

Anleitung

Com-Server



Multi-Port Protokoll-Mode per RS232

Rel. 2.6, Mai 2003

Typen:

C3: PC104 10BT+BNC

C4: 10BT

C5: 100/10BT

C8: Highspeed

1. Der serielle Protokoll-Mode des Com-Servers

Das nachfolgend beschriebene Protokoll soll eine definierte Kommunikation zwischen dem Com-Server und dem an der seriellen Schnittstelle angeschlossenen Gerät ermöglichen. Es realisiert ein Socket-Interface, das dem Endanwender an der seriellen Schnittstelle folgende Möglichkeiten gibt:

- ✓ Öffnen und Schließen von bis zu 8 Handles (Datenverbindungen) gleichzeitig
- ✓ Festlegen der Art der Netzwerkverbindung (UDP oder TCP) und der Verbindungsparameter (Ziel-IP-Adresse und -Port und Local Port des Com-Servers)
- ✓ Übermittlung von IP-Broadcasts
- ✓ Der Com-Server informiert das serielle Endgerät im Rahmen des Protokolls selbständig über Änderungen des System- oder Verbindungsstatus
- ✓ Paketorientierte Übertragung serieller Daten in beide Richtungen, d.h. Senden von Paketen mit Paketanfangszeichen, Paket-Header, Daten, Paketendezeichen
- ✓ Übermittlung der Netzwerk-Zieladresse (IP-Adresse, UDP-Port) an den Com-Server und dem Netzwerk-Absender an das serielle Endgerät mit den Daten
- ✓ Parametrierbares Paketrahmenzeichen (Paketende/Paketanfang) oder Zeichendopplungsverfahren, um keine Zeichen aus dem Datenstrom ausschließen zu müssen
- ✓ Konfiguration der seriellen Schnittstelle des Com-Servers
- ✓ Konfiguration der Netzwerkparameter des Com-Servers
- ✓ Lesen der Konfigurationen

Der Protokoll-Mode soll über Netzwerk fest konfiguriert werden, so daß beim Power Up der Modus bekannt ist. Alle Konfigurationen, die über die serielle Schnittstelle erfolgen, werden ebenfalls im EEPROM (nichtflüchtiger Speicher) abgespeichert und beim Power Up geladen.

1.1 Aktivierung des seriellen Protokolls

Die Verwendung des seriellen Protokolls muß im Com-Server konfiguriert werden. Rufen Sie das Konfigurationstool ("telnet *IP-Adresse* 1111") auf. Wählen Sie das Menü

Mode: SET TCP/IP → Port.. → Serial Socket Interface

Stellen Sie den Schalter "Serial Protocol" auf AKTIV. Soll das Zeichendopplungsverfahren verwendet werden (siehe Kapitel 3), stellen Sie zusätzlich den Schalter "Serial Coding" auf AKTIV.

Arbeiten Sie ohne Zeichendopplungsverfahren, konfigurieren Sie das Paketrahmenzeichen unter dem Menüpunkt "Protocol Char". Dieses Zeichen darf dann im Datenstrom nicht mehr verwendet werden!

Verlassen Sie die Untermenüs und speichern Sie die Eintragungen mit "Mode: SAVE" ab. Im Menü "Mode: MONITOR" des jeweiligen Ports erscheint der Eintrag: Multi Port Interface.

1.2 Aufbau eines Pakets

Festgelegt wird folgende grundsätzliche Paket-Struktur:

0	1	2	4	4+LEN	
PKT- END	TYPE	LEN		DATA	PKT- END
		Low Byte	High Byte		

PKT-END (BYTE)

Jedes Paket beginnt und endet mit einem definierten Character (PKT-END). Wird das Zeichendopplungsverfahren verwendet, hat PKT-END den Wert C0 hex (siehe Kapitel 3). Ansonsten wird der konfigurierte Character verwendet.

TYPE (BYTE)

Dieses Feld bezeichnet den Inhalt der folgenden Daten und ermöglicht die Erweiterbarkeit des Protokolls.

Anmerkung: Ist das LSB von TYPE = 1, handelt es sich um ein Paket mit Inhalt. Ist das LSB = 0, handelt es sich um einen Request zum Lesen (wird nur vom seriellen Endgerät verwendet). Siehe dazu die TYPE-Festlegungen.

LEN (WORD, 16bit)

Das LEN-Feld enthält die Länge der Daten (DATA) in Bytes. Der Com-Server füllt dieses Feld immer aus, das serielle Endgerät muß dieses Feld nicht ausfüllen und kann es auf Null setzen. Bei Anwendung des Zeichendopplungsverfahrens und Verwendung des LEN-Feldes muß die Länge vor der Dopplung der Character berechnet werden.

1.3 Die Paket-Typen

Die folgenden Definitionen für das Feld TYPE (Pakettypen) werden festgelegt. Es müssen nur die Paket-Typen verwendet werden, die zur Realisierung der Applikation benötigt werden und es können weitere TYPEs definiert werden, falls dies die Art der Applikation notwendig macht.

TYPE	Erläuterung
0	OK-Typ (Bestätigung der neuen Schnittstellenkonfiguration)
2	Request zum Senden der Basis-Konfiguration der seriellen Schnittstelle des Com-Servers
3	Übermitteln der Basis-Konfiguration der seriellen Schnittstelle
4	Request zum Senden der erweiterten Konfiguration der seriellen Schnittstelle des Com-Servers
5	Übermitteln der erweiterten Konfiguration der seriellen Schnittstelle
6	Request zum Senden der Netzwerk-Parameter des Com-Servers
7	Übermitteln der Netzwerk-Parameter
8	Request zum Senden der Com-Server-Statusinformation
9	Übermitteln der Com-Server-Statusinformation
10	Request zum Senden des Com-Server-Passworts
11	Übermitteln des Com-Server-Passworts
12	Request zum Senden der Com-Server-Software-Revision
13	Übermitteln der Com-Server-Software-Revision
14	Request zum Senden des System Passwords
15	Übermitteln des System Passwords
16	Request zum Senden des System Namens
17	Übermitteln des System Namens
18	Request zum Senden der MAC-Adresse des Com-Servers
19	Übermitteln der MAC-Adresse des Com-Servers
20	Request zum Senden der Zusatzkonfiguration
21	Übermitteln der Zusatzkonfiguration
22	Request zum Senden der Handle-Information
23	Übermitteln der Handle-Information
100	Öffnen eines Handles (Sockets) für eine Verbindung oder Lesen der Handle-Parameter
101	Schließen eines Handles (Sockets)
102	Request zum Senden des Verbindungsstatus
103	Übermitteln des Verbindungsstatus
104	Beenden einer TCP-Verbindung mit Reset und Schließen des Handles
110	Übertragen von Nutzdaten
253	Request für einen Software-Reset des Com-Servers

