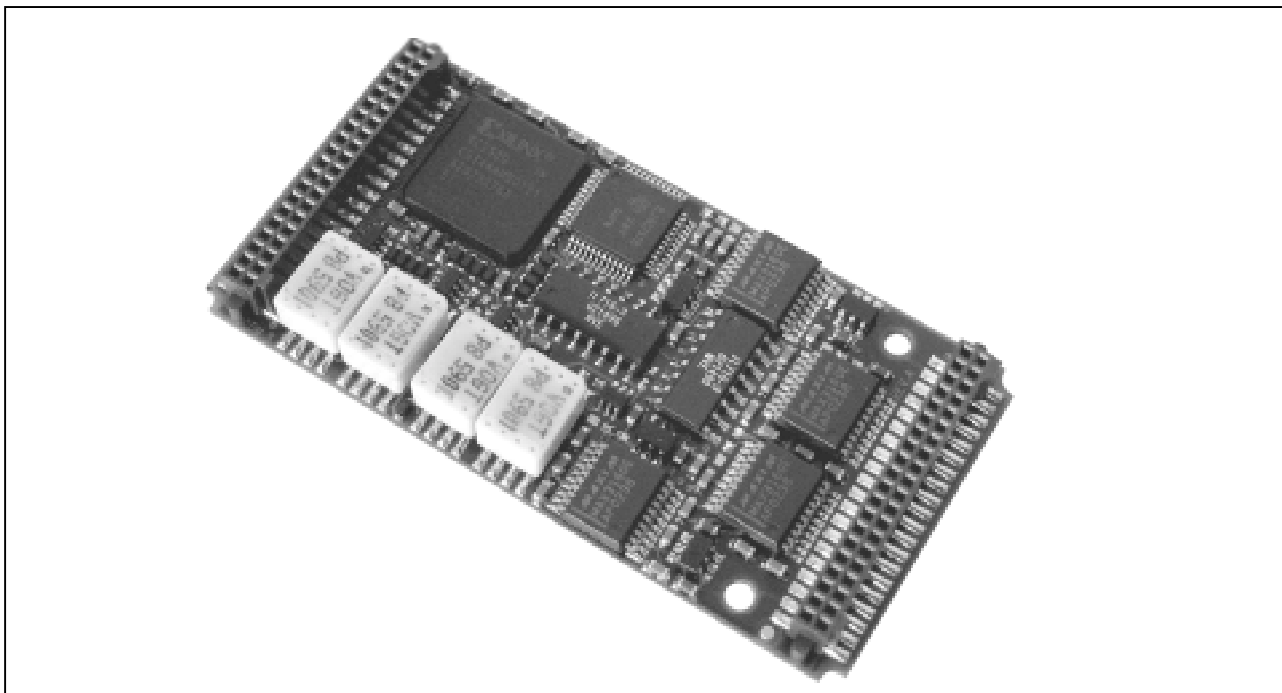


X-COM-8i

8 serielle Schnittstellen
(8 x RS-232 oder 8 x RS-4xx)



10.11. X-COM-8i

Inhaltsverzeichnis

10.11.	X-COM-8i.....	10-159
10.11.1.	Beschreibung	10-160
10.12.	Modul-Device-Treiber	10-161
10.12.1.1.	Installation	10-161
10.12.1.2.	Kanaleigenschaftsstruktur CPS_XCOM8	10-161
10.12.1.3.	Funktionsweise	10-161
10.12.1.4.	Serielle Schnittstellen	10-163
10.12.2.	Anschlusspins des Moduls (bezogen auf den Modul-Stecker A) ...	10-159
10.12.3.	Besondere Eigenschaften.....	10-160

10.11.1. Beschreibung

Das Modul ist in 2 Versionen verfügbar:

Typ	Sub- typ	Modul-Typ	Serielle Schnittstellen	Bemerkung
88	0	X-COM-8i/232	8	8x RS-232
88	1	X-COM-8i/4xx	8	8x RS-422 bzw. RS-485

Die 8 seriellen Schnittstellen sind PC-kompatibel. Das Modul ist in einer Variante mit 8 RS-232 Schnittstellen und einer Variante mit 8 RS-4xx Schnittstellen (per Software zwischen RS-422 und RS-485 umschaltbar) lieferbar. Jede Schnittstelle verfügt über 64 Byte große Empfangs- bzw. Sendepuffer. Die max. Datenrate beträgt 3Mb/s (RS-485) bzw. 1Mb/s (RS-232);

