilbildauschnitt wie auf der Zehnertastatur									
14	96x72									
15	352x288									
\$0315	luma	Bildhelligkeit. luma=0(dunkel)..9(hell)								
\$0415	val	Kontrast. val=0(gering)..9(hoch)								
\$0515	qual	Qualität/Kompressionsstärke. qual=0(beste Qualität)..9(niedrigste Qualität)								
\$0615	sat	Farbsättigung. sat=0..9								
\$0715	lsvol	Speaker Volume. <table border="1" style="margin-left: 20px;"> <tr> <td>0</td> <td>Audio Hangup</td> </tr> <tr> <td>1..9</td> <td>Lautstärke</td> </tr> </table>	0	Audio Hangup	1..9	Lautstärke				
0	Audio Hangup									
1..9	Lautstärke									
\$0815	micvol	Microphone Volume. <table border="1" style="margin-left: 20px;"> <tr> <td>0</td> <td>AudioHangup</td> </tr> <tr> <td>1..9</td> <td>Lautstärke</td> </tr> <tr> <td>14</td> <td>Mikrofon 1 (intern)</td> </tr> <tr> <td>15</td> <td>Mikrofon 2 (extern)</td> </tr> </table>	0	AudioHangup	1..9	Lautstärke	14	Mikrofon 1 (intern)	15	Mikrofon 2 (extern)
0	AudioHangup									
1..9	Lautstärke									
14	Mikrofon 1 (intern)									
15	Mikrofon 2 (extern)									
\$0915	ctrl	Control.								
\$1C	WORD cam	Store Image to RAM. Speichert Bild von Kamera cam ins RAM.								
\$1D	WORD cam	Store Image to Harddisk. Speichert Bild von Kamera cam auf Harddisk.								
\$1E	WORD param	Bildänderungsdetektor für aktuellen Videoeingang aufrufen. Beim Ersten Aufruf nach einer Kameraumschaltung oder aktivem Video-Framefrabber (z.B. während einer Onlineverbindung oder Harddiskspeicherung) werden die übergebenen Parameter (s.u.) neu gesetzt und ein Referenzbild angefordert. Durch Aufruf von FN\$10,cam (Anfordern eines Vollbildes) kann das Initialisieren der durch FN\$1E übergebenen Parameter erzwungen werden.								



Funktion	Parameter	Beschreibung																					
		<table border="1"> <thead> <tr> <th>Bit</th> <th>Wertebereich</th> <th>Bedeutung</th> </tr> </thead> <tbody> <tr> <td>0..3</td> <td>16,1..15</td> <td>Anzahl geänderter Pixel bevor ein Block als geändert gilt.</td> </tr> <tr> <td>4..7</td> <td>16,1..15</td> <td>Min. Grauwertänderung damit Pixel als geändert gilt.</td> </tr> <tr> <td>8..9</td> <td>4,1..3</td> <td>Höhe des Sensitiven Fensters.</td> </tr> <tr> <td>10..11</td> <td>4,1..3</td> <td>Breite des Sensitiven Fensters.</td> </tr> <tr> <td>12..13</td> <td>0..3</td> <td>Linke obere Ecke Y des Sensitiven Fensters.</td> </tr> <tr> <td>14..15</td> <td>0..3</td> <td>Linke obere Ecke X des Sensitiven Fensters.</td> </tr> </tbody> </table>	Bit	Wertebereich	Bedeutung	0..3	16,1..15	Anzahl geänderter Pixel bevor ein Block als geändert gilt.	4..7	16,1..15	Min. Grauwertänderung damit Pixel als geändert gilt.	8..9	4,1..3	Höhe des Sensitiven Fensters.	10..11	4,1..3	Breite des Sensitiven Fensters.	12..13	0..3	Linke obere Ecke Y des Sensitiven Fensters.	14..15	0..3	Linke obere Ecke X des Sensitiven Fensters.
		Bit	Wertebereich	Bedeutung																			
0..3	16,1..15	Anzahl geänderter Pixel bevor ein Block als geändert gilt.																					
4..7	16,1..15	Min. Grauwertänderung damit Pixel als geändert gilt.																					
8..9	4,1..3	Höhe des Sensitiven Fensters.																					
10..11	4,1..3	Breite des Sensitiven Fensters.																					
12..13	0..3	Linke obere Ecke Y des Sensitiven Fensters.																					
14..15	0..3	Linke obere Ecke X des Sensitiven Fensters.																					
		Die Funktion liefert in der Systemvariablen RES die Zahl der geänderten Blöcke des Bildes zurück. Ein volles Bild hat 48x36 Blöcke.																					
\$60	WORD ctrl	Kamera-Control für aktuelle Kamera ändern. LOBYTE: BitNr HIBYTE: Neuer Wert (1=Bit setzen, 0=Bit rücksetzen) <table border="1"> <thead> <tr> <th>Bit</th> <th>Bedeutung</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Schnell abtasten</td> </tr> <tr> <td>2</td> <td>Glätten</td> </tr> <tr> <td>8</td> <td>DCT-Basierte Kompression</td> </tr> <tr> <td>9</td> <td>Kamera 3 S-VHS</td> </tr> </tbody> </table> Z.B.: 108h: DCT-Basierte Kompression einschalten 008h: DCT-Basierte Kompression ausschalten	Bit	Bedeutung	1	Schnell abtasten	2	Glätten	8	DCT-Basierte Kompression	9	Kamera 3 S-VHS											
Bit	Bedeutung																						
1	Schnell abtasten																						
2	Glätten																						
8	DCT-Basierte Kompression																						
9	Kamera 3 S-VHS																						
\$61	WORD  size	Bildgröße/Ausschnitt der aktuellen Kamera ändern. <table border="1"> <thead> <tr> <th>LOBYTE(size)</th> <th>Bildabmessungen</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>192x144 (oder 176x144)</td> </tr> <tr> <td>1..9</td> <td>160x128 Ausschnitt aus 384x288 mit Position: 1 2 3 4 5 6 7 8 9</td> </tr> <tr> <td>14</td> <td>96x72 (oder 88x72)</td> </tr> <tr> <td>15</td> <td>384x288 (oder 352x288)</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>HIBYTE(size)</th> <th>Pixeländerung pro 8x8 Pixel Block</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Default setzen (&lt;120 Zeilen: 1, &lt; 240Zeilen:2 sonst 3)</td> </tr> <tr> <td>1..5</td> <td>Neuen Wert setzen</td> </tr> </tbody> </table>	LOBYTE(size)	Bildabmessungen	0	192x144 (oder 176x144)	1..9	160x128 Ausschnitt aus 384x288 mit Position: 1 2 3 4 5 6 7 8 9	14	96x72 (oder 88x72)	15	384x288 (oder 352x288)	HIBYTE(size)	Pixeländerung pro 8x8 Pixel Block	0	Default setzen (<120 Zeilen: 1, < 240Zeilen:2 sonst 3)	1..5	Neuen Wert setzen					
LOBYTE(size)	Bildabmessungen																						
0	192x144 (oder 176x144)																						
1..9	160x128 Ausschnitt aus 384x288 mit Position: 1 2 3 4 5 6 7 8 9																						
14	96x72 (oder 88x72)																						
15	384x288 (oder 352x288)																						
HIBYTE(size)	Pixeländerung pro 8x8 Pixel Block																						
0	Default setzen (<120 Zeilen: 1, < 240Zeilen:2 sonst 3)																						
1..5	Neuen Wert setzen																						
\$62	WORD  grey	Graustufen/Farbe für aktuelle Kamera ändern: <table border="1"> <thead> <tr> <th>LOBYTE(grey)</th> <th>Graustufen/Farbe</th> </tr> </thead> <tbody> <tr> <td>16,32,64</td> <td>Graustufen</td> </tr> <tr> <td>32+128</td> <td>Farbe mit fünf Helligkeitsstufen</td> </tr> <tr> <td>64+128</td> <td>Farbe mit sechs Helligkeitsstufen</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>HIBYTE(grey)</th> <th>Schwellwert Helligkeitsänderung</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Default (&lt; 64 Graustufen: 2 Grauwerte sonst 3)</td> </tr> <tr> <td>1..8</td> <td>Neuer Schwellwert</td> </tr> </tbody> </table>	LOBYTE(grey)	Graustufen/Farbe	16,32,64	Graustufen	32+128	Farbe mit fünf Helligkeitsstufen	64+128	Farbe mit sechs Helligkeitsstufen	HIBYTE(grey)	Schwellwert Helligkeitsänderung	0	Default (< 64 Graustufen: 2 Grauwerte sonst 3)	1..8	Neuer Schwellwert							
LOBYTE(grey)	Graustufen/Farbe																						
16,32,64	Graustufen																						
32+128	Farbe mit fünf Helligkeitsstufen																						
64+128	Farbe mit sechs Helligkeitsstufen																						
HIBYTE(grey)	Schwellwert Helligkeitsänderung																						
0	Default (< 64 Graustufen: 2 Grauwerte sonst 3)																						
1..8	Neuer Schwellwert																						
\$63	WORD ctrbri	Kontrast/Helligkeit auf neuen Wert (1...255) setzen: LOBYTE: Kontrast (0 = Nicht ändern) HIBYTE: Helligkeit (0 = Nicht ändern)																					