2. Verbindungskontrolle

2.1 Öffnen eines Handles / Sockets und Lesen der Handle-Parameter

Bevor der Com-Server Daten mittels UDP oder TCP übertragen kann, ist es notwendig, die jeweiligen Verbindungsparameter im Com-Server zu definieren. Der Com-Server speichert diese Parameter unter einer Handle-Nummer ab, die er ans Peripheriegerät zurückgibt. Mit der folgenden Struktur werden die Parameter übergeben:

0	1	2	3	4										
PKT- END	TYPE = 100	LENGTH = 11		Proto- col										
		Low Byte	High Byte	Byte										
5	6	7	8	9	10	11	12	13	14	15				
Prio- rity	Handle = 255	Source Port		Destination Port		Destination IP-Address Byte				PKT- END				
Byte	Byte	Low Byte	High Byte	Low Byte	High Byte	0	1	2	3					

Zum Öffnen des Handles müssen die Felder wie folgt ausgefüllt werden:

1. Handle für eine Verbindung über UDP

```

Protocol:          17 für UDP
Priority:          0 (Priorität des Handels beim Transport der Daten in Bezug
                  auf die Verarbeitung im Com-Server und auf die Reihenfolge
                  im Ethernet; noch ungenutzt, genaue Spezifikation erfolgt
                  später)
Handle:           255
Source Port:      Local UDP-Port, auf dem der Com-Server Daten empfangen soll
Destination Port: Ziel-UDP-Port oder 0*)
Destination IP:   Ziel-IP-Adresse oder 0.0.0.0*)
  
```

*) Enthalten diese Felder den Wert 0, muß die Zieladresse beim Versenden der Daten im Header des Datenpakets (TYPE 110) übergeben werden. So kann der Com-Server auf einem Port Daten von unterschiedlichen Absendern im Netz empfangen und auch an unterschiedliche Zieladressen versenden.

2. Handle für einen TCP-Client (der Com-Server öffnet die Netzwerkverbindung)

Protocol: 6 für TCP
Priority: 0 (Priorität des Handels beim Transport der Daten in Bezug auf die Verarbeitung im Com-Server und auf die Reihenfolge im Ethernet; noch ungenutzt, genaue Spezifikation erfolgt später)
Handle: 255
Source Port: Local TCP-Port, auf dem der Com-Server Daten übertragen soll
Destination Port: Ziel-TCP-Port
Destination IP: Ziel-IP-Adresse

3. Handle für einen TCP-Server (der Com-Server akzeptiert Verbindungen auf dem angegebenen Source Port)

Protocol: 6 für TCP
Priority: 0 (Priorität des Handels beim Transport der Daten in Bezug auf die Verarbeitung im Com-Server und auf die Reihenfolge im Ethernet; noch ungenutzt, genaue Spezifikation erfolgt später)
Handle: 200
Source Port: Local TCP-Port, auf dem der Com-Server Verbindungen akzeptieren und Daten übertragen soll
Destination Port: 0
Destination IP: 0.0.0.0

Der Com-Server antwortet zur Kontrolle mit dem gleichen Paket und trägt die Handle-Nummer ein. Gültige Handle-Nummern sind die Werte 0 bis 7. Enthält das Feld den Wert 255, ist der Vorgang fehlgeschlagen, weil alle Handle bereits vergeben sind. Enthält es den Wert 254, war ein oder mehrere Parameter falsch.

Wichtig: Die Handle-Nummer muß beim Versenden von Datenpaketen stets im Header des Datenpaketes angegeben werden.

Lesen der Handle-Parameter

Die Parameter eines bereits geöffneten Handles kann man lesen, indem man im Feld *Handle* die gewünschte Handle-Nummer einträgt und alle anderen Felder mit dem Wert 0 füllt. Der Com-Server antwortet mit der komplett ausgefüllten Struktur. Beinhaltet das Feld *Handle* in der Antwort den Wert 255, so ist der angeforderte Handle nicht geöffnet.

2.2 Schließen eines Handles / Sockets

Wird die Verbindung nicht mehr zum Übertragen von Daten benötigt, muß der Handle in jedem Fall wieder freigegeben werden, auch wenn die Verbindung z.B. von einem TCP-Server bereits beendet wurde. Im Feld "Handle" wird die Nummer des zu schließenden Handles eingetragen (zwischen 0 und 7).

0	1	2	4	5	
PKT- END	TYPE = 101	LENGTH = 1		Handle	PKT- END
		Low Byte	High Byte		

2.3 Zurücksetzen der Verbindung und Schließen des Handles / Sockets

Dieser TYPE kann nur bei einem TCP-Handle verwendet werden und sollte auch nur im Notfall angewendet werden, um einen Handle mit einer hängenden TCP-Verbindung wieder verwenden zu können. Empfängt der Com-Server diesen Type, setzt er die TCP-Verbindung mit Hilfe eines Reset zurück, löscht alle zum Handle gehörigen Buffer (Datenverlust!) und gibt den Handle wieder frei.

Im Feld "Handle" wird die Nummer des zu schließenden Handles eingetragen (zwischen 0 und 7).

