



# Inhaltsverzeichnis

Gewährleistung	1
Lieferumfang	2
Hardware-Voraussetzungen	2
- Allgemeines	3
Wozu dient diese Baugruppe ?	3
Wie funktioniert die Hardware ?	3
Wie programmiert man diese Baugruppe ?	4
Die verschiedenen Versionen	4
- 1. Der VMEbus im Allgemeinen	5
1.1 Geschichte des VMEbus	5
1.2 VMEbus für Europakarten	5
1.3 Leistungsübersicht des VMEbus	6
1.4 Wahl des Europakartenformats	6
1.5 VMEbus für 32 Bit Prozessoren	8
1.6 Der Weg zum Industriestandard	8
1.7 Subbusse als Teil der VMEbus-Architektur	9
- 2. Technische Spezifikationen	10
2.1 Übersicht	10
2.2 Daten-Transfer-Bus	11
2.3 Adreßverwaltung über Adress-Modifler	11
2.4 Arbitrations-Bus für Multiprozessor-Systeme	13
2.5 Interrupt-Verarbeitung	14
2.6 Hilfs- und Versorgungssignale	15
2.7 Steckerbelegung P1 und P2	16
2.8 Bedeutung der VMEbus-Signale	17
2.9 Erforderliche Treiberleistungen	18
2.10 Analogien und Unterschiede VME / 680X0	19

- 3.	Der Expansion Port des FALCON 030	20
3.1	Mechanische Details	20
3.2	Elektrische Details	20
3.3	Steckerbelegung	21
3.4	VME-Adreßbereich des FALCON 030 mit 4 MByte	22
3.5	VME-Adreßbereich des FALCON 030 mit 14 MByte	23
- 4.	Das "rho-VMEbus Interface"	24
4.1	Einbau des Interfaces	24
4.2	Interrupt Verarbeitung beim rho-VMEbus Interface	24
4.3	Interrupt Verarbeitung beim FALCON 030	25
4.4	Cache ON/OFF	26
4.5	Bus-Arbitrierung	26
4.6	Stromversorgung	27
4.7	Jumper des VMEbus Interfaces	27
4.8	Test des Interfaces mit RHOTEST.PRG	28
4.9	Das Programm ERKENN.PRG	29

# rhothron Gewährleistung

- 1. rhothron gewährleistet, daß die entsprechend den rhothron Richtlinien fachgerechte installierte Hard- und Software mit dem von rhothron angegebenen Spezifikationen übereinstimmt.
- 2. rhothron übernimmt keine Gewähr dafür, daß die Hard- und Software unterbrechungs- und fehlerfrei läuft, daß alle Fehler von rhothron beseitigt werden können und daß alle Funktionen in von allen vom Kunden gewählten Kombinationen ausführbar sind bzw. seinen Anforderungen entsprechen.
- 3. rhothron verpflichtet sich Fehler zu berichtigen, welche die vertrags-mäßige Benutzung nicht nur unerheblich beeinträchtigen, und zwar nach Wahl von rhothron und je nach Bedeutung des Fehlers durch die Lieferung einer verbesserten Version oder durch Hinweise zur Beseitigung oder Umgehen der Auswirkungen des Fehlers.
- 4. Der Kunde gewährt rhothron zur etwaigen Mängelbeseitigung die nach dem Ermessen von rhothron erforderliche Zeit und Gelegenheit. Verweigert der Kunde diese, ist rhothron von der Gewährleistung befreit. Jegliche Gewährleistung ist ausgeschlossen, soweit der Kunde oder ein Dritter Veränderungen irgendwelcher Art an der Hard- oder Software oder an dem Datenträger vornimmt oder diese unsachgemäß behandelt.
- 5. rhothron gewährleistet, daß der Datenträger frei von Material- und Fabrikationsfehlern ist. Etwaige fehlerhafte Datenträger wird rhothron durch fehlerfreie ersetzen.

Copyright (C) 1993 by rhothron GmbH

Herausgeber:

rhothron GmbH  
Entenmühlstr. 57  
D-66424 Homburg/Saar  
Tel. 06841 / 64067  
Fax. 06841 / 2467

Homburg, Jull 1993

## Lieferumfang

Zum Lieferumfang gehört neben dem Interface selbst eine 3 1/2 Zoll Diskette, sowie ein Bedienungshandbuch.

Auf der Originaldiskette befinden sich die Ordner TEST und VERSION.

Im Ordner "TEST" sind Demonstrations- und Testprogramme enthalten.

Der Ordner "VERSION" enthält das Datum der vorliegenden Software-Version und gegebenenfalls eine Read-Me-Datei, die ergänzende Informationen enthalten kann. Diese Read-Me-Datei ist nicht bei allen Software-Versionen notwendig und fehlt daher teilweise bewußt.

Sollte aus irgend einem Grund ein Teil des Lieferumfanges fehlen, wenden Sie sich bitte direkt an die rhothron GmbH in Homburg.

## Hardwarevoraussetzung

Um mit dieser Baugruppe arbeiten zu können, benötigen Sie als Mindestkonfiguration folgende Geräte:

- ATARI FALCON 030 mit unbenutztem Expansion Port

---

# Allgemeines

## Wozu dient diese Baugruppe ?

Das rhothron VMEbus Interface dient zur normgerechten Erweiterung des ATARI FALCON 030 Rechners um einen oder mehrere Steckplätze nach VMEbus Norm.

Dadurch kann der FALCON über seine umfangreichen Fähigkeiten hinaus erweitert und um externe Hardware ergänzt werden. Durch den Anschluß eines Steckplatzgehäuses mit zwei, sechs oder elf VMEbus Steckplätzen können nun nahezu beliebig viele Zusatzkarten gleichzeitig am FALCON betrieben werden.

## Wie funktioniert die Hardware ?

Dieses Interface wird auf den internen Expansion Port des FALCON 030 aufgesteckt. Die voll gepufferten Signale werden über zwei Flachbandkabel aus dem Gehäuse herausgeführt und werden extern über eine kleine Adapterplatine auf eine nach VMEbus Norm belegte 96-polige VG-Buchsenleiste geführt.

Durch das Interface werden alle Adreß-, Daten- und Steuerleitungen des FALCON Rechners gepuffert, um eine Beeinflussung des Rechners auszuschließen.

Zusätzlich zur Puffer- und Übertragungsfunktion kann auf dem rho-VME-Interface eine Bus-Arbitrierungs Logik implementiert werden. Diese macht den rho-VMEbus multimasterfähig und erlaubt so den gemeinsamen Zugriff mehrerer Rechner auf einen gemeinsamen VMEbus bzw. die dort angeschlossene Peripherie. Diese Busarbitrierung arbeitet nach dem Prinzip eines "Single-Level-Arbiters" auf Bus-Request Ebene 3. Das Interface selber arbeitet dabei nach dem "release, when done"-Prinzip.

Die Stromversorgung des Interfaces erfolgt direkt aus den +5 Volt des Rechners.

## Wie programmiert man diese Baugruppe ?

Das rho-VMEbus-Interface bedarf keinerlei Programmierung. Es ist lediglich ein Pufferinterface, daß keinerlei Einfluß auf die softwaremäßige Ansprache des VMEbus hat. Die angeschlossenen VMEbus Baugruppen sind Memory-mapped-IO und können wie jede andere Speicherzelle im Rechner angesprochen werden.

