

Softwarebeschreibung V1.1

MODBUS RTU Slave

JDA48-4-x

gültig für SW - Version > 1.2



Inhaltsverzeichnis

Kapitel	Bezeichnung	Seite
1.0	Allgemeine Beschreibung.....	4
2.0	Schematischer Aufbau und Funktion.....	5
3.0	Konfigurationsliste verfügbarer E/A Module	9
4.0	Übersicht der Holdingregister 4XXXX Referenz (16Bit Werte)	10
5.0	Erklärung einzelner Registerfunktionen.....	12
6.0	Zuordnung Eingangsmodule zu Register	19
7.0	Kommunikationsüberwachung	21
8.0	User Register	23
9.0	Register Erweiterungen (nur digitale Ausgänge)	23
10.0	Unterstützte MODBUS Befehle	24
11.0	Registersatz initialisieren	32
12.0	Fehlerbehandlung	32
13.0	CRC - Generierung	33
14.0	Programmierbeispiele	34
15.0	Anschluss-Schema Baugruppen	37
	Abbildung 1 (interne Controllermodule).....	5
	Abbildung 2 (Schematische Darstellung)	7
	Abbildung 3 (Adress-Schalter Einstellung).....	7
	Abbildung 4 (Anschlussbelegung Controller)	8
	Abbildung 5 (Zuordnung Eingang zu Statusregister)	20
	Abbildung 6 (Beispiel Watchdog Programmierung)	22
	Tabelle 1 (LED/7Segmentanzeige)	6
	Tabelle 2 (Standardwerte bei Auslieferung)	8
	Tabelle 3 (Modulkonfigurationsliste).....	9
	Tabelle 4 (Code3 für Modulabschaltung)	21
	Tabelle 5 (MODBUS Befehle) (*) nicht MODBUS konforme Kommandos	24
	Tabelle 6: 01H READ COIL STATUS Anforderung MA-SL	24
	Tabelle 7: 02H READ INPUT STATUS Anforderung MA-SL	25
	Tabelle 8: 03H READ HOLDING REGISTER Anforderung MA-SL	25
	Tabelle 9: 04H READ INPUT REGISTER Anforderung MA-SL	26
	Tabelle 10: 05H FORCE SINGLE COIL Anforderung MA-SL	26
	Tabelle 11: 06H PRESET SINGLE REGISTER Anforderung MA-SL	27
	Tabelle 12: 08H DIAGNOSTIC Anforderung MA-SL	27
	Tabelle 13: 0FH FORCE MULTIPLE COILS Anforderung MA-SL	28
	Tabelle 14: 10H PRESET MULTIPLE REGISTERS Anforderung MA-SL	28
	Tabelle 15: 11H REPORT SLAVE ID Anforderung MA-SL	29
	Tabelle 16: 40H PROG COM Anforderung MA-SL	31
	Tabelle 17: 42H INIT REGISTER Sub-Funktion 1	32

Änderungshistorie

Ab Version	Beschreibung
V1.1	Zuordnung im Register 40199 (Lüfterkontrolle) ist geändert
V1.1	Subfunktionen in Diagnostik 08_n erweitert
V1.1	Zusammensetzung und Anzahl der Bytes im Kommando 11_n (Abfragen Slave-ID) geändert

1.0 Allgemeine Beschreibung

Der JDA48-4 Controller stellt eine modulare Hardwarelösung dar, mit deren Hilfe ein Betrieb von Aktoren im Digital- und Analogbereich, sowie die Verarbeitung von digitalen Eingangssignalen möglich sind. Der JDA48-4 agiert als Slave Baugruppe und wird mit Hilfe eines seriellen MODBUS RTU Protokolls gesteuert. An einem BUS-Segment können bis zu 32 Controller angeschlossen werden.

Die Protokollimplementierung hält sich an die Vorgaben der **Modbus Organization, Inc. 37 Wheeler Rd. North Grafton, MA 01536 USA.**

Die im Einzelnen unterstützten Funktionen sind unter dem Kapitel MODBUS Funktionen näher erläutert.

Über einen vorgeschalteten Gateway können MODBUS TCP/IP Verbindungen realisiert werden. Der JDA48-4 ist mit vier Modulen ausgerüstet, die nach Kundenwunsch zusammengestellt werden können.

Zurzeit stehen E/A Module zu Verfügung, deren Parameter in relevanten 4xxxx Registern abgelegt werden. Dazu gehören Strommessung, Drahtbruch und Kurzschlussüberwachung.

Die zur Verfügung stehenden Module sind der (Modulkonfigurationsliste) zu entnehmen.

Neben den bekannten MODBUS Befehlen zur Steuerung von E/A Referenzen, stehen dem Anwender zusätzliche 4xxxx Register zur Verfügung, die Sonderfunktion wie digitale Ausgaben mit einer Skalierung von 0 – 100% im 4 – oder 6Bit Modus ohne weiteren Softwareaufwand möglich machen.

Der JDA48-4 wird serienmäßig mit RS485-Schnittstelle ausgeliefert.

Mit Hilfe eines extern zugänglichen Adressenschalters können bis zu 128 Adressen eingestellt werden. Die verwendeten RS485 Schnittstellenbausteine erlauben den Betrieb von 32 Slave-Baugruppen an einem BUS Segment.

Der DA48-4 wird mit einer Gleichspannung von 12 – 36V versorgt – Standard ist 24V.

Die Installation des Controllers geschieht mittels eines BUS-Moduls, welches die Spannungsversorgungsanschlüsse und BUS Verbindung zur Verfügung stellt.

Durch das Kommando **Report Slave ID** werden dem Anwender alle relevanten Konfigurationsdaten im Bezug auf I/O Referenzen, Register und Modulparameter als Datenblock zur Verfügung gestellt.

Aufgrund der Packungsdichte des JDA48-4 erreicht das Gerät ein hervorragendes Verhältnis im Bezug zu Preis und Leistung.

Der schematische Aufbau der internen Controllermodule ist auf der nächsten Seite zu sehen (siehe Abbildung 1).

Literaturhinweis:

Vorgaben der MODBUS Organization Inc.

Kostenloser Download unter <http://www.modbus.org/specs.php>

Link: [Modbus Protokoll Specification](#) (Modbus_Application_Protokoll_V1_1b3.pdf)

2.0 Schematischer Aufbau und Funktion

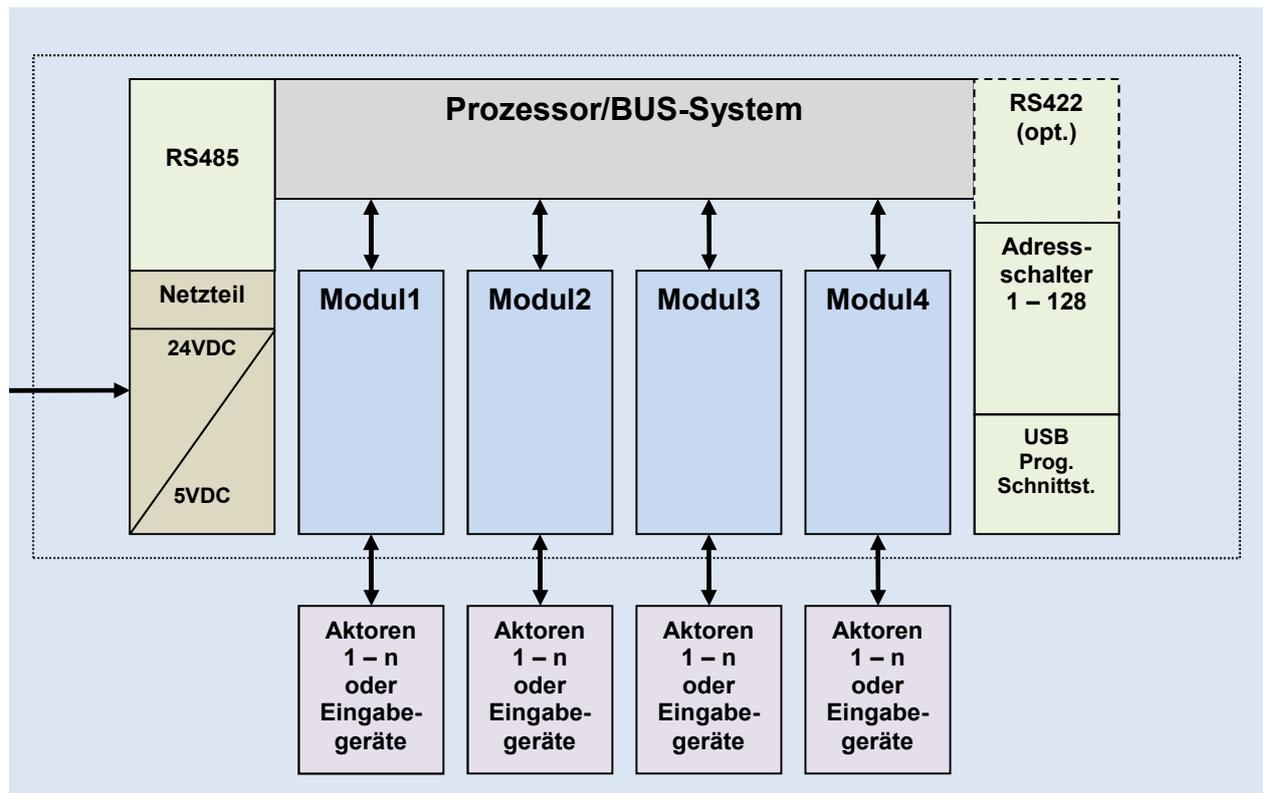


Abbildung 1 (interne Controllermodule)

Die installierten Module werden nach dem Einschalten automatisch erkannt und der Typ in einem internen FRAM nicht flüchtig gespeichert.

Es werden alle programmierten Werte des Anwenders, die keine dynamischen Vorgänge betreffen gespeichert. Ebenfalls interne Zähler, die nach Neustart wieder aus dem nichtflüchtigen Speicher geladen werden

Ausgangszustände werden nicht gespeichert und unterliegen der Kontrolle des BUS Masters. Kundenspezifische Anpassungen sind jederzeit möglich.

Eine grüne LED des JDA48-4 zeigt die Funktion des Netzteils an, sie signalisiert im Sekunden Rhythmus die ordnungsgemäße Funktion des Controllers. Sie ist ebenfalls ein Indikator für die Verarbeitung empfangener MODBUS Befehle. In diesem Fall flackert die LED für die Dauer der Verarbeitung. Kein oder statisches Leuchten signalisiert eine Fehlfunktion des JDA48-4.

Dies betrifft nur den Fall einer statischen Fehlfunktion, denn im Fall einer Softwarefehlfunktion, wird der JDA48-4 durch das Auslösen eines internen Watchdogzählers neu gestartet.

Beginnt nach dem Einschalten die LED an zu flackern, bedeutet dies, dass keine Adresse eingestellt wurde. **Der JDA48-4 ist dann nicht betriebsbereit.** An der rechten Gehäusesseite des DA48-4 ist der Adress-Schalter zugänglich. Einstellung siehe Abbildung 3 (Adress-Schalter Einstellung).

