

## Anwendungs-Programmierung der MFR-Modul Serie

Stand: 2010-01-28 \* Seite 1 von 4

Nachfolgend werden die verwendeten Befehle für den Dialog mit den MFR-Modulen erläutert, um das Erstellen eigener Anwendungs-Programme zu erleichtern. Eventuelle Fehler oder versionsübergreifende Änderungen sind jedoch nicht auszuschließen. In jedem Falle sollten die Ergebnisse der eigenen Programmierung direkt an der Schnittstelle mit geeigneten Tools, wie z.B. einem Terminalprogramm kontrolliert werden, um unliebsame Überraschungen von vornherein zu minimieren.

Befehlsaufbau		
Befehl	Parameter	CR
1 Buchstabe	Parameterliste beliebiger Länge, codiert in der Form 40h + obere oder untere 4 Bit des Nutzbytes. Daraus ergeben sich druckbare ASCII-Zeichen im Bereich von '@' bis 'O'. Für jedes Nutzbyte müssen daher 2 Zeichen in der Reihenfolge MSB, LSB generiert werden.	0Dh

Befehl	Parameter	Zum Modul senden	Vom Modul empfangen
Reine Interface-Funktionen. Die üblicherweise vom Applikationsprogramm verarbeitet werden.			
<b>X</b> (58h)	ohne	Erzwingt den Neustart des Moduls. Es ist ein Soft-Reset, der die Dialogfähigkeit des Moduls voraussetzt!	Wird vom Modul nicht generiert Das Modul sendet jedoch nach dem Neustart die Identität (z.B.: XSP01R)
	ohne	Fordert den Status der Ausgänge an	Wird vom Modul nicht generiert
<b>O</b> (4Fh)	1. Zeichen Kanal 7..4	Setzt die Ausgänge 0..7 gem. übergebenen Bitmuster	Status der Ausgänge 0..7
	2. Zeichen Kanal 3..0		a) als Antwort auf Anforderung b) automatisch bei Veränderung
	3. Zeichen Maske 7..4	Optional - Wenn vorhanden: Es werden nur die Ausgänge beeinflusst, deren Maskenbit gesetzt ist. <b>*** ab Firmware V 1.10 ***</b>	Wird vom Modul nicht generiert
	4. Zeichen Maske 3..0		
<b>o</b> (6Fh)	ohne	Fordert den Status der Ausgänge an, wie 'O'	Wird vom Modul nicht generiert
	1. Zeichen Bit-Adresse	Setzt nur an der Bit-Adresse (Ausgang 0..7) den Bit-Status (0 oder 1), andere Ausgänge werden nicht berührt. <b>*** ab Firmware V 1.10 ***</b>	Wird vom Modul nicht generiert
	2. Zeichen Bit-Status		
<b>I</b> (49h)	ohne	Fordert den Status der Eingänge an	Wird vom Modul nicht generiert
	1. Zeichen Kanal 7..4	<b>Nur für Testzwecke</b> <b>Bewirkt ODER-Verknüpfung mit dem zugehörigen physikalischen Eingang</b>	Status der Eingänge 0..7
	2. Zeichen Kanal 3..0		a) als Antwort auf Anforderung b) automatisch bei Veränderung
<b>D</b> (44h)	1. Zeichen t / 16	Setzt und aktiviert die Dialog-Überwachungszeit <b>t</b> in 100-ms-Einheiten. Bleiben die Kommandos O oder I während der Zeit <b>t</b> aus, werden alle Ausgänge in den passiven Status zurückgesetzt. Der Zeitwert t=0 deaktiviert die Überwachungsfunktion. <b>*** ab Firmware V 1.10 ***</b>	
	2. Zeichen t % 16		

