



SESAM 4

RSC - Kassen
Protokoll
Programmierung

RSC - KASSAPROTOKOLL	3
LSV2 Übertragungsprotokoll	3
LSV2 Senden: int tx_lsv2(char *tx).....	4
LSV2 Empfangen: int rx_lsv2(char *tx).....	5
DATENSÄTZE	6
Bonierungen senden:	7
Bonanfrage:	7
Bonsendung:	7
NODE:	6
Flags:	8
Beispiele:	9
Bonsendung aus der Schankanlage:	9
Credit aufzählen:	10
Guestcheck	11
Tischanfrage: Schank <=> Kassa/PC	12
Tisch Antwort: Schank <=> Kassa/PC	13
Bedeutung Tischantwort:	13
Tisch löschen, blockieren Schank <=> Kassa/PC.....	14
Tisch abschließen Schank <=> Kassa/PC.....	15
Guestcredit	17
Guest Anfrage: Kassa/PC <=> Schank	17
Antwort:	17
Kellner-Programm: Schank <=> Kassa/PC.....	24
Komplette Kellner-Programmierung (KELPAR)	24
Erweiterte Kellner-Programmierung (KELPAREX)	25
Kellner-Karten ID (KELPARID).....	26
Flags der Kellnerprogrammierung:	25
Kellner-Karten ID löschen (KELPARCL).....	26
PLU-Programm: Schank <=> Kassa/PC	27
Komplette Plu-Programmierung	27
Flags der PLU-Programmierung:	27
Reduzierte Plu-Programmierung	29
Flags:	29
SESAM 4 Version: Schank <=> Kassa/PC.....	30
Versionsanfrage:	30
Versionsantwort:	30

RSC - Kassaprotokoll**LSV2 Übertragungsprotokoll**

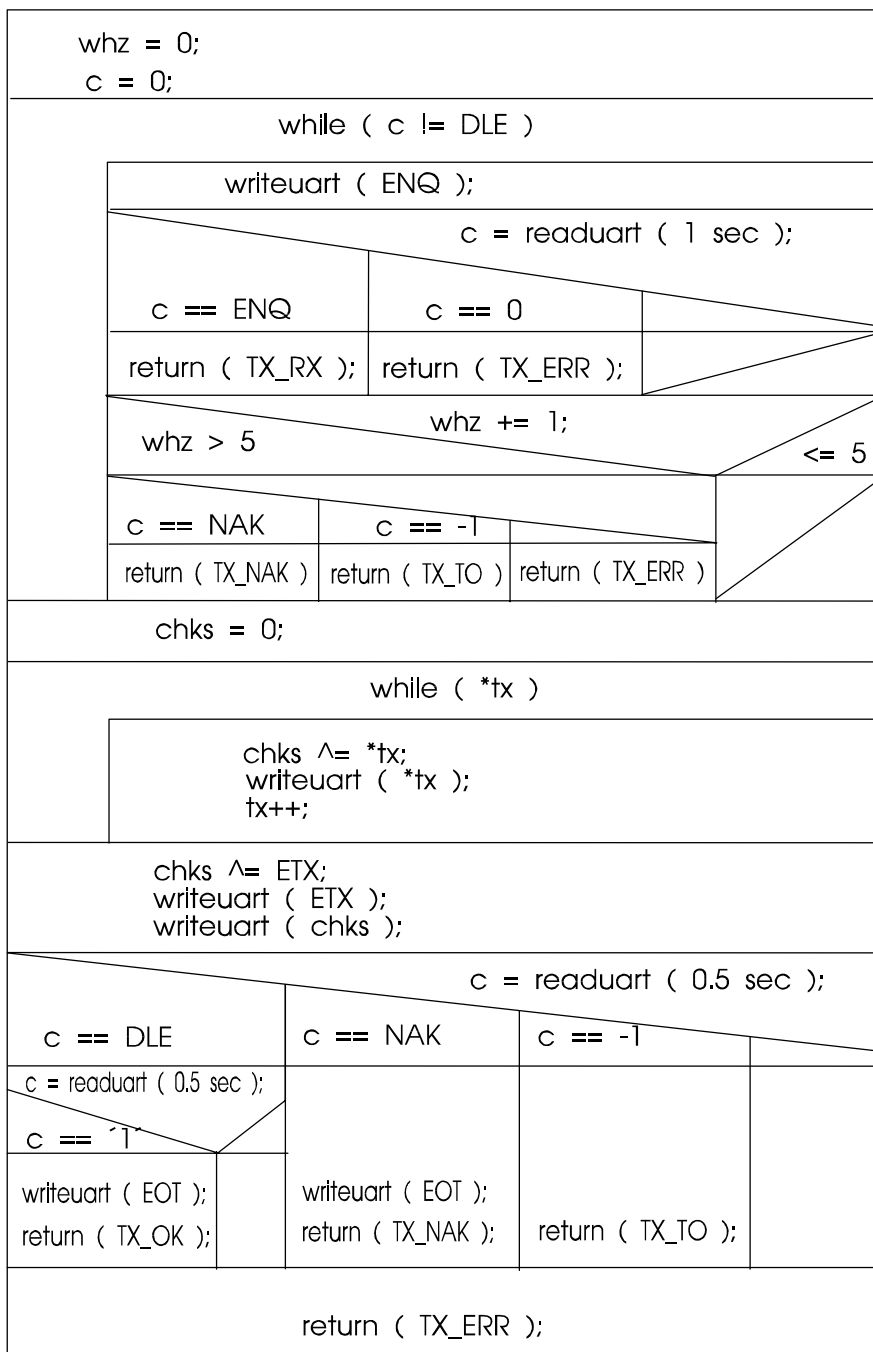
Sender		Empfänger	
<hr/>			
ENQ	(0x05)	=>	Timeout 10 sec.
		<=	DLE '0' (0x010 0x030)
STX	(0x02)	=>	Timeout 5 sec. / Zeichen
Daten....		=>	
ETX	(0x03)	=>	
LRC		=>	LRC = EXOR (Daten+ETX) ohne STX
		<=	DLE '1' (0x010 0x031)
EOT	(0x04)	=>	

