

fischertechnik Robo Interface

Beschreibung der seriellen Robo Interface Firmware Routinen

Version Firmware: 0.35
Stand: 08.02.2005

Knobloch GmbH
Weedgasse 14
55234 Erbes-Büdesheim

entwicklung@knobloch-gmbh.de

1 Inhaltsverzeichnis

1	Inhaltsverzeichnis.....	2
2	Robo-Interface Programme.....	3
3	PC-Schnittstellen.....	3
3.1	Allgemeines.....	3
3.2	Online-Steuerung	5
3.2.1	Aktivierungs-codes für das Robo-Interface.....	5
3.2.2	Kompatibilität zum Intelligent Interface.....	6
3.2.3	Steuern der Ein- / Ausgänge	8
3.2.4	Abfrage von Interfacewerten	11
3.2.5	Programm starten.....	13
3.2.6	Programm stoppen.....	13
3.2.7	Programm löschen	14
3.2.8	Programm aktivieren / deaktivieren	14
3.2.9	Programmname auslesen	15

2 Robo-Interface Programme

In das Robo-Interface können bis zu drei Programme per Download gespeichert werden. Programm 1 und Programm 2 werden dauerhaft in einem FLASH-Speicher abgelegt, ein drittes Programm kann in das RAM gespeichert werden. Der RAM-Speicher wird gelöscht, wenn ein Flash-Programm gestartet wird und bei Stromausfall am Interface.

Die Auswahl des aktiven Programms erfolgt durch den Programmtaster. Wird dieser länger als 0,5 Sek. betätigt, kann das gewünschte Programm ausgewählt werden. Es leuchten nacheinander die beiden Programm-LEDs für Programm 1 / 2 auf. Ist ein Programm im RAM abgelegt, wird dieses durch die beiden leuchtenden LED's angezeigt. Sind die Speicherplätze leer, kann das betreffende Programm nicht ausgeführt werden.

Zum Starten oder Stoppen des angezeigten Programms muß der Programm-Taster kurz betätigt werden (< 500ms). Das Auswählen, Starten und Stoppen von Programmen kann auch über die PC-Schnittstellen erfolgen.

3 PC-Schnittstellen

1.1 Allgemeines

Die Auswahl der Schnittstellen erfolgt am Robo-Interface per Tastendruck. Nach dem Einschalten ist der "AutoScan"-Modus aktiv. Es werden die USB, serielle Schnittstelle und das Funkmodul (falls vorhanden) überprüft, ob Daten vorhanden sind. Erkennbar ist dieser Zustand an dem Aufleuchten der Schnittstellen-Leuchtdioden.

Sobald eine Schnittstelle Daten sendet, werden die anderen Schnittstellen gesperrt. Die aktive Schnittstelle blinkt, um die Datenkommunikation anzuzeigen. Wenn länger als 300ms keine Daten über die aktive Schnittstelle fließen, schaltet sich der AutoScan-Modus wieder ein.

Durch Drücken des "Port"-Tasters wird die nächste Betriebsart anhand nachfolgender Tabelle ausgewählt.

- | | | |
|----|-----------|------------------------------------|
| 1. | AutoScan | USB - Seriell - Funk ¹⁾ |
| 2. | AutoScan | USB - Seriell |
| 3. | USB | |
| 4. | Seriell | |
| 5. | IR-Direkt | |

1)

Diese Betriebsart wird nur aktiviert, wenn das Funkmodul installiert ist.

Wird der Port-Taster länger als 3 Sekunden betätigt, schaltet sich das Interface in den "Intelligent-Interface Online-Modus". Die serielle Schnittstelle arbeitet dann mit den Parametern 9600,n,8,1. Zur Anzeige der Betriebsart blinkt die "SER" - Leuchtdiode sehr schnell. In dieser Betriebsart verhält sich das Interface wie ein Intelligent Interface im

Online Modus. Es können jedoch keine Programme herunter geladen werden. Durch einen Druck auf den Port-Taster wird wieder die AutoScan Betriebsart eingestellt.

