

ASCII-Übertragung von Konfigurationsdaten

History:

Version 1.00

Nur Einzelabfrage, Produkt-Kanalzuordnung 1:1

Version 1.10:

Gruppenabfrage : es können alle verwendeten Elemente einer Gruppe nacheinander abgefragt werden

Version 1.20:

1.20.1 Aufgabengruppe

Aufgabenbereiche werden mit der neuen Gruppe "A", Aufgaben, übermittelt. Mit "AN" wird eine Bezeichnung, z.B. "Bonieren"; mit "AL" eine Liste der dezimal klassifizierten Aufgaben übertragen.

Bis zu 10 Aufgabenbereiche können übertragen werden. Sie sind für Kellnerberechtigungen und Gerätefunktionen vorgesehen.

1.20.2 Kellneraufgaben

Die Syntax von "KA" wird erweitert. Die Mnemonic ist "Kellneraufgaben" statt "Kellnerart". Als drittes Argument wird die Menge der dem Kellner zugeordneten Aufgabenbereiche übertragen. Die Bereiche sind durch "+" getrennt, z.B.

KA1,1,2+4 - Kellnerart verwendet, frei, Aufgabenbereiche 2 und 4

Bei Übertragung in Formaten vor ASCII-I/O 1.20 wird die Kellnerart entsprechend interpretiert. Ab Version 1.20 wird als Kellnerart nur 0 (nicht verwendet) oder 1 (verwendet) übertragen; die Angaben "Schank" und "Handbetrieb" ergeben sich aus den Aufgabenbereichen.

Version 1.30:

1.30.1 Abfrage der unterstützten Baudrates.

Der Master kann die vom Slave unterstützten Baudrates abfragen, und zwar mit <DC1> #B. Die Antwort lautet <DC2> #B { Baudratel, ... }.

1.30.2 Dosierer-Typen.

Für Dosierer werden die beiden Typen "Zapfkopf" (1) und "Gaby" (2) festgelegt.

3) Gruppenübertragung vom Slave (Gerät) zum Master (PC)

Master	Slave
<DC1> <Geräte-ID> <Gruppenanfrage>	
<ACK>	<DC2> <Geräte-ID> <Daten>
<ACK>	<DC2> <Geräte-ID> <Daten>
:	:
<ACK>	<DC2> <Geräte-ID> <Daten>
	<ACK>

Mit dem letzten <ACK> meldet das Gerät, daß alle Daten aus der Gruppe übertragen sind. Die Übertragung kann bereits vorher vom Master mit <ESC> beendet werden, es wird vom Slave mit <ACK> beantwortet.

Der erste Satz, den das Gerät an den PC sendet, soll immer die folgenden beiden Befehle enthalten:

"X!"	Gruppe Löschen
"X\$" <Adressen>	Gruppenadressen

4) Datenformate:

Ganzzahlen	bestehen aus mindestens einer Ziffer; auch für Null muß die Ziffer 0 angegeben werden.
Reele Zahlen	bestehen aus einem optionalen Vorzeichen, Ziffern und einem Dezimalpunkt. Ein Komma darf nicht verwendet werden. Einige Geräte erwarten eine feste Anzahl von Nachkommastellen.
Zeichenketten	Texte stehen in doppelten Anführungszeichen ("). Es wird der IBM-Zeichensatz mit 256 Zeichen verwendet, wobei die Kontrollzeichen (Code 0 bis 31) nicht verwendet werden dürfen.
Uhrzeiten	bestehen aus zwei oder drei durch Doppelpunkt getrennten Zahlen, deren erste die Stunde angibt. Beispiele: 08:15 8 Uhr 15 07:13:12 7 Uhr 13 Minuten 12 Sekunden
Datumsangaben	bestehen aus drei durch Schrägstrich getrennten Ganzzahlen für Jahr, Monat und Tag Beispiel: 1993/01/19
Intervalle	Bereiche werden durch Angabe von Minimum, dem "-" Zeichen und dem Maximum angegeben Beispiel: KT3-99 Kellner Tische 3 bis 99
Mengen	Mengen werden durch Aufzählen der Elemente, getrennt durch das Zeichen "+", oder als hexadezimalen Äquivalent einer Bitmenge angegeben. Beispiel: PV2+7 Produkt Ventile 2 und 7 Ab Version 1.40 werden Mengen auch in geschweiften Klammern angegeben. Intervalle (mit Trennzeichen "-") und Einzelelemente werden durch Kommata getrennt. Beispiel: @V{1,3-5,8} Ventilgruppen 1, 3, 4, 5, 8
Selektoren	Selektoren werden mit dem Zeichen "=" vom nachfolgenden selektierten Feld getrennt.

