

---

# **POSITIONIER- UND BAHNSTEUERUNG MCU-3T SYSTEMZUBEHÖR / SZB**



**1 Optionsprint OPMF-1002 ..... 5**

- 1.1 Kurzbeschreibung..... 5
- 1.2 Software..... 6
  - 1.2.1 Modifizierte Betriebssystemsoftware rwtos.btl ..... 6
  - 1.2.2 Hilfsprogramm mcfg.exe ..... 6
  - 1.2.3 PCAP-Programmierung ..... 6
    - 1.2.3.1 rdAin, read analog inputs ..... 6
    - 1.2.3.2 rddigi, read digital inputs ..... 6
    - 1.2.3.3 rddigib, read digital input bit ..... 8
    - 1.2.3.4 rddigo, read digital outputs..... 8
    - 1.2.3.5 rddigo, read digital output bit..... 8
    - 1.2.3.6 wrdigo, write digital outputs..... 8
    - 1.2.3.7 wrdigob, write digital output bit..... 8
  - 1.2.4 SAP-Programmierung..... 9
    - 1.2.4.1 Achsenqualifizierer digi und digo ..... 9
    - 1.2.4.2 Achsenqualifizierer ain0 .. ain3 ..... 9
- 1.3 Hardware ..... 10
  - 1.3.1 Optionsprint OPMF-1002, Anschlußbelegungen ..... 11
    - 1.3.1.1 Stecker X100: 50-poliger SUB-D-Steckverbinder (Stift) ..... 11
    - 1.3.1.2 Sollwertkanäle..... 12
      - 1.3.1.2.1 Sollwertkanal für Servoachsen ..... 12
        - 1.3.1.2.1.1 Pinbelegung Stecker X100, Achskanal 4 ..... 12
      - 1.3.1.2.2 Sollwertkanal für Schrittmotorachsen..... 12
        - 1.3.1.2.2.1 Pinbelegung Stecker X100, Achskanal 4 ..... 12
    - 1.3.1.3 Pinbelegung Stecker X100, Digitale Eingänge ..... 13
    - 1.3.1.4 Pinbelegung Stecker X100, Digitale Ausgänge ..... 13
    - 1.3.1.5 Pinbelegung Stecker X100, Freigaberelais ..... 13
    - 1.3.1.6 Impulserfassungskanäle ..... 13
      - 1.3.1.6.1 Inkremental-Encoder mit invertierten Signalen ..... 13
      - 1.3.1.6.2 Inkremental-Encoder ohne invertierte Signale ..... 14
      - 1.3.1.6.3 Optische Entkopplung der Impulserfassungskanäle ..... 14
      - 1.3.1.6.4 Steckerbelegung für die Impulserfassungskanäle mit Inkrementalgebern ..... 14
        - 1.3.1.6.4.1 Steckerbelegung X100, Kanal 4..... 14
      - 1.3.1.6.5 Steckerbelegung für die Analogeingangskanäle ..... 15
      - 1.3.1.6.6 Steckerbelegung für die Referenzspannungsausgänge ..... 15
  - 1.3.2 Bestückungsdruck des OPMF-1002 Top-Layer ..... 16
  - 1.3.3 Bestückungsdruck des OPMF-1002 Bottom-Layer ..... 17
  - 1.3.4 Technische Daten des OPMF-1002 ..... 18

**2 Option IBS ..... 19**

- 2.1 Kurzbeschreibung..... 19
- 2.2 Software..... 19
  - 2.2.1 Einführung ..... 19
  - 2.2.2 Konfigurationssoftware ibscfg.btl ..... 20
  - 2.2.3 SAP-Programmierung..... 21
    - 2.2.3.1 System-Parameter IBSCS0 .. IBSCS9 ..... 21

2.2.3.2	System-Parameter IBSPD0 .. IBSPD31 .....	22
2.2.4	PCAP-Programmierung .....	22
2.2.4.1	rdibscs, read IBS command/status registers .....	22
2.2.4.2	wribscs, write IBS command/status registers .....	22
2.2.4.3	rdibsddata, read IBS data registers .....	23
2.2.4.4	wribsddata, write IBS data registers .....	23
2.3	Hardware .....	24
2.3.1	Optionsprint OPMF-1002, Anschlußbelegungen .....	24
2.3.1.1	Stecker X1 (JTAG / ISP): 10-poliger FB-Steckverbinder .....	24
2.3.1.2	Stecker X2 (LCD Diag): 10-poliger FB-Steckverbinder .....	24
2.3.1.3	Stecker X3 (Interbus): 9-poliger SUB-D-Steckverbinder (Buchse) .....	24
2.3.1.4	Stecker X4 (IBS / Diag): 9-poliger SUB-D-Steckverbinder (Stift) .....	25
2.3.1.5	Stecker X14 (BDMI): 9-poliger SUB-D-Steckverbinder (Stift) .....	25
2.3.2	Technische Daten der Option IBS .....	26

### **3 SUB-D-Adapter von 10-poligem FB-Stecker X5 auf 9-poligen SUB-D-Stecker, für Freigabe-Relais, CNC-Bereit-Relais..... 27**

### **4 Optionsprint OPIO-3216..... 28**

4.1	Kurzbeschreibung .....	28
4.2	Software .....	28
4.2.1	PCAP-Programmierung .....	28
4.2.1.1	rdopin, read option-print inputs .....	28
4.2.1.2	rdopout, read option-print outputs .....	28
4.2.1.3	wropout, write option-print outputs .....	29
4.2.2	SAP-Programmierung .....	29
4.2.2.1	System-Parameter SR0 .....	29
4.2.2.2	System-Parameter SR1 .....	31
4.2.3	Optionsprint OPIO-3216, Anschlußbelegungen .....	32
4.2.3.1	Stecker X100: 50-poliger SUB-D-Steckverbinder (Stift) .....	32
	Bestückungs- druck des OPIO-3216 .....	33
4.2.5	Technische Daten des OPIO-3216 .....	34

# 1 Optionsprint OPMF-1002

## 1.1 Kurzbeschreibung

Der Optionsprint OPMF-1002 dient zur Systemerweiterung der Positionier- und Bahnsteuerung MCU-3T. Im wesentlichen beinhaltet er die Zusatzlogik für einen vierten Achskanal. Dieser ist voll kompatibel zu den Achskanälen der MCU-3T. Darüber hinaus kann der OPMF-1002 zusätzlich bis zu 12 analoge Eingangssignale verarbeiten. Optional kann der OPMF-1002 mit einer Interbus-S-Masterbaugruppe (OPTION IBS) bestückt werden. Diese Option ist in Abschnitt 2 beschrieben.

## 1.2 Software

Der OPMF-1002 kann mit der Standard-TOOLSET-Software für die MCU-3T projektiert und programmiert werden (ab Revision V2-10). Die entsprechenden Zusatzfunktionen werden aufbauend auf dem Programmierhandbuch [PHB] der MCU-3T in den nachfolgenden Kapiteln beschrieben.

### 1.2.1 Modifizierte Betriebssystemsoftware *rwtos.btl*

Damit die Zusatzfunktionen des OPMF-1002 genutzt werden können, ist eine spezielle Betriebssystemsoftware *rwtos.btl* notwendig. Bei der Aufrüstung zum Vier-Achsensystem sind die Informationen der nachfolgenden Kapitel relevant. In diesem Fall werden die Zusatzfunktionen des vierten Achskanals genauso genutzt wie beim Standard-Drei-Achsensystem.

Sofern jedoch nur einzelne Funktionsbaugruppen auf dem OPMF-1002 vorhanden sind (Teilbestückung), wird der Zugriff auf diese Funktionsbaugruppen mit Hilfe der SAP-Programmiermethode ausgeführt. Hierzu sind eine Reihe neuer Systemparameter in *rw\_SymPas* definiert. Die im Lieferumfang enthaltenen Beispielprogramme veranschaulichen den einfachen Zugriff dieser Zugriffsmethode.

### 1.2.2 Hilfsprogramm *mcfg.exe*

Im Menü [Configuration Parameters][Set OPMF-1002 parameters] können die zusätzlichen Projektierungsmöglichkeiten für den OPMF-1002 freigeschaltet werden. Bei der Freischaltung werden folgende Bildschirme mit zusätzlichen Feldern aufgelegt:

- [Hardware Parameters] [Edit MCU-3T Input Configuration]
- [Hardware Parameters] [Edit MCU-3T Output Configuration]
- [Dialog Functions Menu] [Show Inputs / Status of MCU-3T]
- [Dialog Functions Menu] [Edit Outputs of MCU-3T]

### 1.2.3 PCAP-Programmierung

In den nachfolgenden Kapiteln wird auf die einzelnen Signale und deren Anschlußpin verwiesen. Die Pinnumerierung bezieht sich auf einen 50-poligen SUB-D Steckverbinder, der per Flachbandkabel 1:1 an X100 angeschlossen wird.