## Video-Ereignisse

Ereignis	Parameter	Beschreibung
#CAM	1..3	Kamera 1..3 wird ausgewählt. Dieses Ereignis wird <i>vor</i> der Umschaltung und Überprüfung der Videoquelle ausgeführt, damit evtl. ein vorgeschalteter Video-Multiplexer auf einen bestimmten Eingang umgeschaltet werden kann.
#NCAM	1..3	Kamera 1..3 wurde ausgewählt. Dieses Ereignis wird <i>nach</i> der Umschaltung und Überprüfung der Videoquelle aufgerufen. Die Systemvariable CAM enthält den gleichen Wert wie der übergebene Parameter.
#VIDEO	0 1 2	Videoteil abgeschaltet. Videoteil ohne TV/Monitor eingeschaltet (nur Kameras) Videoteil mit TV/Monitor aktiv

## Systemvariablen

Variable	R/W	Typ	Beschreibung
CAM	R	WORD	Ausgewählte Kamera (1..3). 0=Videoeingänge inaktiv.
TVO	R	WORD	TV-Output Status: 0=Inaktiv, 1=Aktiv.

## 8.3.3 Audio/Telefonie Unterstützung

### Audio Funktionen

Funktion	Parameter	Beschreibung								
\$20		Simple Beep 200ms. Beep über den externen Lautsprecher (Frequency = \$20)								
\$20	WORD dur	Beep with variable length. Beep über den externen Lautsprecher für 'dur' ms. Lautstärke über Menüpunkt "Ruf-Lautstärke" einstellbar.								
\$21	WORD par	Beep with variable frequency and volume. Dauerton mit vorgegebener Lautstärke und Frequenz über den externen Lautsprecher wiedergeben. LOBYTE(par) = Frequenz (1..255) HIBYTE(par) = Volume (0..9) Falls Volume = 0 ist, wird die "Ruf-Lautstärke" aus dem entsprechenden Menue verwendet.								
\$22	WORD  mic	Select microphone. <table border="1" style="display: inline-table; vertical-align: middle;"> <thead> <tr> <th>mic</th> <th>Bedeutung</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Audioverbindung beenden</td> </tr> <tr> <td>1</td> <td>Internes Mikrofon auswählen</td> </tr> <tr> <td>2</td> <td>Externes Mikrofon auswählen</td> </tr> </tbody> </table>	mic	Bedeutung	0	Audioverbindung beenden	1	Internes Mikrofon auswählen	2	Externes Mikrofon auswählen
mic	Bedeutung									
0	Audioverbindung beenden									
1	Internes Mikrofon auswählen									
2	Externes Mikrofon auswählen									
\$23	WORD par	Beep with variable length and frequency. Beep mit der eingestellten Ruf-Lautstärke, aber gewählter Frequenz und Länge über externen Lautsprecher ausgeben.								

Funktion	Parameter	Beschreibung
\$24		LOBYTE(par) = Frequenz HIBYTE(par) = Länge in ms. Falls HIBYTE(par)=0 wird 200ms angenommen.
		Accept incoming telephone call. Annehmen ankommender Anrufe. Liefert TRUE, falls Anruf angenommen wurde.

### Audio Ereignisse

Ereignis	Parameter	Beschreibung
#MIC	0	Audio-Verbindung beendet (Mikrofon abgeschaltet).
	1	Internes Mikrofon bzw. MIC1 gewählt.
	2	Externes Mikrofon bzw. MIC2 gewählt.

### Systemvariablen

Variable	R/W	Typ	Beschreibung	
MIC	R	WORD	Ausgewählter Mikrofoneingang	
			0	Keine Übertragung per Mikrofon (kein Mikrofon aktiv)
			1	Eingebautes Mikrofon überträgt
			2	Externes Mikrofon überträgt

### 8.3.4 Voice/DTMF Unterstützung

Die DTMF-Funktionen sind während einer bestehenden Voice-Verbindung verfügbar und haben sonst keine Funktion.

#### Voice/DTMF-Funktionen

Funktion	Parameter	Beschreibung
\$30	TEXT dtmf	Send DTMF Tone DTMF-Zeichenfolge als Ton erzeugen. Erlaubte Zeichen im Text <i>dtmf</i> sind 0..9, *, #, A..F, e, b und das Leerzeichen für Pause. Die Töne haben eine Länge von 80ms und werden ohne Pausen nacheinander erzeugt.
\$31	0	Play Beep "Recognized" *
\$31	1	Play Beep "ok"
\$31	2	Play Beep/Announcement "Error"
\$31	3	Play Beep/Announcement "On"
\$31	4	Play Beep/Announcement "Off"
\$31	5	Play Beep/Announcement "Alarm"
\$32	WORD slot	Play announcement. Ansage-Slot abspielen.
\$33	WORD slot	Record announcement. Ansage in Slot 'slot' (0..59) Aufzeichnen mit guter Qualität.

Funktion	Parameter	Beschreibung
\$33	WORD slot+256	Record announcement (low quality). Wie vor, aber mit reduzierter Qualität und halber Datenmenge

### DTMF-Ereignisse

Ereignis	Parameter	Beschreibung
#DTMF	0..999	Die DTMF Funktion wurde per 24..24999 Eingabe über Telefon (Voice/DTMF-Anruf) aufgerufen und der DTMF-Rufannahmemodus A2 ist eingestellt.
#DTMF	0..9	Die DTMF-Taste 0..9 wurde über Telefon (Voice/DTMF-Anruf) aufgerufen und der DTMF-Rufannahmemodus A3 ist eingestellt.
#DTMF#		Die DTMF-Taste # wurde über Telefon (Voice/DTMF-Anruf) aufgerufen und der DTMF-Rufannahmemodus A3 ist eingestellt.
#DTMFS		Die DTMF-Taste * wurde im DTMF-Rufannahmemodus A3 betätigt.

### Systemvariablen

Variable	R/W	Typ	Beschreibung
DTMF	R	WORD	Falls der DTMF-Rufannahmemodus A2 eingestellt war: Letzte mit DTMF 24xxx ausgewählte Funktion per DTMF. Es wird nur der Wert xxx (ohne 24 geliefert).

## 8.3.5 Schnittstellen-Unterstützung

### Schnittstellen-Funktionen

Funktion	Parameter	Beschreibung														
\$41	WORD	Setze Baudrate und Datenbits für com=1 (V.24):														
		<table border="1"> <thead> <tr> <th>cfg Bit</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>0..7</td> <td>Baudrate DIV 1200</td> </tr> <tr> <td>8</td> <td>1: Parity Odd, 0: Parity Even</td> </tr> <tr> <td>9</td> <td>1: Parity, 0: No Parity</td> </tr> <tr> <td>10..11</td> <td>Datenbits (00=5, 01=6, 10=7, 11=8)</td> </tr> <tr> <td>12</td> <td>1: 2 Stopbits, 0: 1 Stopbit</td> </tr> <tr> <td>13</td> <td>1: RTS/CTS Flusskontrolle</td> </tr> <tr> <td>14</td> <td>1: XON/XOFF Flusskontrolle</td> </tr> </tbody> </table>	cfg Bit	Beschreibung	0..7	Baudrate DIV 1200	8	1: Parity Odd, 0: Parity Even	9	1: Parity, 0: No Parity	10..11	Datenbits (00=5, 01=6, 10=7, 11=8)	12	1: 2 Stopbits, 0: 1 Stopbit	13	1: RTS/CTS Flusskontrolle
cfg Bit	Beschreibung															
0..7	Baudrate DIV 1200															
8	1: Parity Odd, 0: Parity Even															
9	1: Parity, 0: No Parity															
10..11	Datenbits (00=5, 01=6, 10=7, 11=8)															
12	1: 2 Stopbits, 0: 1 Stopbit															
13	1: RTS/CTS Flusskontrolle															
14	1: XON/XOFF Flusskontrolle															
\$48	WORD	Set Baudrate for serial TX Port.														
...	cfg	Setze Baudrate und Datenbits für COM=8..COM=15 (Seriell TX via TTL-Port DB25)														