Beispiel:

PL2=0.8 Produkt Lagerkonto 2, Faktor 0.8

Einheiten

Die folgenden Einheiten werden verwendet:

s - Sekunden
I - Impulse

Trennungen

Argumente werden durch ein Komma getrennt.

Beispiel: PL2=0.8,5=0.7

Befehle werden durch Strichpunkt getrennt.

Beispiel: P#12;PN"Gin"

Produkt-Nr. 12, Produktname "Gin"

Für die Überprüfung und ggf. das Verwerfen der Daten ist jeweils der Empfänger zuständig. Das betrifft besonders -

- das Verwerfen von am Gerät nicht verwendeten Datenobjekten
- das Abschneiden zu langer Zeichenketten vor der Ablage
- das Ignorieren von Argumenten außerhalb des am Gerät gültigen Bereichs

5) Daten-ID und Datenfelder - Übersicht.

ID - Befehl	Daten
#B - Baudrate setzen	Baudrate
#D - Datum und Uhrzeit	Datum, Uhrzeit
#S - Software-Ident.	Name, Version, Datum
#H - Hardware-Ident.	Name, Version, Datum
#P - Protokoll-Ident.	Name, Version, Datum
#_ - Gruppenvorgabe	Bezeichner
** - Kommentar	beliebiger Kommentar
_\$ - Bereich	für alle Bezeichner
_! - löschen	für alle Bezeichner
_# - record auswähler	für alle Bezeichner
A# - Nummer	
AL - Aufgabenliste	Liste
AN - Aufgabenname	Text
B# - BitMap Nummer	
BF - BitMap Format;	
BP - BitMap Product;	
BD - BitMap Daten	
CA - Kanalart	Art
CC - Kanal-Code	Code
CD - Dosis	Folge-Dosen
CE - Eingänge	Folge-Eingänge
CF - Faktor	Gewichtsfaktor
CM - Timeout	Folge-Timeouts
CP - Produkt	Produktnummer
CT - Terminal	Terminalnummer
CV - Ventile	Ventile
CW - Pause	(noch nie verw.)
CQ - Sollmenge	
E! - ClearErrors	
E# - number	
EN - text	
EI - input	
EP - level/polarity	
FF - Formularfuß	Zeilen-Nr, Text
FK - Formularkopf	Zeilen-Nr, Text
FT - Formulartext	Zeilen-Nr, Text
FB - Buchungszeile	
GF - Geräte Flags	Flags
GK - Gerätekanal	Kanal, Produkt
GP - Geräte Protokoll	Portnummer
GT - Gerätetype	Type
GN - Gerätename	
KA - Kellneraufgaben	Art, Flags, A-Bereiche
KE - Kellner Ebene	Zeiten, Preisebenen
KN - Kellnername	Text
KT - Kellner Tische	Bereich(e)

LE - Lagereinheit	Text
LN - Lagerkonto Name	Text
LS - Lagerstand	Stand (reel)
LV - Lager-Verbrauch	Verbrauch (reel)
PA - Produktart	Art, Flags
PB - Boncode	Boncode (Drucker)
PC - Barcode	
PD - Produktdosis	Menge, Einheit
PE - Produktfreigabe	Anzahl
PI - Produkt Info	Text
PL - Produkt-Lager	Konten & Faktoren
PM - Maximalzeit	Timeout(s)
PN - Produktname	Text
PO - Farbe	
PP - Produktpreis(e)	1 oder 2 Preis(e)
PR - Produkt-Rezeptur	Komponenten & Faktoren
PS - Produktparte	Sparte
PT - Produkttaste(n)	1 oder 2 Taste(n)
PV - Produktventile	Ventile
PW - Pause	
PQ - Sollmenge	
Q# - Caratfunktionen Nr	
QN - Name	
QF - Funktionsnummer	
QA - Aufgabe	
SA - Sparten-AlSt-Satz	Satz in Prozent
SB - Sparten-Bedg-Satz	Satz in Prozent
SG - Sparten-GeSt-Satz	Satz in Prozent
SM - Sparten-MwSt-Satz	Satz in Prozent
SN - Spartenname	Text
T! - TextParameter	
TPname Textparameter	
U! - Kunden löschen	
U# - Kundennummer	
UA - Kundenname	
UD - Kundendaten	
UG - Datum	
VA - Ventilart	Art
VG - Ventilgruppe	Gruppe
VN - Ventilname	Text
VV - Ventil-Volumsgeber	Geber-Nr. oder 0
VZ - Ventil-Zeiten	Leerzeit, Füllzeit (s)
VE - Fehlereingang, Polarität	
VB - Brixwert	
VT - Timeout, Wert	1,time oder 0
VX - externes Ventil	ID,Ventil
X! Tischplan löschen	
X# Nummer des Tischplan (1)	
XT Tische , alle Argumente integer, ausser Flags	
XJ Sprungtasten	
XR Rechtecke	
X% Zeitstempel	

