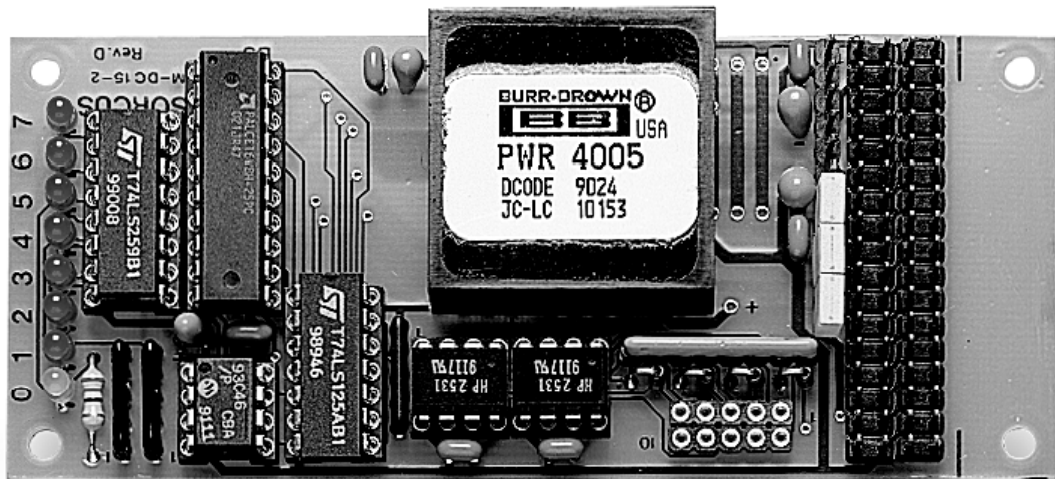


24. M-DC15-2

4 optoentkoppelte Interrupt-Eingänge,
DC/DC-Wandler für ±15 V Versorgungsspannung,
8 frei programmierbare Kontroll-LEDs



Funktionsbeschreibung	24-3
Funktionseinheiten.....	24-3
Blockschaltbild	24-4
Technische Daten	24-5
Lieferumfang.....	24-6

Konfiguration und Einbau	24-7
Lageplan M-DC15-2, Rev. D.....	24-7
Signalanpassung der Eingänge	24-8
Interrupt-Eingänge	24-8
DC/DC-Wandler	24-8

EEPROM-Inhalte	24-9
Steckerbelegung St1	24-14
Modul-Device-Treiber M-DC15-2	24-15
Installationsparameter	24-15
Kanaleigenschaftsstruktur CPS_MDC152	24-15
Digitale Eingänge.....	24-16
Zugriff	24-16
LEDs.....	24-16
Interrupt-Eingänge	24-18
Hochsprachenbibliothek	24-19
Programmierung mit I/O-Zugriffen	24-23
Lokale I/O-Adressen	24-23

Funktionsbeschreibung

Funktionseinheiten

Das Modul M-DC15-2 enthält 4 Funktionseinheiten:

- **Vier galvanisch getrennte Multifunktions-Eingänge (IP-0 bis IP-3):**

- Jeder Eingang kann per Software auf einen Interrupt-Eingang der Basiskarte geschaltet werden:

IP-0 auf IRQ-B, IP-1 auf IRQ-E, IP-2 auf IRQ-C und IP-3 auf IRQ-D.

- Der Zustand der Eingänge kann jederzeit abgefragt werden
- Die Eingangsbeschaltung kann über ein steckbares Widerstandsnetzwerk vom Anwender geändert werden und ist damit anpaßbar an Eingangsspannungen von 4 bis 27 Volt
- Min. erforderlicher Eingangsstrom (Schwelle, typ.): 5 mA
- Die maximale Pulsfrequenz am Eingang beträgt 1 MHz
- Jeder Eingang ist durch eine antiparallele Diode geschützt

Anmerkung: Für verschiedene Funktionen wird die Hardware der Basiskarte verwendet (IRQ-B, IRQ-C, IRQ-D, IRQ-E). Diese Hardware kann, auch wenn mehrere Module vom gleichen Typ aufgesteckt sind, natürlich nur einmal (zur selben Zeit) verwendet werden.