Tabelle 1 (LED/7Segmentanzeige)

DA48-4		Erklärung
LED-HB	○	Permanent AUS = keine Versorgungsspannung oder Controllerfehler
LED-HB	●	Permanent ON = Controllerfehler
LED-HB	◐	Nach Einschalten: Schnelles Blinken = keine Adresse eingestellt
LED-HB	○ ●	Controller läuft: LED blinkt langsam im Sekundenrhythmus
LED-HB	◐	Controller läuft: MODBUS Befehl wird verarbeitet. LED blinkt schnell
LED-Digital	●	Der Controller hat digitale Ausgangsmodule erkannt Blinken: Lüfterproblem
LED-Analog	●	Der Controller hat analoge Ausgangsmodule erkannt Blinken: Lüfterproblem
LED-Brk/OC	●	Bei dauerhaftes Leuchten liegt ein Drahtbruch an einem digitalen Ausgang vor, bei Leuchten und langsames Blinken ein Überstrom am Ausgang
LED-OvTemp	●	Bei dauerhaftem Leuchten wurde der Controller wegen Übertemperatur abgeschaltet.
Fehlercodes		2stellige 7Segment Anzeige
	E1	Versorgungsspannung ist kleiner 10VDC (nominal 24VDC) (Controller stoppt)
	E2	Kein oder falsches Treiberboard installiert (Controller stoppt)
	E3	Versorgungsspannung größer 30VDC (Controller stoppt)
	E4	Übertemperatur hat ausgelöst (Ausgänge werden abgeschaltet)



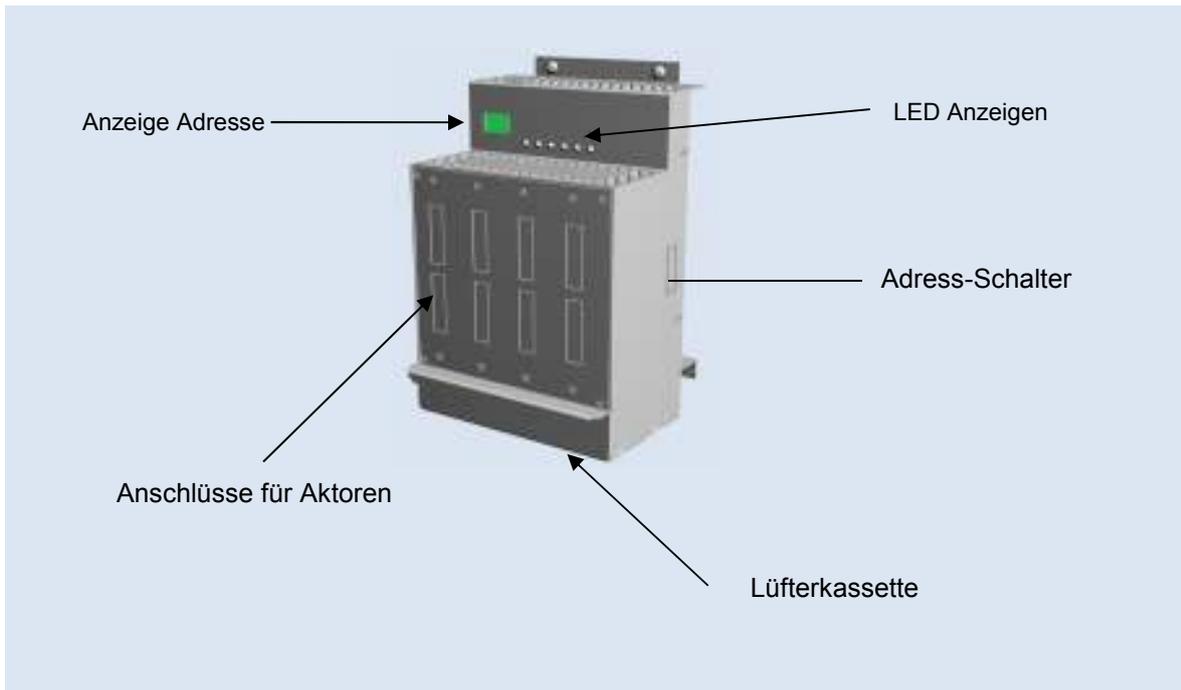


Abbildung 2 (Schematische Darstellung)

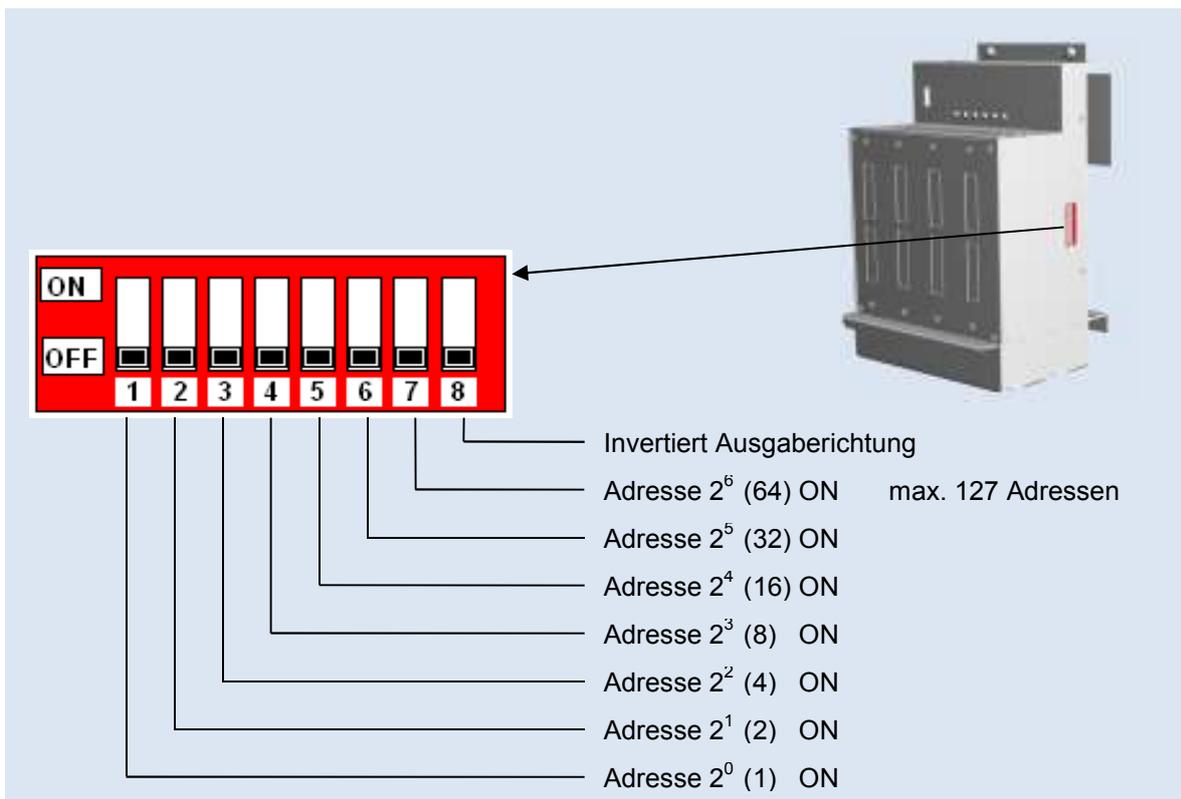


Abbildung 3 (Adress-Schalter Einstellung)

Die eingestellte Adresse wird durch eine Siebensegmentanzeige oben links angezeigt.

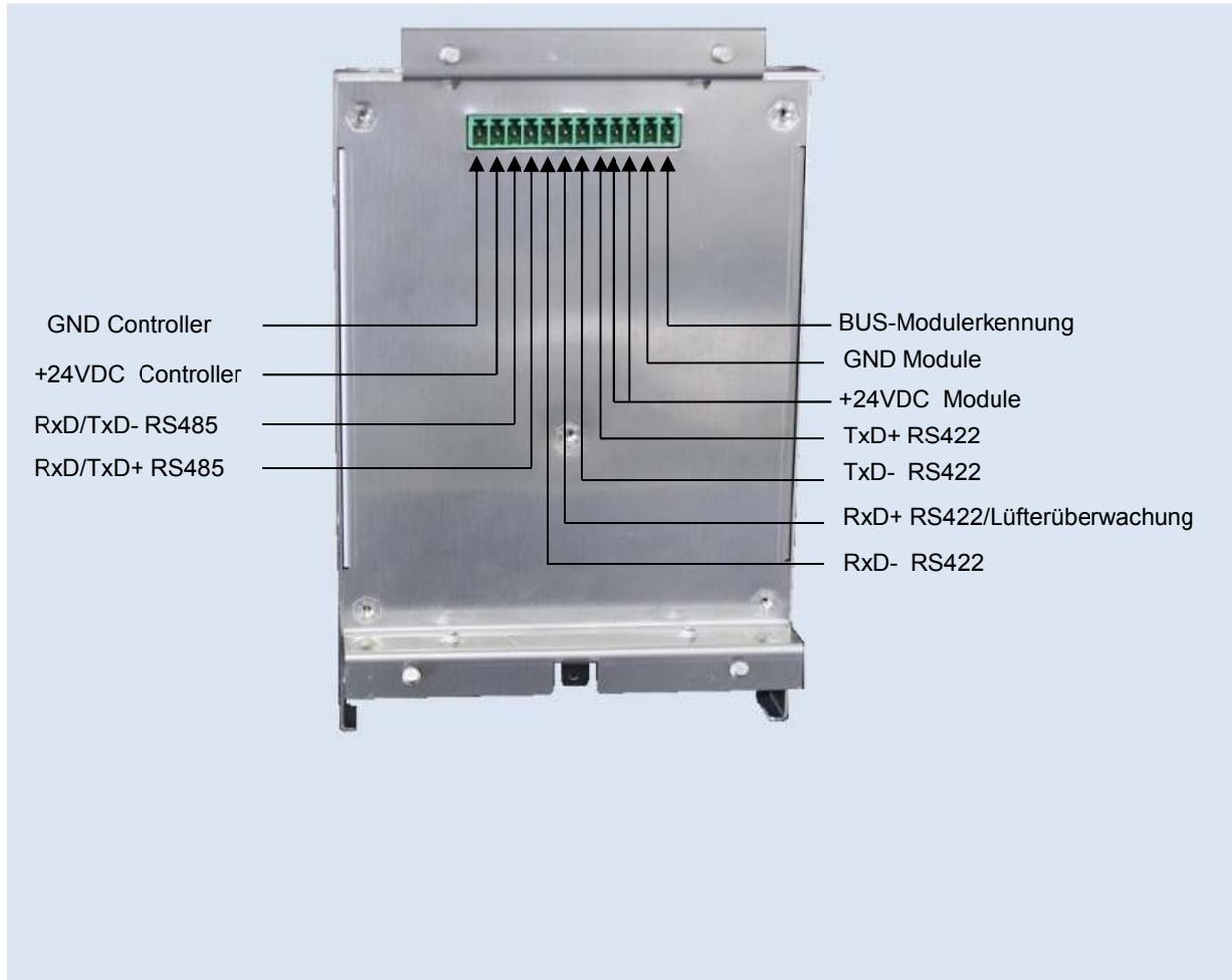


Abbildung 4 (Anschlussbelegung Controller)

Der JDA48-4 wird mit folgender Standardeinstellung und möglichen Standardreferenzen ausgeliefert (Anzahl 0XXX u. 1XXX schliessen sich gegeneinander aus):

Tabelle 2 (Standardwerte bei Auslieferung)

Adresse	1	Einstellung Adress-Schalter
Baudrate	19200	Per Diagnosekommando veränderbar
Datenlänge	8 Bit	Per Diagnosekommando veränderbar
Stopbit	1 Bit	Per Diagnosekommando veränderbar
Parität	Even	Per Diagnosekommando veränderbar
I/O Referenzen		
0XXXX	48	Coil Ausgänge
1XXXX	32	Diskrete Eingänge
3XXXX	4	Eingangsregister
4XXXX	300	Holdingregister

Bei vier analogen Ausgangsmodulen sind keine 0XXX und 1XXX Referenzen verfügbar.

3.0 Konfigurationsliste verfügbarer E/A Module

Die Konfigurationsliste gibt eine Übersicht derzeitig verfügbarer E/A Module und wird ständig erweitert. Jedem Modul ist eine Kennung zugeordnet, die Bezug nimmt auf das entsprechende Holdingregister. Der Modultyp wird in den Registern 40200 bis 40203 abgebildet. Der JDA48-4 erkennt nach dem Start das entsprechende Modul und trägt dies automatisch im relevanten Register ein.