Timeouts gelten für 9600 Bd

In der Schankanlage wird das LSV2 Protokoll folgendermaßen durchgeführt:

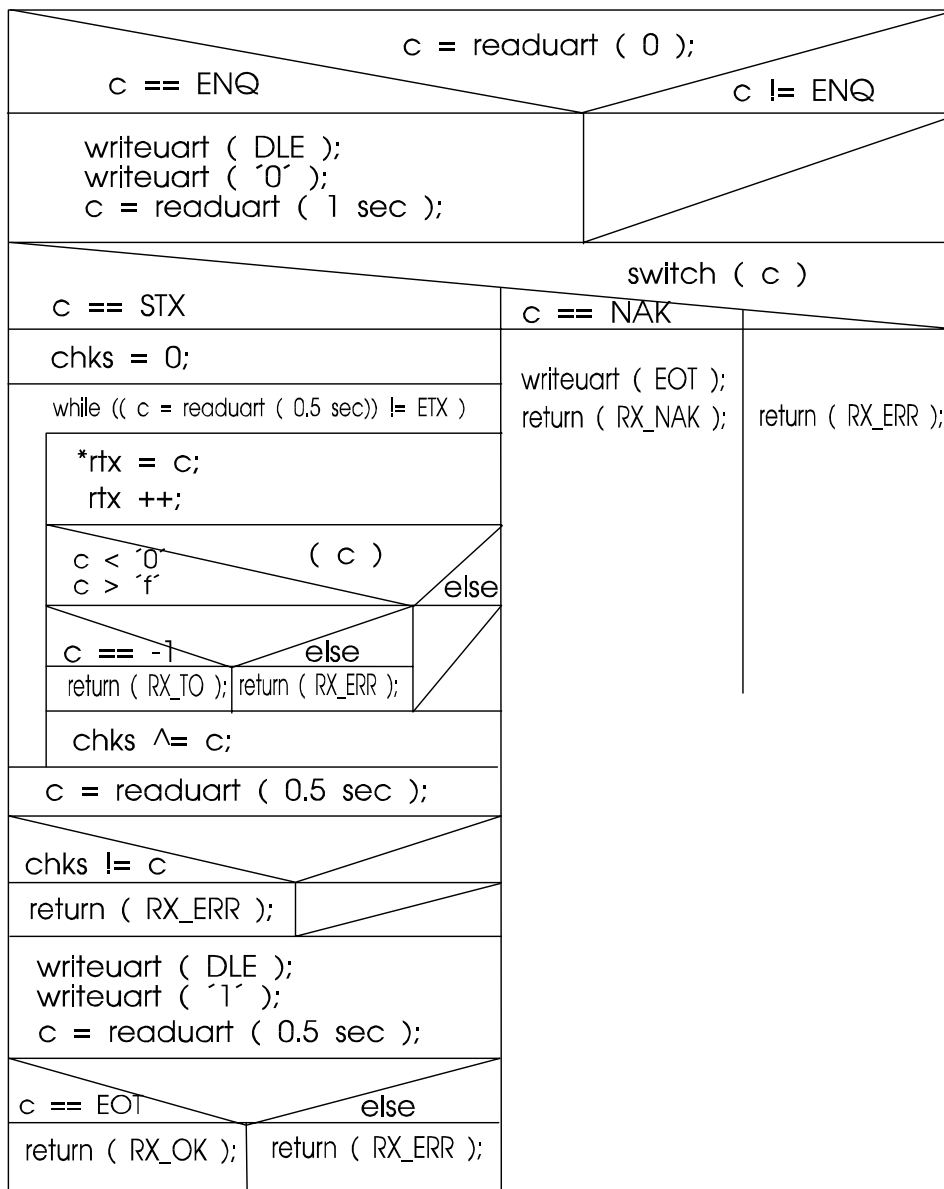
LSV2 Senden: int tx_lsv2(char *tx)

Returnwerte: #define TX_OK 0 /* Sendung OK */
 #define TX_TO 1 /* Timeout Fehler */
 #define TX_RX 2 /* Statt Sendung Datensatz empfangen */
 #define TX_NAK 3 /* Datensatz wurde nicht richtig gesendet */
 #define TX_ERR 4 /* Übertragung fehlerhaft */



LSV2 Empfangen: int rx_lsv2(char *tx)

Returnwerte: #define RX_OK 0 /* Empfang OK */
 #define RX_TO 1 /* Timeout Fehler */
 #define RX_NAK 3 /* Datensatz wurde nicht richtig empfangen */
 #define RX_ERR 4 /* Übertragung fehlerhaft */



Datensätze

Besonderheiten:

NODE:

Der Node gibt die Platinennummer an, an die die Daten gesendet werden sollen.