Funktion	Parameter	Beschreibung		
		Bit	Wert	Bedeutung
\$4F		3	0	V.24 direkt angeschlossen.
			1	Über invertierenden Pegelwandler angeschlossen.
		7..4	0000	9600, 8N1
			0001	1200, 8N1
			0010	2400, 8N1
			0011	4800, 8N1
			0100	19200, 8N1
			0101	38400, 8N1
			1000	9600, 7E1
			1001	1200, 7O1
			1010	2400, 7E1
			1011	4800, 8E1
			1100	1200, 7E1

### Schnittstellen-Ereignisse

Ereignisse	Parameter	Beschreibung.
#COM	1	Zeichen von der seriellen Schnittstelle (mit RT1,x lesen) empfangen. Wird nur aufgerufen, wenn V.24 für Remote-I/O freigegeben ist.
#KEY	31..39	Taste 1..9 wurde auf der Konsole (z.B. Tastatur) eingegeben.
	D	Taste Enter gedrückt.
	8	Taste Backspace/Cursor Left gedrückt.
	C	Taste Cursor Right gedrückt.
	18	Taste C (Clear) gedrückt.
	7F	Taste Delete/Dekrement gedrückt.
	9	Taste Insert/Inkrement gedrückt.
	1B	Taste Esc gedrückt.
	A	Taste Cursor Down gedrückt.
	B	Taste Cursor Up gedrückt.
	1E	Taste Select gedrückt.
	E,F,10	Taste F1-F3 gedrückt

### Systemvariablen

Variable	R/W	Typ	Beschreibung										
CON	R	WORD	Console-Port <table border="1" style="margin-left: 20px;"> <tr><td>0</td><td>Keine Konsole ausgewählt</td></tr> <tr><td>1</td><td>Serielle Schnittstelle (Modem)</td></tr> <tr><td>3</td><td>LCD/KBD (Default)</td></tr> <tr><td>5</td><td>ISDN-Online Verbindung</td></tr> <tr><td>6</td><td>IR/Videomonitor</td></tr> </table>	0	Keine Konsole ausgewählt	1	Serielle Schnittstelle (Modem)	3	LCD/KBD (Default)	5	ISDN-Online Verbindung	6	IR/Videomonitor
0	Keine Konsole ausgewählt												
1	Serielle Schnittstelle (Modem)												
3	LCD/KBD (Default)												
5	ISDN-Online Verbindung												
6	IR/Videomonitor												
KEY	R	WORD	Tastencode der letzten gedrückte Taste auf der aktuellen Console.										
PD	R/W	WORD	Parallel Data. Liefert die Portadresse der TTL-Datenausgänge.										

Variable	R/W	Typ	Beschreibung																											
PS	R	WORD	<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> <th>Pin</th> </tr> </thead> <tbody> <tr><td>0</td><td>PD0</td><td>2</td></tr> <tr><td>1</td><td>PD1</td><td>3</td></tr> <tr><td>2</td><td>PD2</td><td>4</td></tr> <tr><td>3</td><td>PD3</td><td>5</td></tr> <tr><td>4</td><td>PD4</td><td>6</td></tr> <tr><td>5</td><td>PD5</td><td>7</td></tr> <tr><td>6</td><td>PD6</td><td>8</td></tr> <tr><td>7</td><td>PD7</td><td>9</td></tr> </tbody> </table>	Bit	Name	Pin	0	PD0	2	1	PD1	3	2	PD2	4	3	PD3	5	4	PD4	6	5	PD5	7	6	PD6	8	7	PD7	9
			Bit	Name	Pin																									
0	PD0	2																												
1	PD1	3																												
2	PD2	4																												
3	PD3	5																												
4	PD4	6																												
5	PD5	7																												
6	PD6	8																												
7	PD7	9																												
Parallel Status. Liefert die Portadresse des TTL-Statusregisters/Eingänge.																														
PC	R/W	WORD	<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> <th>Pin</th> </tr> </thead> <tbody> <tr><td>0</td><td>Undefiniert</td><td>-</td></tr> <tr><td>1</td><td>Undefiniert</td><td>-</td></tr> <tr><td>2</td><td>Undefiniert</td><td>-</td></tr> <tr><td>3</td><td>ERROR</td><td>15</td></tr> <tr><td>4</td><td>SLCT</td><td>13</td></tr> <tr><td>5</td><td>Paper End</td><td>12</td></tr> <tr><td>6</td><td>-Acknowledge</td><td>10</td></tr> <tr><td>7</td><td>NOT Busy</td><td>NOT 11</td></tr> </tbody> </table>	Bit	Name	Pin	0	Undefiniert	-	1	Undefiniert	-	2	Undefiniert	-	3	ERROR	15	4	SLCT	13	5	Paper End	12	6	-Acknowledge	10	7	NOT Busy	NOT 11
			Bit	Name	Pin																									
0	Undefiniert	-																												
1	Undefiniert	-																												
2	Undefiniert	-																												
3	ERROR	15																												
4	SLCT	13																												
5	Paper End	12																												
6	-Acknowledge	10																												
7	NOT Busy	NOT 11																												
Parallel Control. Liefert die Portadresse des TTL-Controlports.																														
			<table border="1"> <thead> <tr> <th>Bit</th> <th>Name</th> <th>Pin</th> </tr> </thead> <tbody> <tr><td>0</td><td>NOT Strobe</td><td>NOT 1</td></tr> <tr><td>1</td><td>NOT -AFEED</td><td>NOT 14</td></tr> <tr><td>2</td><td>-Init</td><td>16</td></tr> <tr><td>3</td><td>NOT -SlctIn</td><td>NOT 17</td></tr> <tr><td>4</td><td>MASK</td><td>-</td></tr> <tr><td>5</td><td>PD OUT=1,IN=0</td><td>-</td></tr> <tr><td>6</td><td>1</td><td>-</td></tr> <tr><td>7</td><td>1</td><td>-</td></tr> </tbody> </table>	Bit	Name	Pin	0	NOT Strobe	NOT 1	1	NOT -AFEED	NOT 14	2	-Init	16	3	NOT -SlctIn	NOT 17	4	MASK	-	5	PD OUT=1,IN=0	-	6	1	-	7	1	-
Bit	Name	Pin																												
0	NOT Strobe	NOT 1																												
1	NOT -AFEED	NOT 14																												
2	-Init	16																												
3	NOT -SlctIn	NOT 17																												
4	MASK	-																												
5	PD OUT=1,IN=0	-																												
6	1	-																												
7	1	-																												

### 8.3.6 Auf- und Abbau von Verbindungen

#### Verbindungsorientierte Funktionen

Funktion	Parameter	Beschreibung						
\$50	WORD com	Hangup. Host-Verbindung beenden. <table border="1"> <tbody> <tr><td>0</td><td>Alle Verbindungen beenden.</td></tr> <tr><td>1</td><td>Disconnect Modem.</td></tr> <tr><td>5</td><td>Disconnect ISDN.</td></tr> </tbody> </table>	0	Alle Verbindungen beenden.	1	Disconnect Modem.	5	Disconnect ISDN.
0	Alle Verbindungen beenden.							
1	Disconnect Modem.							
5	Disconnect ISDN.							
\$51	TEXT nr	Dial ISDN connection. ISDN Verbindung nach <i>nr</i> wählen.						
\$52	TEXT nr	Dial Modem connection. Modem Verbindung nach <i>nr</i> wählen.						
\$54	TEXT nr	Dial ISDN2 connection. ISDN2 Verbindung nach <i>nr</i> wählen.						
\$56	TEXT nr	Dial Voice/DTMF connection. Voice Verbindung nach <i>nr</i> wählen.						

Funktion	Parameter	Beschreibung
\$57	TEXT nr	Dial to Fax. Fax Verbindung nach <i>nr</i> wählen.
\$58	TEXT nr	Dial telephone connection. Telefonie Verbindung nach <i>nr</i> wählen.
\$151... \$159	TEXT nr	Dial to ... Wie \$5x, aber in NAME stehen die Verbindungsparameter wie z.B. T4 für den Wähltimeout.