Kennung Register	Modulbez.	Anz E/A	Beschreibung
00	---	---	Kein Modul auf diesem Steckplatz erkannt
01	JDA48-4-D12L	12	Digitales Ausgangsmodul 2x6 oder 2x4Bit 240mA 24VDC Low-Side
02	Res.		
03	JDA48-4-D12S-AC	12	Digitales Ausgangsmodul 3x4Bit Solidstate, 1000mA AC
04	JDA48-4-D12S-DC	12	Digitales Ausgangsmodul 3x4Bit Solidstate, 500mA DC kurzschlussfest
05	Res.		
06	Res.		
07	JDA48-4-A12L	12	Analoges Ausgangsmodul 12fach 0/4-20mA, low-side schaltend
08	Res.		
09	Res.		
11	Res.		
12	Res.		
13	JDA48-4-E8-DC	8	Digitales Eingangsmodul 8fach DC, 5-40V aktiv high
14	Res.		
15	Res.		
16	Res.		
17	Res.		
18	Res.		

Tabelle 3 (Modulkonfigurationsliste)

Maximal 4 Module können in einem JDA48-4 installiert werden.

4.0 Übersicht der Holdingregister 4XXXX Referenz (16Bit Werte)

Reg-Nr von	Reg-Nr bis	Anz. 16Bit Reg.	R/W Write	Bezeichnung	Wert
0001	0048	48	W	Analoge Ausgangswerte (0 – 20mA) pro Ausgang	0 – 10000
0050	0057	8	W	8 x dig. Ausgangswerte 4Bit	0 – 10000
0058	0061	4	-	Reserve	
0062	0069	8	W	8 x dig. Ausgangswerte 6Bit	0 – 10000
0070	0117	48	R	Aktueller Stromwert pro Analogausgang	In mA dez.
0118	0125	8	R	Status dig/analoger Ausgänge 1 – 48 max.	01h=Drahtbruch 02h=Kurzschluss
0199		1	R	Lüfterstatus	0h = Lüfter OK 01h = Lüfterfehler
0200		1	R/W	Modultyp und Gruppierung Modul1	
0201		1	R/W	Modultyp und Gruppierung Modul2	
0202		1	R/W	Modultyp und Gruppierung Modul3	
0203		1	R/W	Modultyp und Gruppierung Modul4	
0204		1	R/W	Dynamische Änderung der Ausgabe bei 6Bit Modulen (Gruppenzuordnung)	0 = 1- 6 1 = 6 - 1
0205		1	R/W	Analogausgänge von 0 oder 4mA	0 = 0mA 1 = 4mA
0207		1	R	Max. Anzahl 0XXX Referenzen	12 – 48
0208		1	R	Max. Anzahl analoger Ausgänge	12 – 48
0209		1	R	Max. Anzahl diskreter Eingänge 1XXXX	8 – 32
0210		1	R	Anzahl Eingangsregister 3XXXX	4
0211		1	R	Anzahl Holdingregister 4XXXX	300
0212		1	R/W	Drahtbruchererkennung AUS/AN generell	0/1
0213	0216	4	R/W	Je Modul 12Bit Map Drahtbruch AUS/AN	Bit 0/1 pro Ausgang
0220		1	R	Slaveadresse	1 – 63
0221		1	R	CPU Version	1.00 – X.XX
0222		1	R	Wert Versorgungsspannung der BGR	12.00 – 24.XX
0223		1	R	Temperatur der BGR	20.00 – 70.00
0224		1	R	BGR Status	3 = standard
0225		1	R	BGR Seriennummer	nnnn
0226		1	R/(W)	RS485 Speed	01=4800Bd, 02=9600Bd 03=19200Bd, 04=38400Bd
0227		1	R/(W)	RS485 Datenlänge	01=7Bit, 02=8Bit
0228		1	R/(W)	RS485 Stopbit	01=1 Stopbit, 02=2 Stopbit
0229		1	R/(W)	RS485 Parity	01 = even, 02 = odd 03 = none

Schreiboperationen in () können nicht durch die Befehle Preset Single/Multiple Register durchgeführt werden.

Reg-Nr von	Reg-Nr bis	Anz. 16Bit Reg.	R/W	Bezeichnung	Wert
0230		1	R/(W)	RS422 Speed (Extension BUS)	01=4800Bd, 02=9600Bd 03=19200Bd 04=38400Bd 05=250.000Bd
0231		1	R/(W)	RS422 Datenlänge	01=7Bit, 02=8Bit
0232		1	R/(W)	RS422 Stopbit	01=1 Stopbit, 02=2 Stopbit
0233		1	R/(W)	RS422 Parity	01 = even, 02 = odd 03 = none
0234		1	R	BGR Mode	1=RTU
0235		1	R	Zähler Reset BGR seit Inbetriebnahme Nach jedem Einschalten Zähler (nnnn) +1	0 – 65535 – 0
0236		1	R	Zähler CRC-Fehler nach Einschalten der BGR Nach Einschalten Start mit 0	0 – 65535 – 0
0237		1	R	Telegrammzähler nach Einschalten der BGR Nach Einschalten Start mit 0	0 – 65535 – 0
0238		1	R	Heart-Beat Zähler nach Einschalten der BGR Wird im 500ms Takt erhöht Nach Einschalten Start mit 0	0 – 65535 – 0
0245		1	R/W	Low Register Mapping (INP-Mod.1) 1-4 zeigt auf Register 1 Bit(n)	00-FF
0246		1	R/W	High Register für Mapping (INP-Mod.1) 5-8 zeigt auf Register 1	00-FF
0247		1	R/W	Low Register Mapping (INP-Mod.2) 1-4 zeigt auf Register 2 Bit(n)	00-FF
0248		1	R/W	High Register für Mapping (INP-Mod.2) 5-8 zeigt auf Register 2	00-FF
0249		1	R/W	Low Register Mapping (INP-Mod.3) 1-4 zeigt auf Register 3 Bit(n)	00-FF
0250		1	R/W	Low Register Mapping (INP-Mod.3) 5-8 zeigt auf Register 3 Bit(n)	00-FF
0251		1	R/W	Low Register Mapping (INP-Mod.4) 1-4 zeigt auf Register 4 Bit(n)	00-FF
0252		1	R/W	Low Register Mapping (INP-Mod.4) 5-8 zeigt auf Register 4 Bit(n)	00-FF
0253		1	R	Statusregister Input_Modul1	
0254		1	R	Statusregister Input_Modul2	
0255		1	R	Statusregister Input_Modul3	
0256		1	R	Statusregister Input_Modul4	
0257	0260	4	R/W	Code Register Modul1 – Modul4	0/1 pro Eingangsbit
0261		1	R/W	Freigaberegister für Eingabemodule	01h pro Kanal und Modul
0262		1	R/W	WDT-Zeitregister in ms (Kommunikationsüberwachung)	1000 – 65000ms
0263		1	R/W	WDT-Code und Status 0051h – 005F = Modul1-4 bzw. alle Bei Auslösen Kommando im oberen Byte	0051-005F
0269	0300	32	R/W	32 User Register 16Bit zur freien Verfügung	
301	304	4	W	Binäre Ausgabe für max. 4 Ausgabemodule	0 – 4095

5.0 Erklärung einzelner Registerfunktionen

Register **Read/Write** **Beschreibung**

40001 - 48 **W** Jedem Register ist ein Analogausgang zugeordnet. Durch Laden eines Registers mit einem Wert zwischen 0 -10000 wird der relevante Analogausgang zwischen 0/4 – 20mA angesteuert.

Eingabewert z.B. Register **40001**

High-Byte Register								Low-Byte Register							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	1	1	1	0	0	0	1	1	0	1	0

Register Inhalt: **2710_h = 10000_{dez} = 20mA**
 Registerwert gültig: **während Laufzeit**
 MODBUS Befehl: **06_h, 10_h**

Siehe Register 205 = 0/4mA Startberechnung ([Link](#))

40050 - 61 **W** Jedem Register ist ein 4Bit Digitalausgang zugeordnet. Die Register sind nur relevant, wenn die Gruppierung im relevanten Gruppenregister (200-203) für das zugehörige Modul auf 1 gesetzt wurde. Ist die Gruppierung auf 1 gesetzt, so wird ein Registerwert zwischen 0 – 10000 auf einen 4Bit Ausgang umgesetzt. (10000_{dez} = 0x0F_h)

Eingabewert z.B. Register **40050**

High-Byte Register								Low-Byte Register							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	1	1	1	0	0	0	1	1	0	1	0

Register Inhalt: **2710_h = 10000_{dez} = 100% = 4Bit an = 0x0F_h**
 Registerwert gültig: **während Laufzeit**
 MODBUS Befehl: **06_h, 10_h**

Zugehörige Register 200-203 = Gruppierung **40200**

High-Byte Modulgruppierung								Low-Byte Modultyp							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	0	1	R	R	R	R	R	R	R	R

Register Read/Write Beschreibung

40062 – 69

W

Jedem Register ist ein 6Bit Digitalausgang zugeordnet. Die Register sind nur relevant, wenn die Gruppierung im relevanten Gruppenregister (200-203) für das zugehörige Modul auf 2 gesetzt wurde. Ist die Gruppierung auf 2 gesetzt, so wird ein Registerwert zwischen 0 – 10000 auf einen 6Bit Ausgang umgesetzt. (10000_{dez} = 0x3F_h)

Eingabewert z.B. Register **40062**

High-Byte Register								Low-Byte Register							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	1	1	1	0	0	0	1	1	0	1	0

Register Inhalt = **2710_h** = **10000_{dez}** = 100% = 4Bit an = **0x3F_h**

Registerwert gültig: **während Laufzeit**

MODBUS Befehl: **06_h, 10_h**

Zugehörige Register 200-203 = Gruppierung **40200**

High-Byte Modulgruppierung								Low-Byte Modultyp							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	1	0	R	R	R	R	R	R	R	R

40070 – 117

R

Aktueller Stromwert eines Ausgangs umgerechnet auf Dezimalen Wert/10. (z.B. 20mA = C8). Bei Strommessung eines digitalen Ausgangs ist hier ebenfalls der Stromwert zu finden. Register 40070 ist erster Ausgang am ersten Modul.

Ausgabewert z.B. Register **40070**

High-Byte Register								Low-Byte Register							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1

Register Inhalt: **C8_h** = **200_{dez}/10** = 20mA Analogmodul

Register Inhalt: **7D0_h** = **2000_{dez}/10** = 200mA Digitalmodul

Registerwert gültig: **während Laufzeit**

MODBUS Befehl: **03_h**

Register Read/Write Beschreibung

40118 – 125 **R** Ein Statusregister pro 6 Ausgänge. Eine **01_h** bedeutet Drahtbruch, eine **02_h** Überstrom an diesem Ausgang. Ausgangsmodule, die diese Funktion nicht bieten, belegen **00_h** pro Ausgang. Die Information steht nur zur Verfügung, wenn die relevanten Register 212 – 216 mit den entsprechenden Werten geladen wurden.