Im passiven Modus wird das aktive Robo-Interface über eine serielle Schnittstelle (RS232), über USB oder über Funk gesteuert.

Die serielle Schnittstelle muß mit folgenden Parametern initialisiert werden:

Baudrate: 38400 bps
Parität: keine
Datenbit: 8
Stopbit: 1

Die Kommunikation unterliegt einem Client-Server-Modell. Vom PC wird ein Dienst angefordert, der vom Robo-Interface ausgeführt und bestätigt wird. Jeder Nachricht vom Robo-Interface muss eine Anforderung seitens des PC vorausgegangen sein.

Wichtig:

Unter Windows gibt es Treiber, welche die serielle Schnittstelle nach Geräten (z.B. Mäuse) "abscannen", wenn die Schnittstelle nicht durch ein Programm geöffnet wurde. Diese Steuercodes könnten im Robo-Interface eine Reaktion auslösen. Aus diesem Grunde muß vor dem Datenaustausch über die serielle Schnittstelle diese mittels des Befehls IF3_ON im Robo-Interface geöffnet werden. Vor dem Schließen der Schnittstelle in Windows sollte daher das Robo-Interface mit dem Befehl IF3_OFF wieder deaktiviert werden. Im Intelligent-Interface-Modus ist dies jedoch nicht notwendig, da bei 9600bps keine Probleme in dieser Art auftreten.

Empfohlene Vorgehensweise in einer Windows-Applikation:

- Serielle Schnittstelle im PC öffnen
- IF3_ON an Interface senden
- ...
- ... normale Steuerung über serielle Schnittstelle
- ...
- IF3_OFF an Interface senden um dieses abzuschalten
- Serielle Schnittstelle im PC schließen

Nachfolgende Codes sind zur Steuerung des Interface in der Firmware implementiert.

1.2 Online-Steuerung

Bei serieller Schnittstelle werden nur die angegebenen Datenbytes übertragen.

Die nachfolgenden Befehle arbeiten nur im sogenannten "Online-Modus". Dieser Modus findet dann statt, wenn keine der Programm-LEDs blinkt.

1.2.1 Aktivierungscode für das Robo-Interface

Code:	IF3_ON	OxA1
Byte Anzahl senden:	14	
Byte Anzahl empf.:	5	
1. Sendebyte:	[0]	Code Byte "0xA1"
2. Sendebyte:	[1]	Datenbyte "f" 0x66
3. Sendebyte:	[2]	Datenbyte "t" 0x74
4. Sendebyte:	[3]	Datenbyte "-" 0x2D
5. Sendebyte:	[4]	Datenbyte "R" 0x52
6. Sendebyte:	[5]	Datenbyte "o" 0x6F
7. Sendebyte:	[6]	Datenbyte "b" 0x62
8. Sendebyte:	[7]	Datenbyte "o" 0x6F
9. Sendebyte:	[8]	Datenbyte "-" 0x2D
10. Sendebyte:	[9]	Datenbyte "O" 0x4F
11. Sendebyte:	[A]	Datenbyte "N" 0x4E
12. Sendebyte:	[B]	Datenbyte "-" 0x2D
13. Sendebyte:	[C]	Datenbyte "V" 0x56
14. Sendebyte:	[D]	Datenbyte "1" 0x31
1. Empfangsbyte	[0]	0x5E (<i>negierter Aufruf-Code</i>)
2. Empfangsbyte	[1]	Byte-0 der Firmware Versionsnummer
3. Empfangsbyte	[2]	Byte-1 der Firmware Versionsnummer
4. Empfangsbyte	[3]	Byte-2 der Firmware Versionsnummer
5. Empfangsbyte	[4]	Byte-3 der Firmware Versionsnummer (3.2.1.0) <i>Format: F.W.B.L (Firmware, Bootloaderversionen)</i>

Über den Code IF3_ON wird die serielle Schnittstelle im Robo-Interface "aktiviert". Erst nach diesem Eröffnungscode sind die nachfolgenden Steuercodes über die serielle Schnittstelle zu nutzen. Dieser Befehl ist nicht notwendig, wenn sich das Robo-Interface im "Intelligent-Interface-Modus" befindet.