### Verbindungsorientierte Ereignisse

Ereignis	Parameter	Beschreibung	
#CONN	81	Abgehende ISDN Verbindung besteht.	
	1	Ankommende ISDN Verbindung besteht.	
	82	Abgehende/Ankommende Modem-Verbindung besteht.	
	2	Ankommende Modem-Verbindung besteht.	
	84	Abgehende ISDN2-Verbindung besteht.	
	4	Ankommende ISDN2-Verbindung besteht.	
	86	Abgehende DTMF/Voice-Verbindung besteht.	
	6	Ankommende DTMF/Voice-Verbindung besteht.	
	87	Abgehende Fax-Verbindung besteht.	
	7	Ankommende Fax-Verbindung besteht.	
	88	Abgehende Audio/Telefonie-Verbindung besteht.	
	8	Ankommende Audio/Telefonie-Verbindung besteht.	
	#DISC	1	ISDN-Verbindung beendet.
		2	Modem-Verbindung beendet.
4		ISDN2-Verbindung besteht.	
6		DTMF/Voice-Verbindung beendet.	
7		Fax-Verbindung beendet.	
8		Audio/Telefonie-Verbindung beendet.	
#ONL		0	Host-Verbindung beendet (Offline). In diesem Ereignis-Makro werden alle applikationsspezifischen Makroende-Aufrufe (z.B. durch CA'xx_NAMEe') aufgerufen.
		1	Als Sender Online mit Zugriffsberechtigung durch Host. In diesem Ereignis-Makro werden alle applikationsspezifischen Makroinitialisierungen (z.B. durch CA'xx_NAMEi') aufgerufen.
	2	Als Empfänger Online.	
	3	Als Sender Online ohne Zugriffsberechtigung.	
	4	Als Sender Online. Empfänger kann keine Makros darstellen.	

### Systemvariablen

Variable	R/W	Typ	Beschreibung						
HOST	R	WORD	PC-Hostport, falls Verbindung besteht: <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>0</td> <td>Keine Verbindung.</td> </tr> <tr> <td>1</td> <td>Modem-Verbindung.</td> </tr> <tr> <td>5</td> <td>ISDN-Verbindung.</td> </tr> </table>	0	Keine Verbindung.	1	Modem-Verbindung.	5	ISDN-Verbindung.
0	Keine Verbindung.								
1	Modem-Verbindung.								
5	ISDN-Verbindung.								
CSV	R	WORD	Current Service. Diese Bitmaske gibt an, welche Verbindungen (Dienste) aktuell zwischen Box und Netz bestehen:						

Variable	R/W	Typ	Beschreibung	
			Bit	Verbindungstyp
			1	ISDN
			2	Modem
			4	ISDN2
			6	Voice/DTMF
			7	Fax
			8	Audio/Telefonie
			Falls keine Verbindungen bestehen, ist CSV=0	
CID	R	TEXT	CallerID: Nummer des aktuellen Anrufers.	
HVER	R	WORD	Host-Version während einer Online Verbindung: HIBYTE: Major Version (GUI-Type) LOBYTE: Minor Version (Interface-Type)	
HCAP	R	WORD	Host-Capabilities während einer Online-Verbindung:	
			Bit	Capability
			0	Basic Functionality HDelta und Standard Remote I/O
			1	Remote Makros werden unterstützt
			2	VM422 basierte DCT-Dekompression wird unterstützt
			3	Telefonie Kanal wird unterstützt
ACC	R	WORD	Access Permissions zeigt an, für welche Zugriffe die Remote-Seite nach ihrer Authorisierung Zugriffserlaubnis besitzt. Die Authorisierung geschieht über Browser, DTMF-Pin Eingabe, Password-Eingabe bzw. Rufnummern/IP-Adressüberprüfung.	
			Bit	Zugriffserlaubnis
			4	Access Permission
			5	Control Permission
			6	Config Permission

### 8.3.7 Schaltausgänge und Schalteingänge

#### Systemvariable

Variable	R/W	Typ	Beschreibung
REL	R/W	WORD	Relais. Liefert die Portadresse der Relais-Ausgänge. Beispiele: Durch OR REL,4 wird der Ausgang 3 eingeschaltet, durch NA REL,4 ausgeschaltet, durch MS REL,4 geprüft, ob der Ausgang eingeschaltet ist, durch MC REL,4 geprüft, ob der Ausgang ausgeschaltet ist.

### 8.3.8 Alarme

#### Alarmierungs-Funktionen

Funktion	Parameter	Beschreibung
\$A0		Gespeicherte Alarme und Störung löschen.
\$A1	WORD	Alle Alarmspeicher von Linien deren zugeordnete Bits in <i>linien</i> gesetzt sind, werden gelöscht.



Funktion	Parameter	Beschreibung	
		Bit	Linie
	linien	0 1 2 3	1 2 3 4
\$A2	0	Unscharf Schalten	
\$A2	1	Scharf Schalten	
\$A3	WORD aktion	Alarmaktion 1..9 ausführen.	
\$A4	WORD linie	Get Alarmline Name. Name der Alarmlinie 'linie' (1..4) in der Systemvariablen NAME ablegen.	
\$AE	[TEXT]	Start Makro Event. Erzeugt ein "Startevent" vom Typ "MAKRO" für die Ereignisliste. Falls der TEXT Parameter nicht übergeben wird, wird der Name des aktiven Makros als Ereignisparameter verwendet, anderenfalls der übergebene Textparameter.	
\$1AE	[TEXT]	Stop Makro Event. Erzeugt ein "Stopevent" vom Typ "MAKRO" für die Ereignisliste. Falls der TEXT Parameter nicht übergeben wird, wird der Name des aktiven Makros als Ereignisparameter verwendet, anderenfalls der übergebene Textparameter.	

### Alarmierungs-Ereignisse

Ereignis	Parameter	Beschreibung
#ALARM	1..4	Alarmzustand auf Linie x (x=1..4) eingetreten.
#ALARMO	1..4	Alarmzustand auf Linie x (x=1..3) wurde gelöscht.
#ARMED	0 1	Unscharf geschaltet. Scharf geschaltet.

### Systemvariablen

Variable	R/W	Typ	Beschreibung								
AIN	R	WORD	Alarm Inputs. Aktueller Alarmlinien Status. Bit 0..2 entsprechen den Alarmlinien 1..3.								
AL	R	WORD	Alarm Latched. Gespeicherter Alarmstatus. Bit 0..2 entsprechen den Alarmlinien 1..3.								
ASTAT	R	WORD	Alarm Status. <table border="1"> <thead> <tr> <th>Bit</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1=Scharf, 0=Unscharf</td> </tr> <tr> <td>1</td> <td>1=Alarmrelais an. 0=Alarmrelais aus.</td> </tr> <tr> <td>2</td> <td>1=Störung</td> </tr> </tbody> </table>	Bit	Status	0	1=Scharf, 0=Unscharf	1	1=Alarmrelais an. 0=Alarmrelais aus.	2	1=Störung
Bit	Status										
0	1=Scharf, 0=Unscharf										
1	1=Alarmrelais an. 0=Alarmrelais aus.										
2	1=Störung										

### 8.3.9 CAN-I/O Unterstützung

Unterstützung des CAN-Protokolls setzt den Anschluß eines CAN-Adapters auf dem Parallelport voraus. Das CAN-Protokoll ist ein Multimaster-Protokoll (ein Busteilnehmer kann jederzeit versuchen eine Nachricht auf dem Bus abzusetzen und muss nicht darauf warten von einem Master abgefragt zu werden). Die Nachrichten bestehen unter anderem aus einer 11 Bit langen Message-ID, die in der CAN Version 1.1 2048 Werte annehmen kann, und aus einem Datenteil von 0-8 Bytes. Die Kommunikation mittels Message-IDs und Datenbytes wird als Schicht 2 (Link-Layer) Kommunikation bezeichnet.

Um Interoperabilität zwischen Geräten verschiedener Hersteller herzustellen und die Funktionsvielfalt der CAN-Komponenten strukturiert dem Anwender zur Verfügung zu stellen, wurde auf Schicht 7 (Applikations-Layer) ein Objekt-Verzeichnis mit den entsprechenden Zugriffsmethoden definiert. Ein (vor allem in Europa) gebräuchliches Modell ist CAN-Open. Zur CAN-Open Kommunikation steht ein Satz von Anweisungen zur Verfügung. Schicht 2 und Schicht 7 Kommunikation sollte innerhalb eines CAN-Systems gleichzeitig verwendet werden (obwohl es technisch möglich wäre).

Eine Beschreibung von CAN (Layer 2) und CAN-Open findet sich auf <http://www.can-cia.de>.

#### Kommandos für Layer 2 Kommunikation

Ein CAN-Message Header wird in den folgenden Anweisungen über eine 16-Bit WORD-Variable übergeben. Darin sind mehrere Header-Bestandteile codiert:

Bit 0..10 sind die CAN-Message ID, die einen Wertebereich von 0..2047 annehmen kann.

Bit 11 gibt das RTR-Bit (Remote-Request) an.