0	1	2	4	5	
PKT- END	TYPE = 104	LENGTH = 1		Handle	PKT- END
		Low Byte	High Byte		

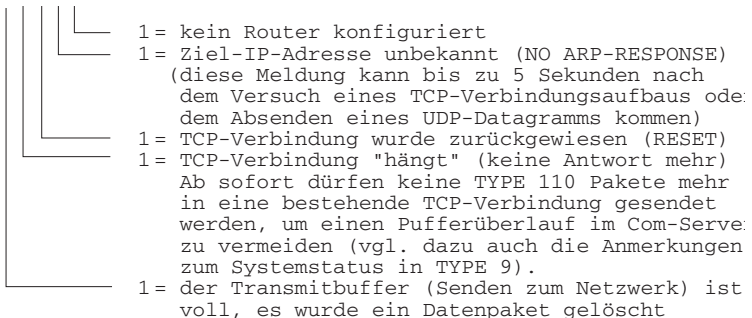
2.4 Übermitteln des Verbindungsstatus eines Handles

Dieses Paket enthält den Verbindungsstatus des entsprechenden Handles. Der Com-Server sendet dieses Paket, wenn sich der Inhalt von CONNECT ändert oder nach Empfang des Paket-Types 102 (Lesen des Verbindungsstatus).

0	1	2	4					
PKT- END	TYPE = 103	LENGTH = 8						
		Low Byte	High Byte					
4	5	6	7	8	9	10	11	12
Handle	Connect	Packet Count	IP-Adresse				State	PKT- END
			0	1	2	3		

- **HANDLE:** Handle der Verbindung. Handle enthält den Wert 255, wenn der Com-Server TYPE 102 für einen Handle empfangen hat, der nicht geöffnet ist. Die Felder IP-Adresse und State sind dann ungültig, das Feld CONNECT enthält den Handle, für den der Request empfangen wurde.
- **CONNECT:** In diesem Feld werden bitweise Fehler kodiert, die sich auf den Verbindungsstatus im Netzwerk beziehen. Hat Connect den Wert 0, ist kein Fehler aufgetreten.

Bit 7 ... 0
DB 000xxxxx B



- Packet Count: Zähler für alle im Com-Server in Bearbeitung befindlichen Datenpakete vom Typ 110 (Senderichtung Seriell -> Netzwerk). Der Zähler wird beim Öffnen des Handles auf Null gesetzt. Wird der Speicherplatz eines seriell empfangenen Paketes wieder freigegeben entweder durch Versenden des Paketes (UDP) oder durch Erhalt einer Quittung für das Paket (TCP) wird dem Zähler der Wert 1 addiert.

Dies ermöglicht eine Kontrolle, wieviele Datenpakete sich in den Netzwerkbuffer des Com-Servers befinden, die noch nicht vermittelt werden konnten.

- IP-Adresse: IP-Adresse, die den Fehler in Connect ausgelöst hat (dieser Wert kann bei UDP variieren)
- STATE: Status der Netzwerkverbindung
 0 = FREE (TCP-Client: keine aktive Verbindung, der Handle muß geschlossen werden)
 1 = IN_USE (UDP/TCP-Client u. -Server: aktive Verbindung)
 3 = WAITS (TCP-Client: wartet auf Verbindungsaufbau)
 4 = LISTEN(TCP-Server ohne aktive Verbindung)
nur internes Protokoll von W&T:
 5 = WAIT CLOSE (TCP-Server wartet auf Close Handle mit TYPE 101)

2.5 Request zum Senden des Verbindungsstatus eines Handles

Der Com-Server antwortet mit dem Paket-Type 103 und trägt den Verbindungsstatus ein.

0	1	2	4	5	
PKT- END	TYPE = 102	LENGTH = 1		Handle	PKT- END
		Low Byte	High Byte		

3. Übertragung von Daten

Dieser Pakettyp dient der Übertragung der Applikationsdaten. Der Com-Server liest die Daten aus dem Paket ein und leitet sie über das Ethernet an die entsprechende Netzwerkstation weiter. Die Daten, die der Com-Server von der Netzwerkstation empfängt, überträgt er mittels dieses Pakettyps an das serielle Endgerät.

0	1	2	3	4	5	6	7	
PKT- END	TYPE = 110	LEN = 10 + sizeof(DATA)		Proto- col	Handle	State		
		Low Byte	High Byte			Low Byte	High Byte	
8	9	10	11	12	13	14	n	
Destination Port		Destination IP-Address Byte:				DATA		PKT- END
Low Byte	High Byte	0	1	2	3			

Protocol (BYTE)

Netzwerkprotokoll (1 = ICMP, 6 = TCP, 17 = UDP)

Handle (BYTE)

Handle-Nummer der Netzwerkverbindung

State (WORD)

0 = mit MAC- und IP-Adresse adressiertes Datenpaket, 1 = Ethernet- und IP-Broadcast

Destination Port (WORD)

Destination UDP- oder TCP-Port, muß bei TCP-Verbindungen nicht angegeben werden. Bei UDP-Verbindungen kann der Port angegeben werden, wenn wechselnde Zieladressen im Netzwerk verwendet werden. Wurde beim Öffnen eines Handles ein Zielport konfiguriert und wird immer dieser Port verwendet, so kann dieses Feld auch mit 0 beschrieben werden.

Destination IP-Address (DWORD, 32bit)

Destination IP-Adresse, muß bei TCP-Verbindungen nicht angegeben werden. Bei UDP-Verbindungen kann die IP-Adresse angegeben werden, wenn wechselnde Zieladressen im Netzwerk verwendet werden. Wurde beim Öffnen eines Handles eine Ziel-IP-Adresse konfiguriert und wird immer diese IP-Adresse verwendet, so kann dieses Feld auch mit 0 beschrieben werden.

DATA (Byte-Strom)

Applikationsdaten, die Länge des Bytestroms darf nicht größer als die im Com-Server konfigurierte MTU sein.

3.1 Übertragung von Daten über einen UDP- oder TCP-Socket

Zur Übermittlung von Daten mit dem UDP- oder TCP-Protokoll müssen dem Com-Server vor Beginn der Datenübertragung die Verbindungsparameter übergeben werden. Beim Öffnen eines TCP-Sockets mit TYPE 100 müssen alle Parameter angegeben werden. Beim Versenden von Applikationsdaten muß dann im Datenpaket (TYPE 110) nur das Feld *Handle* ausgefüllt werden.