Code:	IF3_OFF	OxA2
Byte Anzahl senden:	1	
Byte Anzahl empf.:	1	
1. Sendebyte:	[0]	Code Byte "0xA2"
1. Empfangsbyte	[0]	0x5D

Über den Code IF3_OFF wird die serielle Schnittstelle im Robo-Interface "deaktiviert". Um wieder über die serielle Schnittstelle mit dem Interface zu kommunizieren, muß diese mit dem Code IF3_ON aktiviert werden. Dieser Befehl ist nicht notwendig, wenn sich das Robo-Interface im "Intelligent-Interface-Modus" befindet.

1.2.2 Kompatibilität zum Intelligent Interface

Diese Befehle sind im "Intelligent-Interface-Modus" (bei 9600 Baud), aber auch im normalen seriellen Modus nutzbar (bei 38400 Baud).

Code:	ABF_EA	OxC1 (kompatibel zu Intelligent Interface)
Byte Anzahl senden:	2	
Byte Anzahl empf.:	1	

2. Sendebyte:	Zustand der Master-Ausgänge (8765 4321)
1. Empfangsbyte:	Zustand der Master-Eingänge (8765 4321)

Dieser Befehl setzt die acht Ausgänge und liefert den Zustand der Eingänge.

Code:	ABF_EX	OxC5 (kompatibel zu Intelligent Interface)
Byte Anzahl senden:	2	
Byte Anzahl empf.:	3	

2. Sendebyte:	Zustand der Master-Ausgänge (8765 4321)
1. Empfangsbyte:	Zustand der Master-Eingänge (8765 4321)
2. Empfangsbyte:	Analog AX-Wert Bit 8+9 (---- --98)
3. Empfangsbyte:	Analog AX-Wert Bit 0..7 (7654 3210)

Dieser Befehl setzt die acht Ausgänge und liefert den Zustand der Eingänge, sowie den Wert des Analog-Eingangs AX (0..1023).

Code:	ABF_EY	OxC9 (kompatibel zu Intelligent Interface)
Byte Anzahl senden:	2	
Byte Anzahl empf.:	3	

2. Sendebyte:	Zustand der Master-Ausgänge (8765 4321)
1. Empfangsbyte:	Zustand der Master-Eingänge (8765 4321)
2. Empfangsbyte:	Analog AY-Wert Bit 8+9 (---- --98)
3. Empfangsbyte:	Analog AY-Wert Bit 0..7 (7654 3210)

Dieser Befehl setzt die acht Ausgänge und liefert den Zustand der Eingänge, sowie den Wert des Analog-Eingangs AY (0..1023).

Code:	ABF_EAS	OxC2 (kompatibel zu Intelligent Interface)
Byte Anzahl senden:	3	
Byte Anzahl empf.:	2	

2. Sendebyte:	Zustand der Master-Ausgänge (8765 4321)
3. Sendebyte:	Zustand der Slave1-Ausgänge (8765 4321)
1. Empfangsbyte:	Zustand der Master-Eingänge (8765 4321)
2. Empfangsbyte:	Zustand der Slave1-Eingänge (8765 4321)

Dieser Befehl setzt die acht Ausgänge am Master, sowie am Slave1 und liefert den Zustand der Eingänge am Master und am Slave1.