Bit 12..15 geben die Länge des Datenteils an (0-8 Bytes).

Anweisung	Param 1	Param 2	Funktion
CS	WORD	WORD	Send CAN Message. Sendet die Nachricht mit 0-2 Bytes Datenteil. Parameter 1: Message Header (ID). Sind die Bits 12..15=0, so wird eine Länge von 2 Bytes angenommen. Parameter 2: Databytes to send.
CS	WORD	TEXT	Send CAN Message. Sendet die Nachricht mit 0-8 Bytes Datenteil. Parameter 1: Message Header (ID). Sind die Bits 12..15=0, so wird die Länge aus LENGTH(TEXT) berechnet. Parameter 2: Datenbytes, die gesendet werden sollen.
CS	WORD		Send CAN Message. Sendet die Nachricht ohne Datenteil. WORD: Message Header (ID). Ist Bit 11 (RTR) = 0, so sollten auch Bits 12..15 sollen=0 sein. Ist Bit 11=1, dann wird ein Remote Request für n Bytes Daten angefordert, wobei n in den Bits 12... 15 codiert ist.
CR	WVAR		Read CAN Message. Liest die nächste CAN-Nachricht aus der Empfangswarteschlange. Parameter 1: Message Header der empfangenen Nachricht.
CR	WVAR	WVAR	Read CAN Message.

Anweisung	Param 1	Param 2	Funktion
CR	WVAR	TVAR	Liest die nächste CAN-Nachricht mit Daten (0-2 Bytes). Parameter 1: Message Header der empfangene Nachricht. Parameter 2: 1-2 Bytes CAN Message Identifier.  Read CAN Message. Liest die nächste CAN-Nachricht mit Daten (0-8 Bytes). Parameter 1: Message Header der empfangene Nachricht. Parameter 2: 0-8 Bytes Daten der Nachricht.
CR	0		CAN Test Message. Liefert TRUE, wenn Nachricht (Layer 2 oder Layer 7) verfügbar.

### Kommandos für CAN-Open Kommunikation

Anweisung	Param 1	Param 2	Funktion									
OS	WORD		CAN Open Set Output.									
OC	WORD		CAN Open Clear Output.									
OI	WORD		CAN Open Read Input. Liefert TRUE (in RES), falls Output "1" ist.									
OO	WORD		CAN Open Read Output. Liefert TRUE (in RES), falls Input "1" WORD-Bit: <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td style="padding: 2px;">8..15</td> <td style="padding: 2px;">≠ 0</td> <td style="padding: 2px;">Node Id</td> </tr> <tr> <td style="padding: 2px;">8..15</td> <td style="padding: 2px;">= 0</td> <td style="padding: 2px;">Use Default Node ID (Systemvariable NODE)</td> </tr> <tr> <td style="padding: 2px;">0..7</td> <td style="padding: 2px;"></td> <td style="padding: 2px;">Bitnummer 0..255</td> </tr> </table> Liefert \$C0xx in RES falls ein CAN Error auftrat.	8..15	≠ 0	Node Id	8..15	= 0	Use Default Node ID (Systemvariable NODE)	0..7		Bitnummer 0..255
8..15	≠ 0	Node Id										
8..15	= 0	Use Default Node ID (Systemvariable NODE)										
0..7		Bitnummer 0..255										
OW	WORD TEXT	WORD TEXT	CAN Open Write Object. Schreibt die Object Daten aus TEXT2/WORD2 Als Nodeld wird die Systemvariable NODE verwendet. WORD1: Object Index. Subindex = 0 oder TEXT1: 2 Bytes Object Index + 1 Byte Subindex. WORD2: Schreibt zwei Bytes Daten, falls WORD2 ≥ 100h ist, sonst 1 Byte Daten. Daten in WORD2. oder TEXT2: Schreibt den Text mit seiner Länge in das gewählte Objekt.									
OW	WORD TEXT	WORD TEXT	CAN Open Write Object. Schreibt die Object Daten aus TEXT/WORD2 Als Nodeld wird die Systemvariable NODE verwendet.									

Anweisung	Param 1	Param 2	Funktion
OG	WORD TEXT	WVAR TEXT	<p>WORD1: Object Index. Subindex = 0 oder TEXT1: 2 Bytes Object Index + 1 Byte Subindex WORD2: Schreibt zwei Bytes Daten, falls WORD2 ≥ 100h ist, sonst 1 Byte Daten. Daten in WORD2. oder TEXT2: Schreibt den Text mit seiner Länge in das gewählte Objekt.</p> <p>CAN Open Read Object. Liefert in TEXT/WORD2 die Object Daten. Als NodeId wird die Systemvariable NODE verwendet. WORD1: Object Index. Subindex = 0 oder TEXT1: 2 Bytes Object Index + 1 Byte Subindex WVAR: Liest ein oder zwei Bytes Daten oder TVAR: Liest die Daten in die Variable ein (momentan max. 2 Bytes).</p>

Ist RES nach Ausführung einer OW oder OG Anweisung ≠ 0, dann ist ein CAN-Fehler aufgetreten.

LOBYTE(RES)	Bedeutung
\$00	ok
\$C0...\$EF	CANST Error (Bitmaske siehe dort)
\$F0	No Response from Node
\$F1	Unknown Response from Node
\$F2	Segmented Message received and not implemented
\$F3	CAN Error

### CAN-I/O Funktionen

Funktion	Parameter	Beschreibung														
\$C0	WORD baud	<p>Set CAN/CANOpen Baudrate. Initialisiere CAN/CANOpen für Baudrate-Teiler baud.</p> <table border="1"> <thead> <tr> <th>baud</th> <th>Datenrate</th> </tr> </thead> <tbody> <tr> <td>\$0</td> <td>500kBit (Default)</td> </tr> <tr> <td>\$14</td> <td>1MBit</td> </tr> <tr> <td>\$1C</td> <td>500kBit</td> </tr> <tr> <td>\$11C</td> <td>250kBit</td> </tr> <tr> <td>\$31C</td> <td>125kBit</td> </tr> <tr> <td>\$532F</td> <td>20kBit</td> </tr> </tbody> </table> <p>Diese Initialisierung muß vor dem ersten Sende/Empfangs-Befehl bzw. nach Power-Up des CAN-Adapters einmal aufgerufen werden (z.B. im #RESET1 Makro).</p>	baud	Datenrate	\$0	500kBit (Default)	\$14	1MBit	\$1C	500kBit	\$11C	250kBit	\$31C	125kBit	\$532F	20kBit
baud	Datenrate															
\$0	500kBit (Default)															
\$14	1MBit															
\$1C	500kBit															
\$11C	250kBit															
\$31C	125kBit															
\$532F	20kBit															
\$1C0	[WORD node]	<p>Reset CAN Node. Ist <i>node</i>=0 oder nicht angegeben, wird auf allen angeschlossenen Knoten ein Reset durchgeführt. Diese Funktion sollte einmal im #RESET1 Makro aufgerufen werden.</p>														
\$2C0	[WORD	Start Node.														

Funktion	Parameter	Beschreibung
	node]	Ist <i>node</i> =0 oder nicht angegeben, wird die Systemvariable <i>NODE</i> verwendet.
\$C1	WORD mask	Set Message Filter Mask. In <i>maske</i> kann eine Bitmaske für die akzeptieren CAN-Nachrichten gesetzt werden, damit nicht alle Nachrichten ein Makro-Ereignis auslösen. LOBYTE(mask): Bit 3...11: Acceptance Code. HIBYTE(mask): Bit 3...11: Acceptance Mask. Voreingestellt ist AcceptanceCode/AcceptanceMask=All. Für CAN-Open Kommunikation sollte der Message Filter nicht verwendet werden, sondern nur für CAN Layer 2 Kommunikation. Die Bitbelegung entspricht dem des CAN-Headers (8.3.9)
\$1C2		Reset CAN Module. Der Software CAN-Treiber wird initialisiert.
\$C2		Clear CAN Receive Buffer. Der CAN-Empfangsbuffer wird gelöscht.

### CAN-I/O Ereignisse

Ereignis	Parameter	Beschreibung
#CANIN	nodeid	CAN Open Input on Node <i>nodeid</i> changed (TX PDO 1/2 received)
#CANERR	nodeid	CAN Open Error (Emergency) Message from <i>nodeid</i> received.
#CANMSG	msgid	CAN Layer 2 Message <i>msgid</i> received.

