

<p style="text-align: center;">DL400</p> <p style="text-align: center;">VME Basismodul</p>
--

1. Funktion	2
1.1. Datenblatt	2
1.1.1. Anwendung	2
1.1.2. Daten	2
1.1.3. Besonderheiten	2
1.1.4. Aufbau	2
1.1.5. Stromversorgung	2
1.2. Blockdiagramm	2
1.3. Steckerbelegungen	3
1.3.1. P1-VME	3
1.3.2. P2-VME32/VXI	3
1.3.3. DL400C	5
2. Betrieb	6
2.1. Konfigurierung	6
2.1.1. SHORT-Adresse	6
2.2.1. Application-Board	6
2.2. Programmierung	6
2.2.1. Initialisierung	6
2.2.2. Speicherbelegung	7

1. FUNKTION

1.1. Datenblatt

1.1.1. Anwendung

VME-Interface für DL400-Application-Boards

1.1.2. Daten

Parameter	Wert	Dimension
Datenbreite	32	Bit
Address mode	SHORT (A16) STANDARD (A24) EXTENDED (A32)	
Data mode	standard (D16) longword (D32)	
Geschwindigkeit normal	120	ns/cycle
Geschwindigkeit langsam	1	µs/cycle

1.1.3. Besonderheiten

- VME, VMX Interface (16 / 32Bit)
- Interruptfähig: 8 Bit Vector, Priority 0-7 (0=Disable), ROAC/RORA Mode
- -5.2V Versorgung über Frontstecker
- Wrap-Feld für Einfachanwendungen
- alle Funktionen programmierbar

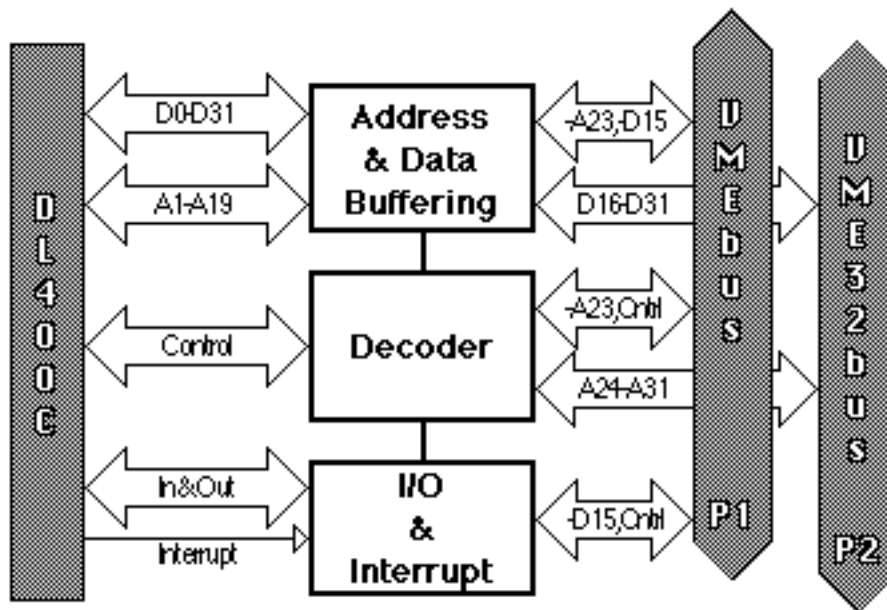
1.1.4. Aufbau

VME-Doppeleuropakarte, 4TE, 6HE

1.1.5. Stromversorgung

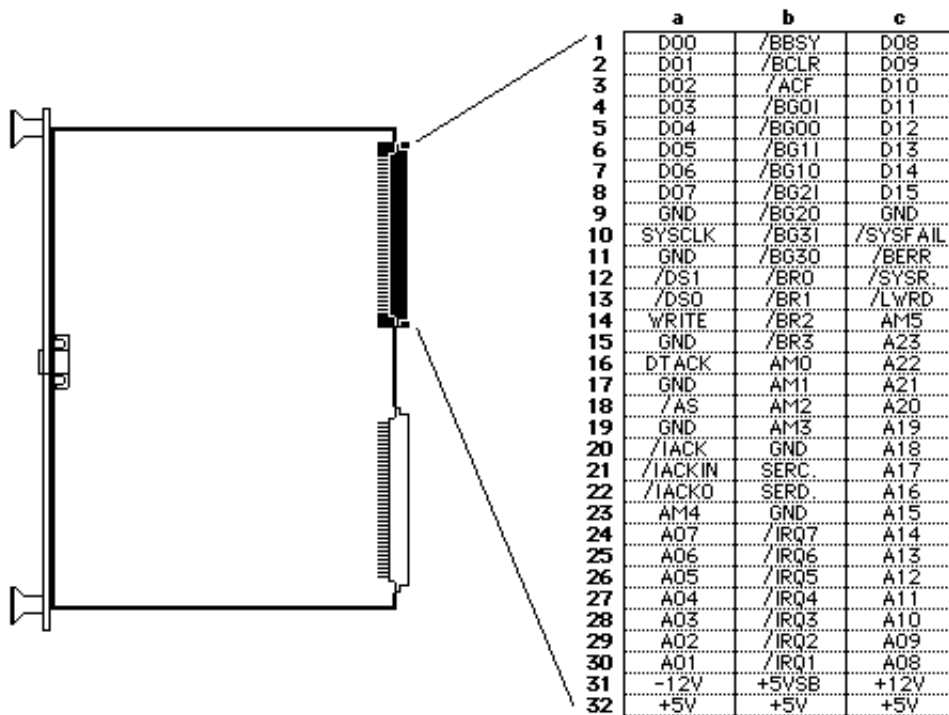
Spannung	Strom	Leistung
+5V	0.75A	3.75W
-5.2V		
+12V		
-12V		
Gesamt		3.75W