### ZEB AUTOMATION LIMITED

Wakefield, West Yorkshire, GB

Niederlassung Deutschland

Obentrautstr. 35, D-10963 Berlin

☎ +49-30-6212667 📠 +49-30-25294347 ✉ info@zeb-automation.de

## Anwendungs-Programmierung der MFR-Modul Serie

Stand: 2010-01-28 \* Seite 2 von 4

Befehl	Parameter	Zum Modul senden	Vom Modul empfangen
Weitere Kommandos, die die internen Funktionen des MFR-Moduls steuern, sie werden üblicherweise vom MFR-Konfigurationsprogramm verarbeitet. Wird das Modul lediglich als I/O-Interface genutzt, sind diese Kommandos belanglos.			
<b>Q</b> (51h)	ohne	Fordert Parameterliste aus dem Modul an Die erwartete Rückmeldung besteht aus: Name ( <b>N</b> ), Version ( <b>V</b> ), Seriennummer ( <b>S</b> ), ID ( <b>U</b> )	Wird vom Modul nicht generiert
<b>N</b> (4Eh)	ohne	Fordert den Namen des Moduls an	Ein freier Name, bestehend aus maximal 20 druckbaren Zeichen
	Maximal 20 Zeichen	Wird vom Modul nicht ausgewertet	
<b>n</b> (6Eh)	Maximal 20 Zeichen	Setzt einen frei vergebbaren Namen, bestehend aus maximal 20 druckbaren Zeichen.	Wird vom Modul nicht generiert
<b>V</b> (56h)	Offene Zeichenfolge	Wird vom Modul nicht ausgewertet	Versionsnummer der Firmware in der Notierung 'Version.Kompilation'
<b>S</b> (53h)	Block 16 Zeichen	Wird vom Modul nicht ausgewertet	Einmalige Seriennummer des Moduls in hexadezimaler Notierung in der Reihenfolge Baustein-Typ, Num0..5, BCC
<b>Z</b> (5Ah)	1. Zeichen Kanal #	Fordert die Funktionseinstellungen des gewählten Kanals an	Funktionsdatenblock des Kanals, je 1 Byte für: Funktionsindex, Eingangsmaske, t1-Wert, t1-Zeitbasis, t2-Wert, t2-Zeitbasis, Flag1, Flag2
	16 Zeichen Datenblock	Nicht erforderlich	
<b>z</b> (7Ah)	1. Zeichen Kanal #	Setzt den Funktionsdatenblock des Kanals	Wird vom Modul nicht generiert
	16 Zeichen Datenblock		
<b>U</b> (55h)	1. Zeichen Ausgang	Wird vom Modul nicht ausgewertet	Modultyp-Identifikation Ausgang : L = Halbleiter, R = Relais
	Offene Zeichenfolge		Modultyp-Identifikation Kurzbezeichnung beinhaltet Schnittstellentyp (Param[2]): E = Ethernet, R = RS-232, U = USB
<b>T</b> (54h)	1. Zeichen Meldung #	Wird vom Modul nicht ausgewertet	Gespeicherte Meldungstexte bis zu 20 Zeichen Systembedingt besteht 1 Meldungstext aus 2 Teilmeldungen (gerade + ungerade #) max. 8 Meldungen (= 16 Teilmeldungen)
	Maximal 10 Zeichen		
<b>t</b> (74h)	1. Zeichen Meldung #	Setzt frei definierbare Meldungstexte Systembedingt besteht 1 Meldungstext aus 2 Teilmeldungen (gerade + ungerade #) max. 8 Meldungen (= 16 Teilmeldungen)	Wird vom Modul nicht generiert
	Maximal 10 Zeichen		

### Hinweise zur Programmierung

Die Vorgehensweise bei der Schnittstellenprogrammierung ist etwas von der Programmiersprache und der strukturellen Konzeption abhängig, daher kann kein allgemein gültiges Konzept vorgeschlagen werden. In der Regel lassen alle Programmiersprachen, entweder direkt oder indirekt über ein-

#### ZEB AUTOMATION LIMITED

Wakefield, West Yorkshire, GB

Niederlassung Deutschland

Obentrautstr. 35, D-10963 Berlin

☎ +49-30-6212667 📠 +49-30-25294347 ✉ info@zeb-automation.de

## Anwendungs-Programmierung der MFR-Modul Serie

Stand: 2010-01-28 \* Seite 3 von 4

gebundene Funktions-Bibliotheken, den Zugriff auf Ausgabekanäle zu. Im Downloadbereich unserer Web-Präsenz haben wir für die eine oder andere Programmiersprache Beispiele für eine mögliche Vorgehensweise bereitgestellt.

Das **Senden** ist in der Regel problemlos. Es wird der gewünschte Ausgabestring zusammengestellt und entweder zeichenweise oder als komplette Zeichenkette dem Ausgabeobjekt übergeben. Es wird vorausgesetzt, dass die zu benutzende Schnittstelle zu Beginn entsprechend parametrisiert wurde. Ob von den jeweiligen Senderroutinen zurückgegebene Statusinformationen ausgewertet werden, liegt einzig im Ermessen des Programmierers. Das MFR-Modul empfängt die Zeichenfolge in einem Empfangspuffer und wertet die Zeichenfolge nach Empfang eines CR (0Dh) in Abhängigkeit des ersten Zeichens aus. Anschließend wird der Empfangspuffer zurückgesetzt. Ist das erste Zeichen nicht als definiertes Kommando zu interpretieren, wird der Rest nicht mehr 'gesehen'. Daher ist zu beachten, dass sich keine vagabundierenden Zeichen im Ausgabekanal befinden dürfen. Dieses Problem tritt gelegentlich bei der Initialisierung von Schnittstellen auf. Falls es nicht vermieden werden kann, sollte vor dem ersten sinnvollen Kommando ein CR ausgegeben werden, damit im MFR-Modul der Empfangspuffer geleert wird. Ein vom System an das CR angehängenes LF (0Ah) stört nicht, es wird beim Empfang überlesen. Bei rascher Folge von Ausgabekommandos und besonders bei Nichtauswertung von Statusmeldungen der angewandten Senderroutine ist nicht auszuschließen, dass während der Auswertung im MFR-Modul das nachfolgende Kommando zwar im Empfangspuffer landet, aber nicht mehr 'gesehen' wird. Das gleiche passiert, wenn ein abschließendes CR verloren geht. Das CR eines nachfolgenden Kommandos führt dann lediglich zur Ausführung des vorherigen, unvollständigen Kommandos. Der zum CR gehörende Funktionsbefehl geht somit verloren.