10.12. Modul-Device-Treiber

10.12.1.1. Installation

Der Modul-Device-Treiber für OsX hat die Programmnummer 8058h und den Dateinamen mxcom8.exe. Der Modul-Device-Treiber für Windows hat den Namen mxcom8.sys. Die Installation aus einem PC-Programm (z.B. für Steckplatz 1, Layer 0):

Error = max_load_mdd (hModul, 1, 0, 0, 0x8058, NULL, &hMDD);

Befehl in einer INS-Datei (z.B. für Steckplatz 1, Layer 0):

MAXLOADMDD slot=1 layer=0 progno=8058

10.12.1.2. Kanaleigenschaftsstruktur CPS_XCOM8

Die CPS für das Modul hat den Namen CPS_XCOM8.

10.12.1.3. Funktionsweise

Der MDD bietet eine Zwischenpufferung der Sende- und Empfangdaten.

Senden

Die Daten, die dem MDD zum Senden übergeben werden, werden zunächst im MDD zwischengespeichert. Der MDD reagiert auf den Interrupt, mit dem das Modul signalisiert, dass im Sende-FIFO-Puffer des Moduls Platz frei ist. Der MDD schreibt dann die zwischengepufferten Daten in das Modul, von wo aus sie versendet werden. Auf diese Weise kann das Anwenderprogramm beliebig Sendedaten nachliefern, ohne dass es sich um die Abwicklung des Sendevorgangs kümmern muss. Die Beschränkung liegt in der Größe des MDD-internen Zwischenpuffers, die beim Öffnen des Kanals festgelegt wird.

Empfangen

Der MDD reagiert auf den Interrupt, mit dem das Modul signalisiert, dass Daten im Empfangs-FIFO der jeweiligen Schnittstelle vorliegen. Der MDD entnimmt daraufhin die Daten aus dem FIFO und puffert sie intern zwischen. Das Anwenderpro-

gramm kann zu jedem Zeitpunkt Daten abholen. Falls gerade keine vorliegen, wird dies gemeldet. Die Größe des MDD-internen Zwischenpuffers wird beim Öffnen des Kanals festgelegt.

Handshake

Das Modul unterstützt 2 Protokolle, mit denen ein Überlauf des Empfangspuffers verhindert werden kann:

- **RTS-CTS Handshake:** Die RTS- und CTS-Signale von Sender und Empfänger sind gekreuzt. Der Sender signalisiert über sein RTS-Signal, dass er Daten zum Senden vorliegen hat. Wenn der Empfänger bereit für den Empfang ist, muss er dies wiederum über sein RTS-Signal melden. Daraufhin beginnt der Sender mit dem Versenden der Daten. Wenn sich eine bestimmte (in *ulXoffLim* einstellbare) Anzahl von Zeichen im Empfangspuffer befinden, setzt der Empfänger RTS =0. Der Sender muss auf diesen Signalwechsel an seinem CTS Eingang mit der Unterbrechung des Sendens reagieren. Sobald der Füllstand des Empfangspuffers im Empfänger eine bestimmte (in *ulXonLim* einstellbare) Schwelle wieder unterschreitet (dadurch dass die Daten vom Anwenderprogramm abgeholt wurden), setzt der Empfänger RTS wieder =1. Daraufhin kann der Sender wieder Daten senden. Hat der Sender keine weiteren Sendedaten vorliegen, beendet er die Sendung durch Zurücksetzen seines RTS-Signals.
- **XON/XOFF Handshake:** Dieses Protokoll arbeitet ohne Steuerleitungen. Statt dessen wird ein Zeichen als XOFF-Zeichen (üblicherweise 13h) und eines als XON-Zeichen (üblicherweise 11h) auf Sender- und Empfängerseite vereinbart. Der Empfänger kann den Sender durch Senden von XOFF anhalten. Durch Senden von XON wird dem Sender mitgeteilt, dass er Daten senden kann. Die vereinbarten Zeichen dürfen in den Sendedaten nicht vorkommen. Sie werden auch nicht in den Empfangspuffer eingetragen.