# 1. Der VMEbus im Allgemeinen

## 1.1 Geschichte des VMEbus

Der VMEbus ist kein völlig neues Bussystem, sondern benutzt als Basiskonzept den 16/32-Bit-VERSAbus, der von Motorola bereits im Jahre 1979 für die "VERSAModul"-Familie, die insbesondere für den Leistungsbereich bis zum Microcomputer konzipiert ist, entwickelt wurde. Beide Bussysteme unterstützen speziell die Busstruktur der 16/32-Bit-Mikroprozessorfamilie MC68000 und der Nachfolgetypen. Dennoch können VERSA- und VMEbus auch als Bussysteme für andere Prozessoren vorteilhaft eingesetzt werden. Da sich die MC68000-Familie zu einem Industriestandard bei 16- und 32-Bit-Mikroprozessoren entwickelt hat, ist der VMEbus damit das optimale Bussystem für die meisten Mikroprozessor-Anwendungen.

VMEbus-Karten mit anderen Prozessoren, z.B. dem 32016 von National Semiconductor, dem 80286 von Intel und dem J11 von Digital Equipment sind ebenfalls am Markt erhältlich.

## 1.2 VMEbus für Europakarten

Für Anwendungen vor allem im industriellen Bereich werden oft kompakte Systeme bei relativ geringer Leistungsfähigkeit, aber höherer Modularität, gefordert. Für diesen rasch wachsenden Marktbereich ist ein kleines Kartenformat erforderlich als es die VERSAModule bzw. der VERSAbus besitzen. Aus diesem Grunde entschied man sich bei Motorola schon frühzeitig für das Europakartenformat mit dem Typ MC680X0 als Zentralprozessor. Der erfolgreiche VERSAbus wurde in seiner Gesamtstruktur übernommen. Allerdings konnte nur eine Untermenge der 140 Signalleitungen des VERSAbus auf den bekannten Europakartensteckverbindern implementiert werden. Es stellt sich heraus, dass diese Untermenge (96 Signalleitungen) für die geforderte Leistungsfähigkeit und die vorgesehenen Einsatzbereiche keine wesentliche Einschränkung darstellte. So entstand der VMEbus.

## 1.3 Leistungsübersicht des VMEbusses

Der VMEbus zeichnet sich durch folgende Leistungsmerkmale aus, auf die später näher eingegangen wird:

- Unterstützung von Mikroprozessor-Architekturen bis zu 32 Bit Wortbreite.
- Unterstützung von Multiprozessor-Systemen.
- Datendurchsatz im praktischen Betrieb 34 MByte/s (theoretisch bis zu 57 MByte/s).
- Vollständig asynchrones, multiplexfreies Busprotokoll.
- Prioritätsgesteuerte Busbelegung über vier Prioritätsebenen und zusätzlichem Daisy-Chain auf jeder Ebene.
- Unterstützung von zentraler oder verteilter Interrupt-Verarbeitung in sieben Prioritäts-Ebenen und Daisy-Chain.
- Zusätzliche Adreßinformationen über Adreßmodifizier-Leitungen zur flexiblen Systemkonfigurierung und Speicherverwaltung.
- Unterstützung von Datenblock-Transfer.
- Zusätzliche Meldeleitungen zur Erhöhung der Systemsicherheit (Bus-, System-, Spannungsfehler).

Als 24-Bit-Adreß- und 16-Bit-Datenmodus ist der VMEbus auf Einfach-Europakarten implementierbar.

Für den vollen 32-Bit-Adreß- und Datenbereich wird eine Doppelpackkarte benötigt.

## 1.4 Wahl des Europakartenformats

Die Europakartennorm für verschiedene Kartenformate und Steckverbindungen hat sich nicht nur in Europa auf breiter Basis durchgesetzt, sondern findet zunehmend auch auf dem amerikanischen Markt wachsendes Interesse. Die Gründe dafür sind zahlreich:

- Die Industrienormen (DIN und IEC) für Kartenformat, Steckverbindung und Einschübe existieren bereits.
- Dadurch ist ein großes Angebot an mechanischem Zubehör (Gehäuse, Stecker, usw.), Entwicklungshilfen (Prototyp-, Extenderkarten) und Netzteilen

von verschiedenen Herstellern weltweit lieferbar.

- Die indirekte Steckverbindung ist wesentlich zuverlässiger als die direkte.
- Das kompakte Europakartenformat bildet den besten Kompromiß zwischen Leistungsfähigkeit und Modularität moderner, auf hochintegrierten VLSI-Bausteinen basierender Computersysteme.

Obwohl die Norm DIN 41494 bzw. IEC48D verschiedene Kartengrößen definiert, haben nur zwei Formate einen breiten Marktanteil erringen können. Das sogenannte Einfach-Europakartenformat (100 x 160 mm<sup>2</sup>) und das Doppel-Europakartenformat (233 x 160 mm<sup>2</sup>). Das kleine Format wird für Karten mit mittlerer Leistungsfähigkeit und in Systemen, bei denen es auf hohe Modularität ankommt, eingesetzt. Höhere Leistungsfähigkeit dagegen erfordert die Doppeleuropakarte.

Beim VMEbus werden alle Signale zum Aufbau eines kompletten 16-Bit-Systems über nur einen DIN-Stecker (P1, 96 polig) geführt. Damit ist es möglich, Einfach- und Doppeleuropakarten im gleichen System in beliebiger Zusammenstellung zu verwenden. Dadurch erzielt man mit dem VMEbus ein Höchstmaß an Systemmodularität und Flexibilität für den Systementwurf.

## 1.5 VMEbus für 32-Bit-Prozessoren

Besonders vorteilhaft ist das VMEbus-Konzept durch die Möglichkeit der Systemerweiterung auf einen 32-Bit Bus. Dazu werden entsprechend der Spezifikation nur die fehlenden 16 Daten- und acht Adreßleitungen auf den zweiten Stecker (P2, 96polig) einer Doppeleropakarte geführt. Da keine zusätzlichen Steuersignale benötigt werden (auch solche für 32-Bit-Operationen werden über den Stecker P1 geführt), ist eine Erweiterung von 16-Bit auf 32-Bit-Systeme problemlos möglich. Die Busstruktur bleibt bei Verwendung vorhandener 8- oder 16 Bit-Karten gleich. So wird ein stufenweiser Übergang auf ein 32-Bit-Doppeleropakarten-System ermöglicht.

## 1.6 Der Weg zum Industriestandard

Ingenieurmäßig ist die Definition und Vorstellung eines neuen Buskonzeptes kein allzu großes Problem. Ein theoretisch noch so leistungsfähiges Bussystem ist für den Anwender allerdings kaum etwas wert, wenn es nicht zusätzlich weitere Bedingungen erfüllt:

- Eine breite Palette buskompatibler Produkte sollte zur Verfügung stehen.
- Möglichst viele unabhängige Hersteller sollten buskompatible Produkte anbieten.
- Gerade im Bereich industrieller Automation wird eine Vielfalt spezieller Ein/Ausgabe-Karten benötigt.
- Wünschenswert ist eine weltweit gültige Bus-Norm.