#### 1.2.3.1 rdAin, read analog inputs

Diese Funktion wurde für Windows 95/98/NT mit Zugriffen unter Delphi oder C realisiert. Die Funktionsdeklaration befindet sich in *specfunc.c* bzw. *specfunc.pas*.

DELPHI:                   function rdAin (an:longint; channel:longint):longint;

C:                         long rdAin(long an, long channel);

BESCHREIBUNG:           Diese Funktion liefert den aktuellen Analogwert des Kanals *channel* der Achse *an* zurück. Jeder Achse sind maximal 4 Kanäle zugeordnet.

RÜCKGABEWERT:          Analogwert als Ganzzahl.

#### 1.2.3.2 rddigi, read digital inputs

Die Struktur- bzw. Record-Komponente *digi* hat nun folgende Erweiterung:

Tabelle 1: Bitkodierter Aufbau des digi-Wortes

Bit-Nr.	Funktion	X100 (SUB-D) Pin
20	Eingang 17	18
21	Eingang 18	19
22	Eingang 19	20
23	Eingang 20 u. Hardware-Strobe-Signal zum Latchen der Istposition Achskanal 4	21
24..31	Nicht belegt, diese Flags haben immer den Wert 0	--

### 1.2.3.3 rddigib, read digital input bit

Diese Funktion liefert zusätzlich den Zustand folgender Eingänge zurück:

Tabelle 2: Zuordnung von *bitnr* zu den jeweiligen MCU-3T bzw. OPMF-1002 Digitaleingängen

'bitnr'	Funktion	X100 (SUB-D) Pin
21	Eingang 17	18
22	Eingang 18	19
23	Eingang 19	20
24	Eingang 20	21
25..32	Nicht belegt, diese Flags haben immer den Wert 0	--

### 1.2.3.4 rddigo, read digital outputs

Siehe wrdigo im Kapitel [1.2.3.6]

### 1.2.3.5 rddigo, read digital output bit

Siehe wrdigob im Kapitel [1.2.3.7]

### 1.2.3.6 wrdigo, write digital outputs

Die Struktur- bzw. Record-Komponente *digo* wurde wie folgt erweitert:

Tabelle 3: Bitkodierter Aufbau des digo-Wortes

Bit-Nr.	Funktion	Stecker X100 (SUB-D) / PIN
8	Ausgang 9	22
9	Ausgang 10	23
10	Ausgang 11	24
11	Ausgang 12	25
12..31	Nicht belegt.	--

### 1.2.3.7 wrdigob, write digital output bit

Mit dieser Funktion kann der Zustand folgender Ausgänge zusätzlich programmiert werden:

Tabelle 4: Zuordnung von *bitnr* zu den jeweiligen MCU-3T bzw. OPMF-1002 Digitalausgängen

'bitnr'	Funktion	Stecker X100 (SUB-D) / PIN
9	Ausgang 9	22
10	Ausgang 10	23
11	Ausgang 11	24
12	Ausgang 12	25
13..32	Nicht belegt.	--

## 1.2.4 SAP-Programmierung

### 1.2.4.1 Achsenqualifizierer *digi* und *digo*

Die Achsenqualifizierer *digi* und *digo* wurden ebenfalls auf die neuen Verhältnisse angepaßt. Die entsprechenden Informationen sind bei den jeweiligen PCAP-Befehlen nachzulesen.

### 1.2.4.2 Achsenqualifizierer *ain0* .. *ain3*

Die Achsenqualifizierer *ain0* .. *ain3* dienen zum Einlesen der Analogen Eingangskanäle. Die Achsenqualifizierer sind den Achskanälen 1, 2 und 3 zugeordnet. Somit können bis zu 12 Analog-Eingänge verarbeitet werden. Die Eingangsinformation wird als *integer*-Typ zurückgeliefert. Der Wertebereich hängt von der Auflösung des entsprechenden Meßwertkanals ab. Im Normalfall sind dies 12 Bit inklusive Vorzeichen, d.h. es werden Werte von -2048..2047 zurückgeliefert. Welchem Spannungswert dieser Meßwert entspricht hängt vom Eingangsspannungsbereich des jeweiligen AD-Kanals ab. Im Normalfall sind dies -5..5V. Es können jedoch auch Meßbereiche von -10..+10V oder 0..10V verarbeitet werden.

## 1.3 Hardware

Der Optionsprint OPMF-1002 ist als Tochterboard für die MCU-3T-Grundplatine realisiert. Die beiden Baugruppen werden mit Steck- und Schraubverbindungen werkseitig assembliert. Die Steuerung benötigt aufgrund der geringen Bauhöhe nur einen PC-Steckplatz. Die Peripherie-Elektronik wird mit Hilfe von 50poligen SUB-D-Steckverbindern auf der MCU-3T und 50poligen FB-Steckverbindern auf dem OPMF-1002 angeschlossen. Für den OPMF-1002 ist u.a. auch ein Kabelsatz mit Kartenhalter und integriertem 50pol. SUB-D-Steckverbinder erhältlich. Dieser Kartenhalter kann in einen freien PC-Slot neben der Steuerung eingebaut werden. Alle nachfolgenden Pinbeschreibungen beziehen sich auf diesen 50poligen SUB-D Steckverbinder. Die Anschlußbelegung des Flachkabelsteckers kann der Tabelle in Abschnitt 1.3.1.1 entnommen werden.

### 1.3.1 Optionsprint OPMF-1002, Anschlußbelegungen

#### 1.3.1.1 Stecker X100: 50-poliger SUB-D-Steckverbinder (Stift)

Pin	Name	Gruppe	50-pol. Flachkabel
1	SERVO4 / PULSE4+	Sollwert 4 / Stepper 4	1
2	AGND4 / PULSE4-	Sollwert 4 / Stepper 4	4
3	CHA4+ / CLKSSI4+	Istwert 4	7
4	CHA4- / CLKSSI4-	Istwert 4	10
5	CHB4+ / DATSSI4+	Istwert 4	13
6	CHB4- / DATSSI4-	Istwert 4	16
7	NDX4+ / SIGN4+	Istwert 4 / Stepper 4	19
8	NDX4- / SIGN4-	Istwert 4 / Stepper 4	22
9	AIN1+	Analog Eingang 1	25
10	AIN1-	Analog Eingang 1	28
11	AIN2+	Analog Eingang 2	31
12	AIN2-	Analog Eingang 2	34
13	AIN3+	Analog Eingang 3	37
14	AIN3-	Analog Eingang 3	40
15	AIN4+	Analog Eingang 4	43
16	AIN4-	Analog Eingang 4	46
17	+24V	Spannungsversorgung dig. Ausgänge 24V	49
18	I17	dig. Eingang	3
19	I18	dig. Eingang	6
20	I19	dig. Eingang	9
21	I20	dig. Eingang	12
22	O9	dig. Ausgang	15
23	O10	dig. Ausgang	18
24	O11	dig. Ausgang	21
25	O12	dig. Ausgang	24
26	AIN5+	Analog Eingang 5	27
27	AIN5-	Analog Eingang 5	30
28	AIN6+	Analog Eingang 6	33
29	AIN6-	Analog Eingang 6	36
30	AIN7+	Analog Eingang 7	39
31	AIN7-	Analog Eingang 7	42
32	AIN8+	Analog Eingang 8	45
33	AIN8-	Analog Eingang 8	48
34	AGND1	Referenzspannungsausgang 1	2
35	AREF1	Referenzspannungsausgang 1	5
36	AGND2	Referenzspannungsausgang 2	8
37	AREF2	Referenzspannungsausgang 2	11
38	AGND3	Referenzspannungsausgang 3	14
39	AREF3	Referenzspannungsausgang 3	17
40	REL-	Freigaberelais	20
41	REL+	Freigaberelais	23
42	AIN9+	Analog Eingang 9	26
43	AIN9-	Analog Eingang 9	29
44	AIN10+	Analog Eingang 10	32
45	AIN10-	Analog Eingang 10	35
46	AIN11+	Analog Eingang 11	38
47	AIN11-	Analog Eingang 11	41
48	AIN12+	Analog Eingang 12	44
49	AIN12-	Analog Eingang 12	47
50	GND-D	Spannungsversorgung dig. Ein- und Ausgänge 24V	50

### 1.3.1.2 Sollwertkanäle

Die OPMF-1002-Systemelektronik unterstützt den wahlweisen Betrieb eines weiteren Schritt- bzw. Servomotors. Die Projektierung und Auswahl des gewünschten Motorsystems wird mit Hilfe des TOOLSET Programms *mcf.exe* durchgeführt.