Code:	ABF_EXS	OxC6 (kompatibel zu Intelligent Interface)
Byte Anzahl senden:	3	
Byte Anzahl empf.:	4	

2. Sendebyte:	Zustand der Master-Ausgänge (8765 4321)
3. Sendebyte:	Zustand der Slave1-Ausgänge (8765 4321)
1. Empfangsbyte:	Zustand der Master-Eingänge (8765 4321)
2. Empfangsbyte:	Zustand der Slave1-Eingänge (8765 4321)
3. Empfangsbyte:	Analog AX-Wert Bit 8+9 (---- --98)
4. Empfangsbyte:	Analog AX-Wert Bit 0..7 (7654 3210)

Dieser Befehl setzt die acht Ausgänge am Master sowie am Slave1 und liefert den Zustand der Eingänge vom Master, Slave1 und vom Master-AX.

Code:	ABF_EYS	OxCA (kompatibel zu Intelligent Interface)
Byte Anzahl senden:	3	
Byte Anzahl empf.:	4	

2. Sendebyte:	Zustand der Master-Ausgänge (8765 4321)
3. Sendebyte:	Zustand der Slave1-Ausgänge (8765 4321)
1. Empfangsbyte:	Zustand der Master-Eingänge (8765 4321)
2. Empfangsbyte:	Zustand der Slave1-Eingänge (8765 4321)
3. Empfangsbyte:	Analog AY-Wert Bit 8+9 (---- --98)
4. Empfangsbyte:	Analog AY-Wert Bit 0..7 (7654 3210)

Dieser Befehl setzt die acht Ausgänge am Master sowie am Slave1 und liefert den Zustand der Eingänge vom Master, Slave1 und vom Master-AY.

1.2.3 Steuern der Ein- / Ausgänge

Code:	ABF_IF3_M_1	Ox81
Byte Anzahl senden:	5	
Byte Anzahl empf.:	7	

1. Sendebyte:	[0]	Code Byte "0x81"
2. Sendebyte:	[1]	Ausgang Master (7654 3210)
3. Sendebyte:	[2]	Geschwindigkeit Master Ausgang 0..2 (22111000)
4. Sendebyte:	[3]	Geschwindigkeit Master Ausgang 2..5 (54443332)
5. Sendebyte:	[4]	Geschwindigkeit Master Ausgang 5..7 (77766655)

1. Empfangsbyte	[0]	Eingang Master
2. Empfangsbyte	[1]	Eingang AX (Bit 0..7)
3. Empfangsbyte	[2]	Eingang AY (Bit 0..7)
4. Empfangsbyte	[3]	Eingang A1 (Bit 0..7)
5. Empfangsbyte	[4]	Eingang A2 (Bit 0..7)
6. Empfangsbyte	[5]	Eingang AX..A2 Bit 8+9 (2211YYXX)
7. Empfangsbyte	[6]	IR-Code (AS2 AS1 - CTTTT)

TTTT=Tastenummer (1..11)
C=Code 1/2:
0 = Code 1
1 = Code 2

Über den Code ABF_IF3_M_1 (Master) werden die Ausgangsdaten an das Master-IF gesendet und die wichtigsten Eingänge gelesen. Die Geschwindigkeitsinformationen pro Interface (8 Ausgänge, je drei Bit = 24 Bit) werden in 3 Bytes übertragen.