### Systemvariablen

Variable	R/W	Typ	Beschreibung														
CANID	R	WORD	CAN Message der nächsten anstehenden Nachricht (Seite 8.3.9). Enthält den Wert 0FFFh, falls keine Nachricht verfügbar.														
CANST	R	WORD	CAN Status <table border="1"> <thead> <tr> <th>Bit</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Transmitter Full</td> </tr> <tr> <td>1</td> <td>Receiver Empty</td> </tr> <tr> <td>2</td> <td>Receiver Overrun</td> </tr> <tr> <td>3</td> <td>Bus Error</td> </tr> <tr> <td>4</td> <td>Bus Off</td> </tr> <tr> <td>5</td> <td>Register Test Error</td> </tr> </tbody> </table>	Bit	Status	0	Transmitter Full	1	Receiver Empty	2	Receiver Overrun	3	Bus Error	4	Bus Off	5	Register Test Error
Bit	Status																
0	Transmitter Full																
1	Receiver Empty																
2	Receiver Overrun																
3	Bus Error																
4	Bus Off																
5	Register Test Error																
NODE	R/W	WORD	CAN Node ID für CAN Layer 2 Kommunikation. Voreingestellte NODE-Id ist 1. Durch <i>SENODE</i> ,3 kann die Node-ID auf 3 verändert werden. Die Systemvariable <i>NODE</i> wird immer dann verwendet, wenn keine NodeID in den CAN-Open Anweisungen angegeben wurde oder "0" ist.														

### 8.3.10 Unterstützung von Benutzer-Interaktion

#### Ereignisse zur Benutzer-Interaktion

Ereignisse, die von einer Remote-seitigen Bedienoberfläche (z.B. auf einem PC) an die pt-200 gesendet werden, besitzen per Namenskonvention als erstes Zeichen des Namens ein \$-Symbol. Der Name des Dialog-Ereignisses wird beim Erzeugen des Dialogelementes (Anweisung MD) übergeben. Abhängig vom Dialog wird noch ein Parameter zur Charakterisierung des Ereignisses angehängt (siehe Dialogelement auf Seite 30).

Weiterhin gibt es noch Ereignisse, die bei Videoverbindung durch Mausklicks in das Bildfenster aufgerufen werden:

Ereignis	Parameter	Beschreibung
#PICM		Während der Online-Verbindung am Host in das dargestellte Bild geklickt. In der Systemvariablen PAR wird die Position des Mausklicks übergeben: LOBYTE=X-Block (0..95) HIBYTE=Y-Block (0..71) Der Wert ist unabhängig von der Auflösung, Vergrößerung und Teilbild-Darstellung. Bei einer PAL-Auflösung von 786x576 Punkten wird durch 8 geteilt, bei 96x72 Pixel Großen Bilder wird jeder Punkt adressiert.
	1	Wie PICM aber ins Bild geklickt und Taste bleibt gedrückt (Move-Funktion).
	2	Wie PICM aber als Doppelklick.
	3	Wie PICM aber Taste wurde nach Move losgelassen. Die Position, die übergeben wird, ist die Position des ersten Tastendrucks von "Move".

### 8.3.11 Zeitabhängige Steuerung

#### Zeitabhängige Ereignisse

Ereignis	Parameter	Beschreibung
#HOUR	0..23	Stundenwechsel auf die übergebene Stunde.

#### Systemvariablen

Variable	R/W	Typ	Beschreibung
DOW	R	WORD	Day of Week (Wochentag).
			0   Sonntag
			1   Montag
			2   Dienstag
			3   Mittwoch
			4   Donnerstag
			5   Freitag
6   Samstag			
WDAY	R	TEXT	Weekday, Aktueller Wochentagsname. So, Mo, Di, Mi, Do, Fr, Sa
SEC	R	WORD	Aktuelle Sekunde (0..59, Nur gerade Sekunden!)
MIN	R	WORD	Aktuelle Minute (0..59)
HOUR	R	WORD	Aktuelle Stunden /0..23)

Variable	R/W	Typ	Beschreibung
DAY	R	WORD	Aktueller Tag (1..31)
MON	R	WORD	Aktueller Monat (1..12)
YEAR	R	WORD	Aktuelles Jahr (1980..2050)
TIME	R	TEXT	Aktuelle Zeit im Format 15:30:23
DATE	R	TEXT	Aktuelles Datum im Format 22.07.1997

## 8.4 GUI-Dialoge

GUI-Dialoge werden durch die MD-Anweisung (Make Dialog) (siehe Seite 8) während der Online-Verbindung auf der grafischen Oberfläche des Bildempfängers angezeigt. Die MD-Anweisung erwartet als ersten Parameter den Namen des Dialogelementes, der nicht länger als 10 Zeichen sein soll. Dieser Name wird bei Interaktion mit dem entsprechenden Dialogelement als Makro-Ereignis an pt-200 geschickt. Als Namenskonvention wurde vereinbart, den Makro/Dialog/Ereignisnamen mit einer Tilde ~ zu beenden.

Als zweiter Parameter wird der MD-Anweisung der Typ des zu erzeugenden Dialogelementes und optionale Gestaltverändernde Parameter übergeben. Die Parameterliste besteht aus Attribut/Wert-Paaren der Form <Attribut>:<Wert>. Die einzelnen Attribut/Wert Paare werden mit Semikolon getrennt.

Beispiel: MD'ptLicht ','PB:Licht;W:3;H:3' ;

### 8.4.1 Farbwerte

R,RED	Rot
G,GRN	Grün
B,BLU	Blau
Y,YEL	Gelb
W,WHI	Weiß
B,BLA	Schwarz
M,MAG	Magenta (Violett)
MED	Mittelgrau
LIG	Hellgrau
DAR	Dunkelgrau
0	Schwarz
1	Blau
2	Grün
3	Cyan(Türkis)
4	Rot
5	Magenta(Violett)
6	Braun
7	Hellgrau
8	Dunkelgrau
9	Mittelgrau
13	Gelb
14	Weiß

### 8.4.2 Anordnung der Dialoge

Die Dialogelemente werden an ein vorgegebenes Raster ausgerichtet (z.B. 16 Bildschirmpixel pro Rastereinheit (Gridunit)). Ein Dialog-Rasterfeld kann z.B.20 Rastereinheiten breit und 5 Rastereinheiten

hoch sein. Für jedes Dialogelement wird ein freies Dialogfeld gesucht, das den Dialog (dessen Größe durch Breite und Höhe bestimmt wird) aufnehmen kann. Falls kein freier Platz mehr gefunden wird, wird das Dialogelement nicht dargestellt. Die Suche nach einem freien Platz wird in der Reihenfolge "von oben nach unten" und dann "von links nach rechts" durchgeführt. Durch die X und Y Attribute hat der Benutzer die Möglichkeit, die Position des Dialogelementes auf dem Rasterfeld selbst zu bestimmen.

Die Aussenabmessung (Outline) eines Dialoges wird aus der Überschriftenbreite, seiner Orientierung (Horizontal/Vertikal) und z.B. der Zahl der Checkbuttons, Radiobuttons oder der Displayfeldbreite berechnet, kann aber auch vom Benutzer durch die W und H Attribute vorgegeben werden.

### 8.4.3 Pushbutton Dialog

#### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
PB	Name	Der Name ist die Button-Beschriftung
W	1...	Breite des Dialogelementes in Grid-Units
H	1...	Höhe des Dialogelementes in Grid-Units
X	0...	X-Position der linken oberen Ecke des Dialogs
Y	0...	Y-Position der linken oberen Ecke des Dialogs

#### Wirkung der WD-Anweisung

WD übergibt ein WORD mit 0000 oder FFFF für TRUE bzw. FALSE. Der Button wird bei TRUE farblich hervorgehoben.

#### Dialogereignis-Parameter

Als Ereignis bei Drücken der Taste wird geliefert:

DialogName	Falls Taste normal angeklickt
DialogName1	Falls Taste lange gedrückt bleibt
DialogName2	Falls Doppelclick auf Taste
DialogName3	Falls Taste nach langem Drücken losgelassen wird

#### Wirkung der QD-Anweisung

Durch die QD-Anweisung kann der aktuelle Zustand des Buttons abgefragt werden. Der Wert 0000 oder FFFF wird gelesen.

### 8.4.4 Radiobutton Dialog

#### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
RB	Name	Der Name ist die Button-Gruppen-Überschrift. Der Name ist optional.
W	1...	Breite des Dialogelementes in Grid-Units
H	1...	Höhe des Dialogelementes in Grid-Units
X	0...	X-Position der linken oberen Ecke des Dialogs
Y	0...	Y-Position der linken oberen Ecke des Dialogs
O	H/V	Orientierung Horizontal/Vertikal.
B0... B15	Name	Ein Name zur Beschriftung des Radiobuttons, für das angegebene Bit



### Wirkung der WD-Anweisung

WD übergibt den aktuellen Wert der Buttons als Bitmaske, wobei ein gesetztes Bit anzeigt, daß der Radiobutton markiert ist. Es sollte genau ein Bit gesetzt sein.