#### 1.3.1.2.1 Sollwertkanal für Servoachsen

Das Analogausgangssignal dient zur Ansteuerung eines Leistungsverstärkers, welcher als Drehzahlregler bzw. Momentenregler (Stromverstärker) geschaltet ist. Der Offset dieses Sollwertkanals wird werkseitig im EEPROM der MCU-3T abgelegt und bei der Ausgabe softwaremäßig berücksichtigt. Die Anlogsollwertausgabe wird nur bei *SERVO*-projektierten Achsen unterstützt.

##### 1.3.1.2.1.1 Pinbelegung Stecker X100, Achskanal 4

Pin	Name	Gruppe	Beschreibung
1	SERVO4	Sollwert 4	Analogausgangssignal 4 zur Ansteuerung eines Leistungsverstärkers (+/-10V, 5mA). Dieses Signal ist von der OPMF-1002-Systemelektronik galvanisch getrennt und hat das Bezugspotential AGND4.
2	AGND4	Sollwert 4	Bezugspotential für SERVO4. Dieses Potential ist von der OPMF-1002-Systemelektronik galvanisch getrennt.

**Achtung:** Die Stiftleiste J1 muß in Stellung 2-3 und J2 muß in Stellung 2-3 gebrückt sein, damit die oben aufgeführten Signale am Stecker X100 verfügbar sind!

#### 1.3.1.2.2 Sollwertkanal für Schrittmotorachsen

Zur Ansteuerung einer Schrittmotor-Leistungsbaugruppe stehen vier Ausgangssignale zur Verfügung. Dies sind ein Pulssignal, ein Richtungssignal und deren invertierte Signale nach EIA Standard RS422. Alle Ausgänge liefern einen typischen Ausgangsstrom von -60mA (max. -150mA). Die maximale Impulsfrequenz der Schrittsignale beträgt 12.5MHz.

**Achtung:** Maßgeblich für die korrekte Anzahl auszuführender Schritte ist die positive Flanke des Schrittsignals PULS4+ bzw. die negative Flanke des Schrittsignals PULS4-.

##### 1.3.1.2.2.1 Pinbelegung Stecker X100, Achskanal 4

Pin	Name	Gruppe	Beschreibung
1	PULS4+	Stepper 4	Pulssignal
2	PULS4-	Stepper 4	Pulssignal invertiert
7	SIGN4+	Stepper 4	Richtungssignal
8	SIGN4-	Stepper 4	Richtungssignal invertiert

**Achtung:** Die Stiftleiste J1 muß in Stellung 1-2 und J2 in Stellung 1-2 gebrückt sein, damit die oben aufgeführten Signale am Stecker X100 verfügbar sind!

1.3.1.3 Pinbelegung Stecker X100, Digitale Eingänge

Die Prinzipschaltbilder der nachfolgend aufgelisteten digitalen Eingänge I17..I19 sind im [IHB / Kapitel 4.1.2.5] und des Eingangs I20 im [IHB / Kapitel 4.1.2.6] abgedruckt.

Eingang Ix / Pin	Funktion
I17 / 18	Dig. Eingang 17
I18 / 19	Dig. Eingang 18
I19 / 20	Dig. Eingang 19
I20 / 21	Dig. Eingang 19 und Hardware-Latchsignal zum Speichern der Istposition Achskanal 4

1.3.1.4 Pinbelegung Stecker X100, Digitale Ausgänge

Die Prinzipschaltbilder der nachfolgend aufgelisteten digitalen Ausgänge O9..O12 sind im [IHB / Kapitel 4.1.2.7] abgedruckt.

Ausgang Ox / Pin	Funktion
O9 / 22	Dig. Ausgang 1
O10 / 23	Dig. Ausgang 2
O11 / 24	Dig. Ausgang 3
O12 / 25	Dig. Ausgang 4

1.3.1.5 Pinbelegung Stecker X100, Freigaberelais

Am Steckverbinder X100 werden Relaiskontakte für die Verstärkerfreigabe zur Verfügung gestellt. Es handelt sich hierbei um Schließer. Das Relais ist nach Einschalten des PC, nach einem Rücksetzvorgang oder nach einem Fehler abgeschaltet.

Das Freigaberelais wird beim PCAP-Befehl *cl()* und beim SAP-Befehl *CL()* für den entsprechend selektierten Achskanal aktiviert.

Pin	Funktion
40	Relais S1, P-Kontakt
41	Relais S1, Schließer, Freigabe für Leistungsverstärker Achskanal 4

1.3.1.6 Impulserfassungschanäle

Der OPMF-1002 ist mit einem Impulserfassungschanal ausgestattet, an welchen unterschiedliche Enkodertypen wie beispielsweise Längenmaßstäbe oder Inkremental- oder Absolut-Drehgeber angeschlossen werden können. Als Eingangssignale werden zwei um 90° phasenverschobene Quadratursignale mit einer maximalen Impulsfrequenz von 5.0MHz und TTL-Pegel verarbeitet. Eine Nullspur (Indexsignal) kann ebenfalls ausgewertet werden. Die von den Enkodern erfaßten Signalpegel werden elektronisch vervierfacht und intern mit einer Auflösung von 32bit (31Bit plus Vorzeichen) verarbeitet. Mit diesem Wertebereich ergibt sich z.B. ein Verfahrweg von über 2000m mit einer Auflösung von 1µm.

1.3.1.6.1 Inkremental-Enkoder mit invertierten Signalen

Die Inkremental-Enkoder mit symmetrischen Ausgängen sind besonders für den industriellen Einsatz geeignet, da die Ausgangssignale mit invertiertem und nichtinvertiertem Signalpegel aller Spuren zur Verfügung stehen. Dies ermöglicht eine zuverlässige Impulserfassung auch in elektromagnetisch stark gestörter Umgebung.

Beim Einsatz dieses Enkodertyps dürfen J9, J10 und J11 für Kanal 4 nicht gebrückt werden.

### 1.3.1.6.2 Inkremental-Encoder ohne invertierte Signale

Es ist auch möglich, Inkremental-Encoder ohne invertierte Impulsfolgen zu verarbeiten. Jedoch sollten diese nur in elektromagnetisch wenig gestörter Umgebung, z.B. in Laboranwendungen, eingesetzt werden. Ebenso ist zu beachten, daß die Leitungslänge des Encoderkabels gerade bei hohen Impulsfrequenzen nur wenige Meter betragen darf.

Sofern am Kanal 4 dieser Enkodertyp angeschaltet wird, müssen J9, J10 und J11 gebrückt werden. Die Encodersignale müssen in diesem Fall an den Pins 4, 6 und 8 von X100 angeschlossen.

### 1.3.1.6.3 Optische Entkopplung der Impulserfassungskanäle

Alle Impulserfassungskanäle des OPMF-1002 sind optisch entkoppelt. Dies ist insbesondere in elektromagnetisch stark gestörter Umgebung von Vorteil.

### 1.3.1.6.4 Steckerbelegung für die Impulserfassungskanäle mit Inkrementalgebern

#### 1.3.1.6.4.1 Steckerbelegung X100, Kanal 4

Pin	Name	Funktion
3	CHA4+	Inkrementalsignal (TTL-Rechteck-Impulsfolgen) Spur A
4	CHA4-	invertiertes Inkrementalsignal Spur A
5	CHB4+	Inkrementalsignal Spur B mit 90° elektrischem Phasenversatz zu Spur A
6	CHB4-	invertiertes Inkrementalsignal Spur B
7	NDX4+	Referenzsignal Spur 0
8	NDX4-	Invertiertes Referenzsignal Spur 0

1.3.1.6.5 Steckerbelegung für die Analogeingangskanäle

Der OPMF-1002 kann mit bis zu 12 Analogeingängen ausgeliefert werden. Jeweils 4 Kanäle sind zu einer Gruppe zusammengefasst. Für jede Gruppe kann die Auflösung 12bit oder 16bit und der Meßwertbereich 0..10V, -5..+5V oder -10..10V individuell eingestellt werden.