Code:	ABF_IF3_M_2	Ox82
Byte Anzahl senden:	5	
Byte Anzahl empf.:	13	

1. Sendebyte:	[0]	Code Byte "0x81"
2. Sendebyte:	[1]	Ausgang Master (7654 3210)
3. Sendebyte:	[2]	Geschwindigkeit Master Ausgang 0..2 (22111000)
4. Sendebyte:	[3]	Geschwindigkeit Master Ausgang 2..5 (54443332)
5. Sendebyte:	[4]	Geschwindigkeit Master Ausgang 5..7 (77766655)
1. Empfangsbyte	[0]	Eingang Master
2. Empfangsbyte	[1]	Eingang AX (Bit 0..7)
3. Empfangsbyte	[2]	Eingang AY (Bit 0..7)
4. Empfangsbyte	[3]	Eingang A1 (Bit 0..7)
5. Empfangsbyte	[4]	Eingang A2 (Bit 0..7)
6. Empfangsbyte	[5]	Eingang AX..A2 Bit 8+9 (2211YYXX)
7. Empfangsbyte	[6]	Eingang AZ (Bit 0..7)
8. Empfangsbyte	[7]	Eingang AS1 (Bit 0..7)
9. Empfangsbyte	[8]	Eingang AS2 (Bit 0..7)
10. Empfangsbyte	[9]	Eingang Versorgung (Bit 0..7)
11. Empfangsbyte	[10]	Eingang AZ..Versorgung Bit 8+9 (VV2211ZZ)
12. Empfangsbyte	[11]	IR-Code (AS2 AS1 - CTTTT)
		TTTT=Tastenummer (1..11)
		C=Code 1/2:
		0 = Code 1
		1 = Code 2
13. Empfangsbyte	[12]	Reserve

Über den Code ABF_IF3_M_1 (Master) werden die Ausgangsdaten an das Master-IF gesendet und die wichtigsten Eingänge gelesen. Die Geschwindigkeitsinformationen pro Interface (8 Ausgänge, je drei Bit = 24 Bit) werden in 3 Bytes übertragen.

Code:	ABF_IF_COMPLETE	OxF2
Byte Anzahl senden:	17	
Byte Anzahl empf.:	21	
1. Sendebyte:	[0]	Code Byte
2. Sendebyte:	[1]	Ausgang Master
3. Sendebyte:	[2]	Geschwindigkeit Master Ausgang 0..2 (22111000)
4. Sendebyte:	[3]	Geschwindigkeit Master Ausgang 2..5 (54443332)
5. Sendebyte:	[4]	Geschwindigkeit Master Ausgang 5..7 (77766655)
6. Sendebyte:	[5]	Ausgang Slave1
7. Sendebyte:	[6]	Geschwindigkeit Slave1 Ausgang 0..2 (22111000)
8. Sendebyte:	[7]	Geschwindigkeit Slave1 Ausgang 2..5 (54443332)
9. Sendebyte:	[8]	Geschwindigkeit Slave1 Ausgang 5..7 (77766655)
10. Sendebyte:	[9]	Ausgang Slave2
11. Sendebyte:	[10]	Geschwindigkeit Slave2 Ausgang 0..2 (22111000)
12. Sendebyte:	[11]	Geschwindigkeit Slave2 Ausgang 2..5 (54443332)
13. Sendebyte:	[12]	Geschwindigkeit Slave2 Ausgang 5..7 (77766655)
14. Sendebyte:	[13]	Ausgang Slave3
15. Sendebyte:	[14]	Geschwindigkeit Slave3 Ausgang 0..2 (22111000)
16. Sendebyte:	[15]	Geschwindigkeit Slave3 Ausgang 2..5 (54443332)
17. Sendebyte:	[16]	Geschwindigkeit Slave3 Ausgang 5..7 (77766655)
1. Empfangsbyte	[0]	Eingang Master
2. Empfangsbyte	[1]	Eingang Slave1
3. Empfangsbyte	[2]	Eingang Slave2
4. Empfangsbyte	[3]	Eingang Slave3
5. Empfangsbyte	[4]	Eingang AX (Bit 0..7)
6. Empfangsbyte	[5]	Eingang AY (Bit 0..7)
7. Empfangsbyte	[6]	Eingang A1 (Bit 0..7)
8. Empfangsbyte	[7]	Eingang A2 (Bit 0..7)
9. Empfangsbyte	[8]	Eingang AX..A2 Bit 8+9 (2211YYXX)
10. Empfangsbyte	[9]	Eingang AZ (Bit 0..7)
11. Empfangsbyte	[10]	Eingang D1 (Bit 0..7)
12. Empfangsbyte	[11]	Eingang D2 (Bit 0..7)
13. Empfangsbyte	[12]	Eingang Versorgung (Bit 0..7)
14. Empfangsbyte	[13]	Eingang AZ..Versorgung Bit 8+9 (VV2211ZZ)
15. Empfangsbyte	[14]	IR-Code (AS2 AS1 - CTTTT) TTTT=Tastenummer (1..11) C=Code 1/2: 0 = Code 1 1 = Code 2
16. Empfangsbyte	[15]	Reserve
17. Empfangsbyte	[16]	Reserve
18. Empfangsbyte	[17]	Eingang Slave1 AX (Bit 0..7)
19. Empfangsbyte	[18]	Eingang Slave2 AX (Bit 0..7)
20. Empfangsbyte	[19]	Eingang Slave3 AX (Bit 0..7)
21. Empfangsbyte	[20]	Eingang Slave3..1 AX Bit 8+9 (--33 22 11)