### Dialogereignis-Parameter und QD-Anweisung

Bei Betätigen eines Radio-Buttons dieser Button-Gruppe wird ein Ereignis mit dem Namen des Dialogelementes erzeugt. In PAR oder mit der QD-Anweisung kann das Ereignismakro den Status der Radiobuttons abfragen (es ist das Bit gesetzt, das einem eingeschalteten Radiobutton entspricht).

## 8.4.5 Checkbox Dialog

### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
CB	Name	Der Name ist die Button-Gruppen-Überschrift. Der Name ist optional.
W	1...	Breite des Dialogelementes in Grid-Units
H	1...	Höhe des Dialogelementes in Grid-Units
X	0...	X-Position der linken oberen Ecke des Dialogs
Y	0...	Y-Position der linken oberen Ecke des Dialogs
O	H/V	Orientierung Horizontal/Vertikal.
B0... B15	Name	Ein Name zur Beschriftung der Checkbox, für das angegebene Bit

### Wirkung der WD-Anweisung

WD übergibt den aktuellen Wert der Buttons als Bitmaske, wobei ein gesetztes Bit anzeigt, daß der Checkbutton markiert ist.

### Dialogereignis-Parameter und QD-Anweisung

Bei Betätigen eines Check-Buttons dieser Button-Gruppe wird ein Ereignis mit dem Namen des Dialogelementes erzeugt. In PAR oder mit der QD-Anweisung kann das Ereignismakro den Status der Checkbuttons abfragen. Jeder aktivierte Checkbutton entspricht einem gesetztem Bit.

## 8.4.6 Entry Text, Texteingabefeld

### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
ET	Name	Der Name ist die Eingabefeld-Überschrift. Der Name ist optional.
W	1...	Breite des Dialogelementes in Grid-Units
H	1...	Höhe des Dialogelementes in Grid-Units
X	0...	X-Position der linken oberen Ecke des Dialogs
Y	0...	Y-Position der linken oberen Ecke des Dialogs
LEN	0... 140	Max. Stringlänge (wird notfalls rechtsbündig mit Leerzeichen aufgefüllt). Voreingestellt ist 140 Zeichen.

### Wirkung der WD-Anweisung

WD übergibt den aktuellen Wert des Eingabefeldes als Text.

### Dialogereignis-Parameter und QD-Anweisung

Nach Verändern des Textfeldes wird das dem Dialogelement entsprechende Ereignismakro aufgerufen. In NAME oder mit der QD-Anweisung kann der Text des Eingabefeldes abgefragt werden.

## 8.4.7 Entry Number, Numerisches Eingabefeld

### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
EN	Name	Der Name ist die Eingabefeld-Überschrift. Der Name ist optional.
W	1...	Breite des Dialogelementes in Grid-Units
H	1...	Höhe des Dialogelementes in Grid-Units
X	0...	X-Position der linken oberen Ecke des Dialogs
Y	0...	Y-Position der linken oberen Ecke des Dialogs
MIN	0...65535	Minimal erforderlicher Eingabewert.
LIM	0...65535	Maximal erlaubter Eingabewert.

### Wirkung der WD-Anweisung

WD übergibt den aktuellen Wert des Eingabefeldes als numerischen Wert.

### Dialogereignis-Parameter und QD-Anweisung

Nach Verändern des Textfeldes wird das dem Dialogelement entsprechende Ereignismakro aufgerufen. In PAR oder mit der QD-Anweisung kann der Wert des Eingabefeldes abgefragt werden.

## 8.4.8 Slider, Schieberegler Dialog

Der Schieberegler zeigt mit prozentualer Darstellung (0-100%) einen Wert aus dem Bereich 0..255 an und verändert ihn bei Betätigung.

### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
SL	Name	Der Name ist die Überschrift des Schiebereglers. Der Name ist optional.
W	1...	Breite des Dialogelementes in Grid-Units
H	1...	Höhe des Dialogelementes in Grid-Units
X	0...	X-Position der linken oberen Ecke des Dialogs
Y	0...	Y-Position der linken oberen Ecke des Dialogs
O	V/H	Gibt an, ob ein horizontaler oder vertikaler Regler verwendet wird.

### **Wirkung der WD-Anweisung**

WD übergibt den aktuellen Wert des Schiebereglers als numerischen Wert im Bereich 0...255. Nur das Lowbyte wird ausgewertet.

### **Dialogereignis-Parameter und QD-Anweisung**

Nach Verändern des Schiebereglers wird das dem Dialogelement entsprechende Ereignismakro aufgerufen. In PAR oder mit der QD-Anweisung kann der Wert des Reglers (0..255) abgefragt werden.

## **8.4.9 LED, Optische Anzeige**

### **Parameter der MD-Anweisung**

Attribut	Wert	Beschreibung
LD	Name	Der Name dient zu Beschriftung der LED.
X	0...	X-Position der linken oberen Ecke der LED
Y	0...	Y-Position der linken oberen Ecke der LED
COL	farbe	Farbe der LED im aktiven Zustand (Default: Grün)

### **Wirkung der WD-Anweisung**

WD übergibt 0000 (TRUE), falls die LED in der vorgewählten Farbe eingeschaltet werden soll, FFFF (FALSE), falls die LED ausgeschaltet werden soll und einen anderen Wert, um die LED in einer ausgewählten Farbe einzuschalten.

### **QD-Anweisung**

Da keine Werteveränderung dieses statischen Dialogelementes stattfinden kann, wird kein Ereignismakro für die LED aufgerufen. Die QD-Anweisung liefert den Status der LED als numerischen Wert: 0000 für "an" und FFFF für "aus".

## 8.4.10 Numerical Display, Numerische Anzeige

### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
ND	Name	Der Name ist die Überschrift des Display. Der Name ist optional.
W	1...	Breite des Dialogelementes in Grid-Units
H	1...	Höhe des Dialogelementes in Grid-Units
X	0...	X-Position der linken oberen Ecke des Dialogs
Y	0...	Y-Position der linken oberen Ecke des Dialogs
COL	farbe	Segmentfarbe (Default: Grün)
BG	farbe	Hintergrundfarbe (Default: Schwarz)
STY	7	Momentan ist nur 7 (7-Segment Display) definiert.
LEN	1...	Maximale Zahl von Ziffern für die Anzeige (Default: 4 für Hex, 8 für Binär, 5 für Dezimal)
F	S	Signed Format. Interpretiert Wert als vorzeichenbehaftete Zahl.
	Z	Leading Zeroes. Zeigt führende Nullen an.
	H	Hexadezimal. Zeigt Hexadezimalzahlen an.
	D	Dezimal. Zeigt Dezimalzahlen an.
	B	Binär. Zeigt Binärzahlen an.
	L	Large. Doppelte Größe.
	0	Decimal point xxxxx. (Default: No decimal point)
	1	Decimal point xxxx.y
	2	Decimal point xxx.yy
	3	Decimal point xx.yyy
	4	Decimal point x.yyyy
	5	Decimal point .yyyy
UN	chars	Unit (Default: None). Einheitenzeichen, die hinter das Anzeigefeld geschrieben werden. Example UN:C

### Wirkung der WD-Anweisung

WD übergibt den numerischen Wert, der in das Display geschrieben werden soll.

### QD-Anweisung

Da keine Werteveränderung dieses statischen Dialogelementes stattfinden kann, wird kein Ereignismakro für das Display aufgerufen. Die QD-Anweisung liefert den numerischen Wert, der im Display steht.

## 8.4.11 Text Display, Alphanumerische Anzeige

### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
TD	Name	Text Display (Name als Überschrift optional)
W	1...	Breite des Dialogelementes in Grid-Units
H	1...	Höhe des Dialogelementes in Grid-Units
X	0...	X-Position der linken oberen Ecke des Dialogs
Y	0...	Y-Position der linken oberen Ecke des Dialogs
COL	farbe	Textfarbe (Default: Grün)
BG	farbe	Hintergrundfarbe (Default: Schwarz)
F	Vn/Bn	Zeichensatz-Typ: V=Vektor Font, B=Bitmap Font. Optional folgt nach dem Font-Typ folgt die Fontgröße als Ziffer zwischen 1 und 9.

### Wirkung der WD-Anweisung

WD übergibt den Text, der in das Display geschrieben werden soll.