10.12.1.4. Serielle Schnittstellen

Ein Zugriff auf eine der vier seriellen Schnittstellen kann mit folgender CPS erfolgen:

Strukturelement	Werte	Bedeutung
<i>.usDevice</i>	<i>DEVICE_COM</i>	Serielle Schnittstelle
<i>.usIndexFirst</i>	0 ... 7	Nummer der Schnittstelle
<i>.usIndexLast</i>	<i>.usIndexFirst</i>	Nummer der Schnittstelle
<i>.usType</i>	<i>COM_RS232</i> <i>COM_RS422</i> <i>COM_RS485</i>	RS232 Schnittstelle (X-COM-8i/232) RS422 Schnittstelle (X-COM-8i/4xx) RS485 Schnittstelle (X-COM-8i/4xx)
<i>.ulFlags</i>	<i>_CP_EXCLUSIVE</i> <i>_COM_REPLACE_ERR_CHAR</i>	Dieses Flag muss gesetzt sein! Der Zugriff auf eine Schnittstelle erfolgt immer exklusiv. Ein empfangenes Zeichen, das einen Fehler aufweist, wird durch das <i>cErrorChar</i> Zeichen ersetzt.
<i>.usReadMode</i>	<i>IO_MODE_RAM</i>	Die Empfangs-Daten werden aus einem Empfangspuffer des MDDs entnommen. Der MDD trägt die empfangenen Daten selbstständig in diesen Puffer ein.
<i>.usWriteMode</i>	<i>IO_MODE_RAM</i>	Die Sende-Daten werden in einen internen Sendepuffer des MDDs geschrieben. Der Puffer wird automatisch vom MDD geleert.
<i>.usFlowControl</i>	<i>COM_NO_FLOW</i> <i>COM_XONXOFF_FLOW</i> <i>COM_RTSCTS_FLOW</i>	Hier wird festgelegt, welches Protokoll verwendet werden soll. Einer der 3 folgenden Werte muss gesetzt werden: Es wird kein Protokoll verwendet. Das Software Protokoll XON/XOFF wird zum Senden und Empfangen benutzt. Die Steuerleitungen RTS und CTS werden für ein Handshake benutzt.

Strukturelement	Werte	Bedeutung
<i>.ulBaudRate</i>	46 ... 3000000	Baudrate (in Baud) Alle Standardwerte sind einstellbar. ¹
<i>.usByteSize</i>	<i>COM_5_BITS</i> <i>COM_6_BITS</i> <i>COM_7_BITS</i> <i>COM_8_BITS</i>	Anzahl der Bits pro Zeichen: 5 Bits 6 Bits 7 Bits 8 Bits
<i>.usStopBits</i>	<i>COM_ONE_STOPBIT</i> <i>COM_ONES5_STOPBITS</i> <i>COM_TWO_STOPBITS</i>	Anzahl der Stopp-Bits: 1 Stoppbit 1,5 Stoppbits 2 Stoppbits
<i>.usParity</i>	<i>COM_NO_PARITY</i> <i>COM_EVEN_PARITY</i> <i>COM_ODD_PARITY</i> <i>COM_MARK_PARITY</i> <i>COM_SPACE_PARITY</i>	Paritäts-Bit: kein Paritätsbit gerades Parität ungerade Parität Paritätsbit = Mark Paritätsbit = Space
<i>.ulXonLim</i>	1 ... <i>ulXoffLim</i>	Erreicht der Empfangspuffer diesen Füllstand, wird die Gegenstelle mittels des gewählten Protokolls veranlasst, Zeichen zu senden.
<i>.ulXoffLim</i>	1 ... <i>ulRcvBuffer</i>	Erreicht der Empfangspuffer diesen Füllstand, wird die Gegenstelle mittels des gewählten Protokolls veranlasst, keine Zeichen mehr zu senden.
<i>.cXonChar</i>		Das XON Zeichen für das XON/XOFF Software Protokoll.
<i>.cXoffChar</i>		Das XOFF Zeichen für das XON/XOFF Software Protokoll.
<i>.cErrorChar</i>		Fehlerzeichen. Dieses Zeichen wird benutzt, um fehlerhafte Zeichen im Empfangsbuffer zu überschreiben.
<i>.ulTmtBuffer</i>	0 ... 2024	Größe des Sendepuffers
<i>.ulRcvBuffer</i>	0 ... 10240	Größe des Empfangspuffers
<i>.ulTimeout</i>	0	Reserviert
<i>.ulCallBackEvents</i>	<i>_XCOM8_EVENT_RCV_BUFFER</i>	Die Füllgrenze im Software-

¹ Die seriellen Schnittstellen Controller werden mit einem Takt von 48 MHz versorgt. Die möglichen Baudraten berechnen sich nach folgender Formel: Baudrate = 48000000 : x mit x=1..65535. Wird eine Baudrate angegeben, die nicht möglich ist, stellt der MDD den nächstgelegenen möglichen Wert ein. Der tatsächlich eingestellte Wert kann in diesem Fall mit einem Sonderdienst abgefragt werden.