Über den Code IF _ABF _COMPLETE werden alle Daten in das Interface gesendet und alle Informationen aus dem Interface abgeholt. Die Geschwindigkeitsinformationen pro Interface (8 Ausgänge, je drei Bit = 24 Bit) werden in 3 Bytes übertragen.

1.2.4 Abfrage von Interfacewerten

Code:	ABF_IF_INTERN	0xF0
Byte Anzahl senden:	2	
Byte Anzahl empf.:	Vom SUB-Code abhängig	

2. Sendebyte: Sub-Code

Dieser Befehl dient zur Abfrage interner Werte. Es werden immer zwei Bytes gesendet. Das zweite Sendebyte bestimmt den SUB-Code. Die Anzahl der empfangenen Bytes ist vom SUB-Code abhängig.

SUB-Code	0x01	ABF_IF_INTERN_FIRMWARE
Byte Anzahl senden:	2	
Byte Anzahl empf.:	5	
1. Sendebyte:	[0]	Code Byte "0xF0"
2. Sendebyte:	[1]	Sub-Code Byte "0x01"
1. Empfangsbyte	[0]	0xFE (=invertierter SUB-Code)
2. Empfangsbyte	[1]	Byte-0 der Firmware Versionsnummer
3. Empfangsbyte	[2]	Byte-1 der Firmware Versionsnummer
4. Empfangsbyte	[3]	Byte-2 der Firmware Versionsnummer
5. Empfangsbyte	[4]	Byte-3 der Firmware Versionsnummer (3.2.1.0)
		<i>Format: F.W.B.L (Firmware, Bootloaderversionen)</i>

Dieser Befehl dient zur Abfrage der Firmwareversion (Format: 3.2.1.0).

SUB-Code	0x02	ABF_IF_INTERN_SN
Byte Anzahl senden:	2	
Byte Anzahl empf.:	5	
1. Sendebyte:	[0]	Code Byte "0xF0"
2. Sendebyte:	[1]	Sub-Code Byte "0x02"
1. Empfangsbyte	[0]	0xFD (=invertierter SUB-Code)
2. Empfangsbyte	[1]	Byte-0 der Seriennummer (Low-Byte)
3. Empfangsbyte	[2]	Byte-1 der Seriennummer
4. Empfangsbyte	[3]	Byte-2 der Seriennummer
5. Empfangsbyte	[4]	Byte-3 der Seriennummer (High-Byte)

Dieser Befehl dient zur Abfrage der Seriennummer.

SUB-Code	0x03	ABF_IF_INTERN_ZUSTAND
Byte Anzahl senden:	2	
Byte Anzahl empf.:	3	
1. Sendebyte:	[0]	Code Byte "0xF0"
2. Sendebyte:	[1]	Sub-Code Byte "0x03"
1. Empfangsbyte	[0]	0xFC (=invertierter SUB-Code)
2. Empfangsbyte	[1]	Modusbyte:
		00 = Online-Modus
		01 = Aktiv Modus, Programm läuft
		(Byte 3 = Nummer des laufenden Programm)
3. Empfangsbyte	[2]	Datenbyte (s. 2. Empfangsbyte)

Dieser Befehl dient zur Abfrage des Zustandes des angeschlossenen Interface.