Reinigungsprogramme:

@N - Name	Text
-----------	------

@V - Ventile	Menge der Ventilgruppen
@D - Einwirkunsdauer	Dauer in Sekunden
@S - Spülzyklen	Anzahl der Zyklen
@I - Intervalle	Auff. / Zwangsint. (d)
@P - Protokollierung	Protokoll ja / nein

Zahlungsarten:

@N - Zahlartname	Text
@K - Währungskurs	Kurs (reel)
@S - Währung Stellen	Stellen
@R - Runden	Ganze, Schwelle
@F - ProvisionFix	
@P - Provision Prozent	
@U - Euro Zahlart	
@A - Rabatt	
@B - Rechnungsmaske	
@D - Drucker	
@Z - Barzahlungsart	

6 Gruppenbefehle

Zur Abwicklung des Datentransfers werden Steuerbefehle verwendet. Sie gehen davon aus, daß die Daten in Gruppen zusammengefaßt sind, die aus dem ersten Zeichen eines Daten-IDs hervorgehen. Diese Zeichen und Gruppen sind -

F	Formulare
G	Geräte
K	Kellner / Schlüssel
L	Lager
P	Produkte
S	Sparten
V	Ventile
@	Erweiterte Gruppen

<DC2> "X!" - Gruppe Löschen

Löscht alle Daten aus einer Gruppe. Der Befehl soll im Zuge der Übertragung (in beiden Richtungen) und in Dateien vor den entsprechenden Daten stehen.

Beispiel: P! Alle Produktdaten Löschen

<DC1> / <DC2> "X#" - Anwahl eines Elements

Wählt ein Element aus einer Gruppe an. Es folgt eine Nummern, die das Element angibt. Steht der Befehl alleine in einem Satz, so wird er mit <Ack> beantwortet, sonst entscheiden die folgenden Befehle.

Der Befehl kann auch in <DC1> - Sätzen verwendet werden und dient dort ebenfalls dem Setzen der Elementnummer. Er wird nicht mit "X#" beantwortet. Damit können in einem einzigen Satz Element angewählt und Daten angefordert werden.

<DC1> "X\$" - Adressenabfrage

Fragt die Gruppenadressierung am Gerät ab. Der Befehl wird beantwortet mit

```
<DC2> X$ <TopNo> <MaxNo> <ActNo>
```

Dabei ist

<TopNo> der maximal zulässige Index (z.B. Produktnummer 999)

<MaxNo> die maximale Anzahl von Datenelementen im Speicher, die bei logischen Nummern von <TopNo> verschieden sein kann

<ActNo> die derzeit am Gerät programmierte Anzahl von Elementen

<DC1> "X?" - Gruppenabfrage

Leitet die Übertragung einer Gruppe vom Gerät zum PC ein.

<DC1> / <DC2> "X%" - Gruppen-Aktualität

<DC1> "X%" Fragt die Aktualität des anderen Geräts ab

<DC1> "X%" <Stempel> Sendet die Aktualität, d.h. Datum und
Uhrzeit der letzten Änderung in der Gruppe

7 Einzelbeschreibungen

#S - Software-Identifikation.

Syntax: #S <Name> "," <Version> "," <Datum>

Bedeutung: Das Gerät identifiziert dem Master auf Anfrage seine Software. <Name> und <Version> sind Texte, <Datum> bezeichnet das Freigabedatum der Software.

Der Befehl wird nie mit <DC2> vom PC zum Gerät geschickt.

Beispiele: #S"Gaby", "3.20", 1993/02/20

#H - Hardware-Identifikation

Syntax: #H <Name> "," <Version> "," <Datum>

Bedeutung: Das Gerät identifiziert dem Master auf Anfrage seine Hardware. <Name> und <Version> sind Texte, <Datum> bezeichnet, wenn dem Gerät bekannt, das Datum der Herstellung, sonst das Freigabedatum der Hardware.

Der Befehl wird nie mit <DC2> vom PC zum Gerät geschickt.

Beispiel: #H"Checkpoint", "2.40", 1993/02/20

#P - Protokoll-Identifikation

Syntax: #P <Name> "," <Version> "," <Datum>

Bedeutung: Das Gerät identifiziert dem PC auf Anfrage das für ASCII-I/O verwendete Protokoll. <Name> ist "ASCII-I/O"; <Version> die vom Gerät unterstützte Version. <Datum> bezeichnet das Freigabedatum des Protokolls.

Der Befehl wird nie mit <DC2> vom PC zum Gerät geschickt.