Um die oben genannten Forderungen zu erfüllen, beschritt Motorola einen bisher für die Mikrocomputerindustrie beispiellosen Weg. Es wurden mit großen Elektronikherstellern in Europa und in den USA Verhandlungen über ein gemeinsam zu unterstützendes Bussystem aufgenommen. Dabei kam man in relativ kurzer Zeit zu einem bemerkenswerten Ergebnis: Die vier Elektronik-Firmen Motorola, Mostek, Phillips/Signetics/Vaivo und Thomson-Efcs gaben anlässlich der Systems 1981 bekannt, daß sie Produkte für den VMEbusentwickeln werden. Gleichzeitig wurde eine VMEbus-Spezifikation veröffentlicht, die auch die Ideen und Erfahrungen der Partner von Motorola umfaßte. Damit war der Weg zum VMEbus als Defacto-Industriestandard geebnet. Ein Standardisierungskomitee mit Mitgliedern aller vier Firmen wurde gegründet, zu dessen Aufgaben die Koordination aller Aktivitäten und das Vorantreiben einer offiziellen VMEbus-Norm gehört (IEEE P1014, IEC-Standard). Heute ist die VITA (VMEbus International Trade Association) als herstellerunabhängige Organisation für alle Normierungsaktivitäten zuständig.

## 1.7 Subbusse

### als Teil der VMEbus-Architektur

Neben dem eigentlichen VMEbus sind die zwei "Sub-Bussysteme" VMS-Bus und der VME-Subsystem-Bus (VSB) mittlerweile als fertige Design-Spezifikationen erhältlich. Alle drei Bussysteme (VMEbus, serieller VMS-Bus und lokale Buserweiterung VSB) bilden die Gesamt-VMEbus-Architektur.

Beim VMS-Bus handelt es sich um einen seriellen Bus, für den auf dem Stecker P1 die beiden Anschlüsse SERDAT und SERCLK reserviert werden. Über den VMS-Bus können kurze Nachrichten (Messages, z.B. Interruptvektoren und Semaphoren) parallel zum VMEbus mit einer Geschwindigkeit von bis 3.2 MBit/s übertragen werden. Er kann auch zum Aufbau fehlertoleranter Systeme eingesetzt werden, da bei Ausfall des VMEbus das System über dem VMS-Bus weiterarbeiten kann. Außerdem kann man über den VMS-Bus räumlich getrennte VMEbus-Systeme verbinden. Auf Grund des komplexen seriellen Übertragungsprotokolls ("selbstzutellender Bus") ist eine Implementierung des VMS-Bus in zwei VLSI-Bausteinen erhältlich.

Beim VME-Subsystem-Bus (VSB) handelt es sich um eine lokale 32-Bit-Buserweiterung, d.h. man kann über diesen Bus zusätzliche Speicherkarten oder Ein/Ausgabe-Baugruppen an eine CPU-Platine anschließen. Mit dem VMS werden also nicht alle Boards eines VMEbus-Systems miteinander verbunden (wie beim VMEbus und VMS-Bus), sondern er stellt eine Erweiterung des lokalen Prozessorbusses dar. Aus diesem Grunde gab es mehrere Vorläufer (VMS, MVMX32) bevor der VSB zum weltweiten IBC-Standard wurde. Es ist ohne Schwierigkeiten möglich, in ein- und demselben VMEbus-System verschiedene lokale Buserweiterungen einzusetzen. Der VSB basiert auf dem von Motorola entwickelten MVMX32bus und ist zu diesem aufwärts-kompatibel, so daß heute verfügbare MVMX32 kompatible Platinen auch in zukünftigen VSB-Systemen eingesetzt werden können.

## 2. Technische Spezifikationen

### 2.1 Übersicht

Der VMEbus gliedert sich in vier unabhängige Teilsysteme:

- Der Daten-Transfer-Bus (DTB) enthält alle Daten- und Adreßleitungen, sowie die zum Datentransfer notwendigen Steuerleitungen
- Der Arbitrations-Bus liefert alle Signale, die zur Steuerung in einem Multi-Master-System notwendig sind
- Der Interrupt-Bus dient zur Behandlung von Unterbrechungsanforderungen
- Unter dem Begriff "Versorgungs- und Hilfsleitungen" werden alle restlichen Signale, die Stromversorgung und eine Fehlererkennung zusammengefaßt

## 2.2 Daten-Transfer-Bus

Der VME-Daten-Transfer-Bus erlaubt eine multiplexfreie, asynchrone Datenübertragung mit einer maximalen Geschwindigkeit von 8.5 Mio 32-Bit Transfers/s (= 34 MByte/s). Die Datenübertragung zwischen Sender (Master) und Empfänger (Slave) läuft folgendermaßen ab.

Das Anlegen einer gültigen Adresse auf dem Adreßbus wird dem Slave durch Aktivierung des Signals "Adress Strobe" (AS) mitgeteilt. Das Signal "WRITE" (R/W) zeigt an, ob es sich um eine Schreib- oder Leseoperation handelt. Bei einem Lesezyklus wird dem Slave über die beiden Data-Strobe-Leitungen "DS0" und "DS1" mitgeteilt, daß der Master zum Einlesen der Daten vom Slave bereit ist. Daraufhin legt der adressierte Slave die Daten auf den Datenbus und signalisiert durch Aktivieren von "DTACK" (Data Acknowledge), daß gültige Daten anliegen. Analog werden beim Schreibzyklus zunächst vom Master die Daten auf den Datenbus gelegt und danach DS0 und DS1 aktiviert. Wenn der Slave die Daten aufgenommen hat, signalisiert er dies durch Aktivierung von "DTACK". In beiden Fällen gibt der Master den Bus für den nächsten Zyklus erst dann frei, wenn er das "Quittierungssignal" "DTACK" vom Slave erhalten hat. Der Slave seinerseits gibt den Datenbus erst dann frei (Lesezyklus), wenn der Master die Datenstrobes "DS0" und "DS1" deaktiviert hat.

Über dieses sogenannte "Handshaking" (die Charakteristik des asynchronen Busses) ist die problemlose Zusammenarbeit von Einheiten unterschiedlicher Geschwindigkeit möglich.

Warum sind die beiden Datenstrobes "DS0" und "DS1" vorgesehen? Der VMEbus erlaubt Byte-, Wort- (16 Bit) und Langwort-Datentransfers (32 Bit). Durch Kombination von "DS0", "DS1" und unter Berücksichtigung des Signals "LWORD" wird der entsprechende Transfer angezeigt.

Falls der Slave aufgrund einer fehlerhaften Signalkombination auf dem DTB nicht in der Lage ist, die Schreib/Lese-Anforderung ordnungsgemäß zu bearbeiten, so sendet er statt "DTACK" das Signal "Bus Error" (BERR) aus. Der Master geht dann in die zugehörige Fehlerbehandlungsroutine. Ein typischer Fall wäre ein Parity-Fehler auf einem Speicherboard, der über das BERR-Handshaking sofort erkannt und entsprechend behandelt werden kann.

Falls die angesprochene Peripherie weder mit "DTACK" noch mit "BERR" antwortet, würde der Prozessor endlos auf eine Antwort warten. Um dieses "Aufhängen" zu vermeiden, sollte in jedem System ein sogenannter Watch-Dog-Timer vorgesehen sein, der durch die fallende Flanke von AS gestartet wird, nach Ablauf einer vordefinierten Zeit einen Bus-Error auslöst und damit den Zyklus beendet.