Beim Öffnen eines UDP-Sockets muß zwingend der lokale Port des Com-Servers und das Protokoll angegeben werden. Ziel-IP-Adresse und Ziel-UDP-Port hingegen können variabel sein. Werden Ziel-IP-Adresse und -Port beim Öffnen des Handles übergeben, ist es ausreichend, beim Versenden von Applikationsdaten den Handle einzutragen. Der Com-Server fügt dann die beim Öffnen des Handles übergebene Zieladresse ein (siehe auch Kapitel 2.1).

Der Com-Server füllt beim Senden von Daten an das serielle Endgerät prinzipiell den Header komplett aus. Die Felder *Destination Port* und *IP-Adresse* enthalten dann die Adresse der Netzwerkstation, also den Absender.

1. Handle öffnen mit TYPE 100 (siehe Kapitel 2.1)

2. Übertragung und Empfang von Daten mit TYPE 110. Header des Datenpakets:

```

PROTOCOL:    6 = TCP, 17 = UDP
HANDLE:      Handle, den der Com-Server nach Öffnen des Handles zurückgibt
STATE:       0
DEST_PORT:   Portnummer der Netzwerkstation (oder 0*)
DEST_IP:     IP-Adresse der Netzwerkstation (oder 0.0.0.0*)

```

*) nur UDP: Enthalten diese Felder den Wert 0, trägt der Com-Server die Werte ein, die ihm beim Öffnen des Handles übergeben worden.

3. Handle schließen mit TYPE 101, wenn die Verbindung nicht mehr benötigt wird (siehe Kapitel 2.2). Es können bis zu acht Handles gleichzeitig geöffnet werden.

Sonderfall: Senden und Empfangen von IP-Broadcasts

IP-Broadcasts können auf einem geöffneten UDP-Handle empfangen und gesendet werden. Beim Senden muß das Feld *STATE* auf 1 gesetzt werden. Als Ziel-IP-Adresse wird die Broadcastadresse eingetragen. Der Com-Server erkennt anhand der Adresse, ob es sich um einen *Local* oder einen *Directed Broadcast* handelt. *Directed Broadcasts* werden an das konfigurierte Gateway weitergereicht.

Empfängt der Com-Server einen Broadcast, setzt er im Header des Datenpaketes (TYPE 110) das Feld *STATE* auf 1.

	IP-Adresse des Com-Servers	Subnet Mask des Com-Servers	Netzwerkadresse des Com-Servers	IP-Broadcast-Adresse im Datenpaket	Zielnetzwerk
Local Broadcast	172.16.233.201	255.255.255.0	172.16.233.0	172.16.233.255	172.16.233.0

Der Local Broadcast wird im Netzwerksegment des Com-Servers als Ethernet-Broadcast gesendet.

Directed Broadcast	172.16.233.201	255.255.255.0	172.16.233.0	172.16.253.255	172.16.253.0
--------------------	----------------	---------------	--------------	----------------	--------------

Der Directed Broadcast wird an das konfigurierte Gateway weitergegeben, welches das Paket in das Zielnetzwerk routet.

3.2 Senden und Empfangen von ICMP-Messages (Ping)

ICMP Echo Request und Response können direkt mit TYPE 110 übertragen werden. Es muß kein Handle dafür geöffnet werden. Für jedes Typ-110-Paket mit Protocol=1 erzeugt der Com-Server einen ICMP Echo Request nach RFC-792. Jedes Typ-110-Paket mit Protocol=1, das das Peripheriegerät vom Com-Server erhält, bedeutet, daß ein Echo Reply empfangen wurde. Die mit dem seriellen Paket übergebenen Daten entsprechen dabei dem benutzerdefinierbaren Datenanteil des ICMP-Paketes (ab Offset 8 in den ICMP-Daten) und sind also frei wählbar.

Das Peripheriegerät kann einen Request senden und den entsprechenden Response empfangen. Empfängt der Com-Server einen Request, wird dieser nicht weitergeleitet, sondern vom Com-Server selbst beantwortet.

```

PROTOCOL:      1 für ICMP
HANDLE:        0
STATE:         0
Destination Port:  0
Destination IP:  IP-Adresse der Netzwerkstation
    
```

4. Konfiguration und Statusanzeigen

4.1 Basis-Konfiguration der seriellen Schnittstelle

Mit diesem Paket werden die Basis-Parameter der seriellen Schnittstelle des Com-Servers konfiguriert. Dieser Konfigurationstyp erfordert eine besondere Behandlung, um die Fortführung der Kommunikation nach Einstellung der neuen Parameter zu garantieren.

0	1	2	4	5	6	8	
PKT- END	TYPE = 3	LEN=4		BAUD	BITS	CKSUM	PKT- END
		Low Byte	High Byte			Low Byte	High Byte

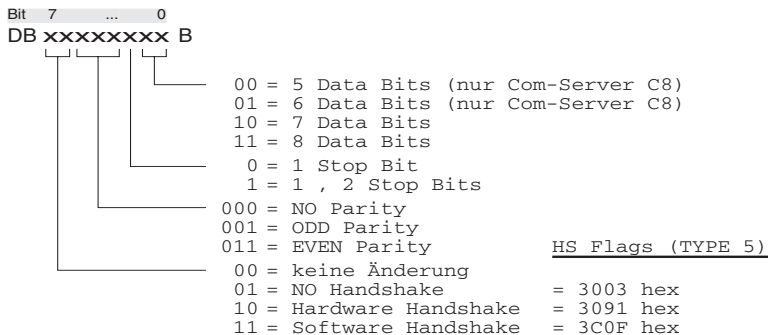
BAUD (BYTE)

In diesem Feld wird die Baudrate der seriellen Schnittstelle des Com-Servers konfiguriert. Tragen Sie folgende Werte für die möglichen Baudrates ein:

Baudrate	BAUD	Baudrate	BAUD	Baudrate	BAUD
230,4k	11	14400	20	600	8
172,8k	14	9600	3	300	9
115,2k	15	7200	4	150	10
57600	0	4800	5	75	12
38400	1	2400	6	50	13
19200	2	1200	7		
...	...	Baudrates nur in den Com-Server-Typen C8 vorhanden			
...	...	Baudrates nur in den Com-Server-Typen C3, C4, C5 vorhanden			

BITS (BYTE)

In diesem Feld werden die Parameter Data Bits, Stop Bits, Parity und Handshake-Mode der seriellen Schnittstelle des Com-Servers konfiguriert. Das BITS-Feld ist folgendermaßen untergliedert:



Achtung: Arbeitet der Com-Server mit dem Software-Handshake, gibt er vor dem Ändern der Konfiguration den Character XOFF aus. Arbeitet er danach mit dem Software-Handshake, sendet er nach Beendigung der Konfiguration den Character XON.