Ausgänge zu Statusinformation: (**Abfragen**
Drahtbruch/Überstrom)

R: 118					A6		A5		A4		A3		A2		A1	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wert 01/10	x	x	x	x	0	0	0	0	0	0	0	0	0	0	0	0
R: 119					A12		A11		A10		A9		A8		A7	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wert 01/10	x	x	x	x	0	0	0	0	0	0	0	0	0	0	0	0
R: 120					A18		A17		A16		A15		A14		A13	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wert 01/10	x	x	x	x	0	0	0	0	0	0	0	0	0	0	0	0
R: 121					A24		A23		A22		A21		A20		A19	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wert 01/10	x	x	x	x	0	0	0	0	0	0	0	0	0	0	0	0
R: 122					A30		A29		A28		A27		A26		A25	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wert 01/10	x	x	x	x	0	0	0	0	0	0	0	0	0	0	0	0
R: 123					A36		A35		A34		A33		A32		A31	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wert 01/10	x	x	x	x	0	0	0	0	0	0	0	0	0	0	0	0
R: 124					A42		A41		A40		A39		A38		A37	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wert 01/10	x	x	x	x	0	0	0	0	0	0	0	0	0	0	0	0
R: 125					A48		A47		A46		A45		A44		A43	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wert 01/10	x	x	x	x	0	0	0	0	0	0	0	0	0	0	0	0

Registerwert gültig: **während Laufzeit**
MODBUS Befehl: **03_h**

Beispiel: Drahtbruch an Ausgang **A2** = Register **118**, Wert: **0004_h**
Überstrom an Ausgang **A7** = Register **119**, Wert: **0002_h**

Zugehöriges Register **40212** = Drahtbruch generell AUS=0/AN=1

High-Byte Register								Low-Byte							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0/1

Zugehörige Register **40213 – 216** = 12Bit pro Modul 0/1=Aus/An

High-Byte Register								Low-Byte							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Register	Read/Write	Beschreibung
----------	------------	--------------

40126 – 198 **R** **Reservierte Register für zukünftige Funktionen**

40199 - **R** **Lüfterstatus des Controllers**

Ausgabewert Register **40199**

High-Byte Register								Low-Byte Register							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0/1	0/1	0	0/1	0/1

Register Inhalt: **0000_h** = kein Lüfter installiert
0001_h = Lüfter installiert und **OK**
0003_h = Lüfter installiert und **Fehler**

Zusätzlich Bit3 = Drahtbruch vorhanden, Bit4 = Überstrom vorhanden

Registerwert gültig: **während Laufzeit**
 MODBUS Befehl: **03_h**

40200-203 **R/W** **Modultyp und Gruppierung Modul1- 4**

Registerrückmeldung:

High-Byte Modulgruppierung								Low-Byte Modultyp							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R	R	R	R	R	W	W	R	R	R	R	R	R	R	R

Modultyp (wird von der Controller selbst erkannt):

Siehe Tabelle 3 (Modulkonfigurationsliste)

Modulgruppierung (nur für digitales Ausgangsmodul **Bit 8+9**)

01 = 3 x 4 Bit digital Daten aus Reg. 4xxxx werden auf 4Bit gemappt
 01 = 2 x 4 Bit digital (bei Modultyp 01)
 02 = 2 x 6 Bit digital Daten aus Reg. 4xxxx werden auf 6Bit gemappt

Modulreihenfolge (00 = 1. Modul eines Typs) = Bit 12 + 13

Daten werden von der BGR festgelegt.

z.B.

00 = 1. Modul, Typ 03 (Reg. 40200), 01 = 2. Modul, Typ 03 (Reg. 40201)

00 = 1. Modul, Typ 13 (Reg. 40202), 01 = 2. Modul, Typ 13 (Reg. 40203)

Diese Werte können nicht extern beeinflusst werden.

Registerwerte gültig: **permanent**
 MODBUS Befehl: **03_h, 06_h, 10_h**

Aufteilung Gruppe zu Registern (digitale Ausgänge)

Modul	3 x 4Bit	2 x 4/6Bit
1	40050 – 40052	40062 – 40063
2	40053 – 40055	40064 – 40065
3	40056 – 40058	40066 - 40067
4	40059 – 40061	40068 - 40069

Register Read/Write Beschreibung

40204 - R/W Ausgaberrichtung der Bits am Modul (normal/reverse)

High-Byte Register (204)								Low-Byte Register (204)							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1

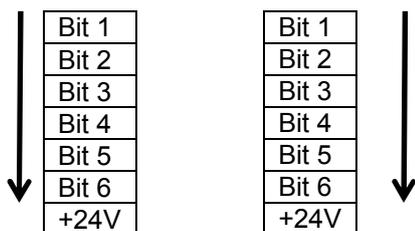
Durch die Einstellung des DIP-Schalters (letzte Position) kann grundsätzlich die Ausgaberrichtung eingestellt werden. Diese wird bei Neustart des Controllers in Abhängigkeit des Schalters ins das Register 204 geladen.

Zur Laufzeit kann das Register überschrieben werden.
 Bit 0 = 0 Ausgabe der Datenreihenfolge von oben nach unten.
 Bit 0 = 1 Ausgabe der Datenreihenfolge von unten nach oben.

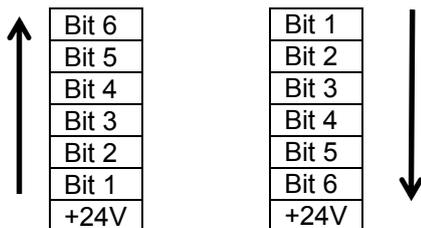
Die Abfrage oder Zuordnung der Drahtbruchfunktionen geschieht in der Zählweise immer von oben nach unten. ! Darauf ist bei der Programmierung der Drahtbruchbits unbedingt zu achten !

Beispiel 6Bit Ausgangsmodul

Normal Drahtbruchmessung/Statusabfrage



Reverse Drahtbruchmessung/Statusabfrage



Register Read/Write Beschreibung

40212 R/W Drahtbruchüberprüfung generell AUS=0/AN=1

High-Byte Register (212)								Low-Byte Register (212)							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1

Register Inhalt: **0000_h** = Drahtbruchererkennung generell **aus**
0001_h = Drahtbruchererkennung generell **an**

Registerwert gültig: **permanent**
 MODBUS Befehl: **03_h, 06_h, 10_h**

Wird hier die Drahtbruchererkennung generell ausgeschaltet, ist der Inhalt der Register 40118 - 40125 und 40213 - 40216 irrelevant. Die Anzeige BRK/OC am JDA48-4 ist deaktiviert.

Drahtbruchüberprüfung pro Ausgang

40213 R/W 12 Bit Map ->Modul1. Bit gesetzt (1) bedeutet aktiv.

High-Byte (213)								Low-Byte (213)							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Registerwert gültig: **permanent** MODBUS Befehl: **03_h, 06_h, 10_h**

40214 R/W 12 Bit Map ->Modul2. Bit gesetzt (1) bedeutet aktiv.

High-Byte (214)								Low-Byte (214)							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Registerwert gültig: **permanent** MODBUS Befehl: **03_h, 06_h, 10_h**

40215 R/W 12 Bit Map ->Modul3. Bit gesetzt (1) bedeutet aktiv.

High-Byte (215)								Low-Byte (215)							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Registerwert gültig: **permanent** MODBUS Befehl: **03_h, 06_h, 10_h**

40216 R/W 12 Bit Map ->Modul4. Bit gesetzt (1) bedeutet aktiv.

High-Byte (216)								Low-Byte (216)							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Registerwert gültig: **permanent** MODBUS Befehl: **03_h, 06_h, 10_h**

Register	Read/Write	Beschreibung
----------	------------	--------------

40221 R CPU-Version

Beispiel: **Version 1.51**

High-Byte								Low-Byte							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1

Register Inhalt: **0151_h = 1.51**

Registerwert gültig: **permanent**

MODBUS Befehl: **03_h**

40222 R Spannungswert (Versorgungssp.)

Beispiel: **24,23V**

High-Byte								Low-Byte							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	1	0	0	0	0	1	0	0	0	1	1

Register Inhalt: **2423_h = 24,23V**

Registerwert gültig: **während Laufzeit**

MODBUS Befehl: **03_h**

40223 R BGR-Temperatur (nur ganze Grad °C)

Beispiel: **26,00°C**

High-Byte								Low-Byte							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0

Register Inhalt: **2600_h = 26,00°C**

Registerwert gültig: **während Laufzeit**

MODBUS Befehl: **03_h**

Gemessen wird in den Grenzen 20°C – 70°C

Ab 70°C herrscht eine Fehlersituation und das betreffende Modul wird im Fall von Digital/Analogausgaben abgeschaltet. In jedem Fall leuchtet dann die **LED OvTemp**.

6.0 Zuordnung Eingangsmodul zu Register

Werden im JDA48-4 digitale Eingabemodule benutzt, so können diese mit dem Befehl **READ INPUT STATUS** aus dem vorhandenen Inputstatuspuffer ausgelesen werden. Eine weitere Möglichkeit ist die Zuweisung einer Eingangsbaugruppe mit 8 Eingängen zu einem spezifizierten Statusregister. Dadurch ergeben sich weitere Konfigurationsmöglichkeiten. Über ein Zusatzregister kann ein Eingang deaktiviert und mit Hilfe eines Mapping Registers an eine ganz bestimmte Bitposition eines zugeordneten Statusregisters zugewiesen werden. Dadurch ist die physikalische Zuordnung zur erwarteten Bitposition im Statusregister zum übergeordneten Leitsystem möglich.

Registerübersicht

40245-246	R/W	Register Mapping (Low/High) für 1. Eingangsmodul (P)
40247-248	R/W	Register Mapping (Low/High) für 2. Eingangsmodul (P)
40249-250	R/W	Register Mapping (Low/High) für 3. Eingangsmodul (P)
40251-252	R/W	Register Mapping (Low/High) für 4. Eingangsmodul (P)
40253	R	Status Register (16Bit) für 1. Eingangsmodul (L)
40254	R	Status Register (16Bit) für 2. Eingangsmodul (L)
40255	R	Status Register (16Bit) für 3. Eingangsmodul (L)
40256	R	Status Register (16Bit) für 4. Eingangsmodul (L)
40257	R/W	Code Register Modul1 (Bit 0=inaktiv, 1=aktiv) (P)
40258	R/W	Code Register Modul2 (Bit 0=inaktiv, 1=aktiv) (P)
40259	R/W	Code Register Modul3 (Bit 0=inaktiv, 1=aktiv) (P)
40260	R/W	Code Register Modul4 (Bit 0=inaktiv, 1=aktiv) (P)
40261	R/W	Freigaberegister für die Zuordnung (P)

(P = Programmierung ist permant)
(L = nur gültige Werte zur Laufzeit)

Register	Read/Write	Beschreibung
----------	------------	--------------

40245-246	R/W	Register Mapping (High/Low) für 1. Eingangsmodul
-----------	-----	--

Jedes Halbbyte zeigt auf eine Position im zugehörigen Statusregister. Dadurch kann jeder Eingang 1-8 eines Moduls einer Bitposition im Statusregister zugewiesen werden.