Der NODE hat einen Wertebereich von 0-15

NODE = 0 bedeutet dieser Datensatz geht an alle Platinen. Das ist der Normalfall.

NODE = 1-15 Platinennummer

NODE = (0-3) * 16 = (Bit 4 und Bit 5) Credit Gruppe. Das muß in der Schank aktiviert sein.

NODE beim Credit aufzählen:

Werden von der Kasse nur Creditdatensätze gesendet, so kann der Datensatz an eine definierte Platine oder Platinengruppe gesendet werden.

Diese Optionen müssen in der Schank aktiviert sein

PLU-Nummer:

Die PLU-Nummer bei Sendungen zum PC enthält in den Bits 10-15 Absenderinformationen:

Das wurde so gelöst, damit das Nachrichtenformat unverändert bleibt.

Bit 10-13 ist Absenderknoten

Bit 14,15 die Absender Creditgruppe

Bonierungen senden:

Bonanfrage:

Kassa/PC => Schank

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	1 8	IDENT = Anfrage

Antwort:

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	1 9	IDENT = Keine Daten

Bonsendung:

Kassa/PC <=> Schank

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 7	IDENT = AUFZPAR
5-12	X X X X X X X X	Flags
13-15	X X X	Kellner (1-127)
16-18	X X X	Sparte (1-20)
19-23	X X X X X	Tisch (0-65535)
24-28	X X X X X	Stück (1-32767)
29-33	X X X X X	PLU-Nummer (1-1000)
34-41	X X X X X X,X X	Summe: Preis*Stück (in Groschen) Max: 16777215 (0x00ffffff)
42 - 44	X X X	Fortlaufende Nummer 1 - 255 Bei Übertragungsfehlern wird ein Datensatz mit der gleichen Nummer wiederholt Die Nummer 0 wird nach dem Einschalten gesendet. 2*Null wird auch als gültige Nachricht akzeptiert.

Bei Verwendung von Gastkarten wird statt der Tischnummer, die Kartennummer verwendet.
In diesem Modus können jedoch nur Karten mit den Nummern von 1 bis 65535 verwendet werden.

Flags:

Flags werden als ASCII-Hex Zahl übertragen und entsprechen einem 32 Bit-Feld.

Bit 0 - 2 Record Type

LEER_RC	0x00000000
BEIL_RC	0x00000001 Beilage
SPA_RC	0x00000002 Summe auf Sparte boniert
PLU_RC	0x00000003 Plu boniert
EXT_RC	0x00000004 Externe PLU (Kaffeemaschine, Schnaps)
GET_RC	0x00000005 Getränke-Portion
KART_RC	0x00000006 Record mit Kartennummer wird als erster Datensatz an Druck und Kassa geschickt

Bit 3 - 4 Preisebene

PR1_RC	0x00000000
PR2_RC	0x00000008
PR3_RC	0x00000010
PR4_RC	0x00000018

Bit	5	HEAD_RC	Header für Bondruck
Bit	6	END_RC	Ende für Bondruck
Bit	7	NACH_RC	Nachfolgebon (Wenn ohne Kellnerschlüssel gezapft wurde)
Bit	8	STOR_RC	Stornobon
Bit	9	STOE_RC	Stornobon für Extrastorno
Bit	10	PSEU_RC	Bonierung auf Pseudorechnung
Bit	11	HAUS_RC	PLU-wurde mit Hausschlüssel boniert
Bit	12	DRUK_RC	Record an Bondrucker
Bit	13	KELL_RC	Karte über Kellner
Bit	14	BEIN_RC	nächster Record ist Beilage
Bit	15	SPAR_RC	Sparten - Detailspeicher aufzählen
Bit	16	CRED_RC	Creditspeicher aufzählen
Bit	17	GAST_RC	Record auf Tischrechnung addieren
Bit	18	KASS_RC	Record an Kassa/PC senden (nur intern verfügbar)
Bit	19	Frei	
Bit	20	LAG_RC	Record von Lager abziehen
Bit	21	ERRO_RC	Netzwerk Fehler Nachfolgebon
Bit	22 – 31	Bondrucker	Bondrucker 1-10 (Bitweise)

Beispiele:**Bonsendung:**

Schank --> Kassa

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 7	IDENT = AUFZPAR
5-12	0 0 D 4 9 0 6 D	Flags: GET_RC Eine Portion wurde gezapft PR2_RC Preisebene 2 HEAD_RC,END_RC es soll ein EinzelBon werden DRUK_RC Ausdruck erlaubt SPAR_RC Auf internen Sparten und Detailspeicher aufzählen. KASS_RC zur Kasse senden LAG_RC Lagerstand bearbeiten 0x00C00000 Ausdruck auf 1. u. 2. Bondrucker
13-15	0 0 1	Kellner 1
16-18	0 0 3	Sparte 3
19-23	0 0 0 2 5	Tisch 25
24-28	0 0 0 0 1	Stück 1
29-33	0 0 0 1 5	PLU-Nummer 15
34-41	0 0 0 0 2 6 0 0	Summe Preis*Stück (in Groschen)

Credit aufzählen:

KASSA --> Schank Minimalversion

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 7	IDENT = AUFZPAR
	0 0 0 1 0 0 0 3	Flags: PLU_RC CRED_RC Auf Creditspeicher aufzählen
13-15	0 0 1	Kellner 1
16-18	0 0 0	Sparte 0 (wird bei Credit nicht verw.)
19-23	0 0 0 0 0	Tisch 0 (wird bei Credit nicht verw.)
24-28	0 0 0 1 0	Stück 10
29-33	0 0 1 3 5	PLU-Nummer 135
34-41	0 0 0 0 0 0 0 0	Summe (wird bei Credit nicht verw.)

