



SESAM 4

HUGIN S 500

HUGIN S 600

SESAM 148

Kassen - Protokoll

HUGIN S 500/S 600 - KASSAPROTOKOLL	3
LSV2 Übertragungsprotokoll	3
LSV2 Senden: int tx_lsv2(char *tx)	4
LSV2 Empfangen: int rx_lsv2(char *tx)	5
Datensätze	6
Tischanfrage: Schank => Kassa/PC	6
Tischantwort: Kassa/PC => Schank	6
Bonsendung: Schank => Kassa/PC	7
Bonantwort: Kassa/PC => Schank	7
Aufzählen des Kreditspeichers: (SESAM 148)	8
Kreditsendung SESAM 148: Kassa/PC => Schank	8
Kreditantwort SESAM 148: Schank => Kassa/PC	8
Aufzählen des Kreditspeichers: (nur SESAM 4 und S 600)	9
Kreditsendung SESAM 4: Kassa/PC => Schank	9
Kreditantwort SESAM 4: Schank => Kassa/PC	9
Löschen des Schankkopfes: (nur SESAM 4 und S 600) Tagesabschluß	10
Speicher löschen: Kassa/PC => Schank	10
Löschantwort: Schank => Kassa/PC	10

HUGIN S 500/S 600 - Kassaprotokoll

In Verbindung mit der Kasse HUGIN S 500/S 600 verwendet der Schankcomputer SESAM 4 Datensätze des Schankcomputers SESAM 148.

Die Datensätze **M_TANF**, **M_TIQT** und **M_BON** werden auch in Verbindung mit **SESAM 148** verwendet.

Beschreibung der im System vorkommenden Messages

jedes Byte einer Message wird in 2 Bytes ASCII umgewandelt !

Jede Message besteht aus einem Messageheader und den Messagedaten

Messanf	+ 0	Zielprozess
	+ 1	Zielknoten
	+ 2	Absenderprozess
	+ 3	Absenderknoten
	+ 4	Messagecode BIT 7 = 0 Keine Daten
	+ 5	Länge der Daten
	+ 6	1.Datenbyte
	+ 7	2. =
		.
		.
		.

Verbindung SESAM 4 mit S 500/S 600:

Die Bytes Zielprozess, Zielknoten, Absenderprozess, Absenderknoten werden im SESAM 4 nicht ausgewertet.