Strukturelement	Werte	Bedeutung
		Empfangspuffer wurde überschritten. Standardmäßig ist diese Grenze auf 50% des Empfangspuffers eingestellt. Um diese Grenze zu ändern, steht ein Sonderdienst zur Verfügung.
	<code>_XCOM8_EVENT_RCV_BUFFER_OVERFLOW</code>	Im Software-Empfangspuffer ist ein Überlauf aufgetreten.
	<code>_XCOM8_EVENT_TMT_BUFFER</code>	Die Füllgrenze des Software-Sendepuffers wurde unterschritten. Standardmäßig ist diese Grenze auf 50% des Sendepuffers eingestellt. Um diese Grenze zu ändern, steht ein Sonderdienst zur Verfügung.
	<code>_XCOM8_EVENT_RCV_ERROR</code>	Beim Empfang eines Zeichens wurde ein Fehler (Frame-Fehler, Parity-Fehler oder Hardware-Puffer-Überlauf) festgestellt.
	<code>_XCOM8_EVENT_BREAK</code>	Das BREAK Signal wurde empfangen.
	<code>_XCOM8_EVENT_RCV_CHAR</code>	Ein neues Zeichen wurde empfangen.

Anmerkungen

Nach dem Öffnen eines RS-485 Kanals ist dieser standardmäßig auf Empfang geschaltet. Zum Umschalten stehen Sonderdienste zur Verfügung.

Eingabe- und Ausgabedienst

Der Datentyp des Kanals ist `DATA_UCHAR`.

- **max_write_channel_block**
- **max_read_channel_block**

Ist bei der Übertragung ein Fehler aufgetreten, liefert der Dienst einen Fehler zurück (`XCOM8_UART_ERROR`). Der Fehler **muss** daraufhin mit einem Sonderdienst (**max_channel_info**, Infotyp `INFO_ERROR`) zurückgesetzt werden.

Sind bei einem Lesevorgang nicht die Anzahl der gewünschten Zeichen vorhanden, werden die im Puffer vorhandenen Zeichen zurückgegeben. Der Parameter `ulSize` enthält nach dem Aufruf die Anzahl der zurückgegebenen Zeichen.

Ist bei einem Schreibzugriff im Puffer nicht genügend Platz vorhanden, werden keine Daten übernommen und der Dienst liefert den Fehler `ERR_BUFFER_FULL` zurück.

Callback-Funktion

Über das Strukturelement *ulCallbackEvents* kann festgelegt werden, über welches Ereignis man informiert werden möchte. Die Callback-Funktion bekommt 4 Byte Nutzdaten übergeben, die angeben, welche Ereignisse aufgetreten sind. Dazu werden die Konstanten, die in *ulCallbackEvents* anzugegeben sind, zurückgegeben.

Sonderdienste

- **max_channel_info**, Infotyp INFO_ERROR: um im Falle eines Fehlers den Fehler zurückzusetzen und eine genauere Fehlerursache zu ermitteln (zuvor **max_clear_error** aufrufen!). Es werden 4 Byte Daten zurückgeliefert, die aus einer Oder-Verknüpfung der folgenden Werte bestehen:
 - **_XCOM8_ERROR_OVERRUN** : Es wurde ein Überlauf im Hardware-Empfangspuffer festgestellt.
 - **_XCOM8_ERROR_FRAME** : Es wurde ein Frame-Fehler in dem Empfangszeichen festgestellt.
 - **_XCOM8_ERROR_BREAK** : Es wurde ein BREAK-Signal festgestellt.
 - **_XCOM8_ERROR_RCV_BUFFER_OVERFLOW** : Der Software-Empfangspuffer im Treiber ist übergelaufen.
 - **_XCOM8_ERROR_PARITY** : Es wurde ein Parity-Fehler in dem Empfangszeichen festgestellt.
 - **_XCOM8_ERROR_TMT** : Beim Senden eines Zeichens wurde ein Fehler festgestellt.
- **max_channel_info**, Infotyp INFO_XCOM8_RCV_BUFFER und INFO_XCOM8_TMT_BUFFER: liefert die Anzahl der Zeichen im Empfangs- bzw. Sendepuffer zurück. Die Anzahl der Zeichen wird als ULONG-Wert zurückgeliefert.
- **max_channel_control**, Steuerbefehle CMD_DIR_OUTPUT, CMD_DIR_INPUT: eine serielle RS-485 Schnittstelle auf Senden bzw. Empfang umschalten. CMD_DIR_AUTO_OUTPUT: die RS-485 Schnittstelle wird automatisch auf Senden beziehungsweise Empfang umgeschaltet. Die Schnittstelle wird solange auf Senden eingestellt, wie Sendedaten zur Verfügung stehen. Danach schaltet Sie automatisch auf Empfang um. Der Funktion werden keine Daten übergeben.
- **max_channel_control**, Steuerbefehle CMD_XCOM8_CLEAR_RCV_BUFFER und CMD_XCOM8_CLEAR_TMT_BUFFER: Empfangs- bzw. Sendepuffer des MDDs löschen. Es werden keine Daten übergeben.
- **max_channel_info**, Infotyp INFO_XCOM8_BAUDRATE: die tatsächlich eingestellte Baudrate abfragen. Der Wert wird als ULONG-Wert zurückgegeben.