## 2.3 Adreßverwaltung über Adress-Modifizier

Der VMEbus verfügt über ein äußerst leistungsfähiges und flexibles Konzept zur Adressverwaltung. Dazu dienen die insgesamt sechs "Adress-Modifizier"-Leitungen AM0 .. AM5, die dem Zeitverhalten der Adreßleitungen folgen und zusätzliche Informationen über die anliegende Adresse liefern. Über die insgesamt 64 verschiedenen Adreßinformationen werden beim VMEbusz.B. folgende Anwendungsfälle abgedeckt:

- Um nicht immer auf allen Modulen den vollen 32-Bit-Adreßbereich decodieren zu müssen, sind über entsprechende AM-Codes drei Adreßbereiche festgelegt:
  - Extended = voller 32-Bit-Bereich
  - Standard = 24-Bit-Bereich
  - Short = 16-Bit-Bereich
- Privilegierter Speicherzugriff: Ein hohes Maß an Systemsicherheit kann durch Aufteilung der Speicherzugriffe in "Supervisor" (Überwachungs-) und "Non-Privileged"- (Anwender-) Modus erzielt werden. Der Supervisor-Modus erlaubt bestimmte geschützte Operationen.
- Aufteilung des Speichers in Daten- und Programmbereich, ein übliches Verwaltungsverfahren, besonders bei der Verwaltung in höheren Sprachen.
- Unterscheidung zwischen Zugriff auf den Speicher oder eine Ein-/Ausgabe-Einheit (Memory-Mapped-E/A)
- Festlegung eines ganzen Datenübertragungszyklus. Die VMEbus-Spezifikation sieht beispielsweise unter AM-Code \$0B die rasche Übertragung der Information aufeinanderfolgender Speicherzellen vor, und zwar durch Erhöhen der Adressen auf dem Slave selbst.

### Wichtige Adress-Modifizier Codes :

AM	5	4	3	2	1	0	Bedeutung
	1	1	1	1	1	0	Standard Supervisor Programm Zugr.
	1	1	1	1	0	1	Standard Supervisor Daten Zugriff
	1	1	1	0	1	0	Standard User Programm Zugriff
	1	1	1	0	0	1	Standard User Daten Zugriff
	1	0	1	1	0	1	Short Supervisor E/A-Zugriff
	1	0	1	0	0	1	Short User E/A-Zugriff
	0	0	1	1	1	0	Extended Supervisor Programm Zugr.
	0	0	1	1	0	1	Extended User Programm Zugriff
	0	0	1	0	0	1	Extended User Daten Zugriff

Neben diesen aufgezeigten Anwendungen können die Adreß-Modifizierer den Aufbau von Multitasking-Systemen vereinfachen und die Sicherheit drastisch erhöhen. In diesem Fall werden jeder Task bestimmte AM-Codes zugeordnet, über die der Zugriff auf alle einer Task zugeordneten Speicherbereiche geregelt wird.

## 2.4 Arbitrations-Bus für Multiprozessor-Systeme

In einem Multiprozessor-System können zwei oder mehrere Master auf den DTB zugreifen. Die Koordination wird vom sogenannten Arbitrer vorgenommen. Jeder Master kann über eine der vier Bus-Request-Leitungen BR0 .. BR3 den Bus anfordern. Jede Leitung entspricht einer bestimmten Priorität. Der Arbitrer vergleicht die Busanforderung mit der Priorität des den Bus Innehabenden Masters; ist letztere höher ( $BG3 > BG2 > BG1 > BG0$ ), so kann die Anforderung erst bedient werden, wenn der Bus freigegeben wird. Andernfalls aktiviert der Arbitrer das Signal "Bus Clear" (BCLR), um den gerade aktiven Master zur Busfreigabe aufzufordern. Wenn dieser an einem passenden Punkt zur Unterbrechung seiner Aktivität angekommen ist, signalisiert er dies durch Deaktivierung von "Bus-Busy" BBSY. Daraufhin schickt der Arbitrer der höchsten anstehenden Priorität eine Busfreigabe (Bus-Grant BG0 .. BG3) und deaktiviert BCLR.

Da auf diese Weise nur vier Bus-Master vom Arbitrer koordiniert werden könnten, wurde beim VMEbus eine weitere Prioritätsebene in den Bus selber eingebaut. Dies geschieht mit "Daisy-Chain"-Technik, bei der die Priorität von der physikalischen Anforderung des Masters auf dem Bus bestimmt wird. Der dem Arbitrer am nächsten befindliche Master hat die höchste Priorität. Die Priorität nimmt also mit wachsender Entfernung vom Arbitrer ab. Jeder der vier Anforderungsebenen BR0 .. BR3 hat eine eigene Daisy-Chain. Das entsprechende Bus-Grant-Signal BG gelangt also vom Arbitrer an den Anschluß BGXIN des ersten Masters. Will dieser den Bus belegen, so gibt er das BG-Signal nicht weiter und übernimmt selbst die Kontrolle über den Bus. Andernfalls aktiviert er BGXOUT und gibt so die Freigabe über BGXIN an den nächsten Masters weiter, usw.. Auf Grund dieser Daisy-Chain-Struktur können beliebig viele Master auf dem VMEbus zusammenarbeiten. Es handelt sich um einen echten Multi-Master-Bus.

## 2.5 Interrupt-Verarbeitung

Der VMEbus erlaubt eine prioritätsgesteuerte Interruptverarbeitung über Interruptvektoren. Eine typische Verarbeitung bei nur einem Interrupthandler im System läuft in drei Phasen ab:

- a) **Interruptanforderung:** Diese kann über sieben Interrupt-Request-Leitungen IRQ0 .. IRQ7 erfolgen. IRQ7 hat die höchste Priorität, IRQ0 die niedrigste. Der Handler entscheidet aufgrund einer Bedingung im Prozessor (z.B. Interruptmaske MC68000), ob der Interrupt zugelassen wird oder nicht. An jede der sieben Interrupt-Request-Leitungen können mehrere Module angeschlossen sein. Zur Unterscheidung dient ein Interrupt-Erkennungssignal, das über eine Daisy-Chain-Verbindung von Modul zu Modul geführt wird. Hier wird also wie beim Arbitrations-Bus eine zweite Prioritätsebene aufgrund der physikalischen Anordnung der Interruptanforderer auf dem VMEbus nicht begrenzt.
- b) **Interrupterkennung:** Der Interrupthandler benutzt den DTB, um den Interruptanforderer zu identifizieren und von ihm die Vektornummer zu erhalten. Dazu muß der Interrupthandler zunächst den DTB anfordern (falls er nicht gerade DTB-Master ist). Dies geschieht über die Bus-Arbitrierungslogik (Im vorigen Kapitel beschrieben).
- c) **Interrupt-Service-Routine:** Über die Vektornummer wird die Adresse der Interrupt-Service-Routine gefunden, wo der Prozessor sich die nächsten abzuarbeitenden Befehle holt.

Der zeitliche Ablauf der Interrupterkennung ist wie folgt. Der Interrupthandler aktiviert das Signal "Interrupt-Acknowledge" IACK und legt den Interruptanforderungscode (= Priorität 1 .. 7) in binärer Form auf die niedrigsten Adreßleitungen A01 .. A03 des Adreßbus. Der nun folgende zeitliche Ablauf ist der gleiche wie bei einem üblichen Lesezyklus auf dem Daten-Transfer-Bus.

Die Aktivierung des Adreßstrokes zeigt das Anliegen einer gültigen Adresse (= Interruptcode) an, das Inaktive "WRITE" zeigt einen Lesezyklus und DS0-Signal im aktiven Zustand zeigt dem Interruptanfrager, daß das niederwertigste Byte auf dem Datenbus gelesen werden soll. Der Interruptanforderer wird über die anliegende Adresse angesprochen. Er kann erst dann die Vektornummer auf den Datenbus legen, wenn er auch das IACKIN-Signal empfangen hat. Diese Signal gelangt über eine Daisy-Chain-Verbindung, an deren Eingang "IACK" liegt, über die Ausgänge

"JACKOUT" eventuell davorliegender Module an den Interruptanforderer. Selbstverständlich dürfen diese Module mit höherer "Daisy-Chain"-Priorität keinen Interrupt derselben Priorität angefordert haben, da sonst kein JACKOUT-Signal erscheint. Nachdem der Interrupt-Auslöser seine geforderte Vektornummer auf den Datenbus gelegt hat, aktiviert er "DTACK". Der Interrupthandler liest die Vektornummer und startet die Interrupt-Service-Routine.