Credit aufzählen:

KASSA --> Schank

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 7	IDENT = AUFZPAR
5-12	0 0 5 1 9 0 6 3	Flags: PLU_RC HEAD_RC,END_RC es soll ein EinzelBon werden DRUK_RC Ausdruck erlaubt SPAR_RC Auf internen Sparten und Detailspeicher aufzählen (Kann entfallen ist jedoch sinnvoll um die Abrechnung in der Schank und Kasse gleich zu halten) CRED_RC Auf Creditspeicher aufzählen LAG_RC Lagerstand bearbeiten (Nur sinnvoll wenn der Lagerstand der Schankanlage verwendet wird) 0x00400000 Ausdruck auf ersten Bondrucker
13-15	0 0 1	Kellner 1
16-18	0 0 3	Sparte 3
19-23	0 0 0 2 5	Tisch 25 (wegen Bondruck)
24-28	0 0 0 1 0	Stück 10
29-33	0 0 1 3 5	PLU-Nummer 135
34-41	0 0 0 7 0 0 0 0	Summe Preis*Stück (in Groschen)

Guestcheck

In der Schankanlage werden Tische im gemeinsamen Speicher verwaltet.

Pro Platine können maximal 1700 Datensätze gespeichert werden.

Bei mehreren Platinen werden die Tischdaten etwas aufgeteilt:

$\text{Platinennummer} = (\text{Tischnummer}/10) \bmod \text{Platinenanzahl}$

Für jeden verwendeten Tisch wird ein Header Datensatz angelegt:

BOOL Tisch blockiert

BYTE Kellner Nummer der den Tisch belegt

WORD Tisch Nummer

Long Tisch Summe

Bei der Einstellung *MIT TISCHDATEN* (Wird im Guestcheckprogramm definiert) werden auch noch die einzelnen Bonierungen gespeichert für den Rechnungsausdruck.

Tischanfrage:

Schank <=> Kassa/PC

Im normalen Guestcheck Betrieb wird bei jeder Tischeingabe eine Anfrage an die Tischverwaltung ausgeführt.

Das kann ein Task in der Schankanlage, oder eine externe Kasse sein.

Es kann auch die Kasse eine Tischanfrage an die Schank senden.

Ist der Tischspeicher in der Schankanlage und Open TTEST oder TMAKE sendet die Schankanlage eine Tischantwort

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	1 3	IDENT = TIFRAGE
5-6	0 X	Open
7-9	X X X	Kellner Nummer
10-14	X X X X X	Tisch Nummer

open: TTEST 0 Teste Tisch
TMAKE 1 Wenn Tisch nicht vorhanden, dann anlegen.

TMAKE2 2 Wenn Tisch nicht vorhanden, dann anlegen.
Tisch wird blockiert.
Weiters werden für diesen Tisch keine detaillierten Records angelegt.
Bei Tischanfrage aus PC erfolgt **KEINE Tischantwort**.

Tisch Antwort:

Schank <=> Kassa/PC

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	2 1	IDENT = TIANZWORT
5-6	X X	Tischantwort
7-9	X X X	Kellner welcher Tisch belegt hat
10-14	X X X X X	Tischnummer
15-22	X X X X X X,X X	Tisch-Summe (in Groschen)

Bedeutung Tischantwort:

TIA_OK	0	"Tisch Su:%8u.%02u"
TIA_TIDIS	1	(nur Intern)
TIA_EXDIS	2	"Tisch extern gesp." (nur Intern)
TIA_NOTISCH	3	"Tischeingabe ?"
TIA_LEER	4	"Tisch leer !"
TIA_BESETZT	5	"Tisch besetzt Kel%3u"
TIA_NOMEM	6	"Tisch Speicher voll"
TIA_NEU	7	"Tisch neu angelegt"
TIA_ZUKLEIN	8	"Tisch zu klein !"
TIA_ZUGROSS	9	"Tisch zu gross !"
TIA_ERROR	10	"Fehler TEST-GAST"
TIA_TIMEOUT	11	"PC/Kassa TIMEOUT"
TIA_TINUM	12	"Tischnummer: %5u"
TIA_TIGESP	13	"Tisch gesperrt"
TIA_NERLAUB	14	"Tisch nicht erlaubt"
TIA_BLOCK	15	"Tisch blockiert"

Tisch löschen, blockieren

Schank <=> Kassa/PC

Dieser Datensatz wird **von der Schank zur Kasse** nur im Modus „*RSC-Touch*“ gesendet.

