

M I C R O P R O C E S S O R S
B U S S T R U C T U R E

■
**IEEE Standard for a
High-Performance
Synchronous 32-Bit
Bus: MULTIBUS II**

Includes ANSI/IEEE Std 1101-1987,
IEEE Standard for Mechanical Core
Specifications for Microcomputers



Published by
The Institute of Electrical and
Electronics Engineers, Inc

SH11619



Documents
Document Sharing Hub

An American National Standard

**IEEE Standard for a
High-Performance Synchronous 32-Bit Bus:
MULTIBUS* II**



Published by
The Institute of Electrical and Electronics Engineers, Inc

*MULTIBUS is a registered trademark of Intel Corporation.

IEEE Standards documents are developed within the Technical Committees of the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Board. Members of the committees serve voluntarily and without compensation. They are not necessarily members of the Institute. The standards developed within IEEE represent a consensus of the broad expertise on the subject within the Institute as well as those activities outside of IEEE which have expressed an interest in participating in the development of the standard.

Use of an IEEE Standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least once every five years for revision or reaffirmation. When a document is more than five years old, and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of all concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason IEEE and the members of its technical committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE Standards Board
345 East 47th Street
New York, NY 10017
USA

IEEE Standards documents are adopted by the Institute of Electrical and Electronics Engineers without regard to whether their adoption may involve patents on articles, materials, or processes. Such adoption does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the standards documents.

**ANSI/IEEE
Std 1296-1987**

An American National Standard

**IEEE Standard for a
High-Performance Synchronous 32-Bit Bus:
MULTIBUS II**

Sponsor
**Technical Committee on Microprocessors and Microcomputers
of the
IEEE Computer Society**

Approved June 11, 1987
IEEE Standards Board

Approved February 8, 1988
American National Standards Institute

Abstract

This standard describes a high-performance backplane bus intended for use in multiple processor systems. The bus incorporates synchronous, 32-bit multiplexed address/data, with error detection and uses a 10 MHz bus clock. This design is intended to provide reliable state-of-the-art operation and to allow the implementation of cost-effective high performance VLSI for the bus interface. The standard defines memory, I/O, message, and geographic address spaces. The memory space supports single and block transfers. The message space defines a high-level protocol for transferring blocks of data (messages) as multiple packets of data (small data bursts; e.g., 32 bytes in one microsecond) over the bus. This provides a high-performance (e.g., 32 megabytes/second) interprocessor communication transfers among many different processors on the bus. Error detection and retry is provided for messages. This message passing design also allows a VLSI implementation, such that virtually all modules on the bus will utilize the bus at its highest performance — 32 to 40 megabytes/second. The standard provides geographic addressing for ease of configurability, initialization, and diagnostics.

ISBN 0-471-50065-8

Library of Congress Catalog Number 88-081186

© Copyright 1988 by

The Institute of Electrical and Electronics Engineers, Inc
345 East 47th Street, New York, NY 10017, USA

*No part of this publication may be reproduced in any form,
in an electronic retrieval system or otherwise,
without the prior written permission of the publisher.*

August 3, 1988

SH11619

Foreword

(This Foreword is not a part of ANSI/IEEE Std 1296-1987, IEEE Standard for a High-Performance Synchronous 32-Bit Bus: MULTIBUS II.)

In the last decade, the avalanche of new microcomputer technology, especially VLSI, threatened to obsolete products almost before they went into production. To buffer users from this onrush of technology, Intel helped develop standard interfaces. One of the most notable was the MULTIBUS I system bus, which was used as the basis for a standard by the IEEE in 1983 as IEEE Std 796-1983 (after going through a 5-year review and revision process).

In the early 1980s, Intel recognized that the trends toward multiprocessing and more sophisticated microcomputer-based systems called for an advanced 32-bit system bus architecture. Intel called this new bus MULTIBUS II. In continuing to pioneer the open systems technology, which included multiprocessing, four critical requirements were observed: technical credibility, processor independence, standardization, and openness to all levels of integration. Early in the development of the new bus, Intel established a "MULTIBUS II Development Consortium." The detailed technical involvement and review by this consortium provided significant thoroughness and depth in the development of the new bus. This consortium included major and technically astute electronics companies. These included:

AMD	Hewlett-Packard	NCR Corporation
Altos	International Computers Limited	Nixdorf
Compagnie des Machines Bull	Intersil Systems	Prime
Diab	Kienzle	Siemens
Ericsson Information Systems	Matra	Tektronix
Foxboro	Microbar Systems	Zilog
	Mupac	

The consortium gave the new bus a technical credibility that few buses, especially those defined only among board vendors, can match. These companies also represented all microprocessor families; included in the group were 68020, 32032, 80386, and Z8000 board and system users, thus ensuring that the bus is easily adaptable to virtually any manufacturer's processor.

The primary benefits being sought in the creation of this new bus were high-performance multiprocessing, high system reliability, ease-of-use by system designers, and improved cost/performance.

Specific bus features were developed in response to these objectives. The 32 megabytes/second message passing of the bus provides a bus that acts like a very high-speed network connection for multiple processors (or processor equivalents).

There is a recognition that the bus is no longer to interconnect a CPU with its memory and I/O; instead the bus is to interconnect whole stand-alone processors with each other and with intelligent "subsystems-on-a-board."

System reliability is enhanced by the features of bus parity, synchronous operation, negative acknowledge, transfer retries, geographic addressing, and advanced backplane design. Ease-of-use by system designers is implemented primarily through the geographic addressing which provides for dynamic system configuration. The bus encourages the use of software programmable configuration options (and discourages any use of mechanical jumpers). The standardization of the high-level message passing protocol also gives the system designer an easy-to-use capability for interprocessor communication.

The cost/performance objective of the bus is delivered through its specification of a realizable 32 to 40 MByte/sec bus bandwidth. Virtually all boards designed to the bus can achieve this bus utilization factor due to the high-level protocol called out in the specification, and thus the availability of standard, high-performance and cost-effective VLSI components to actually implement this level of performance. For example this specification and the VLSI make it possible for eight concurrent 4 megabyte/second transfers to take place on the bus. This, or other combinations of transfers that add up to 32 megabytes/second, demonstrate the real cost/performance advantages of the bus for multiprocessor applications.

In 1983 MULTIBUS II was introduced to the IEEE standards process as a part of the considerations for the P896 (Future Bus) working group activities. In the 1984/1985 time frame the MSC (Microcomputer Standards Committee, of the TCMM) formed an independent study group for MULTIBUS II. During this time the many active participants of the group proceeded to thoroughly review and make changes to the proposed draft. In early 1986 the group was assigned a formal project number P1296. During the remainder of 1986, the draft was passed by the Working Group and the MSC after thorough review, discussion, and changes. In 1987, the draft was presented for Sponsor ballot and, after passing, presented to the June 1987 meeting of the IEEE Standards Board. The standard is written with regard to ISO/IEC documentation requirements to support later submission to the ISO/IEC.

The IEEE Standards Board calls attention to the fact that there are patents claimed and/or pending on many aspects of this bus by Intel Corporation. IEEE takes no position with respect to patent validity. Intel Corporation has assured the IEEE that it is willing to grant a license for these patents on reasonable and nondiscriminatory terms to anyone wishing to obtain such a license. The general terms of the license are a one-time administration fee of \$100 for a nonexclusive perpetual license. Intel Corporation's undertakings in this respect are on file obtained from the legal department of Intel Corporation whose address is Intel

Corporation, 5200 N.E. Elam Young Parkway, Hillsboro, OR 97124.

There were many contributors to the standards review process but the following members deserve special mention for their active participation:

Task Force Coordinators: Jack Blevins
Maurice Hubert
Hubert Kirmann
Jim Nebus

Secretary of Working Group: Steve Cooper
Original Study Group Chairman: Paul Borrill
Original Draft Editor: Scott Tetrick

The P1296 Working Group that prepared this standard had the following membership:

Richard W. Boberg, *Chairman*

Web Augustine
Jack Blevins
Paul Borrill
Steve Cooper
Tom Crawford

Gene Freehauf
Maurice Hubert
Hubert Kirmann
Klaus Mueller
Jim Nebus
Don Nickel

Ken Smith
Michael Thompson
Scott Tetrick
Eike Waltz
Janusz Zalewski

The following members of the Technical Committee on Microprocessors and Microcomputers were on the balloting body that approved this document for submission to the IEEE Standards Board:

Andrew Allison	Martin Freeman	Deene Ogden
Peter J. Ashenden	David Gustavson	Tom Pittman
Matt Biewer	Tom Harkaway	Shlomo Pri-Tal
John Black	Dave Hawley	P. Reghunathan
Jack Blevins	David James	Richard Rawson
Richard Boberg	Laurel Kaleda	Bill Shields
Paul Borrill	Richard Karpinski	Michael Smolin
Bradley Brown	Hubert Kirmann	Robert Stewart
Clyde Camp	Doug Kraft	Subramanganesan
John D. Charlton	Tom Kurihara	Michael Teener
Steve Cooper	Glen Langdon	Scott Tetrick
Randy Davis	Gerry Laws	Eike Waltz
J. Robert Davis	Tom Leonard	Carl Warren
Shirish P. Deodhar	Rollie Linser	George White
Jim Dunlay	Gary Lyons	Fritz Whittington
Wayne Fischer	James Nebus	Tom Wicklund
Jim Flournoy	Gary Nelson	Andrew Wilson
Gordon Force		Anthony Winter

When the IEEE Standards Board approved this standard on June 11, 1987, it had the following membership:

Donald C. Fleckenstein, *Chairman* **Marco W. Migliaro**, *Vice Chairman*
Andrew G. Salem, *Secretary*

James H. Beall	Leslie R. Kerr	L. John Rankine
Dennis Bodson	Jack Kinn	John P. Riganati
Marshall L. Cain	Irving Kolodny	Gary S. Robinson
James M. Daly	Joseph L. Koepfinger*	Frank L. Rose
Stephen R. Dillon	Edward Lohse	Robert E. Rountree
Eugene P. Fogarty	John May	Sava I. Sherr*
Jay Forster	Lawrence V. McCall	William R. Tackaberry
Kenneth D. Hendrix	L. Bruce McClung	William B. Wilkens
Irvin N. Howell	Donald T. Michael*	Helen M. Wood