## 2.6 Hilfs- und Versorgungssignale

Der VMEbus verfügt über spezielle unabhängige Signale für Synchronisation, Initialisierung, Systemtest und Fehlerdiagnose. Diese Hilfssignale stellen einen wesentlichen Beitrag zur Gesamtleistungsfähigkeit eines modernen Bussystems dar. Der VMEbus unterstützt folgende Funktionen:

- Systemtakt (SYSCLK) ist ein vom Prozessortakt unabhängiges 16-MHz-Taktsignal, das in keiner festen Phasenbeziehung zu anderen VMEbus-Signalen steht. "SYSCLK" kann beispielsweise für Zähler und Synchronisationsfunktionen herangezogen werden.
- Über System-Reset (SYSRESET) können alle an den VMEbus angeschlossenen Module in einen definierten (Anfangs-) Zustand gebracht werden. "SYSRESET" kann manuell (Taster an der Bedienfront) oder von einem "Power-Monitor-Modul" automatisch beim Einschalten der Stromversorgung aktiviert werden.
- Über die System-Test-Leitung "SYSFAIL" kann ein Fehler im System gemeldet werden. Ein typischer Anwendungsfall sind Boards, die nach der Aktivierung von "SYSRESET" eine Selbsttest auf der Platine durchführen. Zeigt der Test einen Fehler, so wird "SYSFAIL" nicht deaktiviert und der System-Controller muß entsprechende Maßnahmen ergreifen.
- Über "ACFAIL" wird dem System ein Spannungseinbruch gemeldet. Die VMEbus Spezifikationen schreiben eine Auslegung des Netzteils vor, die für den ordnungsgemäßen Abbruch der laufenden Aktivitäten minimal 4 ms zuläßt.

Die Versorgung geschieht beim VMEbus über Leitungen mit +5 V, +12 V, -12 V und +5 V Notstrom bzw. Akkumulator-Spannung.

## 2.7 Steckerbelegung

Pin-Nr.	Steckverbinder P1			Steckverbinder P2		
	Reihe A	Reihe B	Reihe C	Reihe A	Reihe B	Reihe C
1	D00	/BBSY	D08	n.c.	+5 V <sub>DC</sub>	n.c.
2	D01	/BCLR	D09	n.c.	GND	n.c.
3	D02	/ACFAIL	D10	n.c.	RESERVED	n.c.
4	D03	/BG0IN	D11	n.c.	A24	n.c.
5	D04	/BG0OUT	D12	n.c.	A25	n.c.
6	D05	/BG1IN	D13	n.c.	A26	n.c.
7	D06	/BG1OUT	D14	n.c.	A27	n.c.
8	D07	/BG2IN	D15	n.c.	A28	n.c.
9	GND	/BG2OUT	GND	n.c.	A29	n.c.
10	SYSCLK	/BG3IN	/SYSFAIL	n.c.	A30	n.c.
11	GND	/BG3OUT	/BERR	n.c.	A31	n.c.
12	/DS1	/BR0	/SYSRESET	n.c.	GND	n.c.
13	/DS0	/BR1	/LWORD	n.c.	+5 V <sub>DC</sub>	n.c.
14	/WRITE	/BR2	AM5	n.c.	D16	n.c.
15	GND	/BR3	A23	n.c.	D17	n.c.
16	/DTACK	AM0	A22	n.c.	D18	n.c.
17	GND	AM1	A21	n.c.	D19	n.c.
18	/AS	AM2	A20	n.c.	D20	n.c.
19	GND	AM3	A19	n.c.	D21	n.c.
20	/IACK	GND	A18	n.c.	D22	n.c.
21	/IACKIN	SERCLK	A17	n.c.	D23	n.c.
22	/IACKOUT	SERDAT	A16	n.c.	GND	n.c.
23	AM4	GND	A15	n.c.	D24	n.c.
24	A07	/IRQ7	A14	n.c.	D25	n.c.
25	A06	/IRQ6	A13	n.c.	D26	n.c.
26	A05	/IRQ5	A12	n.c.	D27	n.c.
27	A04	/IRQ4	A11	n.c.	D28	n.c.
28	A03	/IRQ3	A10	n.c.	D29	n.c.
29	A02	/IRQ2	A09	n.c.	D30	n.c.
30	A01	/IRQ1	A08	n.c.	D31	n.c.
31	-12 V <sub>DC</sub>	+5 V <sub>STBY</sub>	+12 V <sub>DC</sub>	n.c.	GND	n.c.
32	+5 V	+5 V	+5 V	n.c.	+5 V	n.c.

n.c.

= not connected (zur freien Verfügung durch den Anwender)

/

= Signal ist active low

## 2.8 Bedeutung der VMEbus-Signale

<u>Signalleitung</u>	<u>Bedeutung</u>	
D0 - D15	Datenleitungen P1	Einfach-Europakarten
D16 - D31	Datenleitungen P2	Doppel-Europakarten
A1 - A23	Adreßleitungen P1	Einfach-Europakarten
A24 - A31	Adreßleitungen P2	Doppel-Europakarten
AM0 - AM5	Adreßmodifizier	Zusätzliche Adreßinformation
/AS	Adreßstrobe	Adressen Valide
/DS0	Datenstrobe untere Hälfte	Daten D0-D7 valide
/DS1	Datenstrobe obere Hälfte	Daten D8-D15 valide
/LWORD	Langwortzugriff	Zugriff auf D16-D31
/WRITE	Richtung des Zugriffs	Lesen = H, Schreiben = L
/SYSRESET	Reset	Reset z.B. bei Power On
/BERR	bus error	Busfehler Rückmeldung
/SYSFAIL	system failure	Systemfehler Rückmeldung
/ACFAIL	Spannungsfehler	warnet 4 ms vor Spannungsabfall
/IRQ1-/IRQ7	Interrupt-Anforderungssignale	
/IACK	Interrupt-Acknowledge	Interrupt-Bestätigung
/IACKIN	Interrupt-Acknowledge Ein	wie IACK, aber in Daisy-Chain
/IACKOUT	Interrupt-Acknowledge Aus	Daisy-Chain Ausgang
/BR0-/BR3	Bus-Request Level 0-3	Busanforderung durch Master
/BG0IN - /BG3IN	Bus-Grant IN Level 0-3	Buszutellung Daisy Chain Ein
/BG0OUT-/BG3OUT	Bus-Grant OUT Level 0-3	Buszutellung Daisy Chain Aus
/BCLR	Aufforderung zur Busfreigabe	durch Arbitr
/BBSY	Bus-Busy	Bus belegt
SYSCLK	16 MHz Systemtakt	asynchron
SERCLK serial clock	Taktleitung des VMS Busses	
SERDAT serial data	Datenleitung des VMS Busses	