- **Acht frei programmierbare Kontroll-LEDs**

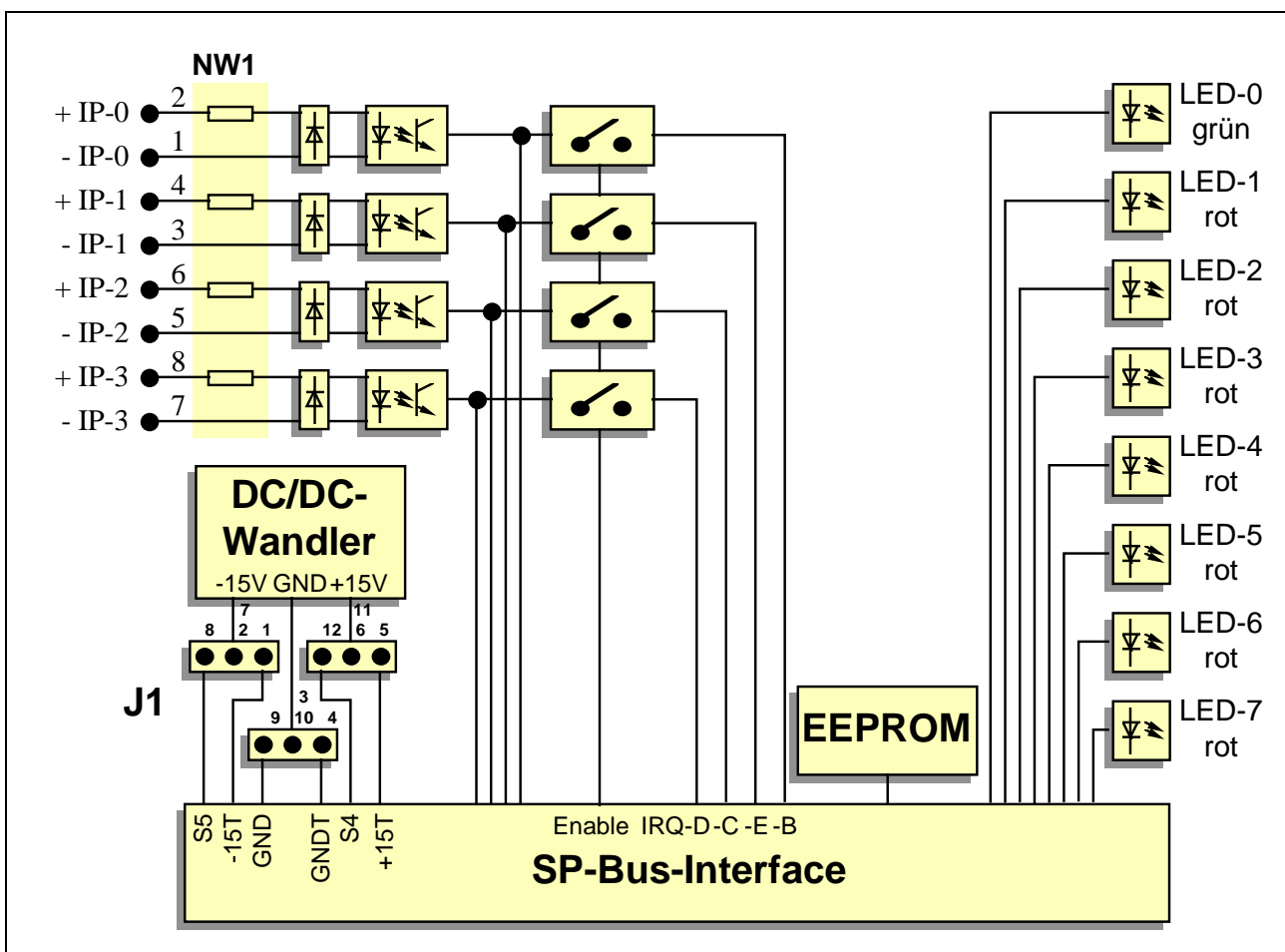
- Jede LED kann unabhängig von den anderen gesetzt und gelöscht werden, z. B. zur Kontrolle selbstgeschriebener Programme
- Setzen und Löschen der LEDs ist ohne Verwendung von CPU-Registern möglich

- **DC/DC-Wandler für Spannungsversorgung mit ± 15 Volt**

- 2, 3, 4 oder 6 Watt Leistung
- Versorgung anderer Module, die keine eigene Versorgung mit ± 15 Volt haben, über den SP-Bus
- beide Spannungen können galvanisch getrennt verwendet werden