Beispiel: #S"ASCII-I/O", "1.10", 1993/02/20

** - Kommentar

Syntax: ** <Kommentar>

Bedeutung: Erlaubt das Einbinden von Kommentaren. <Kommentar> ist ein beliebiger Text, der nicht in Anführungszeichen gesetzt wird. Der Kommentar darf weder Anführungszeichen noch Strichpunkte enthalten.

#* - Erweiterte Gruppenwahl

Syntax: <DC2> #@ <Bezeichner>

Bedeutung: Vereinbart eine erweiterte Gruppe. In der Folge wird der Platzhalter "@" zur Bezeichnung der Gruppe verwendet.

Beispiel: Abfrage der Adressierung der Reinigungsprogramme:

```
<DC2> #* Reinigungsprogramme
<DC1> *$
```

A# - Anwahl Aufgabenbereich

Syntax: A# <a>

Bedeutung: Wählt einen Aufgabenbereich für folgende Übertragungen aus. Bis zu zehn Bereich, von 1 .. 10 nummeriert, können verwendet werden.

AL - Aufgabenliste

Syntax: AL <Liste>

Bedeutung: Definiert die in einem Aufgabenbereich enthaltenen Aufgaben. Die Liste besteht aus einer Aufzählung einzelner dezimal klassifizierter Aufgaben oder zusammenhängender Bereiche.

Bei Einzelangaben und dem Anfang eines Bereichs können folgende Stellen weggelassen werden; sie gelten als beliebig.

Wenn die Angabe des Bereichsendes weniger Stellen als der Bereichsanfang enthält, werden führende Stellen vom Anfang übernommen.

Beispiele:

AL1111,1211-1214	Aufgaben 1111 und 1211 .. 1214
AL3	Aufgaben 3000 .. 3999
AL24-2431	Aufgaben 2400 .. 2431
AL2111-4	Aufgaben 2111 .. 2114
AL2111-34	Aufgaben 2111 .. 2134

F# - Anwahl eines Formulars

Syntax: F# <f>

Bedeutung: Wählt ein Formular für folgende Übertragungen aus.
Es sind die folgenden Offsets von 1000 * n für verschiedene Arten vorgesehen -

Offset	Art
0000	Bildschirm-Masken
1000	Rechnungen
2000	Bons

Derzeit sind die folgenden Formularnummern verwendet -

<f>	Formular
0001	Bildschirm im Bonier / Dosiermodus
0101 .. 0125	Bonierseiten 1 bis 25
1001	Tischrechnung
2001	Bon

Beispiel: F#1001 wählt Rechnungsformular aus.

FF - Formular Fußzeile

Syntax: FF <Nummer> "=" <Text> { ", " <Nummer> "=" <Text> }

Bedeutung: Gibt Fußzeilen des angewählten Formulars an.

Beispiele: FF1="Es war sehr schön"
FF2="Es hat mich",3="sehr gefreut"

Anmerkung: Fußzeilen, die mit "#U" beginnen, werden auf der letzten Seite unterdrückt.

FK - Formular Kopfzeile

Syntax: FK <Nummer> "=" <Text> { ", " <Nummer> "=" <Text> }

Bedeutung: Gibt Kopfzeilen des angewählten Formulars an. Für Rechnungen werden derzeit vier Kopfzeilen verwendet; für Bonierseiten wird nur die erste Kopfzeile als Titel und Tastenbeschriftung verwendet.

Beispiele: F#1001
FK1="Restaurant zur Geilen Hure"
FK1="Guten Morgen",2="Liebe Gäste"

F#101;FK1='Vorspeisen'
F#102;FK1='Suppen'
F#121;FK1='Zigaretten'

Anmerkung: Kopfzeilen, die mit "#U" beginnen, werden auf Seite 1

unterdrückt.

FT - Formular Textzeile

Syntax: FT <Nummer> "=" <Text> { ", " <Nummer> "=" <Text> }

Bedeutung: Gibt Textzeilen des angewählten Formulars an. Für Rechnungen werden derzeit vier Kopfzeilen verwendet; für Bonierseiten wird nur die erste Kopfzeile als Titel und Tastenbeschriftung verwendet.

Beispiele: F#1001
FK1="Restaurant zur Geilen Hure"
FK1="Guten Morgen",2="Liebe Gäste"

F#101;FK1='Vorspeisen'
F#102;FK1='Suppen'
F#121;FK1='Zigaretten'

Anmerkung: In je einer Textzeile können verwendet werden:

#K zum Einfügen der Konsumationen/Buchungen

G# - Gerät anwählen

Syntax: G# <Nummer>

Bedeutung: Wählt ein Gerät aus. Dabei ist <Nummer> zusammengesetzt aus einem Offset für die Geräteart und einer Gerätenummer im Bereich 0 .. 99.