*Member emeritus

Contents

Part I—Introduction to the IEEE 1296 Standard

SECTION	PAGE
1. Scope and References	19
1.1 Scope	19
1.2 References	19
2. Definitions	21
3. Guide to Notation	27
3.1 General	27
3.2 Signal Notation	27
3.3 Figure Notation	28
3.4 Notation in State-Flow Diagrams	28
3.5 Notation for Multiple Bit Data Representation	28

Part II—IEEE 1296 PSB Standard

4. PSB Overview	31
4.1 General	31
4.2 Address/Data Path and System Control Signals	31
4.3 Message Passing Facility	33
4.4 Interconnect Facility	33
4.5 Synchronous Operation of the PSB	33
4.6 Bus Operations on the PSB	33
4.6.1 Arbitration Operation	34
4.6.2 Transfer Operation	35
4.6.3 Exception Operation	37
4.7 Central Services Module	37
5. Signal Descriptions	39
5.1 General	39

SECTION	PAGE
5.2 Signal Groups	39
5.2.1 Arbitration Operation Signal Group	40
5.2.1.1 BREQ* (Bus Request)	40
5.2.1.2 ARB* (Arbitration)	40
5.2.2 Address/Data Bus Signal Group	41
5.2.2.1 AD* (Address/Data Bus)	41
5.2.2.2 PAR* (Parity)	41
5.2.3 System Control Signal Group	41
5.2.3.1 SC0* (Request Phase Or Reply Phase)	42
5.2.3.2 SC1* (Lock)	42
5.2.3.3 SC2* (Data Width Or End-Of-Transfer Operation) ...	42
5.2.3.4 SC3* (Data Width Or Bus Owner Ready)	43
5.2.3.5 SC4* (Address Space Or Replier Ready)	44
5.2.3.6 SC5* (Address Space Or Agent Status)	44
5.2.3.7 SC6* (Read-Write Or Agent Status)	44
5.2.3.8 SC7* (Reserved Or Agent Status)	44
5.2.3.9 SC8* (Parity On SC<7..4>*)	46
5.2.3.10 SC9* (Parity On SC<3..0>*)	46
5.2.4 Exception Operation Signal Group	46
5.2.4.1 BUSERR* Exception	46
5.2.4.2 TIMOUT* Exception	46
5.2.5 Central Control Signal Group	46
5.2.5.1 Reset (RST*)	47
5.2.5.2 Reset Not Complete (RSTNC*)	47
5.2.5.3 DC Power Low (DCLOW*)	47
5.2.5.4 Protect (PROT*)	47
5.2.5.5 Bus Clock (BCLK*)	47
5.2.5.6 Central Clock (CCLK*)	47
5.2.5.7 ID Latch (LACHn*)	47
5.2.6 Power	48
6. PSB Protocol	49
6.1 General	49
6.2 Arbitration Operation	49
6.2.1 Agent Arbitration ID and Cardslot ID Assignment	50
6.2.2 Bus Arbitration Priority	52
6.2.2.1 Normal Priority	52
6.2.2.2 High Priority	53
6.2.3 Bus Ownership	53

SECTION	PAGE
6.2.4	Parking and Bus Release 54
6.2.5	Arbitration Priority Example 54
6.2.6	Resolution Algorithm Example 59
6.3	Transfer Operation 65
6.3.1	Types of Transfer Operations 66
6.3.1.1	Single Data Transfer 66
6.3.1.2	Sequential Data Transfer 67
6.3.1.3	Broadcast Message 67
6.3.2	Transfer Operation Lock 69
6.3.3	Agent Status 71
6.3.3.1	No Error 76
6.3.3.2	Negative Acknowledge Error 76
6.3.3.3	Transfer-Width Error 76
6.3.3.4	Continuation Error 78
6.3.3.5	Data Error 78
6.3.3.6	Transfer-Not-Understood Error 78
6.3.3.7	Priority of Errors 78
6.3.4	Address Space Definitions in Transfer Operations 80
6.3.4.1	Memory Space Attributes 80
6.3.4.2	I/O Space Attributes 81
6.3.4.3	Message Space Attributes 83
6.3.4.4	Interconnect Space Attributes 83
6.3.5	Data Width During Transfer Operations 84
6.3.5.1	Data Alignment in Memory and I/O Spaces 85
6.3.5.2	Data Alignment in Message Space 86
6.3.5.3	Data Alignment in Interconnect Space 86
6.3.6	Transfer-Width Error Reporting During Transfer Operations .. 87
6.3.7	Data Alignment Interface Example 87
6.4	Exception Operation 88
6.4.1	Causes of Exception Operations 91
6.4.1.1	Time Out Exception 91
6.4.1.2	Bus Error Exception 92
6.4.2	Priority of Errors 93
6.5	Central Control Functions 93
6.5.1	Clock Source 95
6.5.2	System Level Resets 95
6.5.2.1	Power-Up Function (Cold-Start) 95
6.5.2.2	Initialization Sequence (Warm-Start) 96
6.5.2.3	Power Failure and Recovery Sequence 97

SECTION	PAGE
6.5.3 Arbitration ID and Cardslot ID Assignment	99
6.5.4 Time Out Monitoring	101
6.6 State-Flow Diagrams	102
6.6.1 State-Flow Sequence for an Agent Monitoring the Bus	103
6.6.2 State-Flow Sequence for an Arbitration Operation	104
6.6.3 State-Flow Sequence for Bus Owners in a Transfer Operation	110
6.6.4 State-Flow Diagram for Replying Agents in a Transfer Operation	114
6.6.5 Effects of Exception Operations on State-Flow Diagrams	117
7. Electrical Characteristics	119
7.1 General	119
7.2 AC Timing Specifications	120
7.3 DC Specifications for Signals	127
7.4 Current Limitations Per Connector	130
7.5 Pin Assignments	130
7.5.1 PSB Pin Assignments	130
7.5.2 P2 Recommended Power Pin Assignments	131
7.6 References	132
8. Mechanical Specifications	133
8.1 General	133
8.2 Board Sizes and Dimensions	134
8.3 Printed Board Layout Considerations	136
8.4 Front Panel	136
8.5 Connectors	136
8.6 Backplanes	138
8.6.1 Mounting	138
8.6.2 Backplane Design	143
8.6.3 PSB Backplane Specifications	143
8.6.4 PSB Backplane Characteristics	145
8.7 References	147
Part III—IEEE 1296 System Interface Specification	
9. General	149
9.1 Scope	149
9.2 Object	149
9.3 Architecture Overview	149
9.4 System Model	149

SECTION	PAGE
9.4.1 Memory-Mode Architecture	150
9.4.2 Message-Mode Architecture	150
9.4.3 Dual-Mode Architecture.....	150
10. Interconnect Space Operation	151
10.1 Introduction	151
10.2 Interface Model.....	151
10.3 Interconnect Address Format	151
10.4 Vendor Identification Registers.....	153
10.5 Compliant Interconnect Template Structure	153
10.5.1 Description of the Header Record	153
10.5.1.1 Product Code Registers	155
10.5.1.2 Vendor Defined Registers.....	155
10.5.2 Description of Function Records	155
10.5.3 Function Record Definitions	157
10.5.3.1 Extended Record	157
10.5.3.2 CSM Record.....	159
10.5.3.3 Time/Date Record	160
10.5.3.4 Memory Space Record	166
10.5.3.5 I/O Space Record	166
10.5.3.6 Message Space Record	166
10.5.3.7 Initialization Record	167
10.5.3.8 End of Template Record	167
10.6 Minimum Compliant Template	167
11. I/O Space Operation	169
11.1 Introduction	169
11.2 Interface Model.....	169
11.3 Address Assignment Rules	169
12. Memory Space Operations	171
12.1 Introduction	171
12.2 Interface Model.....	171
12.3 Address Assignment Rules.....	171
13. Message Space Operations	173
13.1 Introduction	173
13.2 Interface Model.....	173
13.3 Address Assignment Rules.....	173
13.4 Message Space Operation	174
13.5 Message Model	174

SECTION	PAGE
13.5.1 Unsolicited Messages.....	175
13.5.1.1 Characteristics of Unsolicited Messages	176
13.5.1.2 Unsolicited Message Types.....	177
13.5.2 Solicited Messages	178
13.5.2.1 Characteristics of Solicited Messages	178
13.5.2.2 Data Packets	178
13.5.2.3 Liaison ID	178
13.5.2.4 Duty Cycle	178
13.6 Solicited Data Transfer Protocol	179
13.6.1 Negotiation	179
13.6.2 Transfer	180
13.6.3 Completion	181
13.7 Packet Formats.....	181

Part IV—IEEE 1296 Capabilities

14. Characteristic Codes	187
14.1 Introduction	187
14.2 P2 Connector.....	187
14.3 Form Factor	188
14.4 References	188

FIGURES

3.3-1 Figure Notation Example	29
4.1-1 Block Diagram of the PSB Interface	32
4.6-1 Bus Operation Relationships	34
4.6-2 Block Diagram of Bus Operations	36
4.6-3 Block Diagram of Bus Operations with Errors	38
5.2-1 Backplane Connection of LACHn*	48
6.1-1 Bus Operation Details	50
6.2-1 Reset Sequence	51
6.2-2 Timing Sequence with Bus Locked	55
6.2-3 Timing Sequence for Bus Release	56
6.2-4 Typical Arbitration Operation Sequence	57
6.2-5 Arbitration ID Interface Example	60
6.2-6 Timing Sequences for Bus Acquisition	63
6.2-7 Timing Sequence for High Priority Bus Acquisition	64
6.3-1 Block Diagram of Single Read Data Transfer	67
6.3-2 Timing for a Single Read Data Transfer	68