LSV2 Übertragungsprotokoll

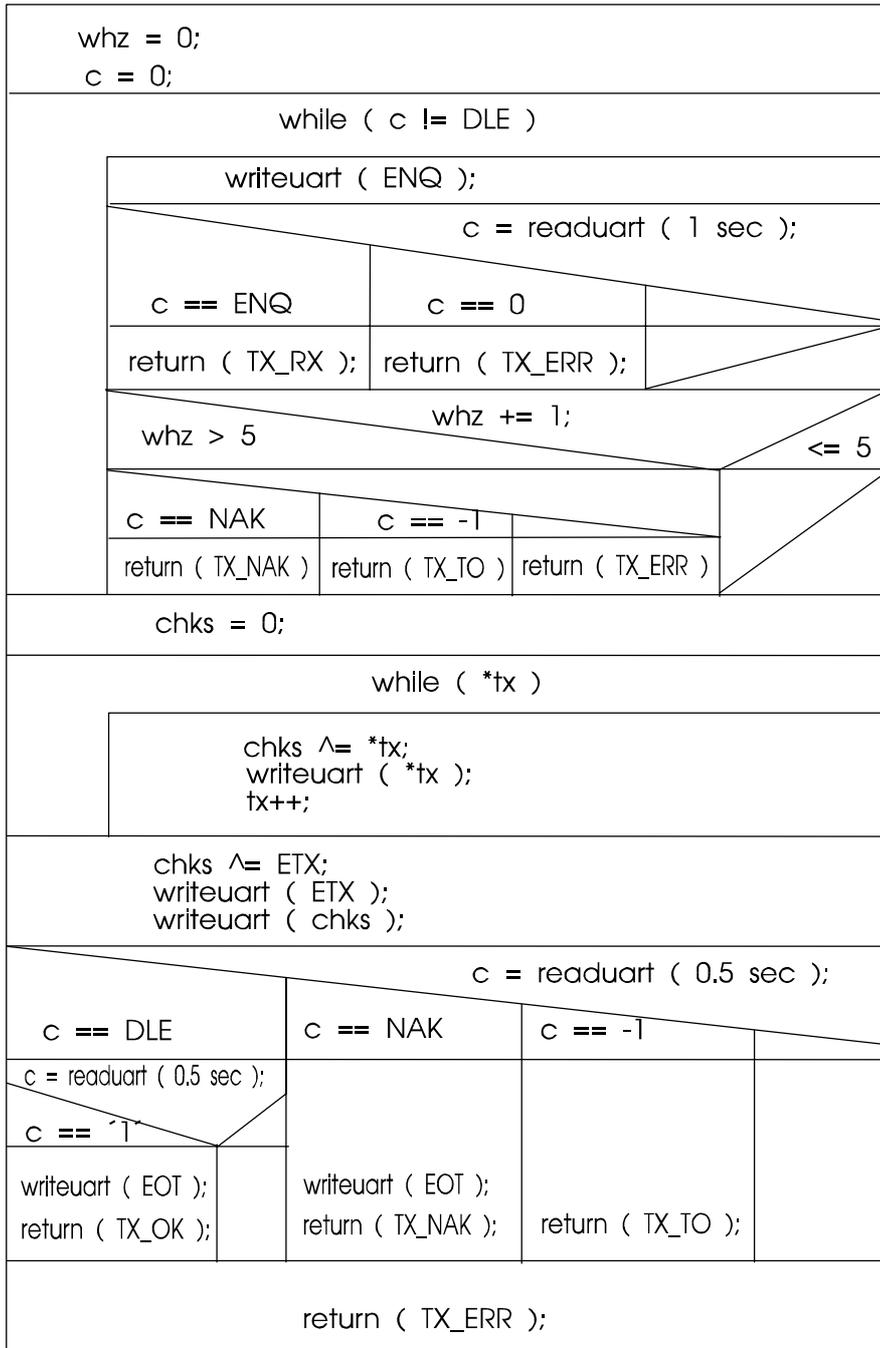
<u>Sender</u>		<u>Empfänger</u>	
ENQ	(0x05)	=>	Timeout 1 sec.
		<=	DLE '0' (0x010 0x030)
STX	(0x02)	=>	Timeout 0,5 sec. / Zeichen
Daten....		=>	
ETX	(0x03)	=>	
LRC		=>	LRC = EXOR (Daten+ETX) ohne STX
		<=	Timeout 0,5 sec DLE '1' (0x010 0x031)
EOT	(0x04)	=>	

Timeouts gelten für 9600 Bd

In der Schankanlage wird das LSV2 Protokoll folgendermaßen durchgeführt:

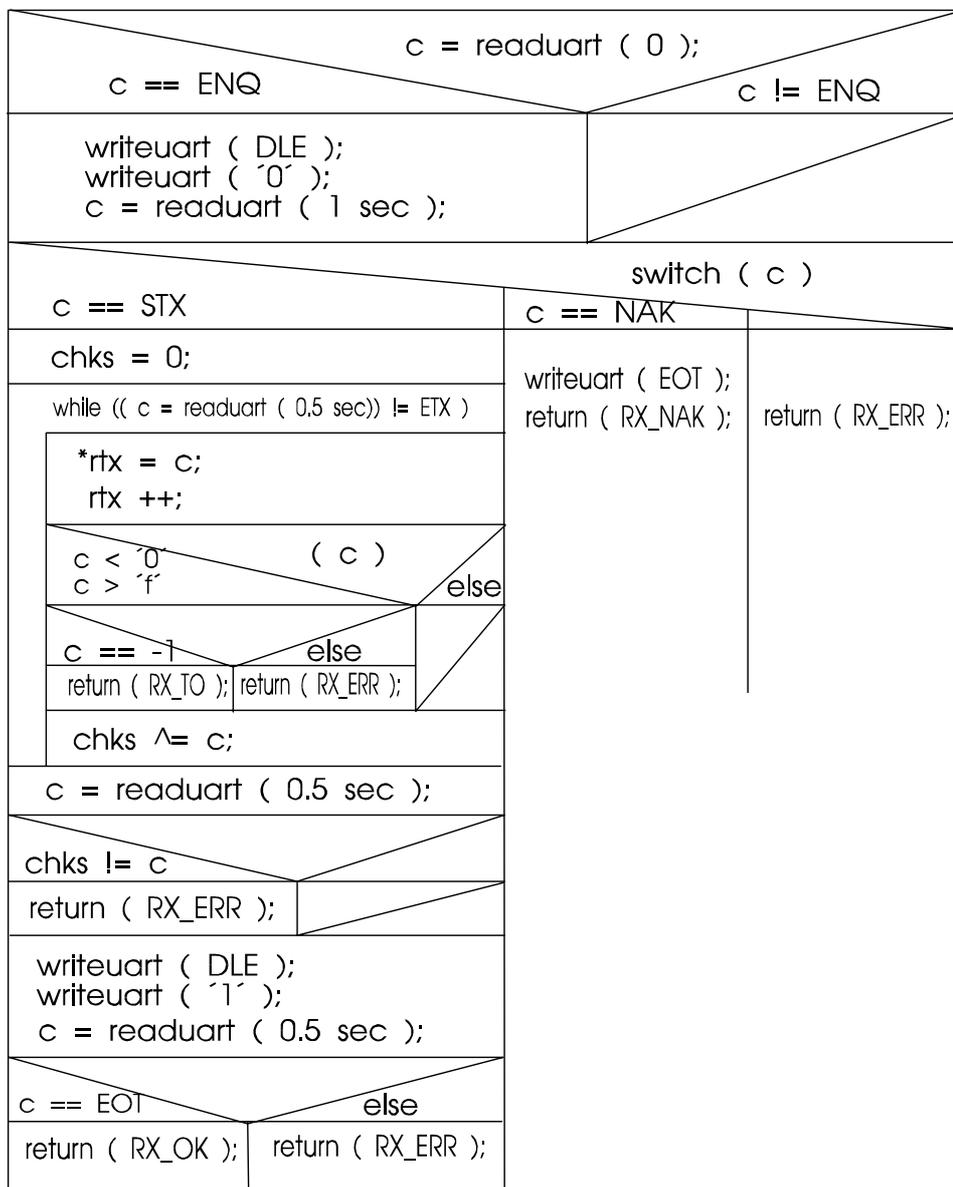
LSV2 Senden: int tx lsv2(char *tx)

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

Tischanfrage: Schank => Kassa/PC

Position	Wert	Bedeutung
1 - 2	5 5	ZIEL-TASK
3 - 4	4 0	ZIEL-KNOTEN
5 - 6	2 0	ABSENDER-TASK
7 - 8	4 0	ABSENDER-KNOTEN
9 - 10	C 9	M_TANF (0xC9) MessageTyp Tischanfrage
11 - 12	0 4	Datensatzlänge = 4
13 - 14	X X	Kellnernummer (0x01 - 0x7F) -127
15 - 18	X X X X	Tischnummer (0x0001 - 0xFFFF) auf den Stellen 15,16 wird das LOW-Byte und dann das HIGH-Byte auf die Stellen 17,18 geschrie
19 - 20	0 X	Open