## 2.9 Erforderliche Treiberleistungen

Signalleitung	Treiberleistung	Ausgangsart
D0 - D15	48 mA	Tri-State
D16 - D31	48 m	Tri-State
A1 - A23	48 mA	Tri-State
A24 - A31	48 mA	Tri-State
AM0 - AM5	48 mA	Tri-State
/AS	64 mA	Tri-State
/DS0	64 mA	Tri-State
/DS1	64 mA	Tri-State
/LWORD	48 mA	Tri-State
/WRITE	48 mA	Tri-State
/SYSRESET	48 mA	Open Collector
/BERR	48 mA	Open Collector
/SYSFAIL	48 mA	Open Collector
/ACFAIL	48 mA	Open Collector
/IRQ1-/IRQ7	48 mA	Open Collector
/IACK	48 mA	Totem-Pole
/IACKIN	48 mA	Totem-Pole
/IACKOUT	48 mA	Totem-Pole
BR0 - BR3	48 mA	Open Collector
BGOIN - BG3IN	48 mA	Totem-Pole
BGOOUT-BG3OUT	48 mA	Totem-Pole
BCLR	64 mA	Totem-Pole
BBSY	48 mA	Totem-Pole
SYSCLK	64 mA	Totem-Pole
SERCLK	64 mA	Totem-Pole
SERDAT	64 mA	Totem-Pole

## 2.10 Analogien und Unterschiede zwischen VMEbus und 680x0-Prozessoren

Signalleitung	68000	68030
D0 - D15	D0 - D15	D0 - D15
D16 - D31		D16 - D31
A1 - A23	A1 - A23	A1 - A23
A24 - A31		A24 - A31
AM0 - AM2	FC0 - FC2	FC0 - FC2
AM3 - AM5	%	%
/AS	/AS	/AS
/DS0	/LDS	A0, SIZE0, SIZE1
/DS1	/UDS	A0, SIZE0, SIZE1
/LWORD	%	A0, SIZE0, SIZE1
/WRITE	/RW	/RW
/SYSRESET	/RESET	/RESET
/BERR	/BERR	/BERR
/SYSFAIL	%	%
/ACFAIL	%	%
/IRQ1-/IRQ7	/IPL0-/IPL2	/IPL0-/IPL2
/IACK	/ (FC0 & FC1 & FC2)	/ (FC0 & FC1 & FC2)
/IACKIN	%	%
/IACKOUT	%	%
BR0 - BR3	BR	BR
BGOIN - BG3IN	BG	BG
BG0OUT-BG3OUT	%	%
BCLR	%	%
BBSY	%	%
SYSCLK	CLK	CLK
SERCLK	%	%
SERDAT	%	%

% = VMEbus-Signal bei 680x0-Prozessor nicht verfügbar

| = VMEbus-Signal wird aus diesem 680x0-Signal gewonnen

---

## 3. Der Expansion-Port Steckplatz des FALCON 030

Der neue ATARI FALCON 030 verfügt im Gehäuse über einen Expansion Port Steckplatz, auf dem nahezu alle Prozessor-Signale anliegen. An diesem Steckplatz können Erweiterungen, wie Coprozessoren (PC-Emulator), Graphikkarten oder das rhothron VMEbus-Interface angeschlossen werden.

### 3.1 Mechanische Details

Der Expansion Port befindet sich im vorderen linken Teil des FALCON Motherboards und ist erst nach Öffnen des Gehäuses und Entfernen des oberen Abschirmbleches zugänglich. Der Expansion-Port besteht aus einer 50-poligen (J 11) und einer 34-poligen Pfostenleiste (J 20). Auf der Leiste J 20 ist die Brücke W 11 standardmäßig gesetzt.

### 3.2 Elektrische Details

Wie erwähnt sind nahezu alle Prozessor Signale auf dem Expansion Port verfügbar. Zusätzlich sind einige Hilfssignale verfügbar, die die Busarbitrierung, die Interruptbearbeitung und die Bearbeitung "langer" Zugriffe betreffen. Auf der folgenden Seite ist die Belegung der beiden Pfostenleisten dargestellt.

### 3.3 Steckerbelegung

#### 30 pin, dual row, upright male header

Pin#	Signal	Pin#	Signal
1	D14	2	D13
3	D12	4	D11
5	D10	6	D9
7	D8	8	D7
9	D6	10	D5
11	D4	12	D3
13	D2	14	D1
15	D0	16	D15
17	GND	18	GND
19	GND	20	CPUBGO
21	EINT1	22	CPUBGI
23	500KHZ	24	n/c
25	MFP_IEI	26	MFP_INT
27	EINT3	28	VCC
29	VCC	30	VCC

#### 50 pin, dual row, upright male header

Pin#	Signal	Pin#	Signal
1	GND	2	GND
3	BGK	4	AS
5	LDS	6	UDS
7	RXW	8	DTACK
9	FC2	10	FC1
11	FC0	12	BMODE
13	n/c	14	IACK
15	BG	16	BR
17	RESET	18	HALT
19	BERR	20	IPL0
21	IPL1	22	IPL2
23	CPUCLK	24	VCC
25	VCC	26	A23
27	A22	28	A21
29	A20	30	A19
31	A18	32	A17
33	A16	34	A15
35	A14	36	A13
37	A12	38	A11
39	A10	40	A9
41	A8	42	A7
43	A6	44	A5
45	A4	46	A3
47	A2	48	A1
49	EXPAND	50	n/c

### 3.4 VME-Adreßbereich im FALCON 030 mit 4 MByte

Der FALCON 030 Rechner mit 4 MByte verfügt über einen Adreßbereich von 16 MByte. Dabei sind die unteren vier Megabyte (\$000000 bis \$3FFFFFF) für das Rechner-RAM, der Bereich von \$E00000 bis \$E7FFFF für das Betriebssystem ROM und die obersten siebenhundert Kilobyte (\$F90000 bis \$FFFFFF) für I/O-Bausteine reserviert.

Für den rhothron VMEbus ist folgender Adreßbereich reserviert:

"Standard-Adreßbereich A"

\$00A00000 bis \$00CFFFFFFF

"Short-Adreßbereich A"

\$00DF0000 bis \$00DFFFFFFF

"Standard-Adreßbereich B"

\$00F10000 bis \$00F7FFFF

"Short-Adreßbereich B"

\$00F90000 bis \$00F9FFFF

Der Adreßbereich \$F80000 bis \$F8FFFF ist für rhothron eigene Anwendungen (Seriennummer, etc.) reserviert.

Wird ein Zugriff auf diesen Speicherbereich durchgeführt, so werden die Adressen und Daten auf den VMEbus gelegt und die Steuersignale AS, DS0, DS1, WRITE aktiviert. Gleichzeitig wird dem FALCON mittels des EXPAND Signales signalisiert, das ein "längerer" externer Zugriff erfolgt und die interne Busfehler-Erzeugung deaktiviert werden muß. Damit der Rechner beim Ausbleiben des externen DTACK-Signales nicht endlos wartet und dadurch "hängt", ist auf dem rhothron VMEbus Interface ein Watchdog-Timer implementiert, der nach ca. 20 us ein Busfehlersignal generiert.

Jede VMEbus Baugruppe verfügt über einen Adreßkomparator, der die auf dem Bus anliegenden Adressen mit einer Solladresse vergleicht. Diese Solladresse kann meistens über einen oder mehrere DIP-Schalter, Brücken oder ähnliches auf jeder VMEbus Karte eingestellt werden. Jede VME-Karte muß dabei selbstverständlich eine von den anderen Karten verschiedene

Solladresse haben. Stimmt die auf dem Bus anliegende Adresse mit dieser Solladresse überein, so erkennt die VMEbus Karte, das sie angesprochen wird und führt Ihre Aufgabe aus. Die Beendigung ihrer Arbeit (Lesen, Schreiben, A/D-Wandeln, etc.) zeigt jede Baugruppe durch Aktivierung des Signales DTACK (Aktiv Low) an. Wird dieses Signal nicht innerhalb von 202 Mikrosekunden erzeugt, so erzeugt ein Watchdog-Timer (s.o.) einen Busfehler (BERR); es erscheinen die berühmten zwei Bomben.