Das **Empfangen** ist etwas diffiziler. Sofern Hardware und Programmierfunktionen es zulassen, sollte ein Hintergrund-Puffer für den Empfang der Zeichenfolgen vom MFR-Modul verwendet werden. Da ausschließlich druckbare Zeichen vorkommen, kann auf das abschließende CR getriggert werden, um ein Empfangsereignis zur Auswertung auszulösen. Da in Abhängigkeit des Programms und der Art der ausgesandten Befehlssequenz mehrere Antworten des MFR-Moduls im Empfangspuffer eingelaufen sein können, muss das Auslesen wiederholt werden, bis der Puffer leer ist. Dabei ist zu beachten, dass das Modul auch ohne besondere Anforderung Zustandsänderungen der Ein- und Ausgänge unaufgefordert senden kann. Die Eingänge werden zyklisch im 100-ms-Takt abgefragt, wird eine Änderung des jeweilig letzten Status festgestellt, erfolgt eine entsprechende Mitteilung. Soll nur die Antwort auf eine gezielte Anfrage ausgewertet werden, muss vor dem Aussenden des Anforderungsbefehls der Empfangspuffer geleert werden, andernfalls werden frühere, noch nicht abgeholte, Mitteilungen ausgewertet, die möglicherweise ein falschen Zustand widerspiegeln. Wird ohne Hintergrundpuffer gearbeitet, was eigentlich nur bei der seriellen RS232-Schnittstelle sinnvoll sein kann, ist die Empfangsschleife so zu gestalten, dass die Übertragungsgeschwindigkeit von 9600 Baud sicher verarbeitet werden kann, da sonst Zeichen verloren gehen können. Besonders bei Hochsprachen-Funktionen und 'langsamer Hardware' ist daher Vorsicht geboten. In diesen Fällen ist es besser, die Register der Kommunikationsbausteine direkt anzusprechen.

Modem-Steuersignale werden vom MFR-Modul mit RS232-Schnittstelle weder generiert noch ausgewertet, es werden nur die Leitungen Rx und Tx genutzt. Sofern die Steuersignale für den Datentransfer zwingend erforderlich sind, müssen sie anderweitig erzeugt werden.

Werden virtuelle COM-Ports bzw. die direkte USB- oder Ethernet-Verbindung genutzt, ist zu beachten, dass diese Schnittstellen ihr eigenes Zeitverhalten haben und dass ein zu übertragender Datenblock nicht zwangsläufig auch als ein einziger Block empfangen werden muss. Das muss u.U. in der Empfangsauswertung entsprechend berücksichtigt werden.

### ZEB AUTOMATION LIMITED

Wakefield, West Yorkshire, GB

Niederlassung Deutschland

Obentrautstr. 35, D-10963 Berlin

☎ +49-30-6212667 📠 +49-30-25294347 ✉ info@zeb-automation.de

## Anwendungs-Programmierung der MFR-Modul Serie

Stand: 2010-01-28 \* Seite 4 von 4

### Kleine Hilfsmittel

Zeichen	Bit 7/3	Bit 6/2	Bit 5/1	Bit 4/0	Zeichen	Bit 7/3	Bit 6/2	Bit 5/1	Bit 4/0
@	0	0	0	0	H	1	0	0	0
A	0	0	0	1	I	1	0	0	1
B	0	0	1	0	J	1	0	1	0
C	0	0	1	1	K	1	0	1	1
D	0	1	0	0	L	1	1	0	0
E	0	1	0	1	M	1	1	0	1
F	0	1	1	0	N	1	1	1	0
G	0	1	1	1	O	1	1	1	1

LSB / MSB	0_	1_	2_	3_	4_	5_	6_	7_
_0	NUL	DLE	SP	0	@	P	`	p
_1	SOH	DC1	!	1	A	Q	a	q
_2	STX	DC2	„	2	B	R	b	r
_3	ETX	DC3	#	3	C	S	c	s
_4	EOT	DC4	\$	4	D	T	d	t
_5	ENQ	NAK	%	5	E	U	e	u
_6	ACK	SYN	&	6	F	V	f	v
_7	BEL	ETB	'	7	G	W	g	w
_8	BS	CAN	(	8	H	X	h	x
_9	HT	EM	)	9	I	Y	i	y
_A	LF	SUB	*	:	J	Z	j	z
_B	VT	ESC	+	;	K	[	k	{
_C	FF	FS	,	<	L	\	l	
_D	CR	GS	-	=	M	]	m	}
_E	SO	RS	.	>	N	^	n	~
_F	SI	US	/	?	O	_	o	DEL

#### ZEB AUTOMATION LIMITED

Wakefield, West Yorkshire, GB

Niederlassung Deutschland

Obentrautstr. 35, D-10963 Berlin

☎+49-30-6212667 📠+49-30-25294347 ✉info@zeb-automation.de