SUB-Code	0x10	ABF_IF_INTERN_MEMORY
Byte Anzahl senden:	2	
Byte Anzahl empf.:	30	
1. Sendebyte:	[0]	Code Byte "0xF0"
2. Sendebyte:	[1]	Sub-Code Byte "0x10"
FLASH-1 Bereich Start – End Adresse (20 Bit: a bb cc – d ee ff)		
1. Empfangsbyte	[0]	cc (der Start Adresse)
2. Empfangsbyte	[1]	bb (der Start Adresse)
3. Empfangsbyte	[2]	0a (der Start Adresse)
4. Empfangsbyte	[3]	ff (der End Adresse)
5. Empfangsbyte	[4]	ee (der End Adresse)
6. Empfangsbyte	[5]	0d (der End Adresse)
FLASH-2 Bereich Start – End Adresse (20 Bit: a bb cc – d ee ff)		
7. Empfangsbyte	[6]	cc (der Start Adresse)
8. Empfangsbyte	[7]	bb (der Start Adresse)
9. Empfangsbyte	[8]	0a (der Start Adresse)
10. Empfangsbyte	[9]	ff (der End Adresse)
11. Empfangsbyte	[10]	ee (der End Adresse)
12. Empfangsbyte	[11]	0d (der End Adresse)
RAM Bereich Start – End Adresse (20 Bit: a bb cc – d ee ff) (Nutzbar für Programme und Variablen, Bereich steht nach Start eines Flash-Programms dem Programm komplett zur Verfügung)		
13. Empfangsbyte	[12]	cc (der Start Adresse)
14. Empfangsbyte	[13]	bb (der Start Adresse)
15. Empfangsbyte	[14]	0a (der Start Adresse)
16. Empfangsbyte	[15]	ff (der End Adresse)
17. Empfangsbyte	[16]	ee (der End Adresse)
18. Empfangsbyte	[17]	0d (der End Adresse)
Interner RAM-1 Start – End Adresse (20 Bit: a bb cc – d ee ff) (Nutzbar für Variablen und Stacks)		
19. Empfangsbyte	[18]	cc (der Start Adresse)
20. Empfangsbyte	[19]	bb (der Start Adresse)
21. Empfangsbyte	[20]	0a (der Start Adresse)
22. Empfangsbyte	[21]	ff (der End Adresse)
23. Empfangsbyte	[22]	ee (der End Adresse)
24. Empfangsbyte	[23]	0d (der End Adresse)
Interner RAM-2 Start – End Adresse (20 Bit: a bb cc – d ee ff) (Nutzbar für Variablen und Stacks, bitweise Adressierung möglich)		
25. Empfangsbyte	[24]	cc (der Start Adresse)
26. Empfangsbyte	[25]	bb (der Start Adresse)
27. Empfangsbyte	[26]	0a (der Start Adresse)
28. Empfangsbyte	[27]	ff (der End Adresse)
29. Empfangsbyte	[28]	ee (der End Adresse)
30. Empfangsbyte	[29]	0d (der End Adresse)

Dieser Befehl liefert die verfügbaren Speicherbereiche im Robo-Interface. Vor einem Programm-Download muß das Programm auf diese Adressen gelinkt werden.
Hinweis: Wir behalten uns vor, bei Firmwareupdates das Speicherlayout zu verändern, wenn dies erforderlich wird.