FIGURES	PAGE
6.3-3 Block Diagram of Single Write Data Transfer	69
6.3-4 Timing for Single Write Data Transfer	70
6.3-5 Block Diagram of Sequential Data Transfer	71
6.3-6 Timing for Sequential Read Data Transfer	72
6.3-7 Timing for Sequential Write Data Transfer	73
6.3-8 Block Diagram of Broadcast Message	74
6.3-9 Timing for Broadcast Message	75
6.3-10 Timing Sequence for LOCK Operation	76
6.3-11 Timing Sequence for Transfer-Width Error	77
6.3-12 Timing Sequence for Continuation Error	79
6.3-13 Address Format for Sequences Using Memory Address Space	81
6.3-14 Block Diagram of Sequential Data Transfers in Memory Space	82
6.3-15 Address Format for Sequences Using I/O Address Space	83
6.3-16 Address Format for Sequences Using Message Address Space	84
6.3-17 Address Format for Interconnect Address Space	85
6.3-18 Interface Requirements for Data Alignment	89
6.4-1 Exception Cycle Signal Relationships	90
6.4-2 Signalling a TIMEOUT* Exception	91
6.4-3 Signalling a BUSERR* Exception	94
6.5-1 Power-on System Reset Sequence	96
6.5-2 Warm-Start Reset Sequence	97
6.5-3 Power Failure Recovery Sequence	98
6.5-4 Example Arbitration/Cardslot ID Assignment Timing Sequence	100
6.5-5 Reset Sequence	102
6.6-1 State-Flow Diagram for Monitoring Transfer Operations	104
6.6-2 State-Flow Diagram for Agents Monitoring Arbitration Operations ..	105
6.6-3 State-Flow Diagram for Arbitration Operations	107
6.6-4 State-Flow Diagram for Requesting Agents in a Transfer Operation ..	111
6.6-5 State-Flow Diagram for Replying Agents	115
7.2-1 BCLK* and CCLK* Timing Relationships at the CSM	121
7.2-2 BCLK* and CCLK* Timing Relationships for Receivers	122
7.2-3 Driver Timing Parameters	123
7.2-4 Timing Parameters for Signal Receivers	124
7.2-5 Cold-Start Timing Parameters	125
7.2-6 Warm-Start Timing Parameters	126
7.2-7 Power Failure and Recovery Timing Parameters	127
8.1-1 Double Eurocard Connector Definitions	134
8.1-2 Triple Eurocard Connector Definitions	135
8.1-3 Typical Subrack	136

FIGURES	PAGE
8.2-1 Double High Board Dimensions	137
8.3-1 Board Layout Considerations for Front Panels	138
8.4-1 Front Panel Dimensions	139
8.4-2 Board Spacing and Front Panel Relationships	140
8.5-1 Relationship of Boards and Connectors	141
8.6-1 Backplane and Mounting Rail Relationship	142
8.6-2 Backplane Connector Layout	144
8.6-3 Off-board Termination	145
8.6-4 On-board Termination	146
8.6-5 PSB Backplane Layer Thickness	146
10.3-1 Interconnect Address Format	152
10.3-2 Interconnect Space Format	152
10.5-1 Structure of the Compliant Interconnect Template	154
10.5-2 Function Record Chaining	156
10.5-3 Function Record Format	157
10.5-4 Interconnect Template Organization	158
10.5-5 Flowchart for Reading Date and Time	163
10.5-6 Flowchart for Writing Time	164
10.5-7 Flowchart for Writing Date	165
13.4-1 Unsolicited Message Format	175
13.4-2 Solicited Message Format	176
13.6-1 Solicited Message Transfer	179
13.7-1 General Interrupt Message Packet	182
13.7-2 Broadcast Interrupt Message Packet	182
13.7-3 Buffer Request Message Packet	184
13.7-4 Buffer Grant Message Packet	185
13.7-5 Buffer Reject Message Packet	185
13.7-6 Data Packet	186
TABLES	
3.2-1 Signal Notation	27
5.2-1 Signal Groups on the PSB	39
5.2-2 Summary of Functions of SC* Signals	42
5.2-3 Functions of SC* During REQUEST Phase	43
5.2-4 Functions of SC* During Reply Phase	45
6.2-1 Arbitration Priority Protocol	53
6.2-2 Arbitration ID Comparison Example	61
6.3-1 Agent Status Codes	74
6.3-2 Address Space Summary	80

TABLES	PAGE
6.3-3 Data Alignments for Operations Using Memory and I/O Space	86
6.3-4 Data Alignment for Message Space Operations	87
6.3-5 Data Alignment for Operations Using Interconnect Space	87
6.3-6 Width Errors for Memory and I/O Space Operations	88
6.5-1 Cardslot and Default Arbitration ID Assignment.....	101
7.1-1 Abbreviations Used for the Electrical Specification.....	119
7.2-1 Clock Specification for the CSM	121
7.2-2 Clock Specification for Receivers	122
7.2-3 Timing Parameters for Signal Driver	123
7.2-4 Timing Parameters for Signal Receivers	124
7.2-5 Cold-Start Control Timing	125
7.2-6 Warm-Start Control Timing	126
7.2-7 Power Failure and Recovery Timing	127
7.3-1 DC Specifications for Signal Drivers	128
7.3-2 DC Specifications for Signal Receivers	129
7.3-3 Backplane Termination Requirements	129
7.4-1 Power Limitations for a One-Connector Agent.....	130
7.4-2 Power Limitations for a Two-Connector Agent	130
7.5-1 PSB Connector Pinout	131
7.5-2 P2 Connector Recommended Power Pinout	132
8.6-1 PSB Backplane Characteristics	143
8.6-2 Cardslot Numbering Sequence for Large Backplanes	143
10.1-1 Interconnect Address Space Summary.....	151
10.5-1 Header Record Format	154
10.5-2 Predefined Function Records	157
10.5-3 Format of the Extended Record	159
10.5-4 CSM Record Format	159
10.5-5 Reset Codes for the CSM Command Register	160
10.5-6 Format of the Time/Date Record.....	161
10.5-7 Time/Date Commands	162
10.5-8 Day of Week Register Encoding	166
10.5-9 End of Template Record Format	167
10.6-1 Minimum Compliant Interconnect Template Format	167
11.1-1 I/O Address Space Summary	169
12.1-1 Memory Address Space Summary	171
13.1-1 Message Address Space Summary	173
13.5-1 Duty Cycle Computation	179
13.7-1 Packet Formats on the Bus	181
14.1-1 Characteristic Codes	187

APPENDIX	PAGE
Recommended Documentation Practices	189
A1. Introduction	189
A2. Bus Owner Address Space Support	189
A3. Replying Agent Address Space Support	189
A4. Data Path	189
A5. Message Support	189
A6. CSM Support	190
A7. Sequential Transfer Support	190
A8. Power Fail Support.....	190
A9. Interconnect Template Support	190
A10. Temperature and Airflow Assumptions	190
A11. RSTNC* Assertion Time.....	190
A12. Example of Recommended Documentation Practices	190
APPENDIX TABLE	
A.12-1 Recommended Documentation Example	191

PART I—Introduction to the IEEE 1296 Standard

1. Scope and References

1.1 Scope. This document defines the operation, functions, and attributes of the IEEE 1296 bus standard.

- (1) This standard defines a high-performance 32-bit synchronous bus standard.
- (2) The bus standard must have a design-in lifetime of 10 years with backward compatibility.
- (3) The standard is intended for general purpose applications to optimize block transfers, including protocol for message passing. For real-time applications, the bus will provide a means of ensuring an upper limit to message delivery time.
- (4) The standard is intended to be compatible with existing IEC [1], [2], [3]¹ mechanical standards with recognition of the need for special front panels to address ESD, EMI, and RFI requirements.
- (5) Options within the standard will be clearly identified.
- (6) The standard is intended to support multiple processor modules in a functionally partitioned configuration and heterogeneous processor types in the same system.
- (7) The standard is intended to support heterogeneous processor types in the same system.
- (8) Message passing format and protocol is intended for future migration to a serial system bus.

1.2 References. The following publications shall be used in conjunction with this standard:

- [1] IEC Pub 297-1 (1986), Dimensions of Mechanical Structures of the 482.6 mm (19 in) Series—Part 1: Panel and Racks.²

¹ The numbers in brackets correspond to those of the references listed at the end of this chapter.

² IEC publications are available in the US from the Sales Department, American National Standards Institute, 1430 Broadway, New York, NY 10018.

- [2] IEC Pub 297-3 (1984), Part 3: Subracks and Associated Plug-In Units.
- [3] IEC Pub 603-2 (1980), Connectors for Frequencies Below 3 MHz for Use with Printed Boards—Part 2: Two-Part Connectors for Printed Boards, for Basic Grid of 2.54 mm (0.1 in), with Common Mounting Features.