Bei allen Baugruppen aus dem Hause rhothron wird die Solladresse mit einem achtpoligen DIP-Schalter bzw. bei neueren Versionen mit zwei Hex-Drehschaltern eingestellt. Es ergibt sich somit ein kleinster dekodierbarer Teilbereich von 64 kByte. Alle rhothron Baugruppen sind zur Vermeidung von Adreßkonflikten mit anderen VMEbus Baugruppen, z.B. Matrix Graphikkarte (\$A00000), standardmäßig auf den Adreßbereich von \$800000 bis \$BEFFFF eingestellt.

### 3.4 VME-Adreßbereich im FALCON 030 mit 14 MByte

Bei FALCON 030 Rechner mit 14 MByte RAM-Speicher ist der Speicherbereich bis \$DFFFFF komplett belegt. Aus diesem Grunde verbleibt für den rhothron-VMEbus Adapter nur noch folgender Adreßbereich, in dem bis zu sieben VMEbus-Karten angesprochen werden können.

"Standard-Adreßbereich B"

\$00F10000 bis \$00F7FFFF

"Short-Adreßbereich B"

\$00F80000 bis \$00F8FFFF

Der Adreßbereich \$F80000 bis \$F8FFFF ist für rhothron eigene Anwendungen (Seriennummer, etc.) reserviert.

Wird ein Zugriff auf diesen Speicherbereich durchgeführt, so werden die Adressen und Daten auf den VMEbus gelegt und die Steuersignale AS, DS0, DS1, WRITE aktiviert. Gleichzeitig wird dem FALCON mittels des EXPAND Signales signalisiert, das ein "längerer" externer Zugriff erfolgt und die interne Busfehler-Erzeugung deaktiviert werden muß. Damit der Rechner beim Ausbleiben des externen DTACK-Signales nicht endlos wartet und dadurch "hängt", ist auf dem rhothron VMEbus Interface ein Watchdog-Timer implementiert, der nach ca. 20 us ein Busfehlersignal generiert.

## 4. Das "rho-VMEbus Interface"

Das "rho-VME Interface" ist eine Platine, die in den Expansion-Port Steckplatz des FALCON 030 Rechners eingesteckt werden kann. Sie puffert alle Signale des FALCON normgerecht und stellt diese über zwei Flachbandkabel und eine kleine Adapterplatine auf einer 96-poligen VG-Buchse zur Verfügung. Diese Buchse kann nun mit jeder standardmäßigen VMEbus Backplane verbunden werden. Dabei empfiehlt sich die Verwendung einer solchen Backplane, die im ersten Slot, auf der Rückseite, über einen 96-poligen VG-Stecker verfügt. e

### 4.1 Einbau des Interfaces

Schalten Sie Ihren Rechner aus und entfernen Sie alle Anschlüsse, wie Netzkabel, Drucker- und Monitorkabel, etc.

Entfernen Sie nun den Deckel Ihres FALCON und anschließend das obere Abschirmblech.

Als nächstes sollte das Netzteil ausgebaut werden. Entfernen Sie nun den Jumper W 11 von der Pfostenleiste J 20 und stecken das rhothron VMEbus Interface auf die Pfostenleisten J 16 und J 20.

Die beiden Flachbandkabel werden nun gerade nach hinten geführt und dann auf Höhe des ROM-Ports rechtwinklig nach links umgelegt. Führen Sie die Flachbandkabel nun unter dem Abschirmblech durch und oberhalb des ROM-Ports seitlich links aus dem Gehäuse heraus. Stecken Sie zuletzt die beiden Flachbandkabel auf die externe Adapterplatine, montieren das Netzteil und schrauben den Rechner in umgekehrter Reihenfolge wieder zu.

Schalten Sie nun den Rechner wieder ein, stecken eine VMEbus Baugruppe auf den Bus und starten Sie das Programm ERKENN.PRG im "TEST"-Ordner. Betätigen Sie nun den "INFO"- Knopf. Die erscheinende Box gibt nun eine Übersicht über den verwendeten Rechner und die Art des Bussystemes. Falls der Einbau richtig vorgenommen wurde und sich die Basisadresse der VMEbus Karte im Adreßbereich des FALCON VMEbus befindet, zeigt ERKENN die Baugruppe als erkannt an.

## 4.2 Interrupt-Verarbeitung beim rhothron VMEbus Interface

Beim rhothron VMEbus Interface wird die Interrupt-Verarbeitung normgerecht über die Leitungen IRQ1, IRQ3, IRQ6, IACK, IACKIN, IACKOUT, A1..A3 durchgeführt. Das Signal IACK ist dabei auf dem Interface mit IACKIN verbunden. Daher muß das Interface von hinten auf den ersten Steckplatz einer VMEbus Backplane gesteckt werden, um so mit dem IACKIN-Anschluß der ersten Karte verbunden zu sein (siehe auch Daisy-Chain). Wegen des Daisy-Chaining der IACK-Leitung (IACKIN - IACKOUT - IACKIN - ...) müssen bei Interruptverarbeitung (Graphikkarte, Netzwerk-Karte, etc.) die VME-Steckplätze unbedingt von Slot 1 an aufwärts, ohne Unterbrechung, aufgefüllt werden. Also erste Karte in Steckplatz 1 (1. von links) und die nächste Karte in Slot 2 (2. von links).

## 4.3 Interrupt-Verarbeitung beim FALCON 030

IRQ 1 und IRQ 3 erzeugen beim FALCON einen Auto-Vektor-Interrupt und können daher sehr einfach abgearbeitet werden. Vorbedingung ist dabei natürlich, daß die Interruptmaske im Statusregister des Prozessors entsprechend gesetzt ist und eine eigene Interrupt-Service Routine eingebunden wird.

Eine Interrupt-Anforderung auf IRQ-Level 6 löst einen "normalen" Interrupt-Acknowledge Zyklus aus und sperrt gleichzeitig die internen Interruptquellen (z.B. MFP). Nach Aktivierung der IACK bzw. IACKIN Signales durch den FALCON und Überstimmung der Adreßleitung A1 bis A3 mit der Adresse 6 (IRQ 6), muß die Peripheriekarte den gewünschten IRQ-Vektor (\$40 bis \$FF) auf den Datenbus legen und den Zyklus mit DTACK bestätigen. Der Prozessor holt sich dann die Adresse der Interrupt-Service Routine aus der Speicherzelle IRQ-Vektor \* 4 und führt diese aus. Nach Ausführen dieser Routine und Rückkehr mittels RTE wird das unterbrochene Programm fortgesetzt. Zu beachten ist, das durch eine Interrupt-Service Routine in keinem Fall (!) Register verändert werden dürfen.

## 4.4 Cache ON / OFF

Der 68030 Prozessor des FALCON verfügt über einen internen Daten- und Instruktionen-Cache von jeweils 256 Byte. In diesem internen High-Speed RAM werden Daten bzw. Befehle zwischengespeichert um bei Bedarf ohne externen Zugriff direkt darauf zugreifen zu können. Dadurch erreicht man eine deutliche Beschleunigung der Programmlaufzeiten, also eine Erhöhung der Rechnerleistung.