Register 245/246

High-Byte								Low-Byte							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Zugehöriges Statusregister = **40253**

Modul JDA48-4-E8-DC

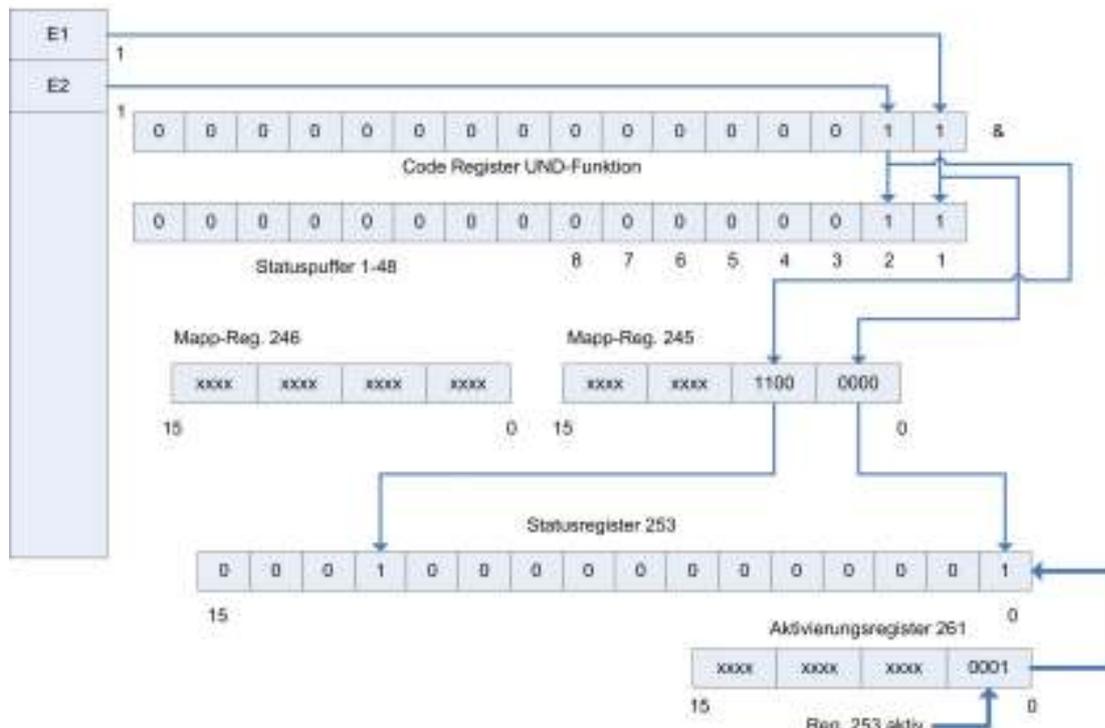


Abbildung 5 (Zuordnung Eingang zu Statusregister)

Beispiel :

Vom Eingabemodul1 soll der erste und zweite Eingang auf Bitposition 0 und 12 des Statusregisters gemappt werden.

Dadurch ist das Auslesen eines systemspezifischen Statuswortes möglich.

Über den MODBUS Befehl READ INPUT Status ist nach wie vor das Einlesen der Eingangszustände der 1xxxx Referenzen möglich.

Werden die MAPP-Register programmiert, so dürfen die Dateninhalte nicht redundant sein! ! Durch Einschreiben von 0-Werten in die MAPP-Register, wird die Zuordnung gelöscht!

1. Code Register Modul1 (**257**) = **0003_h**
Damit werden die beiden Eingänge (E1+E2) auf den Puffer und die Register freigeschaltet.
2. Mapp Register Modul1(**245+246**) = **00C0_h + 0000_h**
Im Mapp-Register 245 steht an Position 0 (E1) und 1 (E2) eine 4Bitadresse, die auf die entsprechende Bitposition im Statusregister zeigt.
3. Freigaberegister (**261**) = **0001_h**
Eine „1“ an der niedrigsten Bitposition des Registers gibt das Statusregister 253 frei. Wäre hier „11“ eingetragen wäre auch das Register 254 freigegeben.

Wird jetzt Eingang 1+2 aktiv, ist der Wert des ausgelesenen Statusregisters (**253**) = **1001_h**

7.0 Kommunikationsüberwachung

In bestimmten Anwendungen ist es erforderlich, bei Ausfall der Datenverbindung einen sicheren Ausgangszustand herzustellen. Das geschieht durch Abschalten der Ausgangssignale. Für diesen Anwendungsfall gibt es zwei Register, mit deren Hilfe dies geschehen kann. Einmal ein Register, das die Zeit bis zur Abschaltung enthält und zum anderen ein Register, welches festlegt, welches Ausgangsmodul in diesem Fall abgeschaltet wird und ob die Abschaltung aktiviert ist.

Register	Read/Write	Beschreibung
40262	R/W	Zeiteinheit in ms für Kommunikations-Watchdog Wert 1000 – 65000ms = 1s – 65s
40263	R/W	Der niederwertige Teil des Registers beinhaltet einen Code, der den Watchdog aktiviert. 51 _h = Modul 1 wird abgeschaltet - danach in 15 Schritten bis 5F _h = alle Module abschalten. Wird das Register auf „0“ gesetzt, ist die Funktion deaktiviert

Code für Modulabschaltung bei Kommunikationswatchdog:

Code	Modul1	Modul2	Modul3	Modul4
00 _H	Funktion ist/wird deaktiviert			
51 _H	●	---	---	---
52 _H	---	●	---	---
53 _H	●	●	---	---
54 _H	---	---	●	---
55 _H	●	---	●	---
56 _H	---	●	●	---
57 _H	●	●	●	---
58 _H	---	---	---	●
59 _H	●	---	---	●
5A _H	---	●	---	●
5B _H	●	●	---	●
5C _H	---	---	●	●
5D _H	●	---	●	●
5E _H	---	●	●	●
5F _H	●	●	●	●

Tabelle 4 (Code3 für Modulabschaltung)

Bei Eintreten der Auslösung ist das Kommando im oberen Byte des Registers als Status auslesbar. Bei Eintreffen des ersten Telegrammes wird der Watchdog auf den programmierten Wert zurückgesetzt und der obere Teil des Aktivierungsregisters auf 0 gesetzt. Der Kommunikationswatchdog darf nicht mit dem internen Watchdog verwechselt werden, der nur bei Hangup der Software auslöst und den kompletten Controller zurücksetzt.

Beispiel: Zeit Watchdog = 5s, Im Fehlerfall Modul 1 + 3 abschalten

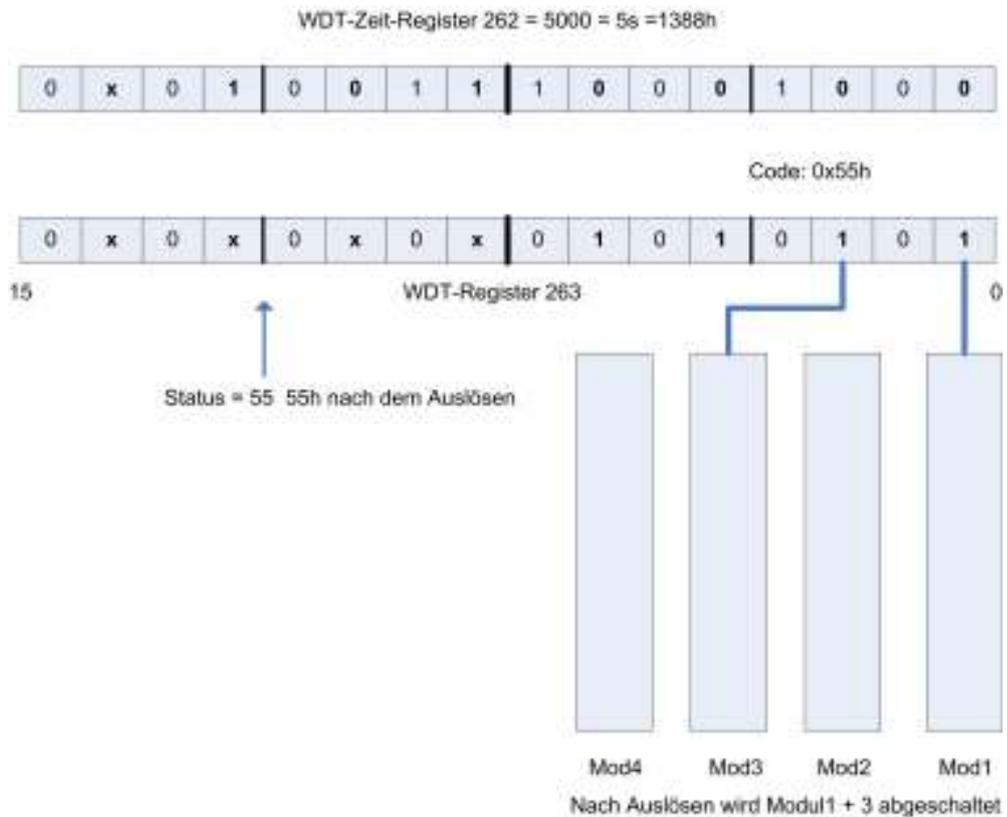


Abbildung 6 (Beispiel Watchdog Programmierung)

Nach dem Programmieren der Zeit wird der Watchdog nach dem Senden des Funktionscodes aktiv.

Ein ausgelöster Watchdog wird durch den Fehlercode „CE“ am Display angezeigt.

8.0 User Register

Register	Read/Write	Beschreibung
40269 - 40300	R/W	32 freie 16Bit Register für den Benutzer Die Werte dieser Register werden nichtflüchtig gespeichert und sind nach Neustart des Controllers verfügbar.

9.0 Register Erweiterungen (nur digitale Ausgänge)

Diese Register sind ebenfalls Holdingregister, die außerhalb der normalen Register liegen und nur dynamisch genutzt werden können.

Register	Read/Write	Beschreibung
40301 – 40304	W	<p>Je Register 12 Bit. Jedes Register ist einem Ausgabemodul zugeordnet. Durch Setzen/Löschen eines Bits kann ein dazugehöriger Ausgang beeinflusst werden. Die Register können mit PRESET SINGLE oder PRESET MULTIPLE gesetzt werden. Die Werte sind nach dem Ausschalten des Controllers nicht mehr verfügbar.</p> <p>Die Registerinhalte werden ohne interne Verarbeitung an die Ausgangsmodule übertragen. Die Reihenfolge der aktivierten Ausgänge zu den Registern ist durch die Hardwareeigenschaften des relevanten Moduls festgelegt. Das erste Register verhält sich variabel zum ersten gefundenen Ausgabemodul. Ist das 1. Ausgabemodul z.B. auf dem 2. Steckplatz des Controllers installiert, so ist dennoch das Register 301 dafür relevant. Die steuernde Software übernimmt die logische Kontrolle der anzusteuernenden Ausgänge.</p>

10.0 Unterstützte MODBUS Befehle

Code	MODBUS Funktion	Reg.	Anwendung	Broadcast
01 _H	READ COIL Status	0xxxx	- Auslesen von Digitalausgangs-Zuständen	nein
02 _H	READ INPUT Status	1xxxx	- Auslesen von Digitaleingangs-Zuständen	nein
03 _H	READ HOLDING REGISTER	4xxxx	- Auslesen von Messwerten, Zählerständen, Mittelwerten, - Auslesen der Geräte-Konfiguration	nein
04 _H	READ INPUT REGISTER	3xxxx	- Auslesen von binären Registerinhalten H/L	nein
05 _H	FORCE SINGLE COIL	0xxxx	- Setzen eines bestimmten Digitalausgangs	Ja
06 _H	PRESET SINGLE REGISTER	4xxxx	- Setzen eines Halteregisters	Ja
08 _H	DIAGNOSTIC		- Geräte-Verbindungstest (Subfunktion 0)	nein
0F _H	FORCE MULTIPLE COILS	0xxxx	- Setzen / Simulieren von Digitalausgangs-Zuständen	Ja
10 _H	PRESET MULTIPLE REGISTERS	4xxxx	- Setzen mehrerer Halteregister	Ja
11 _H	REPORT SLAVE ID	4xxxx	- komplette Slavestatusmeldung (wird separat definiert)	nein
40H	PROG COM (*)	4xxxx	Änderung der Schnittstellenparameter	ja
41H	COM RESET (*)	---	Restart Kommunikationsmodul	ja
42H	INIT REGISTER (*)	4xxxx	Autom. Initialisierung eines Registersatzes	ja

Tabelle 5 (MODBUS Befehle) (*) nicht MODBUS konforme Kommandos

01H READ COIL STATUS

Tabelle 6: 01H READ COIL STATUS Anforderung MA-SL

Adresse	Funktion	Daten				CRC-Check	
		Startadresse		Anzahl Zustände		Crc16	
Adr	01H	High-Byte	Low-Byte	High-Byte	Low-Byte	Low-Byte	High-Byte

Antwort von Slave:

Adresse	Funktion	Daten			CRC-Check
		Anzahl Datenbytes	Zustand 8-1	Zustand 16 - 9	
Adr	01H	8 Bit	8 Bit	8 Bit	Crc16

Wurde die Drahtbruchererkennung aktiviert, so wird bei Drahtbruchererkennung der Status auf 0 gesetzt obwohl der Ausgang aktiviert wurde.