2. Definitions

The following definitions apply for the IEEE 1296 Standard. This section contains all definitions for subsequent parts of the standard.

acquisition phase. The final phase of the arbitration operation entered after determining that an agent has the highest priority and the bus is available. See: arbitration operation; agent.

address transfer. The passing of address information over the multiplexed Address/Data bus from the bus owner in order to select the replying agent(s). See: bus owner; replying agent.

address/data bus signal group. A set of thirty-six (36) signals, consisting of 32 address/data signals and four parity signals that are used for address and data transfers.

agent. A physical unit that has an interface to the Parallel System bus, for example, a single-board computer.

agent error. An agent status which indicates an error condition in a replying agent.

agent status. The condition of the replying agent, transmitted during the reply phase of a transfer operation. See: reply phase; transfer operation.

arbitration operation. The bus operation in which agents attempt to gain exclusive access to the Parallel System bus.

backplane. The physical mechanism by which signals are routed between agents.

bit. A binary digit (abbreviation “b”).

broadcast message. A sequence of one or more data transfers from the bus owner to all replying agents, with uninterrupted bus ownership.

bus clock cycle. An amount of time equal to one bus clock period, nominally 100 nanoseconds.

bus loss. The amount of time required for a valid signal transition to occur at every point on the backplane. This value is equivalent to two bus propagation delays plus the clock skew.

bus operation. The basic unit of processing whereby digital signals effect the transfer of data across an interface by means of a sequence of control signals and an integral number of bus clock cycles.

bus owner. The agent that enters the acquisition phase of the arbitration operation and initiates one or more transfer operations. See: acquisition phase; arbitration operation; transfer operation.

bus request sequence. A set of one or more arbitration operations in which all agents that simultaneously request the bus become the bus owner, one at a time. See: arbitration operation; bus owner.

byte. A group of eight adjacent bits operated on as a unit (abbreviation "B").

central services module. A specific module that is required in all systems using the Parallel System bus (abbreviation "CSM"). Its services, such as starting certain bus operations and guaranteeing uniform initialization of all agents, are required by all agents on the Parallel System bus. It is always located in a specific slot in the system backplane. See: parallel system bus.

client. An agent that requests services of a server. See: server.

cold-start. A sequence of events performed on the application of power that ensures a uniform initialization period for all agents, giving them the ability to begin operation from a known state.

command transfer. The passing of command information over the System Control signal group, from the bus owner to the replying agent(s), during the request phase of a transfer operation. Command information includes parameters for the impending transfer operation, as well as additional address space information not

transmitted with the address transfer. See: system control signal group; request phase.

data transfer. The passing of data over the multiplexed Address/Data bus, between the bus owner and the replying agent(s), during the reply phase of a transfer operation.

driving agent. The agent that is permitted to assert or negate a signal on the bus.

dual-mode agent. An agent that supports both memory-mode and message-mode communication on the Parallel System bus. See: memory-mode agent; message-mode agent.

dual-mode system. A system that supports both memory-mode and message-mode communication. A mixture of both communication types is used on the Parallel System bus.

end of transfer status. A handshake status which indicates the last data transfer of the transfer operation (abbreviation "EOT Status"). See: handshake status.

exception. An abnormal condition on the bus caused by either a bus parity error, a bus time-out, a protocol violation, or a bus owner reply phase termination.

exception operation. A bus operation in which an agent places an error indication on the Parallel System bus. The error indication causes all bus agents to terminate arbitration and transfer operations.

handshake status. A status transfer which indicates the exchange of data between bus owner and replying agent(s).

I/O space. The address space used for accessing peripheral devices such as communication controllers and mass storage devices.

interconnect space. The address space used for board identification, system configuration, and board specific functions such as testing and diagnostics.

interconnect template. A definition of the contents of the interconnect space of an agent.

interface. A shared boundary between modules or agents of a computer system, through which information is conveyed.

locked. A condition of the bus which guarantees exclusive access to the Parallel System bus and to resources on the replying agent(s). This inhibits transfer operations between the replying agent and any other bus interface.

memory space. The address space used for accessing physical memory devices for storage and retrieval of code and data.

memory-mode agent. An agent that communicates with others by using memory and/or I/O space on the Parallel System bus.

memory-mode system. A system in which the agents communicate with one another with data structures in memory and/or I/O space.

message space. The address space used for packet based communications ranging from interrupts to negotiated data movement. See: packet.

message-mode agent. An agent which exclusively uses message space for communication with other agents.

message-mode system. A system in which communication between agents is via blocks of data transmitted in the message space.

module. A basic functional unit within an agent.

nibble. A group of four adjacent bits operated on as a unit.

packet. A block of information which is transmitted within a single transfer operation in message space. See: transfer operation; message space.

parallel system bus. The signals, media and protocol used to interconnect agents in the IEEE 1296 system (abbreviation "PSB").

parking. The state of the bus owner where, after the completion of the current transfer operation, ownership is retained until there is a request by another agent for the use of the bus.

power failure recovery. A sequence of events which provides orderly control of system shutdown during a temporary power failure and start-up after power is restored.

protocol. The set of signaling rules used to convey information between agents.

read data transfer. One or more data transfers from a replying agent to a bus owner, with uninterrupted bus ownership.

receiver. An agent that is the recipient of the data during a solicited message. See: solicited message.

recovery phase. The final phase of an exception operation in which the Parallel System bus is allowed to sit idle for a defined amount of time. See: exception operation.

reply phase. The final phase of a transfer operation which consists of one or more consecutive data and/or status transfers on the Parallel System bus.

replying agent. An agent that participates in a transfer operation with the bus owner.

request phase. The initial phase of a transfer operation in which the bus owner places command and address information on the Parallel System bus.

requesting agent. An agent that has entered arbitration for bus access. See: arbitration operation.

resolution phase. The initial phase of an arbitration operation in which all agents requesting access to the bus drive an arbitration ID onto the Parallel System bus. Agents mutually resolve requests and allow the agent with the highest priority to gain access to the bus. See: arbitration operation.

sender. The agent that supplies the data for a solicited message. See: solicited message.

sequential transfer. A transfer operation with multiple data transfers during the reply phase. See: transfer operation: reply phase.

server. An agent that performs a service for clients. See: client.

signal phase. The initial phase of an exception operation in which all agents are notified of an error condition. See: exception operation.

skew. The time difference between signals because of timing differences for logic and backplane delays.

solicited messages. A negotiated data transfer in message space. See: data transfer; message space.

status transfer. The passing of information over the System Control signal group, between the bus owner and the replying agent, during the reply phase of a transfer operation. See: agent status.

system control signal group. A set of ten (10) signals, including two parity bits, which supply command and status information between the bus owner and the replying agent.

transfer operation. The bus operation in which a bus owner transfers data on the Parallel System bus. See: bus operation; bus owner.

warm-start. A sequence of events performed to reset a running system.

write data transfer. One or more data transfers from the bus owner to a replying agent(s), with uninterrupted bus ownership.

3. Guide to Notation

3.1 General. A consistent convention on all signals, figures, and state-flow diagrams is used throughout this document. The conventions used are described in the following sections.

3.2 Signal Notation. Throughout this standard, the following convention is used for signal notation. An asterisk (*) following a signal name indicates that the signal is active when it is at a low voltage level (as described in Part II, Chapter 4). A signal name that is not followed by an asterisk indicates that the signal is active when it is at a high voltage level. Additionally, a high signal voltage level may be indicated by an uppercase h (H), and a low signal voltage level may be indicated by an uppercase l (L). This convention is summarized in Table 3.2-1.

Table 3.2-1
Signal Notation

Signal Name	Electrical Level	Signal Notation
BREQ	Low or L	Inactive, deasserted, negated
	High or H	Active, asserted
BREQ*	Low or L	Active, asserted
	High or H	Inactive, deasserted, negated

Many signals on the Parallel System Bus (PSB) are more easily or conveniently discussed as a group. Names for these signals follow a decimal radix numbering convention. When discussed as an individual signal, the decimal number is simply appended to the signal name, e.g., AD15*. A group of signal lines with the same

signal group name are collectively referred to by listing the group name and enclosing the decimal numbers within brackets, e.g., $AD\langle 31,23,7,0\rangle^*$. A range of consecutive signals is referred to by using a double period to list the beginning and ending signals, inclusively, e.g., $AD\langle 31..24\rangle^*$. Consecutive and disjoint signals are referred to by using both methods, e.g., $SC\langle 9,3..0\rangle^*$. A bus signal group name with no brackets and digits afterward refers to the entire group, e.g., AD^* is equivalent to $AD\langle 31..0\rangle^*$.

Signal lines on the PSB are always shown in capital letters, e.g., $BREQ^*$. In the case of open-collector drivers, where more than one agent may drive the line at any time, the signal level driven by an agent may not be the logic level currently on the PSB signal line. The signal an agent drives on the corresponding bus line is designated by the same signal name, but in lower case letters, e.g., $breq^*$.

3.3 Figure Notation. An example of the notation used in figures is shown in Fig 3.3-1. All solid vertical lines in the figure represent the sampling point for signals (the falling edge of the $BCLK^*$ signal). Dashed vertical lines indicate phase transitions. Groups of signals used for phases and transfers are enclosed in polygons indicating simultaneous high and low logic levels. All bus operations (arbitration, transfer, and exception) are enclosed in rectangles. Curved lines and arrows indicate cause-and-effect relationships, not asynchronous operation. Double slashed lines in the timing diagrams (//) indicate the occurrence of an indefinite number of bus clock cycles, unless otherwise noted.

3.4 Notation in State-Flow Diagrams. The state-flow diagrams use a consistent notation system in describing the transitions between states during an operation.

Within each diagram there are various components. Each component describes, either graphically or in words, some aspect of the operation of an agent. The various states that an agent may assume are represented as circles. Transitions to other states are shown as arrowed lines and are labeled with a number that corresponds to the numbered paragraph that defines their function. Bold type is used to name conditions that are derived from bus signals or name conditions that describe the internal status of an agent. Generally, such conditions are associated with transitions occurring from one state to another. A condition that causes a transition is shown next to the arrowed line for that transition. Equals signs (=) are used to indicate the state of individual lines or the components of a condition. The state equations and conditions use "AND" to indicate when the AND operator is required and "OR" to indicate when the OR operator is used.