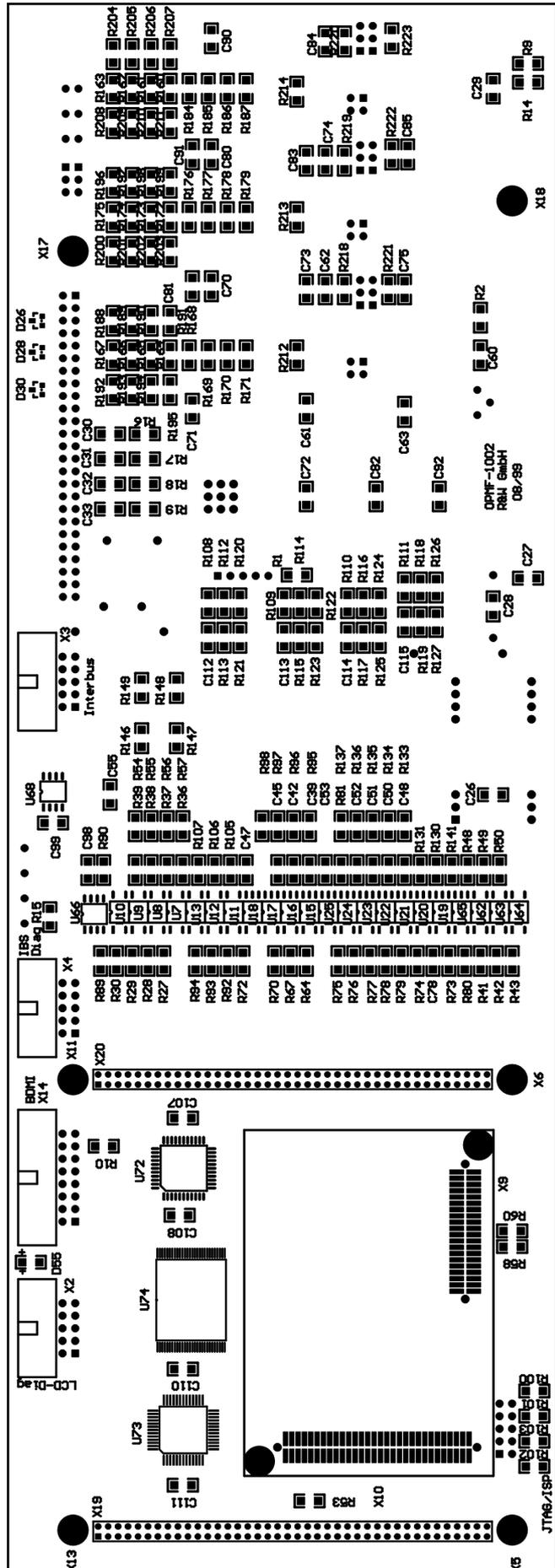
Pin	Name	AD-Gruppe	Funktion
9	AIN1+	1	Analog-Eingang 1
10	AIN1-	1	Bezugs-Potential für AIN1+
11	AIN2+	1	Analog-Eingang 2
12	AIN2-	1	Bezugs-Potential für AIN2+
13	AIN3+	1	Analog-Eingang 3
14	AIN3-	1	Bezugs-Potential für AIN3+
15	AIN4+	1	Analog-Eingang 4
16	AIN4-	1	Bezugs-Potential für AIN4+
26	AIN5+	2	Analog-Eingang 5
27	AIN5-	2	Bezugs-Potential für AIN5+
28	AIN6+	2	Analog-Eingang 6
29	AIN6-	2	Bezugs-Potential für AIN6+
30	AIN7+	2	Analog-Eingang 7
31	AIN7-	2	Bezugs-Potential für AIN7+
32	AIN8+	2	Analog-Eingang 8
33	AIN8-	2	Bezugs-Potential für AIN8+
42	AIN9+	3	Analog-Eingang 9
43	AIN9-	3	Bezugs-Potential für AIN9+
44	AIN10+	3	Analog-Eingang 10
45	AIN10-	3	Bezugs-Potential für AIN10+
46	AIN11+	3	Analog-Eingang 11
47	AIN11-	3	Bezugs-Potential für AIN11+
48	AIN12+	3	Analog-Eingang 12
49	AIN12-	3	Bezugs-Potential für AIN12+

1.3.1.6.6 Steckerbelegung für die Referenzspannungsausgänge

Der OPMF-1002 kann mit bis zu 3 Referenzspannungsausgängen ausgeliefert werden. Diese Ausgänge sind den oben genannten AD-Gruppen zugeordnet. Alle Ausgänge sind auf 10V eingestellt.

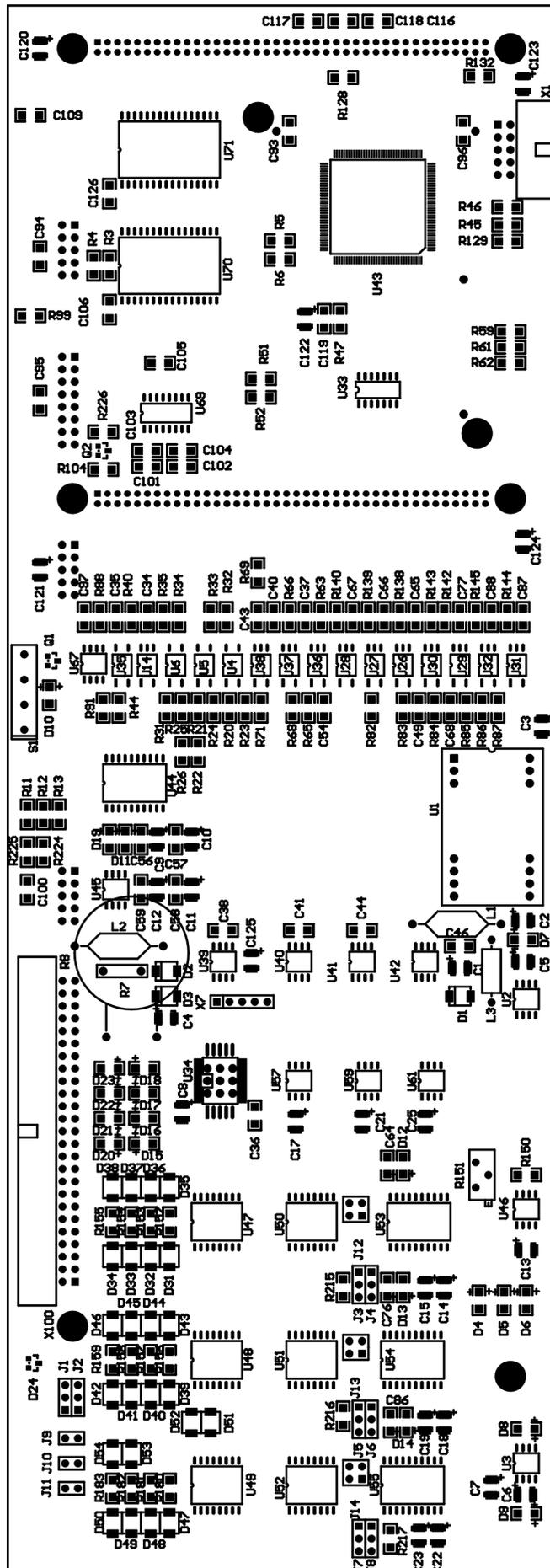
Pin	Name	AD-Gruppe	Funktion
34	AGND1	1	Bezugs-Potential für Referenz-Spannungsausgang 1
35	AREF1	1	Referenz-Spannungsausgang 1
36	AGND2	2	Bezugs-Potential für Referenz-Spannungsausgang 2
37	AREF2	2	Referenz-Spannungsausgang 2
38	AGND3	3	Bezugs-Potential für Referenz-Spannungsausgang 3
39	AREF3	3	Referenz-Spannungsausgang 3

### 1.3.2 Bestückungsdruck des OPMF-1002 Top-Layer



Layer 1 (TOP)

### 1.3.3 Bestückungsdruck des OPMF-1002 Bottom-Layer



Layer 6 (Bottom)