- **EEPROM zur Abspeicherung von Initialisierungsdaten**

Blockschaltbild



Technische Daten

Parameter	Wert	Einheit
Zahl der ext. Eingänge (optoentkoppelt), davon interruptfähig (= IP-0 bis IP-3)	4 4	- -
Max. Eingangsfrequenz IP-0 bis IP-3, min. ¹	1	MHz
Eingangsstrom IP-0 bis IP-3, Schwelle (typ. / max.)	5 / 10	mA
Eingangsspannung, max. ²	27	V
Trennschaltung, max.	500	V
Reverse input voltage, max.:		
$RIV_{\max} = \frac{\text{Vorwiderstand}}{1 \Omega} \cdot 0,05$	RIV_{\max}	V
DC/DC-Wandler:		
±15 V, Genauigkeit U_{out} , typ. / min.	±3 / ±5	%
U_{out} , max.	±18	V
Temperaturkoeffizient, bezogen auf U_{out}	±0,02	%/K
Kurzschlußdauer am Ausgang, max.	5	s
Ausgangsstrom, max. (4-Watt-Typ)	±135	mA
(6-Watt-Typ)	±200	mA
Störspannung und Rauschen (peak to peak), typ.		
(4-Watt-Typ)	140	mV
(6-Watt-Typ)	70	mV
Versorgungsspannung, von der Basiskarte (±5%)	5	V
Stromaufnahme, von der Basiskarte (typ.) ³	120	mA
Betriebstemperatur (Modultemperatur), min. / max.	0 bis 70	°C
Abmessungen (L x B x H)	106 x 45 x 15	mm

Lieferumfang

¹ Auf Wunsch auch bis 10 MHz lieferbar.

² Mit externen Vorwiderständen können auch größere Eingangsspannungen verarbeitet werden. Die Begrenzung ist durch die max. erlaubte Verlustleistung der steckbaren Widerstandsnetzwerke (z. B. Bourns 4608X, 1000 mW) gegeben.

³ Stromaufnahme, gemessen bei Bestückung mit DC/DC-Wandler mit einer max. Ausgangsleistung von 6 Watt (ohne Belastung der ±15-V-Spannungen, alle LEDs ausgeschaltet).

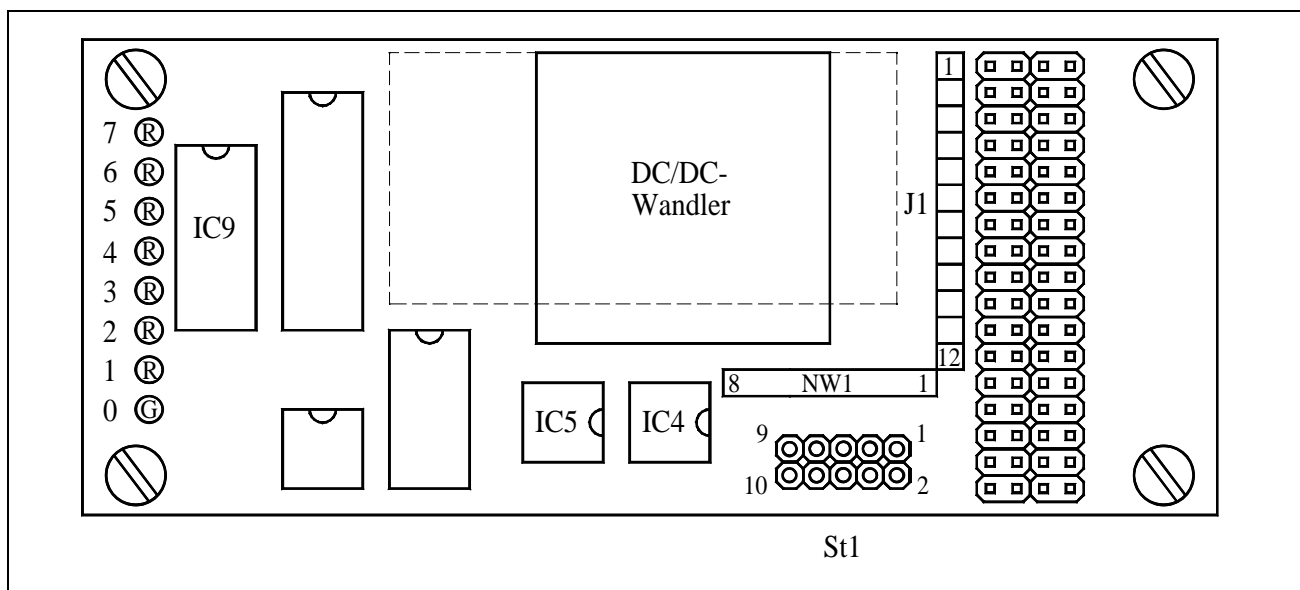
- Modul M-DC15-2
- 10-poliger Pfostenstecker für Flachbandkabel
- Sortiment Widerstandsnetzwerke WN-1531
- Datenträger mit Modul-Device-Treiber und Programmbibliotheken (Pascal und C)