3.5 Notation for Multiple Bit Data Representation. Data values for spanning multiple bits are represented in a consistent manner. The value for the multiple bit

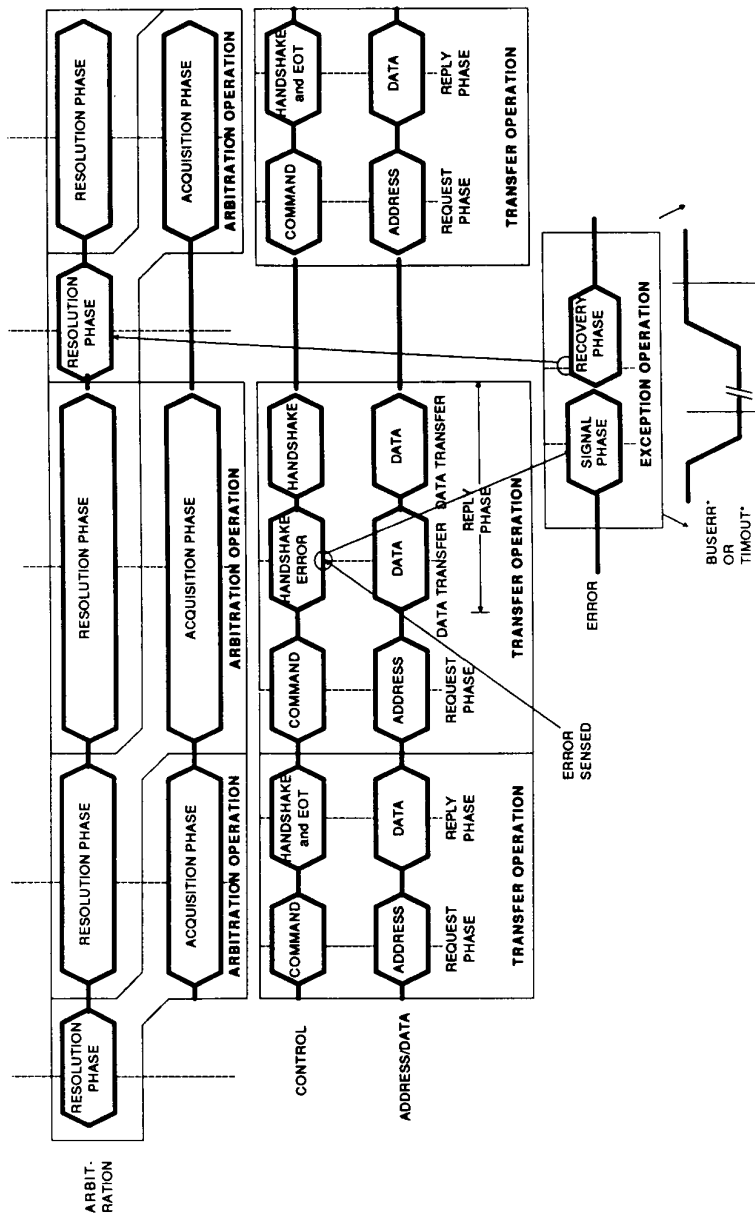


Fig 3.3-1
Figure Notation Example

quantity is shown as a decimal number, except where noted. The value is shown as high-true, making it independent of bus level inversions. For byte values, the highest numbered bit is the most significant bit of the byte. This standard does define the format of multiple data byte specification, except when noted.

If the data is in ASCII format, the data consists of eight consecutive bits, with the value of the eighth (most significant bit) always 0.

If the data is BCD format, each nibble within the byte represents a decimal digit. The value of bits <3..0> is used for the units digit of the number and the value of bits <7..4> is used for the tens digit of the number.

PART II — IEEE 1296 PSB Standard

4. PSB Overview

4.1 General. The PSB is a general purpose interface for multiple agents. Figure 4.1-1 shows a functional diagram of the PSB. As shown, the PSB consists of five signals groups that requesting and replying agents use to communicate with one another.

The PSB has several specific attributes, as follows:

(1) The address/data path is 32-bit plus four parity bits, providing a 4-gigabyte address range and capable of 8-, 16-, 24-, or 32-bit transfers. Ten system control signals, including nibble parity, provide command, status, and handshake information for the address/data path.

(2) The message-passing facility provides interagent communication.

(3) The interconnect facility allows configuration of agents and modules.

(4) The bus operations are synchronous; the bus uses a handshaking protocol that is synchronous with the basic clock rate (10 megahertz) for the system resulting in a 40 megabyte/second maximum transfer rate.

(5) The bus operates in defined bus operations. The three types of bus operations are arbitration operation, transfer operation, and exception operation.

(6) The bus uses a Central Services Module to provide some centralized system functions for up to 21 agents.

Each of these attributes is explained further in the following paragraphs.

4.2 Address/Data Path and System Control Signals. The address/data path on the PSB consists of 36 signal lines, including four parity lines, that are time-multiplexed. During the request phase of a transfer operation, the signal lines provide address information; during the reply phase of a transfer operation, the signal lines contain data. Byte parity is provided for all address and data transfers on the address/data path.

Ten System Control signals, including two parity signals, provide command, status and handshake information for the address/data path. These signals are also time-multiplexed. During the request phase of a transfer operation, the System

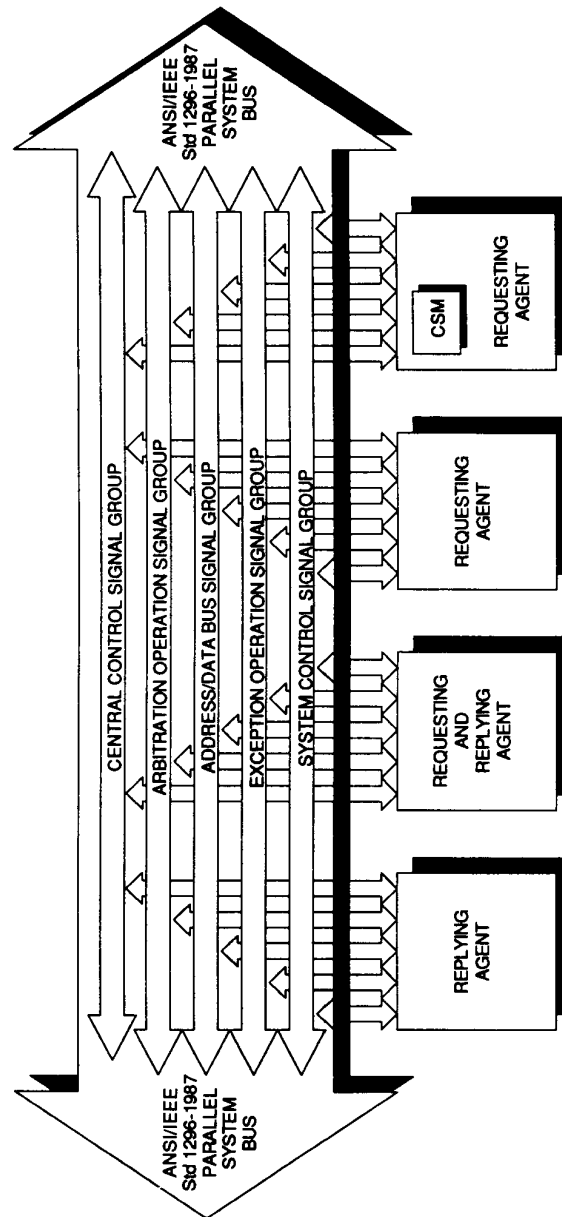


Fig 4.1-1
Block Diagram of the PSB Interface

Control signals provide command information; during the reply phase of a transfer operation, the signals are used for status and handshake information. Nibble parity is provided for the System Control lines at all times.

4.3 Message Passing Facility. The PSB defines a dedicated address space for use by agents in passing messages. The message passing facility provides a standardized method for performing direct transfers of messages (command and data) from one agent to another. Message passing on the PSB provides for interagent communications (interrupts and data movement) through the use of the dedicated message space.

4.4 Interconnect Facility. The interconnect facility within the PSB defines geographic addressing for ease of configuration, initialization and diagnosis of an agent on the PSB. This allows software to identify, diagnose, configure, and initialize an agent within a system.

4.5 Synchronous Operation of the PSB. The PSB is referred to as being “synchronous” in that all operations on the PSB occur relative to the active edge of a bus clock that is distributed to all agents on the bus. The synchronous nature of the bus does not place rigid time constraints on the length of a transfer operation. It requires that agents sample and drive signals at a specific time with respect to the bus clock.

The synchronous nature of the PSB allows each operation on the bus interface to be defined as a sequence of bus states. Agents on the bus proceed from one bus state to the next by monitoring and asserting specific status and control signals in a state control sequence defined by the operation. The specific meaning or function of some status or control signals can vary and is dependent on the bus operation and state.

Bus operations and associated state sequences are diagrammed later in this standard.

4.6 Bus Operations on the PSB. Agents perform three types of bus operations on the PSB: an arbitration operation, a transfer operation, or an exception operation. Each operation has a definite event that signals the start and end of the operation. The three types of bus operations can occur simultaneously, maximizing bus utilization.

A typical read or write sequence on the PSB consists of a series of two operations. First, the requesting agent participates in the arbitration operation for access to the bus. When the requesting agent becomes the bus owner, it performs one or more transfer operations to read or write data. If any agent senses an exception

(error) on the bus, that agent places an exception indication on the bus. The exception indication initiates an exception operation, which terminates the current arbitration and transfer operations.

Figure 4.6-1 shows how the two types of operations form a typical read or write sequence on the bus. The arbitration operation is a time period in which agents decide which will be the next to control the bus; the transfer operation is a subsequent time period in which that agent performs the address, command, data, and status transfers. Note that an exception operation is an error reporting time period that occurs only when an error is sensed.

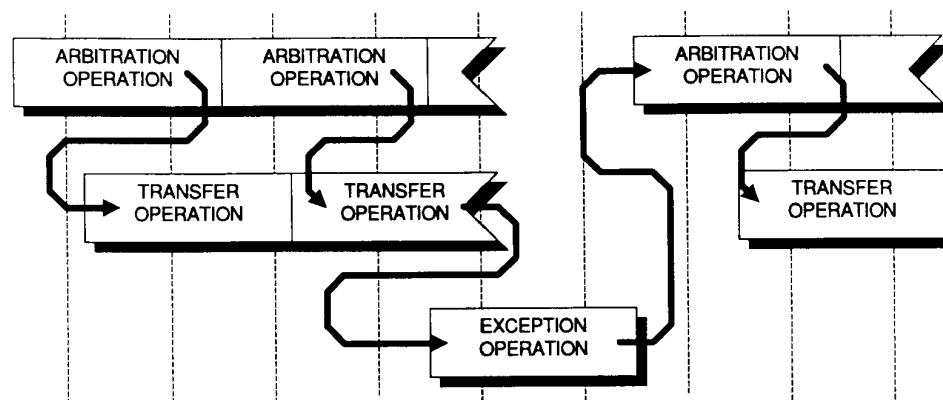
Each of the three types of bus operations is described further in the following paragraphs.

4.6.1 Arbitration Operation. The arbitration operation is the bus operation in which agents attempt to gain exclusive access to the PSB. For each agent that requests access, the arbitration operation includes two phases, the resolution phase and the acquisition phase.