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

Tischantwort: Kassa/PC => Schank

Position	Wert	Bedeutung
1 - 2	A 0	ZIEL-TASK ist (Anfrage ABSENDER-TASK 0x80)
3 - 4	4 0	ZIEL-KNOTEN ist Anfrage ABSENDER-KNOTEN
5 - 6	5 5	ABSENDER-TASK ist Anfrage ZIEL-TASK
7 - 8	4 0	ABSENDER-KNOTEN ist Anfrage ZIEL-KNOTEN
9 - 10	C A	M_TIQT (0xCA) MessageTyp Tischquittierung
11 - 12	0 2 (0 6)	Datensatzlänge = 2 (oder 6 falls zusätzliche
13 - 14	X X	Kellnernummer (0x01 - 0x7F) -127
15 - 16	X X	Tischantwort
17 - 24	X X X X X X X X	optional Tisch-Summe oder Kellnernummer bei Tisch besetzt (in Groschen) Hex zuerst LSByte -MSByte

Tischantwort:

TIA_OK	0	"Tisch Su:%8u.%02u"	17 - 20 ist TischSumme (optional)
TIA_BESETZT	1	"Tisch besetzt Kel%3u"	17 - 20 ist Kellnernummer (optional)
TIA_ZUKLEIN	2	"Tisch zu klein !"	
TIA_ZUGROSS	3	"Tisch zu gross !"	
TIA_NERLAUB	4	"Tisch nicht erlaubt"	
TIA_NEU	5	"Tisch neu angelegt"	
TIA_NOMEM	6	"Tisch Speicher voll"	
TIA_BLOCK	7	"Tisch blockiert"	17 - 20 ist Kellnernummer (optional)
TIA_LEER	8	"Tisch leer !"	

Bonsendung: Schank => Kassa/PC

ACHTUNG: Die Kasse darf aufgrund einer Bonsendung (erfolgt im Autokreditmode wenn der Kellner keinen Kredit hat, oder ein Kellner, der nicht auf Kredit arbeitet) KEINEN Kreditsatz senden.

Position	Wert	Bedeutung
1 - 2	5 5	ZIEL-TASK
3 - 4	4 0	ZIEL-KNOTEN
5 - 6	2 0	ABSENDER-TASK
7 - 8	4 0	ABSENDER-KNOTEN
9 - 10	B 5	M_BON (xB5) MessageTyp Bonsendung
11 - 12	0 7	Datensatzlänge = 7
13 - 14	X X	Kellnernummer (0x01 - 0x7F) -127
15 - 18	X X X X	PLU Nummer (0x0001 - 0xFFFF)
19 - 22	X X X X	Stückzahl (0x0001 - 0xFFFF)
23 - 26	X X X X	Tischnummer(0x0001 - 0xFFFF)

Bonantwort: Kassa/PC => Schank

Position	Wert	Bedeutung
1 - 2	A 0	ZIEL-TASK ist (Anfrage ABSENDER-TASK 0x80)
3 - 4	4 0	ZIEL-KNOTEN ist Anfrage ABSENDER-KNOTEN
5 - 6	5 5	ABSENDER-TASK ist Anfrage ZIEL-TASK
7 - 8	4 0	ABSENDER-KNOTEN ist Anfrage ZIEL-KNOTEN
9 - 10	0 5	M_FERT (0x05) MessageTyp Daten akzeptiert
9 - 10	0 6	M_ERRO (0x06) MessageTyp Daten nicht akzeptiert
9 - 10	0 7	M_WART (0x07) MessageTyp Kasse für Daten nicht be

Beispiel:

Kellner 11
Produkt 21
Stück 13
Tischnummer 150

Message:

Pos 1	55h	ZIEL-TASK	= PC oder KASSA
Pos 3	40h	ZIEL-KNOTEN	= PC oder KASSA
Pos 5	20h	ABSENDER-TASK	ist wichtig für Antwort von PC an Schank wird ZIEL
Pos 7	40h	ABSENDERKNOTEN	
Pos 9	B5h	MESSAGECODE	
Pos11	07h	Messagelänge	
Pos13	0Bh		Kellnernummer (11d)
Pos15	15h		Produktnummer low Byte (21d)
Pos17	00		Produktnummer high Byte
Pos19	0dh		Stückzahl low Byte (13d)
Pos21	00		Stückzahl high Byte
Pos23	96h		Tischnummer low Byte (150d = 96h)
Pos25	00		Tischnummer high Byte

Diese Werte werden in ASCII umgewandelt sodaß, folgender Text über die serielle Schnittstelle gesandt wird:

'55402041B5070B15000D009600'

Aufzählen des Kreditspeichers: (SESAM 148)

Der Kreditspeicher des SESAM 148 wird dann aufgezählt, wenn ein Kellner am SESAM 4 eine PLU boniert, die im SESAM 148 ausgegeben wird.

Das Aufzählen des Kreditspeichers im Schankkopf erfolgt in ähnlicher Weise wie die Bonsendung. Der Datenfluß ist jedoch umgekehrt. (von der Kassa zur Schank)

Zapft ein Kellner auf Kredit, (bzw. Autokredit) wird KEIN Datensatz zur Kassa gesendet.

Kreditsendung SESAM 148: Kassa/PC => Schank

Position	Wert	Bedeutung
1 - 2	5 7	ZIEL-TASK FILEM_
3 - 4	4 0	ZIEL-KNOTEN
5 - 6	2 0	ABSENDER-TASK
7 - 8	4 0	ABSENDER-KNOTEN
9 - 10	D E	M_CRED (xDE) MessageTyp Credit aufzählen
11 - 12	0 5	Datensatzlänge = 5
13 - 16	X X X X	PLU Nummer (0x0001 - 0xFFFF)
17 - 18	2 B	Vorzeichen ('+') aufzählen
19 - 22	X X X X	Stückzahl (0x0001 - 0xFFFF)
23 - 24	0 1	Nur bei subtrahieren Mode 1 = DO (0 = TEST)

Kreditantwort SESAM 148: Schank => Kassa/PC

Position	Wert	Bedeutung
1 - 2	A 0	ZIEL-TASK ist (Anfrage ABSENDER-TASK 0x80)
3 - 4	4 0	ZIEL-KNOTEN ist Anfrage ABSENDER-KNOTEN
5 - 6	5 7	ABSENDER-TASK ist Anfrage ZIEL-TASK
7 - 8	4 0	ABSENDER-KNOTEN ist Anfrage ZIEL-KNOTEN
9 - 10	0 5	M_CRQT (0xDF) Credit Antwort
11 - 12	0 2	Datensatzlänge = 5
13 - 16	X X X X	PLU Nummer (0x0001 - 0xFFFF)
17 - 18	0 0	Mode (=1 wenn nicht storniert werden kann)
19 - 22	X X X X	Stückzahl (0x0001 - 0xFFFF)

Aufzählen des Kreditspeichers: (nur SESAM 4 und S 600)

Der Kreditspeicher der Schank wird dann aufgezählt, wenn ein Kellner an der Kasse eine PLU boniert, die in der Schank ausgegeben wird (wenn möglich in der Kasse programmierbar).

Das Aufzählen des Kreditspeichers im Schankkopf erfolgt in ähnlicher Weise wie die Bonsendung. Der Datenfluß ist jedoch umgekehrt. (von der Kassa zur Schank)

Zapft ein Kellner auf Kredit, (bzw. Autokredit) wird KEIN Datensatz zur Kassa gesendet.

ACHTUNG: Die Kasse darf aufgrund einer normalen Bonsendung (erfolgt im Autokreditmode wenn der Kellner keinen Kredit hat, oder ein Kellner, der nicht auf Kredit arbeitet) KEINEN Kreditsatz zurücksenden.