- **max_channel_control**, Steuerbefehl CTRL_XCOM8_MODEM_LINES: Steuerleitungen manuell schalten.

In dem zu übergebenen ULONG-Wert bestimmen die Flags `_XCOM8_RTS_0` und `_XCOM8_RTS_1` den Zustand des RTS-Signals (nur für RS-232, steht nicht zur Verfügung, wenn `.usFlowControl = COM_RTSCTS_FLOW` gesetzt ist).

- **max_channel_info**, Infotyp INFO_XCOM8_MODEM_LINES: Status der Steuerleitung CTS abfragen. Diese stehen nur für RS-232 Schnittstellen zur Verfügung. Es wird ein ULONG-Wert zurückgeliefert.
- **max_channel_control**, Steuerbefehl CTRL_XCOM8_EVENT_RCV_BUFFER und CTRL_XCOM8_EVENT_TMT_BUFFER: Die Füllgrenze des Empfangs- / Sendepuffers, bei der die Callback-Funktion aufgerufen werden soll wird eingestellt. In dem zu übergebenden ULONG-Wert muss eingetragen werden, auf wieviel Zeichen die Schwelle eingestellt werden soll

10.12.2. Anschlusspins des Moduls (bezogen auf den Modul-Stecker A)

Pin	Kanal	Signal		
		RS-232	RS-422	RS-485
1	0	RCV	RCV+	Do not connect
2	0	RTS	TMT+	RCV+/TMT+
3	0	TMT	TMT-	RCV-/TMT-
4	0	CTS	RCV-	Do not connect
5	0	GND	-	-
6	1	RCV	RCV+	Do not connect
7	1	RTS	TMT+	RCV+/TMT+
8	1	TMT	TMT-	RCV-/TMT-
9	1	CTS	RCV-	Do not connect
10	1	GND	-	-
11	2	RCV	RCV+	Do not connect
12	2	RTS	TMT+	RCV+/TMT+
13	2	TMT	TMT-	RCV-/TMT-
14	2	CTS	RCV-	Do not connect
15	2	GND	-	-
16	3	RCV	RCV+	Do not connect
17	3	RTS	TMT+	RCV+/TMT+
18	3	TMT	TMT-	RCV-/TMT-
19	3	CTS	RCV-	Do not connect
20	3	GND	-	-
21..25	4	Wie Pin 1 bis 5	Wie Pin 1 bis 5	Wie Pin 1 bis 5
26..30	5	Wie Pin 1 bis 5	Wie Pin 1 bis 5	Wie Pin 1 bis 5
31..35	6	Wie Pin 1 bis 5	Wie Pin 1 bis 5	Wie Pin 1 bis 5
36..40	7	Wie Pin 1 bis 5	Wie Pin 1 bis 5	Wie Pin 1 bis 5

10.12.3. Besondere Eigenschaften

Parameter	Wert	Einheit
RS-232		
Eingangsspannung, max.	±25	V
Eingangsschwelle Low, min.	0,6	V
Eingangsschwelle High, max.	2,0	V
Hysterese, typ.	0,5	V
Eingangswiderstand, min./typ./max.	3/5/7	V
Ausgangsspannung (3 kΩ an GND), min./typ.	±5/±5,4	V
Ausgangsstrom (Ausgang an GND), max.	±60	mA
RS-422 und RS-485		
Eingangswiderstand, min.	48	kΩ
Eingangsstrom, max. (-7V .. +12V)	-0,15/0,25	mA
Differentielle Eingangsschwelle, min./max.	-50/-200	mV
Hysterese, typ.	30	mV
Differentielle Ausgangsspannung, min. (RS-422/RS-485) (RS-485 = 27 Ω, RS-422 = 50 Ω)	2/1,5	V