CKSUM (WORD, 16bit)

Die Berechnung einer Checksumme über das Paket ist erforderlich, um die Integrität der übermittelten Parameter zu garantieren. Die Checksumme ist das Einerkomplement der byteweisen Summe der Parameter TYP, LEN, BAUD und BITS.

Die folgende Assembleroutine realisiert den Algorithmus:

```

lea di, TYP      ;Adresse des Parameters TYP
mov cx, 5        ;add 5 Bytes
xor ax, ax       ;ax = 0
xor bx, bx       ;bx = 0, Clear CKSUM

lp_cksum:
mov al, [di]
add bx, ax       ;add next Byte, Carry in bh
inc di
loop lp_cksum

not bx           ;Einerkomplement
mov CKSUM, bx

```

Hat der Com-Server das Paket korrekt erhalten und interpretiert, sendet er auf der alten Konfiguration einen OK-Typ. Erst danach wird die neue Konfiguration aktiviert.

OK-Typ:

0	1	2	4
PKT- END	TYPE =0	LEN = 0	PKT- END

Lesen der Basis-Konfiguration

Der Com-Server antwortet mit dem Paket-Type 3 und setzt in die Felder BAUD und BITS seine aktuelle Basis-Konfiguration ein.

0	1	2	4
PKT- END	TYPE =2	LEN = 0	PKT- END

4.2 Zusatz-Konfiguration der seriellen Schnittstelle

Dieses Paket bietet die Möglichkeit, die serielle Schnittstelle möglichst genau an die Anforderungen anzupassen. Beispielsweise kann die Funktion jeder einzelnen Leitung festgelegt werden, die Zeichen für das Software-Handshake konfiguriert werden, u.s.w.

Wichtig: Die mit diesem Paket-Typ getroffenen Einstellungen werden bis auf die HS Flags nicht im EEPROM abgespeichert, d.h. nach einem Restart des Com-Servers gelten wieder die Default-Einstellungen.

0	1	2	4	5	6	8	10	12				
PKT- END	TYPE = 5	LEN=8		XON Char	XOFF Char	XON Limit		XOFF Limit		HS Flags		PKT- END
		Low Byte	High Byte			Low Byte	High Byte	Low Byte	High Byte	Low Byte	High Byte	

XON Char (BYTE) DEFAULT: 11hex

Festlegung eines Characters, der als XON verwendet wird (Software-Handshake)

XOFF Char (BYTE) DEFAULT: 13hex

Festlegung eines Characters, der als XOFF verwendet wird (Software-Handshake)

XON Limit (WORD) DEFAULT: 300

Überschreitet die Anzahl der freien Bytes im Ringbuffer den Wert von XON Limit, wird das Handshake-Stop-Signal zurückgenommen (Hardware Handshake) oder XON gesendet (Software Handshake).

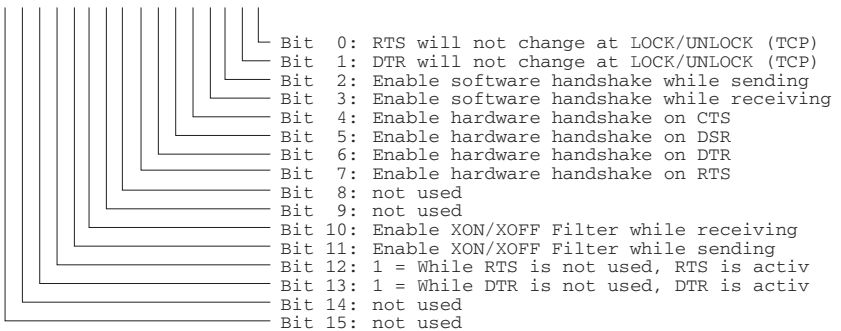
XOFF Limit (WORD) DEFAULT: 256

Unterschreitet die Anzahl der freien Bytes im Ringbuffer den Wert von XON Limit, wird das Handshake-Stop-Signal gesetzt (Hardware Handshake) oder XOFF gesendet (Software Handshake).

HS Flags (WORD) DEFAULT: 3003 hex (NO Handshake)
DEFAULT: 0091 hex (Hardware Handshake)
DEFAULT: 3C0F hex (Software Handshake)

Mit diesen Flags kann das Handshake konfiguriert werden. Die Defaultwerte werden bei Verwendung des Paket-Types 3 eingestellt.

Bit 15 8 7 0
 DB xxxxxxxx xxxxxxxx B



Lesen der Zusatz-Konfiguration

Der Com-Server antwortet mit dem Paket-Type 5 und setzt in die Felder XON Char, XOFF Char, XON Limit, XOFF Limit und HS Flags seine aktuelle Konfiguration ein.

0	1	2	4
PKT- END	TYPE =4	LEN = 0	PKT- END

4.3 Netzwerkkonfiguration des Com-Servers

Dieses Paket erlaubt die Konfiguration der Netzwerkparameter des Com-Servers.