### 1.3.4 Technische Daten des OPMF-1002

Achskanäle:	Einer. Gemischter Betrieb von Servo- oder Schrittmotoren möglich
Gebereingänge:	Richtungsdiskriminator für Inkrementalgeber mit 2 um 90° phasenverschobenen Impulsspuren und Nullimpuls, wahlweise deren invertierte Impulsspuren (6 Kanäle) SSI-Absolutwertgeber
Impulspegel:	5V, TTL
Inkremental-Geberauswertung:	4 fach, 32 bit mit Vorzeichen, 5.0 MHz (20MHz nach Vervierfachung)
SSI-Geberauswertung:	1..32bit, Gray-/Binär-Codes, variable Frequenz 30kHz .. 10MHz
Geberversorgung:	externe Hilfsspannung je nach Gebertyp (5..30V)
Sollwertausgänge für Servo-Leistungsendstufen:	14(16)-Bit-DA-Wandler, +/-10V, 5mA, potentialfrei
Sollwertausgänge für Schrittmotor-Endstufen:	RS422-Puls- und Richtungssignale und deren invertierte Impulsfolgen, Ausgangsstrom typisch: -60mA (max. -150mA) Impulsfrequenz: max. 12.5MHz
Digitale Eingänge:	4 Eingänge optisch entkoppelt 18..36V, Eingangsstrom bei 24V ca. 10mA. Funktionsweise frei programmierbar
Digitale Ausgänge:	4 Ausgänge optisch entkoppelt, Ausgangstyp: PNP 24V, 500mA (interne Strombegrenzung bei 1A) Funktionsweise frei programmierbar, Sollzustand nach Reset programmierbar
Analoge Eingänge:	4, 8 oder 12, differentiell, optisch entkoppelt, Auflösung 12bit (optional 16bit) Eingangsspannungen: -5..5V, -10..+10V, 0..+10V
Externe Stromversorgung:	24V Stromaufnahme je nach Belastung der Digital-Ausgänge
Aufbau:	Aufsteckplatine 110mm * 318mm, 6fach-Multilayer
PC-Stromversorgung:	5V / ca. 0.5A in Maximalausbaustufe
Anschlußstecker:	50-poliger FB-Steckverbinder für komplette Peripherie-Anschaltung

## 2 Option IBS

### 2.1 Kurzbeschreibung

Die Option IBS dient zur Systemerweiterung der Positionier- und Bahnsteuerung MCU-3T. Sie ermöglicht die Anschaltung an den international genormten Feldbus **INTERBUS**. Auf dem Optionsprint OPMF-1002 ist eine Master-Anschaltbaugruppe der Generation 4 eingebettet mit deren Unterstützung eine einfache und flexible Integration in die Interbus-Feldebene realisiert werden kann. Im Moment werden bis zu 32 Interbus-Teilnehmer durch die *rwos*-Betriebssystemfirmware unterstützt. Somit kann die MCU-3T auf einfachste Weise mit neuen E/A-Funktionen ausgestattet werden. Am Markt ist hierzu eine große Anzahl unterschiedlichster Feldbusgeräte, z.T. sogar mit intelligenten Vorverarbeitungsfunktionen, verfügbar.

**Anmerkung:** zum besseren Verständnis der nachfolgend beschriebenen Softwarefunktionen sollte das INTERBUS User Manual Generation 4 Controller Boards (Artikel-Nr.: IBS SYS FW G4 UM E, Bestell-Nr.: 2745 18 5) der Firma Phoenix Contact herangezogen werden.

### 2.2 Software

Die Softwarefunktionen zur Unterstützung der Option IBS werden aufbauend auf dem Programmierhandbuch [PHB] der MCU-3T und dem INTERBUS User Manual G4 Controller Boards (s.o.) in den nachfolgenden Kapiteln beschrieben. Dieses Handbuch wird im weiteren Verlauf mit IUM bezeichnet.

#### 2.2.1 Einführung

Nachfolgend wird erläutert mit welchen Methoden die IBS-Option softwaremäßig bedient wird. Um eine möglichst flexible und schnelle Konfiguration des vorgegebenen INTERBUS-Aufbaus zu realisieren, wurde folgender Weg beschritten. Der Anwender definiert zunächst einen Busaufbau mit allen notwendigen Feldbusgeräten. Mit einem im Lieferumfang enthaltenen Konfigurationsprogramm wird der am OPMF-1002 angeschlossene Busaufbau automatisch ermittelt. Die eingelesene Konfiguration wird in zwei Dateien automatisch abgespeichert. Dies sind ein ASCII-File und ein Binärfile. Mit Hilfe des ASCII-Files kann unter Zuhilfenahme eines Texteditors der vorhandene BUS-Aufbau kontrolliert werden. Zusätzlich sind dort weitere wichtige Informationen für die SAP-Programmierung enthalten. Das Binärfile, in welchem der Sollzustand gespeichert ist, wird zum Vergleich mit der tatsächlichen Buskonfiguration verwendet. Somit ist sichergestellt, daß ein veränderter Busaufbau in der endgültigen Anwendung nicht aktiviert werden kann.

## 2.2.2 Konfigurationssoftware *ibscfg.btl*

Mit Hilfe der Betriebssystemsoftware *ibscfg.btl* kann der am OPMF-1002 angeschlossene Busaufbau automatisch ermittelt werden, sofern sich der angeschaltete Bus in einem ordnungsgemäßen Zustand befindet. Die Konfigurationssoftware wird wie folgt auf die MCU-3T übertragen und gestartet:

```
mcbt /BB ibscfg.btl (Anmerkung: MCU-3T wird zurückgesetzt)
```

Nach einer kurzen Pause sollte die Meldung:

```
Bin-File <IBSCFG.CFG> and ASCII-File <IBSCFG.TXT>
successfully generated.
```

auf dem Bildschirm erscheinen. Mit Hilfe eines Texteditors kann die ermittelte Buskonfiguration überprüft werden. Nachfolgend ist der Inhalt einer IBSCFG.TXT – Datei abgedruckt.

### Beispiel eines ASCII-Files *IBSCFG.TXT*

```
IBSCFG REV 1.0. Date: Thu May 14 17:55:37 1998
INTERBUS Configuration:
-----
Device Bus.SegNo. Position LengthCode IdCode Dev.Level Log.Group.Number
-----
01      01          00          01          0b          0000          ffff
02      02          00          01          03          0000          ffff
03      03          00          00          08          0000          ffff
04      03          01          01          be          0001          ffff
05      03          02          81          ba          0001          ffff
06      03          03          01          bd          0001          ffff
-----
Device          SAP-Variable      Access-Type      Length
-----
01              IBSPD0            Input            16 Bit
02              IBSPD1            Input            16 Bit
04              IBSPD3            Input            16 Bit
05              IBSPD4            Input            8 Bit
01              IBSPD0            Output           16 Bit
02              IBSPD1            Output           16 Bit
06              IBSPD5            Output           16 Bit
```

Alle vorhanden Geräte erhalten eine fortlaufende Nummer. Diese Nummer entspricht der tatsächlichen Position im Busaufbau. Die Spalte *Bus.SegNo.* gibt an zu welchem Bussegment das jeweilige Gerät gehört. Die Spalte *Position* wiederum dient zur Ermittlung der Position innerhalb eines Bussegmentes. *LengthCode* zeigt an welcher Adressraum durch das jeweilige Gerät im Host benötigt wird. Die Spalte *IDCode* dient zur Beschreibung der Gerätefunktion. Normalerweise sind die ID-Codes auf den Interbus-Modulen in dezimaler und bei RT-Modulen in hexadezimaler Schreibweise aufgedruckt. Die Spalte *Dev.Lev* gibt die Ebene an, auf der sich der jeweilige Teilnehmer befindet.

Für die SAP-Programmierung sind ebenfalls wertvolle Informationen in dieser Datei enthalten. Dazu gehören zunächst alle Variablen, mit deren Hilfe die jeweiligen Feldbusgeräte angesprochen werden können. Dies sind IBSPD0 .. IBSPDx (alle Typ Integer), wobei *x* von der tatsächlichen Anzahl der Interbus-Teilnehmer abhängt. Die Spalte *SAP-Variable* enthält die jeweiligen Systemvariablen für die SAP-Programmierung. Mit Hilfe der Spalte *Access-Type* kann ermittelt werden, wie auf die jeweilige Systemvariable zugegriffen werden darf also lesend, schreibend oder beides. Die Spalte *Length* zeigt an, wieviel Bits in der jeweiligen SAP-Variable nutzbar sind.