Beispiel1: (Lesen von 8 Zuständen ab Adresse 0, Slaveadresse = 1)

Master -> Slave: 01 01 00 00 00 08 Crc16
 Slave -> Master: 01 01 01 Zustand8-1 Crc16 1 = niederwertigsten Bit

Beispiel2: (Lesen von 18 Zuständen ab Adresse 0, Slaveadresse = 1)

Master -> Slave: 01 01 00 00 00 12 Crc16
 Slave -> Master: 01 01 03 Zustand8-1 Zustand16-9 Zustand24-17 Crc16

02H READ INPUT STATUS

Lesen der (digital) Eingangszustände 0 = AUS, 1 = EIN

Tabelle 7: 02H READ INPUT STATUS Anforderung MA-SL

Adresse	Funktion	Daten				CRC-Check
		Startadresse		Anzahl Zustände		
Adr	02H	High-Byte	Low-Byte	High-Byte	Low-Byte	Crc16

Antwort von Slave:

Adresse	Funktion	Daten			CRC-Check
		Anzahl Datenbytes	Zustand 8-1	Zustand 16 - 9	
Adr	02H	8 Bit	8 Bit	8 Bit	Crc16

Beispiel1: (Lesen von 8 Zuständen ab Adresse 0, Slaveadresse = 1)

Master -> Slave: 01 02 00 00 00 08 Crc16
 Slave -> Master: 01 02 01 Zustand8-1 Crc16

03H READ HOLDING REGISTER(S)

Tabelle 8: 03H READ HOLDING REGISTER Anforderung MA-SL

Adresse	Funktion	Daten				CRC-Check
		Startadresse Reg.		Anzahl Reg.(incl.Start)		
Adr	03H	High-Byte	Low-Byte	High-Byte	Low-Byte	Crc16

Antwort von Slave:

Adresse	Funktion	Daten		CRC-Check
		Anzahl Datenbytes	Information	
Adr	03H	n (8 Bit)	n/2 Register	Crc16

Es können maximal 125 Register gelesen werden. Registeradressierungen sind immer relativ zur angegebenen Referenznummer. Register 1 = 0, Register 125 = 124.

Beispiel: (Lesen von 2 Registern ab Register 268, Slaveadresse = 1)

Register $268_{dez} = 10C_h - 1 = 10B_h$ (MODBUS Register werden immer -1 adressiert)

Master -> Slave: 01 03 01 0B 00 02 Crc16
 Slave -> Master: 01 03 04 Reg268_H Reg268_L Reg269_H Reg269_L Crc16

04H READ INPUT REGISTER(S)

Tabelle 9: 04H READ INPUT REGISTER Anforderung MA-SL

Adresse	Funktion	Daten				CRC-Check
		Startadresse Reg.		Anzahl Reg.(incl.Start)		
Adr	04H	High-Byte	Low-Byte	High-Byte	Low-Byte	Crc16

Antwort von Slave:

Adresse	Funktion	Daten		CRC-Check
		Anzahl Datenbytes	Information	
Adr	04H	n (8 Bit)	n/2 Register	Crc16

Es können maximal 125 Register gelesen werden. Registeradressierungen sind immer relativ zur angegebenen Referenznummer. Register 1 = 0, Register 125 = 124.

Beispiel: (Lesen von 2 Registern ab Register 1, Slaveadresse = 1)

Register $1_{dez} = 1_h - 1 = 0_h$ (MODBUS Register werden immer -1 adressiert)

Master -> Slave: 01 04 00 00 00 02 Crc16

Slave -> Master: 01 04 04 Reg1_H Reg1_L Reg2_H Reg2_L Crc16

Der JDA48-4 verfügt über 4 (16Bit) INPUT Register.

05H FORCE SINGLE COIL

1 x Spule Ein = (FF00) // Spule Aus = (00 00)

Tabelle 10: 05H FORCE SINGLE COIL Anforderung MA-SL

Adresse	Funktion	Daten				CRC-Check
		Startadr High	Startadr Low	Data Hi	Data Lo	
Adr	05H	00	XX	FF	00	Crc16

Der Slave schickt das gleiche Telegramm als Echo zurück.

Dieses Kommando kann auch im Broadcast Modus ausgeführt werden. Dadurch können Ausgänge auf allen Slaves gleichzeitig verändert werden. In diesem Fall gibt es keine Antwort von einem Slave.

Beispiel1: (Setzen von Ausgang 3, Slaveadresse = 1)

Master -> Slave: 01 05 00 02 FF 00 Crc16

Slave -> Master: 01 05 00 02 FF 00 Crc16

06H PRESET SINGLE REGISTER

Tabelle 11: 06H PRESET SINGLE REGISTER Anforderung MA-SL

Adresse	Funktion	Daten				CRC-Check
		Register Adresse		Daten		
Adr	06H	High-Byte	Low-Byte	High-Byte	Low-Byte	Crc16

Antwort Slave -> Master

Adresse	Funktion	Daten				CRC-Check
		Register Adresse		Daten		
Adr	06H	High-Byte	Low-Byte	High-Byte	Low-Byte	Crc16

Beispiel1: (Setzen von Register 262 mit Wert 10000, Slaveadresse = 1)

Master -> Slave: 01 06 01 05 27 10 Crc16

Slave -> Master: 01 06 01 05 27 10 Crc16

08H DIAGNOSTICS

Mit der Subfunktion 0 wird getestet, ob das Gerät mit (Adr) angeschlossen ist. Das gesendete Telegramm wird 1:1 zurückgesendet.

Tabelle 12: 08H DIAGNOSTIC Anforderung MA-SL

Adresse	Funktion	Daten				CRC-Check
		Subfunktion		Daten		
Adr	08H	Byte1=0	Byte2=0	High-Byte	Low-Byte	Crc16

Master -> Slave: 01 08 00 00 00 00 Crc16

Slave -> Master: 01 08 00 00 00 00 Crc16

Weitere unterstützte Subfunktionen:

Byte1 Byte2 Funktion
 00 0A CRC+Telegr. Zähler löschr., REPORT-SLAVE-ID auf Anfang setzen

Master -> Slave: 01 08 00 0A 00 00 Crc16

Slave -> Master: 01 08 00 0A 00 00 Crc16

00 0C Rückgabe CRC Zähler

Master -> Slave: 01 08 00 0C 00 00 Crc16

Slave -> Master: 01 08 00 0C crc-h crc-l Crc16

00 0E Rückgabe Telegramm Zähler

Master -> Slave: 01 08 00 0E 00 00 Crc16

Slave -> Master: 01 08 00 0E tel-h tel-l Crc16

0FH FORCE MULIPLE COILS

Tabelle 13: 0FH FORCE MULTIPLE COILS Anforderung MA-SL

Adresse	Funktion	Daten						CRC-Check
		Startadresse		Anz. Zustände		Anz. Bytes	Information	
Adr	0FH	High	Low	High	Low	n	n Bytes	Crc16

Antwort Slave -> Master

Adresse	Funktion	Daten				CRC-Check
		Startadresse		Anzahl Zustände		
addr	0FH	High-Byte	Low-Byte	High-Byte	Low-Byte	Crc16

Dieses Kommando kann auch im Broadcast Modus ausgeführt werden. Dadurch können Ausgänge auf allen Slaves gleichzeitig verändert werden. In diesem Fall gibt es keine Antwort von einem Slave.

Beispiel: Setzen der Digitalausgänge 1 – 12 von Slave 2

Senden: 02 0F 00 00 00 0C 02 **FF 0F** crc_l_1 crc_h

FF = 1111 1111b: Ausgang 1 – 8 An

0F = 0000 1111b: Ausgang 9 – 12 An, 13 – 16 = Aus

10H PRESET MULTIPLE REGISTERS

Tabelle 14: 10H PRESET MULTIPLE REGISTERS Anforderung MA-SL

Adresse	Funktion	Daten						CRC-Check
		Startadresse		Anz. Register		Anz. Bytes	Information	
Adr	10H	High	Low	High	Low	1-n	2 Bytes/Reg	Crc16

Antwort Slave -> Master

Adresse	Funktion	Daten				CRC-Check
		Startadresse		Anzahl Register		
Adr	10H	High-Byte	Low-Byte	High-Byte	Low-Byte	Crc16

Es können maximal 125 gleichzeitig Register beschrieben werden. Registeradressierungen sind immer relativ zur angegebenen Referenznummer. Register 1 = 0, Register 125 = 124.

Register, die nur zum Lesen freigegeben sind, können nicht überschrieben werden. Die Schreiboperation wird weiter bis zum nächsten beschreibbaren Register durchgeführt.

Dies wird durch eine interne Lock-Tabelle verhindert, die nur Schreibzugriffe auf freigegebene Register erlaubt.

Schreiben auf ein gesperrtes Register führt zu keiner Fehlermeldung.

Dieses Kommando kann auch im Broadcast Modus ausgeführt werden. Dadurch können Registerinhalte auf allen Slaves gleichzeitig verändert werden. In diesem Fall gibt es keine Antwort von einem Slave.

11H REPORT SLAVE-ID Die Antwort beinhaltet eine Zusammenstellung aller Daten der BGR.

Tabelle 15: 11H REPORT SLAVE ID Anforderung MA-SL

Adresse	Funktion	CRC-Check
Addr	11H	Crc16

Antwort:

Adresse	Funktion	Daten					CRC-Check
		Bytecount	Slave-ID	RUN-Ind.	Byte 1	Byte (n)	
Addr	11H	n=46/h	00h	0xFF	0-ffh	0-ffh	Crc16

Byte	Inhalt
1	Slave-ID = 0
2	Run-Index, 0xFF = Start Controller OK
2,3	Slave Adresse
4,5	CPU Version
6,7	Seriennummer
8,9	Versorgungsspannung
10,11	Baugruppentemperatur
12,13	Lüfterfehler 0=kein Lüfter, 1=OK, 3=Fehler, 8= Drahtbr., 16=Überstrom (allg. Status)
14,15	Controllertyp (siehe folgende Liste)
16,17	Resetcounter (wird bei Einschalten um 1 erhöht)
18,19	CRC-Fehler Counter (0 – 65000 – 0)
20,21	Telegrammzähler seit Einschalten (0 – 65000 – 0)
22,23	Betriebsstundenzähler High-Word in Minuten
24,25	Betriebsstundenzähler Low-Word in Minuten
26,27	Anzahl diskreter Ausgänge
28,29	Anzahl analoger Ausgänge
30,31	Anzahl diskreter Eingänge
32,33	Res.
34,35	Anzahl Eingangsregister
36, 37	Anzahl Holdingregister
38,39	Modul1 , Typ, Gruppierung, Reihenfolge
40, 41	Modul2, Typ, Gruppierung, Reihenfolge
42, 43	Modul3, Typ, Gruppierung, Reihenfolge
44, 45	Modul4, Typ, Gruppierung, Reihenfolge
46,47	Reverse-Einstell. (relev. für Module JDA48-4-D12L) Schalter 8 DIP-SW = ON = 1 = rev.
48	Default = 0
49	Startregister Analogausgänge
50	Default = 0
51	Endregister Analogausgänge
52	Default = 0
53	Startregister diskrete Ausgänge 4 Bit Gruppe
54	Default = 0
55	Endregister diskrete Ausgänge 4 Bit Gruppe
56	Default = 0
57	Startregister diskrete Ausgänge 6 Bit Gruppe
58	Default = 0
59	Endregister diskrete Ausgänge 6 Bit Gruppe

60	COM (RS485) Speed
61	COM (RS485) Datalength
62	COM (RS485) Stopbit
63	COM (RS485) Parity
64	COM (RS422) Speed
65	COM (RS422) Datalength
66	COM (RS422) Stopbit
67	COM (RS422) Parity
68	Default = 0
69	BGR-Mode (1 = default = Modbus RTU)
70	BGR-Status (1=Factory Setup, + 3 = Kalibriert=normal)

Der erste Aufruf des Kommandos sendet alle 64 Byte zurück, alle folgenden Aufrufe senden nur noch die ersten 23 Byte zurück. Durch das Kommando **08_h** mit Subfunktion **00 0A_h** kann das Senden aller 64Bytes wieder einmalig aktiviert werden.