0	1	2		4	5	6	7	8	9	10	11	12	13	14	15
PKT- END	TYPE = 7	LEN = 14		IP-Address Byte:				Subnet-Mask Byte:				Gateway-IP Byte:			
		Low Byte	High Byte	0	1	2	3	0	1	2	3	0	1	2	3
16		18													
MTU		PKT- END													
Low Byte	High Byte														

IP-Address (DWORD, 32bit)

IP-Adresse des Com-Servers

Subnet Mask (DWORD, 32bit)

Subnet Mask des IP-Netzwerkes, in dem der Com-Server angeschlossen ist

Gateway-IP (DWORD, 32bit)

IP-Adresse des Gateways, über das der Com-Server in andere IP-Netze routet

MTU - Maximum Transfer Unit (WORD, 16bit) DEFAULT: 512

Dieser Wert bestimmt die maximale Paketgröße eines TCP/IP-Paketes. Er bezieht sich auf die Anzahl der Bytes (ohne Header), die in einem Paket übertragen werden können. Je kleiner die MTU gewählt wird, desto mehr Netzwerkbuffer stehen insgesamt im Com-Server zur Verfügung. Der wählbare Bereich beginnt bei 512 und endet bei 1024 Bytes. Die Werte sind in Schritten von 128 Bytes einstellbar (automatische Korrektor).

Achtung: Da es sich hier um Basis-Parameter des Systems handelt, führt der Com-Server nach Abspeichern der Werte einen Restart durch, um die Konfiguration zu laden.

Lesen der Netzwerk-Konfiguration

Der Com-Server antwortet mit dem Paket-Type 7 und trägt in die Felder IP-Address, Subnet Mask und Gateway-IP die aktuellen Adressen ein.

0	1	2	4
PKT- END	TYPE =6	LEN = 0	PKT- END

4.4 Statusanzeige des Com-Servers

Dieses Paket enthält den Status des Com-Servers und kann nur vom Com-Server selbst verwendet werden.

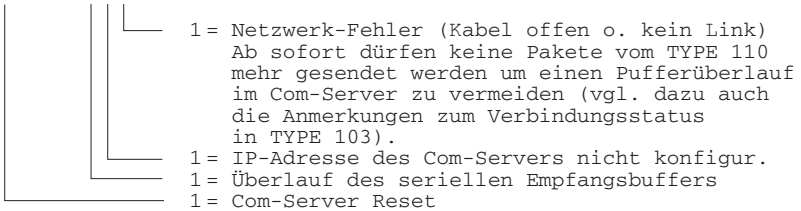
0	1	2	4	5	6	
PKT- END	TYPE = 9	LEN = 2		SYS- TEM	reser- ved	PKT- END
		Low Byte	High Byte			

SYSTEM (BYTE)

In diesem Feld werden bitweise System-Fehler des Com-Servers kodiert. Hat SYSTEM den Wert 0, ist kein Fehler aufgetreten.

Bit 7 ... 0

DB x0000xxx B



Lesen der Statusanzeige

Der Com-Server antwortet mit dem Paket-Type 9 und trägt den aktuellen Systemstatus ein.

0	1	2	4
PKT- END	TYPE =8	LEN = 0	PKT- END

4.5 Das Com-Server-Password

Mit dieser Struktur kann das Passwort des Com-Servers für die Telnet-Konfiguration geschrieben werden. Das Passwort wird am Terminal als hexadecimaler Wert eingegeben (siehe auch Beispiele). Daraus ergibt sich eine Beschränkung der verwendbaren Zeichen auf die Ziffern 0bis 9 und die Buchstaben a bis f.

0	1	2	4					8
PKT- END	TYPE = 11	LENGTH = 4		PASSWORD				PKT- END
		Low Byte	High Byte	Byte 0	Byte 1	Byte 2	Byte 3	

Beispiel 1:

password = 0x12345678,
Eingabe am Terminal:
"1234567"

0	1	2	4
0x78	0x56	0x34	0x12
0	1	2	4
0x00	0x20	0xba	0xab

Beispiel 2:

password = xABBA2000,
Eingabe am Terminal:
"ABBA2000"

Lesen des Com-Server-Passwords

Der Com-Server antwortet mit dem Paket-Type 11 und trägt das aktuelle Passwort ein.

0	1	2	4	
PKT- END	TYPE = 10	LENGTH = 0		PKT- END
		Low Byte	High Byte	

4.6 Die Com-Server Software Revision

Diese Struktur verwendet nur der Com-Server. Der Wert SW-REVISION wird nicht als ASCII übertragen. Der Wert Low Byte=12 High Byte=3 entspricht der Version 3.12.

0	1	2	4	6	
PKT- END	TYPE = 13	LENGTH = 2		SW - REVISION	PKT- END
		Low Byte	High Byte	Low Byte	High Byte

Lesen der Com-Server Software Revision

Der Com-Server antwortet mit dem Paket-Type 13 und trägt die aktuelle Software Revision ein.

0	1	2	4	
PKT- END	TYPE = 12	LENGTH = 0		PKT- END
		Low Byte	High Byte	

4.7 Das Com-Server System Password (nur C8 Highspeed)

Das System Password ist ein nullterminierter String mit maximal 31 Zeichen. Es dürfen außer der Null alle Zeichen verwendet werden.

0	1	2	4	6	
PKT- END	TYPE = 15	LENGTH = strlen		System Password	PKT- END
		Low Byte	High Byte	1...31 character	0
Beispiel:		6	0	'W' '&' 'T' '' '1'	0

Lesen des Com-Server System Passwords (nur C8 Highspeed)

Der Com-Server antwortet mit dem Paket-Type 15 und trägt das aktuelle System Password ein.

0	1	2	4	
PKT- END	TYPE = 14	LENGTH = 0		PKT- END
		Low Byte	High Byte	

4.8 Der Com-Server System Name (nur C8 Highspeed)

Der System Name ist ein nullterminierter String mit maximal 31 Zeichen. Es dürfen außer der Null alle Zeichen verwendet werden.

0	1	2	4							6
PKT- END	TYPE = 17	LENGTH = strlen		System Name						PKT- END
		Low Byte	High Byte	1...31 character						0
Beispiel:		6	0	'W'	'&'	'T'	' '	'1'	0	

Lesen des Com-Server System Namens (nur C8 Highspeed)

Der Com-Server antwortet mit dem Paket-Type 17 und trägt den aktuellen System Namen ein.

0	1	2	4	
PKT- END	TYPE = 16	LENGTH = 0		PKT- END
		Low Byte	High Byte	

4.9 Die MAC-Adresse des Com-Servers

Die MAC-Adresse ist eine 6-stellige Netzwerkadresse, die vom Hersteller vergeben wird und nicht geändert werden kann.