Die Offsets für die Geräteart sind -

100	Lokales Terminal (Display/Tastatur/Schloß)
200	Uhr
300	Drucker
400	Externes Terminal (ATerm/CheckPoint)
500	Dosierer
600	Kaffeemaschine
700	Waage
800	Display

Beispiele: G#302 wählt Drucker Nr. 2 an.
G#504 wählt Dosierer Nr. 4 an.

GF - Geräte Flags

Syntax: GF <Flags>

Bedeutung: Die hexadezimal kodierten Flags enthalten die folgenden Bits:

Bit 0 0 = offline, 1 = online

Alle anderen Bits müssen 0 sein.

GK - Geräte Kanal

Syntax: GK <Kanal> "=" <Produkt> { ", " <Kanal> "=" <Produkt> }

Bedeutung: Ordnet der Kanalnummer eines Geräts eine Produktnummer zu. Nach "G!" brauchen nur die tatsächlich verwendeten Kanäle übertragen werden.

Beispiel: GK1=200,2=202,4=203

GP - Geräte Protokoll

Syntax: GP <Portnummer>

Bedeutung: Ordnet einem Gerät einen Port zu.

Anmerkung: 1) Derzeit sind nur serielle Ports verwendet. Für parallele Ports ist ein Offset vorgesehen, z.B. Port "102" = paralleler Port 2.

2) GP ist auch vorgesehen für optionale Schnittstellenparameter wie Baudrate.

GT - Gerätetype

Syntax: GT <Type>

Bedeutung: Gibt - in Abhängigkeit der Geräteart - die Type des Geräts als Aufzählung an. Folgende Typen sind definiert -

Geräteart 300+ - Drucker:

0	unbenutzt
1	Star DP 8340
2	Citizen IDP 560
3	Epson TM 267
4	Citizen CBM 720
5	Epson 260 BA 230
6	MFP m.Autocutter
7	A4 Line Printer
8	Epson CTM 290
9	Epson 260 datate
10	Star SP 342F
11	Epson TM-T80

Geräteart 500+ - Dosierer:

0	unbenutzt
1	Zapfkopf unspezifisch
2	Gaby unspezifisch

Geräteart 700+ - Waage:

0	unbenutzt
1	BIZERBA PL

Geräteart 800+ - Display

0	unbenutzt
1	Epson DM-D202

PA - Produktart und Flags

Syntax: PA <Produktart> [, <Flags>]

Bedeutung: <Art> gibt die Produktart an. Derzeit sind definiert:

Bonierprodukt	1
Dosierprodukt	2
Zähler	3
Literzähler	4
ser. Kaffee	5
Jetons	6
Anfangsprodukt	7
Folgeprodukt	8
Single-Shot	9

<Flags> ist eine hexadezimal angegebene Bitmenge der folgenden Flags -

Bit	Bedeutung
0	0 = gesperrt, 1 = frei
1	0 = normale Verrechnung, 1 = nicht verrechnen
2..7	reserviert
8	Am Bon hervorheben (Rotdruck)
9	Auf Abrechnungen hervorheben (Rotdruck)
10	1 = Betragsprodukt
11	1 = Zusatzprodukt

Wenn <Flags> nicht enthalten ist, nimmt der Empfänger \$1 an, d.h. frei, normale Verrechnung, keine Hervorhebung, kein Betrags- und kein Zusatzprodukt.

Beispiele: PA1 Bonierprodukt, frei
 PA1,401 Bonierprodukt, Betragsprodukt, frei

PB - Produkt-Boncode

Syntax: PB <Boncode>

Bedeutung: Gibt die Zieldrucker zum Bonieren des Produkts an. In <Boncode> steht die Einerstelle für den Erstbon, die Zehnerstelle für den Doppelbon. Eine Stelle bedeutet jeweils -

0	-	kein Bondruck
1 .. 4	-	Bondruck auf Drucker 1 .. 4
9	-	Bondruck auf Vorgabedruker

Beispiele: PB03 Bon auf Drucker 3
PB29 Bon auf Vorgabedruker, Doppelbon auf Drucker 2

PD - Produktdosis

Syntax: PD <Menge> <Einheit> { , <Menge> <Einheit> }

Bedeutung: Gibt die Produktdosis für eine oder mehrere Folgen in Sekunden oder Impulsen an. Achtung: Derzeitigen Dosierer erwarten fest zwei Nachkommastellen für Angaben in Sekunden.

Beispiele: PD12.50s
PD803I,8.30s,0s

PI - Produkt-Info

Syntax: PI " <Info> "

Bedeutung: Gibt einen Info-Text für das Produkt an.

Beispiel: PI"Mit Olive!"

PL - Produkt Lager

Syntax: PL <LagerNr> = <Faktor> { , <LagerNr> = <Faktor> }

Bedeutung: Gibt für ein Produkt ein oder mehrere Lagerkonten mit den jeweiligen Faktoren an.