### QD-Anweisung

Da keine Werteveränderung dieses statischen Dialogelementes stattfinden kann, wird kein Ereignismarko für das Display aufgerufen. Die QD-Anweisung liefert den Text, der im Display steht.

## 8.4.12 Static Text, Beschriftungstext

### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
ST	Name	Der Name enthält den Beschriftungstext
H	1...	Textgröße und Höhe in Grid-Units
X	0...	X-Position der linken oberen Ecke des Textes
Y	0...	Y-Position der linken oberen Ecke des Textes
COL	farbe	Textfarbe (Default: Dunkelgrau)

### Wirkung der WD-Anweisung

Obwohl dieses Dialogelement einen statischen Text repräsentiert, der also nicht verändert werden sollte, kann mit der WD-Anweisung eine Veränderung des Textes vorgenommen werden. Da sich dadurch aber dessen Länge ändert und so evtl. andere Dialogelemente überschrieben werden können, sollte besser ein Text-Display verwendet werden.

### QD-Anweisung

Da keine Werteveränderung dieses statischen Dialogelementes stattfinden kann, wird kein Ereignismarko aufgerufen. Die QD-Anweisung liefert den statischen Text zurück.

## 8.4.13 Message Text, Mitteilungstext

Der Message Text ist kein sichtbares Dialogelement, benötigt also keinen Platz auf der Dialog-Rasterfläche. Der übergebene Text wird als Hinweistext auf den Bildschirm des Benutzers geschrieben.

### Parameter der MD-Anweisung

Attribut	Wert	Beschreibung
MT	Name	Der Name enthält einen Prefix für das Message-Window. Der Name ist optional.

### Wirkung der WD-Anweisung

Der Text, der der WD-Anweisung an dieses Dialogelement übergeben wird, wird mit dem Prefix des Dialogelementes dem Benutzer z.B. in Form einer Ausgabe in ein Mitteilungsfenster präsentiert.

### QD-Anweisung

Die QD-Anweisung liefert ein undefiniertes Ergebnis.

## 8.4.14 LogFile, Virtueller Dateidialog

Logfile ist ein "virtueller" Dialog, der auf Dateien operiert, die auf dem Bildempfangsgerät gespeichert sind. Dieser Dialog benötigt keinen Platz auf dem Dialograsterfeld.

Beispiel:

```
MD '~pt_lf', 'LF:r1.dat;MO:xxxx;LEN:100' ;
```

### Parameter der MD-Anweisung

Die MD-Anweisung meldet die Datei für Dateioperationen der pt-200 auf dem Host (z.B. Empfänger PC mit ptWin-Bediensoftware) an. Nach dem Mode-Attribut dürfen mehrere Modusauswahl Buchstaben gelistet werden (ohne Trennzeichen).

Attribut	Wert	Beschreibung
LF	Name	Als Name wird der Dateiname der zu verwendenden Datei auf dem lokalen Dateisystem des Benutzers (Hosts) angegeben. Wird kein Name angegeben, wird der Makroname (ohne Sonderzeichen) als Dateiname verwendet. Falls keine Dateinamenerweiterung (.TXT oder ähnlich) nach LF: angegeben wurde, wird der Dateiname mit .TXT erweitert, falls die Datei als Text-Datei geöffnet wird (Default) und mit .DAT erweitert, falls es sich um eine Datendatei handelt.
MO	R/W	Open File for Read or Write (Default)
	D/T	Open File for Data (kein CR+LF) or Text (Default, jede Zeile wird mit CR+LF abgeschlossen)
	L/F	Linemode/Fieldmode: L Linemode (Default): Jeder WD-Befehl wird als eigener Datensatz in die Datei geschrieben. Im FixedRecord Mode (LEN ≠ 0) wird mit dem Füllzeichen auf die gewünschte Zeilen/Recordlänge aufgefüllt. Bei Textdateien wird CR+LF automatisch angefügt. F Fieldmode: Jeder WD-Befehl schreibt nur die übergebene Nummer/Zeichen in die Datei. Ein Zeilenende/Recordende wird mit WD'...' (also einem leeren 2.Parameter) geschrieben.
	A/C	Append/Create File: A Append (Default). Ein Schreibzugriff fügt an das Dateiende an.

Attribut	Wert	Beschreibung								
	B	C Create file. Datei wird bei MD neu angelegt (gelöscht). Buffered Mode (Datei wird nur beim KD oder beim Verbindungsaufbau geschlossen. Dadurch können keine fremdem Programme darauf zugreifen. Die Dateioperationen sind schneller). Default: Unbuffered Mode. Die Datei wird nach jedem Zugriff geschlossen.								
	0...3	Directory: <table border="1"> <tr> <td>0</td> <td>Datei wird im Verzeichnis PicturePath/ObjName/Datum/... angelegt (Default). Hier werden die Presence-Log-Dateien angelegt.</td> </tr> <tr> <td>1</td> <td>Datei wird im Verzeichnis LogPath/... angelegt. Hier werden die Presence-Log-Dateien angelegt.</td> </tr> <tr> <td>2</td> <td>Datei wird im Verzeichnis PicturePath/ObjName/... angelegt. Dies ist auch die Voreinstellung.</td> </tr> <tr> <td>3</td> <td>Datei wird im Verzeichnis PicturePath/... angelegt.</td> </tr> </table> Falls kein Dateiname nach LF: angegeben wurde, wird im Falle Directory=0 die aktuelle Uhrzeit HHMMSSSt mit Hundertstel-Sekunden als Dateiname verwendet, ansonsten der Makroname des Dialogelementes für LF. Zusätzliche Directoryangaben im Dateinamen nach LF beziehen sich immer auf das mit 0-3 festgelegte Verzeichnis und nicht auf das Hauptverzeichnis des PCs (aus Sicherheitsgründen).	0	Datei wird im Verzeichnis PicturePath/ObjName/Datum/... angelegt (Default). Hier werden die Presence-Log-Dateien angelegt.	1	Datei wird im Verzeichnis LogPath/... angelegt. Hier werden die Presence-Log-Dateien angelegt.	2	Datei wird im Verzeichnis PicturePath/ObjName/... angelegt. Dies ist auch die Voreinstellung.	3	Datei wird im Verzeichnis PicturePath/... angelegt.
0	Datei wird im Verzeichnis PicturePath/ObjName/Datum/... angelegt (Default). Hier werden die Presence-Log-Dateien angelegt.									
1	Datei wird im Verzeichnis LogPath/... angelegt. Hier werden die Presence-Log-Dateien angelegt.									
2	Datei wird im Verzeichnis PicturePath/ObjName/... angelegt. Dies ist auch die Voreinstellung.									
3	Datei wird im Verzeichnis PicturePath/... angelegt.									
LEN	0	Variable Length Record/Line Mode (Default). Record und Zeilenlängen sind so kurz wie nötig.								
LEN	1...	Fixed Record length Mode. Jeder Record/Jede Zeile wird auf die hier angegebene Länge mit Füllzeichen aufgefüllt. Bei Text-Dateien ist das Füllzeichen Space, bei Daten-Dateien ist es 00. Falls die aktuelle Recordlänge bereits vor dem Zeilenende länger ist als LEN, wird kein Füllzeichen angefügt.								

### WD-Anweisung

Im *WriteMode* werden der zweite Parameter als Zwei-Byte Zahl oder Text in die Datei geschrieben. In *LineMode* bis eventuell bis zur gewünschten Record/Zeilenlänge mit Füllzeichen aufgefüllt oder im *Field-Mode* so belassen.

Im *ReadMode* hat WD eine andere Bedeutung: Der übergebene (WORD)-Parameter positioniert

- im *Textmode* auf Zeile n (Die erste Zeile einer Datei ist Zeile 1!)
- im *Fixed Recordmode* auf Record n (Der erste Record ist Record 0), falls  $LEN \neq 0$  und
- im *Variable Recordmode* ( $LEN=0$ ) auf Byte n.

Da n max. 64k groß werden kann, ist diese Positionierung auf die ersten 64k einer Datei beschränkt.

Ein nun folgender QD Befehl liest die Daten ab der vorgewählten Position. Nach dem Dateiöffnen (MD-Befehl) steht der Lesezeiger am Dateianfang.

### QD-Anweisung

Die QD-Anweisung liest aus einer Datei, sofern sie zum Lesen geöffnet wurde (sequentiell vom Anfang an). Im *ReadMode* wird der *BufferedMode* und der *FieldMode* ignoriert. Die Datei bleibt immer geöffnet. Es werden immer ganze Records/Zeilen gelesen. Bei Textdateien wird kein CR+LF übertragen.

### KD-Anweisung

Diese Anweisung schließt die Datei, löscht sie aber nicht.