Konfiguration und Einbau

Folgende Punkte sind zu beachten:

- Vor dem Einbau des Moduls ist die Signalanpassung der Eingänge mit dem Widerstandsnetzwerk NW1 vorzunehmen bzw. zu überprüfen.
- Wenn das Modul mit einem DC/DC-Wandler ausgerüstet ist und dieser verwendet werden soll, müssen mit Jumperfeld J1 die ± 15 -Volt-Spannungen und die gemeinsame Masseleitung dafür auf den SP-Bus gelegt werden.
- Alle anderen Einstellungen werden nach dem Einbau des Moduls per Software gemacht.

Lageplan M-DC15-2, Rev. D¹



Die Größe des DC/DC-Wandlers hängt von seiner maximalen Ausgangsleistung ab: die durchgezogene Linie stellt den 4-Watt-Typ, die gestrichelte den 6-Watt-Typ.

¹ Bei Rev. A ist die Reihenfolge der LEDs wie folgt (von oben nach unten): 4, 5, 6, 7, 3, 2, 1, 0

Signalanpassung der Eingänge

Die 4 Optokoppler-Eingänge können über ein steckbares Widerstandsnetzwerk (ein 8-poliges Single-In-Line-Netzwerk für alle 4 Eingänge) an unterschiedliche Eingangsbedingungen angepaßt werden. Weitere Hinweise über Art und Widerstandswert des Netzwerkes finden Sie in der Einführung im Abschnitt 'Digitaleingänge'.

Interrupt-Eingänge

IP-0 bis IP-3 können per Software auf die Interrupt-Eingänge der Basiskarte geschaltet werden. Dieser Zustand, also ob die Eingänge durchgeschaltet sind, kann auch per Software abgefragt werden.

Nach einem Hardware-Reset der Basiskarte sind die Eingänge zunächst nicht auf die Interrupt-Eingänge der Basiskarte durchgeschaltet, außer wenn es im EEPROM des Moduls in WORT-3 angegeben ist.

Moduleingang	an Interrupt-Eingang der Basiskarte
IP-0	IRQ-B
IP-1	IRQ-E
IP-2	IRQ-C
IP-3	IRQ-D

DC/DC-Wandler

Die Ausgangsspannungen von ± 15 Volt können per Jumperfeld J1 über zwei verschiedene Leitungssysteme des SP-Bus zu anderen Modulen geführt werden. In einem Fall sind die ± 15 Volt nicht galvanisch von der Basiskarte getrennt, im anderen Fall bleiben sie getrennt.

Bei dem Modul M-AD16-3 zum Beispiel, das selbst nicht für galvanische Trennung vorgesehen ist, muß die Zuführung über das nicht getrennte Leitungssystem des SP-Bus erfolgen (siehe hierzu auch Beschreibung des Moduls M-AD16-3). Die Zuführung über das galvanisch getrennte Leitungssystem ist bei diesem Modul nicht möglich.

Jumper J1 für nicht galvanisch getrenntes Leitungssystem:

Signal	an SP-Bus-Leitung	Pin auf SP-Bus	Jumper
-15 V	S5	B2	7-8
COM	GND	B17 und C17	9-10
+15 V	S4	B16	11-12

Jumper J1 für galvanische Trennung:

Signal	an SP-Bus-Leitung	Pin auf SP-Bus	Jumper
-15 V	-15T	D1	1-2
COM	GND	D2	3-4
+15 V	+15T	D17	5-6

EEPROM-Inhalte

Werkseitig ist bereits eine Konfiguration im EEPROM voreingestellt:

WORT	Binär		Hex.	Bedeutung (Kurzinfo)
0	0010 0100	0010 0010	2422h	Modultyp M-DC15-2, Rev. D ¹
1	0000 0000	0000 0001	0001h	Initialisierung
2	0000 0000	0000 0000	0000h	Bestückung
3	0000 0000	0000 0000	0000h	Einstellung von Jumper J1
4	0000 0000	0000 0000	0000h	Initialisierung
5	0000 0010	0111 0000	0270h	Bestückung des Widerstandsnetzwerks
6	0000 0000	0000 0000	0000h	Reserviert
...
31	0000 0000	0000 0000	0000h	Reserviert

¹ Bei Rev. A enthält WORT-0 2022h oder 2122h

WORT-0: Typ und Version des Moduls (darf nicht geändert werden)

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	0

WORT-0: Kennung

								0	0	1	0	0	0	1	0
								0	1	0	0				
				0											
0	0	1													

Modultyp: 34 = M-DC15-2

Revision: 1 = A, 2 = B, 3 = C, etc.