The resolution phase is the time period during which agents that desire the bus collectively arbitrate for access rights to the bus. The agents decide among themselves which one of them is going to control the bus on completion of the current transfer operation.

The arbitration method used during the resolution phase for the PSB is referred to as a parallel contention or self-selecting method. In this method, each agent on

Fig 4.6-1
Bus Operation Relationships



the bus is assigned an encoded bit pattern that gives it a priority level when it makes a request for bus use.

The agent that has the highest priority request during the resolution phase obtains access rights to the bus and begins the acquisition phase while the remaining agents begin the resolution phase of the next arbitration operation. The agent in the acquisition phase is referred to as the bus owner.

Once in the acquisition phase, the bus owner begins its transfer operation while the other requesting agents conduct another resolution phase (resolving for next access rights). Figure 4.6-2 shows the end-of-a-transfer operation (EOT) initiating an acquisition phase of the arbitration operation to allow the next bus owner to acquire control of the bus.

The arbitration scheme is such that no agent is prevented from gaining access to the bus. Agents that enter arbitration enter collectively as a group, and each agent within the group receives bus access before any new agent with normal priority is allowed to participate in the bus arbitration process. A bus request sequence is a period consisting of at least one arbitration operation in which all requesting agents (at least one) within a group arbitrate for and receive bus access. When all agents within the group have received bus access, a new bus request sequence is entered and a new group of requesting agents begin arbitration for bus access. An agent may enter the bus request sequence if it has a high priority request as defined in section 6.2.2.2.

Figure 4.6-2 shows bus operations that include three consecutive, errorless transfer operations on the bus. This type of operation sequence is typical of a system with more than one requesting agent. The first operation consists of a single data transfer, the second consists of three data transfers, and the third consists of one data transfer. The diagram shows the relationship between the arbitration operation and the transfer operation and shows the two phases of each arbitration operation: the resolution phase and the acquisition phase.

Note that the resolution phase of the next arbitration operation overlaps the acquisition phase of the current arbitration operation.

4.6.2 Transfer Operation. After arbitrating for and winning control of the bus, the bus owner performs transfer operations to move data to or from another agent.

Figure 4.6-2 shows three consecutive transfer operations on the bus. The diagram shows the two phases of the transfer operation, the request phase and the reply phase, and shows the transfers within each phase, including the command, the address, the data, the handshake, and the EOT.

The request phase is controlled by the bus owner. During the request phase, the bus owner places address and control information onto the bus. The address and control information defines the replying agent(s), the type of operation, and the type of address space involved in the transfer operation.

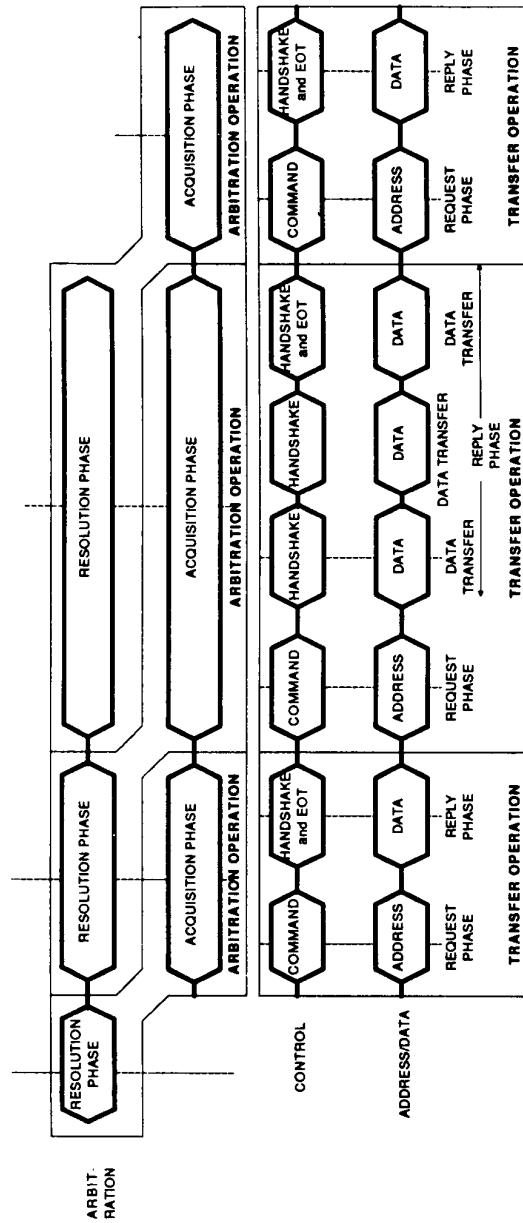


Fig 4.6-2
Block Diagram of Bus Operations

After the bus owner transmits the address and control information, the reply phase of the transfer operation begins, in which the replying agent(s) satisfies the request.

During the reply phase, the bus owner and replying agent engage in a handshake that synchronizes the data transfer. In the case of a read transfer, the handshake status signifies that the replying agent has placed the data on the bus and the bus owner has received it. In the case of a write transfer, the handshake status signifies that the bus owner has placed the data on the bus and that the replying agent has received it. The bus owner and replying agent may perform one or more data transfers in a reply phase. The final data transfer is accompanied by an end-of-transfer operation (EOT) indication. With the EOT, the bus owner releases ownership of the bus if other agents request access to the bus. Otherwise, the bus owner retains ownership of the bus.

4.6.3 Exception Operation. The exception operation is an error-reporting and recovery tool. If an agent detects an error (exception), that agent provides an exception indication on the bus and all agents begin an exception operation. The exception operation terminates any arbitration and transfer operations.

Figure 4.6-3 shows the same operation sequence presented in Fig 4.6-2, except that in Fig 4.6-3 an agent detects and signals an exception, thus initiating an exception operation. (Any agent can detect and signal an exception.) The diagram shows the two phases of the exception operation, the signal phase and the recovery phase, and shows how the exception operation terminates transfer and arbitration operations.

During the signal phase, the agent that detected the exception condition indicates the problem to all agents on the bus by asserting an exception line. As a result of the signal phase, all agents terminate any arbitration and transfer operation that may be in progress.

After the exception line is negated by the agent that signalled the error, the recovery phase begins. One bus clock cycle following the negation of the exception line, arbitration operations may begin. Four bus clock cycles following the negation of the exception line, transfer operations may begin.

As Fig 4.6-3 shows, the exception operation has a specific time relationship with respect to sensing an exception in the transfer operation. That is, the current exception operation reflects an exception condition sensed during the previous transfer operation.

4.7 Central Services Module. The PSB uses a single Central Service Module (CSM) to provide central control of the bus functions. The CSM is responsible for clock and time out generation, monitoring the state of the power supplies and initialization of the bus interface logic. One and only one CSM is active in the PSB backplane.

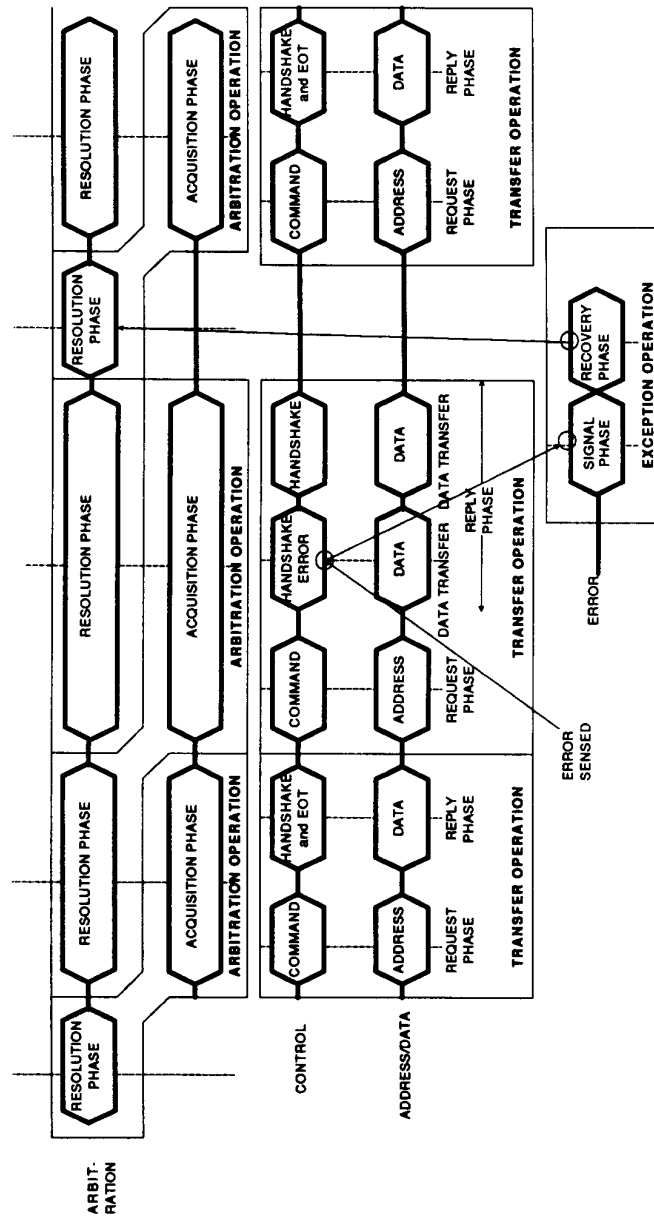


Fig 4.6-3
Block Diagram of Bus Operations with Errors

5. Signal Descriptions

5.1 General. This chapter of the standard lists and defines the five signal groups that make up the PSB. Each signal is discussed in detail.

5.2 Signal Groups. The PSB contains five groups of signals over which requesting and replying agents enact the bus protocol. All signals are sampled and driven relative to the falling edge of the bus clock unless otherwise noted. The signal groups, signal names and their functions are listed in Table 5.2-1. Each signal is discussed in the following text. Refer to section 7.5.1 for information regarding physical signal position on the connector.