Controllertyp (Byte 14,15)

Die nachfolgende Liste definiert den Typ des Controller durch eine ID:

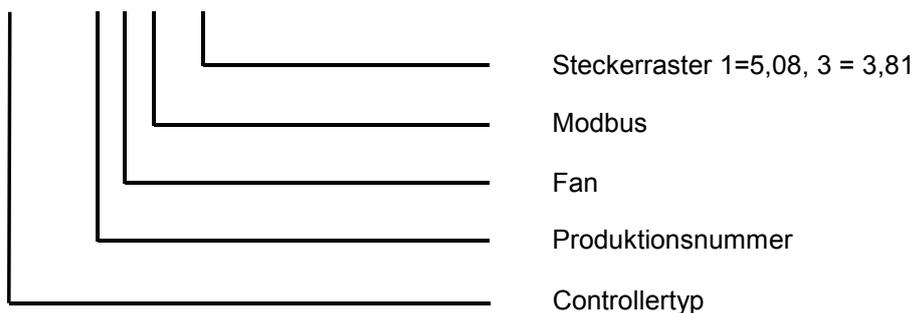
ID	Controller	Typ	Bezeichnung	Raster
5	JDA48-4-1FM-3	D	Contr. mit Lüfter/Lüfterüberw. Ausg. 8x6Bit 24V/240mA*	3,81
6	JDA48-4-2FM-3	A	Contr. mit Lüfter/Lüfterüberw. Ausg. 48, 24V/20mA*	3,81
7	JDA48-4-3FM-3	D	Contr. mit Lüfter/Lüfterüberw. Ausg. 12x4Bit, 24VDC/240mA*	3,81
8	JDA48-4-4FM-3	D	Contr. mit Lüfter/Lüfterüberw. Ausg. 12x4Bit, 24VDC/500mA, SolidST	3,81
9	JDA48-4-5FM-1	D	Contr. mit Lüfter/Lüfterüberw. Ausg. 8x4Bit, 24V/AC/2000mA, SolidST	3,81
10	JDA48-4-6FM-3	D/A/E	Contr. mit Lüfter/Lüfterüberw. Gemischte Ausstattung	3,81

Typ: D = digital, A = analog, E = dig. Eingang

(*) = Drahtbruch/Überstromüberwachung

Zusammensetzung der Controllerbezeichnung:

JDA48-4 – 1 F M - 3



40H PROG COM (Baugruppen spezifisch)

41H COM RESET

Durch das Kommando **40H** können die Schnittstellenparameter der Baugruppe(n) verändert werden. Mit dem Kommando **41H** werden im Anschluss die Kommunikationsmodule der Baugruppe(n) mit den programmierten Parametern neu gestartet.

Es ist unbedingt darauf zu achten, dass der BUS Master im Anschluss ebenfalls geändert werden muss, da sonst keine Kommunikation mehr möglich ist!

Wird das Kommando **COM RESET** nicht innerhalb von 10 Sekunden nach dem Kommando **PROG COM** gesendet, so wird die vorher programmierte Veränderung automatisch annulliert.

Diese Kommandos können im Broadcast-Modus (Adresse 0) gesendet werden. Dadurch tritt die Veränderung bei allen Slavebaugruppen gleichzeitig ein.

Nach dem Senden von **COM RESET** darf der Master innerhalb von 5 Sekunden kein weiteres Kommando senden.

Die erfolgreiche Änderung wird an der Baugruppe durch längeres Flackern (~ 5 Sek.) der HB-LED signalisiert.

Die veränderten Parameter werden in der BGR gespeichert und sind nach dem Neueinschalten wieder relevant.

Die relevanten Parameter bitte der Übersicht Holdingregister entnehmen.

Tabelle 16: 40H PROG COM Anforderung MA-SL

Adresse	Funktion	Daten										CRC-Check
		Count	485 Speed	485 DataLe	485 StopB	485 Parity	422 Speed	422 DataL	422 StopB	422 Parity		
Adr/ (0)	40H	1 Byte 0x08	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	Crc16

1. Antwort Slave -> Master (nicht bei Adresse 0)

Adresse	Funktion	Daten										CRC-Check
		Count	485 Speed	485 DataLe	485 StopB	485 Parity	422 Speed	422 DataL	422 StopB	422 Parity		
Adr	40H	1 Byte 0x08	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	Crc16

2. Aufforderung Master->Slave (innerhalb von 10 Sekunden)

Adresse	Funktion	Daten			CRC-Check
		Check1	Check2	Check3	
Adr/ (0)	41H	0x88	0x55	0x33	Crc16

Wird nach 10 Sekunden dieses Kommando geschickt, oder dieses Kommando ohne das vorherige antwortet der Slave mit einem Fehlercode.

Zur Zeit sind die RS422 Parameter nicht relevant, müssen aber dennoch ordnungsgemäß mitgeschickt werden, da jeder einzelne Parameter geprüft wird.

11.0 Registersatz initialisieren

Mit dem Befehl **42H** und Subfunktion **1** kann ein kompletter Registersatz mit Werten initialisiert werden, die zum Betrieb von 4 Modulen **JDA48-4-D12L** notwendig sind. Dabei wird ein Mapping auf 2 x 6Bit pro Modul und Drahtbruch und Überstromabfrage selektiert.

Tabelle 17: 42H INIT REGISTER Sub-Funktion 1

Adresse	Funktion	Sub	CRC-Check
Adr/ (0)	42H	01h	Crc16

1. Antwort Slave -> Master (nicht bei Adresse 0)

Adresse	Funktion	Sub	CRC-Check
Adr/ (0)	42H	01h	Crc16

Der Controller führt den Befehl nur aus, wenn 4 gleiche Module vom Typ JDA48-4-D12L vorhanden sind. Ist dies nicht der Fall, gibt der Slave eine Fehlermeldung zurück. Der Befehl kann auch im Broadcast-Modus verwendet werden. Damit werden alle Controller gleichzeitig programmiert. Nach Erhalt des Kommandos werden folgende Register mit folgenden Werten geladen:

- R200:** xxxx xx**10** xxxx xxxx = Gruppe 2: 6Bit mapping selektiert
- R201:** xxxx xx**10** xxxx xxxx = Gruppe 2: 6Bit mapping selektiert
- R202:** xxxx xx**10** xxxx xxxx = Gruppe 2: 6Bit mapping selektiert
- R203:** xxxx xx**10** xxxx xxxx = Gruppe 2: 6Bit mapping selektiert

- R213:** 0000 **1111 1111 1111** = **0FFF_H** Drahtbruchüberwachung Ausgang 01 – 12
- R214:** 0000 **1111 1111 1111** = **0FFF_H** Drahtbruchüberwachung Ausgang 13 – 24
- R215:** 0000 **1111 1111 1111** = **0FFF_H** Drahtbruchüberwachung Ausgang 25 – 36
- R216:** 0000 **1111 1111 1111** = **0FFF_H** Drahtbruchüberwachung Ausgang 37 – 48

- R212:** 0000 0000 0000 0001 = **0001_H** Drahtbruchererkennung eingeschaltet

12.0 Fehlerbehandlung

Stimmt das vom Slave berechnete CRC des Telegramms nicht mit dem übertragenen CRC überein, so wird keine Quittung gesendet und damit ein Timeout erzwungen. Stellt der Slave fest, das nicht gültige Daten, z.B. falsche Registeradressen, Inhalte oder Funktionscodes gesendet wurden, so sendet er eine entsprechende Fehlermeldung an den Master zurück.

Antwort Slave -> Master

Adresse	Funktion	Daten	CRC-Check
		addr	

Fehlercode	Bedeutung
01h	Nicht unterstützter Funktionscode
02h	Verwendung eines unerlaubten oder nicht vorhandenen Speicherregisters
03h	Unerlaubte Datenwerte; z.B. falsche Anzahl Register
06h	Gerät kann Anfrage momentan nicht bearbeiten. Anfrage später wiederholen

13.0 CRC - Generierung

Ein CRC-Fehler wird durch den Fehlercode „CC“ am Display angezeigt.

(C-Beispiel)

```
WORD Mod_crc_rtu(BYTE * buf, WORD len)
{
    WORD crc ,n;
    BYTE i;
    crc = 0xffff;

    for(n=0;n<len-2;n++)
    {
        crc = crc^((* buf++)&0xff);
        for(i=8;i>0;i--)
        {
            if(crc&1)
            {
                crc=crc>>1;
                crc=crc^0xA001;
            }
            else crc=crc>>1;
        }
    }
    return (crc);
}
```

Achtung! CRC-Daten werden zuerst mit dem LSB Anteil gesendet.

14.0 Programmierbeispiele

Beispiel 1:

Ein Controller JDA48-4-1F mit 4 Ausgangsbaugruppen JDA48-4-D12L (je 12 Ausgänge) soll konfiguriert und betrieben werden. Die E/A Baugruppen ermöglichen eine Drahtbruch- und Überstromüberwachung. Die Ausgänge sollen zu je 6Bit gruppiert und mit dem relevanten Registersatz angesprochen werden.

Folgende Register müssen **einmalig** mit folgenden Werten programmiert werden:

1. Gruppierung festlegen, Register 200 – 203 auf Gruppe 2 setzen

Senden

Adresse	FC	Daten												CRC-Check	
		Startadresse		Anz. Register		Anz. Bytes	R200		R201		R202		R203		
Adr	10H	00 _h	C7 _h	00 _h	04 _h	08 _h	00 _h	02 _h	Crc16						

Anwort (nicht bei Adresse 0)

Adresse	FC	Daten				CRC-Check
		Startadresse		Anz. Register		
Adr	10H	00 _h	C7 _h	00 _h	04 _h	Crc16

2. Drahtbruch einschalten (212) und alle Drahtbruch-Checkbits (213-216) setzen

Senden

Adresse	FC	Daten														CRC-Check	
		Start		Anz.		Bytes	R212		R213		R214		R215		R216		
Adr	10H	00 _h	D3 _h	00 _h	05 _h	0A _h	0F _h	FF _h	Crc16								

Anwort (nicht bei Adresse 0)

Adresse	FC	Daten				CRC-Check
		Startadresse		Anz. Register		
Adr	10H	00 _h	D3 _h	00 _h	05 _h	Crc16

Dieser Vorgang muss auch bei Neustart des Controllers nicht wiederholt werden, da alle Werte permanent abgelegt wurden. Werden die Daten mit Adresse 0 geschickt, so übernehmen alle Controller die Werte.