Reserviert

Kennung

WORT-1: Initialisierung

In diesem Wort kann eingestellt werden, ob das Modul nach dem Einschalten und bei einem Reset der Basiskarte entsprechend den Eintragungen im EEPROM initialisiert wird (Bit-0 = 1) oder nicht (Bit-0 = 0).

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	1

WORT-1: Initialisierung

geändert am:

von:

															1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Init nach Hardreset: 0 = nein, 1 = ja

Reserviert

WORT-4: Initialisierung von LEDs und Interrupt-Kanälen

Bit-0 bis Bit-7 bestimmen den Initialisierungszustand von LED-0 bis LED-7 nach Reset.

Bit-8 bis Bit-11 legen fest, ob der entsprechende Eingang nach Reset per Software mit dem zugehörigen Interrupt-Eingang verbunden wird oder nicht.

WORT-4: Software-Konfiguration
(werks. Einst.)

geändert am: von:

LED-0 grün (1 = ein, 0 = aus)

LED-1 rot (1 = ein, 0 = aus)

LED-2 rot (1 = ein, 0 = aus)

LED-3 rot (1 = ein, 0 = aus)

LED-4 rot (1 = ein, 0 = aus)

LED-5 rot (1 = ein, 0 = aus)

LED-6 rot (1 = ein, 0 = aus)

LED-7 rot (1 = ein, 0 = aus)

IP-0 an IRQ-B (1 = verbunden)

IP-1 an IRQ-E (1 = verbunden)

IP-2 an IRQ-C (1 = verbunden)

IP-3 an IRQ-D (1 = verbunden)

Reserviert

Steckerbelegung St1

Das Modul wird über einen 10-poligen (2 x 5) Steckverbinder und ein entsprechendes Flachbandkabel mit der Außenwelt verbunden.

Pin	Signal¹
1	- IP-0
2	+ IP-0
3	- IP-1
4	+ IP-1
5	- IP-2
6	+ IP-2
7	- IP-3
8	+ IP-3
9	n. c.
10	n. c.

¹ - entspricht der Kathode der LED des Eingangsoptokopplers, + der Anode

Modul-Device-Treiber M-DC15-2

Allgemeine Hinweise zum Umgang mit Modul-Device-Treibern finden Sie in der Einführung im Abschnitt 'Grundlagen zum Modul-Device-Treiber'.

Installationsparameter

Parameter	Wert
Dateiname:	ML8D2200.LIB
Programmnummer:	425h
Tasknummer:	Steckplatz des Moduls
Interruptnummer:	0
Länge des Datenbereichs:	0
Flags:	800h

Befehl in INS-Datei (z.B. für Steckplatz 1):

M8INST ML8D2200 0425 0001 00 000000 00000800

Kanaleigenschaftsstruktur CPS_MDC152

Ofs	Strukturelement	Datentyp	Bedeutung
0	<i>.usDevice</i>	USHORT	Device-Typ
2	<i>.usIndexFirst</i>	USHORT	Index des ersten Device
4	<i>.usIndexLast</i>	USHORT	Index des letzten Device
6	<i>.usFlags</i>	USHORT	Flags
8	<i>.usReadMode</i>	USHORT	Lesemodus
10	<i>.usWriteMode</i>	USHORT	Schreibmodus

Digitale Eingänge

Ofs	Strukturelement	Datentyp	Bedeutung
0	<i>.usDevice</i>	USHORT	Device-Typ: <i>DEVICE_DIN</i>
2	<i>.usIndexFirst</i>	USHORT	Index des ersten Device (= 0 bis 3)
4	<i>.usIndexLast</i>	USHORT	Index des letzten Device (= <i>.usIndexFirst</i> oder 3)
6	<i>.usFlags</i>	USHORT	Mögliche Flags: Bit 0: <i>_CP_EXCLUSIVE</i> = 1 (exklusiv) Alle anderen Bits = 0.
8	<i>.usReadMode</i>	USHORT	Wert zurücklesen: <i>IO_MODE_DIRECT</i> = 1 (direkt)

Alle anderen Strukturelemente werden nicht ausgewertet.