### 2.2.3 SAP-Programmierung

Die Software-Zusatzfunktionen der Option IBS werden bis auf wenige Ausnahmen durch die SAP-Programmiermethode unterstützt.

**Achtung:** Zugriffe auf die IBS-Funktionalität sollten nur aus einer Task erfolgen. Ansonsten muß gewährleistet werden, daß nur sequentiell (nicht gleichzeitig) aus mehreren Tasks auf den Optionsprint zugegriffen wird. Ist dies nicht gewährleistet, kann dies zu einem unerklärlichen Laufzeitverhalten führen!

#### 2.2.3.1 System-Parameter IBSCS0 .. IBSCS9

Mit Hilfe diverser Systemvariablen IBSCS (alle Typ Integer) können verschiedene Firmware-Kommandos des Interbus-Masters aufgerufen werden. Zusätzlich können diverse Systemzustände des INTERBUS und Fehlerzustände mit Hilfe dieser Variablen abgefragt werden.

Sytem-Parameter	Modus	Wert / Opcode	Beschreibung
IBSCS0	Lesen		Lesen des INTERBUS diagnostic status registers (IUM / Kapitel 1.3.2)
	Schreiben	0	1. Driver-Error (->IBSCS9) löschen. 2. Interne Treiber-Initialisierung. <u>Muß</u> unmittelbar beim Programmstart ausgeführt werden, im Multitaskbetrieb in allen relevanten Tasks. Darf mehrfach ausgeführt werden.
		1	Projektierte Buskonfiguration auf IBS-Masterboard laden und starten. Im Detail werden folgende Fimware-Kommandos aufgerufen (alle Kommandos hex): 1. Control Parameterization Start (030E) 2. Initiate Load Configuration (0306) 3. Complete Load Configuration (030A) 4. Terminate Load Configuration (0308) 5. Compact Load Process Data Reference List Inputs (0328) 6. Compact Load Process Data Reference List Outputs (0328) 7. Control Parameterization Stop (030E) 8. Activate Configuration (0711) 9. Start Data Transfer (0701) <u>Anmerkung:</u> Bevor dieses Kommando ausgeführt werden kann, muß dafür gesorgt sein, daß des Konfigurationsfile <i>ibscfg.txt</i> auf die Steuerung übertragen wurde [Kapitel 2.2.4]!
		2	Alarm Stop Service (Kommando 1303) ausführen.
		3	Confirm Diagnostics Service (0760)
		4	Start Data Transfer (0701)
		5	Stop Data Transfer (0702)
		50	Interne Variable (Driver Error -> IBSCS9) löschen.
		IBSCS1	Lesen
	Schreiben	0	LED auf OPMF1002 ausschalten
	Schreiben	1	LED auf OPMF1002 einschalten
IBSCS9	Lesen		Interne Variable (Driver-Error) lesen. Diese Variable zeigt interne Fehlerzustände an und sollte immer den Wert 0 haben. Sofern IBSCS9 den Wert 1000 annimmt, wurde ein ungültiges Kommando ausgeführt oder ein Lesebefehl mit einem ungültigen System-Parameter aufgerufen.

Anmerkung: Im SAP-Anwenderprogramm sollte der INTERBUS durch zyklisches Auswerten der System-Parameter IBSCS0 und IBSCS1 überwacht werden.

### 2.2.3.2 System-Parameter IBSPD0 .. IBSPD31

Mit Hilfe der Systemvariablen IBSPD0 .. IBSPD31 können Prozeßdaten des INTERBUS eingelesen und ausgegeben werden. Da es sich bei diesen Parametern um Integerwerte mit 32bit Wortbreite handelt, können alle handelsüblichen Feldbusgeräte auch mit unterschiedlichen Wortbreiten angesprochen werden. Die effektiv nutzbare Wortbreite der jeweiligen Systemparameter wird einerseits durch das Feldbusgerät selbst und andererseits durch die vorgegebene Bus-Konfiguration bestimmt. Hierzu sind die Erläuterungen im Kapitel 2.2.2 zu beachten.

## 2.2.4 PCAP-Programmierung

Wie im Kapitel 2.2.1 schon erwähnt kann mit Hilfe des Konfigurationsprogrammes *ibscfg.btl* der aktuelle Busaufbau automatisch ermittelt werden. Unter anderem wird dabei auch ein Binärfile *ibscfg.cfg* erzeugt. Sofern Zugriffe auf die Interbusfeldgeräte erfolgen sollen muß dieses Binärfile mit Hilfe des *txbf()*-Ladebefehls [PHB] an die Steuerung übertragen werden. Dieses Konfigurationsfile wird zum Vergleich mit dem tatsächlich vorhandenen Busaufbau herangezogen, sobald der Wert 1 auf die Systemvariable IBSCS0 geschrieben wird [Kapitel 2.2.3.1]. Ab diesem Moment sollte sich der Interbus im Zustand RUN befinden. Die Zugriffe auf die IBS-Systemparameter sind unter Windows 95/98/NT auch per PCAP-Programmierung mit Delphi oder C möglich. Hierzu ist die jeweilige Deklarationsdatei *specfunc.h* bzw. *specfunc.bas* erforderlich.

### 2.2.4.1 rdibscs, read IBS command/status registers

DELPHI:                   function rdibscs (index:longint): longint;

C:                        long rdibscs (long index);

BESCHREIBUNG:        Mit dieser Funktion werden die Interbus diagnostic status registers gelesen (siehe Abschnitt System-Parameter IBSCS0..IBSCS9). Mit dem Parameter *index* wird der Index des Registers (0..9) angegeben.

RÜCKGABEWERT:        Inhalt des angegebenen Registers.

### 2.2.4.2 wribscs, write IBS command/status registers

DELPHI:                   procedure wribscs (index:longint; Data:longint);

C:                        void wribscs (long index, long Data);

BESCHREIBUNG:        Mit dieser Funktion werden die Interbus diagnostic status registers mit dem Wert *Data* beschrieben (siehe Abschnitt System-Parameter IBSCS0..IBSCS9). Mit dem Parameter *index* wird der Index des Registers (0..9) angegeben.

RÜCKGABEWERT:        keiner

### 2.2.4.3 rdibsdata, read IBS data registers

DELPHI:	function rdibsdata (index:longint): longint;
C:	long rdibsdata (long index);
BESCHREIBUNG:	Mit dieser Funktion werden die Interbus Prozessdaten gelesen (siehe Abschnitt System-Parameter IBSPD0..IBSPD31). Mit dem Parameter <i>index</i> wird der Index des Registers (0..9) angegeben.
RÜCKGABEWERT:	Inhalt des angegebenen Registers.

### 2.2.4.4 wribsdata, write IBS data registers

DELPHI:	procedure wribsdata (index:longint; Data:longint);
C:	void wribsdata (long index, long Data);
BESCHREIBUNG:	Mit dieser Funktion werden die Interbus Prozessdaten mit dem Wert <i>Data</i> beschrieben (siehe Abschnitt System-Parameter IBSPD0..IBSPD31). Mit dem Parameter <i>index</i> wird der Index des Registers (0..9) angegeben.
RÜCKGABEWERT:	keiner

## 2.3 Hardware

Die Option IBS ist auf dem Tochterboard OPMF-1002 für die MCU-3T Grundplatine realisiert. Die beiden Baugruppen werden mit Steck- und Schraubverbindungen werkseitig assembliert. Elektrisch wird nur ein ISA PC-Steckplatz für die Steuerung benötigt. Aufgrund des gegebenen mechanischen Aufbaus werden jedoch zwei ISA PC-Steckplätze belegt. Die Peripheriegeräte werden mit Hilfe von 10poligen FB-Steckverbindern auf dem OPMF-1002 angeschlossen. Für den OPMF-1002 ist u.a. auch ein Kabelsatz mit Kartenhalter mit 2 integrierten 9pol. SUB-D-Steckverbindern erhältlich. Dieser Kartenhalter kann in einen freien PC-Slot neben der Steuerung eingebaut werden.

Anmerkung: Im Normalfall wird nur Stecker X3 (Interbus) benötigt.

### 2.3.1 Optionsprint OPMF-1002, Anschlußbelegungen

#### 2.3.1.1 Stecker X1 (JTAG / ISP): 10-poliger FB-Steckverbinder

Dieser Stecker dient zur Programmierung (JTAG / ISP) der Hardwarelogik auf dem OPMF-1002. Dieser Programmiervorgang wird jedoch nur werkseitig vorgenommen. Zur Nutzung dieser Schnittstelle sind entsprechende Hardwareschnittstellen und Softwaretools erforderlich.