SUB-Code	0x28	ABF_IF_INTERN_AG_RESET
Byte Anzahl senden:	2	
Byte Anzahl empf.:	1	

1. Sendebyte:	[0]	Code Byte "0xF0"
2. Sendebyte:	[1]	Sub-Code Byte "0x28"

1. Empfangsbyte [0] 0x01 als Antwortbyte

Dieser Befehl schaltet die Ausgänge am Robo-Interface, sowie an allen angeschlossenen I/O-Extensions (Slave-Modulen) aus und setzt die PWM-Werte auf Maximum.

1.2.5 Programm starten

Code:	START_PRG	0xF4
Byte Anzahl senden:	2	
Byte Anzahl empf.:	1	

2. Sendebyte: Zu startendes Programm:
0 = FLASH 1
1 = FLASH 2
2 = RAM

1. Empfangsbyte: 0xFB (-5) = Fehler - Prüfsumme des gesp. Programms ist falsch
(Fehler im Speicher)
0xF4 (-12) = Fehler - kein Programm gespeichert
0x01 = Programm gestartet...

Mit diesem Befehl können Programme in den drei möglichen Speicherbereichen des Interface gespeichert werden. Dies entspricht dem Tastendruck am Interface mittels der START-Taste.

1.2.6 Programm stoppen

Code:	STOP_PRG	0xF8
Byte Anzahl senden:	1	
Byte Anzahl empf.:	1	

1. Empfangsbyte: 0x01 = Programm gestoppt
(-20) = Kein Programm aktiv

Dieser Befehl unterbricht ein laufendes Programm und schaltet in den Online-Modus zurück.

1.2.7 Programm löschen

Code:	DELETE_PRG	OxF5
Byte Anzahl senden:	2	
Byte Anzahl empf.:	1	
2. Sendebyte:	Zu löschendes Programm: 0 = FLASH 1 1 = FLASH 2 2 = RAM	
1. Empfangsbyte:	0xFF = Fehler: falscher Parameter angegeben 0xFE = Fehler: Programm kann nicht gelöscht werden (Fehler im Speicher / Hardwarefehler) 0x01 = Programm gelöscht	

Mit diesem Befehl können Programme in den drei möglichen Speicherbereichen des Robo-Interface gelöscht werden.

Hinweis:

Für das löschen des FLASH-Speichers werden ca. 10sek. benötigt. Dies muß beachtet werden, wenn auf PC-Seite eine TimeOut - Routine die Kommunikation überwacht.

1.2.8 Programm aktivieren / deaktivieren

Code:	SET_PROG	OxF9
Byte Anzahl senden:	2	
Byte Anzahl empf.:	1	
2. Sendebyte:	Zu aktivierendes Programm: 0 = FLASH 1 1 = FLASH 2 2 = RAM 0xFF = Programmauswahl zurücksetzen (Prog 1/2 Led abschalten)	
1. Empfangsbyte:	0x01 = kein Fehler ansonsten Fehlercode	

Mit diesem Befehl können Programme in den drei möglichen Speicherbereichen des Robo-Interface aktiviert werden (LED's werden geschaltet), die mit einem Tastendruck am Interface zu starten sind.

Mit diesem Befehl könne auch die Programm Led's ausgeschaltet werden.

1.2.9 Programmname auslesen

Code:	GET_PROG_NAME	OxFA
Byte Anzahl senden:	2	
Byte Anzahl empf.:	81	
2. Sendebyte:	Gewünschter Programmname	
	0 = FLASH 1	
	1 = FLASH 2	
	2 = RAM	
1. Empfangsbyte:	0x01 = kein Fehler	
	ansonsten Fehlercode z.B.:	
	0xFC: falscher Mem-Block angegeben	
	0xF3: falscher Versionsnummer = kein Programm gespeichert	
2..81 Empfangsbyte:	gespeicherter Beschreibungstext / Programmname	

Mit diesem Befehl kann ein gespeicherter Programmname ausgelesen werden. Da die Daten binär gespeichert werden, können diese mit diesem Befehl wieder ausgelesen werden (z.B. für Datum / Uhrzeit) um Datenbytes zu sparen.