Zugriff

Die Eingänge können mit der Funktion **mddx_read_channel_byte** gelesen werden.

LEDs

Ofs	Strukturelement	Datentyp	Bedeutung
0	<i>.usDevice</i>	USHORT	Device-Typ: <i>DEVICE_LED</i>
2	<i>.usIndexFirst</i>	USHORT	Index des ersten Device (= 0)
4	<i>.usIndexLast</i>	USHORT	Index des letzten Device (= 7)
6	<i>.usFlags</i>	USHORT	Mögliche Flags: Bit 0: <i>_CP_EXCLUSIVE</i> = 1 (exklusiv) Alle anderen Bits = 0.
8	<i>.usReadMode</i>	USHORT	Wert zurücklesen: <i>IO_MODE_RAM</i> = 4 (aus RAM)
10	<i>.usWriteMode</i>	USHORT	Wert ausgeben: <i>IO_MODE_DIRECT</i> = 1 (direkt)

Die Ausgänge werden durch einen Schreibzugriff (BYTE) auf den Kanal gesteuert. Das entsprechende Byte setzt sich aus folgenden Flags zusammen:

<code>_MDC152_LED_0_ON</code>	(= 1)	Schaltet die LED-0 ein
<code>_MDC152_LED_1_ON</code>	(= 2)	Schaltet die LED-1 ein
<code>_MDC152_LED_2_ON</code>	(= 4)	Schaltet die LED-2 ein
<code>_MDC152_LED_3_ON</code>	(= 8)	Schaltet die LED-3 ein
<code>_MDC152_LED_4_ON</code>	(= 16)	Schaltet die LED-4 ein
<code>_MDC152_LED_5_ON</code>	(= 32)	Schaltet die LED-5 ein
<code>_MDC152_LED_6_ON</code>	(= 64)	Schaltet die LED-6 ein
<code>_MDC152_LED_7_ON</code>	(= 128)	Schaltet die LED-7 ein

Sollen alle LED-Ausgänge abgeschaltet werden, so muß eine 0 in den Kanal geschrieben werden. Sollen z.B. die LED-Ausgänge LED-0 und LED-1 eingeschaltet werden, so werden die Flags `_MDC152_LED_0_ON` und `MDC152_LED_1_ON` ODER-verknüpft und das Ergebnis in den Kanal geschrieben.

Der aktuelle Status der Ausgänge kann durch einen Lesezugriff (BYTE) auf den Kanal ermittelt werden. Die Prüfung kann durch eine UND-Verknüpfung des entsprechenden Flags mit dem gelesenen Byte erfolgen.

Interrupt-Eingänge

Diese Kanäle ermöglichen es, die Interrupt-Eingänge IP-0 bis IP-3 mit den Interruptleitungen IRQ-B, IRQ-E, IRQ-C bzw. IRQ-D zu verbinden oder zu trennen.

Ofs	Strukturelement	Datentyp	Bedeutung
0	<i>.usDevice</i>	USHORT	Device-Typ: <i>DEVICE_CTRL</i> = 0401h
2	<i>.usIndexFirst</i>	USHORT	Device-Index: 0 = IRQ-B 1 = IRQ-E 2 = IRQ-C 3 = IRQ-D
4	<i>.us IndexLast</i>	USHORT	= <i>.usIndexFirst</i>
6	<i>.Flags</i>	USHORT	Mögliche Flags: Bit 0: <i>_CP_EXCLUSIVE</i> = 1 (exklusiv) Alle anderen Bits = 0.

Alle anderen Strukturelemente werden nicht ausgewertet.

Der Interrupt-Eingang wird durch einen Schreibzugriff (BYTE) auf den Kanal gesteuert.

<i>_MDC152_CON_IP_0_IRQ_B</i>	(= 1)	Verbinde IP-0 mit Interruptleitung IRQ-B
<i>_MDC152_CON_IP_1_IRQ_E</i>	(= 2)	Verbinde IP-1 mit Interruptleitung IRQ-E
<i>_MDC152_CON_IP_2_IRQ_C</i>	(= 4)	Verbinde IP-2 mit Interruptleitung IRQ-C
<i>_MDC152_CON_IP_3_IRQ_D</i>	(= 8)	Verbinde IP-3 mit Interruptleitung IRQ-D

Soll der Interrupt-Eingang deaktiviert werden, so muß eine 0 in den Kanal geschrieben werden.