Table 5.2-1
Signal Groups on the PSB

Signal Group Name	Signals in Group	Number of Signals	Description
Arbitration Operation signal group	BREQ*, ARB*	7	Provides the signals to request the bus and resolve arbitration for the next bus owner.
Address/Data Bus signal group	AD*, PAR*	36	Provides the address, data, and parity signals for read and write sequences.
System Control signal group	SC*	10	Provides the control signals needed to transfer address and data on the address/data signal group.
Exception Operation signal group	TIMOUT*, BUSERR*	2	Provides the error indication signals to stop transfer operations.
Central Control signal group	RST*, RSTNC*, DCLOW*, PROT*, BCLK*, CCLK*, LACHn*	7	Provides system-wide services such as reset and initialization control.

Refer to Section 7 for electrical specifications of the signals.

5.2.1 Arbitration Operation Signal Group. The arbitration signals on the PSB provide for the requesting, arbitrating, and granting of bus ownership to bus agents. All agents that require access to bus resources must arbitrate for use of the bus.

The seven signals within the Arbitration Operation signal group that implement the arbitration operation protocol on the PSB are Bus Request (BREQ*) and the arbitration ID signals ARB<5..0>*. These signals are described in the following paragraphs.

5.2.1.1 BREQ* (Bus Request). BREQ*, when asserted, indicates that a bus request sequence is in progress. Any bus agent that requires access to the bus asserts its breq*. All breq* signals are ORed to become the BREQ* signal. Each requesting agent negates breq* as it receives bus access. As the last requesting agent in the current bus request operation is granted the bus, it negates breq*, which causes the negation of BREQ* on the bus. The negation of BREQ* indicates to all bus agents that the bus request sequence has ended. BREQ* is used only in arbitration operations; once an agent has ownership of the bus, it negates breq* and performs the transfer operation. Other agents wishing access to the bus must sample BREQ* to determine if arbitration is already underway. If arbitration has not begun, an agent may assert its breq* and enter arbitration.

5.2.1.2 ARB* (Arbitration). The arbitration signals are ARB<5..0>*. The arbitration signals provide three functions:

- (1) carry cardslot ID assignment during reset
- (2) carry arbitration ID assignment during reset
- (3) arbitration

During reset, the CSM uses the ARB* lines to assign the agent in each cardslot a cardslot ID and an arbitration ID. The CSM drives the cardslot ID when ARB5* is asserted and the arbitration ID when ARB5* is negated. The cardslot and arbitration IDs are placed on ARB<4..0>*. The cardslot ID gives the agent a geographical address that is used in addressing interconnect space. The arbitration ID defines the arbitration priority level for the agent. During reset, all agents are required to latch their cardslot ID and requesting agents are required to latch their arbitration ID from the ARB<4..0>* lines.

The ARB* lines are used to resolve priorities for agents requesting access to the bus. During the resolution phase, a requesting agent places its arbitration ID on its arb* lines. Then, the agent with an arbitration ID that matches the ORed arbitration code present on ARB* gains access to the bus. As an agent gains bus access, it stops asserting its arb* and breq* lines and leaves arbitration to perform transfer operations as the bus owner. The remaining agents in the bus request sequence successively gain access to the bus in the same fashion as described.

An agent uses the ARB5* line to request high-priority access to the bus during arbitration. If two or more agents assert ARB5* at the same time, their bus access is determined by the priority of their arbitration IDs on ARB<4..0>*.

5.2.2 Address/Data Bus Signal Group. Only the bus owner and the selected replying agent(s) use the multiplexed address/data bus of the PSB. The Address/Data Bus signal group includes two sets of signals: address/data signals and parity signals. The address/data signals and the parity signals of the Address/Data Bus signal group are described next.

5.2.2.1 AD* (Address/Data Bus). The address/data signals on the PSB are AD<31..0>*. These 32 signals are multiplexed and serve a dual purpose depending on the phase of the transfer operation.

During the request phase of the transfer operation, AD* signals contain the address for the ensuing data transfer (AD0* is the least significant and AD31* is the most significant address bit). This address corresponds to a location in memory or I/O space, a processing agent in message space, or a cardslot/register location in interconnect space. The bus owner drives these lines during the request phase.

During the reply phase of the transfer operation, AD* signals provide for a variable-width data transfer, carrying 8, 16, 24, or 32 bits of data. The replying agent drives the AD* lines during the reply phase of a read transfer. The bus owner drives the AD* lines during the reply phase of a write transfer.

5.2.2.2 PAR* (Parity). The parity signals on the Address/Data Bus signal group are PAR<3..0>*. The parity signals help ensure the integrity of address and data information transferred on the AD* lines. Each parity signal carries an even-parity bit for one byte of a 4-byte (32-bit) address/data bus. An agent that drives less than 4 bytes need not place parity for the unused bytes onto the bus. The parity signals are assigned to bytes as follows:

- PAR0* — parity for AD<7..0>*
- PAR1* — parity for AD<15..8>*
- PAR2* — parity for AD<23..16>*
- PAR3* — parity for AD<31..24>*

Even parity means that when the number of asserted bits within a byte on the bus (not including the parity bit) is even, the corresponding parity signal is negated. If the number is odd, the parity signal is asserted.

5.2.3 System Control Signal Group. The System Control signal group on the PSB consists of ten signals, SC<9..0>*, that provide control between agents during transfer operations. Agents use the SC* signals to define commands or report status, depending on the phase of the transfer operation.

During the request phase of a transfer operation, the bus owner drives the SC* signals providing command information to the replying agent(s).

During the reply phase of a transfer operation (not broadcast messages), the bus owner drives the $SC\langle 9,3..0\rangle^*$ signals and the replying agent drives the $SC\langle 8..4\rangle^*$ signals. The SC^* signals provide for handshake and the transfer of status information between the bus owner and the replying agent(s). During the reply phase of broadcast messages, the bus owner continues to drive all SC^* signals.

The following paragraphs describe the functions of each SC^* signal during the request and reply phases. Table 5.2-2 summarizes the functions of the SC^* signals. Tables 5.2-3 and 5.2-4 list the request phase and reply phase functions for each of the SC^* signals.

Table 5.2-2
Summary of Functions of SC^* Signals

Signal	Function During Request Phase	Function During Reply Phase
SC0*	Request Phase	Reply Phase
SC1*	LOCK	LOCK
SC2*	Data Width	End-of-Transfer
SC3*	Data Width	Bus Owner Ready
SC4*	Address Space	Replying Agent Ready
SC5*	Address Space	Agent Status
SC6*	Read/Write Data Transfer	Agent Status
SC7*	Reserved	Agent Status
SC8*	Even Parity on $SC\langle 7..4\rangle^*$	Even Parity on $SC\langle 7..4\rangle^*$
SC9*	Even Parity on $SC\langle 3..0\rangle^*$	Even Parity on $SC\langle 3..0\rangle^*$

5.2.3.1 SC0* (Request Phase Or Reply Phase). SC0* is always driven by the bus owner and serves the same function during the request and reply phases of a transfer operation. When asserted, SC0* indicates to all agents that the bus owner is in the request phase of a transfer operation and that the address/data bus and the SC^* lines contain valid request phase information. When negated, SC0* indicates to all agents that the bus owner is in the reply phase of the transfer operation.

5.2.3.2 SC1* (Lock). SC1* is always driven by the bus owner and serves the same function during the request and reply phases of a transfer operation. When the bus owner asserts SC1*, pending bus requests are unheeded and the bus owner retains ownership of the bus. Agents in the resolution phase remain there until the current bus owner relinquishes the bus. Additionally, when SC1* is asserted, all multiported resources on the PSB that are accessed during the time that the bus owner asserted SC1* are locked from access by other devices. The locked resources are dedicated to the current bus owner until SC1* is negated.

5.2.3.3 SC2* (Data Width Or End-Of-Transfer Operation). SC2* changes functions depending on the phase of the transfer operation.

During the request phase, the bus owner uses SC2* (with SC3*) to tell the replying agent(s) the data-width format for the ensuing data transfer. The bus owner codes SC<3..2>* to select one of four possible data widths: 8-, 16-, 24-, or 32-bits. Table 5.2-3 shows the four codes and the associated data widths.

Table 5.2-3
Functions of SC* During REQUEST Phase

Signal	Driving Agent	Functions During Request Phase	Meaning															
SC0*	Bus Owner	Identify transition between request phase and reply phase of the transfer operation.	L=request phase															
SC1*	Bus Owner	Locked access	L=lock active H=lock inactive															
SC<3..2>*	Bus Owner	Identify the width of the data during the transfer operation.	<table><tr><td>SC3*</td><td>SC2*</td><td></td></tr><tr><td>H</td><td>H</td><td>= 8-bit transfers</td></tr><tr><td>H</td><td>L</td><td>= 16-bit transfers</td></tr><tr><td>L</td><td>H</td><td>= 24-bit transfers</td></tr><tr><td>L</td><td>L</td><td>= 32-bit transfers</td></tr></table>	SC3*	SC2*		H	H	= 8-bit transfers	H	L	= 16-bit transfers	L	H	= 24-bit transfers	L	L	= 32-bit transfers
SC3*	SC2*																	
H	H	= 8-bit transfers																
H	L	= 16-bit transfers																
L	H	= 24-bit transfers																
L	L	= 32-bit transfers																
SC<5..4>*	Bus Owner	Identify the address space for the transfer operation.	<table><tr><td>SC5*</td><td>SC4*</td><td></td></tr><tr><td>H</td><td>H</td><td>= memory space</td></tr><tr><td>H</td><td>L</td><td>= I/O space</td></tr><tr><td>L</td><td>H</td><td>= message space</td></tr><tr><td>L</td><td>L</td><td>= interconnect space</td></tr></table>	SC5*	SC4*		H	H	= memory space	H	L	= I/O space	L	H	= message space	L	L	= interconnect space
SC5*	SC4*																	
H	H	= memory space																
H	L	= I/O space																
L	H	= message space																
L	L	= interconnect space																
SC6*	Bus Owner	Identify the type of data transfer in the reply phase.	L=write data transfer H=read data transfer															
SC7*	Bus Owner	Not used; must be high.																
SC8*	Bus Owner	Provide even parity for SC<7..4>*:	L=odd number of lows on SC<7..4>* H=even number of lows on SC<7..4>*															
SC9*	Bus Owner	Provide even parity for SC<3..0>*:	L=odd number of lows on SC<3..0>* H=even number of lows on SC<3..0>*															

During the reply phase, the bus owner asserts SC2* to inform all agents that the current data transfer is the last in the transfer operation, the end-of-transfer (EOT).