0	1	2	4							6
PKT- END	TYPE = 19	LENGTH = 6		MAC-Adresse						PKT- END
		Low Byte	High Byte	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	
Beispiel:		6	0	00h	C0h	3Dh	01h	4Eh	0Fh	

Lesen der MAC-Adresse des Com-Servers

Der Com-Server antwortet mit dem Paket-Type 19 und trägt die MAC-Adresse ein.

0	1	2	4	
PKT- END	TYPE = 18	LENGTH = 0		PKT- END
		Low Byte	High Byte	

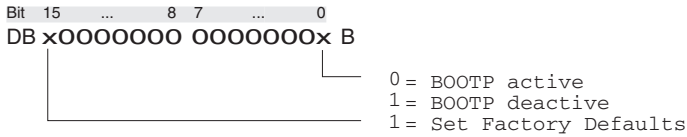
4.10 Die Zusatz-Konfiguration des Com-Servers

Dieser Type ist vorgesehen zur Konfiguration von Randparametern.

0	1	2	4	6		
PKT- END	TYPE = 21	LENGTH 12		FLAGS	unused	PKT- END
		Low Byte	High Byte	Low Byte	High Byte	Bytes 2...11

FLAGS (WORD)

In diesem Feld werden bitweise einzelne Flags des Com-Servers kodiert, die geschrieben und gelesen werden können.



Lesen der Zusatz-Konfiguration des Com-Servers

Der Com-Server antwortet mit dem Paket-Type 19 und trägt die MAC-Adresse ein.

0	1	2	4	
PKT- END	TYPE = 20	LENGTH = 0		PKT- END
		Low Byte	High Byte	

4.11 Die Handle Information

Diese Struktur verwendet nur der Com-Server. Der Wert MAX Handles gibt die Anzahl der maximal verfügbaren Handle an. Der Wert FREE Handles gibt die Anzahl der momentan verfügbaren Handles an.

0	1	2	4	6	8			
PKT- END	TYPE = 23	LENGTH = 4		MAX Handles		FREE Handles		PKT- END
		Low Byte	High Byte	Low Byte	High Byte	Low Byte	High Byte	

Lesen der Handle Information

Der Com-Server antwortet mit dem Paket-Type 23 und trägt die verfügbare Handleanzahl ein.

0	1	2	4	
PKT- END	TYPE = 22	LENGTH = 0		PKT- END
		Low Byte	High Byte	

4.12 Auslösen eines Software-Resets am Com-Server

Der Com-Server führt sofort einen Software-Reset durch. Vor Senden dieses Pakettyes sollten unbedingt alle Handle geschlossen werden.

0	1	2	4	
PKT- END	TYPE = 253	LENGTH = 0		PKT- END
		Low Byte	High Byte	

5. Das Zeichendopplungsverfahren

Das Zeichendopplungsverfahren ermöglicht eine sichere Paketrahmenerkennung bei der seriellen Übertragung binärer Daten, ohne im Nutzdatenstrom Zeichen ausschließen zu müssen.

Das PKT-END Zeichen markiert Paketanfang und -ende, d.h. es darf nur als Rahmenzeichen auftreten. Das erste Zeichen nach einem PKT-END, das kein PKT-END ist, bezeichnet den Paketbeginn (hier das TYPE-Feld).

Definiert werden die folgenden 4 Zeichen:

```
#define PKT-END      C0 hex    // Paketanfang- und -endezeichen
#define ESCAPE      DB hex    // ESCAPE Sequence
#define PKT_ESC     DC hex    // Zeichenersetzung für PKT_END
#define ESC_ESC     DD hex    // Zeichenersetzung für ESCAPE
```

Beim Versenden der Daten muß der Datenstrom auf die Zeichen PKT-END und ESCAPE untersucht werden. Taucht eines der beiden Zeichen im Nutzdatenstrom auf, wird zunächst der Character ESCAPE gesendet, gefolgt von dem entsprechenden Ersatzzeichen (für PKT-END das Zeichen PKT-ESC und für ESCAPE das Zeichen ESC_ESC).

```
PKT-END  → ESCAPE + PKT_ESC    ( C0h → DBh DCh )
ESCAPE   → ESCAPE + ESC_ESC    ( DBh → DBh DDh )
```

Beim Empfang der Daten werden die Zeichen analog dazu wieder rückgewandelt. Wird das Zeichen ESCAPE empfangen, bedeutet dies, daß das folgende Zeichen ersetzt werden muß.

```
ESCAPE + PKT_ESC  → PKT-END    ( DBh DCh → C0h )
ESCAPE + ESC_ESC  → ESCAPE     ( DBh DDh → DBh )
```

Anwendung des Zeichendopplungsverfahrens

Beim Versenden der Daten wird die Zeichendopplung erst vorgenommen, wenn alle Header (auch das LEN-Feld) ausgefüllt sind. Nach dem Doppeln der Zeichen PKT_END und ESCAPE darf an dem zu versendenden String nichts mehr geändert werden.

Bsp.: Der Paket-Typ 3 (Basis-Konfiguration der seriellen Schnittstelle) wird versendet.

TYPE = 03 hex, LEN = 0002 hex, BAUD = 03 hex, BITS = C0 hex

- 1.) Senden des PKT_END-Characters (Paketanfang)
- 2.) Anwendung der Zeichendopplung auf alle zu versendenden Zeichen
- 3.) Senden des PKT_END-Characters (Paketende)

Der gesendete String sieht für das Beispiel folgendermaßen aus (hexadezimale Darstellung):

C0h 03h 02h 00h 03h DBh DCh C0h

Beim Empfang ist analog dazu vorzugehen:

- 1.) Empfang des PKT_END-Characters (Paketanfang) --> Zeichen verwerfen
- 2.) Empfang der Daten und Dekodierung aller Zeichen, die auf das Zeichen ESCAPE folgen bis
- 3.) wieder das Zeichen PKT_END empfangen wird (Paketende) --> Zeichen verwerfen

6. Beispiele in hexadezimaler Darstellung

OPEN HANDLE REQUEST (UDP, Priorität: 0, Handle: 255, Dest.Port:2000, IP: 172.16.233.1)