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 9	IDENT = STMPPAR
5-6	1 2	SID = STISCLR
7-8	0 X	CLR Type
9-12	0 0 0 0	reserve
13-17	X X X X X	Tisch Nummer
18-20	0 0 0	reserve
21-23	X X X	Kellner Nummer kann bei TYP 0,1 Null sein

CLR Type:

0 = Tisch löschen. Je nach Einstellung im Guestcheckprogramm (*TISCH KOMPLETT LÖSCHEN*) werden nur die Tischdaten oder auch der Tischheader gelöscht.

1 = Tisch komplett löschen. Diese Funktion wird nur von Kassen aufgerufen. Dadurch kann eine Tischsynchronisation zwischen Kassa und Schank erfolgen:

- Kellner dürfen an der Schank keine Tische eröffnen (Kellnerprogramm)
- Die Kasse eröffnet die Tische für die Kellner = Kellner können nur an der Kasse Tische öffnen und anlegen.
- Dadurch wird der Tisch mittels Tischanfrage an der Schank eröffnet und für diesen Kellner belegt.
- Nach dem Abschluß des Tisches an der Kasse sendet diese STISCLR mit CLR Type=1 um den Tisch komplett zu löschen oder CLR Type=0 um nur die Daten aber nicht die Zuordnung zu löschen. (Das muß in der Schank so eingestellt sein)

2 = Tisch blockieren: Damit kann verhindert werden, daß andere Kellner die auf alle Tische bonieren dürfen, (BIT KB_ALTI im Kellnerprogramm) auf diesen Tisch zugreifen. Wird normalerweise gesetzt, wenn ein Kellner gerade mit einem Tisch arbeitet

3 = Tisch Blockierung aufheben.

Tisch abschließen

Schank <=> Kassa/PC

Dieser Datensatz wird **von der Schank zur Kasse** nur im Modus „*RSC-Touch*“ gesendet.

Beim Rechnungsausdruck erfolgt ein Tischabschluß.

- Die Rechnung wird ausgedruckt
- Die Rechnungssumme wird in das Rechnungsjournal mit Datum und Uhrzeit eingetragen
- In allen Platinen wird die aktuelle Rechnungsnummer um Eins erhöht.
- Der Tisch wird gelöscht (STISCLR Typ = 0)

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 9	IDENT = STMPPAR
5-6	1 4	SID = STISJAD
7-9	X X X	Kellner Nummer
10-12	0 0 0	Mode nicht verwendet
13-17	X X X X X	Tisch Nummer
18-25	X X X X X,X X	Tisch Summe
26-33	X X X X X X X	Rechnung Nummer

Revier übergeben

Schank <=> Kassa/PC (ab V1.66bv)

Dieser Datensatz wird **von der Schank zur Kasse** nur im Modus „*RSC-Touch*“ gesendet.

Bei der Revierübergabe wird einem oder allen Tischen eines Kellners (kelvon) die neue Kellnernummer (kelnach) zugeordnet.

Bei den Sparten wird dem Kellner kelvon der Betrag unter SP_ABKE und dem Kellner kelnach unter SP_ZUBU verbucht.

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 9	IDENT = STMPPAR
5-6	1 4	SID = STISJAD
7-9	X X X	Kellner Nummer
10-12	0 0 0	Mode nicht verwendet
13-17	X X X X X	Tisch Nummer
18-25	X X X X X,X X	Tisch Summe

Guestcredit**Guest Anfrage:**

Kassa/PC <=> Schank

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	3 4	IDENT = Anfrage
5-7	X X X	Kellner Nummer
8-9	X X	Karten Ident[0]
10-11	X X	Karten Ident[1]
12-13	X X	Karten Ident[2]
14-15	X X	Karten Ident[3]
16-17	X X	Karten Ident[4]
18-19	X X	Karten Ident[5]

Guest Antwort:

Es sind 2 Antworten möglich

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	3 5	IDENT = Credit mit Summe
5-7	X X X	Kellner Nummer
8-9	0 X	Preisebene (0 0) ist Default
10-17	X X X X X X,X X	Credit Summe(in Groschen)

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	3 6	IDENT = Credit unbegrenzt
5-7	X X X	Kellner Nummer

Abrechnungsanforderungen:

NODE = 00 Alle Platinen Gemeinsam

IDENT = PROGPART (5)

SID = PRTSPATA (42) Gesamt Sparten Tag
 PRTSPAMA (43) Gesamt Sparten Monat
 PRTAKTAB (44) Alle Kellner Tag Summe, Bedienungsgeld
 PRTAKMAB (45) Alle Kellner Monat Summe, Bedienungsgeld
 PRTKETAB (46) Kellner Sparten Tag
 PRTKEMAB (47) Kellner Sparten Monat

 PRTKETDA (48) Kellner Detail Tag (alt Nicht verwenden)
 PRTKEMDA (49) Erweiterte Kellner Detail Abfrage
 PRTDETTA (50) Gesamt Detail Tag
 PRTDETMO (51) Gesamt Detail Monat

 PRTAKTST (62) Teste Kellner Bonierungen

INDEX = Kellnernummer (1-128)

Gesamt Sparten Abrechnung

Position	Wert	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPAR
5-6	4 2 (4 3)	SID = Gesamt Sparten Tag (Monat)

Antwort:

Position	Wert	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPAR
5-6	4 2 (4 3)	SID = Gesamt Sparten Tag (Monat)
7-10	X X X X	INDEX = Nicht Verwendet
11-18	X X X X X X,X X	Sparte 1 Preis 1
19-26	X X X X X X,X X	Sparte 1 Preis 2
27-34	X X X X X X,X X	Sparte 1 Preis 3
35-42	X X X X X X,X X	Sparte 1 Preis 4
43-50	X X X X X X,X X	Sparte 2 Preis 1
51-58	X X X X X X,X X	Sparte 2 Preis 2
59-66	X X X X X X,X X	Sparte 2 Preis 3
67-74	X X X X X X,X X	Sparte 2 Preis 4
75-		.
-618		.
619-626	X X X X X X,X X	Sparte 20 Preis 1
627-634	X X X X X X,X X	Sparte 20 Preis 2
635-642	X X X X X X,X X	Sparte 20 Preis 3
643-650	X X X X X X,X X	Sparte 20 Preis 4
651-658	X X X X X X,X X	Summe Storno Preis 1
659-666	X X X X X X,X X	Summe Storno Preis 2
667-674	X X X X X X,X X	Summe Storno Preis 3
675-682	X X X X X X,X X	Summe Storno Preis 4
683-690	X X X X X X,X X	Summe Scheck Preis 1
691-698	X X X X X X,X X	Summe Scheck Preis 2
699-706	X X X X X X,X X	Summe Scheck Preis 3
707-714	X X X X X X,X X	Summe Scheck Preis 4
715-722	X X X X X X,X X	Summe Kreditk Preis 1
723-730	X X X X X X,X X	Summe Kreditk Preis 2
731-738	X X X X X X,X X	Summe Kreditk Preis 3
739-746	X X X X X X,X X	Summe Kreditk Preis 4
747-754	X X X X X X,X X	Summe Zubuch Preis 1
755-762	X X X X X X,X X	Summe Zubuch Preis 2
763-770	X X X X X X,X X	Summe Zubuch Preis 3
771-778	X X X X X X,X X	Summe Zubuch Preis 4
779-786	X X X X X X,X X	Summe Abbuch Preis 1

787-794	X X X X X X,X X	Summe Abbuch Preis 2
795-802	X X X X X X,X X	Summe Abbuch Preis 3
803-818	X X X X X X,X X	Summe Abbuch Preis 4
819-822	X X X X	Jahr Letzte Löschung
823-824	X X	Monat 1-12
825-826	X X	Tag 1-31
827-828	X X	Std 1-23
829-830	X X	Min 1-59
831-834	X X X X	Jahr Aktuelles Datum
835-836	X X	Monat 1-12
837-838	X X	Tag 1-31
839-840	X X	Std 1-23
841-842	X X	Min 1-59

Alle Kellner Abrechnung

Position	Wert	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPAR
5-6	4 4 (4 5)	SID = Alle Kellner Tag (Monat)

Antwort:

Es werden nur die Kellner übertragen, welche einen Umsatz aufweisen.

Die Kellnersumme werden in Aufsteigender Reihenfolge nach Kellnern gesendet.

ACHTUNG: Wenn größere Lücken zwischen den Kellnernummern vorhanden sind, müssen längere Timeouts gesetzt werden, da der Rechner alle Kellner testen muß.

Position	Wert	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPAR
5-6	4 4 (4 5)	SID = Alle Kellner Tag (Monat)
7-10	X X X X	INDEX = Nicht Verwendet
11-13	X X X	Kellner Nummer (1-128)
14-21	X X X X X X.X X	Kellner Summe
22-29	X X X X X X.X X	Kellner Bediengeld
30-32	X X X	Kellner Nummer (1-128)
33-40	X X X X X X.X X	Kellner Summe
41-48	X X X X X X.X X	Kellner Bediengeld
43-		.
-		.
-		.

Kellner Spartenabrechnung

Position	Wert	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPAR
5-6	4 6 (4 7)	SID = Kellner Sparten Tag (Monat)
7-10	X X X X	INDEX = Kellnernummer

Antwort:

Position	Wert	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPAR
5-6	4 6 (4 7)	SID = Kellner Sparten Tag (Monat)
7-10	X X X X	INDEX = Kellnernummer
11-18	X X X X X X,X X	Sparte 1
19-26	X X X X X X,X X	Sparte 2
27-		.
-		.
-		.
163- 170	X X X X X X,X X	Sparte 20
171- 178	X X X X X X,X X	Summe Storno
179- 186	X X X X X X,X X	Summe Scheck in vorbereitung
187- 194	X X X X X X,X X	Summe Kreditk in vorbereitung
195- 202	X X X X X X,X X	Summe Zubuch in vorbereitung
203- 210	X X X X X X,X X	Summe Abbuch in vorbereitung
211- 218	X X X X X X,X X	Summe Abbuch Tische
219- 226	X X X X X X,X X	Reserve
227- 230	X X X X	Jahr Letzte Löschung
231- 232	X X	Monat 1-12
233- 234	X X	Tag 1-31
235- 236	X X	Std 1-23
237- 238	X X	Min 1-59
239- 242	X X X X	Jahr Aktuelles Datum
243- 244	X X	Monat 1-12
245- 246	X X	Tag 1-31
247- 248	X X	Std 1-23
249- 250	X X	Min 1-59

Löschen der Speicher:

Es werden immer die Detailspeicher mitgelöscht.