5.2.3.4 SC3* (Data Width Or Bus Owner Ready). SC3* changes functions depending on the phase of the transfer operation.

During the request phase, the owner uses SC3* (with SC2*) to tell the replying agent(s) the data-width format for the ensuing data transfer. The bus owner drives SC<3..2>* to select one of four possible data widths: 8-, 16-, 24-, or 32-bits. Table 5.2-3 shows the encoding of each data width.

During the reply phase, the bus owner asserts SC3* to notify the replying agent when it is ready to send or receive data. SC3* and SC4* provide a two-directional handshake during the reply phase of the transfer operation.

5.2.3.5 SC4* (Address Space Or Replier Ready). SC4* changes functions depending on the phase of the transfer operation.

During the request phase, the bus owner drives SC4* (with SC5*) to select which address space will be used for the impending data transfer. For a read transfer, the address space is a source of data; for a write sequence, the address space is a destination. The bus owner has four address space options from which to choose: memory, I/O, message, or interconnect. Table 5.2-3 shows the four codes available using SC4* and SC5* and the associated address space selection.

During the reply phase, the replying agent asserts SC4* to indicate to the bus owner that it is ready to send or receive data. SC4* and SC3* provide a two-directional handshake during the reply phase of the transfer operation.

5.2.3.6 SC5* (Address Space Or Agent Status). SC5* changes functions depending on the phase of the transfer operation.

During the request phase, the bus owner drives SC5* and SC4* to select which address space will be used for the impending data transfer. For a read transfer, the address space is a source of data; for a write sequence, the address space is a destination. Table 5.2-3 shows the four codes available and the associated address space selection.

During the reply phase, SC5* is driven (with SC<7..6>*) by the replying agent to provide agent status information. Table 5.2-4 shows the assigned agent status codes.

5.2.3.7 SC6* (Read-Write Or Agent Status). SC6* changes functions depending on the phase of the transfer operation.

During the request phase, the bus owner uses SC6* to indicate the direction of data flow for the transfer. When the bus owner negates SC6*, the reply phase of the transfer operation is a read data transfer. When the bus owner asserts SC6*, the transfer operation is a write data transfer.

During the reply phase, SC6* is driven with SC<7,5>* by the replying agent to provide agent status information. Table 5.2-4 shows the assigned agent status codes.

5.2.3.8 SC7* (Reserved Or Agent Status). SC7* changes functions depending on the phase of the transfer operation.

During the request phase, SC7* is not used and must be negated by the bus owner.

Table 5.2-4
Functions of SC* During Reply Phase

Signal	Driving Agent	Functions During Reply Phase	Meaning
SC0*	Bus Owner	Identify transition between request phase and reply phase of the transfer operation.	H=not request phase
SC1*	Bus Owner	Locked access	L=lock active H=lock inactive
SC2*	Bus Owner	Place an end-of-transfer indication onto the bus.	L=end-of-transfer for transfer operation H=not end-of-transfer
SC3*	Bus Owner	Provide a bus owner ready indication on the bus (part of the reply phase handshake).	L=agent ready for data transfer operation H=agent not ready
SC4*	Replying Agent	Provide a replying-agent-ready indication on the bus (part of the reply phase handshake).	L=agent ready for data transfer operation H=agent not ready
SC<7..5>*	Replying Agent	Place an agent status indication onto the bus.	<u>SC7* SC6* SC5* Description</u> L L L Reserved. L L H Reserved. L H L Agent data error. L H H NACK; Replying agent cannot respond because of a temporarily unavailable resource. H L L Transfer not understood error; multiple errors sensed. H L H Continuation error; replying agent is unable to continue. H H L Width error; width of transfer is not compatible with replying agent. H H H No error; transfer completed without errors.
SC8*	Replying Agent	Provide even parity for SC<7..4>:	L=odd number of lows on SC<7..4>* H=even number of lows on SC<7..4>*
SC9*	Bus Owner	Provide even parity for SC<3..0>:	L=odd number of lows on SC<3..0>* H=even number of lows on SC<3..0>*

During the reply phase, SC7* is driven with SC<6..5>* by the replying agent to provide agent status information. Table 5.2-4 shows the assigned agent status codes.

5.2.3.9 SC8* (Parity On SC<7..4>*). The function of SC8* remains the same in both the request phase and the reply phase of the transfer operation. Agents use SC8* to provide even parity for the SC<7..4>* signals. If SC<7..4>* contain an odd number of asserted signals on the bus, then the agent driving SC<7..4>* must also assert SC8* on the bus.

5.2.3.10 SC9* (Parity On SC<3..0>*). The function of SC9* remains the same in both the request phase and the reply phase of the transfer operation. Agents use SC9* to provide even parity for the SC<3..0>* signals. If SC<3..0>* contain an odd number of asserted signals on the bus, then the agent driving SC<3..0>* must also assert SC9* on the bus.

5.2.4 Exception Operation Signal Group. The PSB provides the Exception Operation signal group for passing indications of exception errors to all agents. This group contains two signals: Bus Error (BUSERR*) and Bus Time Out (TIMOUT*). Each of these signals is on a dedicated line and are defined in the following sections.

5.2.4.1 BUSERR* Exception. An agent asserts Bus Error to indicate the detection of a data integrity problem during a transfer. Problems reported through the BUSERR* signal are: detection of a parity error on the valid bytes of the AD* lines or the SC<7..0>* lines, a bus owner reply phase termination, or a protocol violation occurring during the reply phase. BUSERR* is received by all agents and can be asserted by any agent capable of detecting transfer integrity problems.

5.2.4.2 TIMOUT* Exception. An agent monitors TIMOUT* to determine when a bus time out condition occurs. Bus time out on the PSB is a result of the CSM determining that an agent is taking too much time to assert a handshake signal on the bus. TIMOUT* may also be generated by the CSM after determining that the bus owner has maintained ownership for an excessive length of time, which is user defined. TIMOUT* is received by all active agents on the bus and is generated from time out circuitry located on the CSM.

5.2.5 Central Control Signal Group. The Central Control signal group provides bus status and control information for devices operating on the PSB. This signal group contains seven signals. They are:

- Reset (RST*)
- Reset Not Complete (RSTNC*)
- DC Power Low (DCLOW*)
- Protect (PROT*)
- Bus Clock (BCLK*)
- Central Clock (CCLK*)
- ID Latch (LACHn*)

Each of the signals is described in more detail in the following paragraphs. Electrical specifications of the signals are contained in Section 7.

5.2.5.1 Reset (RST*). The CSM asserts RST* as a system-level initialization signal to all agents. Agents may monitor RST* along with DC Power Low (DCLOW*) and Protect (PROT*) to distinguish between the types of reset: warm-start, cold-start or power failure recovery (see section 6.5.2).

5.2.5.2 Reset Not Complete (RSTNC*). Agents may assert RSTNC* during reset to extend the initialization time period provided by the CSM. RSTNC* is an OR-tied signal that is low when one or more agents on the PSB have not completed their reset requirements. Agents cannot perform bus operations while RSTNC* is asserted. The rstnc* signal must be driven by any agent that needs more initialization time than the RST* signal provides. The rstnc* signal should be negated by the agent as soon as possible, as no bus operations can be performed until the RSTNC* signal is negated. RSTNC* is received by all bus agents.

5.2.5.3 DC Power Low (DCLOW*). The CSM asserts DCLOW* to all agents as a warning of an imminent power failure. The CSM monitors the power level from the power supply and holds DCLOW* high during normal system operation. When the power supply identifies a power fail for the CSM, the CSM asserts DCLOW* to indicate to all agents that a power failure is imminent. Systems use DCLOW* as the control signal for enabling back-up power and storage. DCLOW* is asynchronous to the bus clock.

5.2.5.4 Protect (PROT*). The CSM asserts PROT* as a power-failure protection signal to other system agents. PROT* is a delayed result of the CSM sensing a power failure and indicates that system-supplied voltages are not within tolerance. Agents in the system that contain battery-backed resources receive PROT*. The CSM negates PROT* after AC line voltage is stable and system voltages are available. PROT* is asynchronous to the bus clock.

5.2.5.5 Bus Clock (BCLK*). Only the CSM generates BCLK* and all agents in a system receive BCLK*. An agent uses BCLK* to drive the PSB state machines. BCLK* has a fixed frequency of 10 MHz on the bus for normal operation, though the frequency can be reduced to DC for testing of the PSB. Agents should be designed to operate at BCLK* frequencies from 10 MHz to DC. The falling edge of BCLK* provides all system timing references.

5.2.5.6 Central Clock (CCLK*). CCLK* is generated by the CSM and may be used for additional timing among agents on the PSB. CCLK* operates with a specific relationship to BCLK* and at twice the frequency.

5.2.5.7 ID Latch (LACHn*). An agent uses its ID latch signal only during a reset. The LACHn* signal is driven by the CSM and serves two functions: it tells the agent when to latch the arbitration ID and when to latch the cardslot ID from the ARB<4..0>* lines, depending on the state of ARB5*.

The ID latch signal is called $LACH_n^*$ (where “n” is the cardslot to which the ID is assigned). Each of the cardslots, except 0, contains a $LACH_n^*$ signal that is activated when the CSM drives one of the address/data lines ($AD<20..1>^*$). For example, the agent in cardslot 2 accepts its ID when the CSM asserts $AD2^*$ which is connected on the backplane to $LACH_n^*$ at cardslot 2. This is shown in Fig 5.2-1.

5.2.6 Power. The PSB provides power for bus agents. The system power includes DC power at +5 volts, +12 volts, -12 volts, ground, and facilities for +5 volts battery back-up.

Fig 5.2-1
Backplane Connection of $LACH_n^*$