Beispiel: PL1=0.8,44=0.25

PM - Produkt Maximaldauer

Syntax: PM <Dauer> { , <Dauer> }

Bedeutung: Gibt für eine oder mehrere Folgen eines Produkts die Maximaldauer (Timeout) für Zapfen/Dosieren in Sekunden an.

Beispiele: PM30
PM30,30,10

PN - Produktname

Syntax: PN <Text>

Bedeutung: Gibt den Produktnamen an.

Beispiel: PN"Gulasch"

PC - Produktcode

Syntax: PC <Barcode>

Bedeutung: Gibt den Produktcode an.

Beispiel: PN"1123123123"

PP - Produktpreis(e)

Syntax: PP <Preis> [, <Preis 2>]

Bedeutung: Gibt einen oder zwei Preise (für zwei Ebenen) an.

Beispiele: PP25.00
PP48.00,55.00

Achtung: Manche Geräte erwarten fest zwei Nachkommastellen.

Achtung: Wird nur Preis 1 gesendet, muß der Empfänger, wenn er zwei Preise unterstützt, den zweiten gleich dem ersten setzen. Ein Sender kann zwecks Beschleunigung Preis 2 weglassen, wenn die Preise gleich sind.

PR - Produkt Rezeptur

Syntax: PR <KompNr> = <Faktor>:<Flags> { , <KompNr> = <Faktor>:<Flags> }

Beispiel: PR1=0.8,44=0.25,2=1.0:1

Bedeutung: Gibt für ein Produkt eine Rezeptur mit einer oder mehreren Komponenten an. Für jede Komponente werden Produktnummer und Faktor angegeben.
Flags (Hex) : Bit 0 = Vorbonieren, Bit 1 = Kredit

PS - Produktparte

Syntax: PS <Spartennummer>

Bedeutung: Gibt die Produktparte an, meist im Bereich 1 .. 20.

Beispiel: PS15

PT - Produkttaste(n)

Syntax: PT { [<Seite> "="] <Taste> }

Bedeutung: Gibt die Taste(n) für das angewählte Produkt an. Für Geräte, die mehrere Terminals (Schlösser) ansteuern oder mehrere Seiten unterstützen, werden Terminal und / oder Seite in <Seite> kodiert wie folgt:

$\langle \text{Seite} \rangle = 1000 * \text{Terminalnummer} + \text{Seitennummer}$

Terminalnummer 1 (Offset 1000) kann weggelassen werden. Sind Terminal- und Seitennummer 1, kann <Seite> überhaupt weggelassen werden.

Beispiele: PT95 Taste 95 (Terminal 1, Seite 1)
PT1=3 Taste 3 auf Seite 1 (Terminal 1)
PT1=3,21=5 Taste 3 auf Seite 1, Taste 5 auf Seite 21
PT2001=3 Taste 3 an Terminal 2
PT4002=3 Taste 3 an Terminal 4, Seite 2

PG - Produkt Rezeptur (Dosierprodukt)

PK - Produkt Rezeptur (Dosierprodukt)

PV - Produkt-Ventile.

Syntax: PV <Ventil> { + <Ventil } { , <Ventil> { + <Ventil } }

Bedeutung: Gibt für eine oder mehrere Folgen jeweils ein oder mehrere Ventile an.

Beispiele: PV1
PV1+2,12,14+15

KA - Kellneraufgaben (Schlüsselart)

Syntax:

vor 1.20: KA <Kellnerart> , <Flags>

ab 1.20: KA <Verwendung> , <Flags> , { <Aufgabenbereiche> }

Bedeutung: Gibt die Art/Verwendung und Flags des angewählten Kellnerschlüssels an. Die folgenden Arten sind definiert -

vor ASCII-I/O 1.20:

- 0 nicht verwendet
- 1 normal
- 2 Schankkellner 1. Art (gibt Zielschlüssel ein)
- 3 Schankkellner 2. Art (entnimmt für Zielschl.)
- 4 Programmierungsschlüssel
- 5 Handbetrieb

ab Version 1.20:

- 0 nicht verwendet
- 1 verwendet

Die Flags sind eine hexadezimal angegebene Bitmenge. Die folgenden Bits sind definiert -

- Bit 0 0 = gesperrt, 1 = frei
- Bit 1 0 = normale Verrechnung, 1 = Hausschlüssel
- Bit 2 0 = keine Zeiterfassung, 1 = Zeiterfassung

Ein Kellner kann dann arbeiten, wenn

Art	gesperrt	Zeiterfassung	Angemeldet
normal	frei	nein	X
normal	frei	ja	ja

Die optionalen Aufgabenbereiche sind durch "+" voneinander getrennt.