Alle offenen Tische und noch nicht übertragene Bons werden ebenfalls gelöscht.

Position	Wert	Bedeutung
1-2	0 0	NODE = Global
3-4	0 6	IDENT = EXECPAR
5-6	4 2 (4 3)	SID = Gesamt Sparten Tag (Monat)

Position	Wert	Bedeutung
1-2	0 0	NODE = Global
3-4	0 6	IDENT = EXECPAR
5-6	4 4 (4 5)	SID = Alle Kellner Tag (Monat)

Position		Wert		Bedeutung
1 - 2		0 0		NODE = Global
3 - 4		0 6		IDENT = EXECPAR
5 - 6		4 6 (4 7)		SID = Kellner Sparten Tag (Monat)
7 - 10		X X X X		INDEX = Kellnernummer

Alle offenen Tische und noch nicht übertragene Bons werden gelöscht.

Position		Wert		Bedeutung
1 - 2		0 0		NODE = Global
3 - 4		0 6		IDENT = EXECPAR
5 - 6		5 5 (5 6)		SID = Alle Kellner+Gesamt Tag (Monat)

Position		Wert		Bedeutung
1 - 2		0 0		NODE = Global
3 - 4		0 6		IDENT = EXECPAR
5 - 6		5 7		SID = Div. Sonder Speicher

Anfrage: Credit-Speicher

Position	Wert	Bedeutung
1 - 2	0 0	NODE = Global
3 - 4	0 5	IDENT = PROGPARG
5 - 6	6 4	SID = Creditspeicher
7 - 10	X X X X	INDEX = Kellnernummer

NODE:

wenn der Wert für **NODE = 0** dann wird der globale Creditnode Übertragen.
Ist in der Schankanlage jedoch *Credit lokal* eingestellt, dann müssen die Nodes einzeln abgefragt werden.

INDEX (Kellnernummer):

Wenn als Kellnernummer=0 angefordert wird so wird der komplette Creditspeicher übertragen.
Ansonsten werden nur die dem Kellner entsprechenden Datensätze gesendet. (Autocreditmode)
Kellnernummern größer als 128 sind Cocktailcredite

Antwort:

Position	Wert	Bedeutung
1 - 2	0 0	NODE = Global
3 - 4	0 5	IDENT = PROGPARG
5 - 6	6 4	SID = Creditspeicher
7 - 10	X X X X	INDEX = Kellnernummer
11 - 13	X X X	Kellnernummer
14 - 18	X X X X X	Stück
19 - 23	X X X X X	Plu-Nummer
24 - 26	X X X	Kellnernummer
27 - 31	X X X X X	Stück
32 - 36	X X X X X	Plu-Nummer
37 -		.
-		.
-		.

Kellner-Programm:

Schank <=> Kassa/PC

Komplett:

IDENT = KELPAR (20)

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPARG
5-6	0 2 (9 0)	IDENT = KELPAR (KELPAREX)
7-10	X X X X	INDEX = Kellner-Nummer
11-18	X X X X X X X X	Flags
19-21	X X X	Tages Ausdruck Nummer
22-33 (12)	ASCII	12 Zeichen Name
34-38	X X X X X	Untere Reviergrenze (kl.Tischnummer)
39-43	X X X X X	Obere Reviergrenze (gr.Tischnummer)

44-48	X X X X X	Aktueller Tisch
-------	-----------	-----------------

Erweitert:

IDENT = KELPAREX (90)

Position	WERT	Bedeutung
1-48		Werte siehe Kellnerprogramm
49-50	X X	Karten Ident[0]
51-52	X X	Karten Ident[1]
53-54	X X	Karten Ident[2]
55-56	X X	Karten Ident[3]
57-58	X X	Karten Ident[4]
59-60	X X	Karten Ident[5]

Flags:

Flags werden als ASCII-Hex Zahl übertragen und entsprechen einem 32 Bit-Feld.

KB_ENA	00000001	0 = Kellner Enable
KB_OBER	00000002	1 = Oberkellner
KB_HAUS	00000004	2 = Hauskellner
KB_TIZW	00000008	3 = Tischzwang
KB_STOR	00000010	4 = Storno erlaubt
KB_FREI	00000020	5 = Freipreiseingabe
KB_TIAB	00000040	6 = Tischabrechnung erlaubt
KB_TIOF	00000080	7 = Tischeröffnung erlaubt
KB_ABRE	00000100	8 = Eigene Kellnerabrechnung erl.
KB_DRUCK	00000200	9 = Kellner ausdruck erlaubt
KB_CLEAR	00000400	10 = Kellner loeschen erlaubt
KB_PSEUD	00000800	11 = Pseudorechnung erlaubt
KB_PRE2	00001000	12,13= Preis 1-4
KB_SUP2	00002000	
KB_EINST	00004000	14 = Einstellschlüssel
KB_KASS	00008000	15 = Kassa-übertragen
KB_ALTI	00010000	16 = Tisch Besetzt wird ignoriert
KB_CRDZ	00020000	17 = Kellner arbeitet auf Credit
KB_CRDB	00040000	18 = Kellner boniert auf Credit
KB_CRDA	00080000	19 = Kellner arbeitet auf Autocredit
KB_STOE	00100000	20 = reserviert (intern)
KB_BODR	00200000	21 = Bon - Druck Sperre
KB_LAGD	00400000	22 = Lagerausdruck enable
KB_SYST	00800000	23 = Kellnerstift ist CHEFSCHLOSS
KB_REIN	01000000	24 = Kellnerstift ist Reinigung
	02000000	25 = frei
	04000000	26 = frei
	08000000	27 = frei