Der aktuelle Status des Ausgangs kann durch einen Lesezugriff (BYTE) auf den Kanal ermittelt werden.

Hochsprachenbibliothek

Wie die Bibliothek eingebunden und verwendet wird, finden Sie in der Einführung im Abschnitt 'Hochsprachenbibliotheken'. Der Name der Bibliothek (*libname*) lautet **M034_LIB**, Sie finden sie im Verzeichnis (*pathname*) **SPB_MOD\BIB\M-DC15-2**. Vor allen anderen Routinen muß die Prozedur **m034_bib_startup** einmal aufgerufen werden.

m034_bib_startup **Initialisiere Modulbibliothek**

Pascal	PROCEDURE m034_bib_startup;
C	void EXPORT m034_bib_startup (void);
Funktion	Diese Prozedur initialisiert die Modulbibliothek. Es werden u. a. die Initialisierungsdaten aus den EEPROMs aller Module M-DC15-2 übernommen, die sich auf der Basiskarte befinden.

m034_set_conf_all **Konfiguriere alle Funktionseinheiten**

Pascal	PROCEDURE m034_set_conf_all (micro_slot, ledconf, intconf: byte);
C	void EXPORT m034_set_conf_all (byte micro_slot, byte ledconf, byte intconf);
Funktion	Diese Prozedur konfiguriert alle Funktionseinheiten des Moduls (Interrupt-Eingänge und LEDs). Der Parameter <i>ledconf</i> enthält die einzustellende Konfiguration für die LEDs und der Parameter <i>intconf</i> die einzustellende Konfiguration für die Interrupt-Eingänge. Die Codierung von <i>ledconf</i> entspricht der des EEPROM-Wortes 3 (Bit-0 bis Bit-7) und die Codierung von <i>intconf</i> der des EEPROM-Wortes 3 (Bit-8 bis Bit-15, siehe Seite 24-11).

Parameter *ledconf*:

Bit	nicht gesetzt (= 0)	gesetzt (= 1)
0	LED-0 ausgeschaltet	LED-0 eingeschaltet
1	LED-1 ausgeschaltet	LED-1 eingeschaltet
2	LED-2 ausgeschaltet	LED-2 eingeschaltet
3	LED-3 ausgeschaltet	LED-3 eingeschaltet
4	LED-4 ausgeschaltet	LED-4 eingeschaltet
5	LED-5 ausgeschaltet	LED-5 eingeschaltet
6	LED-6 ausgeschaltet	LED-6 eingeschaltet
7	LED-7 ausgeschaltet	LED-7 eingeschaltet

Parameter *intconf*:

Bit	nicht gesetzt (= 0)	gesetzt (= 1)
0	IP-0 nicht verbunden	IP-0 mit IRQ-B verbunden
1	IP-1 nicht verbunden	IP-1 mit IRQ-E verbunden
2	IP-2 nicht verbunden	IP-2 mit IRQ-C verbunden
3	IP-3 nicht verbunden	IP-3 mit IRQ-D verbunden

m034_set_conf_detail

Konfiguriere eine Funktionseinheit

Pascal	PROCEDURE m034_set_conf_detail (micro_slot, ipx, data_var: byte);
C	void EXPORT m034_set_conf_detail (byte micro_slot, byte ipx, byte data_var);
Funktion	Diese Prozedur konfiguriert den in Parameter <i>ipx</i> angegebenen Eingang (0 = IP-0 bis 3 = IP-3), wie in Parameter <i>data_var</i> angegeben (0 = verbinden, 1 = nicht verbinden).

m034_get_conf_all**Lies aktuelle Konfiguration**

Pascal	PROCEDURE m034_get_conf_all (micro_slot: byte; var ledconf, intconf: byte);
C	void EXPORT m034_get_conf_all (byte micro_slot, byte *ledconf, byte *intconf);
Funktion	Nach Aufruf dieser Prozedur enthalten die Variablen <i>ledconf</i> und <i>intconf</i> die aktuelle Konfiguration des Moduls. Die Codierung von <i>ledconf</i> entspricht der des EEPROM-Wortes 3 (Bit-0 bis Bit-7) und die Codierung von <i>intconf</i> der des EEPROM-Wortes 3 (Bit-8 bis Bit-15, siehe Seite 24-11).

m034_set_conf_eeprom**Setze EEPROM-Konfiguration**

Pascal	PROCEDURE m034_set_conf_eeprom (micro_slot: byte);
C	void EXPORT m034_set_conf_eeprom (byte micro_slot);
Funktion	Diese Prozedur konfiguriert und initialisiert das Modul M-DC15-2 entsprechend den Einstellungen im EEPROM des Moduls.