- Beispiele: KA0,1 Schlüssel nicht verwendet, frei, keine Aufgaben
- KA1,3,1+4 Schlüssel verwendet, frei, Haus, Aufgabenbereiche 1 und 4

KE - Kellner-Preisebene.

Syntax: KE <Zeit> = <Ebene> { , <Zeit> = <Ebene> }

Bedeutung: Gibt die Preisebene(n) für einen Kellner an. Die Uhrzeiten geben jeweils der Beginn eines neuen Zeitbereichs an. Die erste Angabe muß mit 00:00 beginnen. Die letzte Angabe gilt bis 23:59.

Beispiel: KE00:00=1,08:30=2,17:00=1

KN - Kellnername.

Syntax: KN <Text>

Bedeutung: Gibt den Kellnernamen an.

Beispiel: KN"Pepi"

KT - Kellner Tischbereiche.

Syntax: KT <Anfang1>-<Ende1> { , <AnfangN>-<EndeN> }

Bedeutung: Gibt einen oder mehrere Tischbereiche für den Kellner an.

Beispiele: KT1-99
KT1-3,200-299,80-99

LE - Lagereinheit.

Syntax: LE <Text>

Bedeutung: Gibt die für das Lagerkonto verwendete Einheit an.

Beispiel: LE"Stk" Einheit ist "Stk"

LN - Lagerkonto Name.

Syntax: LN <Text>

Bedeutung: Gibt den Namen des Lagerkontos an.

Beispiel: LN"Messwein"

LS - Lagerstand.

Syntax: LS <Stand>

Bedeutung: Gibt den Stand des Lagerkontos an.

Achtung: Manche Geräte erwarten fest zwei Nachkommastellen.

LV - Lagerverbrauch.

Syntax: LV <Verbrauch>

Bedeutung: Gibt den akkumulierten Verbrauch, d.h. den Stand einer Verbrauchs- oder Füll-Liste an.

Achtung: Manche Geräte erwarten fest zwei Nachkommastellen.

SA - Sparten-Alkoholsteuersatz.

Syntax: SA <Satz>

Bedeutung: Gibt den Alkoholsteuersatz der angewählten Sparte in Prozent an.

Achtung: Manche Geräte erwarten fix eine Nachkommastelle.

Beispiel: SA10.0 Alkoholsteuersatz 10 %

SB - Sparten-Bedienungsgeldsatz.

Syntax: SB <Satz>

Bedeutung: Gibt den Bedienungsgeldsatz der angewählten Sparte in Prozent an.

Achtung: Manche Geräte erwarten fix eine Nachkommastelle.

Beispiel: SB10.5 Bedienungsgeldsatz 10.5 %

SG - Sparten-Getränksteuersatz.

Syntax: SG <Satz>

Bedeutung: Gibt den Getränkesteuersatz der angewählten Sparte in Prozent an.

Achtung: Manche Geräte erwarten fix eine Nachkommastelle.

Beispiel: SG0.0 Getränkesteuersatz 0 %

SM - Sparten-Mehrwertsteuersatz.

Syntax: SM <Satz>

Bedeutung: Gibt den Mehrwertsteuersatz der angewählten Sparte in Prozent an.

Achtung: Manche Geräte erwarten fix eine Nachkommastelle.

Beispiel: SM20.0 Mehrwertsteuersatz 20 %

SN - Spartenname.

Syntax: SN <Text>

Bedeutung: Gibt den Text für die angewählte Sparte an.

Beispiel: SN"Bier"

Ventile

VA - Ventilart

Syntax: VA <Art>

Bedeutung: Art gibt die Art oder Reinigungsfunktion des Ventils an. Definiert sind derzeit -

0	Produktventil
1	CO2 (Leerdrücken)
2	Reinigungsmittel
3	Spülwasser

VG - Ventilgruppe

Syntax: VG <Gruppe>

Bedeutung: <Gruppe> gibt die Reinigungsgruppe im Bereich 0 .. 9 an; 0 sollte für bei der Reinigung nicht geöffnete Ventile verwendet werden.

VN - Ventilname

Syntax: VN <Name>

Bedeutung: <Name> ist die Bezeichnung des Ventils

VV - Ventil-Volumsgeber

Syntax: VV <Geber>

Bedeutung: <Geber> ist die Nummer des dem Ventil zugeordnete Volumsgebers, oder 0, wenn kein Geber zugeordnet ist.

Geräte mit fester 1:1-Zuordnung der Geber ignorieren empfangene VV-Befehle, senden aber auf Anforderung die 1:1-zugeordneten Gebernummern.