KB_DEFT 10000000 28 = Defaulttisch
KB_TIAK 20000000 29 = Aktuelle Tischnummer gilt
40000000 30 = frei (intern)
KB_WECHS 80000000 31 = Kellnerwechsel (intern)

Kellner-Karten ID setzen:

IDENT = KELPARID

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPAR
5-6	9 1	IDENT = KELPARID
7-10	X X X X	INDEX = Kellner-Nummer
11-12	X X	Karten Ident[0]
13-14	X X	Karten Ident[1]
15-16	X X	Karten Ident[2]
17-18	X X	Karten Ident[3]
19-20	X X	Karten Ident[4]
21-22	X X	Karten Ident[5]

Kellner-Karten ID löschen

IDENT = KELPARCL

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPAR
5-6	9 2	IDENT = KELPARCL
7-10	0 0 0 0	INDEX = immer alle Kellner

PLU-Programm:

Schank <=> Kassa/PC

Komplette Plu-Programmierung

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGP
5-6	0 3	IDENT = PLUPAR
7-10	X X X X	INDEX = PLU-Nummer
11-18	X X X X X X X X	Flags
19-23	X X X X X	1. Credit PLU
24-26	X X X	Sparten Nummer
27-42 (16)	ASCII	16 Zeichen Name
43-50	X X X X X X,X X	8 Stellen Preis 1 in Groschen
51-58	X X X X X X,X X	8 Stellen Preis 2
59-66	X X X X X X,X X	8 Stellen Sub Preis 1
67-74	X X X X X X,X X	8 Stellen Sub Preis 2
75-79	X X X X X	2. Credit PLU
80-84	0 0 0 0 0	Reserviert
85-89	X X X X X	Cocktail Getränk
90-94	X X X X X	Convertierungs PLU (für Registrierkassen)
95-99	X X X X X	3. Credit PLU

Flags der PLU-Programmierung:

Flags werden als ASCII-Hex Zahl übertragen und entsprechen einem 32 Bit-Feld.

PL_ENA 00000001	PLU erlaubt/gesperrt
PL_COC 00000002	PLU = Coctail
PL_SU1 00000004	Sub Preis 1 erlaubt
PL_SU2 00000008	Sub Preis 2 erlaubt
PL_SCH 00000010	Schulteskassa enable
PL_GEW 00000020	Statt Stueck Gewicht (nicht verwendet)
PL_FPR 00000040	Statt Fixpreis Freipreis
PL_KAS 00000080	Plu nicht an Kasse senden (beim Zapfen)
PL_BON 00000100	Plu darf über Nummer boniert werden
PL_2CR 00000200	nur 2 CreditPLU's aber mit Stk statt Credit 3
PL_EXT 00000400	PLU=in externem Geraet

PL_KDR 0x00200000	Ausdruck auf Schlossdrucker
PL_DR1 0x00400000	Ausdruck Drucker 1
PL_DR2 0x00800000	
PL_DR3 0x01000000	
PL_DR4 0x02000000	
PL_DR5 0x04000000	
PL_DR6 0x08000000	
PL_DR7 0x10000000	
PL_DR8 0x20000000	

PL_DR9 0x40000000

PL_DR10 0x80000000 Ausdruck Drucker 10

Reduzierte Plu-Programmierung

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGPAR
5-6	9 3	IDENT = PLUPAREX
7-10	X X X X	INDEX = PLU-Nummer
11-18	X X X X X X X X	Flags
19-21	X X X	Sparten Nummer
22-37 (16)	ASCII	16 Zeichen Name
38-45	X X X X X X,X X	8 Stellen Preis 1 in Groschen
46-53	X X X X X X,X X	8 Stellen Preis 2
54-61	X X X X X X,X X	8 Stellen Sub Preis 1
62-69	X X X X X X,X X	8 Stellen Sub Preis 2

Flags:

Flags werden als ASCII-Hex Zahl übertragen und entsprechen einem 32 Bit-Feld.

PL_ENA 00000001	PLU erlaubt/gesperrt
PL_SU1 00000004	Sub Preis 1 erlaubt
PL_SU2 00000008	Sub Preis 2 erlaubt

SESAM 4 Version:

Schank <=> Kassa/PC

Versionsanfrage:

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGP
5-6	1 3	IDENT = SESPAR
7-10	0 0 0 1	INDEX = 1

Versionsantwort:

Position	WERT	Bedeutung
1-2	0 0	NODE = Global
3-4	0 5	IDENT = PROGP
5-6	1 3	IDENT = SESPAR
7-10	0 0 0 1	INDEX = 1
11-30	ASCII	20 Zeichen Versionstext
31-32	X X	Knoten Nummer
33-34	X X	Anzahl der Platinen
35-36	ASCII	2 Zeichen Sub Versionstext Die Subversion wird erst ab der Version V 1.65af gesendet