m034_get_ipx**Lies Zustand der Eingänge IPx**

Pascal	PROCEDURE m034_get_ipx (micro_slot: byte; var data_var: byte);
C	void EXPORT m034_get_ipx (byte micro_slot, byte *data_var);
Funktion	Nach Aufruf dieser Prozedur enthält die Variable <i>data_var</i> die Zustände der Eingänge IP-0 bis IP-3. Bit-0 entspricht dem Eingang IP-0, Bit-1 dem Eingang IP-1, Bit-2 dem Eingang IP-2 und Bit-3 dem Eingang IP-3. Ist das entsprechende Bit = 0, ist der Eingang stromlos.

m034_set_led**Definiere Zustand einer LED**

Pascal	PROCEDURE m034_set_led (micro_slot, led, data_var: byte);
C	void EXPORT m034_set_led (byte micro_slot, byte led, byte data_var);
Funktion	Diese Prozedur setzt die LED, die in Parameter <i>led</i> (0 = LED-0 bis 7 = LED-7) angegeben ist, entsprechend der Angabe in Parameter <i>data_var</i> (0 = aus, 1 = ein);

m034_get_led**Lies Zustand aller LEDs**

Pascal	PROCEDURE m034_get_led (micro_slot: byte; var ledconf: byte);
C	void EXPORT m034_get_led (byte micro_slot, byte *ledconf);
Funktion	Nach Aufruf dieser Prozedur enthält die Variable <i>ledcon</i> den Zustand der LEDs des Moduls. Bit-0 entspricht dem Zustand von LED-0, usw. bis Bit-7 entspricht dem Zustand von LED-7. Ist das entsprechende Bit = 0, ist die LED ausgeschaltet, ist das Bit = 1, ist die LED eingeschaltet.

Programmierung mit I/O-Zugriffen

Lokale I/O-Adressen

Adresse	Zugriff ¹	Funktion
MBA+10h	W8x	Trenne IP-0 von IRQ-B
MBA+11h	W8x	Verbinde IP-0 mit IRQ-B
MBA+12h	W8x	Trenne IP-1 von IRQ-E
MBA+13h	W8x	Verbinde IP-1 mit IRQ-E
MBA+14h	W8x	Trenne IP-2 von IRQ-C
MBA+15h	W8x	Verbinde IP-2 mit IRQ-C
MBA+16h	W8x	Trenne IP-3 von IRQ-D
MBA+17h	W8x	Verbinde IP-3 mit IRQ-D
MBA+18h	R8	Lies Status von IP-0 und IP-1 (0 = Eingang ist stromlos): Bit-0 = IP-0, Bit-1 = IP-1, Bit 2 bis 7 = ungültig
MBA+19h	R8	Lies Status von IP-2 und IP-3 (0 = Eingang ist stromlos): Bit-0 = IP-2, Bit-1 = IP-3, Bit 2 bis 7 = ungültig
MBA+1ah	R8	Lies Status, ob verbunden (1 = verbunden): Bit-0 = IP-0 mit IRQ-B, Bit-1 = IP-1 mit IRQ-E, Bit 2 bis 7 = ungültig
MBA+1bh	R8	Lies, ob verbunden (1 = verbunden): Bit-0 = IP-2 mit IRQ-C, Bit-1 = IP-3 mit IRQ-D, Bit 2 bis 7 = ungültig
MBA+00h	W8x	LED-0 (grün): ein
MBA+01h	W8x	aus
MBA+02h	W8x	LED-1 (rot): ein
MBA+03h	W8x	aus
MBA+04h	W8x	LED-2 (rot): ein
MBA+05h	W8x	aus
MBA+06h	W8x	LED-3 (rot): ein
MBA+07h	W8x	aus

¹ W8x bedeutet ein 8-Bit I/O-Schreibzugriff, bei dem die Daten keine Rolle spielen

R8 bedeutet ein 8-Bit I/O-Lesezugriff

Adresse	Zugriff^d	Funktion
MBA+08h	W8x	LED-4 (rot): ein
MBA+09h	W8x	aus
MBA+0ah	W8x	LED-5 (rot): ein
MBA+0bh	W8x	aus
MBA+0ch	W8x	LED-6 (rot): ein
MBA+0dh	W8x	aus
MBA+0eh	W8x	LED-7 (rot): ein
MBA+0fh	W8x	aus