C0	64	0B	00	11	00	FF	40	1F	D0	07	01	E9	10	AC	C0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

OPEN HANDLE REQUEST (TCP-Client, Priorität: 0, Handle: 255, Dest.Port:2000, IP: 172.16.233.1)

C0	64	0B	00	06	00	FF	40	1F	D0	07	01	E9	10	AC	C0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

OPEN HANDLE REQUEST (TCP-Server, Priorität: 0, Handle: 200, Dest.Port: 0, IP: 0.0.0.0)

C0	64	0B	00	06	00	C8	40	1F	00	00	00	00	00	00	C0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

OPEN HANDLE RESPONSE (UDP, Priorität: 0, Handle: 0, Dest.Port: 2000, IP: 172.16.233.1)

C0	64	0B	00	11	00	00	40	1F	D0	07	01	E9	10	AC	C0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

DATA TYPE ohne Coding (UDP, Handle: 0, State: 0, Port: 2000, IP: 172.16.233.1, Data: DB DD)

C0	6E	0C	00	11	00	0	0	D0	07	01	E9	10	AC	DB	DD	C0
----	----	----	----	----	----	---	---	----	----	----	----	----	----	----	----	----

DATA TYPE mit Coding (UDP, Handle: 0, State: 0, Port: 2000, IP: 172.16.233.1, Data: DB DD)

C0	6E	0C	00	11	00	0	0	D0	07	01	E9	10	AC	DB	DD	DD	C0
----	----	----	----	----	----	---	---	----	----	----	----	----	----	----	----	----	----

CLOSE HANDLE (Handle: 0)

C0	65	01	00	00	00	C0
----	----	----	----	----	----	----

BASE_CONF TYPE (19200, N, 8, 1, NONE)

C0	03	00	00	02	43	B7	FF	C0
----	----	----	----	----	----	----	----	----

BASE_CONF REQUEST

C0	02	00	00	C0
----	----	----	----	----

CONF_PLUS TYPE

C0	05	08	00	11	13	90	01	2C	01	91	30	C0
----	----	----	----	----	----	----	----	----	----	----	----	----

CONF_PLUS REQUEST

C0	04	00	00	C0
----	----	----	----	----

NET_CONF TYPE

C0	07	0E	00	05	E7	10	AC	80	FF	FF	FF	64	E9	10	AC	00	04	C0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

NET_CONF REQUEST

C0	06	00	00	C0
----	----	----	----	----

STATE TYPE

C0	09	02	00	00	00	C0
----	----	----	----	----	----	----

STATE REQUEST

C0	08	00	00	C0
----	----	----	----	----

PASSWORD TYPE

C0	0B	04	00	78	56	34	12	C0
----	----	----	----	----	----	----	----	----

Password = "12345678"

PASSWORD REQUEST TYPE

C0	0A	00	00	C0
----	----	----	----	----

SOFTWARE REVISION TYPE

C0	0D	02	00	0C	03	C0
----	----	----	----	----	----	----

Software Revision = 3.12

SOFTWARE REVISION REQUEST TYPE

C0	0C	00	00	C0
----	----	----	----	----

Änderungen in den Versionsständen

1.0 nach 2.0:

- Kapitel 1.3: neue TYPE's:
10/11: Lesen und Schreiben des Com-Server-Passworts,
12/13: Lesen und Übermitteln der Software Revision des Com-Servers,
100/101: Öffnen und Schließen von Handles für die Datenübertragung
102/103: Lesen und Übermitteln des Verbindungsstatus eines Handles
110: TYPE zum Übertragen der Nutzdaten, TYPE 1 darf nicht mehr verwendet werden
- Kapitel 2: vor dem Senden und Empfangen von Daten muß ein Handle geöffnet werden
Es können TCP- und UDP-Handles geöffnet werden.
- Kapitel 4.4: Statusanzeige des Com-Servers (TYPE 9)
Der Wert Connect wird nicht mehr mit diesem TYPE übermittelt, sondern mit TYPE 103
(Verbindungsstatus eines Handles, Kapitel 2.3)
- Der Com-Server sendet TYPE 9 (Statusanzeige des Com-Servers) und TYPE 103
(Verbindungsstatus eines Handles) selbständig, wenn ein Ereignis eintritt (z.B. kein
Netzwerk mehr, keine Antwort beim Auflösen der Ziel-IP-Adresse, ...)

2.0 nach 2.1:

- Kapitel 2.1: TYPE 100 wurde um einen Character vorbereitend erweitert. Das Feld
"Priority" wird aber im Moment noch nicht genutzt.

2.1 nach 2.2:

- Kapitel 2.1: TYPE 100 ermöglicht jetzt auch das Öffnen eines Handles für eine TCP-
Server-Applikation, d.h. der Com-Server akzeptiert TCP-Verbindungen auf dem angege-
benen lokalen Port.
- Kapitel 2.3: exaktere Beschreibung des Feldes CONNECT in TYPE 103
- Kapitel 4.7: Einführung von TYPE 253 zum Auslösen eines Hardware-Resets des Com-
Servers

2.2 nach 2.3:

- Kapitel 2.3: Einführung des Wertes "packet count" im TYPE 103
- Kapitel 4.1: Aktualisierung und Erweiterung der Werte der Felder BAUD und BITS für die
Com-Server C8 Highspeed
- Kapitel 4.8 bis 4.11: Einführung der TYPES 14-21 zum Schreiben und Lesen des System
Passwords, des System Namen, der Zusatz-Konfiguration und zum Lesen der MAC-
Adresse des Com-Servers

2.3 nach 2.4:

- Kapitel 4.10: Einführung des Bits 15 im Feld FLAGS: set Factory Defaults und Reset

2.4 nach 2.5:

- Kapitel 4.11: Einführung der TYPES 22 und 23 zum Lesen der Handle-Information

2.5 nach 2.6:

- Kapitel 2.3: Einführung des TYPES 104 zum Killen einer TCP-Verbindung und Schließen des Handles

Irrtum und Änderungen vorbehalten
Wiesemann & Theis GmbH