Die Register enthalten nun folgende Werte:

R200	xxxxxx10 xxxxxxxx	Gruppe 2 Mod 1	Eingangsreg. 0062+0063	gemappt auf 2x6Bit Ausgang
R201	xxxxxx10 xxxxxxxx	Gruppe 2 Mod 2	Eingangsreg. 0064+0065	gemappt auf 2x6Bit Ausgang
R202	xxxxxx10 xxxxxxxx	Gruppe 2 Mod 3	Eingangsreg. 0066+0067	gemappt auf 2x6Bit Ausgang
R203	xxxxxx10 xxxxxxxx	Gruppe 2 Mod 4	Eingangsreg. 0068+0069	gemappt auf 2x6Bit Ausgang
R212	00000000 00000001	Drahtbruchüberwachung generell eingeschaltet		
R213	00001111 11111111	Alle Ausgänge Modul 1 werden überwacht		
R214	00001111 11111111	Alle Ausgänge Modul 2 werden überwacht		
R215	00001111 11111111	Alle Ausgänge Modul 3 werden überwacht		
R216	00001111 11111111	Alle Ausgänge Modul 4 werden überwacht		

Durch Senden der Registerinhalte **40062 – 40069** können jetzt 8 x 6Bit Ausgänge angesprochen werden. Z.B alle Ausgänge auf dem Slave mit Adresse 1 aktiv schalten:

Master -> 01 10 00 3D 00 08 0F 2710 2710 2710 2710 2710 2710 2710 Crc16
 Slave -> 01 10 00 3D 00 08 Crc16

Selbstverständlich kann ein Ausgang nach wie vor als 0XXXX Referenz angesprochen werden. Dies ist durch das Kommando FORCE SINGLE COIL/FORCE MULTIPLE COILS möglich. Dabei wird jedoch das Ausgangsregister des jeweiligen Moduls überschrieben. Die Drahtbruch/Überstromüberwachung ist davon aber unabhängig.

Abfragen Drahtbruch/Überstrom

40118 (zurück zur Registerbeschreibung)

Die Register **118 – 125** enthalten den Drahtbruch/Überstromstatus als **2Bit Ergebnis** pro Ausgang.

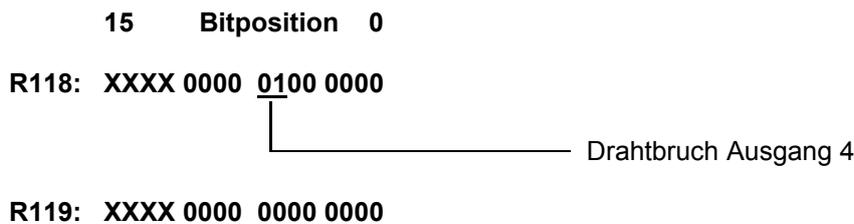
Um die ersten beiden Ausgänge abzufragen sind die Register 118+119 relevant. Im Betrieb werden alle Register eines Controllers gleichzeitig abgefragt.

Abfrage Register 118+119 auf Slave mit Adr. 1 mit Befehl READ HOLDING REGISTER und Annahme auf Ausgang 4 sei ein **Drahtbruch** detektiert worden:

Master -> 01 03 00 75 00 02 Crc16

Slave -> 01 03 00 75 00 04 00 40 00 00 Crc16

Registerinhalte:

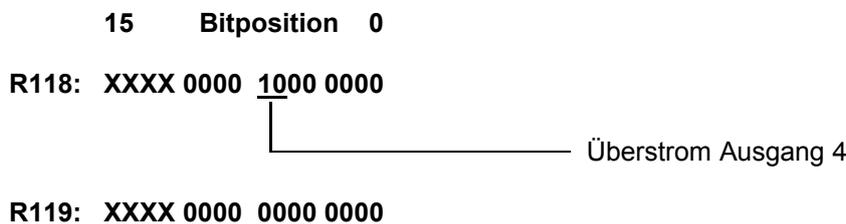


Abfrage Register 118+119 auf Slave mit Adr. 1 mit Befehl READ HOLDING REGISTER und Annahme auf Ausgang 4 sei ein **Überstrom** detektiert worden:

Master -> 01 03 00 75 00 02 Crc16

Slave -> 01 03 00 75 00 04 00 80 00 00 Crc16

Registerinhalte:



Ende Beispiel 1

Beispiel 2:

Die unter Beispiel 1 durchgeführte Grundprogrammierung der Drahtbruchregister wurde bereits durchgeführt. Allerdings sollen nun die Ausgänge mit Hilfe der Erweiterungsregister angesteuert werden. Da jede Ausgangskarte 12 Ausgänge besitzt, ist jeder Ausgang eine Bitposition im Register 301-304 zugeordnet. Es obliegt nun dem Programmierer die Bitkombinationen in den Registern so zusammen zu fassen, dass die Ausgangslogik sinnvoll ist, denn die Ausgabe ist absolut, d.h. es findet keine Skalierung oder Gruppenzuweisung statt.

Übersicht der Bitpositionen in den Registern zu den Ausgängen:

Zuordnung Reg. 301:	High-Byte Register (301)								Low-Byte Register (301)							
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1

Zuordnung Reg. 302:	High-Byte Register (302)								Low-Byte Register (302)							
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13

Zuordnung Reg. 303:	High-Byte Register (303)								Low-Byte Register (303)							
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	A36	A35	A34	A33	A32	A31	A30	A29	A28	A27	A26	A25

Zuordnung Reg. 304:	High-Byte Register (304)								Low-Byte Register (304)							
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	A48	A47	A46	A45	A44	A43	A42	A41	A40	A39	A38	A37

Der Vorteil dieser Steuerung ist, dass pro Controller zum Steuern aller Ausgänge nur 4 Register geschickt werden müssen. Bei bestimmten spezifischen Anlagen (z.B. VIB alter Bauart), muss auf die Ausgangsreihenfolge geachtet werden, da diese in umgekehrter Reihenfolge der Ausgänge verdrahtet sind. Bei jedem Senden werden die Ausgänge mit den Inhalten der Register überschrieben. Deswegen müssen die Ausgangswerte logisch verknüpft werden (OR/AND).

Annahme: Es sollen die Ausgänge 1-6, 31 – 36 und 37-40 angesteuert werden.

R301: 0000 0000 0011 1111 = 003F _h		Start: 301-1 = 300 = 012C _h
R302: 0000 0000 0000 0000 = 0000 _h		Anz: Anzahl Reg. = 0004 _h
R303: 0000 1111 1100 0000 = 0FC0 _h		Bytes: Anzahl Bytes = 08 _h
R304: 0000 0000 0000 1111 = 000F _h		

Mit dem Befehl 10h (PRESET MULTIPLE REGISTER) werden die Daten übertragen:

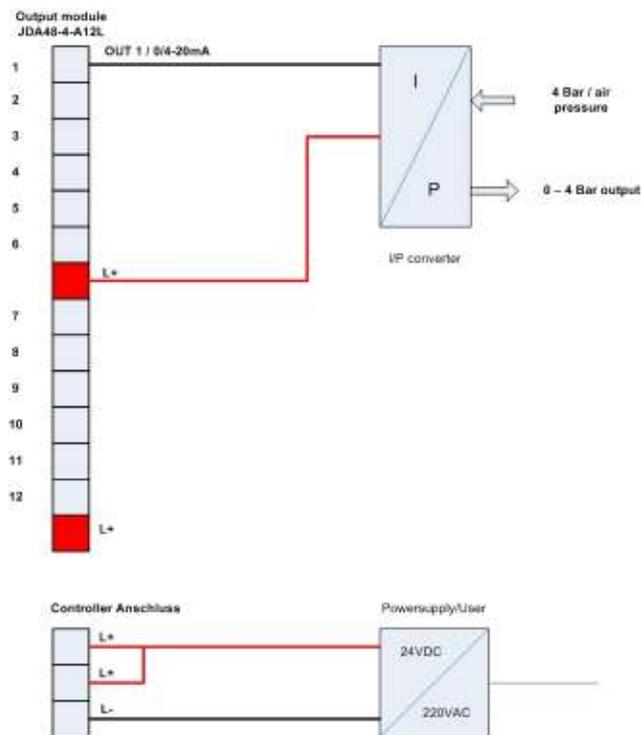
Master -> 01 10 01 2C 00 04 08 00 3F 00 00 0F C0 00 0F Crc16
 Start Anz. Bytes 301 302 303 304

Slave -> 01 10 01 2C 00 04 Crc16

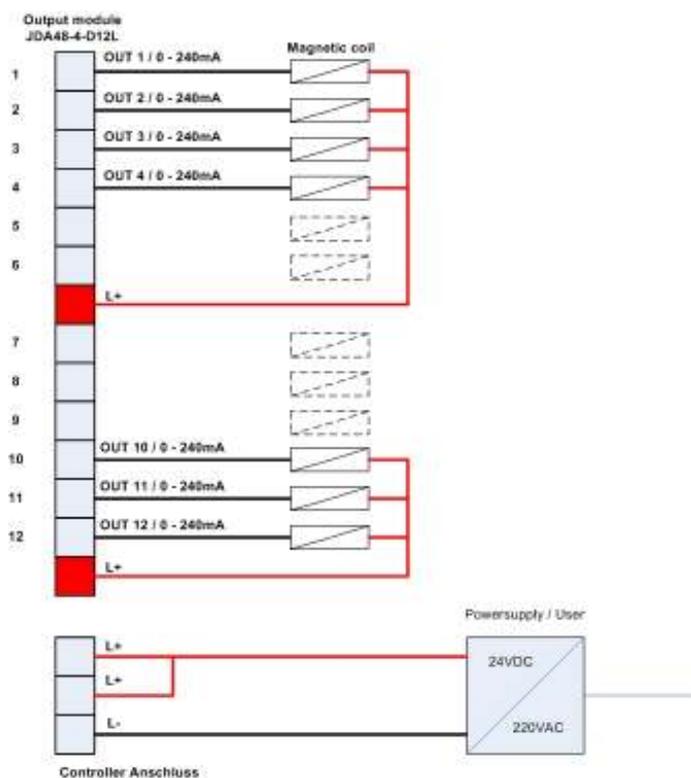
Ende Beispiel 2

15.0 Anschluss-Schema Baugruppen

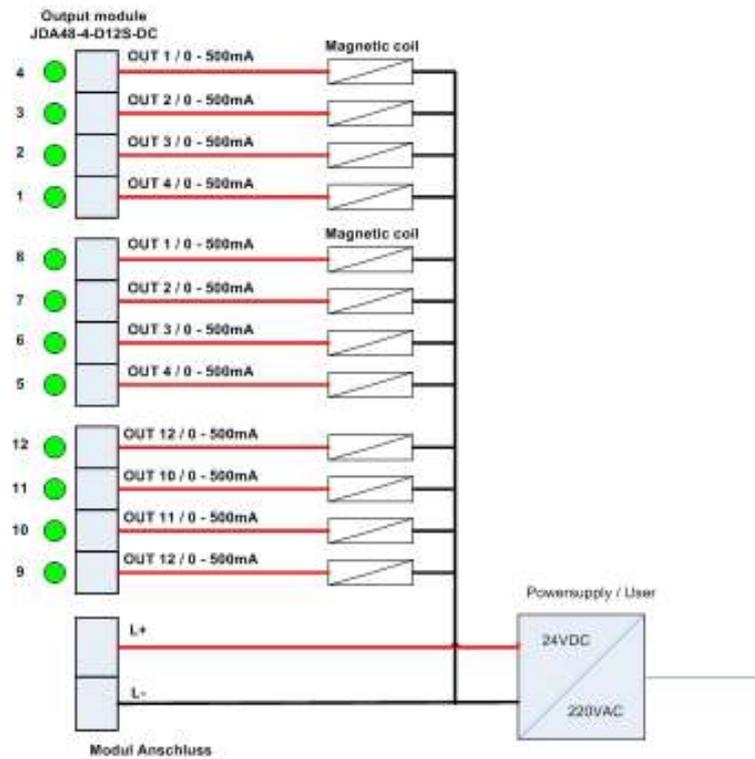
1. Anschluss an ein Analogmodul (JDA48-4-A12L)



2. Anschluss an ein Digitalmodul (JDA48-4-D12L)



3. Anschluss an ein Ausgangsmodul (JDA48-4-D12S-DC)



4. Anschluss an ein Eingangsmodul (JDA48-4-E8-DC)