#### 2.3.1.2 Stecker X2 (LCD Diag): 10-poliger FB-Steckverbinder

An diesem Stecker kann eine LCD-Diagnose-Anzeige angeschlossen werden. Mit dieser ist es möglich verschiedene Informationen des INTERBUS-Masters abzufragen. Dazu gehören z.B. Busaufbau, Fehleranzeige, Störungen u.a.

#### 2.3.1.3 Stecker X3 (Interbus): 9-poliger SUB-D-Steckverbinder (Buchse)

Am Stecker X3 wird der Fernbus (2-Leiter) des IBS-Systems angeschlossen. Sie ist am Kartenhalter als 9-polige D-SUB-Buchse ausgeführt und galvanisch vom Potential des Host-PCs getrennt. Das Gehäuse der Buchse ist leitend mit dem PC-Kartenhalter verbunden.

Tabelle: Anschlußbelegung der Fernbusschnittstelle (SUB-D)

Pin	Name	Beschreibung
1	DO	Fernbus Sendeleitung
2	DI	Fernbus Empfangsleitung
3	COM	Masse
4		Nicht belegt / reserviert
5	+5V	Interne Versorgungsspannung
6	/DO	Fernbus Sendeleitung invertiert
7	/DI	Fernbus Empfangsleitung invertiert
8		Nicht belegt / reserviert
9		Nicht belegt / reserviert

2.3.1.4 Stecker X4 (IBS / Diag): 9-poliger SUB-D-Steckverbinder (Stift)

Am Stecker X4 ist eine Diagnoseschnittstelle mit RS232-Pegel realisiert. Hier kann ein PC als Diagnosegerät angeschlossen werden. Hierbei kann beispielsweise das Windows-Programm INTERBUS-CMD G4 zum Einsatz kommen. Mit dieser Software können sowohl Inbetriebnahme- Projektierung- und Diagnose-Funktionen ausgeführt werden. Weiterhin ist es möglich einen Firmware-Download über diese Schnittstelle vorzunehmen. Die Diagnoseschnittstelle ist am Kartenhalter als 9-polige D-SUB-Stiftleiste ausgeführt. Die Diagnoseschnittstelle kann mit Hilfe eines 1:1 Verbindungskabels (keine Überkreuzung notwendig) mit dem PC verbunden werden.

Zur Verbindung des IBF-1002 mit einem PC als Diagnose- oder Programmiergerät sind nur die Signale TXD, RXD und GND notwendig.

Tabelle: Anschlußbelegung der Diagnoseschnittstelle (SUB-D)

Pin	Name	Beschreibung
1		Reserviert
2	TXD	Transmitted Data
3	RXD	Received Data
4		Reserviert
5	GND	Digital Ground
6		Reserviert
7	CTS	Clear to send (nicht Firmwareunterstützt)
8	RTS	Request to send (nicht Firmwareunterstützt)
9		Reserviert

2.3.1.5 Stecker X14 (BDMI): 9-poliger SUB-D-Steckverbinder (Stift)

Dieser Steckverbinder dient zum ersten Firmware-Setup des Interbus-Masterboards und wird nur werkseitig genutzt. Zur Nutzung dieser Schnittstelle sind entsprechende Hardwareschnittstellen und Softwaretools erforderlich.

### 2.3.2 Technische Daten der Option IBS

Max. Anzahl	
Interbus-Teilnehmer:	256
Max. Anzahl I/O:	2048
Aufbau:	Aufsteckplatine 110mm * 318mm, 6fach-Multilayer
PC-Stromversorgung:	5V / ca. 0.3A
Anschlußstecker:	10-poliger FB-Steckverbinder für Interbus-Peripherie-Anschaltung, 10-poliger FB-Steckverbinder für Diagnoseschnittstelle RS232 10-poliger FB-Steckverbinder für Diagnose-Display 10-poliger FB-Steckverbinder für Programmierung von FPGA 16-poliger FB-Steckverbinder für Background Debug Mode

### 3 SUB-D-Adapter von 10-poligem FB-Stecker X5 auf 9-poligen SUB-D-Stecker, für Freigabe-Relais, CNC-Bereit-Relais

Am Steckverbinder X5 werden Relais-Kontakte für die CNC-Bereit-Abfrage und Verstärkerfreigaben zur Verfügung gestellt. Es handelt sich hierbei um Schließer. Alle Relais sind nach Einschalten des PC, nach einem Rücksetzvorgang oder nach einem Fehler abgeschaltet.

Das CNC-Bereit-Relais wird nach dem Bootvorgang (*mcbt.exe*) aktiviert.

Die Freigabe-Relais werden beim PCAP-Befehl *cl()* und beim SAP-Befehl *CL()* für die entstprechend selektierten Achskanäle aktiviert.

Mit diesem Adapterkabel wird der Stecker X-5 auf einen 9-poligen Sub-D-Steckverbinder konvertiert. Dieser Steckverbinder ist an einem PC-Slotblech montiert. Durch die Konvertierung ergibt sich eine andere Steckerbelegung als an X5.

---

Pin	Funktion
1	Relais 1, Ruhekontakt
6	Relais 1, Schliesser, CNC Bereit
2	Relais 2, Ruhekontakt
7	Relais 2, Schliesser, Freigabe für Leistungsverstärker Achskanal 1
3	Relais 3, Ruhekontakt
8	Relais 4, Schliesser, Freigabe für Leistungsverstärker Achskanal 2
4	Relais 4, Ruhekontakt
9	Relais 4, Schliesser, Freigabe für Leistungsverstärker Achskanal 3
5	Nicht belegt

---

## 4 Optionsprint OPIO-3216

### 4.1 Kurzbeschreibung

Der Optionsprint OPIO-3216 dient zur Systemerweiterung der Positionier- und Bahnsteuerung MCU-3T. Er kann 32 digitale Eingänge und 16 digitale Ausgänge verarbeiten. Die Ein- und Ausgänge sind elektrisch voll kompatibel zu den Ein- und Ausgängen der MCU-3T Grundplatine.

### 4.2 Software

Die Software-Zusatzfunktionen des OPIO-3216 werden z. Zt. durch die SAP-Programmiermethode und für die PACP-Programmiermethode unter Win 32 per C und Delphi unterstützt.

Die Softwarefunktionen zur Unterstützung des OPIO-3216 werden aufbauend auf dem Programmierhandbuch [PHB] der MCU-3T in den nachfolgenden Kapiteln beschrieben.

#### 4.2.1 PCAP-Programmierung

Die nachfolgenden Funktionen stehen nur für die Programmierung unter Windows 95/98/NT zur Verfügung.

##### 4.2.1.1 rdopin, read option-print inputs

DELPHI:                   function rdopin (an:longint):longint;

C:                         long rdopin(long an);

BESCHREIBUNG:         Mit dieser Funktion wird der aktuelle Zustand der digitalen Eingänge des Optionsprints OPIO-3216 eingelesen. Der Parameter *an* hat i.M. keine Bedeutung, da nur Optionsprints auf der Masterplatine unterstützt werden.

Rückgabewert:         Digitales Eingangswort bitweise codiert (siehe hierzu auch Systemparameter SR0 bei der SAP-Programmierung).

##### 4.2.1.2 rdopout, read option-print outputs

DELPHI:                   function rdopout (an:longint):longint;

C:                         long rdopout(long an);

BESCHREIBUNG:         Mit dieser Funktion wird der aktuelle Zustand der digitalen Ausgänge des Optionsprints OPIO-3216 eingelesen. Der Parameter *an* hat i.M. keine Bedeutung, da nur Optionsprints auf der Masterplatine unterstützt werden.

Rückgabewert: Digitales Ausgabewort bitweise codiert (siehe hierzu auch Systemparameter SR1 bei der SAP-Programmierung).

#### 4.2.1.3 wropout, write option-print outputs

DELPHI: wropout (an:longint; Data:longint);

C: void wropout(long an, long Data);

BESCHREIBUNG: Mit dieser Funktion werden die digitalen Ausgänge des Optionsprints OPIO-3216 gesetzt/rückgesetzt. Der Parameter *an* hat i.M. keine Bedeutung, da nur Optionsprints auf der Masterplatine unterstützt werden. Mit dem Parameter *Data* wird das digitale Ausgabewort übergeben codiert (siehe hierzu auch Systemparameter SR1 bei der SAP-Programmierung).