VZ - Ventilzeiten

Syntax: VZ <Leerzeit> "," <Füllzeit> "," <Spülzeit>

Bedeutung: <Leerzeit> gibt die Zeit zum Leerdrücken mit CO2, <Füllzeit> die Zeit zum Füllen mit Reinigungsmittel und <Spülzeit> die Zeit zum Spülen mit intermittierenden CO2 und Wasser an. Alle Zeiten sind in ganzen Sekunden angegeben.

Beispiel: VZ2,3,90

Reinigungsprogramme

Diese erweiterte Gruppe wird mit dem Bezeichner "Reinigungsprogramme" eingeleitet. In der Folge wird "@" als Gruppenzeichen verwendet.

@N - Reinigungsprogramm Name

Syntax: @N <Text>

Beispiel: @N "3-Monatlich"

@V - Reinigungsprogramm Ventilgruppen

Syntax: @V <Menge>

Bedeutung: <Menge> ist die Menge der Ventilgruppen. Sie wird in der neuen Syntax mit geschweiften Klammern und Kommata / Minuszeichen angegeben.

Beispiel: @V {1,5-7}

@D - Reinigungsprogramm Einwirkungsdauer

Syntax: @D <Dauer>

Bedeutung: Dauer gibt die Einwirkungsdauer in Sekunden an.

Beispiel: @D 1800

@I - Reinigungsprogramm Intervalle

Syntax: @I <Aufforderung> , <Zwang>

Bedeutung: <Aufforderung> ist das Aufforderungsintervall, <Zwang> das Zwangsintervall in Tagen. Die Angabe von 0 sperrt Aufforderung bzw. Zwang.

Beispiel: @I 90, 95

@P - Reinigungsprogramm Protokollierung

Syntax: @P <Flag>

Bedeutung: <Flag> = "0" bedeutet keine Protokollierung; <Flag> = "1" bedeutet, daß der Benutzer bei der Ausführung Protokollierung anfordern kann.

Beispiel: @P 1

Zahlungsarten

Diese erweiterte Gruppe wird mit dem Bezeichner "Zahlungsarten" eingeleitet. In der Folge wird "@" als Gruppenzeichen verwendet.

Eine Besonderheit der Gruppe liegt darin, daß das Element 1 die Vorgabewährung beschreibt. Entsprechend gilt es als immer verwendet und hat einen Kurs von 1.

@N - Währungsname

Syntax: @N <Text>

Bedeutung: Gibt die Bezeichnung der Währung an. Es wird empfohlen, maximal drei Stellen zu verwenden.

Beispiel: @N "DM"

@K - Währungskurs (in Einheiten der Währung 1)

Syntax: @K <Kurs>

Bedeutung: Gibt an, wieviel eine Währungseinheit in Einheiten der Währung 1 wert ist.

Beispiel: @K 6.954

@S - Währung Stellen nach dem Komma

Syntax: @S <Stellen>

Bedeutung: Gibt an, wieviele Nachkommastellen in der Währung verwendet werden.

Beispiel: @S 2

@R - Währung Rundungsparameter

Syntax: @R <Ganze> , <Schwelle>

Bedeutung: beim Umrechnen wird ab „Schwelle“ auf „Ganze“ gerundet.

Beispiel: @R 0.50,0,25

@F - Währung Provision Fix

@M - Währung Provision Min

@P - Währung Provision Prozent

@U - Währung Eurozahlart

@A - Währung Rabatt

@B - Währung Rechnungsmaske

@D - Währung Drucker

@Z - Währung Barzahlungsart

Tischplan

X! Plan löschen
 X# Nummer des Tischplan (1)
 XT Tische , alle Argumente integer, ausser Flags
 XJ Sprungtasten
 XR Rechtecke
 X% Zeitstempel

TNr Tischnummer
 Seite Seite des Tischplans auf dem dieses Objekt gezeichnet wird
 Seite 0 wird auf allen Seiten des Plans zusätzlich angezeigt
 x,y Koordinaten in Pixel, absolut, Ursprung ist links oben
 w,h Grösse in Pixel
 hexflags 0x00 standard
 0x01 rund
 0x03 45° , 1:1
 0x07 45° , 1 : 1,38
 0x0B 45° , 1 : 1,77

 sx,sy optional Sesselkoordinaten (auch mehrere),
 relativ zu TischX-15, TischY-15
 bezeichnung optionale Tischbezeichnung
 sprungziel auf diese Seite wechseln
 fg Schriftfarbe
 bg Tastenfarbe

```

X!;
X#1;
XT TNr , seite , x , y , w , h , hexflags , [ sx , sy , ][ "bezeichnung" ];
XT ...
XT ...
XJ seite , x , y , w , h , fg , bg , sprungziel , hexflags , "bezeichnung" ;
XJ ...
XJ ...
XR seite , x , y , w , h , fg , bg , hexflags , "bezeichnung" ;
XR ...
XR ...
X% Datum ;
  
```