Kreditsendung SESAM 4: Kassa/PC => Schank

Position	Wert	Bedeutung
1 - 2	5 5	ZIEL-TASK
3 - 4	4 0	ZIEL-KNOTEN
5 - 6	2 0	ABSENDER-TASK
7 - 8	4 0	ABSENDER-KNOTEN
9 - 10	D E	M_CRED (xDE) MessageTyp Credit aufzählen
9 - 10	D F	M_CRDS (xDF) MessageTyp Credit abziehen
11 - 12	0 5	Datensatzlänge = 5
13 - 14	X X	Kellnernummer (0x01 - 0x7F) -127
15 - 18	X X X X	PLU Nummer (0x0001 - 0xFFFF)
19 - 22	X X X X	Stückzahl (0x0001 - 0xFFFF)

Kreditantwort SESAM 4: Schank => Kassa/PC

Position	Wert	Bedeutung
1 - 2	A 0	ZIEL-TASK ist (Anfrage ABSENDER-TASK 0x80)
3 - 4	4 0	ZIEL-KNOTEN ist Anfrage ABSENDER-KNOTEN
5 - 6	5 5	ABSENDER-TASK ist Anfrage ZIEL-TASK
7 - 8	4 0	ABSENDER-KNOTEN ist Anfrage ZIEL-KNOTEN
9 - 10	0 5	M_FERT (0x05) MessageTyp Daten akzeptiert
9 - 10	0 6	M_ERRO (0x06) MessageTyp Daten nicht akzeptiert
11 - 12	0 2	Datensatzlänge = 2 im Fehlerfall
13 - 16	X X X X	stornierte Stückzahl (0x0001 - 0xFFFF)

Löschen des Schankkopfes: (nur SESAM 4 und S 600) Tagesabschluß

Die Funktion Löschen muß in der Kassa gesperrt werden können, da die Bedeutung der Kellnernummern im Getränkecomputer SESAM 148 anders ist.

Bedeutung der Kellnernummer zum Löschen:

Kellnernummer 0: Alle Kellner Tag & Tag gesamt löschen: Es werden alle Tages-Kellnerspeicher, der Tagesgesamtspeicher alle offenen Rechnungen sowie der Kreditspeicher gelöscht.

Kellnernummer 1-127: (1-0x7F) Bit 7 = 0 Einzel Kellner-Tagesspeicher

Kellnernummer 128: (0x80) Alle Kellner Monat & Monatgesamt löschen: Es werden alle Monats-Kellnerspeicher und der Monatsgesamtspeicher gelöscht.

Kellnernummer 129-255: (0x81-0xFF) Bit 7 = 1 Einzel Kellner-Monatsspeicher

Speicher löschen: Kassa/PC => Schank

Position	Wert	Bedeutung
1 - 2	5 5	ZIEL-TASK
3 - 4	4 0	ZIEL-KNOTEN
5 - 6	2 0	ABSENDER-TASK
7 - 8	4 0	ABSENDER-KNOTEN
9 - 10	9 4	M_CLER (x94) MessageTyp Löschen
11 - 12	0 1	Datensatzlänge = 1
13 - 14	X X	Kellnernummer (0x00 - 0xFF)

Löschantwort: Schank => Kassa/PC

Position	Wert	Bedeutung
1 - 2	A 0	ZIEL-TASK ist (Anfrage ABSENDER-TASK 0x80)
3 - 4	4 0	ZIEL-KNOTEN ist Anfrage ABSENDER-KNOTEN
5 - 6	5 5	ABSENDER-TASK ist Anfrage ZIEL-TASK
7 - 8	4 0	ABSENDER-KNOTEN ist Anfrage ZIEL-KNOTEN
9 - 10	0 5	M_FERT (0x05) Speicher gelöscht
9 - 10	0 6	M_ERRO (0x06) Fehler beim Löschen