Beim Zugriff auf externe Peripherie kann dieser Cache jedoch überaus unangenehme Randeffekte bewirken. Wird zum Beispiel das Statusregister einer VMEbus Baugruppe gepollt, um die Änderung eines Statusbits abzufragen, so holt sich der Prozessor beim ersten Zugriff den Wert des Statusregisters einmal in seinen Cache und liest beim nächsten Zugriff nur noch den Wert aus dem Cache. Da dieser sich jedoch im Gegensatz zum Statusregister nicht mehr ändert, wird die Änderung des Statusregister nicht mehr erkannt.

Um diesen Effekt auszuschließen, muß der Cache des Prozessors disabled werden. Zu diesem Zweck befinden sich unterhalb des Expansion-Ports auf der Oberseite des Motherboards zweimal acht Lötbrücken, U 46 und U47. Um den Cache auszuschalten muß bei U 46 die siebte Lötbrücke vom linken Platinenrand (Netzteil weist vom Betrachter weg) gesetzt werden.

## 4.5 Bus-Arbitrierung (optional)

Um mehrere FALCON 030 Rechner oder andere Prozessoren den Zugriff auf einen gemeinsamen VMEbus zu ermöglichen, müssen die Zugriffe synchronisiert bzw. die Zugriffsberechtigungen geregelt werden. Diese Aufgabe übernimmt allgemein ein sogenannter Arbitrer, in diesem Fall das "rho-VME-Interface".

Unter den verschiedenen Arbitrierungsmöglichkeiten arbeitet das "rho-VME-Interface" als Single Level Arbitrer auf Request Ebene 3. Das BBSY-Signal muß wegen der Verwendung nur einer Prioritätsebene nicht verwendet werden. Der am rho-VME-Interface angeschlossene ATARI Rechner arbeitet wegen des Single Level Modus in diesem Fall im "release, when done" Modus. Das heißt er fordert den Bus bei jedem Zugriff an und gibt den Bus nach jedem Zugriff direkt wieder frei. Alle anderen Master sollten ebenfalls diesen Modus wählen, um den Bus nicht unnötig lange zu blockieren. Wichtig ist selbstverständlich, daß der Arbitrer in Slot 1 steckt.

Falls das "rho-VME-Interface" als Arbitrer arbeiten soll, muß dies bei der Bestellung gesondert angegeben werden. Vorhandene Interfaces rüsten wir selbstverständlich gerne nach.

## 4.6 Stromversorgung

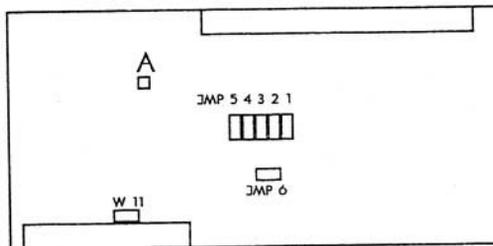
Das rho-VMEbus Interfaces benötigt nur + 5 Volt Spannung und wird direkt vom Netzteil des FALCON versorgt. Über die beiden Flachbandkabel wird diese + 5 Volt Spannung auf den 96-poligen VME-Adapter geführt. Falls nur eine VMEbus Karte ohne Backplane am VMEbus betreiben werden soll, können die + 5 Volt des Rechner über einen Jumper auf der Adapterplatine auf die 96-polige VG-Leiste aufgelegt werden.

Zusätzlich zu den + 5 Volt können bei Bedarf auch die + 12 Volt auf die Adapterplatine geführt werden. Zu diesem Zweck ist Punkt A auf der VMEbus Puffer-Platine mit den + 12 Volt des Netztesles zu verbinden (Lötverbindung). Zusätzlich ist der Jumper auf der Adapterplatine zu setzen.

Falls eine Backplane für mehrere VMEbus-Karten verwendet wird, müssen die Stromversorgungs-Jumper entfernt und die Backplane durch ein eigenes Netzteil versorgt werden. Werden die Jumper nicht entfernt, so sind beide Netzteile miteinander verbunden, was einen ernsten Defekt zur Folge haben könnte.

## 4.7 Jumper des VMEbus Interfaces

Auf dem VMEbus Interface befinden sich in der Mitte der Platine fünf, nebeneinanderliegende Steckbrücken. Das folgende Schema zeigt die Belegung.



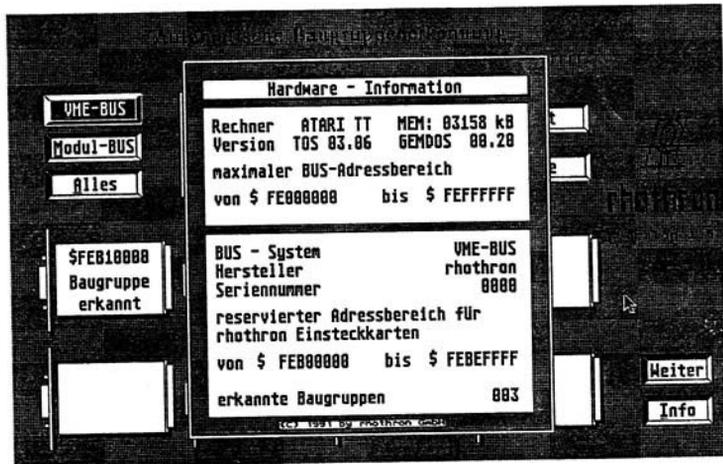
Mittels der Steckbrücke JMP 5 kann der VMEbus mit dem 16 MHz Systemtakt (CLK) versorgt werden. Falls der VMEbus eine eigene Takterzeugung hat, muß diese Verbindung gelöst, also der Jumper entfernt werden. Wird nur eine Platine am VMEbus betrieben, so muß diese Brücke immer gesetzt sein.

Jumper JMP 2 dient zur Zuschaltung des Adreßbereiches \$A00000 bis \$DFFFFF.



Eigenentwicklungen Timinganalysen einfacher durchführen zu können.

Neben dem Speichertest dient das Programm RHOTEST.PRG zum Auslesen der Seriennummer, zum Test des IEEE-488 Interfaces und zum Festplatten-Test.



## 4.9 Das Programm ERKENN.PRG

Nach Aufruf des Programmes ERKENN.PRG wird zuerst die Art des verwendeten Rechners, die Seriennummer und die Art des Bus-Systemes (VMEbus / Modul-Bus) abgetestet. Anschließend wird in Abhängigkeit von den zuerst ermittelten Daten der Speicherbereich des Bus-Systems auf das Vorhandensein von Baugruppen überprüft. Zu diesem Zweck werden in 64 kByte Schritten Dummy-Zugriffe auf den Bus durchgeführt und überprüft, ob ein BERR oder DTACK Signal auftritt. Tritt ein Bus-Fehler auf, so ist unter dieser Adresse keine Baugruppe vorhanden (Ausnahme: Matrix-Graphikkarte vor der Installation!). Ist der Test beendet, so werden die Adressen der "gefundenen" Baugruppen angezeigt. Sind mehr als acht Baugruppen vorhanden, so kann mit Hilfe des "Weiter"-Knopfes die nächste Achtergruppe abgerufen werden. Der Knopf "Test" wiederholt die Prüfung, der Knopf "Info" gibt grundlegende Information über Rechner- und Bustyp, sowie über die Seriennummer an.

Nach Druck auf den "Ende"-Knopf kehrt das Programm in die aufrufende Betriebssystem-Ebene zurück.