Rückgabewert: keiner

## 4.2.2 SAP-Programmierung

### 4.2.2.1 System-Parameter SR0

Mit Hilfe der Systemvariablen SR0 (Typ Integer) können die Digital-Eingänge des OPIO-3216 wortweise abgefragt werden. Sofern ein Eingang aktiv ist, wird dies mit dem Wert 1 des jeweiligen Bit im SR0-Wort angezeigt. Die Eingänge sind alle frei programmierbar, es gibt keine festgelegten Achszuordnungen. Eine Zuweisung an SR0 bleibt ohne Wirkung.

Tabelle 5: Bitkodierter Aufbau des SR0-Wortes

Bit-Nr.	Funktion	SUB-D X100/Pin
0	Eingang 17	1
1	Eingang 18	2
2	Eingang 19	3
3	Eingang 20	4
4	Eingang 21	5
5	Eingang 22	6
6	Eingang 23	7
7	Eingang 24	8
8	Eingang 25	9
9	Eingang 26	10
10	Eingang 27	11
11	Eingang 28	12
12	Eingang 29	13
13	Eingang 30	14
14	Eingang 31	15
15	Eingang 32	16
16	Eingang 33	34
17	Eingang 34	35
18	Eingang 35	36
19	Eingang 36	37
20	Eingang 37	38
21	Eingang 38	39
22	Eingang 39	40
23	Eingang 40	41
24	Eingang 41	42
25	Eingang 42	43
26	Eingang 43	44
27	Eingang 44	45
28	Eingang 45	46
29	Eingang 46	47
30	Eingang 47	48
31	Eingang 48	49

4.2.2.2 System-Parameter SR1

Mit Hilfe der Systemvariablen SR1 (Typ Integer) können die Digital-Ausgänge des OPIO-3216 wortweise gesetzt werden. Sofern ein Ausgang gesetzt werden soll, muß das jeweilige Bit im SR1-Wort auf den Wert 1 gesetzt werden. Die Ausgänge sind alle frei programmierbar, es gibt keine festgelegten Achszuordnungen. Der momentane Zustand der Digital-Ausgänge kann durch Abfragen des SR1 System-Parameters ermittelt werden.

Tabelle 6: Bitkodierter Aufbau des SR1-Wortes

Bit-Nr.	Funktion	SUB-D X100/Pin
0	Ausgang 9	18
1	Ausgang 10	19
2	Ausgang 11	20
3	Ausgang 12	21
4	Ausgang 13	22
5	Ausgang 14	23
6	Ausgang 15	24
7	Ausgang 16	25
8	Ausgang 17	26
9	Ausgang 18	27
10	Ausgang 19	28
11	Ausgang 20	29
12	Ausgang 21	30
13	Ausgang 22	31
14	Ausgang 23	32
15	Ausgang 24	33

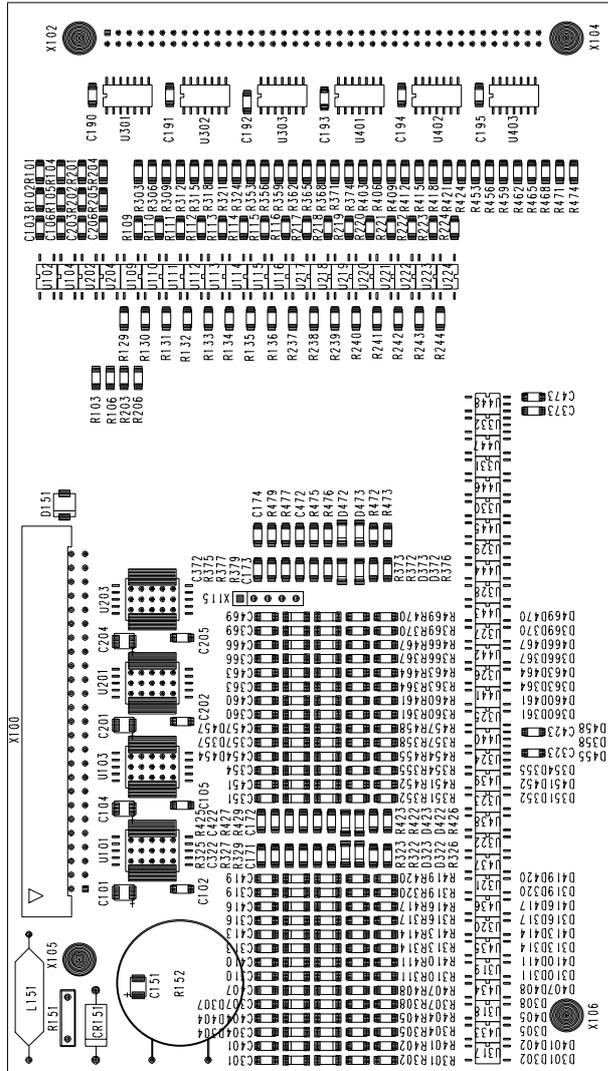
### 4.2.3 Optionsprint OPIO-3216, Anschlußbelegungen

#### 4.2.3.1 Stecker X100: 50-poliger SUB-D-Steckverbinder (Stift)

Pin	Name	Gruppe	50-pol. Flachkabel
1	I17	dig. Eingang	1
2	I18	dig. Eingang	4
3	I19	dig. Eingang	7
4	I20	dig. Eingang	10
5	I21	dig. Eingang	13
6	I22	dig. Eingang	16
7	I23	dig. Eingang	19
8	I24	dig. Eingang	22
9	I25	dig. Eingang	25
10	I26	dig. Eingang	28
11	I27	dig. Eingang	31
12	I28	dig. Eingang	34
13	I29	dig. Eingang	37
14	I30	dig. Eingang	40
15	I31	dig. Eingang	43
16	I32	dig. Eingang	46
17	+24V	Spannungsversorgung dig. Ausgänge 24V	49
18	O9	dig. Ausgang	3
19	O10	dig. Ausgang	6
20	O11	dig. Ausgang	9
21	O12	dig. Ausgang	12
22	O13	dig. Ausgang	15
23	O14	dig. Ausgang	18
24	O15	dig. Ausgang	21
25	O16	dig. Ausgang	24
26	O17	dig. Ausgang	27
27	O18	dig. Ausgang	30
28	O19	dig. Ausgang	33
29	O20	dig. Ausgang	36
30	O21	dig. Ausgang	39
31	O22	dig. Ausgang	42
32	O23	dig. Ausgang	45
33	O24	dig. Ausgang	48
34	I33	dig. Eingang	2
35	I34	dig. Eingang	5
36	I35	dig. Eingang	8
37	I36	dig. Eingang	11
38	I37	dig. Eingang	14
39	I38	dig. Eingang	17
40	I39	dig. Eingang	20
41	I40	dig. Eingang	23
42	I41	dig. Eingang	26
43	I42	dig. Eingang	29
44	I43	dig. Eingang	32
45	I44	dig. Eingang	35
46	I45	dig. Eingang	38
47	I46	dig. Eingang	41
48	I47	dig. Eingang	44
49	I48	dig. Eingang	47
50	GND-D	Spannungsversorgung dig. Ein- und Ausgänge 24V	50

Die Prinzipschaltbilder der digitalen Eingänge I17..I48 sind im [IHB / Kapitel 4.1.2.5] abgedruckt.  
 Die Prinzipschaltbilder der digitalen Ausgänge O9..O24 sind im [IHB / Kapitel 4.1.2.7] abgedruckt.

**4.2.4 Bestückungsdruck des OPIO-3216**



Boutelle

#### 4.2.5 Technische Daten des OPIO-3216

Digitale Eingänge:	32 Eingänge optisch entkoppelt 18..36V, Eingangsstrom bei 24V ca. 10mA. Funktionsweise frei programmierbar
Digitale Ausgänge:	16 Ausgänge optisch entkoppelt, Ausgangstyp: PNP 24V, 500mA (interne Strombegrenzung bei 1A) Funktionsweise frei programmierbar alle Ausgänge sind nach Reset abgeschaltet
Externe Stromversorgung:	24V Stromaufnahme je nach Belastung der Digital-Ausgänge
Aufbau:	Aufsteckplatine 110mm * 195mm, 4fach-Multilayer
PC-Stromversorgung:	5V / ca. 0.5A
Anschlußstecker:	50-poliger FB-Steckverbinder für komplette Peripherie-Anschaltung