1.2. Blockdiagramm

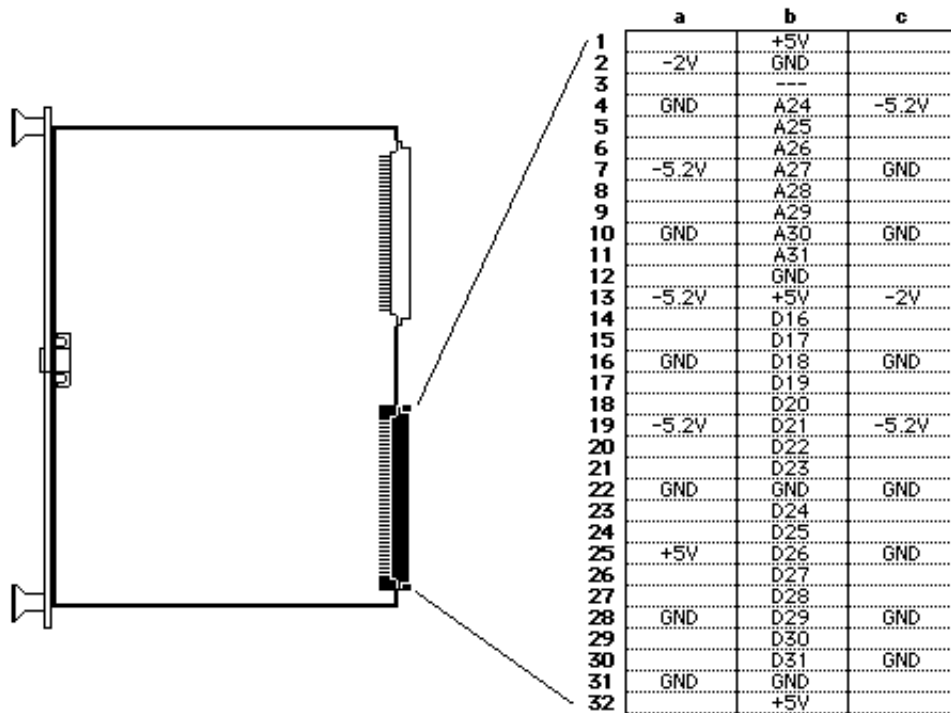


1.3. Steckerbelegungen

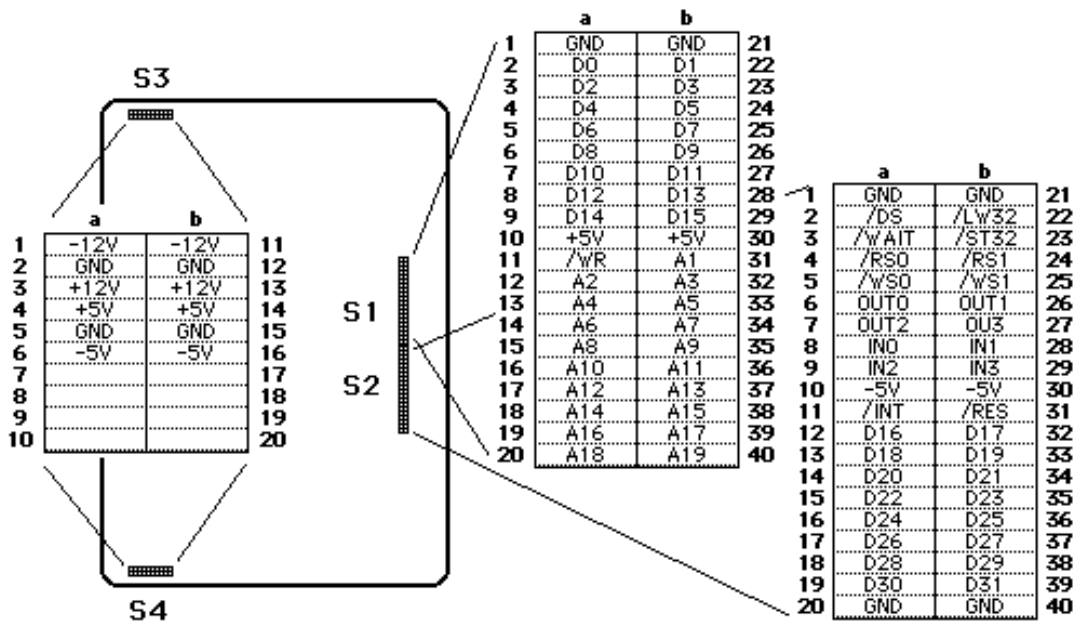
1.3.1. P1-VME



1.3.2. P2-VME32/VXI



1.3.3. DL400C



2. BETRIEB

2.1. Konfigurierung

2.1.1. SHORT-Adresse

Die zwei Rotary-HEX-Schalter auf der DL400 Basisplatine legen die Basisadresse für den Zugriff im SHORT-Adressraum fest (im folgenden \$ss.. bezeichnet!)

2.2.1. Application-Board

Für die eigentliche Anwendung wird ein spezielles Board auf die Übergabestecker S1-S4 aufgesteckt und mit der entsprechenden Frontplatte versehen.

2.2. Programmierung

W=Write, R=Read, SH=SHORT-Access, \$=HEX, ss=Address-Switches;

2.2.1. Initialisierung

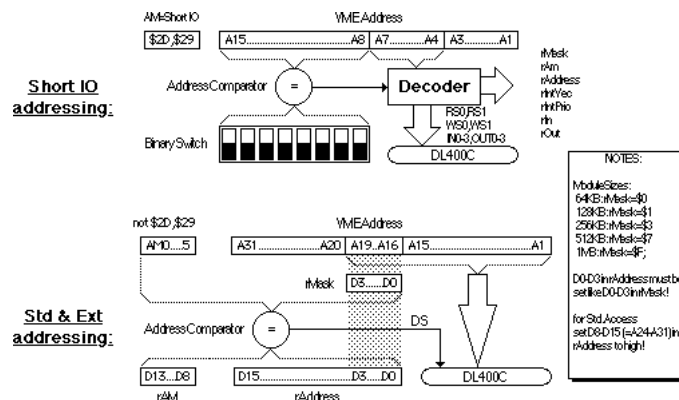
Das Modul ist zunächst nur im **Short**-Adressbereich (AM=\$2D oder \$29) ansprechbar.

Für die Festlegung der Standard-Basisadresse des Moduls, der Modulgröße, des gewünschten Addressmodifiers und der Parameter für einen Interrupt müssen verschiedene Register (im **Short**-bereich) auf der DL400 Basisplatine geladen werden!

	D 31 HighWord	D 15 LowWord	D 0
W. SH. \$ss50		D0...D15: Address	Std-Baseaddress
W. SH. \$ss60		D0...D3: Mask D8...D13: AM D15: Enable	Mask for Submodule Size Address Modifier enable STD-access
R/W. SH. \$ss70		D0...D7: IntVector D8...D10: IntPrio D11: IntMode	Interrupt Vector Interrupt Priority (0=Disable) 0=RORA, 1=ROAC

Festlegung der Modulgröße:

ModuleSize	Mask
64KB	\$0
128KB	\$1
256KB	\$3
512KB	\$7
1MB	\$F



ACHTUNG:

- D0...D3 in Address muß genauso wie D0...D3 in Mask gesetzt werden!
- D8...D15 in Address bei **Standard**-Zugriff auf HIGH setzen!

2.2.2. Speicherbelegung

	D31	HighWord	D15	LowWord	D0
R/W.SH.ss0a	D31..D16 (D15..D0 bei Wordacc.)		D15..D0		generiert RS0/WS0 mit Subadresse a (a=even!)
R/W.SH.ss8a	D31..D16 (D15..D0 bei Wordacc.)		D15..D0		generiert RS0/WS0 mit Subadresse a (a=even!) Zyklus=1µs;
R/W.SH.ss1a	D31..D16 (D15..D0 bei Wordacc.)		D15..D0		generiert RS1/WS1 mit Subadresse a (a=even!)
R/W.SH.ss9a	D31..D16 (D15..D0 bei Wordacc.)		D15..D0		generiert RS1/WS1 mit Subadresse a (a=even!) Zyklus=1µs;
R.SH.ss70			D12=IN0 D13=IN1 D14=IN2 D15=IN3		Get Port
W.SH.ss70			D12=OUT0 D13=OUT1 D14=OUT2 D15=OUT3		Set Port
W.SH.ss20					RESET